

BAB II

LANDASAN TEORI

2. 1 Tinjauan Studi

Penelitian tentang Penerapan Data Mining untuk memprediksi penjualan kendaraan Mobil dengan algoritma Naïve Bayes telah dilakukan oleh beberapa peneliti terdahulu antara lain:

1. Susanto, 2015, Dengan Judul Penelitian Penerapan Data Mining *Classification* Untuk Prediksi Perilaku Pola Pembelian Terhadap Waktu Transaksi Menggunakan Metode Naïve Bayes. Semakin banyak pembeli, semakin besar kemungkinan transaksi yang dapat diperoleh. Jumlah pembeli setiap harinya berbeda-beda dan memiliki persentase yang tidak sama dalam melakukan transaksi penjualan. Penelitian ini bertujuan dengan memprediksi penjualan berdasarkan rata-rata jumlah pembeli. Algoritma Naïve Bayes bertujuan untuk melakukan klasifikasi data pada kelas tertentu, kemudian pola tersebut dapat digunakan untuk memperkirakan data transaksi penjualan dalam memprediksi perilaku pola pembeli. Hasil yang dicapai pada penelitian ini mencapai akurasi 97%.
2. Sulistiawati Rahayu Ningsi Ahmad, 2014, penelitian ini berjudul Prediksi Tingkat Penjualan Dengan Menggunakan Metode Naive Bayes Berbasis Forward Selection Untuk Produk Nestle, pada penelitian menerapkan

algoritma Naive Bayes berbasis Forward selection dalam memprediksi tingkat penjualan produk Nestle mampu memprediksi tingkat penjualan produk Nestle pada tahun ke depan hal ini terbukti dengan perolehan nilai akurasi Naive bayes berbasis *Forward selection* mampu mencapai nilai akurasi 94.69 %. Hasil akurasi yang dicapai lebih baik jika dibandingkan dengan menggunakan metode Naïve bayes tanpa forward Selection.

3. Diana Laily Fithri, 2014, dengan judul penelitian Sistem Pendukung Keputusan Untuk Memprediksi Kelulusan Mahasiswa Menggunakan Metode *Naïve Bayes*, Penelitian ini bertujuan untuk menghasilkan sistem pendukung keputusan untuk memprediksi kelulusan mahasiswa di tingkat perguruan tinggi. Dalam penelitian ini hasil yang dicapai memiliki akurasi untuk tepat waktu sebesar 93% dan akurasi untuk terlambat sebesar 71% dengan menggunakan metode Naïve Bayes yang semakin optimal dengan menentukan mahasiswa lulus tepat waktu atau terlambat.
4. Burhan Alfironi Muktamar, 2013, Penelitian Ini Berjudul Implementasi Data Mining Dengan Naive Bayes *Classifier* Untuk Mendukung Strategi Pemasaran Di Bagian Humas Stmik Amikom Yogyakarta, tujuan dari penelitian ini dapat memberikan informasi penting seperti hasil prediksi yang menarik dalam studi siswa yang dapat digunakan untuk membantu Tim Pemasaran STMIK AMIKOM Yogyakarta dan meningkatkan jumlah mahasiswa baru yang mendaftar. Hasil prediksi yang dicapai dengan akurasi 92,7 %.
5. Aris Nugroho, 2013, dengan judul penelitian klasifikasi naïve bayes untuk prediksi kelahiran pada data ibu hamil, dalam bidang kesehatan terkhusus

Kesehatan Ibu dan Anak, memprediksi suatu kejadian resiko tinggi (resti) pada kehamilan ibu sehingga kemunculan resiko secara dini bisa ditanggulangi akan sangat mempengaruhi penurunan Angka Kematian Ibu (AKI) maupun Angka Kematian Bayi (AKB). Dengan Model pendekatan Bayesian berupa Klasifikasi Naïve Bayes dengan HMAP (Hipotesis Maksimum A Posteriori) dipakai memprediksi kelahiran yang akan dialami ibu hamil dengan karakteristik Usia ibu, Tinggi Badan, Jumlah Hb, Tekanan Darah, Riwayat Kehamilan lalu dan Penyakit bawaan. Hasil prediksi yang dicapai pada penelitian ini adalah 88 %.

2.2 Tinjauan Pustaka

2.2.1 PT. Hasrat Abadi

PT Hasrat Abadi merupakan perusahaan yang bergerak dalam bidang industri yang berkantor pada Cabang yang berlokasi di Jalan Raya Limboto, Dusun 3 Pentadio Barat, Telaga Biru, Gorontalo ini terletak sangat strategis. Cabang ini didirikan di lahan seluas 5.285 m², dengan luas bangunan 2.712 m². Dilengkapi dengan ruang tunggu yang nyaman untuk pelanggan Sales dan Service, ruang pameran (showroom) kendaraan yang luas, 15 service stall untuk melayani perawatan dan perbaikan berbagai macam kendaraan Toyota.

