UNIT PELAKSANA TEKNIS PERPUSTAKAAN UNIKOM

BAB IV

PERANCANGAN DAN IMPLEMENTASI SISTEM

4.1. Perancangan Sistem Basis Data

4.1.1. Pengertian

Perancangan sistem merupakan strategi untuk memecahkan masalah dan mengembangkan solusi terbaik bagi permasalahan tersebut. Termasuk bagaimana mengorganisasi sitem kedalam subsitem, serta alokasi subsistem kedalam komponen perangkat keras, lunak, serta prosedur – prosedur. Perancangan sistem merupakan arsitektur sistem.

4.1.2. Penggolongan Basis Data

Peneliti (*Programmer*) mengestimasi keterbatasan waktu dalam pembuatan basis data perusahaan secara komputerisasi dengan menggunakan bahasa pemrograman C++ Builder 6.0 menerapkan metode observasi langsung berdasarkan data yang ada dan membandingkan dengan dasar teori basis data secara *literature*. Sehingga peneliti dalam membuat basis data bertitik tolak dari hubungan tabel-tabel yang ada dalam perusahaan. Untuk membangun basis data perusahaan, peneliti menggunakan struktur tabel **Database Desktop**.

Tabel basis data yang disimpulkan oleh peneliti mencakup :

1. **Database master**, yaitu kumpulan data yang proses masukannya tidak memerlukan pengambilan data dari *database* lain. Sehingga *database*

ini menjadi *primary key* bagi *database* lain. Tabel data master pada PT. Mitratama Rasa Sejati meliputi :

a. *Database* Karyawan Tetap

b. *Database* Karyawan Kontrak

c. *Database* Bagian / Dept. Perusahaan

d. *Database* Jabatan

e. *Database* Jamsostek (Jaminan Sosial Tenaga Kerja)

2. **Database transaksi**, yaitu kumpulan data yang dapat menerima proses *input*, *edit*, dan hapus data setiap saat sesuai yang dibutuhkan, sehingga *database* ini memerlukan data dari *database* lain untuk menjalankan proses tersebut. *Database* transaksi ini meliputi :

   a. *Database* Masa Jabatan (seluruh karyawan dan staff perusahaan)

   b. *Database* Materi *Training* (per tahunnya)

   c. *Database Recruitment* Karyawan Perusahaan

   d. *Database* Evaluasi *Training* Karyawan

3. **Database History**, yaitu pengelompokkan data yang sudah diarsipkan karena data tersebut tidak aktif pada masa tertentu. Ini berarti data tersebut diambil dari *database* aktif dan setelah mengalami proses pe-*non*-aktifan menjadi data yang tidak diperlukan dalam waktu yang telah ditentukan. Pada PT. Mitratama Rasa Sejati, tiap 3 (tiga) tahun sekali ada refreshing data karyawan yang tidak aktif. Metode ini baik diterapkan untuk meng-urangi penggunaan *database memory*. Hanya

saja apabila tidak adanya database secara komputerisasi, maka akan terjadi duplikasi data. *Database history* ini oleh peneliti meliputi *database* karyawan tetap dan karyawan kontrak.

4.1.3. Distribusi struktur *database* perusahaan

Pada tahapan distribusi ini, merupakan pendistribusian setiap *database* yang ada, biasanya pendistribusian ini disertai dengan deskripsi *database* yang disebut sebagai kamus data (*Data Dictionary*). Kamus data dibuat untuk merancang *input*, laporan-laporan, dan *database*.

Struktur data yang terjadi dalam *database* program hasil kerja praktek adalah sebagai berikut :

