

BAB II

LANDASAN TEORI

2.1 Pengertian Ketidakhadiran

(Panggabean, 2002), Definisi ketidakhadiran (*absenteism*) adalah kegagalan untuk melapor pada waktu kerja. Dengan kata lain ketidakhadiran merupakan kegagalan seorang karyawan untuk hadir di tempat kerja pada hari kerja. Ketidakhadiran berbeda dengan terlambat (*lateness*) atau lamban (*tardiness*) yang menunjukkan kegagalan untuk datang tepat waktu.

Alasan pegawai tidak hadir ditempat kerja dan melakukan aktivitas beraneka ragam. **Michael J. Julius** mengemukakan bahwa alasan yang sering digunakan seorang pegawai jika tidak masuk kerja antara lain :

1. Sakit, merupakan alasan paling umum digunakan jumlahnya dapat mencapai 50% dari seluruh ketidakhadiran
2. Terjadinya kecelakaan kerja
3. Pengawasan kurang baik
4. Jam kerja yang tidak padat
5. Kurangnya minat dan tanggung jawab serta adanya perasaan tidak berguna
6. Setelah menerima upah, kondisi yang terlalu berat dan upah yang turun
7. Cuaca buruk, kurangnya transportasi, mencari pekerjaan lain, kunjungan saudara dari luar kota.
8. Sikap pikiran yang disebabkan oleh faktor lingkungan dan sosiologi

2.2 Pengertian Cuti

Cuti secara umum dapat diartikan tidak masuk kerja yang diizinkan dalam jangka waktu tertentu dan mematuhi semua peraturan yang berlaku di perusahaan tersebut.

2.3 Jenis Cuti

Jenis cuti merupakan macam-macam cuti yang dapat diajukan oleh pegawai Badan Kepegawaian Daerah Kota Semarang diantaranya:

a. Cuti Tahunan

1. Telah bekerja minimal 1 Tahun secara terus menerus.
2. Lamanya 12 hari kerja dan dapat dipecah-pecah minimal 3 hari.
3. Cuti atau sisa cuti tahunan yang tidak diambil dalam Tahun yang bersangkutan, dapat diambil dalam tahun berikutnya untuk paling lama 18 hari kerja termasuk cuti Tahunan dalam Tahun yang sedang berjalan.
4. Cuti Tahunan yang tidak diambil 2 (dua) Tahun berturut-turut atau lebih dapat diambil dalam Tahun berikutnya untuk paling lama 24 hari kerja termasuk cuti Tahunan dalam Tahun yang sedang berjalan.
5. Cuti Tahunan yang tidak diambil secara penuh dalam beberapa Tahun, dapat diambil dalam Tahun berikutnya untuk paling lama 24 (dua puluh empat) hari kerja termasuk cuti Tahunan yang sedang berjalan.
6. PNS yang menjadi guru pada sekolah dan dosen pada perguruan tinggi yang mendapat liburan menurut peraturan perundang-undangan yang berlaku, tidak berhak atas cuti Tahunan.

b. Cuti Besar

1. Telah bekerja sekurang-kurangnya 6 Tahun secara terus menerus berhak atas cuti besar selama 3 (tiga) bulan termasuk cuti Tahunan dalam Tahun yang bersangkutan.

2. Cuti besar dapat digunakan oleh PNS yang bersangkutan untuk memenuhi kewajiban agama, umpamanya menunaikan ibadah haji.
3. PNS yang mengambil cuti besar kurang dari 3 (tiga) bulan, maka sisa cuti besar yang menjadi haknya hapus.
4. Selama menjalankan cuti besar, PNS yang bersangkutan menerima penghasilan penuh. Yang dimaksud dengan penghasilan penuh, kecuali tunjangan jabatan struktural.

c. Cuti Sakit

1. PNS yang menderita sakit berhak atas cuti sakit.
2. Sakit selama 1 atau 2 hari memberitahukannya kepada atasannya baik secara tertulis maupun dengan pesan dengan perantaraan orang lain.
3. Sakit lebih dari 2 hari sd. 14 hari wajib mengajukan permintaan secara tertulis kepada pejabat yang berwenang dengan melampirkan surat keterangan dokter, baik dokter pemerintah atau swasta.
4. Sakit lebih dari 14 hari wajib mengajukan permintaan secara tertulis kepada pejabat yang berwenang dengan melampirkan surat keterangan dokter pemerintah atau dokter swasta yang ditunjuk oleh Menteri Kesehatan.
5. Cuti sakit tersebut diberikan untuk paling lama 1 Tahun, dan dapat ditambah untuk paling lama 6 bulan apabila dipandang perlu berdasarkan surat keterangan dokter pemerintah atau dokter swasta yang ditunjuk oleh Menteri Kesehatan.
6. Apabila sakit setelah 1 Tahun 6 bulan belum sembuh, harus diuji kembali kesehatannya oleh dokter yang ditunjuk oleh Menteri Kesehatan.

Apabila berdasarkan hasil pengujian kesehatan tersebut PNS yang bersangkutan :

- a. Belum sembuh dari penyakitnya tetapi ada harapan untuk dapat bekerja kembali sebagai PNS, maka ia diberhentikan dengan hormat

dari jabatannya karena sakit dengan mendapat uang tunggu menurut peraturan perundang-undangan yang berlaku.

b. Belum sembuh dari penyakitnya dan tidak ada harapan lagi untuk dapat bekerja kembali sebagai PNS, maka ia diberhentikan dengan hormat sebagai PNS, dengan mendapat hak-hak kepegawaian menurut peraturan perundang-undangan yang berlaku.

7. PNS wanita yang mengalami gugur kandungan berhak atas cuti sakit untuk paling lama 1 ½ (satu setengah) bulan.

8. PNS yang mengalami kecelakaan dan oleh karena menjalankan tugas kewajibannya yang mengakibatkan Pegawai tersebut perlu mendapat perawatan, berhak atas cuti sakit sampai ia sembuh dari penyakitnya.

9. Selama menjalankan cuti sakit PNS yang bersangkutan menerima penghasilan penuh.

d. Cuti Bersalin

1. Untuk persalinan pertama, kedua dan ketiga, PNS wanita berhak atas cuti bersalin. Persalinan pertama yang dimaksud adalah persalinan pertama sejak yang bersangkutan menjadi PNS.

2. Untuk persalinan yang keempat dan seterusnya kepada pns wanita diberikan cuti di luar tanggungan negara untuk persalinan. Dalam hal ini yang bersangkutan tidak diberhentikan dengan hormat dari jabatan organiknya.

3. Lamanya cuti bersalin adalah 1 (satu) bulan sebelum dan 2 (dua) bulan sesudah persalinan.

4. Apabila ada seorang PNS wanita yang mengambil cuti bersalin 2 (dua) minggu sebelum persalinan, maka haknya sesudah persalinan tetap 2 (dua) bulan.

5. PNS wanita yang akan bersalin untuk keempat kalinya dan seterusnya, apabila menjelang persalinan tersebut mempunyai hak atas cuti besar, dapat menggunakan cuti besar tersebut sebagai cuti persalinan.