2.2.2 Prediksi penjualan mobil

Prediksi penjualan adalah kegiatan untuk mengestimasi besarnya penjualan kendaraan atau jasa oleh produsen, distributor pada periode waktu dan wilayah pemasaran tertentu. Peramalan penjualan merupakan bagian fungsi manajemen sebagai salah satu kontributor keberhasilan sebuah perusahaan. Ketika

penjualan diprediksi dengan akurat maka pemenuhan permintaan konsumen dapat diusahakan tepat waktu, kerjasama perusahaan dengan relasi tetap terjaga dengan baik, kepuasan konsumen terpenuhi, perusahaan dapat mengatasi hilangnya penjualan atau kehabisan stok, mencegah pelanggan lari ke kompetitor. Di sisi lain perusahaan dapat menentukan keputusan kebijakan rencana produksi, persediaan barang, investasi aktiva dan *cash flow*. Dengan kata lain, tidak ada perusahaan yang dapat menghindar dari kegiatan memperkirakan atau meramalkan penjualan untuk keperluan perencanaan aktivitas-aktivitas yang harus dilakukan (Sunneng Sandino Berutu, 2015).

1. Penjualan

Menurut (Baskoro Fariz Adhi, 2014). Pada masa sekarang mobil adalah salah satu alat transportasi darat yang penting pada saat sekarang ini. Memiliki mobil bagi sebagian besar kalangan masyarakat pada saat ini bagaikan suatu hal yang pokok dimana dapat membantu mereka dalam beraktivitas khususnya dalam bekerja. Oleh karena itu, para produsen mobil berlomba – lomba untuk menciptakan mobil dengan keunggulan dan kelebihan yang berbeda sehingga dipasaran jumlah mobil ini sangat banyak dan bervariasi.

Dalam hal ini Penjualan merupakan salah satu indikator paling penting dalam sebuah perusahaan, bila tingkat penjualan yang dihasilkan oleh perusahaan tersebut besar, maka laba yang dihasilkan perusahaan itu akan besar pula sehingga perusahaan dapat bertahan dalam persaingan bisnis dan bisa mengembangkan usahanya (Baskoro Fariz Adhi, 2014).

2. Prediksi

Prediksi adalah memperkirakan sesuatu yang akan terjadi pada masa yang mendatang. Prediksi didapatkan melalui metode ilmiah maupun hanya subjektif belaka. Prediksi juga dapat digunakan dalam pengklasifikasian, tidak hanya untuk memprediksi *time series*, karena sifatnya yang bisa menghasilkan *class* berdasarkan berbagai atribut yang kita sediakan (Susanto, 2015).

2.2.3 Data Mining

Data mining adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai *database* besar (Turban, 2005).

Istilah data mining memiliki hakikat sebagai disiplin ilmu yang tujuan utamanya adalah untuk menemukan, menggali, atau menambang pengetahuan dari data atau informasi yang kita miliki. Data mining, sering juga disebut sebagai *Knowledge Discovery in Database* (KDD). KDD adalah kegiatan yang meliputi pengumpulan, pemakaian data, historis untuk menemukan keteraturan, pola atau hubungan dalam set data berukuran besar (Santosa, B. 2007).

1. Metode Pelatihan

Secara garis besar metode pelatihan yang digunakan dalam teknik-teknik data mining dibedakan ke dalam dua pendekatan, yaitu : Unsupervised learning, metode ini diterapkan tanpa adanya latihan (training) dan tanpa ada guru (teacher). Guru di sini adalah label dari data. Sedangkan Supervised learning, yaitu metode

belajar dengan adanya latihan dan pelatih. Dalam pendekatan ini, untuk menemukan fungsi keputusan, fungsi pemisah atau fungsi regresi, digunakan beberapa contoh data yang mempunyai output atau label selama proses training.

2. Pengelompokan Data Mining

Ada beberapa teknik yang dimiliki *data mining* berdasarkan tugas yang bisa dilakukan, yaitu :

1. Deskripsi

Para peneliti biasanya mencoba menemukan cara untuk mendeskripsikan pola dan trend yang tersembunyi dalam data.

2. Estimasi

Estimasi mirip dengan klasifikasi, kecuali variabel tujuan yang lebih kearah numerik dari pada kategori.

3. Prediksi

Prediksi memiliki kemiripan dengan estimasi dan klasifikasi. Hanya saja, prediksi hasilnya menunjukkan sesuatu yang belum terjadi (mungkin terjadi di masa depan).

4. Klasifikasi

Dalam klasifikasi variabel, tujuan bersifat kategorik. Misalnya, kita akan mengklasifikasikan pendapatan dalam tiga kelas, yaitu pendapatan tinggi, pendapatan sedang, dan pendapatan rendah.

