BAB II

LANDASAN TEORI

Pada bab ini menjelaskan mengenai tempat dilakukannya penelitian dan menjelaskan mengenai teori yang akan digunakan dalam membuat sistem yang akan dibuat.

2.1 Tinjauan Perusahaan

Tinjauan Perusahaan adalah untuk mengetahui keadaan di perusahaan diantaranya adalah sejarah perusahaan dan struktur organisasi perusahaan

2.1.1 Sejarah Perusahaan

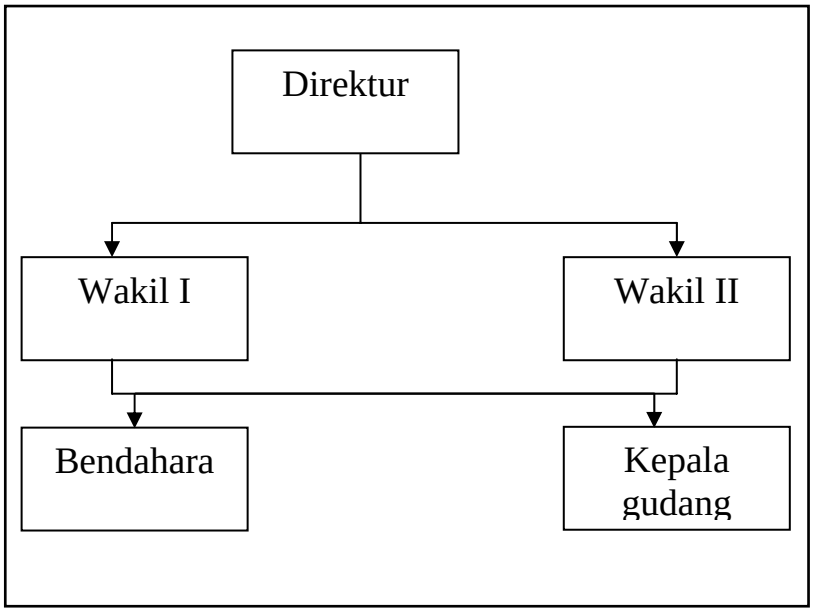
Pada sejarahnya Toko R H MART bernama R H (RIYADLUL HUDA) MART dikarenakan namanya terlalu panjang maka jadi di singkat menjadi R H MART. Toko didirikan pada tahun 2005 oleh bapak jejen tepatnya pada tanggal 10 juni 2005. pada mulanya Toko ini hanya toko sambilan akan tetapi pada masa sekarang Toko mengalami kemajuan sehingga perusahaan yang dulunya dagangannya kecil sekarang mengalami perubahan yang cukup besar yaitu dengan dibangunnya toko yang cukup besar di daerah padalarang dan sekarang mengalami keuntungan yang cukup besar. Toko ini bergerak di bidang penjualan antara lain makanan, minuman, peralatan rumah tangga dan lain-lain. Toko ini membantu masyarakat yang membutuhkan yang di butuhkan oleh masyarakat

tersebut, karena toko ini jaraknya cukup jauh dengan dengan pasar sehingga mempermudah bagi orang yang ingin membeli kebutuhannya.

2.1.2 Struktur Organisasi

Struktur organisasi menunjukkan bagian-bagian jabatan, fungsi dan menyatakan hubungan yang ada dalam suatu Toko. Bila seorang atasan tidak berada di tempat pada saat dibutuhkan, maka tanggungjawab dapat dilimpahkan kepada bawahan sesuai dengan bagan organisasi atau kepada orang lain yang ditunjuk.

Di toko RH MART dalam menjalankan aktivitasnya memiliki struktur organisasi yang cukup sederhana, karena sedikitnya sumber daya manusia dan mengingat Toko ini yang masih dalam taraf perkembangan.



Gambar 2.1 Struktur Organisasi RH Mart Padalarang

Deskripsi Kerja

Pada bagian gudang terdiri dari empat orang, satu orang sebagai Kepala gudang dan tiga orang lainnya sebagai pegawai gudang. Tiga orang pegawai tersebut bertanggungjawab kepada Kepala Gudang.

Tugas dari Kepala gudang yaitu :

1. Bertanggung jawab atas segala aktivitas yang terjadi di gudang.
2. Mengkoordinasikan, memimpin, dan mengendalikan pelaksanaan kegiatan-kegiatan yang berkaitan dengan pemesanan, penerimaan, penyimpanan, pemeliharaan dan pendistribusian alat.

Adapun tugas dari pegawai gudang yaitu :

1. Menjaga barang-barang yang terdapat di gudang agar tidak rusak dan hilang.
2. Mengatur barang-barang sesuai dengan jenisnya sehingga mudah dicari.
3. Menyiapkan barang-barang yang akan keluar.
4. Memeriksa barang-barang yang masuk dan keluar dalam buku persediaan.

2.2 Konsep Dasar Sistem

Sistem berasal dari bahasa Yunani "*Sistema*" yang memiliki pengertian yang luas, tidak ada satu definisi yang pasti. Terdapat dua kelompok pendekatan didalam pendefinisian sistem, yaitu yang menekankan pada prosedur dan menekankan pada komponen / elemennya. Pendekatan pada sistem yang lebih menekankan pada prosedur, mendefinisikan sistem sebagai berikut:

"Suatu sistem adalah jaringan kerja dari prosedur – prosedur yang saling berhubungan, berkumpul bersama – sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu".[1]

Sedangkan prosedur didefinisikan sebagai berikut:

"*Suatu prosedur adalah suatu aturan – aturan operasi Klerikal (Tulis Menulis), biasanya melibatan beberapa orang didalam satu atau lebih departemen, yang diterapkan untuk menjamin penanganan yang seragam dari transaksi – transaksi bisnis yang terjadi*".[1]

Pendekatan sistem yang lebih menekankan pada elemen atau komponennya, mendefinisikan sistem sebagai berikut :

" Sistem adalah kumpulan – kumpulan dari elemen – elemen yang berinteraksi untuk mencapai tujuan tertentu".

2.2.1 Klasifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandang, anatara lain adalah sebagai berikut :

1. Sistem diklasifikasikan sebagai sistem abstrak (*Abstract System*) dan sistem fisik. Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Sistem fisik adalah sistem yang ada secara fisik.
2. Sistem diklasifikasikan sebagai sistem alamiah (*Natural System*) dan sistem buatan manusia (*Human Made System*). Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia. Sistem buatan adalah sistem yang dirancang oleh manusia.

