BAB II

TINJAUAN TEORI

Pada bab ini, penulis membahas tentang definisi-definisi yang terkait dengan penelitian yang dilakukan

2.1 PENGEMBANGAN SISTEM

Suatu organisasi dalam melakukan proses pencapaian tujuan memerlukan suatu sistem yang dinamis sehingga pembuat keputusan dapat melakukan pengembangan secara terarah. Teori yang membahas tentang pengembangan tersebut dapat dilihat dari pejelasan berbagai sumber, antara lain :

2.1.1 Definsi Pengembangan

menurut [Jog99] *" Analisis dan Desain Sistem Informasi"* , Pengembangan sistem dapat berati menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada. Sistem yang lama perlu diperbaiki atau diganti disebabkan beberapa hal, yaitu :

1. Adanya permasalahan-permasalahan yang timbul di sistem yang lama, permasalahan yang timbul dapat berupa :

a. Ketidakberesan

Ketidakberesan dalam sistem yang lama menyebabkan sistem yang lama tidak dapat beroperasi sesuai dengan yang diharapkan. Ketidakberesan ini dapat berupa:

1. Kecurangan-kecurangan disengaja yang menyebabkan tidak amannya harta kekayaan perusahaan dan kebenaran dari data menjadi kurang terjamin.
2. Kesalahan-kesalahan yang tidak disengaja yang juga dapat menyebabkan kebenaran data kurang terjamin.
3. Tidak efisiennya operasi.
4. Tidak ditaatinya kebijaksanaan manajemen yang telah ditetapkan.

b. Pertumbuhan organisasi

Pertumbuhan organisasi diantaranya adalah kebutuhan informasi yang semakin luas, volume pengolahan data semakin meningkat, perubahan prinsip akuntansi yang baru. Karena adanya perubahan ini, maka menyebabkan sistem yang lama tidak efektif lagi.

2. Untuk meraih kesempatan-kesempatan

Keadaan pasar bersaing, kecepatan informasi atau efisiensi waktu sangat menentukan berhasil tidaknya strategi atau rencana-rencana yang telah disusun untuk meraih kesempatan-kesempatan yang ada. Kesempatan-kesempatan dapat berupa peluang-peluang pasar, pelayanan yang meningkat kepada langganan.

2.1.2 Prinsip Pengembangan Sistem

Mengenai prinsip pengembangan system, [Jog99] menambahkan bahwa :

1. Sistem yang dikembangkan adalah untuk manajemen
2. Sistem yang dikembangkan adalah investasi modal yang besar, Investasi modal harus mempertimbangkan 2 hal :

1. Semua alternatif yang ada harus diinvestigasi
2. Investasi yang terbaik harus bernilai

3. Sistem yang dikembangkan memerlukan orang yang terdidik
4. Proses pengembangan sistem tidak harus urut
5. Jangan takut membatalkan proyek
6. Dokumentasi harus ada untuk pedoman dalam pengembangan sistem

2.1.3 Tahapan Pengembangan sistem

menurut [Jog99] *" Analisis dan Desain Sistem Informasi"* ,Tahapan utama siklus hidup Pengembangan Sistem terdiri dari :

1. Perencanaan *(Systems Planning)*
2. Analisis *(System Analysis*)
3. Perancangan *(Systems Design)* secara umum
4. Seleksi *(System Selection)*
5. Perancangan *(Systems Design)* secara khusus
6. Implementasi dan Pemeliharaan *(System Implementation & Maintenance)*

2.1.4 Metode Pengembangan Sistem

Metode adalah suatu cara atau teknik sistematis untuk mengerjakan sesuatu. Metodologi adalah metode-metode, prosedur-prosedur, konsep-konsep pekerjaan, aturan-aturan yang akan digunakan sebagai pedoman bagaimana dan apa yang harus dikerjakan selama proses pengembangan sistem [Jog99].

Model prototipe (prototyping model), merupakan suatu teknik untuk mengumpulkan informasi tertentu mengenai kebutuhan-kebutuhan informasi

pengguna secara cepat. Tahapan yang dilakukan :

1Reaksi awal dari pengguna, diawali dengan menampilkan sebuah prototipe sistem informasi, kemudian melihat reaksi dari pengguna saat bekerja dengan prototipe apakah fitur-fitur sistem pada prototipe tersebut sudah sesuai dengan kebutuhannya. Reaksi tersebut dikumpulkan dalam lembar observasi, wawancara dan kuesioner.

2Saran-saran pengguna, saran-saran merupakan hasil interaksi pengguna dengan prototipe yang ditampilkan (evaluasi pengguna) yang merupakan masukan untuk perbaikan, pengubahan atau menghentikan prototipe sehingga dapat memenuhi kebutuhan pengguan dengan lebih baik.

3Inovasi, adalah kemampuan-kemampuan sistem baru yang sebelumnya tidak ada pada saat pengguna berinteraksi dengan prototipe. Inovasi prototipe jika berhasil akan menjadi bagian dari sistem hasil jadi.