a. *Database* Karyawan Tetap

No.	Nama Item	Tipe	Lebar	Keterangan
1	Kary_No	*Auto Increment*	-	Nomer arsip karyawan
2	No_Induk	*Alphabet*	10	No Induk Pegawai
3	Nama_Kary	*Alphabet*	25	Nama karyawan
4	Status	*Alphabet*	8	Status kawin atau tidak
5	J_K	*Alphabet*	4	Jenis kelamin (L/P)
6	Tgl_Lhr	*Date*	-	Tanggal lahir karyawan
7	Pend_A	*Alphabet*	13	Pendidikan akhir karyawan
8	Agama	*Alphabet*	8	Status agama karyawan
9	Jml_Anak	*Numeric*	5	Jumlah anak
10	Nama_Pas	*Alphabet*	25	Nama pasangan (Isteri)
11	J_Ki	*Alphabet*	4	Jenis kelamin pasangan
12	Tgl_Lhr_Pas	*Date*	-	Tanggal lahir pasangan
13	Pend_Pas	*Alphabet*	13	Pendidikan Akhir Pasangan
14	Anak_1	*Alphabet*	25	Nama anak pertama
15	Tgl_Lhr1	*Date*	-	Tanggal lahir anak pertama
16	Pend_1	*Alphabet*	13	Pendidikan anak pertama

17	Anak_2	*Alphabet*	25	Nama anak kedua
18	Tgl_Lhr2	*Date*	-	Tanggal lahir anak kedua
19	Pend_2	*Alphabet*	13	Pendidikan akhir anak kedua
20	Anak_3	*Alphabet*	25	Nama anak ketiga
21	Tgl_Lhr3	*Date*	-	Tanggal lahir anak ketiga
22	Pend_3	*Alphabet*	13	Pendidikan akhir anak ketiga
23	Alm_Asal	*Memo*	100	Alamat asal karyawan
24	Alm_Baru	*Memo*	100	Alamat baru karyawan
25	Telp	*Alphabet*	13	Nomer kontak karyawan
26	Tgl_Masuk	*Date*	-	Tanggal masuk karyawan
27	Ket	*Alphabet*	20	Rasa / Kencana
28	Status_Kerja	*Alphabet*	15	Aktif / tidak aktif

b. *Database* Karyawan Kontrak

No.	Nama Item	Tipe	Lebar	Keterangan
1	Nama	*Alphabet*	30	Nama karyawan kontrak
2	JK	*Alphabet*	4	Jenis kelamin
3	Tgl_Lhr	*Date*	-	Tanggal lahir
4	Alamat	*Memo*	100	Alamat karyawan kontrak
5	Awal_K	*Date*	-	Awal kontrak karyawan
6	Akhir_K	*Date*	-	Akhir kontrak karyawan
7	Bagian	*Alphabet*	25	Bagian / Dept. Perusahaan
8	Status_Kerja	*Alphabet*	8	Aktif / tidak aktif
9	Alasan	*Alphabet*	40	Keterangan masa kontrak

c. *Database* Bagian / Department Perusahaan

No.	Nama Item	Tipe	Lebar	Keterangan
1	Bagian	*Alphabet*	35	Nama *Department* perusahaan

d. *Database* Jabatan Perusahaan

No.	Nama Item	Tipe	Lebar	Keterangan
1	Jabatan	*Alphabet*	15	Posisi/jabatan di perusahaan

e. *Database* Jamsostek

No.	Nama Item	Tipe	Lebar	Keterangan
1	No_Induk	*Alphabet*	10	Nomer Induk (Acak)
2	No_KPJ	*Alphabet*	13	Berdasarkan lembaga

f. *Database* Masa Jabatan

No.	Nama Item	Tipe	Lebar	Keterangan
1	No_Induk	*Alphabet*	10	Nomer Induk (Berurut)
2	Akhir_Kerja	*Date*	-	Tanggal pemutusan kerja
3	Ket	*Memo*	45	Alasan pemutusan kerja

g. *Database* Materi *Training*

No.	Nama Item	Tipe	Lebar	Keterangan
1	No	*Numeric*	-	No urut masuknya materi
2	Materi	*Alphabet*	45	Berdasarkan Mgr
3	Trainner	*Alphabet*	5	Kode pelatih *training*
4	Tahun	*Numeric*	-	Tahun *training*

h. *Database* Mutasi Karyawan Perusahaan

No.	Nama Item	Tipe	Lebar	Keterangan
1	No_Induk	*Alphabet*	10	Nomer induk karyawan
2	Jabatan	*Alphabet*	15	Jabatan karyawan
3	Bagian	*Alphabet*	25	Bagian / Dept. karyawan
4	Status_Kerja	*Alphabet*	15	Aktif / tidak aktif

