

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

Pengertian dari sistem yang dikemukakan oleh *American National Standart Institute (ANSI)*, mendefinisikan sistem sebagai berikut : "Sistem adalah serangkaian metode prosedur atau teknik yang disatukan oleh interaksi yang teratur sehingga membentuk suatu kesatuan yang terpadu". Menurut *International Organization for TechNIPal Committee* : "Sistem adalah suatu kelompok manusia, mesin, metode yang terpadu dan diperlukan untuk mencapai fungsi-fungsi yang spesifik".

Secara umum dasar dari suatu sistem informasi mempunyai beberapa pengertian, seperti :

1. Komponen dari suatu sistem yang saling berhubungan dan tidak tergantung kepada yang lain (berdiri sendiri).
2. Sebuah sistem mempunyai masukan dan keluaran.
3. Semua sistem mentransformasikan semua input ke dalam output.
4. Suatu sistem biasanya terdapat suatu sub sistem didalamnya.

2.1.1 Pengertian Dasar Sistem Informasi

Informasi adalah data yang telah disusun sedemikian rupa sehingga bermakna dan bermanfaat karena dapat dikomunikasikan kepada seseorang yang akan menggunakannya untuk membuat keputusan. Suatu sistem informasi merupakan koordinasi dari *hardware*, *software* dan data yang bekerja sama untuk menghasilkan suatu set-set tujuan tertentu.

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi yang bersifat manajerial dan kegiatan strategi dari suatu organisasi serta menyediakan pihak luar tertentu dengan laporan yang diperlukannya.

(Jogiyanto HM, 1995).

2.1.2 Komponen-komponen sistem informasi

Sebuah sistem informasi menggunakan sumber daya manusia (Pengguna dan ahli sistem informasi), *hardware* (mesin dan media), *software* (program-program dan prosedur-prosedur), data (data dan pengetahuan dasar), dan jaringan kerja (media komunikasi dan dukungan terhadap jaringan) untuk meningkatkan kegiatan masukan, pemrosesan, keluaran, penyimpanan, dan pengontrolan yang menjadikan sumber data ke dalam suatu produk informasi.

(Jogiyanto HM, 1995)

Manusia, *hardware*, *software*, data dan jaringan kerja adalah lima dasar sumber daya sistem informasi. Sumber daya manusia termasuk pengguna dan

ahli sistem informasi, *hardware* terdiri dari mesin dan media, software termasuk diantaranya program-program dan prosedur-prosedur. Sumber daya dapat berupa data dan pengetahuan dasar serta jaringan kerja yang didalamnya termasuk media komunikasi dan dukungan terhadap jaringan.

2.2 Database

Database atau basis data adalah komponen-komponen data yang memiliki kaitan satu file dengan file lainnya, sehingga membentuk suatu bagun data untuk menginformasikan suatu organisasi atau instansi dalam batasan tertentu atas tabel-tabel, sementara tabel berisi *field* (kolom) dan *record* (baris).

Hubungan antara tabel dalam suatu database dikenal sebagai relasional. Terdapat beberapa metode untuk mewujudkan hubungan antara tabel dalam database agar tabel-tabel yang dibuat efektif. Metode ini dikenal dengan sebagai prosedur normalisasi, yaitu prosedur yang memaparkan suatu metode untuk mewujudkan pola hubungan antar tabel dan pola penyimpanan database yang terstruktur (D. Suryaadi H.S).

2.3 Normalisasi

Proses normalisasi data dimaksudkan untuk menghilangkan *redudancy* semaksimal mungkin dan meningkatkan kemudahan operasi untuk merubah, menghapus dan memasukkan data pada satu database. Suatu relation dikatakan sudah berada pada bentuk normalisasi tertentu bila sudah memenuhi batasan

tertentu pada tingkat tersebut. Tingkat normalisasi yang lebih tinggi dianggap lebih baik dari tingkat bawahnya.

Tingkat-tingkat normalisasi adalah:

- a. Relation umum (yang belum ada dan sudah ternormalisasi)
- b. 1 NF (*First Normal Form*)

Suatu relation dikatakan sudah berada pada 1 NF, jika semua nilai atributnya adalah atomik atau tunggal.

- c. 2 NF (*Second Normal Form*)

Suatu relation sudah berada 2 NF jika sudah berada pada 1 NF dan setiap atribut yang bukan kunci haruslah bergantung secara fungsi pada *primary key*.

- d. 3 NF

Suatu relation sudah berada pada 3 NF jika sudah berada di dalam 2 NF dan setiap atributnya yang bukan *key* tidak punya hubungan yang *transitive*.

2.4 Unified Modeling Language

Unified Modeling Language (UML) adalah sebuah bahasa yang telah menjadi standar dalam industri visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. Menggunakan UML kita dapat membangun model untuk segala bentuk dan jenis aplikasi perangkat lunak, serta aplikasi yang

dibangun dapat berjalan pada perangkat lunak dengan sistem operasi dan jaringan apapun. Selain itu, dengan menggunakan UML programmer dapat mengerti, memahami dengan jelas maksud, tujuan dan arah rancangan sistem, sehingga sistem perangkat lunak yang dirancang dapat dibangun dengan bahasa program apapun. Namun karena UML merupakan suatu model pengembangan sistem perangkat lunak yang berbasis *Object Oriented* sehingga menggunakan bentuk class dan operation dalam konsep dasarnya, maka lebih cocok dalam pemrogramannya dengan menggunakan bahasa-bahasa pemrograman yang berbasis OOP (*Object Oriented Programming*) seperti C, C++, Java, VB dan lain sebagainya.

Secara lebih mendalam, UML lebih dari sekedar sebuah standart dan penemuan dari suatu notasi-notasi yang disatukan, tetapi juga berisi konsep-konsep baru yang menarik yang tidak ditemukan secara umum dalam komunitas *object oriented*. (Errikson and Penker, 1998).

