BAB II
TINJAUAN PUSTAKA

2.1 Profil Tempat Kerja Praktek

PT. Electronic Solution merupakan salah satu perusahaan ritel modern di Indonesia. Saat ini PT. Electronic Solution Indonesia sedang menuju sebagai ritel electronik modern terbesar di Indonesia.

2.1.1 Visi PT Electronic Solution

kami berusaha untuk mengimbangi keinginan dan kebutuhan akan trend berbelanja baru yang dapat memenuhi ketersediaan waktu yang mulai terbatas, terutama untuk masyarakat kota besar,tapi juga untuk kota-kota lain di seluruh Indonesia,

2.1.2 Misi PT Electronic Solution

memperbanyak pilihan bagi pelanggan setia Electronic Solution dan diharapkan dapat memperluas kategori dan variasi produk yang tersedia saat ini.

2.1.3 Sejarah Instansi

PT. Electronic Solution mulai berdiri pada tahun 2005 dan merupakan anak perusahaan dari TT International Ltd Singapore. Jadi, PT. Electronic Solution merupakan PMA dari Singapura, yang sedang fokus menggarap ritel elektronik modern.

Sejak berdirinya pada tahun 2005 Electronic Solution menjadi salah satu perusahaan electronic terbesar dan memiliki outlet terbanyak yang tersebar di beberapa wilayah di Indonesia. seiring dengan perkembangannya yang terus berkembang dan membuka cabang diberbagai tempat di indonesia, maka pada

Tahun 2006 Electronic Solution membuka cabang kantor pemasaran di Bandung yang bertempat di Metro Indahmall (MIM), disinilah awal pembukaan kantor cabang yang berada di wilayah bandung,

Setelah Electronic Solution membuka cabang di metro indah mall (MIM) akhirnya Pada tahun 2007 Electronic Solution membuka cabang lagi yang bertempat di festifal city link yang terletak di jalan peta, disinilah Electronic Solution mendapat

respon yang tinggi dari konsumen, dengan meningkatnya penjualan pada 1 bulan pertama sejak mulai dibukanya toko di city link, dengan adanya respon tersebut dari konsumen maka Electronic Solution berkembang menjadi lebih baik.

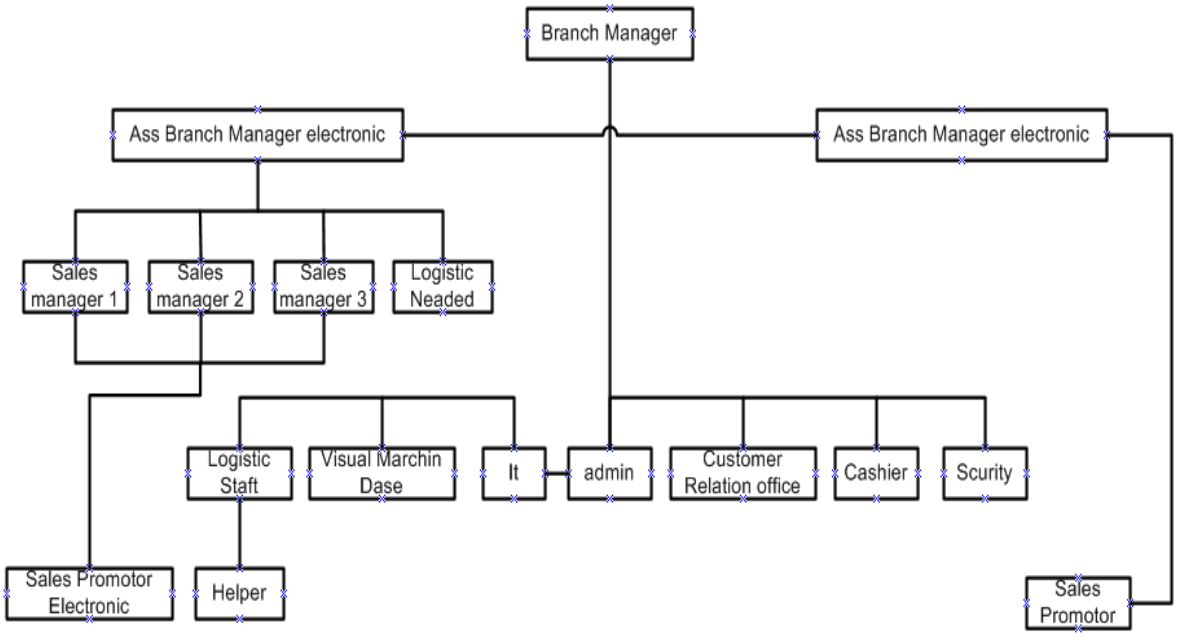
2.1.4 Logo

Logo PT. Electronic Solution



Gambar 2.1 Logo PT Electronic Solution

2.1.5 Struktur Pengurus



2.2 Landasan Teori

2.2.1 Sistem

2.2.1.1 Pengertian Sistem

Pengertian atau definisi sistem menurut penekanan pada prosedurnya adalah suatu jaringan dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatusasaran tertentu.