4Rencana revisi, prototipe menggambarkan sistem di masa datang. Rencana revisi membantu mengidentifikasikan prioritas-prioritas apa saja yang akan diprototipekan selanjutnya.

2.2 Pengertian Penduduk

Penduduk, peraturan menteri dalam negeri no 28 tahun 2005, adalah WNI dan orang asing yang masuk secara serta bertempat tinggal di wilayah Indonesia sesuai perundang-undangan

Menurut UU RI no 23 tahun 2006, menyatakan bahwa :

1. Administrasi kependudukan adalah rangkaian kegiatan penataan dan penertiban dalam penerbitan dokumen dan data kependudukan melalui

pendaftaran penduduk, pencatatan sipil, pengelolaan informasi administrasi kependudukan serta pendayagunaan hasilnya untuk pelayanan publik dan pembangunan sektor lain.

2. Penduduk adalah warga Negara Indonesia dan orang asing yang bertempat tinggal di Indonesia
3. Warga Negara Indonesia (WNI) adalah orang-orang bangsa Indonesia asli dan orang-orang bangsa lain yang disahkan denga Undang – Undang sebagai warga Negara Indonesia
4. Orang asing adalah orang yang bukan warga Negara Indonesia
5. Menteri adalah menteri yang bertanggung jawab dalam urusan pemerintahan dalam negeri
6. Penyelenggara adalah pemerintah, pemerintah Provinsi dan pemerintah Kabupaten / kota yang bertanggungjawab dan berwenang dalam urusan administrasi kependudukan
7. Instansi pelaksana adalah perangkat pemerintah kabupaten / kota yang bertanggungjawab dan berwenang melaksanakan pelayanan dalam urusan administrasi kependudukan
8. Dokumen kependudukan adalah dokumen resmi yang diterbitkan oleh instansi pelaksana yang mempunyai kekuatan hukum sebagai alat bukti autentik yang dihasilkan dari pelayanan pendaftaran penduduk dan pencatatan sipil
9. Data kependudukan adalah data perseorangan dan atau data agregat yang terstruktur sebagai hasil dari kegiatan pendaftaran penduduk dan pencatatan sipil

2.3 Konsep Dasar Sitem Informasi

Pengertian sistem informasi dapat dilihat dari definisi beberapa sumber yang menyatakan pengertian tersebut

1.1 Pengertian Sistem

Dalam memahami suatu sistem diperlukan pemahaman mengenai sistem itu sendiri. Terdapat beberapa pendekatan dalam mendefinisikan sistem. Pendekatan yang lebih menekankan pada prosedur, [Jog95] menerangkan bahwa :

> "Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu".

Sedangkan pendekatan lain yang lebih menekankan pada komponen atau elemen, menurut [Jog95] menambahkan bahwa :

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

Jadi dapat disimpulkan sistem itu adalah suatu cara atau metode yang digunakan untuk menyelesaikan masalah agar memudahkan dalam pencapaian tujuan.

2.3.2 Pengertian Informasi

Pengertian informasi menurut [Jog95] adalah :

> "Informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian (event yang nyata (fact) yang digunakan untuk pengambilan keputusan".

2.3.3Sistem Informasi

Pengertian sistem informasi, [Teg95] menerangkan :

> "Sistem informasi adalah suatu cara tertentu untuk menyediakan informasi yang dibutuhkan oleh organisasi untuk beroperasi dengan cara yang sukses dan untuk organisasi bisnis dengan cara yang menguntungkan".

Sedangkan Robert A. Leitch dan K. Roscoe Davis mendefinisikan Sistem informasi sebagai berikut :

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

Jadi dapat disimpulkan bahwa sistem informasi itu pada dasarnya merupakan suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam sebuah wadah organisasi yang saling berinteraksi untuk mencapai tujuan yaitu menyajikan informasi yang sedang dibutuhkan.

2.4 Arsitektur Aplikasi

Arsitektur aplikasi diartikan sebagai suatu rancangan dalam membuat aplikasi sehingga dapat mendukung terbentuknya suatu sistem informasi berupa jaringan komputer dengan berbagai jenis topologi dan manfaat yang dihasilkan.

2.4.1 Jaringan Komputer

Jaringan komputer adalah sekumpulan komputer yang berhubungan antara yang satu dengan yang lainnya menggunakan *protocol* komunikasi sehingga dapat saling bertukar data dan informasi serta dapat menggunakan perangkat keras secara bersama. Selain itu jaringan komputer dapat diartikan sebagai sekumpulan terminal komunikasi yang berada di berbagai lokasi berbeda yang terdiri dari dua komputer atau lebih yang saling berhubungan [Bud05].