5. *Clustering*

Clustering lebih ke arah pengelompokan *record*, pengamatan, atau kasus dalam kelas yang memiliki kemiripan.

6. Asosiasi

Mengidentifikasi hubungan antara berbagai peristiwa yang terjadi pada satu waktu.

3. Tahap-tahap Data Mining

Sebagai suatu rangkaian proses, data mining dapat dibagi menjadi beberapa tahapan. Tahap-tahap tersebut bersifat interaktif, pemakai terlibat langsung atau dengan perantaraan *knowledge base*.

Tahap-tahap *data mining* adalah sebagai berikut:

a. Pembersihan data (*data cleaning*)

Pembersihan data merupakan proses menghilangkan *noise* dan data yang tidak konsisten atau data tidak relevan.

b. Integrasi data (*data integration*)

Integrasi data merupakan penggabungan data dari berbagai *database* ke dalam satu *database* baru.

c. Seleksi data (*data selection*)

Data yang ada pada database sering kali tidak semuanya dipakai, oleh karena itu hanya data yang sesuai untuk dianalisis yang akan diambil dari *database*.

d. Transformasi data (*data transformation*)

Data diubah atau digabung ke dalam format yang sesuai untuk diproses dalam data mining.

e. Proses *mining*

Merupakan suatu proses utama saat metode diterapkan untuk menemukan pengetahuan berharga dan tersembunyi dari data.

f. Evaluasi pola (*pattern evaluation*)

Untuk mengidentifikasi pola-pola menarik ke dalam *knowledge based* yang ditemukan.

g. Presentasi pengetahuan (*knowledge presentation*)

Merupakan visualisasi dan penyajian pengetahuan mengenai metode yang digunakan untuk memperoleh pengetahuan yang diperoleh pengguna.

2.2.4. Prediksi

Prediksi adalah memperkirakan sesuatu yang akan terjadi pada masa yang mendatang. Prediksi didapatkan melalui metode ilmiah maupun hanya subjektif belaka. Prediksi juga dapat digunakan dalam pengklasifikasian, tidak hanya untuk memprediksi *time series*, karena sifatnya yang bisa menghasilkan *class* berdasarkan berbagai atribut yang kita sediakan (Susanto, 2015).

Berdasarkan teknik yang digunakan untuk memprediksi maka prediksi dapat dibagi menjadi dua bagian yaitu prediksi kualitatif dan prediksi kuantitatif. Prediksi Kualitatif didasarkan atas data kualitatif pada masa lalu. Metode *kualitatif* digunakan jika data masa lalu dari variabel yang akan diprediksi tidak ada, tidak cukup atau kurang dipercaya. Hasil prediksi yang dibuat sangat tergantung pada individu yang menyusunnya. Hal ini penting karena hasil prediksi tersebut ditentukan berdasarkan pemikiran yang bersifat *judgement* atau opini, pengetahuan dan pengalaman dari penyusunnya. Oleh karena itu metode kualitatif ini disebut juga *judgemental*, *subjective*, *intuitive*. Prediksi kuantitatif didasarkan atas data kuantitatif pada masa lalu. Hasil prediksi yang dibuat sangat tergantung pada metode yang dipergunakan dalam prediksi tersebut. Dengan metoda yang

berbeda akan diperoleh hasil prediksi yang berbeda. Hal yang perlu diperhatikan dari penggunaan metoda tersebut adalah baik tidaknya metoda yang digunakan dan sangat ditentukan dari penyimpangan antara hasil prediksi dengan kenyataan yang terjadi.

2.3 Algoritma *Naïve Bayes*

Algoritma *Naive Bayes* merupakan teknik prediksi berbasis probabilistik sederhana yang berdasar pada penerapan *Teorema Bayes* (atau aturan Bayes) dengan asumsi *independensi* (ketidaktergantungan) yang kuat. (Eko Prasetyo, 2012). Dalam Bayes (terutama *Naive Bayes*), maksud independensi yang kuat pada fitur adalah bahwa sebuah fitur pada sebuah data tidak berkaitan dengan ada atau tidaknya fitur lain dalam data yang sama. Kaitan antara *Naïve Bayes* dengan Klasifikasi, korelasi hipotesis, dan bukti dengan klasifikasi adalah bahwa hipotesis dalam teorema Bayes merupakan label kelas yang menjadi target pemetaan dalam klasifikasi, sedangkan bukti merupakan fitur-fitur yang menjadi masukan dalam model klasifikasi.

Teorema keputusan *bayes* adalah pendekatan statistik yang fundamental dalam pengenalan pola (*pattern recognition*). Pendekatan ini didasarkan pada kuantifikasi *tradeoff* antara berbagai keputusan klasifikasi dengan menggunakan probabilitas dan ongkos yang ditimbulkan dalam keputusan keputusan tersebut. Ide dasar dari *bayes* adalah menangani masalah yang bersifat hipotesis yakni mendesain suatu klasifikasi untuk memisahkan objek. Misalkan terdapat dua jenis objek dengan kemungkinan kemunculan random, selanjutnya ingin diprediksi objek apa yang akan lewat selanjutnya (Diana Laily Fithri, 2014).