3. Sistem diklasifikasikan sebagai sistem tertentu (*Deterministic System*) dan sistem tak tentu. Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi interaksi diantara bagian-bagian dapat di deteksi dengan pasti sehingga keluaran dari sistem dapat diramalkan. Sistem tak tentu adalah sistem yang kondisi masa depannya diprediksi karena mengandung unsur probalitas.
4. Sistem diklasifikasikan sebagai sistem tertutup (*Closed system*) dan sistem terbuka. Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luar. Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luar.

2.2.2 Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat yang tertentu, antara lain adalah sebagai berikut :

1. Komponen sistem

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi artinya saling bekerjasama membentuk sistem.
2. Batas sistem

   Batas sistem (*boundary*) merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lain atau lingkungan luar.
3. Lingkungan luar sistem

   Lingkungan luar sistem (*enviromment*) dari suatu sistem adalah apapun yang diluar batas dari sistem yang mempengaruhi operasi sistem.

4. Penghubung sistem

   Penghubung (*interface*) merupakan media penghubung antara satu subsistem dengan subsistem yang lain.

5. Masukan sistem

   Masukan sistem (*input*) adalah energi yang dimasukkan ke dalam sistem, masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dpat beroperasi. *Signal input* adalah energi yang diproses unutk di dapatkan keluaran sistem

6. Keluaran sistem

   Keluaran sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

7. Pengolah sistem

   Suatu sistem dapat mempunyai suatu bagian pengolah yang akan mengubah menjadi keluaran.

8. Sasaran sistem

   Suatu sistem pasti mempunyai tujuan (*goal*) dan sasaran (*objective*) suatu sistem dikatakan berhasil apabila mengenai sasaran arau tujuan dari sistem tersebut.

2.2.3 Analisis Sistem

Tahap analisis sistem dilakukan setelah tahap perancangan sistem dan sebelum tahap perancangan sistem. Tahap analisis sistem merupakan tahap yang kritis dan

sangat penting, karena kesalahan dalam tahap ini akan menyebabkan kesalahan di tahap berikutnya.

Menurut **Jogiyanto HM [1]**, analisis sistem didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian – bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan – permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan yang langkah-langkah dalam menganalisis sistem adalah sebagai berikut :

1. *Identify*, yaitu mengidentifikasi masalah.
2. *Understand*, yaitu memahami kerja dari sistem yang ada.
3. *Analyze*, yaitu menganalisis sistem.
4. *Report*, yaitu membuat laporan hasil analisis.

2.2.4 Metode Perancangan Sistem

Paradigma perancangan system yang digunakan dalam perancangan perangkat lunak untuk tugas akhir ini adalah *Waterfall* yang terdapat pada gambar 2.2. Tahapan dalam model *waterfall* terdiri dari:

1. System Enginering

Tahapan ini untuk mendefinisikan sistem, batasan – batasan, maksud dan tujuan dari kebutuhan pembangunan sistem.

2. *Analysis*

   Menentukan kebutuhan yang difokuskan pada perangkat lunak, pemahaman tentang domain informasi, fungsi, kelakuan, performasi, interaksi sesuai dengan kebutuhan dan keinginan pemakai.

3. *Design*

   Merupakan proses multi tahap yang difokuskan pada atribut – atribut program yaitu struktur data, arsitektur perangkat lunak, representasi antar muka, procedural algoritma detail, proses perancangan menterjemahkan kebutuhan – kebutuhan hasil analisis kedalam representasi perangkat lunak sebelum pembuatan kode pemrograman.

4. *Coding / Implementation*

   Tahap penterjemahan hasil perancangan kedalam program – program yang mengguanakan bahasa pemrograman yang sesuai.

5. *Testing*

   Program yang telah dibuat harus diisi, proses pengujian harus difokuskan pada kebenaran logika interal perangkat lunak dan fungsional sistem serta interaksi antara sistem dengan pemakai.

6. *Mainentance*

   Merupakan tahap akhir dimana perangkat lunak yang telah selesai dibuat, dapat mengalami perubahan sesuai permintaan dari pengguna.

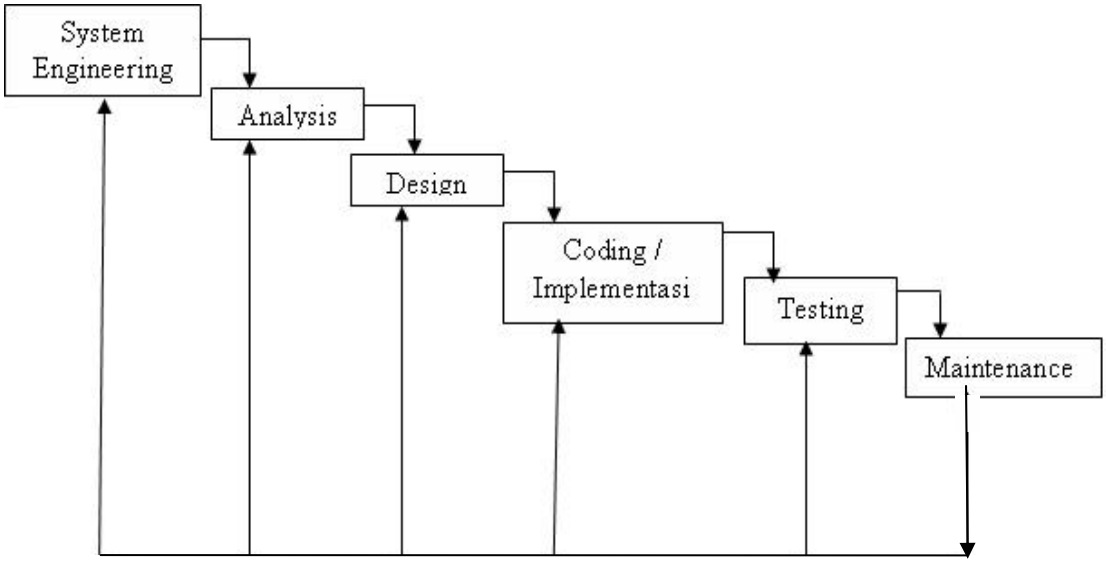
Kelemahan model ini adalah:

1. Proyek-proyek nyata jarang mengikuti alur sekuen yang diusulkan model. Meskipun linear model dapat mengakomodasikan iterasi, namun model melakukan secara tidak langsung. Sebagai hasilnya, perubahan-perubahan dapat menyebabkan kebingungan saat tim pengembangan melakukannya.
2. Sering *customer* sulit menyatakan semua kebutuhannya secara eksplisit. Model ini memerlukan pernyataan eksplisit itu dan sulit mengakomodasi ketidakpastian yang terdapat di awal dari kebanyakan proyek.
3. *Customer* harus memiliki kesabarann. *Versi* yang dapat bekerja dari program tidak akan tersedia sampai akhir dari proyek. Kesalahan besar utama, jika tidak terdeteksi sampai pada program kerja dikaji ulang, maka kesalahan itu dapat mengakibatkan program sama sekali tidak dapat digunakan. Menyiapkan banyak sumber daya.