2.4.2 Jenis – jenis Jaringan

Menurut [Bud05]*"Jaringan Komputer"*, Berdasarkan jarak dan area kerjanya jaringan komputer dibedakan menjadi tiga kelompok, salah satunya yaitu:

1. Local Area Network

Local Area Network (LAN) adalah jaringan yang dibatasi oleh area yang relatif kecil, umumnya dibatasi oleh area lingkungan seperti perkantoran di sebuah gedung atau sebuah sekolah dan biasanya jaraknya tidak lebih dari 1 km. Beberapa model konfigurasi LAN biasanya berupa satu komputer yang dijadikan *file server* yang digunakan untuk menyimpan perangkat lunak (*Software* yang mengatur aktifitas jaringan) serta beberapa komputer yang terhubung ke *file server* yang disebut dengan *workstation*. LAN terbagi menjadi dua model hubungan, yaitu :

a. *Peer-To-Peer*

Model hubungan *peer-to-peer* memungkinkan *user* membagi sumber daya baik itu berupa *file*, layanan printer atau lainnya. Namun model ini tidak mempunyai *file server* atau sumber daya yang terpusat. Di dalam model *peer-to-peer* ini, seluruh komputer adalah sama, berarti mempunyai hak yang sama untuk memakai sumber daya yang tersedia di dalam jaringan. Model ini didesain untuk jaringan bersekala kecil dan menengah.

b. *Client-Server*

Model hubungan ini memungkinkan jaringan untuk mensentralisasi fungsi atau aplikasi kepada satu atau lebih *dedicated file server.* Sebuah *file server* menjadi jantung bagi keseluruhan sistem, memungkinkan untuk mengakses sumber daya, dan menyediakan keamanan. Model hubungan ini menyediakan mekanisme untuk mengintegrasikan seluruh komponen yang ada pada jaringan dan memungkinkan banyak pengguna secara bersama-sama memakai sumber daya pada *file server*.

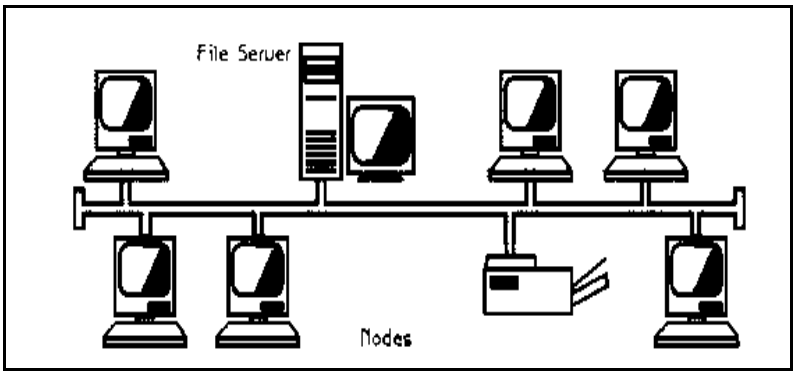
2.4.3 Topologi Jaringan

Topologi secara fisik dari suatu jaringan lokal adalah merujuk kepada konfigurasi kabel, komputer serta perangkat lainnya. Tipe-tipe utama topologi jaringan :

1. *Linear bus*

Topologi *linear bus* terdiri dari satu jalur kabel utama, dimana pada masing-masing ujungnya diberikan sebuah *terminator*. Semua *nodes* pada jaringan (*file*

server, *workstation*, dan perangkat lainnya) terkoneksi pada kabel utama (*backbone*). Jaringan *Ethernet* dan *local talk* mengunakan topologi ini.

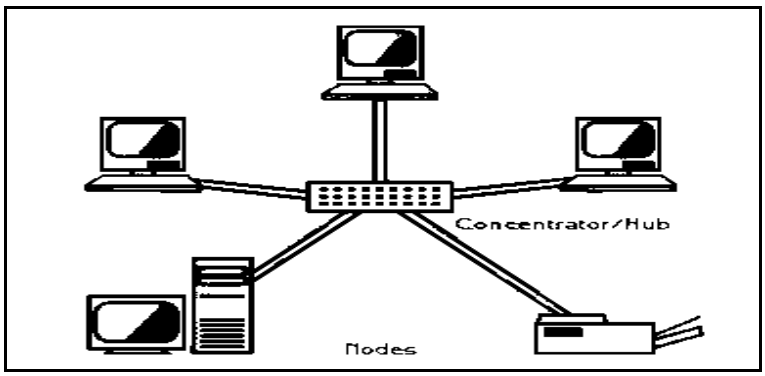


Gambar 2.1 Topologi Linear Bus

[Sumber : http://fcit.coedu.usf.edu/network/chap5/chap5.htm]

2. *Star*

Topologi model ini dirancang dimana setiap *nodes* terkoneksi ke jaringan melewati sebuah *concentrator*. Data yang dikirim ke jaringan lokal akan melewati *concentrator* sebelum melanjutkan ke tempat tujuannya. *Concentrator* akan mengatur dan mengendalikan keseluruhan fungsi jaringan, dan juga bertindak sebagai *repeater* (penguat aliran data). Konfigurasi pada model ini menggunakan kabel *twisted pair*, dan dapat pula digunakan kabel *coaxial* atau kabel *fiber optic*