Algoritma Naïve Bayes bertujuan untuk melakukan klasifikasi data pada kelas tertentu, kemudian pola tersebut dapat digunakan untuk memperkirakan data transaksi penjualan dalam memprediksi perilaku pola pembeli. Hasil yang dicapai pada penelitian ini mencapai akurasi 97% (Susanto, 2015). Definisi lain mengatakan Naive Bayes merupakan pengklasifikasian dengan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya (Bustami, 2014).

Naive Bayes didasarkan pada asumsi penyederhanaan bahwa nilai atribut secara kondisional saling bebas jika diberikan nilai output. Dengan kata lain, diberikan nilai output, probabilitas mengamati secara bersama adalah produk dari probabilitas individu (Mujib Ridwan, 2013). Keuntungan penggunaan Naive Bayes adalah bahwa metode ini hanya membutuhkan jumlah data pelatihan (Training Data) yang kecil untuk menentukan estimasi parameter yang diperlukan dalam proses pengklasifikasian.

Persamaan dari teorema *Bayes* adalah (Han & Kamber, 2006) :

$$P(Y|X) = \frac{P(X|Y) P(Y)}{P(X)} \dots\dots\dots(2.1)$$

Di mana :

X : Himpunan data training

Y : Hipotesis

$(Y|X)$: Probabilitas posterior, yaitu probabilitas bersyarat dari hipotesis Y berdasarkan kondisi X .

(Y) : Probabilitas posterior dari hipotesis Y , yaitu probabilitas bahwa hipotesis Y bernilai benar sebelum data X muncul.

(X) : Probabilitas dari data X

$(X|Y)$: Probabilitas posterior, yaitu probabilitas bersyarat dari hipotesis X berdasarkan kondisi Y , dan bisa disebut dengan likelihood ini mudah untuk dihitung ketika memberikan nilai 1 saat X dan Y konsisten, dan memberikan nilai 0 saat X dan Y tidak konsisten.

Untuk menjelaskan metode *Naive Bayes*, perlu diketahui bahwa proses klasifikasi memerlukan sejumlah petunjuk untuk menentukan kelas apa yang cocok bagi sampel yang dianalisis tersebut. Karena itu, metode *Naive Bayes* di atas disesuaikan sebagai berikut:

$$P(Y|X) = \frac{P(X_1|Y) P(X_2|Y) \dots P(X_n|Y) P(Y)}{P(X)} \dots\dots\dots(2.2)$$

Dimana :

X : Himpunan Data Training

Y : Hipotesis

$(Y|X)$: Probabilitas posterior, yaitu probabilitas bersyarat dari hipotesis Y berdasarkan kondisi X .

(Y) : Probabilitas posterior dari hipotesis Y , yaitu probabilitas bahwa hipotesis Y bernilai benar sebelum data X muncul.

$P(X)$: Probabilitas dari data X

$P(X_1|Y), P(X_2|Y), \dots, P(X_n|Y)$: Probabilitas dari X_1, X_2, \dots, X_n untuk hipotesis Y , biasa disebut dengan likelihood.

Karena $P(X)$ irrelevant, maka untuk mencari peluang hanya menggunakan rumus berikut ini :

$$P(Y|X) = P(X_1|Y)P(X_2|Y) \dots P(X_n|Y)P(Y) \quad \dots\dots\dots(2.3)$$

Jika ada $P(X_n|Y)$ yang memiliki nilai 0, maka $P(Y|X)=0$. Maka klasifikasi naïve Bayesian tidak bisa memprediksi record yang salah satu atributnya memiliki probabilitas bersyarat (likelihood) = 0, untuk mengatasi hal itu, dilakukan penambahan nilai 1 ke setiap evidence dalam perhitungan sehingga probabilitas tidak akan bernilai 0.

Langkah ini sering disebut laplace estimator dengan rumus sebagai berikut :

$$P(X_i|Y_j) = \frac{n_{c_i} + 1}{n + m} \quad \dots\dots\dots(2.4)$$

dimana :

n : total instances dari kelas Y_j

n_{c_i} : jumlah contoh training dari Y_j yang menerima nilai X_i

m : Parameter yang dikenal sebagai ukuran sampel ekuivalen.

Adapun cara kerja klasifikasi naïve bayes :

1. Baca *data training*
2. Hitung Jumlah dan probabilitas, namun apabila data numerik maka:

- a. Cari nilai *mean* dan standar deviasi dari masing-masing parameter yang merupakan data numerik.