Alasan kelemahan model ini adalah:

1. Kebutuhan harus telah ditetapkan di awal di siklus hidup.
2. Kebutuhan divalidasi terlalu lambat.

   Meskipun terdapat kelemahan, namun model ini memberikan daftar lengkap jumlah aktivitas minimal yang perlu terdapat di model pengembangan perangkat lunak. Umumnya, model-model pengembangan yang lain merupakan superset dari model pengembangan model *waterfall*, memuat aktivitas-aktivitas yang lebih banyak daripada model *waterfall* ini.



Gambar 2.2 Model *Waterfall*

1. Prototyping

Langkah-langkah dalam metode *prototyping* meliputi :

1. Mengidentifikasi kebutuhan pemakai

   Pada tahap ini, analisis sistem akan melakukan studi kelayakan dan studi terhadap kebutuhan pemakai baik meliputi model interface, teknik prosedural maupun dalam teknologi yang akan digunakan.

2. Pengembangan *prototyping*

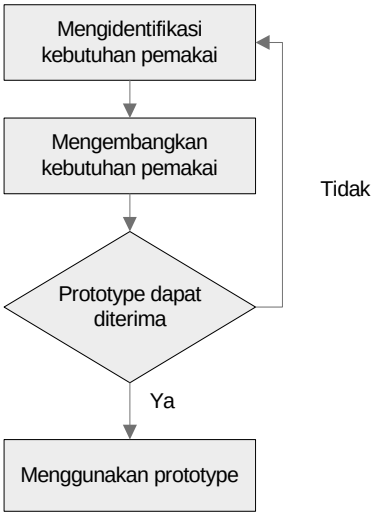
   Pada tahap kedua ini, analisis sistem bekerja sama dengan pemrograman mengembangkan *prototyping* sistem yang akan dibangun.

3. Menentukan *prototyping*

   Apakah dapat dikirim oleh pengguna atau pemakai dan mengidentifikasi sejauh mana pemodelan yang dibuat dapat diterima oleh pemesan.

4. Penggunaan *prototyping*

   Pada tahap ini analisis sistem akan menyerahkan pemrograman untuk mengimplementasikan pemodelan yang dibuatnya menjadi sebuah sistem.



Gambar 2.3 Model Prototyping

2. System Development Life Cycle (SDLC)

System Development Life Cycle (SDLC) merupakan sebuah metode siklus sistem yang terus berputar, untuk menjaga sistem tersebut biasanya digunakan sebagaimana fungsinya (mempermudah pengolahan informasi). SDLC terdiri dari beberapa fase, namun secara garis besar SDLC terbagi kedalam 4 tahapan yaitu :

1. Perencanaan

   Pada fase ini adalah fase kebijaksanaan atau kesiapan pra proyek dalam merancang/mengembangkan suatu sistem.

2. Analisis

   Fase ini adalah fase yang sangat penting dimana fase ini akan diputuskan sistem apa yang akan dibuat, menilai masih layak atau tidaknya suatu sistem dengan memodelkan sistem dan memilih alternatif sistem yang paling baik.

3. Perancangan

   Fase perancangan adalah fase lanjutan dari analisis, dimana alternatif sistem yang dipilih di desain bai untuk input, output, database dan file semuanya diintegrasikan agar bisa terinplementasikan secara baik.

4. Implementasi

   Pada fase ini secara garis besar terdiri dari tiga bentuk kegiatan yaitu :

   a. *Programing and testing*

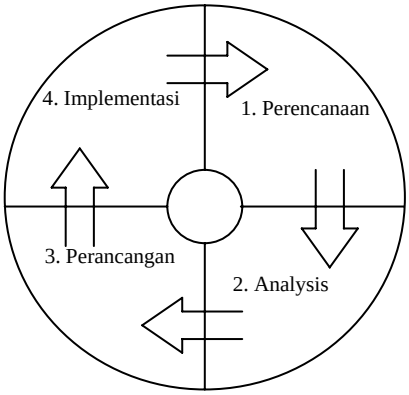
      yaitu mengkonfersikan hasil perancangan ke dalam kegiatan oprasi dengan menggunakan bahasa pemrograman tetentu dan mencoba semua program serta memastikan semua fungsi dapat berjalan secara benar.

   b. *Training*

      yaitu pelatihan sekaligus pengenalan sistem yang baru kepada user dengan modul panduan sistem.

c. *System hangover*

yaitu penggantian sistem lama ke dalam sistem yang barudan ini merupakan tanggung jawab tim desainer.



Gambar 2.4 Struktur model CDLC

2.3 Sekilas Tentang Sistem Pengolahan Data

Sistem pengolahan data adalah suatu kesatuan yang saling berhubungan uang terdiri dari peralatan, tenaga, prosedur, dan data. Setiap sistem informasi mencakup sebuah sistem yang mengerjakan data adalah unutk mengumpulkan data, mengklasifikasikan data, menyimpan data, mengkonfirmasikan data, mentransformasi data, menyeleksi data dan membandingkan data.

2.4 Pengertian Informasi

"*Informasi adalah data yang diolah menjadi bentuk yang lebih berharga dan lebih berguna serta lebih berarti bagi penerimanya"*.[1]

Sumber dari informasi adalah data. Data terbentuk dari karakter – karakter yang dapat berupa alfabet, angka maupun simbol khusus. Data disusun untuk

diolah dalam bentuk struktur data, struktur file dan database. Terdapat perbedaan antara data dan informasi yaitu jika data merupakan bahan baku yang diolah untuk memberikan informasi, sedangkan informasi digunakan dalam pengambilan keputusan, karena itu informasi memiliki tingkat lebih tinggi dari data.

2.4.1 Kualitas Informasi

Kualitas dari suatu informasi tergantung dari tiga hal yaitu informasi harus akurat (*Accurate*), tepat pada waktunya (*Timeliness*), dan relevan (*Relevance*) :

1. Akurat

   Berarti informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan.