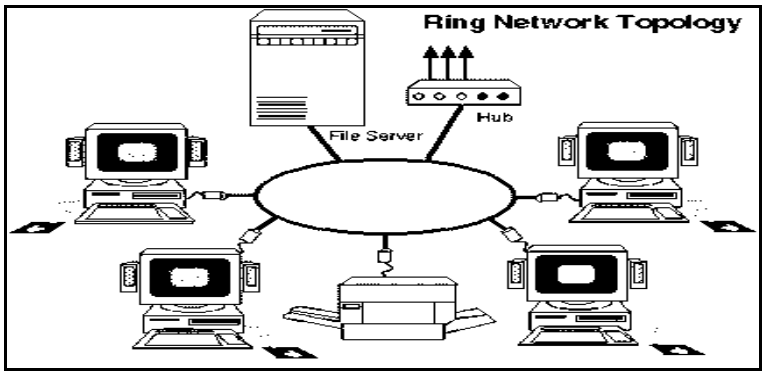


Gambar 2.2 Topologi Star (Bintang)

[Sumber : http://fcit.coedu.usf.edu/network/chap5/chap5.htm]

3. *Ring*

Pada dasarnya topologi ini menggunakan teknik konfigurasi yang sama dengan topologi *star*, tetapi pada topologi ini terlihat bahwa jalur media transmisi menyerupai suatu lingkaran tertutup seperti cincin (*ring*).



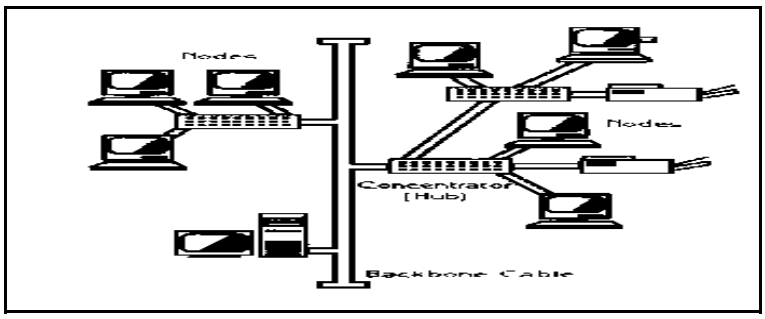
Gambar 2.3 Topologi Ring

[Sumber :

http://myschoolnet.ppk.kpm.my/tutor_it/rangkai/topologi/cincin.htm]

4. *Tree*

Topologi model ini merupakan perpaduan antara topologi *linear bus*, dan *star* yang terdiri dari kelompok-kelompok *workstation* dengan konfigurasi *star* yang terkoneksi ke kabel utama yang menggunakan topologi *linear bus*. Topologi ini memungkinkan untuk pengembangan jaringan yang telah ada.



Gambar 2.4 Topologi Tree

[Sumber : http://fcit.coedu.usf.edu/network/chap5/chap5.htm]

2.4.4 Manfaat Jaringan Komputer

Menurut [Bud05]*" Jaringan Komputer"*, Manfaat dalam membangun jaringan komputer adalah sebagai berikut:

1. *Sharing Resource*

Bertujuan agar seluruh program, peralatan atau peripheral lainnya dapat dimanfaatkan oleh setiap orang yang ada pada jaringan computer tanpa terpengaruh oleh lokasi maupun pengaruh dari pemakai.

2. Media Komunikasi

Jaringan komputer memungkinkan terjadinya komunikasi antar pengguna, baik untuk teleconference maupun untuk mengirim pesan atau informasi yang penting lainnya. Dengan menggunakan jaringan computer, dua orang atau lebih yang jaraknya sangat jauh akan lebih mudah bekerja sama.

3. Integrasi Data

Pembangunan jaringan komputer dapat mencegah ketergantungan pada komputer pusat.

4. Pengembangan dan Pemeliharaan

Dengan adanya jaringan komputer, maka pengembangan peralatan dapat dilakukan dengan mudah dan menghemat biaya.

5. Keamanan Data

Sistem jaringan komputer memberikan perlindungan terhadap data.

6. Sumber Daya Lebih Efisien dan Informasi Terkini

Dengan adanya pemakaian sumber daya secara bersama-sama, maka pemakai bisa mendapatkan hasil yang maksimal dan dengan kualitas yang tinggi.

2.5 Sistem *Client-Server*

Sistem *client server* mempunyai dua komponen utama yaitu komputer *client* dan komputer *server*. *Server* merupakan komputer induk yang melakukan pemrosesan terbanyak untuk memenuhi permintaan-permintaan dari komputer *client* dan bertindak *server database* yang menyimpan data.

Client yaitu komputer atau *workstation* yang melakukan pengiriman permintaan-permintaan data pada *server* kemudian menampilkan data tersebut pada *interface* aplikasi yang dimilikinya. Selain itu *client* komputer yang melibatkan proses-proses *client* yang meminta suatu pelayanan data kepada komputer *server* yang menyediakan layanan data tersebut. Sehingga *client* maupun *server* sama-sama melakukan pekerjaan.

Dengan adanya kombinasi *client* (*front-end*) dan *server* (*back-end*) ini, maka kumpulan dari modul-modul program tidak dieksekusi dalam memori yang sama namun terbagi dalam komputer *client-server*. Dengan arsitektur yang seperti ini, modul yang memanggil menjadi *client* yang meminta suatu layanan data dan modul yang dipanggil menjadi *server* yang menyediakan layanan data.