6. Selama menjalankan cuti bersalin PNS wanita yang bersangkutan menerima penghasilan penuh.

Tata cara untuk mendapatkan cuti bersalin :

1. PNS wanita yang akan bersalin harus mengajukan permintaan cuti bersalin secara tertulis kepada pejabat yang berwenang memberikan cuti melalui hirarkhi.
2. Harap diperhatikan bahwa pejabat yang berwenang memberikan cuti harus memberikan cuti bersalin secara tertulis.
3. Untuk cuti persalinan yang pertama, kedua dan ketiga, dan
4. Untuk cuti di luar tanggungan negara untuk persalinan.
5. PNS wanita yang telah selesai menjalankan cuti di luar tanggungan negara untuk persalinan. Dengan surat keputusan pejabat yang berwenang memberikan cuti diaktifkan kembali dalam jabatan semula menurut contoh sebagai tersebut dalam lampiran.

e. Cuti karena alasan penting

1. PNS berhak atas cuti karena alasan penting untuk paling lama 2 (dua) bulan, tergantung alasan pentingnya, misalnya karena : Ibu, bapak, isteri atau suami, anak, adik, kakak, mertua atau menantu sakit keras atau meninggal dunia, melangsungkan perkawinan yang pertama, dsb.
2. Selama menjalankan cuti alasan penting, pegawai yang bersangkutan menerima penghasilan penuh.
3. Untuk mendapatkan cuti alasan penting PNS mengajukan permintaan tertulis kepada pejabat yang berwenang memberikan cuti dengan menyebut alasan-alasannya.
4. Dalam hal yang mendesak, PNS dapat mengajukan permintaan izin sementara kepada kepala pemerintahan setempat sambil menunggu keputusan pejabat yang berwenang memberikan cuti.

2.4 Definisi Data Mining.

Data mining adalah suatu istilah yang digunakan untuk menemukan pengetahuan yang tersembunyi di dalam database. Data mining merupakan proses semi otomatis yang menggunakan teknik statistik, matematika, kecerdasan buatan dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi pengetahuan dan berguna yang bermanfaat yang tersimpan di dalam database besar. (Turban et al, 2005). Selain definisi di atas beberapa definisi juga diberikan seperti, “*data mining* adalah serangkaian proses untuk menggali nilai tambah dari suatu kumpulan data berupa pengetahuan yang selama ini tidak diketahui secara manual.” (Pramudiono, 2006). “*Data mining* adalah analisis otomatis dari data yang berjumlah besar atau kompleks dengan tujuan untuk menemukan pola atau kecenderungan yang penting yang biasanya tidak disadari keberadaannya.” (Pramudiono, 2006).

“*Data mining* merupakan analisis dari peninjauan kumpulan data untuk menemukan hubungan yang tidak diduga dan meringkas data dengan cara yang berbeda dengan sebelumnya, yang dapat dipahami dan bermanfaat bagi pemilik data.” (Larose, 2005). “*Data mining* merupakan bidang dari beberapa keilmuan yang menyatukan teknik dari pembelajaran mesin, pengenalan pola, statistik, database, dan visualisasi untuk penanganan permasalahan pengambilan informasi dari database yang besar.” (Larose, 2005).

Kemajuan luar biasa yang terus berlanjut dalam bidang *data mining* didorong oleh beberapa faktor, antara lain : (Larose, 2005)

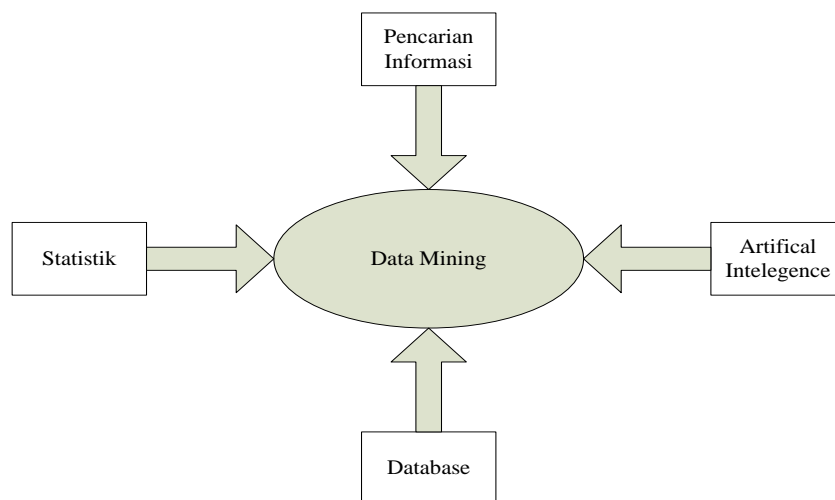
1. Pertumbuhan yang cepat dalam kumpulan data.
2. Penyimpanan data dalam *data warehouse*, sehingga seluruh perusahaan memiliki akses ke dalam database yang baik.
3. Adanya peningkatan akses data melalui navigasi web dan intranet.
4. Tekanan kompetisi bisnis untuk meningkatkan penguasaan pasar dalam globalisasi ekonomi.

5. Perkembangan teknologi perangkat lunak untuk *data mining* (ketersediaan teknologi).
6. Perkembangan yang hebat dalam kemampuan komputasi dan pengembangan kapasitas media penyimpanan.

Berdasarkan definisi-definisi yang telah disampaikan, hal penting yang terkait dengan *data mining* adalah :

1. *Data mining* merupakan suatu proses otomatis terhadap data yang sudah ada.
2. Data yang akan diproses berupa data yang sangat besar.
3. Tujuan *data mining* adalah mendapatkan hubungan atau pola yang mungkin memberikan indikasi yang bermanfaat.

Hubungan yang dicari dalam *data mining* dapat berupa hubungan antara dua atau lebih dalam satu dimensi. Misalnya dalam dimensi produk, dapat dilihat keterkaitan pembelian suatu produk dengan produk yang lain. Selain itu, hubungan juga dapat dilihat antara dua atau lebih atribut dan dua atau lebih objek. (Ponniah, 2001).



Gambar 2.1 Bidang Ilmu *Data Mining*

Data mining bukanlah suatu bidang yang sama sekali baru. Salah satu kesulitan untuk mendefinisikan *data mining* adalah kenyataan bahwa *data mining* mewarisi banyak aspek dan teknik dari bidang-bidang ilmu yang sudah mapan terlebih dahulu. Gambar 2.1 menunjukkan bahwa *data mining* memiliki akar yang panjang dari bidang ilmu seperti kecerdasan buatan (*artificial intelligent*), *machine learning*, statistik, database, dan juga *information retrieval* (Pramudiono, 2006).

Istilah *data mining* dan *Knowledge Discovery in Database* (KDD) sering kali digunakan secara bergantian untuk menjelaskan proses penggalian informasi tersembunyi dalam suatu basis data yang besar. Sebenarnya kedua istilah tersebut memiliki konsep yang berbeda, tetapi berkaitan satu sama lain. Dan salah satu tahapan dalam keseluruhan proses KDD adalah *data mining*. Proses KDD secara garis besar dapat dijelaskan sebagai berikut (Fayyad, 1996).

1. Data Selection

Pemilihan (seleksi) data dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalian informasi dalam KDD dimulai. Data hasil seleksi yang akan digunakan untuk proses *data mining*, disimpan dalam suatu berkas, terpisah dari basis data operasional.

2. Pre-processing/Cleaning

Sebelum proses *data mining* dapat dilaksanakan, perlu dilakukan proses *cleaning* pada data yang menjadi fokus KDD. Proses *cleaning* mencakup antara lain membuang duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data, seperti kesalahan cetak (tipografi). Juga dilakukan proses *enrichment*, yaitu proses “memperkaya” data yang sudah ada dengan data atau informasi lain yang relevan dan diperlukan untuk KDD, seperti data atau informasi eksternal.

3. Transformation

Coding adalah proses transformasi pada data yang telah dipilih, sehingga data tersebut sesuai untuk proses *data mining*. Proses *coding* dalam KDD merupakan proses kreatif dan sangat tergantung pada jenis atau pola informasi yang akan dicari dalam basis data.

4. Data mining

Data mining adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik, metode, atau algoritma dalam data mining sangat bervariasi. Pemilihan metode dan algoritma yang tepat sangat bergantung pada tujuan dan proses KDD secara keseluruhan.