Tujuan dari UML adalah:

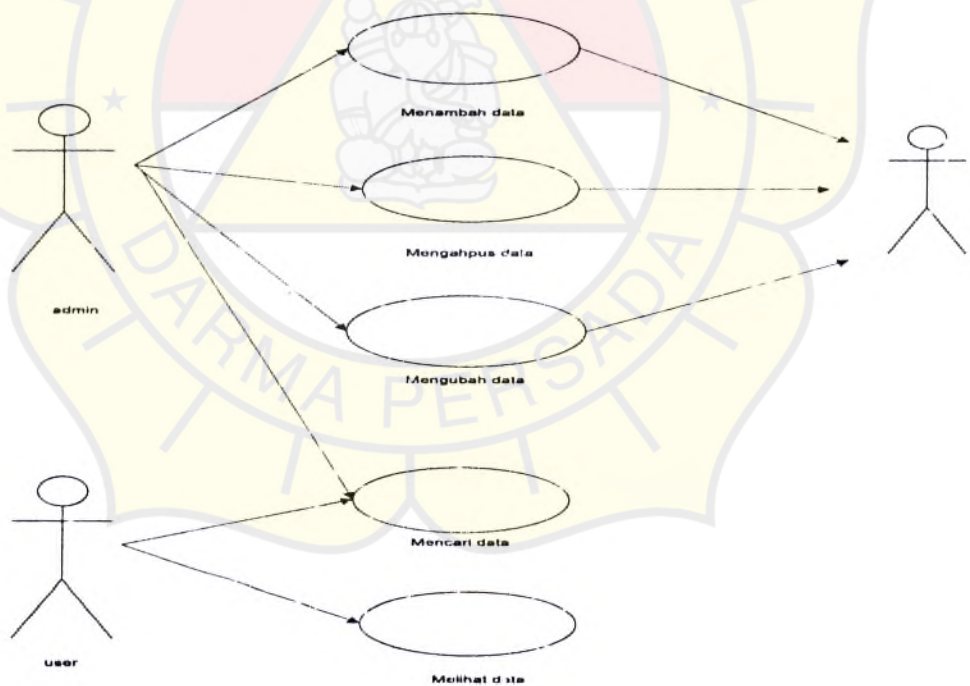
1. Memberikan model yang siap pakai, bahasa permodelan visual yang ekspresif untuk mengembangkan dan saling menukar model yang mudah dan dimengerti secara umum.
2. Memberikan bahasa permodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
3. Menyatukan praktek-praktek terbaik yang terdapat dalam permodelan.

Membangun suatu model perangkat lunak dengan UML, digunakan bentuk-bentuk diagram atau simbol untuk merepresentasikan elemen-elemen dalam sistem. Bentuk diagram yang digunakan untuk merepresentasikannya adalah sebagai berikut : (Adi Nugroho 2005)

o *Use case Diagram*

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Sebuah *use case* merepresentasikan interaksi antara aktor dengan sistem. *Use case* dapat menggambarkan bagaimana seseorang akan menggunakan / memanfaatkan sistem.

Contoh use case diagram :



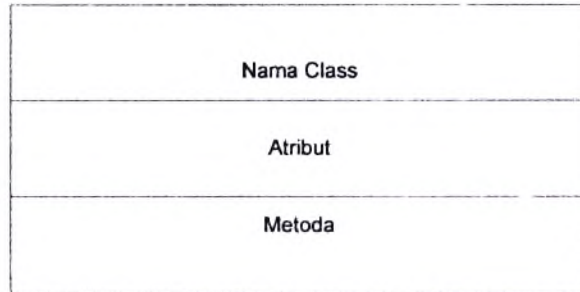
Gambar 2.1 contoh diagram use case

o *Class Diagram*

Class Diagram adalah sebuah spesifikasi yang jika diinstasiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class diagram menunjukkan struktur yang statis dari beberapa class dalam suatu sistem. Class-class merepresentasikan suatu keadaan (atribut / properti) dan yang akan dikerjakan oleh sistem (metoda / fungsi).

Hubungan Antar Class :