i. *Database Evaluasi Trining* Karyawan

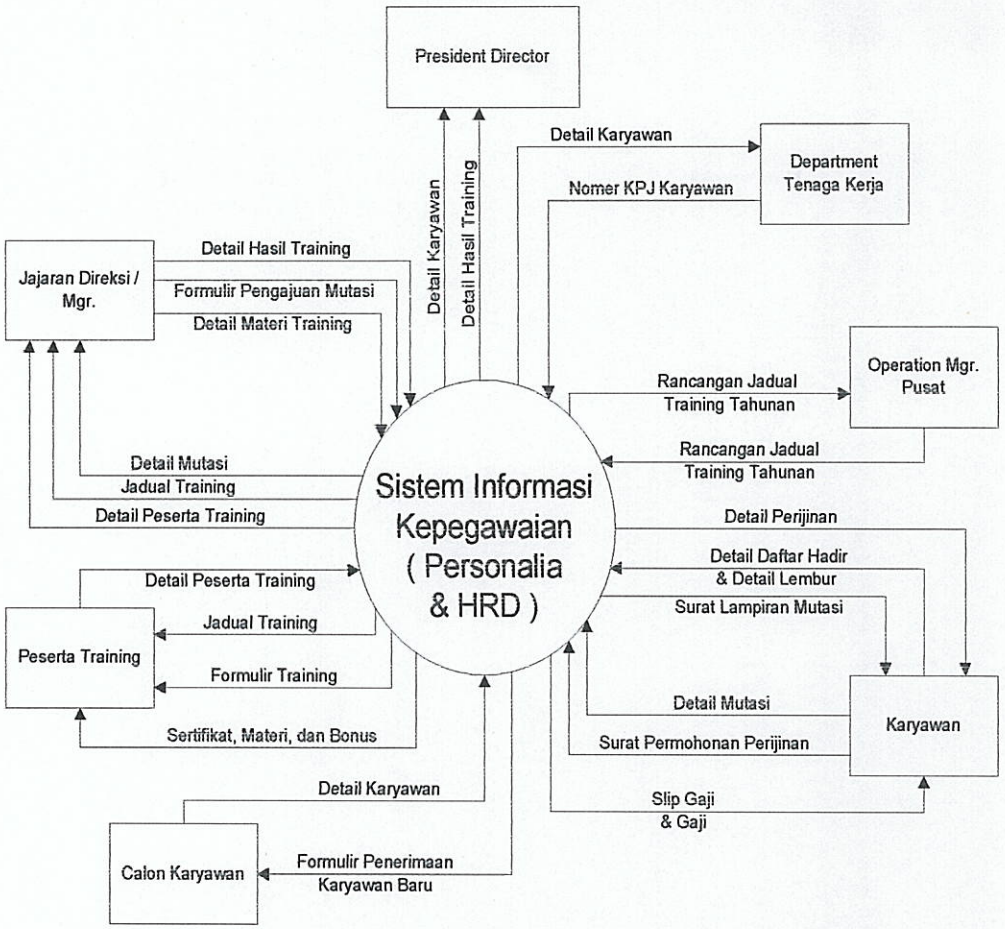
No.	Nama Item	Tipe	Lebar	Keterangan
1	No_Induk	*Alphabet*	10	Nomer Induk Karyawan
2	No	*Numeric*	-	Nomer materi yang ada
3	Materi	*Alphabet*	45	Materi yang telah ada
4	N	*Numeric*	-	Nilai *training* tiap materi

4.1.4. Distribusi proses dalam aktivitas perusahaan

Beberapa proses yang mendasari dari pembuatan program basis data di bagian HRD pada PT. Mitratama Rasa Sejati adalah sebagai berikut :

1. Proses pengangkatan (*Recruitment*) karyawan baru
2. Proses mutasi (promosi & demosi) karyawan
3. Proses perancangan jadual *training* tahunan
4. Proses pembuatan formulir *training* untuk calon peserta *training*
5. Proses pembuatan materi *training*
6. Proses dokumentasi
7. Proses perijinan karyawan (termasuk cuti)
8. Proses pendaftaran Jamsostek
9. Proses penghitungan lembur
10. Proses penghitungan gaji
11. Proses pembuatan laporan