5. Interpretation/Evaluation

Pola informasi yang dihasilkan dari proses data mining perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak yang berkepentingan. Tahap ini merupakan bagian dari proses KDD yang disebut *interpretation*. Tahap ini mencakup pemeriksaan apakah pola atau informasi yang ditemukan bertentangan dengan fakta atau hipotesis yang ada sebelumnya

2.5 Pengelompokan *Data Mining*

Data mining dibagi menjadi beberapa kelompok berdasarkan tugas yang dapat dilakukan, yaitu (Larose, 2005).

1. Deskripsi

Terkadang peneliti dan analisis secara sederhana ingin mencoba mencari cara untuk menggambarkan pola dan kecenderungan yang terdapat dalam data. Sebagai contoh, petugas pengumpulan suara mungkin tidak dapat menemukan keterangan atau fakta bahwa siapa yang tidak cukup profesional akan sedikit didukung dalam pemilihan

presiden. Deskripsi dari pola dan kecendrungan sering memberikan kemungkinan penjelasan untuk suatu pola atau kecendrungan.

2. Estimasi

Estimasi hampir sama dengan klasifikasi, kecuali variabel target estimasi lebih ke arah numerik dari pada ke arah kategori. Model dibangun menggunakan *record* lengkap yang menyediakan nilai dari variabel target sebagai nilai prediksi. Selanjutnya, pada peninjauan berikutnya estimasi nilai dari variabel target dibuat berdasarkan nilai variabel prediksi. Sebagai contoh, akan dilakukan estimasi tekanan darah sistolik pada pasien rumah sakit berdasarkan umur pasien, jenis kelamin, berat badan, dan level sodium darah. Hubungan antara tekanan darah sistolik dan nilai variabel prediksi dalam proses pembelajaran akan menghasilkan model estimasi. Model estimasi yang dihasilkan dapat digunakan untuk kasus baru lainnya.

3. Prediksi

Prediksi hampir sama dengan klasifikasi dan estimasi, kecuali bahwa dalam prediksi nilai dari hasil akan ada di masa mendatang.

Contoh prediksi dalam bisnis dan penelitian adalah:

- a. Prediksi harga beras dalam tiga bulan yang akan datang.
- b. Prediksi presentase kenaikan kecelakaan lalu lintas tahun depan jika batas bawah kecepatan dinaikan.

Beberapa metode dan teknik yang digunakan dalam klasifikasi dan estimasi dapat pula digunakan (untuk keadaan yang tepat) untuk prediksi.

4. Klasifikasi

Dalam klasifikasi, terdapat target variabel kategori. Sebagai contoh, penggolongan pendapatan dapat dipisahkan dalam tiga kategori, yaitu pendapatan tinggi, pendapatan sedang, dan pendapatan rendah.

Contoh lain klasifikasi dalam bisnis dan penelitian adalah:

- a. Menentukan apakah suatu transaksi kartu kredit merupakan transaksi yang curang atau bukan.
- b. Memperkirakan apakah suatu pengajuan hipotek oleh nasabah merupakan suatu kredit yang baik atau buruk.
- c. Mendiagnosa penyakit seorang pasien untuk mendapatkan termasuk kategori apa.

5. Pengklusteran

Pengklusteran merupakan pengelompokan *record*, pengamatan, atau memperhatikan dan membentuk kelas objek-objek yang memiliki kemiripan.

Kluster adalah kumpulan *record* yang memiliki kemiripan suatu dengan yang lainnya dan memiliki ketidakmiripan dengan *record* dalam kluster lain. Pengklusteran berbeda dengan klasifikasi yaitu tidak adanya variabel target dalam pengklusteran. Pengklusteran tidak mencoba untuk melakukan klasifikasi, mengestimasi, atau memprediksi nilai dari variabel target. Akan tetapi, algoritma pengklusteran mencoba untuk melakukan pembagian terhadap keseluruhan data menjadi kelompok-kelompok yang memiliki kemiripan (homogen), yang mana kemiripan dengan *record* dalam kelompok lain akan bernilai minimal.

Contoh pengklusteran dalam bisnis dan penelitian adalah:

- a. Mendapatkan kelompok-kelompok konsumen untuk target pemasaran dari suatu produk bagi perusahaan yang tidak memiliki dana pemasaran yang besar.
- b. Untuk tujuan audit akutansi, yaitu melakukan pemisahan terhadap perilaku finansial dalam baik dan mencurigakan.
- c. Melakukan pengklusteran terhadap ekspresi dari gen, dalam jumlah besar.

6. Asosiasi

Tugas asosiasi dalam *data mining* adalah menemukan atribut yang muncul dalam suatu waktu. Dalam dunia bisnis lebih umum disebut analisis keranjang belanja.

Contoh asosiasi dalam bisnis dan penelitian adalah:

- a. Meneliti jumlah pelanggan dari perusahaan telekomunikasi seluler yang diharapkan untuk memberikan respon positif terhadap penawaran *upgrade* layanan yang diberikan.
- b. Menemukan barang dalam supermarket yang dibeli secara bersamaan dan barang yang tidak pernah dibeli bersamaan.

2.6 Metode deskripsi

Metode deskripsi merupakan salah satu fungsi dalam data mining untuk menemukan atau menggali sekumpulan data yang banyak. Tiga cara dalam mendeskripsikan data yaitu dengan menggunakan deskripsi grafis, deskripsi lokasi dan deskripsi keberagaman

1. Deskripsi Grafis

Cara deskripsi grafis merupakan cara untuk mendeskripsikan data dalam bentuk gambar yang sesungguhnya. Gambar yang umum digunakan dalam deskripsi grafis adalah diagram titik (*dot diagram*) dan histogram. Dengan menggunakan histogram dapat diperoleh pengetahuan dimana lokasi kecenderungan berkumpulnya data dari batang tertinggi.

2. Deskripsi lokasi

Meskipun deskripsi grafis sudah menggambarkan karakteristik data namun sifatnya masih terlalu kasar dan kurang praktis untuk dilakukan, oleh karena itu masih diperlukan sebuah angka yang cukup dapat mewakili data yang ada serta dapat diperoleh dengan cara yang

lebih praktis dari pada menggambar. Ada beberapa ukuran yang umum digunakan untuk menyatakan lokasi tersebut, yaitu :

a. Rata-rata

Rata-rata berarti membuat menjadi rata dan nilai perataan tersebut dianggap sebagai lokasi pusat, titik berat, atau keseimbangan data

$$\bar{X} = \frac{x_1 + x_2 + \dots + x_n}{n} = \sum_{i=1}^n \frac{x_i}{n}$$

Gambar 2.2 Rumus Rata-rata.

b. Nilai Tengah

Nilai observasi secara harafiah bertempat ditengah-tengah. Langkah pertama yang tidak boleh dilupakan mengurutkan semua data dari terkecil hingga terbesar.

Berikut rumus nilai tengah jika rata-rata bernilai ganjil:

$$\text{Nilai median terletak pada data ke-} = [(n+1)/2].$$

Gambar 2.3 Rumus Median(Ganjil).

Berikut rumus nilai tengah jika rata-rata bernilai genap:

$$\text{Nilai median terletak pada data ke-} = \frac{\text{data ke-} [n/2] + \text{data ke-} [(n/2)+1]}{2}$$

Gambar 2.4 Rumus Median(Genap).

c. Modus

Mencari nilai observasi yang sering muncul. Apabila terdapat dua atau lebih nilai kekerapan munculnya sama, semua nilai tersebut juga disebut modus

3. Deskripsi keberagaman

Meskipun deskripsi lokasi sudah memberikan gambaran tentang lokasi pusat data (rata-rata, median, modus), namun belum memiliki gambaran atas keberagaman data. Untuk melengkapi gambaran terhadap data yang ada diperlukan ukuran keberagaman. Varians merupakan ukuran sebaran data yang menggunakan prinsip pencarian jarak antara setiap data dengan pusatnya (rata-rata)

$$s^2 = \frac{\sum_{i=1}^n (x_i - \bar{X})^2}{n - 1}$$

Gambar 2.5 Rumus Varians.