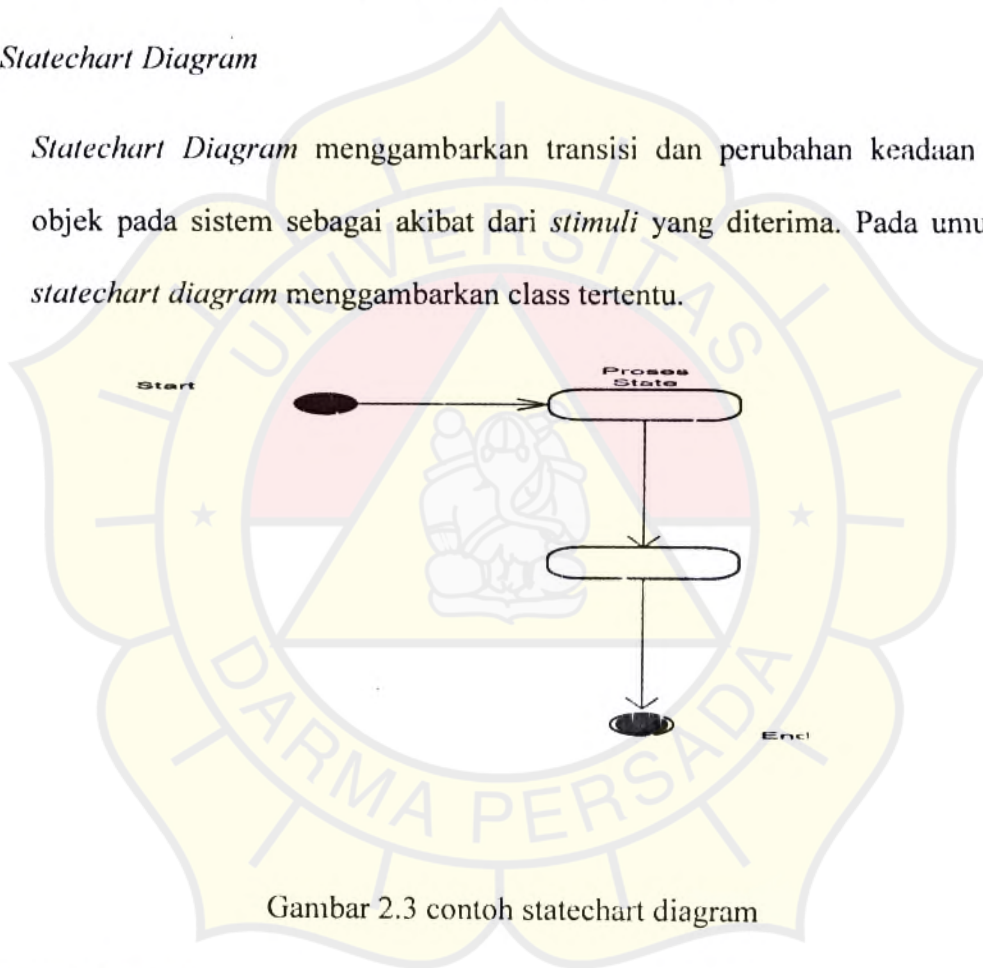
1. Asosiasi yaitu hubungan statis antar class. Umumnya menggambarkan class yang memiliki atribut berupa class lain, atau class yang harus mengetahui eksistensi class lain. Asosiasi juga merupakan koneksi semantik antara suatu class dengan class yang lainnya.
2. Agregasi, yaitu hubungan yang menyatakan bagian.
3. Pewarisan, yaitu hubungan antar class. Class dapat diturunkan dari class lain dan mewarisi semua atribut dan metoda class asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari class yang mewarisinya.
4. Hubungan dinamis, yaitu rangkaian pesan yang di-*passing* dari suatu class kepada class lain.



Gambar 2.2 contoh class diagram

o *Statechart Diagram*

Statechart Diagram menggambarkan transisi dan perubahan keadaan suatu objek pada sistem sebagai akibat dari *stimuli* yang diterima. Pada umumnya *statechart diagram* menggambarkan class tertentu.

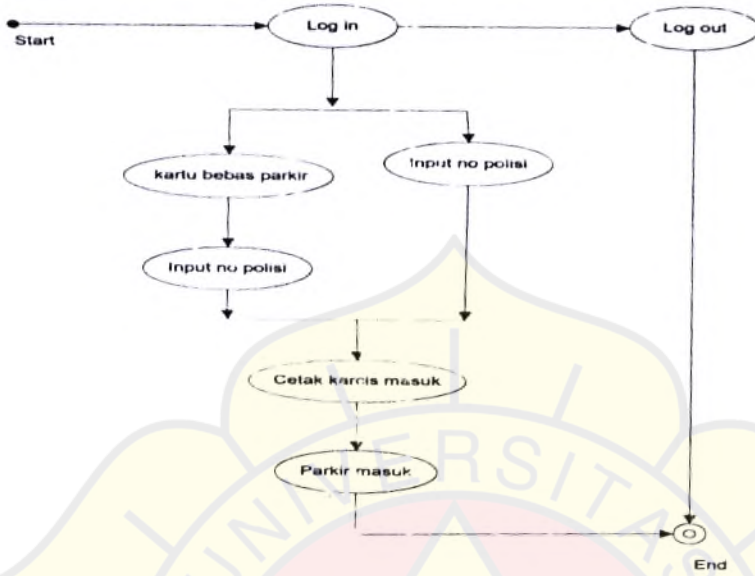


Gambar 2.3 contoh statechart diagram

o *Activity Diagram*

Activity Diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, *activity diagram* merupakan *state diagram* khusus, dimana

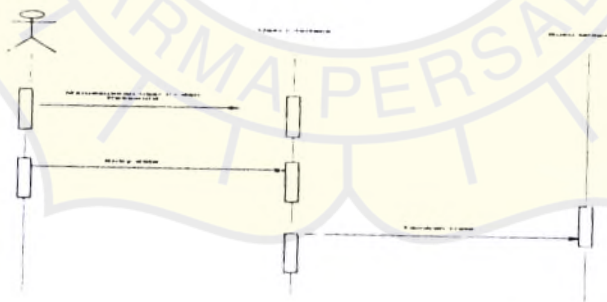
sebagian besar *state* sebelumnya. Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih.



Gambar 2.4 contoh activity diagram

o *Sequence Diagram*

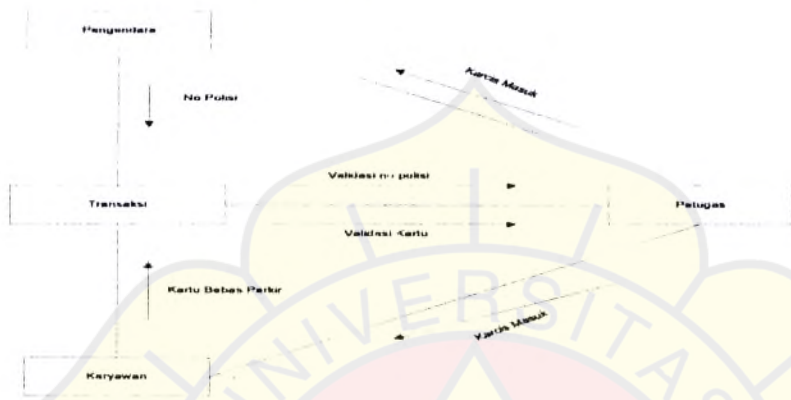
Sequence Diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem berupa *message* yang digambarkan terhadap waktu.