2. Tepat pada waktunya

   Berarti informasi yang dating pada penerima tidak boleh terlambat.

3. Relevan

   Berarti informasi mempumyai manfaat unutk penerimanya.

2.5. Pengertian Sistem Informasi

Informasi merupakan hal yang sangat penting bagi manajemen di dalam pengambilan keputusan, informasi diperoleh dari sistem informasi (*information systems*) atau sering juga disebut dengan *processing systems* atau *information processing systems* atau *information generating system*. Sistem informasi didefinisikan oleh **Robert A Leitch** dan **K. Roscoe Davis** dalam **Jogiyanto HM [1]** sebagai berikut :

"Sistem informasi adalah suatu sistem didalam organisasi yang mempertemukan kebutuhan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu ogranisasi dan menyediakan bagi pihak luar tertentu dengan laporan yang diperlukan".[1]

2.5.1 Alat – Alat Pemodelan Sistem Informasi

Dalam memodelkan sistem ada beberapa cara untuk menggambarkannya yaitu:

1. *Flowmap*
2. *Context Diagram*
3. *Data Flow Diagram* (DFD)
4. Kamus Data

2.5.1.1 *Flowmap*

Merupakan suatu diagram yang menggambarkan sistem yang di dalamnya terdapat subsistem-subsistem. Di dalam subsistem-subsistem tersebut terdapat dokumen-dokumen yang mengalir yang menghubungkan antara subsistem-subsistem yang ada pada sistem tersebut.

Beberapa simbol yang dignakan dalam *flow map:*

1. Dokumen

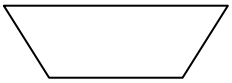
   Menunjukan dokumen sebagai masukan/keluaran baik secara manual atau melalui komputer



Gambar 2.5 Dokumen

2. Proses Manual

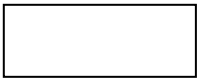
   Merupakan proses yang dikerjakan secara manual



Gambar 2.6 Proses Manual

3. Operasi Komputerisasi

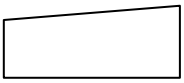
   Menunjukan proses yang dikerjakan oleh komputer



Gambar 2.7 Operasi Komputerisasi

4. Manual Input

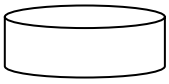
   Menunjukan operasi input secara manual melalui keyboard



Gambar 2.8 Manual Input

5. Magnetic Disk

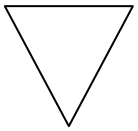
   Merupakan penyimpanan data dalam harddisk



Gambar 2.9 Magnetic Disk

6. Penyimpanan dokumen

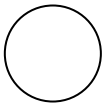
   Digunakan untuk penyimpanam data sebagai arsip secara manual



Gambar 2.10 Penyimpanan dokumen

7. Penghubung atau konektor

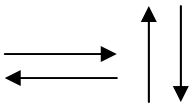
   Digunakan sebagai penghubung kedalam halaman berbeda



Gambar 2.11 Penghubung atau Konektor

8. Alran data

   Menunjukan aliran data antar proses



Gambar 2.12 Aliran Data

.

2.5.1.2 *Context Diagram*

Context Diagram merupakan diagram arus data yang berfungsi untuk menggambarkan yang dirancang disuatu objek. *Context Diagram* ini

menggambarkan secara global atau menyeluruh dari suatu sistem informasi berkaitan dengan aliaran – aliran sistem dengan bagian-bagian luar.

2.5.1.3 *Data Flow Diagram* (DFD)

Data Flow Diagram (DFD) adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data, kemana tujuan data yang keluar sistem, dimana data disimpan, proses apa yang dihasilkan data tersebut dan interaksi antara data yang tersimpan dari proses yang dikenakan pada data tersebut.

Data Flow Diagram (DFD) sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir (misal lewat telepon, surat dan lain – lain) atau lingkunagn fisik dimana data tersebut akan disimpan, misalnya file kertas, hardisk, tape, disket dan lain sebagainya.

DFD merupakan alat yang cukup popular saat ini, karena dapat menggambarkan arus data didalam sistem dengan terstruktur dan jelas. Lebih lanjut DFD merupakan dokumentasi dari sistem yang baik.

Beberapa simbol yang digunakan di DFD antara lain:

a. Kesatuan Luar (*Entity Eksternal*)

Setiap sistem pasti mempunyai batasan sistem yang memisahkan suatu sistem dengan lingkungan luarnya. Sistem akan menerima iput dan menghasilkan output kepada lingkungan luarnya. Kesatuan luar (*entity eksternal*) merupakan kesatuan dilingkungan luar sistem, dapat berupa orang, organisasi atau sistem lainnya yang

berada dilingkungan luarnya yang akan memberikan input atau menerima output dari sistem. Kesatuan luar ini kebanyakan adalah salah satu dari yang berikut ini:

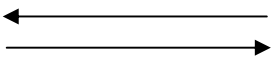
1. Suatu kantor, departemen atau divisi dalam perusahaan tetepai diluar sistem yang sedang dikembangkan.
2. Orang atau sekelompok orang di organisasi tetapi diluar sistem yang sedang dikembangkan.



Gambar 2.13 External Entity.

b. Aliran Data (*Data Flow*)

Aliran data si DFD diberikan simbol suatu panah. Aliran data ini mengalir diantara proses (*Process*), simpanan data (*data storage*) dan kesatuan luar (*entity eksternal*). Aliran data ini menunjukan aliran dari data yang dapat berupa masukan dri sistem atau hasil dari proses sistem.

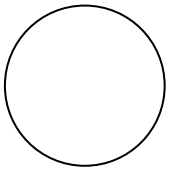


Gambar 2.14 Data Flow.

c. Proses (*Process*)

Proses adalah simbol pertama data flow diagram. Proses dilambangkan dengan lingkaran, dimana proses ini menunjukan bagian dari sistem yang mengubah satu atau lebih input dan output. Nama proses dituliskan dengan satu

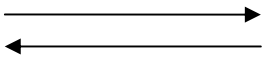
kata, singkatan atau kalimat sederhana. Lambang sebuah proses dapat dilihat pada gambar 2.3 berikut :



Gambar 2.15 Simbol proses

d. Aliran Data (Flow)

Aliran Data digambarkan dengan tanda panah dari proses. Aliran data juga digunakan untuk menunjukan bagian – bagian informasi dari satu bagian ke bagian lain. Pembagian nama untuk aliran ini menunjukan sebuah arti untuk sebuah aliran. Untuk kebanyakan sistem yang dibuat, aliran data sebenarnya mengambarkan data yakni angka, huruf, pesan, floating point, dan macam - macam informasi lainnya. Lambang aliran data dapat dilihat pada gambar 2.4 berikut :