2.7 Metode *WaterFall*

Metode *Waterfall* adalah suatu proses pengembangan perangkat lunak berurutan, di mana kemajuan dipandang sebagai terus mengalir ke bawah (seperti air terjun) melewati *fase-fase* perencanaan, pemodelan, implementasi (konstruksi), dan pengujian. Berikut adalah gambar pengembangan perangkat lunak berurutan/ linear (Pressman, Roger S. 2001).

2.7.1 Tahapan Metode *Waterfall*

Dalam pengembangannya metode *waterfall* memiliki beberapa tahapan yang runtut: *requirement* (analisis kebutuhan), *design sistem* (*system design*), *Coding & Testing*, Penerapan Program, pemeliharaan.

a. *Requirement* (analisis kebutuhan).

Dalam langkah ini merupakan Analisis terhadap kebutuhan sistem. Pengumpulan data dalam tahap ini bisa melakukan sebuah penelitian, wawancara atau study literatur. Seseorang system analisis akan menggali informasi sebanyak-banyaknya dari user sehingga akan tercipta sebuah sistem komputer yang bisa melakukan tugas-tugas yang diinginkan oleh user tersebut. Tahapan

ini akan menghasilkan dokumen user *requirement* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan user dalam pembuatan sistem. Dokumen inilah yang akan menjadi acuan *system analysis* untuk menterjemahkan kedalam bahasa pemrograman.

b. *Design System* (desain sistem)

Proses design akan menterjemahkan syarat kebutuhan sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat koding. Proses ini berfokus pada : struktur data, arsitektur perangkat lunak, *representasi interface*, dan detail (algoritma) prosedural. Tahapan ini akan menghasilkan dokumen yang disebut *software requirement*. Dokumen inilah yang akan digunakan programmer untuk melakukan aktivitas pembuatan sistemnya.

c. *Coding & Testing* (penulisan *kode* program / *implementation*)

Coding merupakan penerjemahan desain dalam bahasa yang bisa dikenali oleh komputer. Dilakukan oleh programmer yang akan menterjemahkan transaksi yang diminta oleh user. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu sistem. Dalam artian penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem yang telah dibuat tadi. Tujuan *testing* adalah menemukan kesalahan-kesalahan terhadap sistem tersebut dan kemudian bisa diperbaiki.

d. Penerapan / Pengujian Program (*Integration & Testing*)

Tahapan ini bisa dikatakan final dalam pembuatan sebuah sistem. Setelah melakukan Analisis, design dan pengkodean maka sistem yang sudah jadi digunakan oleh user.

e. Pemeliharaan (*Operation & Maintenance*)

Perangkat lunak yang sudah disampaikan kepada pelanggan pasti akan mengalami perubahan. Perubahan tersebut bisa karena mengalami kesalahan karena perangkat lunak harus menyesuaikan dengan lingkungan (peripheral atau sistem operasi baru) baru, atau karena pelanggan membutuhkan perkembangan fungsional.

2.7.2 Manfaat Metode Waterfall

Keunggulan model pendekatan pengembangan *software* dengan metode *waterfall* adalah pencerminan kepraktisan rekayasa, yang membuat kualitas *software* tetap terjaga karena pengembangannya yang terstruktur dan terawasi. Disisi lain model ini merupakan jenis model yang bersifat dokumen lengkap, sehingga proses pemeliharaan dapat dilakukan dengan mudah. Akan tetapi dikarenakan dokumentasi yang lengkap dan sangat teknis, membuat pihak klien sulit membaca dokumen yang berujung pada sulitnya komunikasi antar pengembang dan klien. Dokumentasi kode program yang lengkap juga secara tak langsung menghapus ketergantungan pengembang terhadap pemrogram yang keluar dari tim pengembang. Hal ini sangat menguntungkan bagi pihak pengembang dikarenakan proses pengembangan perangkat lunak tetap dapat dilanjutkan tanpa bergantung pada pemrogram tertentu.

2.7.3 Kelemahan Metode *Waterfall*

Kelemahan pengembangan *software* dengan metode *waterfall* yang utama adalah lambatnya proses pengembangan perangkat lunak. Dikarenakan prosesnya yang satu persatu dan tidak bisa diloncat-loncat menjadikan model klasik ini sangat memakan waktu dalam pengembangannya. Disisi lain, pihak klien tidak dapat mencoba sistem sebelum sistem benar-benar selesai pembuatannya. Kelemahan yang lain adalah kinerja personil yang tidak optimal dan efisien karena terdapat proses menunggu suatu tahapan selesai terlebih dahulu.

Secara keseluruhan model pendekatan pengembangan *software* dengan metode *waterfall* cocok untuk pengembangan *software*/perangkat lunak dengan tingkat resiko yang kecil, dan memiliki ukuran yang kecil serta waktu pengembangan yang cukup panjang.

2.8 Pengertian Rancang

Rancang berarti merancang segala sesuatu (sebelum bertindak, mengerjakan, atau melakukan sesuatu). (Purwanto, 2008:1)

2.9 Pengertian Bangun

Bangun berarti cara menyusun atau susunan yang merupakan suatu wujud. (Komputer, 2007:1)

2.10 Pengertian Sistem

Kata sistem berasal dari bahasa Yunani yaitu kumpulan elemen yang saling berkait dan bertanggung jawab memproses masukan (*input*) sehingga menghasilkan keluaran (*output*).

Menurut perkembangannya ada beberapa pengertian mengenai sistem diantaranya :

- 1 Menurut Gordon B Davis.
Sistem adalah kumpulan dari elemen-elemen yang beroperasi bersama-sama untuk menyelesaikan suatu sasaran.
- 2 Menurut Jogiyanto HM, 2001
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.
- 3 Menurut Dr.Ricardus Eko Indrajit
Pengertian sistem adalah suatu kumpulan dari berbagai prosedur yang dirancang dan disusun sedemikian rupa untuk mencapai suatu sasaran objektif yang telah ditetapkan.

Dari beberapa pengertian diatas dapat disimpulkan bahwa sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai tujuan tertentu. Di dalam suatu sistem diperlukan adanya keterkaitan yang terdiri dari elemen-elemen yaitu:

1. Tujuan
Merupakan tujuan dari sistem tersebut dimana komputer digunakan untuk mengurangi dan membantu tugas-tugas yang di lakukan oleh manusia dalam pengolahan data.
2. Kontrol
Merupakan pengawasan dari pelaksanaan pencapaian tujuan sistem pengawas dapat berupa:
 - a. Kontrol pemasukan data (*input*)
 - b. Kontrol pengeluaran data (*output*)
 - c. Kontrol pengoperasian (*processing*)
3. *Input*
Merupakan kajian dari sistem yang bertugas menerima data, dimana data yang masuk dapat berupa: asal masukan, jenis masukan.

4. Transformasi

Sistem komputer yang bertugas memproses data masukan (*input*) menjadi keluaran (*output*) sesuai dengan keinginan atau tujuan.

5. Output

Sistem komputer yang bertugas menghasilkan keluaran, tugasnya antara lain menghasilkan laporan dan grafik.

6. Umpan balik

Umpan balik bertujuan untuk melihat kembali apakah sistem telah berjalan sesuai dengan tujuan. Umpan balik yang dilakukan dapat berupa perbaikan dan pemeliharaan.

