BAB II

TINJAUAN PUSTAKA

2.1. Pengertian Aplikasi

Aplikasi merupakan penerapan, pemgimplementasian suatu hal, data, permasalahan, pekerjaan kedalam suatu sarana atau media yang dapat digunakan untuk menerapkan atau mengimplementasikan hal atau permasalahan tersebut sehingga berubah menjadi suatu bentuk yang baru tanpa menghilangkan nilai-nilai dasar dari hal, data, permasalahan atau pekerjaan.

Jadi dalam hal ini hanya bentuk dari tampilan data yang berubah, sedangkan isi yang termuat dalam data tersebut tidak mengalami perubahan.

2.2. Alasan Pemilihan Program

Saat ini berbagai program aplikasi dengan segala macam bahasa pemrograman yang digunakan sudah banyak beredar dipasaran. Dimana semua itu diciptakan untuk membantu meringankan pekerjaan manusia seperti dibidang bisnis, komunikasi, militer, pendidikan dan lain sebagainya. Perkembangan teknologi komputer saat ini sungguh luar biasa. Hal ini terbukti dengan kemampuan komputer yang mempunyai tingkat kecepatan dan kecermatan yang luar biasa, sehingga secara normal hampir mustahil dapat dilakukan oleh manusia. Dari sekian banyak

program-program aplikasi dan bahasa pemrograman yang beredar yaitu Borland Delphi.

Alasan pemilihan bahasa pemrograman Borland Delphi karena didalam Delphi banyak sekali terdapat kemudahan-kemudahan yang tidak dimiliki oleh bahasa pemrograman yang lain sepeti berikut ini :

1. Berpindah-pindah dari platform satu ke platform lainnya.
2. Otomatisasi penerjemahan jenis data antara jenis data C++ yang digunakan Delphi dengan jenis data platform tabel yang sedang aktif.
3. Kemampuan joint antar tabel dari platform yang berlainan.
4. Pengaksesan data secara dua arah pada tabel dengan berbagai platform.

Selain kemudahan diatas, Delphi menyediakan Fasilitas yang luas dari fungsi untuk membuat Form hingga fungsi untuk menggunakan beberapa Format File Basis Data yang popular dan Kelebihan - kelebihannya meliputi :

1. 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
2. Template Aplikasi dan Template Form

   Dalam Delphi telah didefinisikan Template Aplikasi dan Template Form yang dapat dipakai untuk membuat semua Aplikasi dengan cepat

3. Lingkungan Pengembangan Delphi

   Beberapa contoh Fasilitas di dalam Delphi yang dapat diatur sesuai kebutuhan adalah Palet komponen (Component Pallete), Editor Program dan Template Form

4. Program Yang Terkompilasi

   Kebanyakan lingkungan pengembangan Visual pada Window menyatakan dapat mengkompilasi Program. Namun sebenarnya mereka hanya dapat mengkompilasi sebagian program dan kemudian menggabungkan interpreter dan Pcode dalam sebuah File. Dengan cara ini didapatkan eksekusi yang lambat. Di dalam Delphi program yang dihasilkan benar-benar program yang terkompilasi tanpa interpreter dan pcode sehingga dapat berjalan lebih cepat. Program Delphi yang kecil dapat diserahkan dalam bentuk sebuah File EXE tanpa harus menyertakan File dan lain sebagainya.

5. Kemampuan Mengakses Data Dalam Bermacam Format

   Dalam Delphi terdapat Borland Database Engine (BDE) yang digunakan untuk mengakses Format File data yang ada. BDE telah melalui beberapa tahap perkembangan, sebelumnya BDE dikenal sebagai ODAPI, kemudian IDAPI, dan sekarang menjadi standar untuk akses semua jenis data yang ada saat ini. BDE juga dapat mengakses Database Client atau Server.

2.3. Sejarah Singkat Borland Delphi

Ide munculnya Delphi berasal dari bahasa pemrograman yang cukup terkenal yaitu Pascal. Bahasa Pascal sendiri telah diciptakan pada tahun 1971 oleh seorang ilmuwan dari Swiss yaitu Niclous Wirth. Nama Pascal diambil dari nama ahli matematika yang berasal dari prancis yaitu Bloise Pascal (1623-1663).