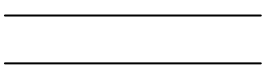


Gambar 2.16 Simbol input dan output

e. Simpanan Data (Storage**)**

Simpanan data digunakan sebagai penyimpanan bagi paket – paket data. Notasi penyimpanan data digambarkan dengan garis horizontal yang pararel. Simpanan data merupakan simpanan data dari data yang berupa suatu file atau database di sistem komputer ataupun berupa arsip atau catatan manual. Nama dari

simpanan data menunjukan nama filenya. Lambang simpanan data dapat dilihat pada gambar 2.5 berikut :



Gambar 2.17 Simbol simpanan data

f. Terminator

Terminator digambarkan dengan sebuah kotak yang menggambarkan kesatuan luar (*eksternal entitty*) yang berhungan dengan sistem. Kesatuan luar merupakan kesatuan (*entity*) dilingkungan luar sistem yang dapat berupa orang, Organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan input atau output dari sistem, lambang terminator dapat dilihat pada gambar 2.6 berikut :



Gambar 2.18 Simbol terminator

.Berkas / Simpanan Data (*Data Storage*)

Berkas atau simpanan data merupakan simpanan dari data yang dapat berupa :

1. Suatu file atau database di sistem komputer.
2. Suatu arsip atau catatan manual.
3. Suatu kotak tempat data di meja seseorang.
4. Suatu tabel acuan manual.
5. Suatu agenda – agenda atau buku.

2.5.1.4 Kamus Data

Kamus data atau disebut juga *Data Dictionary* adalah catalog fakta tentang data dan kebutuhan – kebutuhan informasi dari suatu sistem informasi. Dengan mengggunakan kamus data, analisis sistem dapat mendefinisikan data yang mengalir di sistem dengan lengkap. Salah satu komponen kunci dalam sistem manajemen database (DBMS) adalah file khusus yang disebut kamus data (*Data Dictionary*). Kamus data berisi informasi tentang struktur database, untuk setiap elemen data yang disimpan dalam database seperti nomor pokok dan diuraikan secara lengkap mulai dari nama, tempat penyimpanan, program kumpulan yang berhubungan dan lain – lain. Kamus data biasanya dipelihara secara otomatis oleh sistem manajemen database.

2.6 Pengertian Basis Data

Basis data terdiri dari dua kata, basis dan data. Basis dapat diartikan kurang lebih dapat diartikan sebagai markas, gudang atau tempat berkumpul. Sedangkan data adalah repesentasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli dan lain – lain), barang dan sebagainya yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi atau kombinasi lainnya. Basis data sendiri dapat didefinisikan dalam sejumlah sudut pandang seperti [2] :

1. Himpunan kelompok data (arsip) saling berhubungan yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.

2. Kumpulan file / tabel / arsip saling berhubungan yang disimpan dalam media penyimpanan elektronis.

2.6.1 Sistem Pengelola Basis Data (*Database Manajemen System*/DBMS)

Pengelolaan basis data secara fisik tidak dilakukan oleh pemakai secara langsung, tetapi ditangani oleh sebuah perangkat lunak (sistem) yang khusus / spesifik. Perangkat lunak inilah (sebagai DBMS) yang akan menentukan bagaimana data diorganisasikan, disimpan, diubah dan diambil kembali. Ia juga menerapkan mekanisme pengamanan data, pemakaian data secara bersama, keakuratan data dan sebagainya. [3]

Perangkat lunak yang termasuk DBMS seperti dBase III+, dBase IV, FoxBase, Ms-Access, Borland Paradox, Ms-SQL Server, Oracle Borland-interbase dan lain sebagainya. Salah satu tujuan DBMS adalah untuk menyediakan fasilitas / antar muka (*interface*) dalam melihat data (yang lebih ramah atau *user friendly*) kepada pemakai.

2.6.2 Bahasa Basis Data

DBMS merupakan perantara bagi pemakai, dengan basis data dalam disk. Cara berinteraksi / berkomunikasi antar pemakai dengan basis data tersebut diatur dalam suatu bahasa khusus yang diterapkan oleh perusahaan pembuat DBMS. Bahasa ini dapat kita sebut sebagai bahasa basis data yang terdiri dari sejumlah perintah yang diformulasikan dan dapat diberikan user serta dapat dikenali / diproses oelh DBMS untuk melakukan suatu aksi atau pekerjaan tertentu.

Sebuah bahasa basis data dapat dipilah dalam 2 bentuk yaitu :

1. *Data Definition Language* (DDL)
2. *Data Manipulation Language* (DML)

Struktur atau skema basis data yang menggambarkan desain basis data secara keseluruhan dispesifikasikan dengan bahasa khusus yang disebut *Data Definition Language* (DDL). Dengan bahasa inilah dapat dibuat tabel baru, membuat indeks, merubah table, menentukan struktur penyimpanan table dan sebagainya, yang mana hasil dari kompilasi perintah *Data Definition Language* (DDL) adalah kumpulan table yang disimpan dalam file khusus yang disebut kamus data (*Data Dictionary*).

Sedangkan *Data Manipulation Language* (DML) merupakan bentuk bahasa basis data yang berguna untuk melakukan manipulasi dan pengambilan data pada suatu basis data. Manipulasi data dapat berupa:

a. Penyisipan atau penambahan data baru dari suatu basis data.
b. Penghapusan data dari suatu basis data.
c. Pengubahan dari suatu basis data.

Data Manipulation Language (DML) merupakan bahasa yang bertujuan memudahkan pemakai untuk mengakses data sebagaimana direpresentasikan oleh model data.

2.7 Perancangan Basis Data

Dalam perancangan basis data yang harus dilakukan yaitu :

1. *Entity Relationship Diagram* (ERD)

ERD merupakan notasi grafik dalam pemodelan data konseptual yang mendeskripsikan hubungan antar penyimpanan. ERD digunakan untuk memodelkan struktur data dan hubungan antar data, karena hal ini relative kompleks. Dengan ERD kita dapat menguji model dengan menjabarkan proses yang harus dilakukan. ERD menggunakan sejumlah notasi dan simbol untuk menggambarkan struktur dan hubungan antar data. Pada dasarnya ada 3 macam simbol yang digunakan yaitu :

a. Entitas

   Merupakan suatu objek yang dapat diidentifikasikan dalam lingkungan pemakai, sesuatu yang penting bagi pemakai dalam konteks sistem yang akan dibuat

b. Atribut

   Entitas mempunyai elemen yang disebut atribut yang berfungsi untuk mendeskripsikan karakter entitas.

c. Hubungan

   Relationship sebagaimana halnya entitas maka dalam hubungan pun harus dibedakan antar hubungan atau bentuk hubungan antar entitas dengan isi dari hubungan itu sendiri.