Gambar 2.5 contoh Sequence diagram

- *Collaboration Diagram*

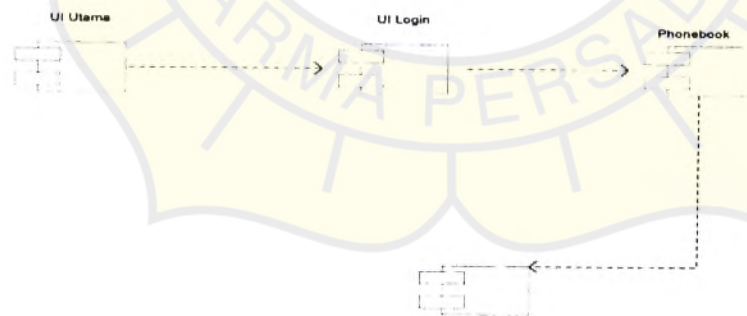
Collaboration Diagram juga menggambarkan interaksi antar objek seperti *sequence diagram*, tetapi menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian *message*.



Gambar 2.6 Contoh Collaboration Diagram

- *Component Diagram*

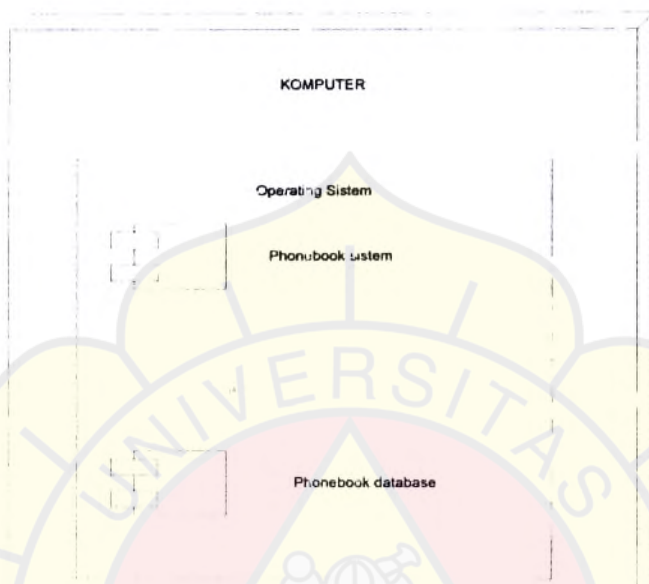
Component Diagram menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan di antaranya.



Gambar 2.7 contoh component diagram

- *Deployment Diagram*

Menggambarkan detail bagaimana komponen di deploy dalam infrastruktur sistem, dimana komponen akan terletak pada mesin server atau piranti keras.



Gambar 2.8 Contoh deployment diagram

2.5 Kamus Data

Kamus data adalah daftar yang menjelaskan elemen-elemen data yang terdapat didalam suatu yang dibuat berdasarkan arus data yang ada pada diagram aliran data. Menggunakan kamus data, analisa sistem dapat mendefinisikan data yang mengalir di sistem yang lengkap. Simbol-simbol kamus data menurut Jogiyanto HM(1995:726) adalah sebagai berikut:

Tabel 2.1 Kamus Data

| SIMBOL | KETERANGAN |
|--------|---|
| = | Terbentuk dari atau terdiri atau sama dengan |
| + | Dan |
| [] | Pilih salah satu dari (memilih satu dari elemen-elemen data dalam kurung) |
| { } | Iterasi (Pengulangan dari komponen yang ada di dalamnya) |
| () | Optional, komponen yang ada didalamnya bias ada atau bias tidak ada |
| / | Untuk memisahkan beberapa alternatif pilihan yang ada didalam Iterasi |
| * * | Komentar dari komponen yang ada didalamnya |
| ” “ | Harga Diskrit |

2.6 Spesifikasi Proses

Digunakan untuk menguraikan apa yang terjadi di dalam kotak proses yang ada di dalam data flow diagram. Spesifikasi proses adalah penjelasan tentang informasi aliran data masukan menjadi aliran data keluaran. Spesifikasi proses pada tingkat yang paling rendah pada umumnya ditulis dengan menggunakan BIT (Bahasa Indonesia Terstruktur) atau *pseudocode*. Semakin tinggi tingkatnya, semakin bersifat ikhtisar dari suatu spesifikasi proses.

2.7 Pengertian Sistem Informasi sewa parkir

Menurut pendapat Porter dan Perry (1989.P.117) pengendalian sistem informasi akan mengurangi frekuensi yang akan diperkirakan dari kejadian yang tidak diinginkan atau rata-rata kerugian yang diakibatkan oleh kejadian tersebut.

Apabila sistem dilengkapi dengan sistem pengendalian yang baik, yang berguna untuk mencegah atau menjaga hal-hal yang negatif, maka sistem tersebut akan dapat terus melangsungkan hidupnya. Suatu sistem harus dapat melindungi dirinya sendiri dari definisi tersebut di atas, dapat disimpulkan pengendalian sistem informasi dibagi atas :

a. Pengendalian secara umum

Merupakan pengendalian akurasi yang mempunyai tujuan untuk menjaga keamanan harta kekayaan milik perusahaan.

b. Pengendalian Aplikasi

Pengendalian aplikasi meliputi pengendalian masukan, pengolahan dan pengeluaran.

Sistem perhitungan sewa parkir ini merupakan suatu struktur interaksi antara manusia, peralatan, metode-metode, dan kontrol-kontrol yang disusun untuk mencapai tujuan kegiatan jasa sewa parkir dan menyediakan laporan-

laporan mengenai keadaan sewa parkir seperti laporan sewa masuk kendaraan, laporan keluar kendaraan, perkiraan biaya sewa parkir dan lain-lain diantara tujuan yang ingin dicapai tersebut adalah :

1. Mendukung rutin kerja untuk bagian perparkiran merekam data yang berhubungan dengan kegiatan sewa parkir. Mendukung pengambilan keputusan khususnya personil-personil yang terkait dengan kegiatan parkir.
2. Mendukung laporan-laporan internal dan laporan eksternal yaitu dengan memberikan laporan mengenai biaya sewa.