Sejak 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 windows 3.x yaitu Turbo Pascal for Windows. Pada tahun 1992, Borland International Incorporation menggabungkan Turbo Pascal for Dos dan Turbo Pascal for Windows menjadi satu paket bahasa pemrograman yang dikenal dengan Turbo Pascal versi 7.0.

Karena pemrograman Wndows dengan Turbo Pascal dirasa masih sulit, maka sejak tahun 1993, Borland International Incorporation mengembangkan bahasa Pascal yang bersifat visual. Hasil dari pengembangan ini adalah diri isnya 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, yaitu tahun 1997 Borland Interriational 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 dan yang terbaru yaitu Borland Delphi 5.0 telah dirilis pada tahun 1999.

2.3.1. Prosedur Eksekusi

Borland Delphi merupakan program compiler yang paling baik saat ini. Didalam delphi tersedia banyak kemudahan-kemudahan termasuk dalam hal mengeksekusi suatu program. Program delphi berbeda dengan program compiler lain yang berbasis Dos, dimana untuk mengeksekusi suatu program, terlebih dahulu harus dikompilasi dengan di linking oleh orang yang membuat program tersebut, sedangkan pada Borland Delphi untuk mengeksekusi suatu program, prosedur kompilasi dan linkingnya tidak perlu dilakukan oleh orang yang membuat program tersebut, karena akan dikompilasi dan di linking sendiri secara otomatis oleh Borland Delphi. Dalam pemrograman Borland Delphi, ekskusi program dapat dilakukan dengan 3 (tiga) cara secara langsung yaitu :

1. Pilih menu run
2. Tekan F9
3. Klik speed bar yang bertanda ▸

2.4. Manajemen Database

Manajemen Database merupakan hal yang terpenting dalam sebuah Aplikasi Database yang memempunyai data yang cukup besar. Dengan jumlah data yang besar memerlukan pengatur data dengan baik supaya dalam penyajian datanya dapat disajikan dengan cepat, tepat dan akurat

sesuai dengan data yang dibutuhkan oleh pihak yang memerlukan data tersebut, maka untuk hal tersebut diperlukan sebuah Manajemen Database.

Dalam Perancangan Manajemen Database ada tahap perancangannya yang harus ditempuh supaya data yang dihasilkannya sesuai dengan harapan dan disini akan dikemukakan beberapa cara yang harus dilakukan dalam pendisainan Database.

2.4.1. Database

Database merupakan suatu hal yang sangat penting dalam pembuatan program, karena nantinya akan memudahkan pengaksesan file di dalam program itu sendiri. Kesulitan utama dalam desain database adalah bagaimana agar database dapat memuaskan keperluan saat ini dan masa mendatang. selain perancangan model konseptual dilakukan juga perancangan model phisik. Pada perancangan konseptual akan menunjukkan entity dan relasinya berdasarkan proses yang diinginkan oleh organisasi.

Database adalah kumpulan file-file yang saling berelasi, relasi tersebut biasanya ditunjukkan dengan kunci dari tiap file yang ada. Satu database menunjukkan satu kumpulan data yang dipakai dalam satu lingkup Perusahaan, Instansi.

Penerapan database dalam sistem informasi disebut dengan sistem database (*database system*). Dengan sistem dasar database ini tiap-tiap

orang atau bagian dapat memandang database dari beberapa sudut pandang yang berbeda.

Dalam database biasanya terdiri dari entity, atribut, data value, record dan file. Entity adalah orang, tempat, kejadian atau konsep yang informasinya direkam. Atribut adalah setiap entity mempunyai atribut atau sebutan untuk mewakili suatu entity. Data value adalah data aktual atau informasi yang disimpan pada tiap data elemen yang saling berkaitan menginformasikan tentang suatu entity secara lengkap. File adalah kumpulan record-record sejenis yang mempunyai panjang elemen yang sama, atribut yang sama namun berbeda data valuenya.

2.4.2. Jenis-jenis Database

Lebih dari setengah kegiatan manusia dalam menggunakan komputer adalah sebagai sarana pengolahan data. Untuk itu mutlak diperlukan perangkat lunak database.