2.11 Pengertian Informasi

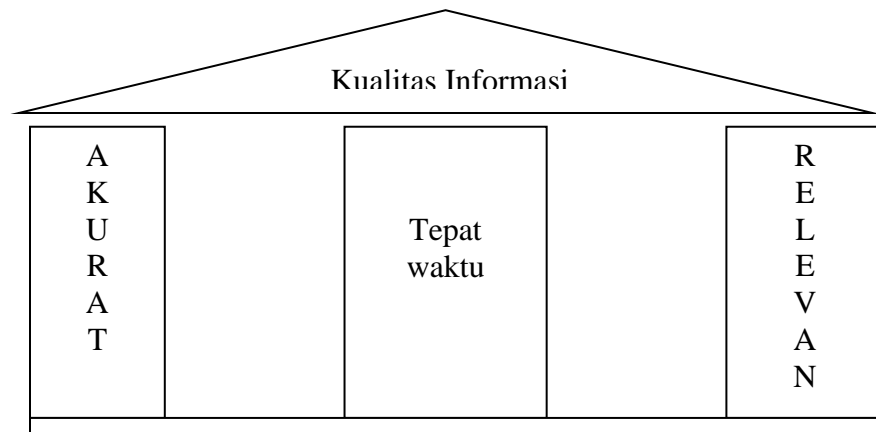
Terdapat beberapa definisi tentang pengertian informasi antara lain:

- a. Data yang di olah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerima.
- b. Sesuatu yang nyata atau setengah nyata yang dapat mengurangi derajat ketidakpastian tentang suatu keadaan atau kejadian.

Berdasarkan definisi diatas maka dapat disimpulkan informasi merupakan data yang telah diproses atau diolah yang memiliki arti penting bagi si penerima dan dapat mengurangi derajat ketidakpastian tentang suatu keadaan atau suatu kejadian.(Raymond McLeod. Jr,2001)

2.11.1 Kualitas Informasi

Kualitas dari suatu informasi (*quality of information*) tergantung dari 3 hal, yaitu informasi harus akurat (*accurate*), tepat pada waktunya (*timeliness*) dan relevan (*relevance*). *John Burch* dan *Gari Grudnitski* menggambarkan kualitas dari informasi dengan bentuk bangunan yang ditunjang oleh tiga buah pilar



Gambar 2.6 : Pilar kualitas informasi

Sumber : Analisis & Desain Sistem Informasi (Jogiyanto HM, 2001)

Kualitas Informasi tergantung dari 3 hal :

1. Akurat

Berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi memungkinkan banyak terjadi gangguan (*noise*) yang dapat merubah atau merusak informasi tersebut.

2. Tepat pada waktunya

Berarti informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi, karena informasi merupakan landasan didalam pengambilan keputusan. Bila pengambilan keputusan terlambat, maka dapat berakibat fatal untuk organisasi.

3. Relevan

Berarti informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan lainnya berbeda. Misalnya informasi mengenai asal usul kerusakan mesin

produksi kepada akuntan perusahaan adalah kurang relevan dan akan lebih relevan bila ditunjukkan kepada ahli teknik perusahaan. Sebaliknya informasi mengenai harga pokok produksi untuk ahli teknik merupakan informasi yang kurang relevan, tetapi relevan untuk akuntan.

2.11.2 Nilai Informasi

Nilai dari informasi (*value of information*) ditentukan dari dua hal, yaitu manfaat dan biaya mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya, akan tetapi perlu diperhatikan bahwa informasi yang digunakan didalam suatu sistem informasi umumnya digunakan untuk beberapa kegunaan. Sehingga tidak memungkinkan dan sulit untuk menghubungkan suatu bagian informasi pada suatu masalah yang tertentu dengan biaya untuk memperolehnya, karena sebagian besar informasi dinikmati tidak hanya oleh satu pihak dalam perusahaan. Lebih lanjut sebagian besar informasi tidak dapat persis ditaksir keuntungannya dengan satuan nilai uang, tetapi dapat ditaksir nilai efektivitasnya. Pengukuran nilai informasi biasanya dihubungkan dengan analisis *cost effectiveness* atau *cost benefit*.

2.12 Pengertian *Monitoring*

Monitoring adalah pemantauan yang dapat dijelaskan sebagai kesadaran tentang apa yang ingin diketahui. Proses rutin pengumpulan data dan pengukuran kemajuan atas objektif program. Memantau perubahan, yang fokus pada proses dan keluaran.

(<http://id.wikipedia.org/wiki/Monitoring>)

2.13 Perancangan Sistem

Merupakan pengembangan sistem baru di sistem lama yang ada, dimana masalah-masalah yang terjadi pada sistem lama diharapkan sudah teratasi pada sistem yang baru.

2.13.1 Pengertian Perancangan Sistem

Perancangan sistem merupakan pengembangan sistem baru dari sistem lama yang ada, dimana masalah-masalah yang terjadi pada sistem lama diharapkan sudah diatasi pada sistem. Tujuan dari perancangan sistem adalah untuk mendesain sistem baru yang dapat menyelesaikan masalah-masalah yang di hadapi perusahaan yang diperoleh dari pemilihan alternatif yang baik. (Merle P. Martin, 2000).

Kegiatan yang di lakukan dalam perancangan sistem adalah:


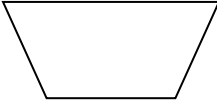
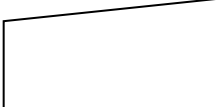

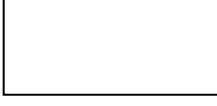
1. Memberikan bentuk laporan sistem dan dokumentasinya yang menghasilkan bentuk dari dokumentasi keluaran (*Output Design*)
2. Memberikan bentuk masukan di dokumen dan di layar ke sistem informasi yang menghasilkan bentuk dari dokumentasi masukan (*Input Design*).
3. Memberikan bentuk file-file yang dibutuhkan dalam sistem informasi yang menghasilkan bentuk dari dokumentasi file (*File Design*)


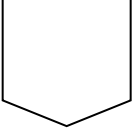
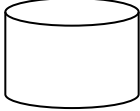
2.13.2 Alat Bantu dalam Perancangan Sistem

2.13.2.1 Sistem Procedure Diagram

Sistem procedure diagram digunakan untuk membuat *flow of document* (manual) maupun *flow of system* (komputerisasi). Fungsi diagram ini untuk mendefinisikan hubungan antara bagian (pelaku proses), proses (manual atau

berbasis komputer) dan aliran data (dalam bentuk dokumen masukan dan keluaran).

Simbol	Fungsi
<p data-bbox="646 583 873 613">Simbol Dokumen</p> 	<p data-bbox="954 634 1373 781">Menunjukkan dokumen masukan (formulir dan dokumen keluaran laporan).</p>
<p data-bbox="613 814 906 844">Simbol Proses Manual</p> 	<p data-bbox="954 865 1373 1012">Menunjukkan proses kerja manual. Seperti ACC, Terima gaji</p>
<p data-bbox="604 1054 915 1083">Simbol Pemasukan Data</p> 	<p data-bbox="954 1104 1373 1188">Menunjukkan input/masukan data</p>
<p data-bbox="646 1327 873 1356">Simbol Transaksi</p> 	<p data-bbox="954 1377 1373 1524">Mendefinisikan penyimpanan yang bukan master berupa transaksi.</p>
<p data-bbox="652 1549 867 1633">Simbol Berbasis Komputer</p> 	<p data-bbox="954 1600 1373 1801">Mendefinisikan proses yang dilakukan dengan komputer seperti: penghitungan, pencetakan laporan, penjualan.</p>

<p>Simbol Display</p> 	<p>Menunjukkan output yang akan ditampilkan dilayar monitor.</p>
<p>Simbol Penghubung</p> 	<p>Menunjukkan penghubung kehalaman lain atau pindah halaman.</p>
<p>Simbol Master</p> 	<p>Menunjukkan file penyimpanan untuk data-data master.</p>

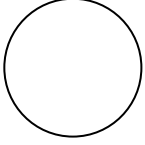
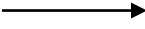
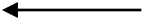
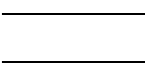

Tabel 2.1 : Prosedur Diagram Sistem

Sumber : Analisis dan Desain (Jogiyanto HM, 2001)

2.13.2.2 *Data Flow Diagram (DFD) atau Diagram Arus Data (DAD)*

DFD adalah diagram dari aliran data melalui sebuah sistem. DFD dapat di bagi menjadi dua yaitu DFD fisik dan DFD logis. DFD digunakan untuk komunikasi antara analis dan pemakai.