Sedangkan definisi sistem menurut penekanan pada komponennya adalah suatu kesatuan yang terdiri atas komponen atau elemen yang saling berinteraksi, saling terkait, atau saling bergantung membentuk keseluruhan yang kompleks.

2.2.1.2 Bentuk Umum Sistem

Bentuk umum dari suatu sistem terdiri atas masukan (input), proses, dan keluaran (output). Dalam bentuk umum sistem ini biasa melakukan satu atau lebih masukan yang akandiproses dan menghasilkan keluaran sesuai dengan rencana yang telah ditentukansebelumnya.

2.2.1.3 Karakteristik Sistem

a. Komponen / elemen (*component*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artiny saling bekerjasama untuk membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen system dapat berupa suatusubsistem atau bagian-bagian dari sistem. Subsistem tersebut mempunyai sifat-sifat yang dari siste itu sendiri dalam menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

b. Batas Sistem (*boundary*)

Batas sistem (Boundary) merupakan daerah yang membatasi antara satu sistem yanglain atau dengan linkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan, karena dengan batas sistem ini fungsi dan tugas dari subsistem yang satu dengan yang lainnya berbeda tetapi tetap saling berinteraksi.

c. Lingkungan Luar Sistem (*environment*)

Lingkungan luar sistem adalah segala sesuatu di luar dari batas sistem yang mempengaruhi operasi dari suatu sistem.

d. Perhubungan (*interface*)

Merupakan media penghubung antara satu subsistem dengan subsistem lain.

e. Masukan (*input*)

Energi yang dimasukkan ke dalam sistem disebut *input*, masukan ini dapat berupa Masukan Perawatan (*Maintenance Input*) yaitu energi yang dimasukkan supaya sistem itu dapat beroperasi.

f. Keluaran (*output*)

Keluaran adalah hasil dari energu yang diolah dan diklasifikasikan menjadi keluaran yang berguna.

g. Pengolahan (*process*)

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

h. Sasaran (*objective*) dan Tujuan Sistem (*goal*)

Setiap sistem memiliki tujuan ataupun sasaran yang mempengaruhi input yang dibutuhkan dan output yang akan dihasilkan. Dengan kata lain, suatu sistem akan dikatakan berhasil kalau pengoperasian sistem itu mengenai sasaran dan tujuannya.

2.2.1.4 Perancangan Sistem

Perancangan sistem adalah tahap setelah analisis sistem dan siklus pengembangansistem, pendefinisian kebutuhan-kebutuhan fungsional dan persiapan untuk merancangbangun implementasi, menggambarkan suatu sistem yang akan dibentuk. Perancangan sistemdapat menggunakan pemodelan secara terstruktur menggunakan grafik atau diagram.

Langkah-langkah yang dilakukan pada tahap-tahap perancangan sistem ini adalahmembuat usulan pemecahan masalah secara logika. Alat bantu yang digunakan adalah :

a. Flowmap

flowmap merupakan gambaran hubungan antara entity yangterlihat berupa aliran-aliran dokumen yang ada. Bagian alir dokumen merupakan baganalir yang menunjukan arus dari laporan dan formulir termasuk tembusannya.

b. Diagram Konteks

Diagram Konteks adalah diagram arus data yang berfungsi untukmenggambarkan yang dirancang disuatu objek, diagram konteks ini menggambarkansecara global atau menyeluruh dari suatu sistem informasi keterkaitan aliran-aliran dataantara

sistem dengan bagian-bagian luar. Kesatuan luar ini merupakan sumber arus datayang berhubungan dengan sistem informasi tersebut.

c. Kamus Data

Kamus data adalah merupakan faktor tentang data dan kebutuhan-kebutuhan informasi dari suatu informasi, kamus data yang dibuat berdasarkan arus data dari DFD.

d. Struktur Data

Struktur data merupakan hasil transformasi dari entitas-entitas pada diagram E-R.

e. Struktur Menu

Struktur menu adalah bentuk umum dari suatu rancangan program untuk memudahkan pemakai dalam menjalankan program komputer, sehingga saat menjalankan program user tidak mengalami kesulitan dalam memilih menu-menu yang diinginkan.

2.2.1.5 Arsitektur Sistem

Arsitektur sistem merujuk pada konfigurasi sistem secara keseluruhan yang akan menjadi "tempat hidup" dari DBMS, basis datadan aplikasi yang memanfaatkannya. Sudah seharusnya ditetapkan sejak awal sebelum memulai perancangan basis data, atau paling tidak sebelum melaksanakan tahap implementasi basis data.

2.2.2 Basis Data

Basis Data adalah kumpulan informasi yang disimpan didalam komputer secara sistematik sehingga dapat diperiksa menggunakan suatu program komputer unutk memperoleh informasi dari basis data tersebut.