Bila kita kaji lebih mendasar tentang batasan suatu database, maka dapat disebutkan bahwa segala bentuk koleksi informasi adalah suatu database. Mulai dari kelompok data pegawai sampai sekelompok file, merupakan suatu database. Kelompok informasi tersebut kemudian diorganisir terhadap jenisnya dan diwujudkan dalam bentuk tabel sehingga terbentuklah bermacam-macam tabel atau sejenisnya.

Namun demikian, meskipun tabel-tabel telah dipisahkan terhadap jenis yang berlainan, suatu tabel biasanya memiliki kaitan informasi

dengan tabel lainnya. Untuk mewujudkan keterkaitan antar tabel itulah maka muncul konsep Relational database.

Dalam perancangan sistem database, ada dua pertimbangan utama yang tak boleh diabaikan yaitu keamanan dan otoritas. Keamanan menyangkut segala potensi database dalam mendata data dari pihak-pihak yang tidak diinginkan. Sedangkan otoritas terkait dengan kewenangan para pemakai program. Harus ada aturan yang jelas agar tidak terjadi pertentangan maksud dalam pemakaian program.

Secara fisik, suatu database terdiri dari dua bagian, bagian utama adalah berupa mesin (Engine) yang mampu menghasilkan dan mengedit struktur tabel dan membentuk sistem database. Bagian utama ini sering disebut Back end. Sedangkan bagian kedua adalah yang lazim disebut Front End. Pada Front end yang merupakan sarana komunikasi dengan pemakai ini, kita bisa menyusun form maupun laporan sesuai dengan kebutuhan selera kita.

Sebagai model sistem, suatu database bisa dipandang dari berbagai aspek. Diantaranya berdasarkan karakter dan konfigurasi kemampuan yang dimiliki. Dari sana lalu diperoleh tiga model dasar database yaitu Stand Alone Database, File-Share Database dan Client-Server Database.

2.4.2.1. Stand Alone Database

Sebuah database bisa saja tersusun atas sejumlah kecil tabel. Bahkan hanya tersusun atas sebuah tabel. Stand alone Database merupakan jenis yang paling sederhana, banyak hal yang bisa

dikesampingkan pada database jenis ini, misalnya tentang pengaturan otoritas ataupun concurent. Sebab memang tidak diperlukan dan tidak akan pernah terjadi masalah dengan hal-hal tersebut.

Pada database jenis ini, file database seperti tabel, report, dan sebagainya tersimpan dalam sistem file lokasl dan engine databasenya juga tersimpan jadi satu pada sebuah komputer.

Database stand alone ini sangat cocok untuk jenis aplikasi yang harus disebarkan kebanyak komputer dan pemakainya secara mandiri.

2.4.2.2. File Share Database

Database ini diterapkan pada suatu jaringan, sehingga meskipun database tersimpan dalam salah satu komputer , komputer lain bisa juga menggunakan database tersebut.

Dalam hal ini database berupa sebuah file dan engine-nya juga bersifat lokal. Jadi sebenarnya database ini bersifat seperti stand alone, hanya saja komputernya terpasang pada suatu jaringan. Komputer lain yang ingin ikut menggunakan file database tersebut hanya bersifat " menumpang " saja.

Pada database ini tidak akan terjadi masalah hardware, seperti kompatibilitas berbagai NIC (Network Interface card). Dimana hal ini bisa merupakan masalah bagi suatu aplikasi Client Server. Namun tentu saja ada kekurangannya. Kalau suatu proses perhitungan yang kompleks, apalagi sebagaian besar pemakai melakukannya, maka akan terjadi masalah yang cukup pelik. Atau bahkan databasenya mogok.

2.4.2.3. Client-Server Database

Client-Server Database merupakan jenis terbaik model sistem Database. Jenis ini sangat cocok untuk keperluan bisnis bagi suatu perusahaan, yang bisa diterapkan pada suatu kantor atau bahkan beberapa lokasi sekaligus. Misalnya beberapa kantor dan pabrik.