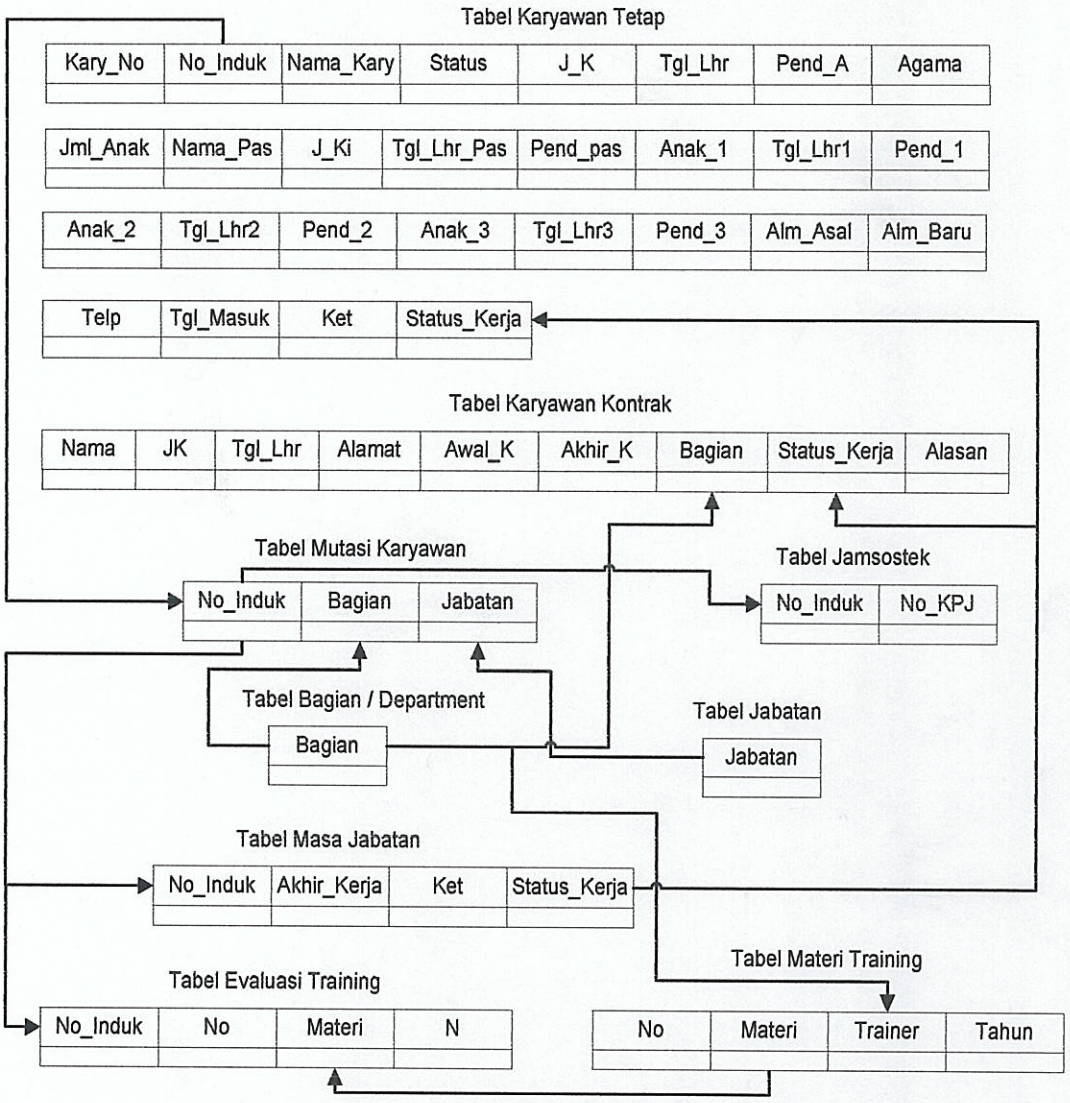
Dikarenakan begitu komplek dan rumit, serta masih belum terdokumentasikannya jalur atau aliran informasi yang terjadi, maka oleh penulis selama melakukan kerja praktek tidak mengadakan analisis data secara lengkap untuk kebutuhan Sistem Informasi Kepegawaian. Sehingga pada pembuatan laporan ini, penulis hanya mencantumkan ***Data Flow Diagram level* 0** (DFD *level* 0 atau **Diagram Konteks**). Sebagai penguat kebenaran data ini dicantumkan oleh penulis Surat Keterangan Ref. 101/ HRD/ - MRS/ IX/ 03.