Adapun persamaan yang digunakan untuk menghitung nilai rata – rata hitung (*mean*) dapat dilihat sebagai berikut :

$$\mu = \frac{\sum_{i=1}^n x_i}{n} \dots\dots\dots(2.5)$$

atau

$$\mu = \frac{x_1 + x_2 + x_3 + \dots + x_n}{n} \dots\dots\dots(2.6)$$

di mana :

μ : rata – rata hitung (*mean*)

x_i : nilai sample ke -*i*

n : jumlah sampel

Dan persamaan untuk menghitung nilai simpangan baku (standar deviasi) dapat dilihat sebagai berikut:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \mu)^2}{n - 1}} \dots\dots\dots(2.7)$$

di mana :

σ : standar deviasi

x_i : nilai x ke -*i*

μ : rata-rata hitung

n : jumlah sampel

b. Cari nilai probabilistik dengan cara menghitung jumlah data yang sesuai dari kategori yang sama dibagi dengan jumlah data pada kategori tersebut.

3. Mendapatkan nilai dalam tabel *mean*, standard deviasi dan probabilitas.

4. Solusi kemudian dihasilkan.

2.4 Penerapan Metode *Naïve Bayes*

Dalam tahap ini akan dijelaskan Perhitungan Metode *Naïve Bayes* secara manual:

Satu data uji yang memiliki ciri sebagai berikut:

IPS, Pria, memiliki alamat luar surakarta, asal sekolah luar surakarta, $SKS \leq 18$, $MK \leq 7$, tidak anggota asisten.

Apakah mahasiswa tersebut akan lulus tepat waktu atau terlambat? Penghitungan data test berdasarkan data training: Asumsi :

Y = Lama studi, X_1 = Jurusan SMA/SMK, X_2 = Gender, X_3 = Daerah / Alamat, X_4 = Asal sekolah, X_5 = Rata-rata SKS, X_6 = Rata-rata mata kuliah, X_7 = Asisten

Fakta menunjukkan :

$$P(Y = \text{TEPAT}) = 73/341 = 0,214076$$

$$P(Y = \text{TERLAMBAT}) = 268/341 = 0,785924$$

Fakta :

$$P(X_1 = \text{IPS} | Y = \text{TEPAT}) = 22/73 = 0,30137$$

$$P(X_1 = \text{IPS} | Y = \text{TERLAMBAT}) = 127/268 = 0,473881$$

$$P(X_2 = \text{PRIA} | Y = \text{TEPAT}) = 46/73 = 0,630137$$

$$P(X_2 = \text{PRIA} | Y = \text{TERLAMBAT}) = 165/268 = 0,615672$$

$$P(X_3 = \text{LUAR} | Y = \text{TEPAT}) = 14/73 = 0,191781$$

$$P(X_3 = \text{LUAR} | Y = \text{TERLAMBAT}) = 78/268 = 0,291045$$

$$P(X4=LUAR | Y=TEPAT) = 14/73 = 0,191781$$

$$P(X4=LUAR | Y=TERLAMBAT) = 70/268 = 0,261194$$

$$P(X5=SKS \leq 18 | Y=TEPAT) = 4/73 = 0,054795$$

$$P(X5=SKS \leq 18 | Y=TERLAMBAT) = 166/268 = 0,619403$$

$$P(X6=MK \leq 7 | Y=TEPAT) = 0/73 = 0$$

$$P(X6=MK \leq 7 | Y=TERLAMBAT) = 141/268 = 0,526119$$

$$P(X7=TIDAK | Y=TEPAT) = 54/73 = 0,739726$$

$$P(X7=TIDAK | Y=TERLAMBAT) = 248/268 = 0,925373$$

HMAP dari keadaan ini dapat dihitung dengan :

$$\begin{aligned}
 &P(X1=IPS, X2=PRIA, X3=LUAR, X4=LUAR, X5=SKS \leq 18, X6=MK \leq 7, \\
 &X7=TIDAK | Y=TEPAT) = 22/73 * 46/73 * 14/73 * 14/73 * 4/73 * 0/73 * 54/73 \\
 &* 73/341 = 0 \\
 &P(X1=IPS, X2=PRIA, X3=LUAR, X4=LUAR, X5=SKS \leq 18, \\
 &X6=MK \leq 7, X7=TIDAK | Y=TERLAMBAT) \\
 &= 127/268 * 165/268 * 78/268 * 70/268 * 166/268 * 141/268 * 248/268 * 268/341 = \\
 &0,005256, \text{ KEPUTUSAN LAMA STUDI} = \mathbf{TERLAMBAT}.
 \end{aligned}$$

2.5 Confusion Matrix

Confusion matrix memberikan keputusan yang diperoleh dalam training dan testing, *Confusion matrix* memberikan penilaian dalam performance klasifikasi berdasarkan objek dengan benar atau salah (Eko Prasetyo, 2012).