2.6 **UML (*Unified Model Language*)**

Buku UML yang penulis gunakan sebagai pegangan dalam menysun skripsi ini adalah menurut [Mar04], yang akan dijelaskan pada beberapa Subbab dibawah ini

2.6.1 Pengertian UML

Unified Modelling Language (UML) adalah sebuah "bahasa" yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. UML mendefinisikan notasi dan *syntax*/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML *syntax* mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi UML terutama diturunkan dari 3 notasi yang telah ada sebelumnya: Grady Booch OOD (Object-Oriented Design), Jim Rumbaugh OMT (*Object Modeling Technique*), dan Ivar Jacobson OOSE (*Object-Oriented Software Engineering*).

UML Menggabungkan konsep berarah objek seperti Booch, OMT, OOSE, dsb sehingga menjadi bahasa pemodelan tunggal. UML menekankan pada apa yang dapat dikerjakan dengan metode-metode tersebut. UML berfokus pada suatu bahasa pemodelan standar bukan pada proses standar.

2.6.2 Bagian-bagian UML

UML memiliki bagian-bagian yang dapat mempermudah pembuatan sistem diantaranya :

a. ***Use case diagram****, Use case diagram* menggambarkan interaksi antara *actor* dengan proses atau sistem yang dibuat. *Use case diagram* mempunyai beberapa bagian penting seperti: *Actor, Use Case, Undirectional Association , Generalization*

b. **Class *diagram***, *class diagram* menggambarkan interaksi antar *class* serta atribut-atribut yang melekat pada *class* tersebut.

c. ***Sequence diagram*****, *Sequence diagram*** merupakan interaksi antara objek-objek dalam suatu sistem dan terjadi komunikasi yang berupa pesan (*message*) serta parameter waktu.

d. ***Collaboration Diagram***, *Collaboration Diagram* hampir sama dengan *sequence diagram* tetapi berbeda pada objek yang di titik tekankan, *collaboration* lebih menekankan pada pemunculan objek itu sendiri sedangkan *sequence diagram* lebih pada penyampaian *message* dengan parameter waktu.

e. ***Activity diagram*****, Activity *diagram*** menggambarkan proses-proses yang terjadi mulai aktivitas dimulai sampai aktivitas berhenti.

f. ***Statechart Diagram*****,** *Statechart Diagram* merupakan diagram yang menggambarkan perubahan *state* dari *state* yang satu ke *state* lainnya.

g. ***Component diagram,*** *Component diagram* menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (*dependency*) di antaranya. Komponen piranti lunak adalah modul berisi *code*, baik berisi *source code* maupun *binary code*, baik *library* maupun *executable*, baik yang muncul pada *compile time, link time*, maupun *run time*. Umumnya komponen terbentuk dari beberapa *class* dan/atau *package*, tapi dapat juga dari komponen-komponen yang lebih kecil. Komponen dapat juga berupa *interface*.

h. ***Deployment/physical diagram,*** *Deployment diagram* menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, *server* atau piranti keras apa),bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi *server*, dan hal-hal lain yang bersifat fisikal.

2.6.3 Perbedaan UML dengan bahasa Pemodelan yang lain

1. **UML sebagai Bahasa Pemodelan**

UML merupakan bahasa pemodelan yang memiliki pembendaharaan kata dan cara untuk mempresentasikan secara fokus pada konseptual dan fisik dari suatu sistem. Contoh untuk sistem software yang intensive membutuhkan bahasa yang menunjukkan pandangan yang berbeda dari arsitektur sistem, ini sama

seperti menyusun atau mengembangkan *software development life cycle*. Dengan UML akan memberitahukan kita bagaimana untuk membuat dan membaca bentuk model yang baik, tetapi UML tidak dapat memberitahukan model apa yang akan dibangun dan kapan akan membangun model tersebut. Ini merupakan aturan dalam *software development process*.

2. UML sebagai bahasa untuk Menggambarkan Sistem (Visualizing)

UML tidak hanya merupakan rangkaian simbol grafikal, cukup dengan tiap simbol pada notasi UML merupakan penetapan semantik yang baik. Dengan cara ini, satu pengembang dapat menulis model UML dan pengembang lain atau perangkat yang sama lainnya dapat mengartikan bahwa model tersebut tidak ambigu. Hal ini akan mengurangi *error* yang terjadi karena perbedaan bahasa dalam komunikasi model konseptual dengan model lainnya.

UML menggambarkan model yang dapat dimengerti dan dipresentasikan ke dalam model tekstual bahasa pemograman. Contohnya kita dapat menduga suatu model dari sistem yang berbasis web tetapi tidak secara langsung dipegang dengan mempelajari *code* dari sistem. Dengan model UML maka kita dapat memodelkan suatu sistem web tersebut dan direpresentasikan ke bahasa pemrograman. UML merupakan suatu model eksplisit yang menggambarkan komunikasi informasi pada sistem. Sehingga kita tidak kehilangan informasi code implementasi yang hilang dikarenakan developer memotong coding dari implementasi.