2.2.3 Sistem Basis Data

Basis data hanyalahsebuah objek yang pasif atau mati. Ia ada karena ada pembuatnya. Iatidak akan pernah berguna jika tidak ada pengelola atau penggeraknya.Yang menjadi penggeraknya secara langsung adalah program atauaplikasi (*software*). Gabungan antara program / aplikasi dengan basis data menghasilakn sebuah sistem. Karena itu, secara umum sebuah sistem basis data merupakan sistemyang terdiri atas kumpulan file (table) yang saling berhubungan (dalam sebuah basis data di sebuah sistem komputer) dan sekumpulan program(DBMS) yang memungkinkan beberapa pemakai dan atau program lain untuk mengakses dan memanipulasi file – file (table-tabel) tersebut.

Lebih jauh lagi, dalam sebuah sistem basis data, secara lengkapakan terdapat komponen – komponen utama sebagai berikut :

a. Perangkat Keras (*Hardware*)

Perangkat keras yang biasanya terdapat dalam sebuah sistembasis data adalah :

1. Komputer (satu untuk sistem yang stand – alone atau lebih darisatu untuk sistem jaringan).
2. Memori sekunder yang on – line (harddisk).
3. Memori sekunder yang off – line (Tape atau Removable Disk)untuk keperluan backup data.
4. Media atau perangkat komunikasi (untuk sistem jaringan).

b. Sistem Operasi (*Operating System*)

Secara sederhana, sistem operasi merupakan program yangmengaktifkan atau memfungsikan sistem komputer, mengendalikanseluruh sumber daya (resource) dalam komputer dan melakukanoperasi – operasi dasar dalam computer (operasi I/O, pengelolaanfile, dll). Program pengelola basis data hanya dapat aktif (running)jika sistem operasi yang dikehendakinya atau sesuai telah aktif.

c. Basis Data (*Database*)

Sebuah sistem basis data dapat memiliki beberapa basis data.Setiap basis data dapat berisi sejumlah objek basis data seperti file /table, indeks, dll. Di samping berisi atau menyimpan data, setiapbasis data juga mengandung definisi struktur (baik untuk basis datamaupun objek – objeknya secara detail).

d. Sistem (aplikasi / perangkat lunak) Pengelola Basis Data (DBMS).

Pengelolaan basis data secara fisik tidak dilakukan olehpemakai secara langsung, tetapi ditangani oleh sebuah perangkatlunak yang khusus. Perangkat lunak ini disebut DBMS (DatabaseManagement Sistem) yang yang akan menentukan bagaimana datadiorganisasi, disimpan, diubah dan diambil kembali. Ia juga menerapkan mekanisme pengamanan data, pemakaian data secarabersama, pemaksaan keakuratan atau konsistensi data, dan sebagainya.

e. Pemakai (*user*)

Ada beberapa jenis pemakai terhadap suatu sistem basis data yang dibedakan berdasarkan cara mereka berinteraksi terhadap sistem :

1. Programmer Aplikasi

Pemakai yang berinteraksi dengan basis data melalui Data Manipulation Language (DML), yang disertakan(*embedded*) dalam program yang ditulis dalam bahasapemrograman induk (seperti C, Pascal, Cobol, dan lain lain).

2. User Mahir (*Casual User*)

   Pemakai yang berinteraksi dengan sistem tanpa menulismodul program. Mereka menyatakan query (untuk akses data dengan bahasa query yang disediakan oleh suatu DBMS.

3. User Umum (*End User*)

   Pemakai yang berinteraksi dengan sistem basis datamelalui pemanggilan satu program aplikasi permanen(executable program) yang telah ditulis atau disediakansebelumnya.

4. User Khusus (*Specialized User*)

   Pemakai yang menulis aplikasi basis data non-konvensional, tetapi untuk keperluan – keperluan khusus,seperti untuk aplikasi AI, Sistem Pakar, Pengolahan Citra, dll,yang bisa saja mengakses basis data dengan / tanpa DBMSyang bersangkutan.

f. Aplikasi (perangkat lunak) lain (bersifat opsional)

   Aplikasi (perangkat lunak) lain ini bersifat opsional. Artinya,ada atau tidaknya tergantung pada kebutuhan kita. DBMS yang kitagunakan lebih berperan salam pengorganisasian data dalam basisdata, sementara bagi pemakai basis data (khususnya yang menjadi *end user / naive user*) dapat dibutuhkan / disediakan programkhusus / lain untuk melakukan pengisian, pengubahan danpengambilan data. Pemrogram ini ada yang sudah disediakanbersama dengan DBMS – nya, ada juga yang harus dibuat sendiridengan menggunakan aplikasi lain yang khusus untuk itu (*delevopment tools*).