Tabel 2.1 Contoh confusion matrix

f_{ij}		Kelas hasil prediksi (j)	
		Kelas = 1	Kelas = 0
Kelas asli i	Kelas = 1	f_{11}	f_{10}
	Kelas = 0	f_{01}	f_{00}

Berdasarkan hasil pengisian confusion matrix, maka dapat diketahui jumlah data dari setiap kelas yang diklasifikasikan dengan benar ($f_{11} + f_{00}$) dan data yang diklasifikasikan dengan salah ($f_{01} + f_{10}$). Kuantitas dari confusion matrix dapat

disimpulkan dengan dua nilai, yakni akurasi dan laju error. Untuk menghitung akurasi dan laju error dapat digunakan persamaan berikut :

$$Accuracy = \frac{\text{Jumlah data yang diprediksi benar}}{\text{Jumlah prediksi yang dilakukan}}$$

Pada umumnya, model yang di bangun dapat memprediksi kelas dengan benar pada semua data yang menjadi data training (Eko Prasetyo, 2012).

2.6 Siklus Hidup Pengembangan Sistem

Pengembangan sistem informasi yang berbasis komputer dapat merupakan tugas kompleks yang membutuhkan banyak sumber daya dan dapat memakan waktu berbulan-bulan bahkan bertahun tahun untuk menyelesaikannya. Proses pengembangan sistem melewati beberapa tahapan dari mulai sistem itu direncanakan sampai dengan sistem tersebut diterapkan, dioperasikan dan dipelihara. Bila operasi sistem yang sudah dikembangkan masih timbul kembali permasalahan-permasalahan yang kritis serta tidak dapat diatasi dalam tahap pemeliharaan sistem, maka perlu dikembangkan kembali suatu sistem untuk mengatasinya dan proses ini kembali ke tahap yang pertama, yaitu tahap perencanaan sistem. Siklus ini disebut dengan siklus hidup suatu sistem (*systems life cycle*). Siklus hidup pengembangan sistem dengan langkah-langkah utamanya yang akan digunakan adalah sebagai berikut : (Jogiyanto HM, 2005 : 52).



Gambar 2.1. Siklus Hidup Pengembangan Sistem

2.6.1 Analisa Sistem

Jogiyanto HM (2005:129) mendefinisikan analisa sistem sebagai berikut: "Analisa sistem (*systems analysis*) sebagai penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya".

Tahap analisis merupakan tahap yang kritis dan sangat penting, karena kesalahan di dalam tahap ini akan menyebabkan juga kesalahan ditahap selanjutnya. Tahap analisa sistem mencakup studi kelayakan dan analisis kebutuhan.

Di dalam tahap analisis sistem terdapat langkah-langkah dasar yang harus dilakukan oleh analisis sistem, yaitu sebagai berikut :

1. *Identify*, yaitu mengidentifikasi masalah.

Mengidentifikasi (mengenal) masalah merupakan langkah pertama yang dilakukan dalam tahap analisis sistem. Masalah (*Problems*) dapat didefinisikan sebagai suatu pertanyaan yang diinginkan untuk dipecahkan.

2. *Understand*, yaitu memahami kerja dari sistem yang ada.

Langkah kedua dari tahap analisis sistem adalah memahami kerja dari sistem yang ada. Langkah ini dapat dilakukan dengan mempelajari secara terinci bagaimana sistem yang ada beroperasi. Untuk mempelajari operasi dari sistem ini diperlukan data yang dapat diperoleh dengan cara melakukan penelitian.

3. *Analyze*, yaitu menganalisis sistem tanpa *report*

Langkah ini dilakukan berdasarkan data yang telah diperoleh dari hasil penelitian yang telah dilakukan

4. *Report*, yaitu membuat laporan hasil analisis

Tujuan utama dari pembuatan laporan hasil analisis :

- a. Pelaporan bahwa analisis telah selesai dilakukan
- b. Meluruskan kesalahan-pengertian mengenai apa yang telah ditemukan dan dianalisis oleh analisis sistem tetapi tidak sesuai menurut manajemen.

2.6.2 Desain Sistem

Setelah tahap analisis sistem selesai dilakukan, maka analisis sistem telah mendapatkan gambaran dengan jelas apa yang harus dikerjakan. Tiba waktunya sekarang bagi analisis sistem untuk memikirkan bagaimana membentuk sistem tersebut. Tahap ini disebut dengan desain sistem (*system design*).

Menurut Robert J. Verzello dan John Reuter, dalam Jogiyanto HM (2005 : 196) desain sistem adalah tahap setelah analisis dari siklus pengembangan sistem; pendefinisian dari kebutuhan-kebutuhan fungsional dan persiapan untuk rancang bangun implementasi menggambarkan bagaimana suatu sistem dibentuk.