3. UML sebagai bahasa untuk Menspesifikasikan Sistem (Specifying)

Maksudnya membangun model yang sesuai, tidak ambigu dan lengkap. Pada faktanya UML menunjukan semua spesifikasi keputusan analisis, desain dan implementasi yang penting yang harus dibuat pada saat pengembangan dan penyebaran dari system software intensif.

4. UML sebagai bahasa untuk Membangun Sistem (Constructing)

UML bukan bahasa pemograman visual, tetapi model UML dapat dikoneksikan secara langsung pada bahasa pemograman visual. Maksudnya membangun model yang dapat dimapping ke bahasa pemograman visual atau tabel pada database relational atau penyimpanan tetap pada database berorientasi object.

5. UML sebagai bahasa untuk Pendokumentasian Sistem (Documenting)

Maksudnya UML menunjukan dokumentasi dari arsitektur sistem dan detail dari semuanya.UML hanya memberikan bahasa untuk memperlihatkan permintaan dan untuk tes. UML menyediakan bahasa untuk memodelkan aktifitas dari perencanaan *project* dan manajemen pelepasan (release management).

2.6.3 Metode pengujian

Dalam penelitian ini, penulis melakukan pengujian secara Black Box. Pengujian BlackBox menekankan pada aspek fundamental sistem tanpa memperhatikan struktur logika internal perangkat lunak. Pengujian Black Box berusaha menemukan kesalahan dalam kategori :

1. fungsi_fungsi yang tidak benar.
2. kesalahan interface
3. kesalahan dalam struktur data atau akses database eksternal
4. kesalahan kinerja, inisialisasi, dan kesalahan terminasi

2.7 Perangkat Lunak Pendukung

Perangkat lunak yang digunakan untuk mendukung perancangan sistem informasi kependudukan adalah bahasa pemrograman Borland Delphi 7 dan basis data yang digunakan adalah *SQL Server 2000* serta Rational rose untuk pemodelan sistem. Berikut penjelasannya ;

1. Borland Delphi 7.0

Borland Delphi 7.0 adalah salah satu bahasa pemrograman berbasis windows yang popular saat ini. Borland Delphi 7.0 memiliki fasilitas Object Oriented Programming (OOP) yang menyediakan object-object yang sangat kuat, powerful dan mudah digunakan dalam mendesain suatu aplikasi program.

Delphi sebenarnya merupakan kelanjutan dari Turbo Pascal, yang merupakan produk Borland yang berkedudukan di Scotsts Valley, California, ysng setelah melebur dengan Visigenic berganti nama menjadi Inprise Corporation pada pertengahan 1998. Delphi secara resmi diumumkan pada tanggal 14 Februari 1995 dan rilisnya dilakukan pada tanggal 28 Februari tahun yang sama.

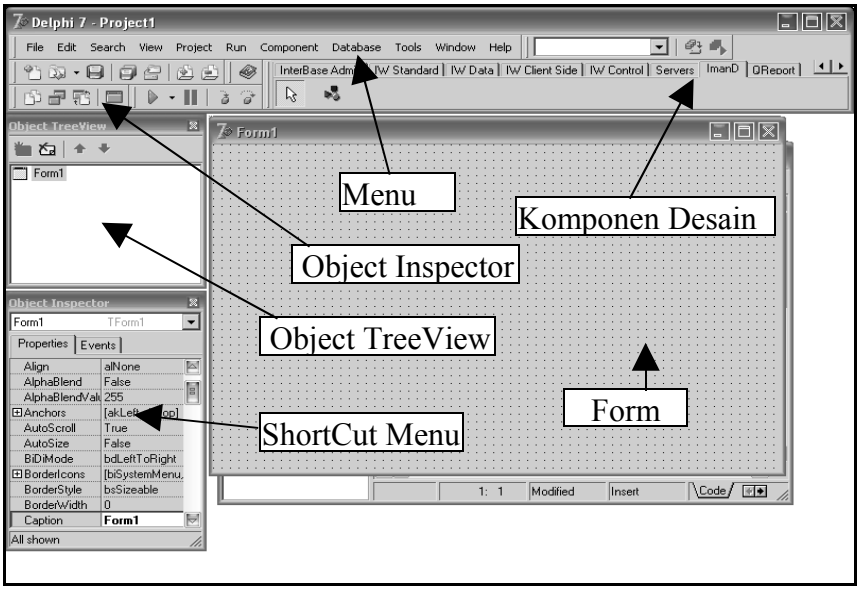
Delphi tidak saja dapat mengakses database seperti Paradox, xBase, Ms. Access tetapi dapat juga mengakses database lainnya seperti Oracle, Sybase,

Interbase, DB2, MS. SQL Server dan lain-lain dengan menggunakan ODBC (Open Database Connectivity). Begitu aplikasi selesai, untuk memindahkan dari satu jenis aplikasi database ke yang lainnya tidak perlu mengubah aplikasi secara keseluruhan.