Relasi antar 2 tabel atau 2 file dapat dikategorikan / dibedakan menjadi 3 macam yaitu :

1. *One To One Relationship*

   Yang berarti, entitas pada himpunan entitas A berhubungan paling banyak dengan 1 entitas B, dan begitu juga sebaliknya setiap entitas pada himpunan B berhubungan paling banyak 1 dengan entitas pada himpunan entitas B.

2. *One To Many Relationship*

   Yang berarti, entitas pada himpunan entitas A berhubungan dengan banyak pada himpunan entitas B, tetapi tidak sebaliknya setiap entitas pada himpunan entitas B, hubungan paling banyak dengan satu entitas pada himpunan entitas A.

3. *Many To Many Relationship*

   Yang berarti, entitas pada himpunan entitas A berhubungan dengan banyak pada himpunan entitas B dan begitu juga sebaliknya, setiap entitas pada himpunan entitas B berhubungan banyak dengan himpunan entitas B.

2. Normalisasi

Normalisasi merupakan proses yang menggunakan pendekatan formal untuk menelaah dan kemudian mengelompokan data item / field/ attribut ke bentuk yang lebih baik dalam menghadapi perubahan-perubahan bisnis dimasa mendatang serta meminimumkan pengaruh perubahan pada sistem aplikasi atau program. Ada beberapa bentuk nomalisasi, yaitu :

1. Bentuk Normal Pertama

   Aturan bentuk normal I sebagai relasi yang tidak boleh mengandung kelompok berulang (*Repeating Group*).

2. Bentuk Normal Kedua

   Aturan normalisasi kedua berbunyi bahwa setiap field yang tidak bergantung sepenuhnya (fungsionil) pada kunci primer harus dipindahkan ke tabel lain. Bergantung secara fungsionil adalah sebuah istilah matematika, yang berarti bahwa sebuah field sangat ditentukan oleh kunci.

3. Bentuk Normal Ketiga

   Aturan normalisasi ketiga berbunyi bahwa tidak boleh ada kebergantungan antara field field non kunci.

4. Bentuk Normal Keempat

   Suatu relasi disebut dalam bentuk normal IV, jika relasi tersebut sudah dalam bentuk normal III dan seluruh attribut yang bukan primary key tidak tergantung bernilai banyak (*Multivalued Dependencies*) kepada primary key-nya.

   Contoh normalisasi

No_kota	Tgl_nota	Tgl_sewa	Tgl_beli	Kode	Nama_pgl	Alamat	Kode_vcd	Jdl_vcd
001A	01-01-04	01-01-04	04-01-04	02001	Ineke p	Osamaliki	V-002	The lost
							V-003	vampire

sutradara	aktor	jumlah	harga	total
jackly	Donna	1	1500	1500
Goen	ciblexa	2	2000	4000

Bentuk tidak normal

No_kota	Tgl_nota	Tgl_sewa	Tgl_beli	Kode	Nama_pgl	Alamat	Kode_vcd	Jdl_vcd
001A	01-01-04	01-01-04	04-01-04	02001	Ineke p	Osamaliki	V-002	The lost
001A	01-01-04	01-01-04	04-01-04	02001	Ineke p	Osamaliki	V-003	vampire

sutradara	aktor	jumlah	harga	total
jackly	Donna	1	1500	1500
Goen	ciblexa	2	2000	4000

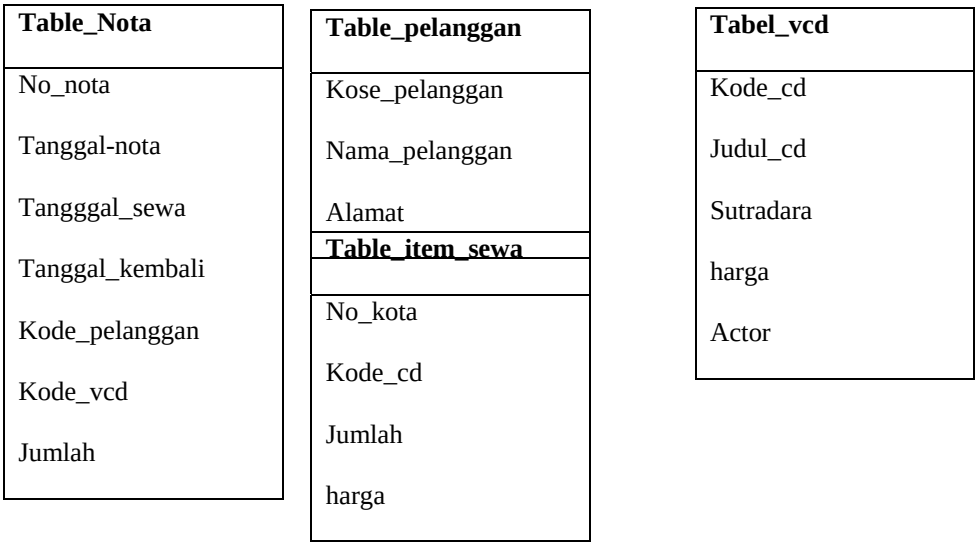
Bentuk normal ke-1

Table_Nota
No_nota
Tanggal-nota
Tangggal_sewa
Tanggal_kembali
Kode_pelanggan
Kode_vcd
Jumlah