2.2.4 Model Data

Model data sendiri dapat didefinisikan sebagai kumpulanperangkat konseptual untuk menggambarkan data, hubungan data,semantic (makna) data dan batasan data. Oleh karena yang inginditunjukkan adalah makna dari data dan keterhubungannya dengan datalain, maka Model Data ini lebih tepat jika disebut Model Data Logik.Penyingkatannya menjadi Model Data dilakukan demi penyederhanaanpenulisan.

Ada sejumlah secara dalam meprestasikan Model Data dalamperancangan basis data, yang secara umum dapat dibagi dalam 2 (dua)kelompok yaitu, yaitu :

a. Model Logik Data Berdasarkan Objek (Objek – Based LogicalModels) yang terdiri dari :
    1. Entity – Relationship Model
    2. Object – Oriented Model
    3. Semantic Data model
    4. Functional Data model.
b. Model Lojik Data Berdasarkan Record (Record – Based LocicalModels) yang terdiri dari :
    1. Relational Model
    2. Hierarchical Model
    3. Network Model

2.2.4.1. ERD

ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi.

a. Entiti

Entiti merupakan objek yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain (Fathansyah, 1999: 30). Simboldari entiti ini biasanya digambarkan dengan persegi panjang.

b. Atribut

Setiap entitas pasti mempunyai elemen yang disebut atribut yangberfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. Isi dariatribut mempunyai sesuatu yang dapat mengidentifikasikan isi elemen satudengan yang lain. Gambar atribut diwakili oleh simbol elips.

c. Hubungan / Relasi

Hubungan antara sejumlah entitas yang berasal dari himpunanentitas yang berbeda. Relasi dapat digambarkan sebagai berikut :

Relasi yang terjadi diantara dua himpunan entitas (misalnya A dan B)dalam satu basis data (Abdul Kadir, 2002: 48) yaitu :

1. Satu ke satu (*One to one*)

    Hubungan relasi satu ke satu yaitu setiap entitas pada himpunanentitas A berhubungan paling banyak dengan satu entitas pada himpunanentitas B.

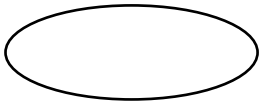
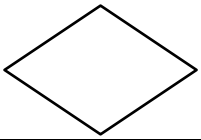
2. Satu ke banyak (*One to many*)

   Setiap entitas pada himpunan entitas A dapat berhubungan denganbanyak entitas pada himpunan entitas B, tetapi setiap entitas pada entitas Bdapat berhubungan dengan satu entitas pada himpunan entitas A.

3. Banyak ke banyak (*Many to many*)

   Setiap entitas pada himpunan entitas A dapat berhubungan denganbanyak entitas pada himpunan entitas B.

Tabel 2. 1 Notasi – Notasi Simbolik ERD (Entity RelationshipDiagram)

SIMBOL	ARTI
	Persegi panjang, ,menyatakan Himpunan entitas.
	Elips menyatakan Atribut (Atribut yang berfungsisebagai key digarisbawahi).
	Belah ketupat menyatakan Himpunan Relasi.
	Garis sebagai penghubung antara Himpunan Relasi dengan Himpunan Entitas dan HimpunanEntitas dengan atributnya.
ın 1 untukrelasi satu-ke-atu. ke N untuk relasi satu ke banyak e N untuk relasi banyak ke banyak	Kardinalitas Relasi dapat dinyatakan denganbanyaknya garis cabang atau dengan pemakaianangka.

2.2.4.2. Varian Entitas dan Relasi

Idealnya himpunan entitas yang terlibat dalam sebuah ERDadalah himpunan entitas kuat / bebas. Himpunan entitas demikian tidakmemiliki ketergantungan dengan himpunan entitas lainnya. Namundemikian, dalam pembentukan ERD kita tidak selalu dapat melibatkanhimpunan entitas seperti itu. Ada kalanya kita juga melibatkanhimpunan entitas yang lemah atau merupakan bagian dari himpunan entitas lainnya.

2.2.4.3. Himpunan Entitas Lemah (*Weak Entity Sets)*

Berisi entitas – entitas yang kemunculannya tergantung padaeksistensinya dalam sebuah relasi terhadap entitas lain. Himpunanentitas demikian biasanya tidak memiliki atribut yang dapat berfungsisebagai key, yang dapat menjamin keunikan entitas di dalamnya.

2.2.4.4. Sub Entitas (*Subtype Entity)*.

Merupakan himpunan entitas yang beranggotakan entitas–entitas yang merupakan bagian dari himpunan entitas yang lebihsuperior (utama). Sub Entitas ini merupakan hasil dekomposisi(spesialisasi) himpunan entitas berdasarkan pengelompokan tertentu.

2.2.5 SQL

SQL (Structured Query Language) adalah sebuah bahasa yang dipergunakanuntuk mengakses data dalam basis data relasional. Bahasa ini secara de factomerupakan bahasa standar yang digunakan dalam manajemen basis data relasional.Saat ini hampir semua server basis data yang ada mendukung bahasa ini untukmelakukan manajemen datanya.