Simbol-simbol yang di gunakan dalam DFD antara lain :

Simbol	Fungsi
	<p>PROSES Digunakan untuk menunjukkan transformasi dari masukan menjadi keluaran.</p>
 	<p>ALIRAN DATA Digunakan untuk menggambarkan gerakan paket data atau informasi dari satu bagian ke bagian lain dari sistem dimana penyimpanan mewakili bakal penyimpanan data.</p>
	<p>PENYIMPANAN Dapat digunakan untuk mendefinisikan file atau basis data atau sering kali mendefinisikan bagaimana penyimpanan diimplementasikan dalam sistem komputer.</p>
	<p>TERMINATOR (<i>asal /tujuan data</i>) Melambangkan orang atau kelompok orang yang merupakan asal data atau tujuan informasi</p>

Tabel 2.2 : Simbol DFD *Leveled*

Sumber : Analisis dan Desain (Jogiyanto HM, 2001)

Diagram arus data itu sendiri dibagi menjadi dua bagian yaitu :

a. DFD Context

Merupakan alat untuk menjelaskan struktur analisis. Pendekatan ini mencoba untuk menggambarkan sistem pertama kali secara garis besar dan memecahkannya menjadi bagian yang terinci yang disebut dengan lower level. Dan yang pertama kali digambar adalah level yang teratas sehingga disebut Diagram Context.

b. DFD level

Setelah context diagram dirancang kemudian akan digambar lebih terinci lagi yang disebut dengan *over view* diagram (*level 0*). Tiap-tiap proses di *over view* diagram akan digambar lebih terinci lagi dan disebut dengan *level 1*, dan kemudian di teruskan ke *level* berikutnya sampai tiap-tiap proses tidak dapat digambar lebih terinci.



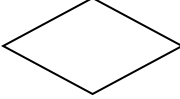

2.13.2.3 ERD (*Entity Relationship Diagram*)

ERD adalah merupakan suatu model data untuk mengilustrasikan desain logika dari skema *database*. (Marle P. Martin 2000). ERD terdiri dari tiga bagian :

1. Entitas, yaitu suatu objek yang terdiri dari kumpulan data dari *database*.
2. Relasi, yaitu pengukur antar entitas
3. Atribut, yaitu menggambarkan hubungan antara entitas dan relasi.

ERD menggunakan sejumlah notasi dan simbol untuk menggambarkan struktur dan hubungan antar data.

Simbol-simbol yang digunakan yaitu.

Simbol	Fungsi
<i>Entity</i> 	Digunakan untuk menggambarkan obyek yang dapat didefinisikan dalam lingkungan pemakai sistem.
Atribut 	Digunakan untuk menggambarkan elemen-elemen dari suatu entity, yang menggambarkan karakter entity
Hubungan 	<i>Entity</i> dapat berhubungan satu dengan yang lain. Hubungan ini disebut <i>Relationship</i> .
Garis 	Digunakan untuk menghubungkan <i>entity</i> dengan <i>entity</i> dan <i>entity</i> dengan atribut.

Tabel 2.3 : Simbol-Simbol ERD

Sumber : Analisis dan Desain (Jogiyanto HM, 2001)

2.13.3 Merancang Database

Merancang *database* mempunyai tujuan yaitu meminimumkan pengulangan data dan independensi data (Fatansyah .Ir, 2000). Perancangan *database* diperlukan untuk menghindari permasalahan di dalam *database*.

2.13.4 Perancangan Masukan dan Keluaran (*Input Output Design*)

a) Desain *Input*

Untuk membuat laporan baru ke dalam perkembangan sistem adalah dengan menggunakan dokumen dan prosedur desain input sehingga perkembangan sistem dapat berkembang dengan cepat dan akurat pada sistem informasi. (Merle P. Martin, 2000).

Tujuan desain input :

1. Untuk mengefektifkan biaya pemasukan data.
2. Untuk menjamin kemasukan data dapat diterima dan dimengerti oleh pemakai.

b) Desain *Output*

Desain *output* merupakan keberhasilan dari sistem informasi. Setelah desain *input* dan *file* berlangsung maka akan terjadi desain *output* (Marle P. Martin, 1999).

2.14 Konsep Database MySQL

Structured Query Language yang biasa disingkat dengan SQL merupakan bahasa yang digunakan untuk berkomunikasi dengan database. Bahasa ini merupakan bahasa non prosedural yang digunakan untuk mendefinisikan, memanipulasi seperti (menyimpan, mengambil, mengupdate dan menghapus data).

MySQL menggunakan bahasa standart SQL (*Structured Query Language*) sebagai bahasa interaktif dalam mengelola data.

Keunggulan database MySQL antara lain (Simmarta, 2006) :

1. Cepat : Tujuan utama dari pengembangan MySQL adalah kecepatan sebagai konsekuensi software yang dirancang untuk kecepatan

2. Tidak Mahal : MySQL adalah Cuma-Cuma dibawah lisensi GPL (*General Public Licence*) *open source*, sementara pembiayaan untuk *licence* komersialnya sangatlah pantas.
3. Mudah digunakan : Dapat membangun dan berinteraksi dengan database MySQL hanya dengan sedikit pernyataan (*statement*) sederhana dalam bahasa SQL, yang menjadi bahasa standar untuk komunikasi dengan RDBMS (*Relation Database Management System*)
4. Dapat berjalan pada beberapa sistem operasi. MySQL berjalan pada sistem yang beragam, seperti Windows, Linux, Mac OS, kebanyakan variasi Unix (termasuk Solaris, AIX, dan DEC Unix) FreeBSD, OS/2, Irix, dan lainnya.
5. Dukungan teknis secara luas tersedia. MySQL menyediakan dukungan secara Cuma-Cuma untuk pengguna *via mailing list*.
6. Aman. MySQL adalah sistem otorasi fleksibel yang mengijinkan beberapa atau semua *privilage database*.
7. Mendukung database yang besar. MySQL mendukung database sampai 50 juta baris atau lebih. Batasan ukuran file secara *default* untuk tabel adalah 4GB, tetapi dapat dinaikkan hingga 8TB jika sistem operasi mendukung.

Secara implisit pengguna tidak perlu secara langsung menulis perintah SQL, karena sudah termasuk di dalam pilihan membuat *table/view* baru (*create table atau crate view*) melalui “*table / view editor windows*”, maka *Oracle Power Objects* akan secara otomatis mengirimkan perintah SQL yang sesuai kepada database untuk diproses.

Secara eksplisit, penggunaan SQL dapat dilakukan dengan dua cara, yaitu:

1. Penggunaan sebagian perintah SQL yaitu dengan memanfaatkan *method query where ()* pada form dan properti *order by pada report*.
2. Penggunaan secara penuh perintah SQL dengan bantuan *EXEECT SQL* dan *SqlLookUp ()*.

2.15 Borland Delphi 7.0

Dasar bahasa pemrograman yang digunakan dalam Delphi adalah Pascal, sebuah bahasa yang didesain khusus oleh Niklaus Wirth untuk mengajarkan pemrograman terstruktur. Dibandingkan dengan bahasa generasi ketiga lainnya, seperti bahasa C, Pascal lebih mudah dipelajari dan digunakan. Hal ini karena Pascal memiliki struktur bahasa seperti bahasa Inggris sehingga mudah untuk dibaca. Tipe data dalam pascal antara lain adalah : *Integer, Real, Boolean, Char, String, Pointer, Pchar*.