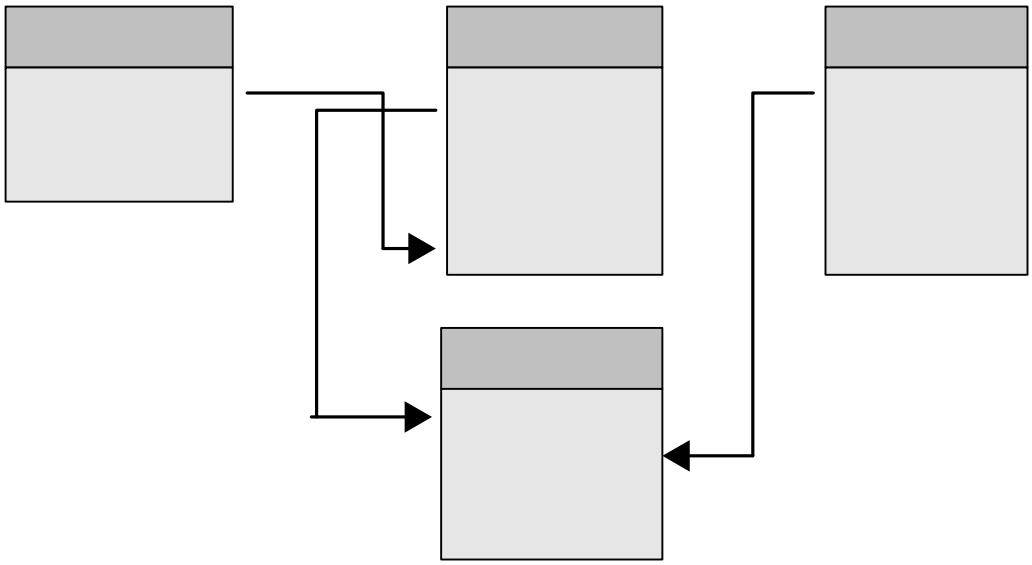
Table_pelanggan
Kose_pelanggan
Nama_pelanggan
Aamat

Tabel_vcd
Kode_cd
Judul_cd
Actor
Sutradara
harga

Bentuk normal ke 2



Bentuk norma ke 3



Bentuk normal Ke 3

Gambar 2.19 Contoh normalisalai

2.8 Pengertian Tentang Inventory

inventory / persediaan barang adalah suatu aktiva yang meliputi barang-barang milik perusahaan dengan maksud untuk di jual / digunakan dalam suatu periode tertentu, atau persediaan barang-barang yang masih dalam pengerjaan / proses produksi, ataupun persediaan bahan baku yang menunggu penggunanya dalam suatu proses produksi.

Persediaan merupakan bahan-bahan yang disediakan dan bahan-bahan dalam proses yang terdapat dalam perusahaan untuk produksi suatu barang-barang jadi / produk yang disediakan untuk memenuhi permintaan dari konsumen atau langganan setiap periode.

2.8.1 Jenis-jenis Persediaan

Menurut fungsi, persediaan dibagi menjadi 3 jenis, yaitu :

1. *Batch Stok / Lot Size Inventory*

   Persediaan yang diadakan karena membeli atau membuat bahan-bahan atau barang-barang dalam jumlah yang lebih besar dari jumlah yang dibutuhkan saat itu.

2. *Fluctuation*

   Persediaan yang diadakan unutk menghadapi fluktuasi permintaan yang tidak dapat diramalkan.

3. *Anticipation Stock*

   Persediaan yang di adakan untuk menghadapi fluktuasi permintaan yang dapat diramalkan, berdasarkan pola musiman yang terdapat dalam satu

tahun dan untuk menhadapi penggunaan atau penjualan atau permintaan yang meningkat.

2.9 Perangkat Lunak Pendukung

Software atau perangkat lunak yang mendukung dalam sistem informasi ini yaitu :

2.9.1 Sejarah Singkat Borland Delphi

Delphi merupakan perangkat pengembangan aplikasi yang sangat terkenal dilingkungan Windows. Ide munculnya Delphi berasal dari bahasa pemograman yang cukup terkenal yaitu Pascal. Bahasa pascal sendiri telah diciptakan oleh seorang ilmuwan dari Swiss yaitu Niklaus Wirth. Nama pascal diambil dari ahli matematika dan filsafat dari Prancis, yaitu Blaise Pascal (1623-1663).

Sejak saat itu, muncul beberapa versi yang dirilis oleh Borland International Incorporation pada tahun 1983. Turbo Pascal yang muncul pertama kali hanya dapat dijalankan di sistem operasi DOS. Namun dalam perkembangan selanjutnya, Borland International Incorporation juga merilis turbo pascal yang berjalan di Window 3.x yaitu Turbo Pascal for Windows pada tahun 1992. Borland International Corporation menggabungkan Turbo Pascal for DOS dan Tirbo Pascal for Windows menjadi satu paket bahasa pemograman yang dikenal dengan turbo Pascal versi 7.0.

Karena pemograman windows dengan Turbo Pascal dirasa masih sulit, maka sejak tahun 1993 Borland International Incorporation mengembangkan

bahasa pascal yang bersifat visual. Hasil dari pengembangan ini adalah dari isinya Borland Delphi pada tahun 1995. Sekitar satu tahun sesudahnya yaitu tahun 1996, Borland International Corporation merilis Borland Delphi 2.0 yang dikhususkan untuk sistem operasi windows 95 dan windows NT. Kemudian satu tahun berikutnya, tahun 1997 Borland International Corporation merilis Borland Delphi 3.0 yang dikhususkan untuk sistem operasi Windows 97 dan Windows NT. Pada tahun 1998 telah dirilis Borland Delphi 4.0. Sedangkan Borland Delphi 5.0 telah dirilis pada tahun 1999.

Salah satu kecanggihan yang ditawarkan Delphi adalah MIDAS (*Multiuser Distribution Aplication Service*), yang memungkinkan sebuah aplikasi untuk berkomunikasi dan berbagai data secara mendunia melalui jaringan komunikasi. Delphi sebelumnya dipasarkan dengan harga berbeda terdiri dari Delphi Standart, Delphi Profesional, Delphi Client/Server dan Delphi Interpise.

Borland menyatakan bahwa Delphi merupakan alat yang dapat digunakan untuk *Rapid Aplication Development* (RAD), yang hingga kini teteap terbukti menjadi bahasa pemograman yang lebih baik. Dengan keberhasilan ini Borland mengembangkan produk lain yang mirip yakni C++ Builder, JBuilder(*Java Language*), kedua produk ini dipasarkan dan mendapat sambutan yang baik pula.

Delphi tidak saja dapat mengakses database seperti paradox, xbase tetapi dapat mengakses database lainnya seperti oracle, sybase, mc-sql, dll. Dengan menggunakan ODBC (*Open Database Connectivity*), begitu aplikasi selesai untuk memindahkan dari satu jenis database ke yang lainnya tidak pelu

mengubah aplikasi secara keseluruhan. Delphi dapat dinyatakan oleh banyak pengamat sebagai no_limit (tanpa batas).

Pernyataan ini didukung dengan kemampuan dan keunggulan seperti :

1. Native code compiler yang cepat,
2. Debugger yang terpadu dan text editor,
3. High level database akses,
4. Teknologi two ways tools yang canggih,
5. Dapat dibuat dalam source code yang sama,
6. Dapat digunakan untuk membuat DOS/windows object lainnya.