Secara umum, SQL terdiri dari dua bahasa, yaitu Data Definition Language(DDL) dan Data Manipulation Language (DML). Implementasi DDL dan DMLberbeda untuk tiap sistem manajemen basis data (SMBD), namun secara umumimplementasi tiap bahasa ini memiliki bentuk standar yang ditetapkan ANSI.

2.2.6 DBMS (*Database Management System*)

DBMS (DataBase Management System) adalah sistem yang secarakhusus dibuat untuk memudahkan pemakai dalam mengelola basis data.Sistem ini dibuat untuk mengatasi kelemahan sistem pemrosesan yangberbasis berkas.Pada pendekatan yang berbasis berkas, umumnyaperancangan sistem didasarkan pada kebutuhan individual pemakai, bukanberdasarkan kebutuhan sejumlah pemakai. Setiap kali terdapat kebutuhanbaru dari seorang pemakai, kebutuhan segera diterjemahkan kedalamprogram komputer. Akibatnya, kemungkinan besar setiap program aplikasimenuliskan data tersendiri. Sementara itu ada kemungkinan data yangsama juga terdapat pada berkas-berkas lain yang digunakan oleh programaplikasi lain.

DBMS merupakan perantara bagi pemakai dengan basis data.Untuk merinteraksi dengan DBMS (basis data) menggunakan bahasa basisdata yang telah ditentukan oleh perusahaan DBMS. Bahasa basis databiasanya terdiri atas perintah-perintah yang di formulasikan sehinggaperintah tersebut akan diproses olah DBMS. Perintah-perintah biasanyaditentukan oleh user. Ada 2 bahasa basis data:

2.2.6.1 DDL (*Data Definition Language*)

DDL digunakan untuk mendefinisikan, mengubah, serta menghapus basis data danobjek-objek yang diperlukan dalam basis data, misalnya tabel, view, user, dansebagainya. Secara umum, DDL yang digunakan adalah CREATE untuk membuat objek baru, USE untuk menggunakan objek, ALTER untuk mengubah objek yangsudah ada, dan DROP untuk menghapus objek. DDL biasanya digunakan olehadministrator basis data dalam pembuatan sebuah aplikasi basis data.

CREATE

CREATE digunakan untuk membuat basis data maupun objek-objek basis data. SQL

yang umum digunakan adalah:

“CREATE DATABASE nama_database”

“CREATE TABLE nama_tabel”

2.2.6.2 DML (*Data Manipulation Language*)

DML digunakan untuk melakukan menipulasi dan pengambilandata pada suatu basis data seperti penambahan data baru ke dalam basisdata, menghapus data dari suatu basis data dan pengubahan data di suatubasis data. Dalam pembuatan DBMS diperlukan beberapa komponenfungsional penyusunnya sebagai berikut:

a. DML *Precompiler* yaitu mengkonversi pernyataan-pernyataanDML yang dimasukkan di dalam program aplikasi ke dalampemanggilan prosedur normal di dalam bahasa induknya.Procompiler harus berinteraksi dengan query processor untukmembuat kode-kode yang diperlukan.
b. *Query Processor* yaitu menterjemahkan pernyataan-pernyataanbahasa query ke dalam instruksi-instruksi *low-level* yangdimengerti oleh database *manager*.
c. DDL *Compiler* yaitu mengkonversi pernyataan DDL ke dalamsekumpulan tabel yang mengandung metadata atau "datamengenai data".
d. *Database Manager* yaitu menyediakan *interface* antara data*low-level* yang disimpan di dalam basis data dengan program-program aplikasi dan *queries* yang dikirimkan ke sistem.

Salah satu tujuan dari DBMS adalah memberikan tampilan kepada pengguna dalam hal menyampaikan data. Untuk itu dalam DBMS terdapatLevel Abstraksi Data. Level ini berguna untuk menyembunyikan detailatau kompleksitasnya basis data seperti bagaimana data disempan dandiolah. Sehingga pengguna hanya melihat tampilan yang dibutuhkan olehpengguna. Contoh dari DML adalah *select* * pelanggan *where name* = 'ade'.

2.2.6.3 Level Fisik

Level fisik merupakan level yang paling bawah. Pada level inimemperlihatkan bagaimana sesungguhnya data disimpan.

2.2.6.4 Level Konseptual

Level ini menggambarkan bagaimana sebenarnya basis datadisimpan dan berhubungan dengan data lainnya.

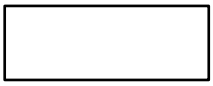
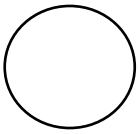
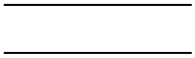
2.2.6.5 Level View

Level abstaraksi ini hanya menunjukkan sebagian dari basis data.Pada umumnya pengguna tidak melibatkan secara langsung sehinggapengguna hanya melihat data sesuai dengan yang dibutuhkan.