Perhitungan parkir pada Pasar JatiAsih yang sedang berjalan adalah sebagai berikut :

1. Motor, dikenakan biaya Rp. 2.000,00,- per kendaraan untuk 1 kali masuk dan tidak dikenakan biaya perjam berikutnya.
2. Mobil, dikenakan biaya Rp.3.000,00,- per kendaraan untuk 1 kali masuk dan tidak dikenakan biaya perjam berikutnya.

Perhitungan parkir pada Pasar JatiAsih yang diusulkan adalah sebagai berikut :

1. Dikenakan biaya Rp. 1.000,00,- per jam untuk setiap kendaraan.
2. Kendaraan pegawai tidak dikenakan biaya, tetapi karyawan wajib lapor nomor polisi kendaraan masing-masing karyawan akan

tersimpan di dalam database sistem parkir dan mendapatkan kartu bebas parkir.

3. Kendaraan bongkar muatan barang-barang pasar dan kendaraan Taxi tidak dikenakan biaya.

2.8 Sejarah singkat *Visual basic*

Sejarah singkat Visual basic diawali dengan pengembangan bahasa Basic (*Beginners All- Purpose Symbolic Instruction Code*) di Darmouth college AS. Bahasa basic dirancang pada tahun 1950an dan ditunjuk untuk dipergunakan oleh para programmer pemula dalam mempelajari bahasa pemrograman. Pada tahun 1955, komputer pribadi (PC) bermerk Mits Altair dibuat untuk pertama kalinya, namun bekerjanya komputer ini sangat lambat karena hanya memiliki RAM 4 kb yang hanya bisa menggunakan assembly dalam pemrogramannya.

Bill Gates dan Paul Alen kemudian menulis bahasa pemrograman Basic untuk Altair yang ternyata mendapat sambutan yang sangat besar dari masyarakat pengguna komputer. Keduanyaapun mendirikan Microsoft, sebuah perusahaan besar pembuat perangkat lunak bagi PC.

Pada tahun 1982, microsoft membuat sistem operasi MS-DOS bagi komputer IBM / PC yang didalamnya disertakan pula bahasa pemrograman Basic yang dikenal sebagai Quick Basic (Qbasic), dengan semakin

meningkatnya kebutuhan sistem operasi yang handal, pada tahun 1983, microsoft mulai mengembangkan sistem operasi berbasis windows yang memiliki kemampuan untuk menjalankan program secara bersamaan (Multi tasking) dan memiliki tampilan grafik yang handal.

Tampilan grafik Windows yang lebih interaktif mengubah pemrograman menjadi sebuah pekerjaan yang menyenangkan sehingga microsoft akhirnya membuat Basic versi Windows, yang diberi nama Microsoft Visual Basic.

Bahasa pemrograman *Visual Basic* memiliki beberapa perbedaan dengan bahasa pemrograman Basic konvensional yang menggunakan sistem operasi DOS. Perbedaan utama yang dapat dilihat yaitu pada saat perancangan program, pada *Visual Basic* lebih dulu merencanakan tampilan keluaran (*Output*) sebelum menuliskan pemrograman, hal ini dapat dilakukan karena *Visual Basic* menggunakan *windows* sebagai sistem operasinya yang memungkinkan pemrograman menggunakan tampilan grafis sebagai alat komunikasi dengan pemakai (*User*) atau disebut juga GUI (*Grafical User Interface*).

Beberapa keistimewaan dari *Visual Basic*:

- 1) Menggunakan *platform* pembuatan program yang diberi nama developer studio yang memiliki tampilan dan sarana yang sama dengan *Visual C++*

dengan begitu dapat bermigrasi atau belajar bahasa pemrograman lainnya dengan mudah, cepat dan tanpa belajar harus dari nol lagi.

- 2) Mempunyai compiler yang handal yang dapat mengklasikan file executable yang lebih cepat dan lebih efisien dari sebelumnya.
- 3) Memiliki kemampuan membuat *ActiveX* dan fasilitas internet yang lebih banyak.
- 4) Memiliki beberapa tambahan sarana *wizard* yang baru. *Wizard* adalah sarana yang mempermudah didalam pembuatan aplikasi dengan menngotomitasikan tugas-tugas tertentu.
- 5) Sarana akses data yang lebih cepat dan handal untuk membuat aplikasi database yang berkemampuan tinggi.

Layar *Visual Basic* adalah suatu lingkungan besar yang terdiri dari beberapa bagian-bagian kecil yang kesemuanya memiliki sifat :

- *Floating*

Floating yaitu dapat digeser-geser ke posisi mana saja. Untuk menggeser elemen layar *Visual Basic*, klik dan tahan tombol mouse pada judul (*Title Bar*) elemen tersebut, lalu geserlah ke tempat yang diinginkan.

- *Sizeable*

Sizeable yaitu dapat diubah-ubah ukurannya, seperti mengubah ukuran jendela *Windows*. Untuk mengubah ukuran suatu elemen atau jendela, klik dan tahan tombol mouse pada sisi (*border*) jendela tersebut, lalu geser hingga ke ukuran yang diinginkan.