Demikian pula Menurut John Burch dan Gary Grudnitski, dalam Jogiyanto HM (2005 : 196) desain sistem dapat didefinisikan sebagai penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi.

Tahap desain sistem mempunyai dua tujuan utama yaitu :

- a. Untuk memenuhi kebutuhan kepada pemakai sistem.
- b. Untuk memberikan gambaran yang jelas dan rancang bangun yang lengkap kepada pemrogram komputer dan ahli-ahli teknis lainnya.

Desain sistem dapat dibagi dalam dua bagian yaitu desain sistem secara umum (*general systems design*) dan desain sistem secara terinci (*detailed system design*).

1. Desain sistem secara umum (*General System Design*).

Tujuan dari desain sistem secara umum adalah untuk memberikan gambaran secara umum kepada *user* tentang sistem yang baru, yang mana merupakan persiapan dari desain sistem secara rinci. Desain secara umum dilakukan oleh analis sistem untuk mengidentifikasi komponen-komponen sistem informasi yang akan didesain secara rinci oleh pemrogram komputer dan ahli teknik lainnya.

Pada tahap ini, komponen-komponen sistem informasi dirancang dengan tujuan untuk dikomunikasikan kepada *user*. Komponen sistem informasi yang didesain adalah model, *input*, *database*, *output*, teknologi dan kontrol.

2. Desain sistem secara rinci (*detailed system design*).

a. Desain *input* terinci

Masukan merupakan awal dimulainya proses informasi. Bahan mentah dari informasi adalah data yang terjadi dari transaksi-transaksi yang dilakukan oleh organisasi. Data hasil transaksi merupakan masukan untuk sistem informasi. Hasil dari sistem informasi tidak lepas dari data yang dimasukkan. Desain *input* terinci dimulai dari desain dokumen dasar sebagai penangkap *input* yang pertama kali. Jika dokumen dasar tidak didesain dengan baik, kemungkinan *input* yang tercatat dapat salah bahkan kurang.

Fungsi dokumen dasar dalam penanganan arus data :

- 1) Dapat menunjukkan macam dari data yang harus dikumpulkan.
- 2) Data dapat dicatat dengan jelas, konsisten dan akurat.
- 3) Dapat mendorong lengkapnya data disebabkan data yang dibutuhkan disebutkan satu persatu di dalam dokumen dasarnya.

b. Desain *output* terinci.

Desain *output* terinci dimaksudkan untuk mengetahui bagaimana dan seperti apa bentuk *output-output* dari sistem yang baru. Desain *Output* Terinci terbagi atas dua yaitu desain *output* berbentuk laporan dimedia kertas dan desain *output* dalam bentuk dialog pada layar terminal.

a) Desain *Output* Dalam Bentuk laporan

Desain ini dimaksudkan untuk menghasilkan *output* dalam bentuk laporan di media kertas. Bentuk laporan yang paling banyak digunakan adalah dalam bentuk tabel dan berbentuk grafik atau bagan.

b) Desain *Output* Dalam Bentuk Dialog Layar Terminal

Desain ini merupakan rancang bangun dari percakapan antara pemakai sistem (*user*) dengan komputer. Percakapan ini dapat terdiri dari proses memasukkan data ke sistem, menampilkan *output* informasi kepada *user* atau keduanya.

Beberapa strategi dalam membuat layar dialog terminal :

1. Dialog pertanyaan / jawaban.
2. Menu.

Menu banyak digunakan karena merupakan jalur pemakai yang mudah dipahami dan mudah digunakan. Menu berisi beberapa alternatif atau option atau pilihan yang disajikan kepada *user*. Pilihan menu akan lebih baik bila dikelompokkan sesuai fungsinya.

c. Desain *database* terinci.

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di sistem komputer dan digunakan perangkat lunak tertentu untuk memanipulasinya. *Database* merupakan salah satu komponen yang penting di sistem informasi karena berfungsi sebagai

basis penyedia informasi bagi para pemakainya. Penerapan *database* dalam sistem informasi disebut *database system*.

Sistem basis data (*database system*) adalah suatu sistem informasi yang mengintegrasikan kumpulan dari data yang saling berhubungan satu dengan yang lainnya dan membuatnya tersedia untuk beberapa aplikasi yang bermacam-macam di dalam suatu organisasi. Dalam sistem basis data, tiap-tiap orang atau bagian dapat memandang *database* dari beberapa sudut pandang yang berbeda.

Pada tahap ini, desain *database* dimaksudkan untuk mendefinisikan isi atau struktur dari tiap-tiap file yang telah diidentifikasi dan didesain secara umum.

d. Desain teknologi.