Pada jenis ini pula bisa dilaksanakan suatu proses perhitungan yang kompleks dan simultan. Dan sebagai konsekuensi yang harus ditanggung tidaklah ringan. Komputer server haruslah memenuhi spesifikasi kelas berat , Kompatibilitas NIC harus mutlak diperhatikan dan disusun dalam sistematika yang jelas dan terstruktur. Berbagai protokol serta urusan administrasi jaringan harus benar-benar terjamin dengan baik. Dan untuk pembangunan itu semua tentu saja memerlukan dana, waktu, Keterampilan maupun sumber daya yang baik pula.

2.5. Teknik Pengkodean

Teknik pengkodean sangatlah penting dalam aplikasi database ini, kerena dengan teknik pengkodean ini akan memudahkan baik dalam pemasukkan data maupun dalam hal proses pencarian data. Dengan penggunaan pengkodean ini akan menguntungkan kepada pengguna sistem tersebut dalam hal ini yaitu petugas dan pengguna sistem secara umum.

Adapun yang dimaksud dengan kode adalah penyajian dalam pengklasifikasikan data sehingga mudah dalam proses memasukkan

kedalam sistem tersebut. Pengguaan kode biasanya dipakai untuk mengindentifikasikan data orang, barang, rekening dan lain-lain.

2.5.1. Kodefikasi

Kode merupakan simbol dari kelas yang dapat digunakan untuk mencatat, mengkomunikasikan dan memproses data. Kode dapat berfungsi ganda yaitu dari memperpendek, menghindarkan indentifikasi rangkap dari sebuah item data, dan dengan kode data item, record atau file dapat dikenal dengan jelas. Selain kode memberikan pengertian yang khusus terhadap struktur data yang digunakan yang berguna dalam pencarian, manipulasi, dan pengolahan data.

Penggunaan kode pada item data dapat membuat pengolah data lebih efesien dan lebih cepat, kerena data yang diolah sudah berbentuk baku. Kode yang ringkas menghemat waktu dalam pelaporan, pencatatan, pengamatan dan penghematan ruang yang dipakai. Sebagai standar simbol, kode tidak memungkinkan terjadinya kesalahan pengkodean dan selain itu setiap kalakter dalam struktur kode mempunyai arti tertentu.

Kode data yang berupa huruf, angka, kata atau simbol lainnya (karakter) maupun kombinasinya. Kombinasi angka (numeric) dengan huruf (alfabet) disebut dengan alfanumerik. Gabungan ini dapat dibuat menjadi berbagai struktur. Dalam merancang kodefikasi untuk item data, perlu kiranya diperhatikan hal-hal sebagai berikut :

- Skema kode harus logis dan mudah dimengerti, sesuai dengan kebutuhan pemakai dan pengolah.

- Rancangan kode harus freksibel terhadap perubahan.
- Struktur kode harus frekibel terhadap perubahan
- Dimengerti oleh pemakai dalam Organisasi

Disamping itu kode dapat disusun dalam berbagai cara, kemudian dipilih secara seksama. Tipe-tipe kode yang akan digunakan masing-masing memiliki kekurangan dan kelebihan sesuai dengan yang sudah dijelaskan diatas, sehingga dapat mempertimbangkan penggunaan kode tersebut sesuai dengan kebutuhannya.

2.5.2. Simbol Kode

Simbol yang sering digunakan yang dalam hampir semua proses yang keterkaitannya dengan data. Untuk pengkodean ini biasa digunakan dengan bilangan maupun abjad atau modifikasi dari penggabungan kedua-duanya.

2.5.3. Desain kode

Banyak kemungkinan pendisainan kode dari digit, huruf atau karakter dalam penyajian kode. Dalam pendisainan ada ha-hal yang harus diperhatikan dalam hal mendisain kode tersebut adalah sebagai berikut:

- Kode sebaiknya serasi dengan kebutuhan dari User maupun metode pemrosesan yang dipakai.
- Tiap dari kode harus mewakili satu item saja.
- Kode sebaiknya didisain se-flexibel mungkin, tetapi jangan terlalu berlebihan.

- Panjang kode sebaiknya selalu sama.
- Kode yang terdiri dari 4 alfhabet atau 5 angka berturut-turut sebaiknya dipisahkan dengan kalakter (-) atau (/).