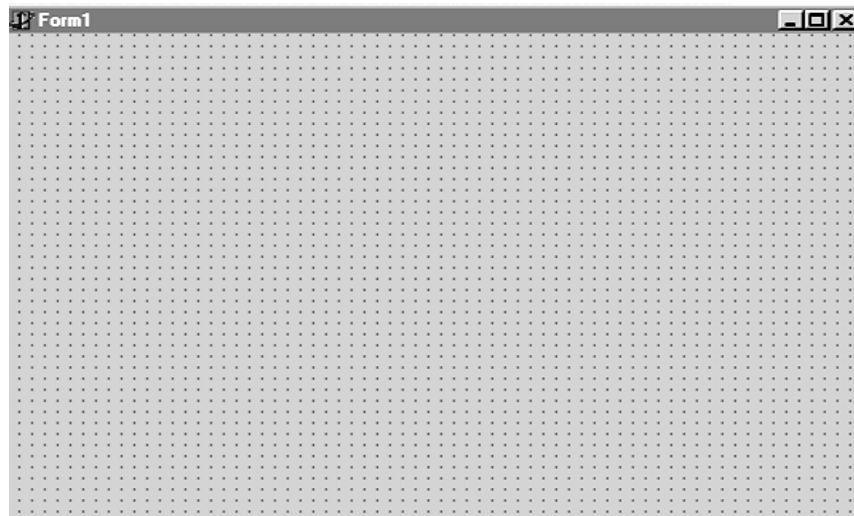
Pascal dalam Delphi berbeda dengan Pascal pada versi-versi sebelumnya, bahkan bila dibandingkan dengan Borland Pascal Versi 7.0, objek Pascal dalam Pascal 7 merupakan pengembangan kompiler-kompiler Pascal versi sebelumnya. Sekarang dalam Delphi bentuk objek Pascal lebih ditingkatkan lagi oleh Borland. Delphi dikembangkan dengan tujuan untuk menetapkan standar baru bahasa Pascal. Akan tetapi Delphi masih mampu mengenal bentuk-bentuk lama objek Pascal dari versi kompiler yang lama.

Bahasa pascal adalah bahasa yang *strongly-typed*. Artinya, variabel ini harus selalu diberi nilai yang tipenya sama dengan deklarasinya. Bila tidak maka kompiler akan memberikan pesan kesalahan. Sedangkan bahasa yang *weak-typed* seperti bahasa C, variabelnya dapat menerima nilai yang berbeda dengan type deklarasinya.

2.15.1 Form Dan Editor Program

Windows Form dan *Windows Editor* program berhubungan sangat erat. Istilah Unit dalam Delphi adalah sebuah file program sumber (file PAS) untuk suatu form.

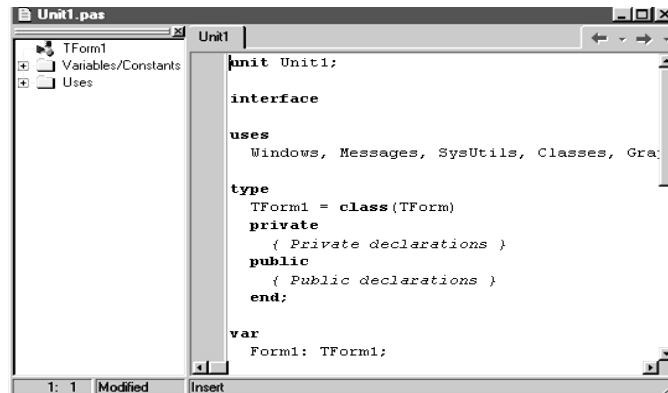
Window form adalah tempat dimana semua komponen visual dan komponen non visual diletakkan untuk membentuk sebuah aplikasi.



Gambar 2.7 : *Window Form* Pada Delphi

(Sumber : *Teddy, Pemrograman Delphi untuk Pemula: IDE dan Struktur Pemrograman, Kuliah Umum IlmuKomputer.Com, 2003*)

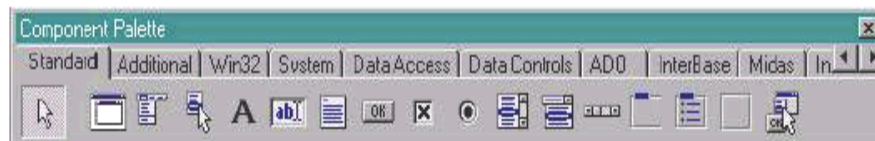
Sedangkan pada saat program *editor* aktif, akan dilengkapi fasilitas *editor* yang berdaya guna. Cara kerja *editor* ini sama dengan aplikasi *Brief*, sebuah *editor* yang sangat populer dikalangan para pemrogram.



Gambar 2.8 : *Window Editor* Pada Delphi

(Sumber : Teddy, *Pemrograman Delphi untuk Pemula: IDE dan Struktur Pemrograman*, Kuliah Umum IlmuKomputer.Com, 2003)

2.15.2 Komponen



Gambar 2.9: *Component Palette*

(Sumber : Teddy, *Pemrograman Delphi untuk Pemula: IDE dan Struktur Pemrograman*, Kuliah Umum IlmuKomputer.Com, 2003)

Dalam membuat program, Delphi telah menyediakan banyak kemudahan, yaitu dengan disediakannya komponen-komponen. Komponen ini merupakan sebuah *procedure*/program yang sudah di kompilasi dan langsung dapat digunakan, sesuai dengan fungsinya masing-masing. Untuk menggunakan komponen ini kita dapat mengklik komponen yang diinginkan, kemudian kita klik di form, maka komponen tersebut akan muncul di form.

Kode yang akan dilihat, serupa dengan struktur Bahasa Pascal. Delphi adalah bahasa pemrograman berbasis objek, artinya pendekatan

pembuatan program melalui objek-objek yang ada. Misalnya objek form, text dsb. Setiap objek akan memiliki properti (atribut) dan method yang diaktifkan / dipicu oleh *event*.

Kegunaan beberapa komponen :

a. *Button/ Bitbtn*

Biasa digunakan sebagai tombol kendali. Perbedaan antara *bitbtn* dengan *button*: pada *bitbtn* kita dapat menyisipkan warna pada tombol dan icon tertentu, lain halnya bila kita menggunakan *button*.

b. *Panel*

Panel berfungsi untuk mengelompokkan komponen-komponen didalamnya.

c. *Label*

Kita dapat menamakan atau memberi keterangan pada program

d. *Edit*

Edit berfungsi sebagai masukan data (*input*) dalam bentuk *string*, dari bentuk *string* ini kita dapat mengolahnya menjadi bentuk *integer* atau bentuk lainnya yang kemudian dapat digunakan untuk operasi selanjutnya.

e. *Chart*

Data-data yang telah kita analisa, dapat kita tampilkan ke dalam grafik, sehingga memudahkan kita untuk menganalisanya.

f. *Stringgrid*

Stringgrid berguna untuk menaruh data *string* kedalam bentuk kolom tabel, seperti pada Excel. Kita harus mengubah *type* data ke dalam bentuk *string* bila data yang ingin kita tampilkan data bukan *string*.

g. *Popup Menu*

Popup Menu berfungsi sebagai perintah yang aktif bila kita mengklik kanan *mouse*, Untuk mengaktifkannya kita harus mengaktifkan

Popup Menu pada komponen yang diinginkan, caranya : ubah pada *object inspector*.

h. *MainMenu*

Contoh *Main Menu* adalah *Option* pada tiap aplikasi program, dengan komponen ini, kita bisa menaruh fungsi-fungsi program seperti pada aplikasi umumnya.

i. *ComboBox*

Combo Box berfungsi sebagai petunjuk untuk pemilihan berbagai masukan.

j. *CheckBox*

Bila komponen ini di *check* maka ada aplikasi yang bisa disetting untuk bekerja dibawahnya.

k. *RadioButton*

Prinsip kerjanya hampir sama dengan *check box*, cuma tampilannya saja yang berbeda.