Gambar 4.1. Diagram Konteks Sistem Informasi Bagian Personalia dan HRD Pada PT. Mitratama Rasa Sejati

4.1.5. Perancangan tabel implementasi *Entity Relationship Diagram*

Diagram hubungan entitas pada dasarnya adalah diagram yang memperlihatkan entitas – entitas yang terlibat dalam sistem serta hubungan – hubungan (relasi) antar entitas tersebut. Penekanan utama pada ERD adalah tabel – tabel yang merepresentasikan entitas – entitas serta tabel – tabel yang merepresentasikan relasi antar entitas itu sendiri. Pada gambar 4.2. akan memperlihatkan tabel – tabel yang dihasilkan dari pengadaan sistem basis data yang telah dirancang sebelumnya.



Gambar 4.2. Tabel - tabel Implementasi ERD Perusahaan

4.2. Perancangan Program C++ Builder V.6.0

Seluruh tampilan program dibuat dengan memenuhi beberapa syarat bahwa sebuah program telah layak digunakan, meliputi aspek :

1. *Simple and natural dialogue.*

Dengan menerapkan penggunaan kata dan tampilan yang wajar atau normal

2. Tata bahasa

   Penggunaan tata bahasa Indonesia sehingga lebih menyentuh kepada pengguna di tanah air.

3. Pengurangan beban penggunaan pemikiran pengguna

   Beberapa tampilan kontrol yang bersifat *user - friendly*, sehingga dapat mengurangi pemakaian pemikiran pengguna dalam menjalankan program.

4. Adanya keseragaman (*Consistency*)

   Dalam melakukan reformulasi, *programmer* mencoba mebuat keseragaman diseluruh tampilan *form* mulai dari awal hingga akhir program.

5. *System feedback*

   Programmer mencantumkan pula adanya indikator proses *refreshing* pada data *input* karyawan tetap, sehingga pengguna dapat mengetahui program masih atau telah selesai diproses.

6. *Clearly Mark Exit*

   Dengan mencantumkan *button exit* disetiap *form*-nya, maka pengguna dapat dengan langsung keluar dari tiap tampilan menu. Oleh *programmer* telah diantisipasi apabila terjadinya penutupan program dari jendela *form* yang tidak aktif, maka secara otomatis *form* yang masih aktif akan tertutup.

7. *Shortcut*

   Beberapa shortcut disertakan terutama pada perintah keluar dari program.

8. *Good Error Message*

Adanya beberapa *statement error message* untuk membantu pengguna dalam menjalankan program.

9. *Prevent Error*

   Untuk menghindari kesalahan yang mungkin terjadi, maka *programmer* menyisipkan prosedur pembantu dalam mengadakan *refreshing* data. Untuk lebih lanjut akan dijelaskan *sub-bab* 4.2.2.

10. Bantuan dan Dokumentasi

    Programmer sudah ada keinginan menyertakan alat bantu (*help*), tetapi karena keterbatasan waktu yang ada, maka pada program ini hanya terlihat sebagai menu kosong.

4.2.1. Perancangan *interface* (tampilan muka program)

Dengan menggunakan bahasa pemrograman berorientasi objek *visual* dan memiliki konsep *event driven programming*, maka dapat mempermudah *programmer* dalam membuat prosedur yang diinginkan. Dengan memilih Borland C++ Builder (dengan melihat keunggulan dari bahasa Visual Basic dan Borland Delphi) *programmer* juga mengarahkan kepada penerapan versi *Enterprise*, untuk pengembangan aplikasi *enterprise*, berisi fasilitas XML, arsitektur *three-tier* dan fasilitas-fasilitas lain yang dirasa dapat menunjang dengan baik fasilitas komputer berbasis jaringan.