2.5.4. Tipe-tipe Struktur Kode

Bentuk kode dalam tipe struktur yang secara umum dipergunakan dalam pemrograman maupun dalam kegiatan sehari-hari akan dijelaskan dibawah ini sebagai berikut:

1. Kode Urut

Kode urut ini sering dipakai dalam penyimpanan arsip pada sistem manual yang sederhana.

Contoh : B001 Elang
C002 Singa
M003 Gajah
" "
R004 Kadal

Dari penggunaan kode urut diatas, adanya keuntungan dan kerugian yang ditimbulkan dari penggunaan kode tesebut. Keuntungan yang ditimbulkannya adalah sebagai berikut:

- Banyak sebagian besar menggunakannya karena sederhana dan mudah.
- Sudah tersortir dan unique.
- Sudah disajikan dalam penempatan record yang urut atau dukumen yang telah tersesun drngan rapih.

- Sangat mudah dalam proses Adminisrasi.
- Dan kerugiannya adalah sebagai berikut:
- Tidak dapat berbasis atau tersusun secara logical.
- Tidak flexible kerena rtidak dapat memuat perubahan-perubahan yang terjadi.

2.6. Perbandingan Antara Interpreter dan Compiler

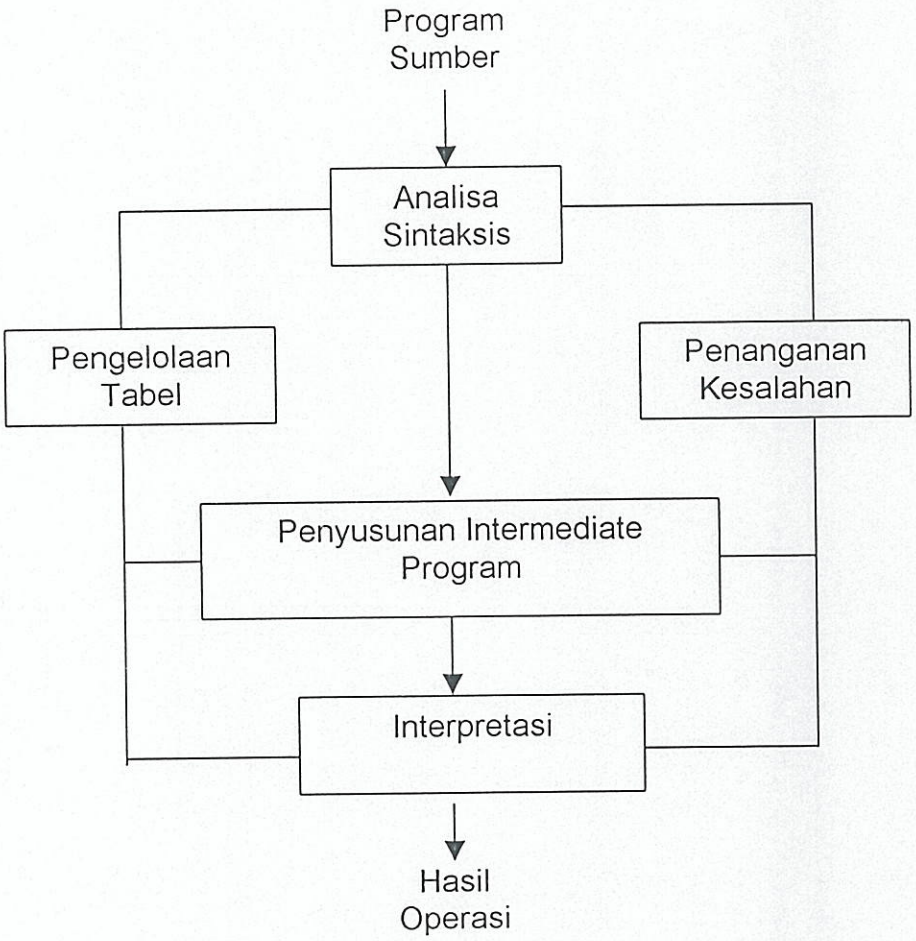
Perbedaan mendasar antara Interpreter dan Compiler, terletak pada hasil kerja masing-masing. Interpreter memberikan hasil dari pengerjaan instruksi-instruksi masukannya, sedangkan compiler menerima masukan berupa program komputer dalam suatu bahasa pemrograman.