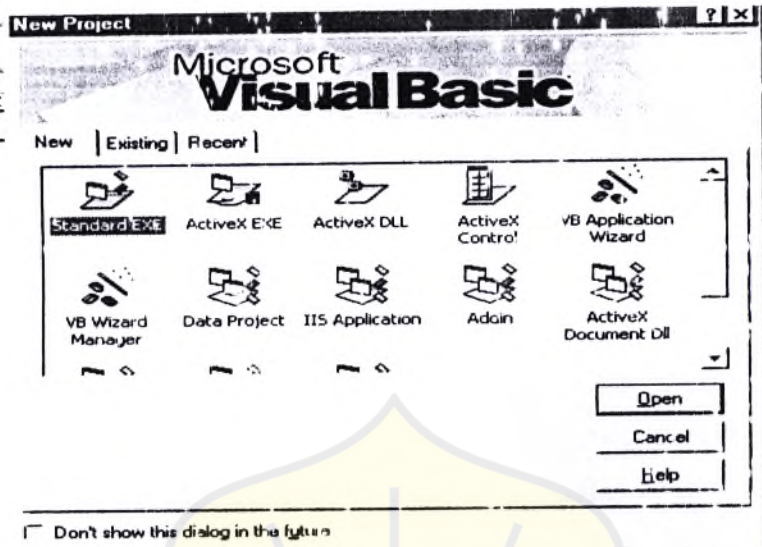
- *Dockable*

Dockable yaitu dapat menempel dengan bagian lain yang berdekatan. Untuk menempelkan elemen layar *Visual Basic* ke elemen lainnya, cukup tempelkan sisi-sisi elemen tersebut dan secara otomatis akan menempel ke tempat yang diinginkan.

2.8.1 Cara Memulai *Visual Basic 6.0*

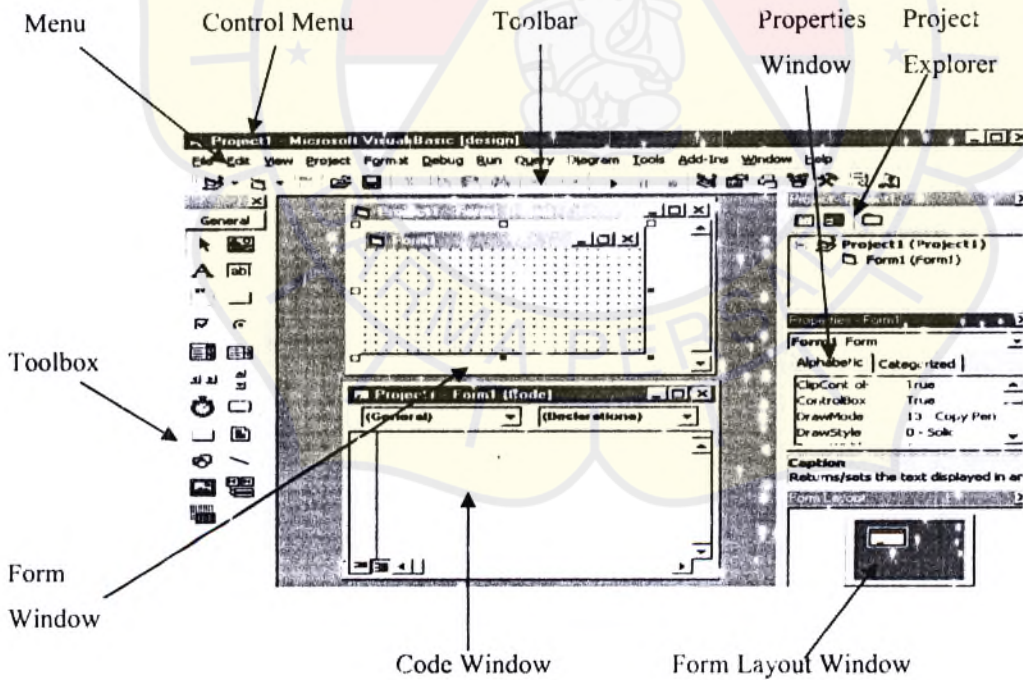
Sebelum mulai membuat suatu program aplikasi, sebaiknya kita harus mengetahui terlebih dahulu bagaimana cara untuk menjalankan *Visual Basic 6.0*, yaitu :

- a. Klik start, kemudian sorot Programs (*Windows 98*) atau *All Program (Windows XP)*.
- b. Maka akan keluar semua seluruh program yang ada dalam *Windows*, kemudian sorot *Microsoft Visual Studio 6.0*.
- c. Kemudian akan keluar tujuh pilihan, klik *Microsoft Visual Basic 6.0*.
- d. Maka akan keluar kotak dialog *New Project* seperti pada gambar dibawah ini.
- e. Klik *open* setelah anda memilih bentuk dari tampilan *Visual Basic* (biasanya menggunakan *standard EXE*).
- f. Setelah itu akan tampil jendela dari *Visual Basic 6.0*



Gambar 2.9 Kotak Dialog *New Project*

2.8.2 Komponen – Komponen Pada *Visual Basic 6.0*



Gambar 2.10 Layar Visual Basic

Komponen–komponen pada *Visual Basic* menurut Erhans A., berupa :

1. Control Menu

Control menu merupakan menu yang digunakan untuk memanipulasi jendela *Visual Basic*. Dalam menu ini dapat mengubah ukuran memindahkan atau menutup jendela *Visual Basic* atau jendela *Windows* lainnya.

2. Menu

Menu berisi semua perintah *Visual Basic* yang dapat dipilih untuk melakukan tugas tertentu. Isi dari menu ini sebagian layar sama dengan

File Edit View Project Format Debug Run Query Diagram Tools Add-Ins Window Help

program–program *Windows* pada umumnya. Untuk memilih menu caranya juga sama dengan program *Windows* lainnya.

Gambar 2.11 Menu


3. Toolbar








Toolbar merupakan tombol–tombol yang mewakili suatu perintah tertentu dari *Visual Basic*. Setiap tombol tersebut dapat langsung di klik untuk melakukan perintah tertentu. Biasanya tombol–tombol ini merupakan perintah–perintah yang sering digunakan dan terdapat pula pada menu *Visual Basic*.