l. *Media Player*

Biasa digunakan untuk menyalakan atau memainkan musik (format *wav* atau *midi*) dan menjalankan film (format *avi*).

m. *Timer*

Timer berfungsi sebagai jam yang telah disediakan Delphi. Dengan *timer* kita juga dapat mendecode time, sehingga dapat terjadi akusisi data.

n. *Clientsocket*

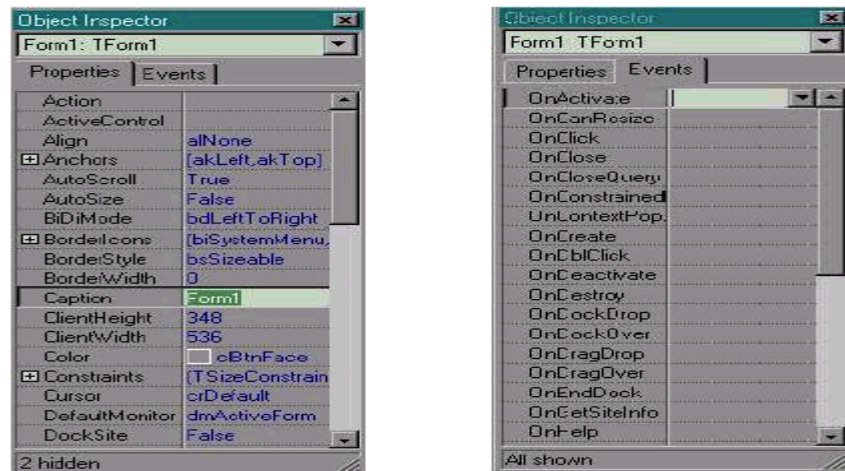
Digunakan di *client* pada saat koneksi ke *server*.

o. *Serversocket*

Digunakan di *server*, yang akan menyediakan layanan koneksi ke *client*. Masih banyak komponen lain. Untuk mempelajari lebih lanjut dapat dibaca pada *help menu*.

2.15.3 Object Inspector

Object inspector (gambar dibawah) berguna sebagai *options* dari masing-masing komponen. Dengan object inspector ini kita dapat memanipulasi komponen yang kita gunakan (walaupun sebenarnya kita juga dapat menggunakannya dengan menuliskannya lewat text mode).



Gambar 2.10 : Object Inspektor

(Sumber : Teddy, Pemrograman Delphi untuk Pemula: IDE dan Struktur Pemrograman, Kuliah Umum IlmuKomputer.Com, 2003)

Beberapa contoh pada *Properties* :

a. Font

Menunjukkan *font* yang akan kita gunakan. Dalam menu ini juga terdapat macam-macam *font* berikut *style* dan ukurannya.

b. Caption

Merupakan judul dari *form* itu. Perhatikan perbedaannya dengan *name*.

c. Name

Merupakan nama dari komponen itu. Biasanya bila kita mengambil sebuah komponen, Delphi otomatis menyediakan nama sesuai dengan nama komponen tersebut. Namun nama ini dapat dengan

mudah kita ubah agar mudah mengingatnya. Berbeda dengan *caption* pada *caption* kita tidak mengubah nama komponen, melainkan hanya mengubah judul yang tertera pada komponen tersebut. Sehingga bila kita ingin menggunakan komponen tersebut, yang kita gunakan adalah namanya, dan bukan nama pada *caption*.

d. *Enabled*

Menunjukkan bahwa komponen ini bisa digunakan atau tidak. Perintah ini cukup penting bila kita tidak ingin *user* menekan tombol yang tidak diinginkan.

e. *Height*

Merupakan tinggi dari komponen yang bakalan kita taruh pada *form*, atau komponen lainnya.

f. *Width*

Merupakan lebar komoponen.

g. *Popup Menu*

Fungsi ini berguna saat kita mengkilik kanan dan akan tampil perintah. Pada penggunaanya kita mengambil komponen *popup menu* dan menset *popup menu* pada *form*.

h. *Auto Size*

Fungsi ini berisi mengenai apakah *form* yang kita miliki bisa diperbesar atau tidak.

i. *Text*

Biasanya tedapat pada komponen edit dan masuk edit. Kegunaanya untuk menaruh kata-kata pada komonen ini.

j. *Color*

Warna pada komponen. Kita dapat mengubah warna-warna tiap komponen (hanya yang disediakan) dengan *point* ini.

k. Visible

Berfungsi untuk menampilkan dan tidak menampilkan komponen yang kita inginkan.

l. Hint

Bila kursor kita diatas komponen, maka akan muncul suatu keterangan. Keterangan ini yang disebut dengan *hint*. Jangan lupa untuk mengganti properti *show hint* menjadi *true* untuk mengaktifkannya.

m. Borderstyle

Menunjukkan berbagai macam bingkai yang diinginkan. Masih banyak properti lainnya, bisa dipelajari lebih lanjut.

Dengan *Object inspector* kita juga dapat menentukan *event* apa yang akan kita buat dengan komponen tersebut.

Beberapa contoh *Event*.

a. Onclick

Bila kita mengklik komponen tersebut maka prosedur yang kita inginkan akan dijalankan oleh program tersebut.

b. Onkeypress

Bila kita menekan suatu *key* (tombol) maka komponen tersebut akan aktif perlu diperhatikan pada *event* ini, Delphi mengenal *character* yang kita tekan melalui kode yang masuk kedalam variabel *key*.

Lihat contoh ini.

```
Procedure TForm1.Edit1KeyPress(Sender: TObject; var Key:
Char);
begin
if key =#13 then .....(isikan perintah)
end;
```

Bila key =#13 (enter) maka perintah yang ikuti akan dijalankan.

2.15.4 File-File Pada Delphi

a. *.pas

Merupakan *source file*, disini akan disimpan kode pascal yang kita tulis.

b. *.dpr

Merupakan *project file*. Sebagai *project file*, *file* ini berguna untuk menggabungkan satu atau lebih *file-file source* (*.pas).

c. *.dcu (Delphi Compiled Unit)

Pada saat kita membuat sebuah aplikasi, Delphi akan membuat *file* ini. *File* ini berfungsi untuk me-*link*-kan kita dengan file lain (*.dcu) sehingga kita bisa membuat beberapa *form* yang terhubung satu dengan yang lainnya.

d. *.dfm (Delphi Form)

File ini berisi informasi mengenai data-data *form*.

e. *.dof (Delphi Project Options)

Kita dapat mengubah *icon* aplikasi tersebut, dan datanya disimpan pada *file* ini.

f. *.exe

Merupakan *application file* setelah kita *compile* program kita.

2.15.5 Type Data Pada Delphi

Pada umumnya Delphi menggunakan type data yang sama dengan pascal. Namun type data pada Delphi lebih lengkap dibanding pascal. Tipe data yang biasa digunakan pada Delphi :

1. String

Biasa digunakan untuk masukan data *string* (*String* merupakan gabungan beberapa *character*).

2. ***Integer***

Merupakan bilangan asli (1, 2, 3, 4, ...).

3. ***Real***

Bilangan bulat (berupa pecahan dan *integer*).

4. ***Byte***

Besar dari type ini hanya 8 *bit*.

5. ***Word***

Besar bilangan ini 16 *bit*.

6. ***Date And Time***

Type waktu dan tanggal yang disediakan Delphi

2.15.6 Konversi Type Data

Dalam pengolahan data biasanya kita memerlukan suatu konversi type data, banyak sekali konversi data yang bisa dilakukan pada Delphi. Contoh konversi type data :

b. *Strtoint*

Mengubah type data *string* ke *integer*.

c. *Inttostr*

Mengubah type data *integer* ke *string*.

d. *Timetostr*

Mengubah type data *time* ke dalam bentuk *string*.

e. *Strtfloat*

Mengubah type data *string* ke dalam bentuk *real*.

f. *Floattostr*

Mengubah type data *real* ke dalam bentuk *string*.