2.6.1. Interpreter

Interpreter berasal dari kata *to interpret* yang berarti menterjemahkan atau mengartikan. Interpreter merupakan penerjemah bahasa pemrograman yang menterjemahkan instruksi demi instruksi pada saat eksekusi program.

Pada saat menterjemahkan, interpreter akan memeriksa sintaksis, leksikal dan semantik dari setiap instruksi program. Jika ditemukan kesalahan sintaksis (syntax error), maka interpreter akan menampilkan pesan kesalahan dan proses eksekusi program akan langsung terhenti pada saat itu juga. untuk lebih mengetahui cara kerja interpreter dapat dilihat pada gambar dibawah ini :

Ruang pengingat yang dibutuhkan interpreter tidak besar, karena program yang dieksekusi tidak berurutan (sequence). Namun ruang pengingat yang tersedia harus cukup besar, karena pada saat eksekusi itu seluruh source program, data dan interpreter secara bersama-sama berada didalam ruang pengingat. untuk lebih mengetahui cara kerja interpreter dapat dilihat pada gambar dibawah ini :



Gambar 2.1 Tahapan- Tahapan kerja Interpreter

Penerjemah jenis interpreter ini lebih cocok digunakan oleh para pemula, sedangkan untuk pemrogram yang telah beranjak menjadi profesional maka penerjemah ini kurang cocok, karena interpreter akan

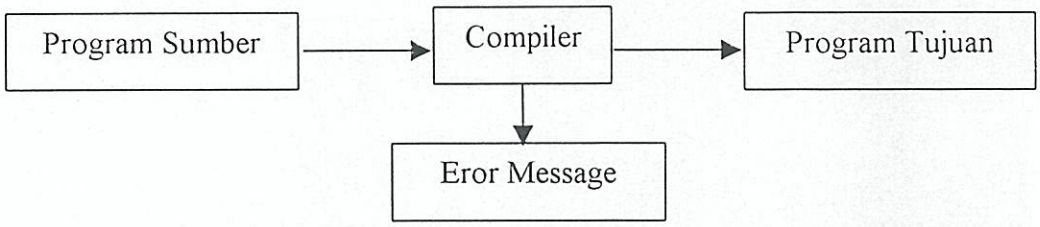
membiasakan pemrogram untuk menyusun program dengan logika yang kurang baik. Hal ini dapat dilihat dari proses penerjemahan yang dilakukan interpreter yang tidak mampu mendeteksi adanya kesalahan logika yang terjadi, tetapi hanya mampu mendeteksi kesalahan sintaksis saja. Selain itu, pemrogram akan senantiasa bergantung pada penggunaan paket bahasa pemrograman tersebut dalam melakukan eksekusi program.

2.6.2. Compiler

Compiler berasal dari kata *to compile* yang berarti menyusun, mengumpulkan atau menghimpun. Compiler merupakan penerjemah bahasa pemrograman yang menterjemahkan instruksi-instruksi dalam satu kesatuan modul, kedalam bahasa mesin (objek program). Kemudian objek program akan mengalami proses linking yang berfungsi untuk menggabungkan modul tersebut dengan modul lain yang berkaitan seperti data tentang karakteristik mesin, file-file pustaka atau objek program lainnya yang berkaitan dengan objek program tersebut sehingga dihasilkan suatu file executable program yang akan dieksekusi oleh komputer. Bila pada saat proses penerjemahan atau proses linking terjadi kesalahan, maka akan tampil pesan kesalahan dan proses eksekusi akan berhenti.

Proses kompilasi dilakukan setelah seluruh program selesai ditulis, karena compiler juga akan memeriksa sintaksis, leksikan, semantik, dan logika programnya. Program yang dikompilasi umumnya berjalan dengan kecepatan tinggi dan tidak tergantung lagi pada bahasa pemrograman

yang digunakan untuk menyusunnya. Untuk mengetahui proses kerja suatu kompilator dapat dilihat pada gambar dibawah ini :



Gambar 2.2 Proses kompilasi dari suatu kompilator

Kekurangan penerjemah jenis compiler ini adalah pada penggunaan ruang pengingat komputer yang besar pada saat proses kompilasi source program untuk menghasilkan objek program dan program executable.