2.2.7 DFD(*Data Flow Diagram*)

Data Flow Diagram (DFD) adalah alat pembuatan model yangmemungkinkan profesional sistem untuk menggambarkan sistem sebagaisuatu jaringan proses fungsional yang dihubungkan satu sama laindengan alur data, baik secara manual maupun komputerisasi.

DFD ini merupakan alat perancangan sistem yang berorientasipada alur data dengan konsep dekomposisi dapat digunakan untukpenggambaran analisa maupun rancangan sistem yang mudahdikomunikasikan oleh profesional sistem kepada pemakai maupunpembuat program.

KOMPONEN	ARTI
	s luar
	s
	*Store*
	Data

2.2.7.1 Komponen Terminator / Entitas Luar

Terminator mewakili entitas eksternal yang berkomunikasi dengansistem yang sedang dikembangkan. Biasanya terminator dikenal dengannama entitas luar (external entity).

Komponen terminator ini perlu diberi nama sesuai dengan dunialuar yang berkomunikasi dengan sistem yang sedang dibuat modelnya,danbiasanya menggunakan kata benda, misalnya Admin.

2.2.7.2 Komponen Proses

Komponen proses menggambarkan bagian dari sistem yangmentransformasikan input menjadi output. Proses diberi nama untukmenjelaskan proses/kegiatan apa yang sedang/akan dilaksanakan.Pemberian nama proses dilakukan dengan menggunakan kata kerja*transitif* (kata kerja yang membutuhkan obyek).

2.2.7.3 Komponen Data Store

Komponen ini digunakan untuk membuat model sekumpulan paketdata dan diberi nama dengan kata benda jamak, misalnya Mahasiswa.Data store ini biasanya berkaitan dengan penyimpanan-penyimpanan, seperti file atau database yang berkaitan denganpenyimpanan secara komputerisasi, misalnya file disket, file harddisk, filepita magnetik. Data store juga berkaitan dengan penyimpanan secaramanual seperti buku alamat, file folder, dan agenda.Suatu data store dihubungkan dengan alur data hanya padakomponen proses, tidak dengan komponen DFD lainnya. Alur data yangmenghubungkan data store dengan suatu proses mempunyai pengertiansebagai berikut :

a. Alur data dari data store yang berarti sebagai pembacaan ataupengaksesan satu paket tunggal data, lebih dari satu paket data,sebagian dari satu paket tunggal data, atau sebagian dari lebih darisatu paket data untuk suatu proses.
b. Alur data ke data store yang berarti sebagai pengupdatean data, sepertimenambah satu paket data baru atau lebih, menghapus satu paket ataulebih, atau mengubah/memodifikasi satu paket data atau lebih.

2.2.7.4 Komponen *Data Flow* / Alur Data

Suatu data flow / alur data digambarkan dengan anak panah, yangmenunjukkan arah menuju ke dan keluar dari suatu proses. Alur data inidigunakan untuk menerangkan perpindahan data atau paket data/informasidari satu bagian sistem ke bagian lainnya. Alur data perlu diberi namasesuai dengan data/informasi yang dimaksud, biasanya pemberian namapada alur data dilakukan dengan menggunakan kata benda.

2.2.8 Desain Perangkat Lunak

Menurut *Whitten et al*(2004), Desain perangkat luank yaitu tahapan atau aktivitas yangdifokuskan pada spesifikasi detil dari solusi berbasis komputer. Disain perangkat lunak sering juga disebut sebagai physical design.Jika tahapan analisis sistem menekankan pada masalah bisnis (*businessrule*), maka sebaliknya disain perangkat lunak fokus pada sisi teknis danimplementasi sebuah perangkat lunak.

Output utama dari tahapan desain perangkat lunak adalah spesifikasi disain. Spesifikasi ini meliputi spesifikasi desain umum yangakan disampaikan kepada *stakeholder* sistem dan spesifikasi desain rinciyang akan digunakan pada tahap implementasi. Spesifikasi desain umumhanya berisi gambaran umum agar *stakeholder* sistem mengerti akanseperti apa perangkat lunak yang akan dibangun. Biasanya diagram USDtentang perangkat lunak yang baru merupakan point penting dibagian ini.Spesifikasi disain rinci atau kadang disebut desain arsitektur rinciperangkat lunak diperlukan untuk merancang sistem sehingga memilikikonstruksi yang baik, proses pengolahan data yang tepat dan akurat,bernilai, memiliki aspek user friendly dan memiliki dasar-dasar untukpengembangan selanjutnya.

Desain arsitektur ini terdiri dari desain database, desain proses,desain *user interface* yang mencakup desain *input, output form* danreport, desain hardware, software dan jaringan. Desain proses merupakankelanjutan dari pemodelan proses yang dilakukan pada tahapan analisis.