Gambar 2.12 Toolbar

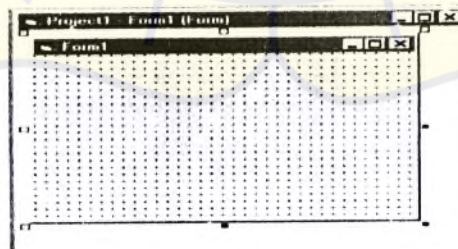
Tabel 2.2 Toolbar

| Toolbar | Nama | Fungsi |
|---|---------------------------|--|
|  | <i>Add Project</i> | menambah proyek ke dalam proyek yang sudah ada |
|  | <i>Add Item</i> | menambahkan komponen atau objek ke dalam jendela form |
|  | <i>Menu Editor</i> | menampilkan Menu Editor untuk mengubah tampilan menu |
|  | <i>Open Project</i> | Membuka proyek Visual Basic yang sudah ada |
|  | <i>Save Project Group</i> | menyimpan proyek Visual Basic pada komputer |
|  | <i>Cut</i> | memotong elemen yang dipilih pada layar |
|  | <i>Copy</i> | meng-copy elemen yang dipilih pada layar |
|  | <i>Paste</i> | menyalin elemen yang sebelumnya sudah dipotong (<i>Cut</i>) atau disalin (<i>Copy</i>) |
|  | <i>Find</i> | mencari text tertentu |
|  | <i>Undo</i> | membatalkan perintah atau tindakan yang terakhir |
|  | <i>Redo</i> | mengulangi perintah atau tindakan terakhir yang dibatalkan |
|  | <i>Start</i> | menjalankan proyek yang dibuat pada Visual Basic |
|  | <i>Break</i> | menghentikan running program untuk sementara |
|  | <i>End</i> | menghentikan running program |

| | | |
|---|---------------------------------|--|
|  | <i>Project Explorer</i> | menampilkan jendela Project Explorer |
|  | <i>Properties Window</i> | menampilkan jendela properties |
|  | <i>Form Layout Window</i> | menampilkan jendela Form Layout Window |
|  | <i>Object Browser</i> | menampilkan jendela Objek Browser |
|  | <i>Toolbox</i> | menampilkan jendela Toolbox |
|  | <i>Data View Window</i> | menampilkan jendela Data View Window |
|  | <i>Visual Component Manager</i> | menampilkan jendela Visual Component Manager |

4. Form Window

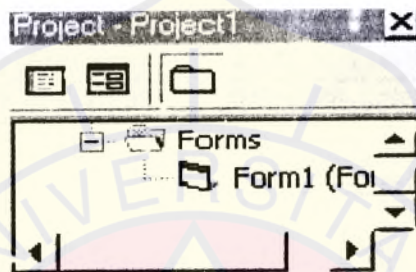
Form Window merupakan daerah kerja utama untuk “menggambar” atau merancang aplikasi. Pada form ini dapat diletakkan objek-objek: seperti text, gambar, scrollbar, dan lain-lain. Form tersebut dapat diperbesar dan diperkecil sesuai dengan ukuran yang diinginkan.



Gambar 2.13 Form

5. Project Explorer

Project Explorer merupakan jendela yang menampilkan daftar form dan modul proyek. Proyek merupakan kumpulan dari modul form, modul class, modul layar dan file sumber yang membentuk suatu aplikasi. Lokasinya terletak paling kanan atas jendela *Visual Basic*.



Gambar 2.14 Project Explorer

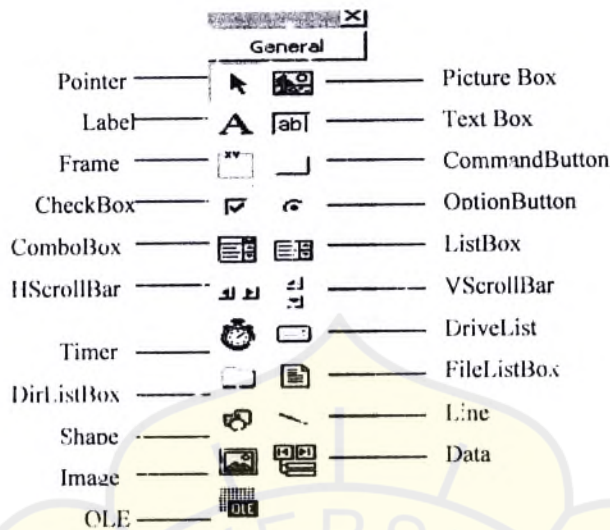
6. Toolbox

Toolbox merupakan kumpulan dari objek / icon, digunakan untuk membuat user interface serta layar bagi program aplikasi. Semua icon yang terdapat dalam Toolbox disebut *Controls* (layar).

Tabel 2.3 Toolbox

| Kontrol | Keterangan |
|-----------|---|
| Label | Menampilkan teks di form |
| Frame | Sebagai tempat pengelompokan bagi kontrol yang mempunyai fungsi sejenis |
| Combo Box | Pengguna program dapat memilih dari daftar yang telah ada atau |

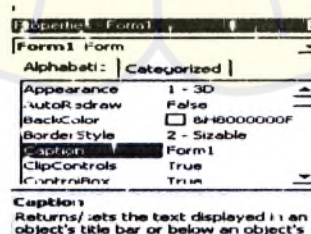
| | |
|---------------|---|
| | memilih sesuatu yang baru |
| Check Box | Memberi pilihan kepada pengguna program |
| HScroll Bar | Scrollbar horizontal untuk melihat daftar data di kontrol lain |
| Timer | Mengatur program melakukan proses sendiri dalam waktu tertentu tanpa berinteraksi dengan pengguna program |
| DirList Box | Memuat daftar direktori yang dapat dipilih pengguna program |
| Shape | Menampilkan shape di form |
| Image | Menampilkan grafik (image) di form tetapi tidak bisa sebagai kontainer |
| OLE container | Memampukan anda untuk menambahkan fungsi dari kontrol di program lain ke dalam program anda |
| PictureBox | Menampilkan grafik (image) di form dan dapat berfungsi sebagai kontainer |
| TextBox | Digunakan untuk menampilkan, memasukkan dan mengedit data |
| CommandButton | Digunakan oleh pengguna program untuk memulai sebuah proses |
| OptionButton | Digunakan untuk memilih dari kumpulan pilihan, minimal untuk dua pilihan |
| ListBox | Digunakan untuk memilih dari daftar list |
| VScrollBar | Scrollbar vertikal untuk melihat daftar data di kontrol lain |
| DriveListBox | Memuat daftar drive yang dapat dipilih pengguna program |
| FileListBox | Memuat daftar file yang dapat dipilih pengguna program |
| Line | Menampilkan garis di form |
| Data | Menghubungkan program anda ke sebuah database |