Hasil yang diperoleh *programmer* dalam melakukan penelitian selama sebulan di perusahaan PT. Mitratama Rasa Sejati adalah sebuah tampilan program *input* basis data yang dapat digunakan secara

menyeluruh, walaupun ada beberapa tampilan program yang masih dalam proses pembentukan, setidaknya dapat digunakan pada sebuah PC. Melalui proses penginstalasian program tersebut ke dalam komputer, maka pengguna dapat dengan leluasa mengetahui dan melakukan *input* data seluruh karyawan perusahaan. Tetapi menjadi catatan penting, program ini hanya sebagai program dasar yang khusus diperuntukkan bagian personalia (dalam hal ini bagian HRD) yang dalam kenyataannya masih menggunakan komputer tunggal (*stand alone* PC) dalam melakukan proses masukkan karyawan. Seluruh tampilan muka program dapat dilihat di lembar lampiran *interface* program.

Dapat diperhatikan pada tampilan awal telah ditanamkannya sistem keamanan sederhana berupa pemasukkan *password*. Kemudian pengguna dihadapkan pada tampilan menu yang sekarang banyak diminati oleh pengguna. Dari tampilan menu ini akan mempermudah pengguna dalam mengakses data tertentu tanpa harus membuka tampilan program satu – persatu. Selain itu oleh *programmer* juga disertakan menu pembantu (*Help*), pada program awal masih belum ada data pembantu, sehingga *programmer* mencantumkan aplikasi peringatan kepada pengguna bahwa menu tersebut tidak terdapat dalam program awal.

Dapat dilihat dari perancangan jadual proyek pada Gambar 1.2., adanya tahapan *reformulasi* dan *packing* yang dilakukan oleh *programmer* dikarenakan terdapatnya kelemahan yang selama ini belum diketahui saat masa perkuliahan, dimana program yang tidak disertai fasilitas *refresh* data akan menjadi kurang aman. Hal ini dapat dibuktikan dengan

pengimplementasian program di komputer Pentium III keatas, saat pengguna memasukkan data dalam jumlah kecil, maka saat keluar dari program, *database* akan langsung menyimpan hasil olah data akhir secara tetap di dalam *hardisk*. Tetapi saat memasukkan data yang cukup banyak dan komplek, komputer membutuhkan waktu beberapa detik dalam melakukan penyimpanan, sehingga apabila pengguna keluar dari program dan langsung kembali memanggil program, maka database yang masih dalam proses penyimpanan tersebut akan hilang.

Untuk menanggulangi kecacatan program tersebut, maka oleh *programmer* setiap aplikasi tampilan *input* disisipkan prosedur *refresh* data, sehingga data secara otomatis disimpan hasil olah akhirnya setiap kali pengguna meng-*input* data. Jadi hal ini berarti *programmer* telah menerapkan metode replikasi data untuk mencegah terjadinya kesalahan – kesalahan secara mekanis. Dapat dilihat di lampiran *listing* program pada laporan ini.

4.2.2. Perancangan *input* data

Untuk merancang *input* data, *programmer* lebih menekankan pada tampilan yang setidaknya dapat menarik perhatian pengguna, misalnya dengan pengaturan posisi dan lebar tampilan tiap menu yang berbeda – beda. Hal ini disesuaikan dengan tingkat kebutuhan dan tingkat prioritas program. Serta membedakan tampilan untuk kedua perusahaan (PT. Mitratama Rasa Sejati dan PT. Mitratama Kencana Sejati).

Disisi lain, *programmer* membuat hubungan yang mudah berdasarkan Gambar 4.1. dimana untuk proses *input* jabatan dan bagian menjadi suatu tabel terpisah. Hal ini berguna agar saat ada restruktur perusahaan tidak sampai merubah data secara otomatis yang dapat menyulitkan pengguna. Lain halnya dengan perihal materi pelatihan, dimana data materi pelatihan dapat menjadi *primary key* disetiap karyawan dan dapat mempermudah penggolongan karyawan berdasarkan materi pelatihan.