2.2.9 Pengujian (*Blackbox*)

Pengujian Perangkat Lunak adalah elemen kritis dari jaminankualitas perangkat lunak dan merepresentasikan kajian pokok darispesifikasi, desain, dan pengkodean.

Pentingnya pengujian perangkat lunak dan implikasinya yangmengacu pada kualitas perangkat lunak tidak dapat terlalu ditekan karenamelibatkan sederetan aktivitas produksi di mana peluang terjadinyakesalahan manusia sangat besar dan arena ketidak mampuan manusiauntukmelakukandanberkomunikasidengan sempurna makapengembangan perangkat lunak diiringi dengan aktivitas jaminan kualitas.

Meningkatnya visibilitas (kemampuan) perangkat lunak sebagaisuatu elemen sistem dan “biaya” yang muncul akibat kegagalan perangkatlunak, memotivasi dilakukannya perencanaan yang baik melalui pengujianyang teliti. Pada dasarnya, pengujian merupakan satu langkah dalamproses rekayasa perangkat lunak yang dapat dianggap sebagai hal yangmerusak dari pada membangun.

Sejumlah aturan yang berfungsi sebagai sasaran pengujian padaperangkat lunak adalah:

1. Pengujian adalah proses eksekusi suatu program dengan maksudmenemukan kesalahan.
2. *Test case* yang baik adalah *test case* yang memiliki probabilitas tinggiuntuk menemukan kesalahan yang belum pernah ditemukansebelumnya.
3. Pengujian yang sukses adalah pengujian yang mengungkap semuakesalahan yang belum pernah ditemukan sebelumnya.

Sasaran itu berlawanan dengan pandangan yang biasanya dipegangyang menyatakan bahwa pengujian yang berhasil adalah pengujian yangtidak ada kesalahan yang ditemukan. Data yang dikumpulkan pada saatpengujian dilakukan memberikan indikasi yang baik mengenai reliabilitasperangkat lunak dan beberapa menunjukkan kualitas perangkat lunaksecara keseluruhan, tetapi ada satu hal yang tidak dapat dilakukan olehpengujian,yaitu "*pengujian tidak dapat memperlihatkan tidak adanyacacat, pengujian hanya dapat memperlihatkan bahwa ada kesalahanperangkat lunak*".

Sebelum mengaplikasikan metode untuk mendesain test case yangefektif, perekayasa perangkat lunak harus memahami prinsip dasar yangmenuntun pengujian perangkat lunak, yaitu:

a. Semua pengujian harus dapat ditelusuri sampai ke persyaratanpelanggan, maksudnya mengungkap kesalahan dari cacat yangmenyebabkan program gagal.
b. Pengujian harus direncanakan lama sebelum pengujian itu mulai,maksudnya semua pengujian dapat direncanakan dan dirancangsebelum semua kode dijalankan.
c. Prinsip *Pareto* berlaku untuk pengujian perangkat lunak, maksudnyadari 80% kesalahan yang ditemukan selama pengujian dapat ditelusurisampai 20% dari semua modul program.
d. Pengujian harus mulai "dari yang kecil" dan berkembang ke pengujian"yang besar", Selagi pengujian berlangsung maju, pengujianmengubah fokus dalam usaha menemukan kesalahan pada *cluster*modul yang terintegrasi dan akhirnya pada sistem.
e. Pengujian yang mendalam tidak mungkin karena tidak mungkinmengeksekusi setiap kombinasi jalur skema pengujian

dikarenakanjumlah jalur permutasi untuk program menengah pun sangat besar.

f. Untuk menjadi paling efektif, pengujian harus dilakukan oleh pihakketiga yang *independent*.

Dalam lingkungan yang ideal, perekayasa perangkat lunakmendesain suatu program komputer, sebuah sistem atau produk dengantestabilitas dalam pikirannya. Hal ini memungkinkan individu yangberurusan dengan pengujian mendesain test case yang efektif secara lebihmudah. Testabilitas adalah seberapa mudah sebuah program computerdapat diuji. Karena sangat sulit, perlu diketahui apa yang dapat dilakukanuntukmembuatnyamenjadilebihmudah.Proceduraldanmenggunakannya sebagai pedoman untuk menetapkan basis set dari jalureksekusi.

2.2.10 Bahasa Pemrograman PHP

PHP (Personal Home Page) adalah server-side programming yang popular digunakan untuk membuat web-based application. PHP dikenal sebagai sebuah bahasa scripting yangmenyatu dengan tag-tag HTML yang diekseskusi di server dan digunakan untuk membuathalaman webyang dinamis seperti guestbook, statistik pengunjung, polling, email, dan masihbanyak lagi.