Delphi dinyatakan oleh banyak pengamat sebagai *No Limit* (tanpa batas). Suatu pernyataan yang tendensius tetapi banyak yang mengakuinya. Pernyataan ini didukung dengan kemampuan dan keunggulannya seperti:

1. Native-code compiler yang cepat.
2. Debugger yang terpadu dengan text editor.
3. High-level database akses.
4. Teknologi two-ways tools yang canggih.
5. Kemampuan untuk mengakses langsung ke Windows API.
6. Build in assembler, dapat dibuat dalam source code yang sama.
7. Dapat membangun aplikasi untuk internet (Internet Browser, Email, TCP/IP Socket dan sebagainya) serta aplikasi multi-tier, dan aplikasi Web Server (Win-CGI, ISAPI, NISAPI).
8. Dapat menggunakan COM (Common Object Model).
9. Dapat membuat sendiri VCL (Visual Component Library), OCX (ActivaX) dan COBRA (Common Object Request Broker Architecture).
10. Pemrograman Object Oriented Murni.
11. Dapat berjalan di berbagai sistem operasi baik Linux (Borland Kylix), maupun Windows (Borland Delphi)

Di bawah ini merupakan tampilan Borland Delphi 7.



Gambar 2.5 Tampilan Borland Delphi 7

Borland Delphi 7.0 adalah suatu perangkat yang handal untuk membuat aplikasi dan bahasa pemogramannya merupakan bahasa pemograman berbasis *Windows* dan merupakan pengembangan dari bahasa Pascal. Borland Delphi 7.0 ini dapat membuat sebuah program dari yang sederhana sampai program yang berbasis *Client Server* atau jaringan.

Delphi menyediakan arsitektur yang memungkinkan penggunaan ulang komponen-komponen program, hal ini menyebabkan pembuatan aplikasi menjadi lebih cepat, komponen-komponen tersebut terdapat pada IDE (*Integrated Development Environment*). IDE (*Integrated Development Environment*) adalah bagian dari Delphi yang digunakan untuk menciptakan aplikasi, dengan IDE inilah para programmer secara visual merancang tampilan untuk para *user* (antarmuka pengguna) dan menulis listing program (kode).

Delphi 7.0 terdiri dari beberapa bagian yang masing-masing memiliki komponen-komponen dan fungsi tertentu. Bagian-bagian tersebut diantaranya adalah MenuBar, ToolBar/SpeedBar, Component Palette, Jendela Form, jendela Unit, Object Tree View, dan Object Inspector. Untuk keterangan lebih lanjut, dibawah ini terdapat penjelasan tentang komponen-komponen serta fungsi-fungsi tersebut.

1. *MenuBar*

Berfungsi untuk memilih tugas-tugas tertentu seperti memulai, memilih, atau mengakhiri suatu aplikasi.



Gambar 2.6 MenuBar

2. *ToolBar/SpeedBar*

Tidak jauh berbeda dengan fungsi MenuBar, tetapi ToolBar/SpeedBar lebih praktis dan cepat dalam penggunaannya.



Gambar 2.7 ToolBar/ SpeedBar

3. *Component Palette*

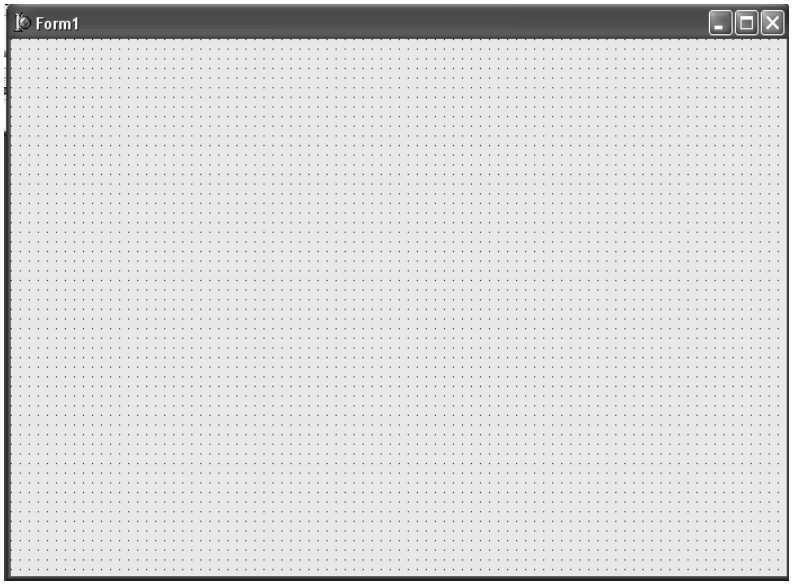
Component Pallete berisi kumpulan komponen yang akan ditempelkan atau diletakkan dalam form dan digunakan untuk mendesain form sehingga membentuk antarmuka pengguna.