Ada yang lain dari *database* yang biasanya kita jumpai, yaitu adanya metode *autoincrement*, sehingga setiap kali data dimasukkan maka ia akan langsung menempati *record* berikutnya. Sehingga nomer indeks data yang dimilikinya menjadi nomer arsip yang dipergunakan untuk pembuatan laporan.

4.2.3. Perancangan *output* program

Dari program yang telah dibuat, *programmer* memberikan beberapa *output* program diantaranya berupa laporan data karyawan tetap secara keseluruhan yang masih aktif, laporan data karyawan kontrak perusahaan, serta laporan data karyawan yang dibutuhkan untuk proses tunjangan karyawan (pajak dan status). Laporan ini kemudian dapat langsung dicetak atau disimpan untuk digunakan oleh bidang lain (lebih efektif apabila diterapkan pada komputer berbasis jaringan lokal), sehingga bagian HRD dapat melihat laporan karyawan dan materi pelatihan secara langsung dari *database* yang ada.

Untuk meningkatkan kualitas dan mutu laporan yang dihasilkan, dicantumkannya tanggal dan waktu pencetakkan laporan secara otomatis dan *real time*. Kemungkinan besar *programmer* dapat menyelesaikan seluruh program dan laporan yang dapat dihasilkan oleh program yang ada, apabila ditunjang oleh buku – buku yang lengkap, dikerjakan secara tim, waktu yang cukup, dan biaya untuk penelitian dan pengembangan.

4.3. Implementasi dan *Packing* Program

Dalam tahapan implementasi program yang telah dibuat, kita tidak lepas dari masalah perangkat penunjang, dalam hal ini komputerlah yang menjadi piranti utama dan penting. Komputer terdiri dari 2 (dua) bagian utama, yaitu *Hardware* (perangkat keras) dan *Software* (perangkat lunak). Oleh *programmer*, kedua piranti ini telah dirinci melalui file *readme*. Ada beberapa estimasi perangkat keras dan perangkat lunak yang harus dipenuhi sehingga program dapat sepenuhnya berjalan dengan baik, seperti :

1. Kebutuhan perangkat keras (*hardware requirement*), meliputi :
   a. Standard kebutuhan Pentium III keatas (jika dibutuhkan untuk proses *Editing*), minimal kebutuhan Pentium II 600 Mhz (hanya untuk aplikasi).
   b. Kapasitas ruang kosong *Hardisk* pada *drive* sistem minimal 1 (satu) Gigabyte. (1024 MB).
   c. VGA mendukung tampilan layar / *screen* monitor 1024 x 768, *default* monitor 800 x 600.
   d. *Color Quality Medium* (16 Bit).

e. Untuk bekerja di Sistem Operasi Win9x, maka minimal memori PC 64 MB (tidak di-*Share* dengan *Chipset Onboard*), rekomendasi 128 MB (untuk aplikasi yang lebih baik).

f. Jika bekerja di bawah Sistem Operasi 2000 keatas (XP atau Win 2003) maka sebaiknya memori perlu ditingkatkan lebih besar dari 128 MB.

g. CD-Rom minimal kecepatan baca 24x.

2. Kebutuhan perangkat lunak (*Software Requirement*), meliputi :

a. Sistem Operasi 9x keatas (Win98 Second Edition, Win Me, 2000, NT4, XP Proffesional, XP Home Edition).

b. Telah terinstalasi Software **Borlad C++ 6.0 Enterprise Suite** atau **Borland C++ 6.0 Proffesional**.

Sedangkan untuk mempermudah proses instalasi, maka *programmer* telah membuat kemasan program yang siap diinstalasikan kedalam sebuah sistem operasi komputer (apabila telah memenuhi seluruh kebutuhan program baik *Hardware* maupun *Software*). Proses pengemasan ini menggunakan *software* yang telah dipasarkan secara *shareware*. Dengan metode ini, *programmer* memberikan jaminan keamanan data dan mempermudah proses *update* program dengan yang baru (jika telah siap).