Adapun keunggulan yang dimiliki oleh bahasa pemrograman PHP adalah sebagaiberikut :

a. Life Cycle yang sangat singkat , sehingga PHP selalu up to date mengikutiperkembangan teknologi internet.
b. Cross Platform, yakni PHP dapat dipakai di hampir semua webserver yang ada dipasaran (terutama Apache dan Microsoft IIS) dan dijalankan pada berbagai sistemoperasi (Linux, Windows, FreeBSD).
c. PHP mendukung koneksi ke banyak database baik yang gratis maupun komersil,seperti MySQL, mSQL, Oracle, Microsoft SQL Server, Interbase, dan banyak lagi.
d. PHP bersifat open source dan gratis.
e. Kemudahan dalam mendapatkan dokumentasi. di Internet, kita tidak akan sulit untukmencari baik itu referensi, kode-kode PHP yang sudah jadi dan juga mengajukanpertanyaan pada grup-grup diskusi yang di dalamnya banyak sekali para master PHP.

2.2.11 Pengenalan MySQL

MySQL merupakan sebuah software yang berguna sebagai suatu Database Serveryang cukup terkenal. Kepopulerannya seiring dengan penggunaan script PHP untuk webprogramming. Database server itu sendiri merupakan suatu software yang bertugas untukmelayani permintaan (request) query dari client.

MySQL sebagai suatu database server mempunyai beberapa kemampuan, salahsatunya harus menyediakan suatu sistem manajemen database yang dapat mengaturbagaimana menyimpan, menambah, mengakses data dan transaksi-transaksi database lainnya.MySQL cepat sekali berkembang, karena MySQL merupakan suatu software yang OpenSource.

2.2.12 Macromedia Dreamweaver 8

Salah satu *software* web editor adalah Macromedia Dreamweaver 8 yang merupakan penyempurnaan dari versi sebelumnya dan tentu saja semakin mudah dalam penggunaannya. Oleh karena itu, *software* ini paling inovatif dan lebih lengkap dibandingkan *software* web editor lain. Adapun pengertian dari Macromedia Dreamweaver 8 ini adalah program aplikasi professional untuk mengedit HTML secara visual. Program Aplikasi Macromedia Dreamweaver 8 menyertakan banyak perangkat yang berkaitan dengan pengkodean dan fitur seperti HTML, CSS, serta *JavaScript*.

Fasilitas terbaru dari Macromedia Dreamweaver 8 adalah *Zoom Tool and Guides*, Panel CSS yang baru, *Code Collapse*, *Coding Toolbar*, dan *Insert Flash Video*. Macromedia Dreamweaver 8 mendukung pemrograman *script server-side*, seperti PHP, ASP, ASP.NET, *ColdFusion* dan JSP. Pemrograman *script server-side* maksudnya adalah *script* yang digunakan dalam pemrograman web dinamis dimana semua perintahnya dieksekusi pada *server*. Fungsi *server* disini adalah sebagai pemroses *script* dan hasilnya dikembalikan dalam bentuk tag-tag HTML yang kemudian ditampilkan dalam *browser*.

2.2.13 Pengenalan Wamserver

Wamp adalah sebuah aplikasi yang dapat menjadikan komputer kita menjadi sebuah server. Kegunaan wamp server ini untuk membuat jaringan local sendiri dalam artian kita dapat membuat website secara offline untuk masa coba-coba di komputer sendiri. Jadi fungsi dari wamp server itu sendiri merupakan server website kita untuk

cara memakainya. Mengapa harus menjadi server ? Karena dalam hal ini komputer yang akan kita pakai harus memberikan pelayanan untuk pengaksesan web, untuk itu komputer kita harus menjadi server.

Biasanya para perancang web atau web master jika akan merencanakan (planing), kemudian membangun (buliding)dilakukan di komputer local atau bisa juga di jaringan local, tidak langsung di host --> internet. Oleh karena itu perlu dikomputer kita di jadikan server sehingga kita seolah olah sedang meng update di hostnya (tempat penyimpanan file2 yang diperlukan website) ---> internet.

Dengan di tempatkannya file file pendukung website di komputer kita kita tidak perlu buka masuk internet .. sehingga hal ini mengurangi presentasi waktu dan biaya, yang kalo di lakukan langsung di jaringan internet akan memakan waktu yang lama juga untuk uploadnya (edit file file websitenya seperti html, php atau asp).

2.2.14 Mozilla Firefox

Mozilla Firefox (aslinya bernama Phoenix dan kemudian untuk sesaat dikenal sebagai Mozilla Firebird) adalah penjelajah *web* antar-*flatform* gratis yang dikembangkan oleh yayasan mozilla. Yayasan mozilla bertujuan untuk mengembangkan sebuah *browser web* yang kecil, cepat, dan simpel melalui firefox. Salah satu fitur populer firefox adalah pemblokir *pop-up* yang sudah terpasang di dalamnya dan sebuah mekanisme pengembangan untuk menambah fungsionalitas tambahan.