Kelebihan Delphi dapat menyediakan fasilitas yang luar dari fungsi untuk membuat form hingga fungsi untuk menggunakan beberapa format file basisdata yang popular. Fasilitas-fasilitas yang tersedia di Delphi antar lain :

1. Komponen yang dapat dipakai ulang dan dapat dikembangkan.

Delphi dapat mempermudah pembuatan program bagi komponen visual dan komponen non visual sudah tersedia didalam Delphi antara lain komponen button, komponen database, komponen menu dan dialog. Sehingga dapat memungkinkan kita untuk menampilkan data tanpa memprogram cukup dengan menggunakan tombol mouse.

2. Dapat mengakses VBX.

Delphi dapat mengakses objek-objek VBX secara langsung. Dalam Delphi VBX dianggap sebagai kumpulan komponen yang dapat digunakan langsung untuk membuat aplikasi.

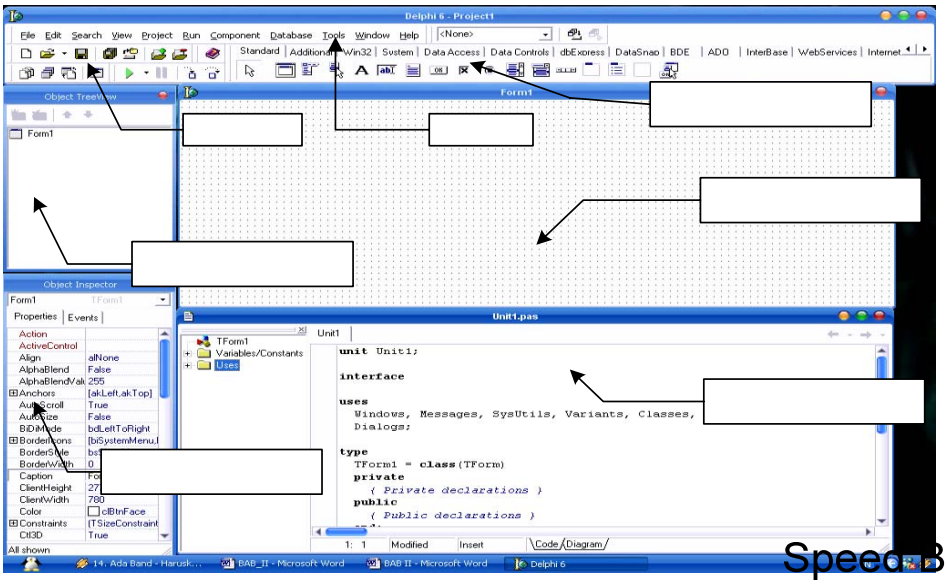
3. Template Aplikasi dan Template Form.

Dalam Delphi telah didefinisikan template aplikasi dan template form untuk membuat semua aplikasi dengan lebih cepat.

2.9.2 Sarana Yang Disediakan Delphi

Pemrograman Delphi atau yang lebih dikenal dengan *Delphi for windows* semakin populer dan banyak digunakan dikalangan *programmer*. Satu hal yang membuat bahasa pemrograman ini familiar adalah *performance*-nya yang sangat menarik dan handal. Fasilitas-fasilitas yang diberikan memberikan kemudahan dalam menyusun suatu program aplikasi dan komponen-komponen yang diberikan memungkinkan penyusunan pogram aplikasi skala besar dan kompleks.

Delphi menyediakan fasilitas secara lengkap, diantaranya dengan disediakannya (*IDE = Integrated Development Environment*) yang merupakan sebuah lingkungan dimana semua tools yang diperlukan untuk desain, menjalankan dan mengetes sebuah aplikasi disajikan dan terhubung dengan baik sehingga memudahkan pengembangan program. Pada *Delphi*, *IDE* terdiri dari *editor program*, *debugger*, *toolbar*, *editor image* dan lain-lain. Integrasi ini dapat memberikan kepada kita kemudahan dalam membangun suatu aplikasi yang kompleks. *IDE Delphi* mempunyai bagian-bagian diantaranya :



Gambar 2.20 Interface Delphi

1. Menu

Menu pada Delphi memiliki kegunaan seperti pada aplikasi Windows lainnya. Dari menu ini, bisa memanggil atau menyimpan program, menjalankan program dsb. Singkatnya segala sesuatu yang berhubungan dengan IDE Delphi dapat dilakukan dari menu.

2. Speed Bar

Speed Bar atau sering juga disebut *toolbar* merupakan bagian dari *IDE Delphi* yang terdiri dari kelompok item-item yang dapat membantu pemrogram untuk menjalankan fungsi-fungsi dengan cepat dan mudah. Fungsi item-item tersebut sebenarnya adalah sama dengan fungsi yang ada pada main menu. Dengan kata lain, setiap tombol pada Speed Bar menggantikan salah satu item menu.

3. Component Pallete

Component Pallete adalah visual dari *Visual Component Library* (*VCL*). Komponen-komponen ini dibagi menjadi beberapa *group* (*pages*) yang dikelompokkan sesuai dengan fungsinya dan dinyatakan dengan tab. *Component pallet* berfungsi untuk memilih komponen-komponen yang akan digunakan dalam pembuatan program.

4. Form

Form adalah basis dari setiap aplikasi *Delphi*. *Form* sama seperti *window* (jendela) yang kita lihat dalam aplikasi-aplikasi *windows.* Pada *Delphi Form* adalah tempat untuk meletakkan semua komponen yang berada pada *Component Pallet.*

5. Object Inspector

Object Inspector sangat dibutuhkan dalam pembuatan aplikasi. Dengan *Object Inspector*, kita dapat mengubah property dari setiap item dengan mudah dan juga dapat mngontrol tindakan yang diambil jika terjadi *event*. *Object Inspector* terdiri dari dua tab, yaitu tab *Properties* dan tab *Events*.

6. Jendela Penyunting *(Code Explorer)*

Jendela penyunting merupakan bagian terpenting di lingkungan *Delphi*. Jendela ini dipakai unuk menuliskan program *Delphi* yang berupa logika dalam bentuk struktur program *pascal*. Jendela penyunting pada *Delphi* sangat canggih karena dilengkapi dengan fasilitas-fasilitas highlight untuk memudahkan menemukan kesalahan.

7. Object TreeView

Object TreeView berisi daftar komponen yang telah diletakkan pada form designer.