Gambar 2.8 Component Palette

4. Jendela Form

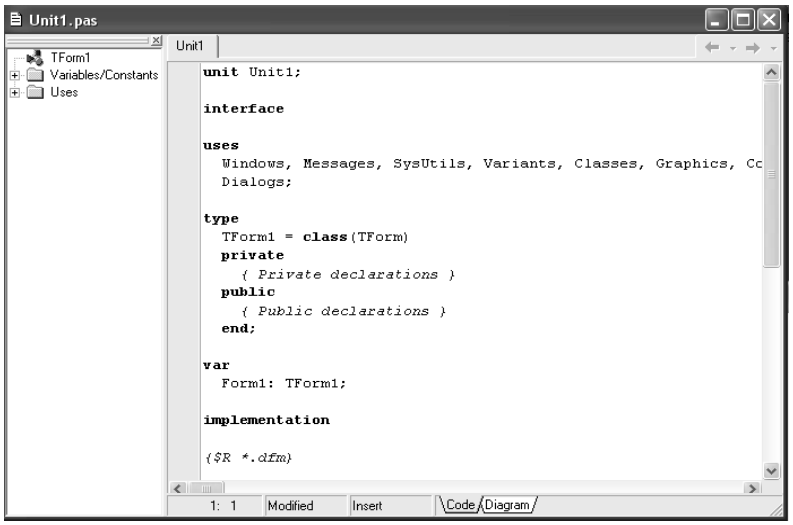
Jendela ini digunakan untuk merancang dan mendesain antarmuka pengguna dan di sini pula kontrol-kontrol komponen dari palet ditempatkan.



Gambar 2.9 Jendela Form

5. Jendela Unit

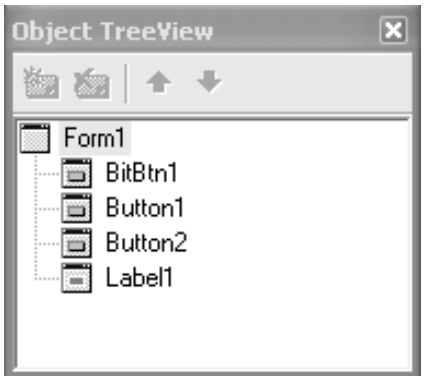
Jendela ini digunakan secara umum untuk menuliskan listing program dalam suatu aplikasi dan didalamnya terdapat sebuah struktur unit.



Gambar 2.10 Jendela Unit

6. Jendela Object Tree View

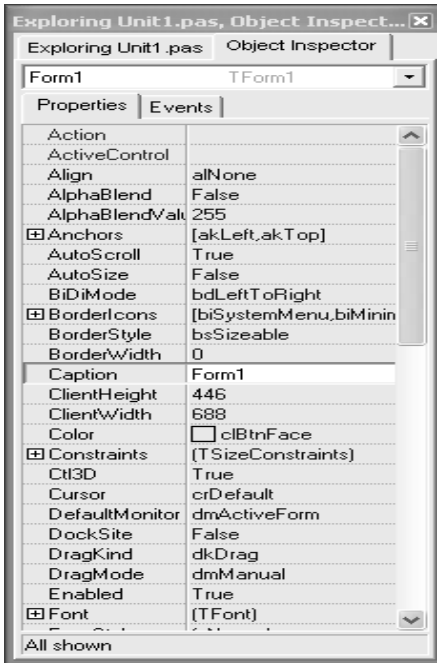
Jendela ini digunakan untuk melihat kontrol-kontrol apa saja yang berada didalam form tersebut secara hirarki seperti Windows Explorer.



Gambar 2.11 Jendela Object Tree View

7. Jendela Object Inpector

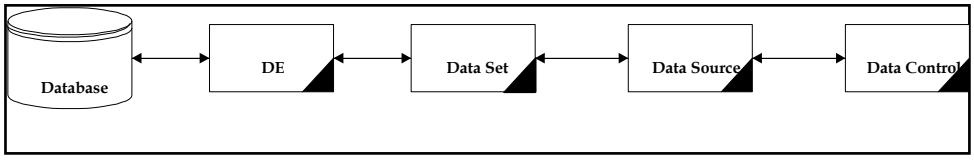
Jendela ini digunakan untuk memanipulasi kontrol-kontrol yang sudah ada dalam form. Jendela Object Inpector ini memiliki dua halaman, masing-masing disebut property (*Properties*) dan kejadian (*Events*).



Gambar 2.12 Jendela Object Inpector

8. Database Pada Borlad Delphi 7.0

Delphi merupakan salah satu software pembantu untuk membuat suatu aplikasi yang mendukung dalam melakukan interface dengan database. Berikut ini adalah gambaran arsitektur aplikasi database dengan menggunakan Borlad Delphi 7.0



Gambar 2.13 Arsitektur Database

2. SQL Server 2000

SQL server adalah sebuah database relational (RDBMS) yang dirancang untuk mendukung aplikasi dengan arsitektur client-server, dimana database terdapat pada komputer pusat yang disebut server, dan informasi digunakan bersama-sama oleh beberapa user yang menjalankan aplikasi didalam komputer lokalnya yang disebut dengan client.

3. Rational Rose

Rational Rose merupakan salah satu software yang paling banyak digunakan untuk melakukan design software melalui pendekatan UML (Unified Modelling Language). Rational Rose merupakan software yang menyediakan banyak fungsi - fungsi seperti : design proses, generate code, reverse engineering, serta banyak fungsi-fungsi yang lain.