Gambar 2.15 Toolbox

7. Properties Windows

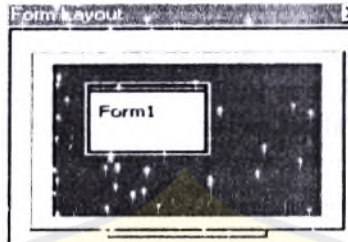
Properties Windows merupakan kotak dialog untuk membuat setting terhadap form, serta untuk membuat setting terhadap layar yang telah kita masukkan ke dalam form. Properties adalah sifat dari sebuah objek, misalnya nama, warna, ukuran, posisi, dan sebagainya. Di bagian paling atas terdapat kotak yang menunjukkan nama objek yang sedang aktif. Ada dua tampilan yaitu Alphanumeric (urut abjad) dan Categorized (urut berdasarkan kelompok).



Gambar 2.16 Properties Window

8. Form Layout Window

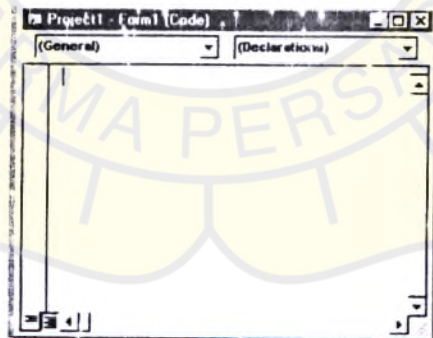
Form Layout Window merupakan jendela yang menggambarkan posisi dari form yang ditampilkan pada layar monit



Gambar 2.17 Form Layout Window

9. Code Window

Jendela kode merupakan jendela yang berisi kode program yang merupakan instruksi untuk aplikasi Visual Basic. Kode-kode program tersebut dapat digunakan untuk melakukan tugas-tugas tertentu, seperti menutup aplikasi, membatalkan perintah, dan lainnya.



Gambar 2.18 Code Window

2.9 MySQL

MySQL merupakan software sistem manajemen basis data SQL (bahasa Inggris: database management system) atau DBMS yang multithread dan multi-user. MySQL AB membuat MySQL tersedia sebagai software gratis dibawah lisensi GNU General Public License (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL. MySQL dimiliki dan disponsori oleh sebuah perusahaan komersial Swedia MySQL AB, dimana memegang hak cipta hampir atas semua kode sumbernya. Kedua orang Swedia dan satu orang Finlandia yang mendirikan MySQL AB adalah: David Axmark, Allan Larsson, dan Michael "Monty" Widenius. (MySql ilmukomputer.com 2003-2007)

2.9.1 Keistimewaan MySQL

Sebagai database server yang memiliki konsep database modern, MySQL memiliki banyak sekali keistimewaan. Berikut ini beberapa keistimewaan yang dimiliki oleh MySQL.

1. Portability

MySQL dapat berjalan stabil pada berbagai sistem operasi di antaranya adalah seperti Windows, Linux, FreeBSD, Mac OS X Server, Solaris, Amiga, HP-UX dan masih banyak lagi.

2. Open Source

MySQL didistribusikan secara *open source* (gratis), di bawah lisensi GPL sehingga dapat digunakan secara cuma-cuma tanpa dipungut biaya.

3. *Multiuser*

MySQL dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami masalah atau konflik. Hal ini memungkinkan sebuah database server MySQL dapat diakses client secara bersamaan.

4. *Performance Tuning*

MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.

5. *Column Types*

MySQL memiliki tipe kolom yang sangat kompleks, seperti signed/ unsigned integer, float, double, char, varchar, text, date, time, timestamp, year, set serta enum.

6. *Command dan Functions*

MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah SELECT dan WHERE dalam query.

7. *Security*

MySQL memiliki beberapa lapisan sekuritas seperti level subnetmask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta password terenkripsi.

8. *Scalability dan Limits*

MySQL mampu menangani database dalam skala besar, dengan jumlah records lebih dan 50 juta dan 60 ribu tabel selain itu, batas indeks yang dapat

ditampung mencapai 32 indeks pada tiap tabelnya.

9. Connectivity

MySQL dapat melakukan koneksi dengan client menggunakan protokol TCP/IP, Unix soket (Unix), atau Named Pipes (NF).

10. Clients dan Tools

MySQL dilengkapi dengan berbagai tool yang dapat digunakan untuk administrasi database. dan pada setiap tool yang ada disertakan petunjuk online.

2.9.2 Tipe Data di MySQL

SQL menyediakan beberapa jenis tipe data dasar: boolean, bilangan bulat, bilangan pecahan desimal, teks, data biner, tanggal / waktu. Sebenarnya standar SQL:1999 menyediakan beberapa tipe data yang lain, seperti tipe data komposit (array dan row) serta tipe data ref/ pointer. Standar SQL:2003 pun menambahkan tipe data seperti MULTISSET (yang mirip array) dan XML. Namun banyak database, termasuk MySQL, yang tidak atau belum mendukungnya. Tipe data yang ada di Mysql sebagai berikut:

- Boolean
- Bilangan Bulat
- Bilangan Pecahan Desimal
- Teks
- Data Tunggal dan Waktu
- Data geometri
- Data Biner