Tahap desain teknologi terbagi atas dua yaitu desain teknologi secara umum dan terinci. Pada tahap ini kita menentukan teknologi yang akan dipergunakan dalam menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi yang dimaksud meliputi :

1. Perangkat Keras (*hardware*), yang terdiri dari alat masukan, alat pemroses, alat *output* dan simpanan luar.
2. Perangkat Lunak (*software*), yang terdiri dari perangkat lunak sistem operasi (*operating system*), perangkat lunak bahasa (*language software*) dan perangkat lunak (*application software*).
3. Sumber Daya Manusia (*brainware*), misalnya operator komputer, pemrogram, spesialis telekomunikasi, sistem analis dan sebagainya.

Desain teknologi sangat diperlukan pada tahap implementasi dan pengujian untuk membuktikan bahwa sistem dapat berjalan secara semestinya.

e. Desain model.

Tahap desain model terbagi menjadi dua yaitu desain model secara umum dan terinci. Tahap desain model secara umum berupa desain sistem secara fisik dan logika. Desain fisik dapat digambarkan dengan bagan alir dokumen.

Desain secara logika digambarkan dengan diagram arus data (DAD). Pada tahap desain model terinci, model akan mendefinisikan secara rinci urutan-urutan langkah dari masing-masing proses yang digambarkan di DAD. Urutan langkah proses ini diwakili oleh suatu program komputer.

2.6.3 Perancangan Konseptual

Perancangan konseptual sering kali disebut dengan perancangan logis. Pada perancangan ini kebutuhan pemakai dan pemecahan masalah yang teridentifikasi selama tahap analisis sistem mulai dibuat untuk di implementasikan. Ada tiga langkah penting yang dilakukan dalam perancangan konseptual, yaitu evaluasi alternatif rancangan, penyiapan spesifikasi rancangan dan penyiapan laporan rancangan sistem secara konseptual.

Menurut Romney, Seinbart dan Cushing, 1997 dalam Abdul Kadir (2003 : 407) evaluasi yang dilakukan mengandung hal-hal berikut :

- 1) Bagaimana alternatif-alternatif tersebut memenuhi sasaran sistem dan organisasi dengan baik?.

- 2) Bagaimana alternatif-alternatif tersebut memenuhi kebutuhan pemakai dengan baik?.
- 3) Apakah alternatif-alternatif tersebut layak secara ekonomi?.
- 4) Apa saja keuntungan dan kerugian masing-masing?

Setelah alternatif rancangan dipilih, tahap selanjutnya adalah penyiapan spesifikasi rancangan yang elemen-elemen sebagai berikut :

a. Keluaran

Rancangan laporan mencakup frekuensi laporan (harian, mingguan, dsb), isi laporan, bentuk laporan dan laporan cukup ditampilkan pada layar atau perlu dicetak.

b. Penyimpan Data

Dalam hal ini, semua data yang diperlukan untuk membentuk laporan ditentukan lebih detail, termasuk ukuran data dan letaknya dalam berkas.

c. Masukan

Rancangan masukan meliputi data yang perlu dimasukkan kedalam sistem.

d. Prosedur Pemrosesan dan Operasi

Rancangan ini menjelaskan bagaimana data masukan diproses dan disimpan dalam rangka untuk menghasilkan laporan.

Langkah berikutnya adalah menyiapkan laporan rancangan sistem konseptual. Berdasarkan laporan inilah, perancangan sistem secara fisik dibuat.

2.6.4 Perancangan Fisik

Pada perancangan ini, rancangan yang masih bersifat konsep diterjemahkan dalam bentuk fisik sehingga terbentuk spesifikasi lengkap tentang modul sistem dan antarmuka antar modul serta rancangan basis data secara fisik.

Beberapa hasil akhir setelah tahap perancangan fisik berakhir :

1. Rancangan keluaran

Rancangan keluaran berupa bentuk laporan dan rancangan dokumen.

2. Rancangan masukan

Rancangan masukan berupa rancangan layar untuk pemasukan data.

3. Rancangan antarmuka pemakai dan sistem.

Rancangan ini berupa rancangan interaksi antar pemakai dan sistem, misalnya berupa menu, icon dan lain-lain.

4. Rancangan *platform*.

Rancangan ini berupa rancangan yang menentukan *hardware* dan *software* yang akan digunakan.

5. Rancangan basis data.

Rancangan ini berupa rancangan-rancangan berkas dalam basis data termasuk penentuan kapasitas masing-masing.

6. Rancangan modul.

Rancangan ini berupa rancangan program yang dilengkapi dengan algoritma (cara modul / program kerja).

7. Rancangan kontrol.

Rancangan ini berupa rancangan kontrol-kontrol yang digunakan dalam sistem seperti validasi, otorisasi dan audit data.

8. Dokumentasi.

Berupa hasil dokumentasi hingga tahap perancangan fisik.

9. Rencana pengujian.




Berupa rencana yang dipakai untuk menguji sistem.




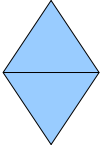

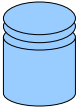

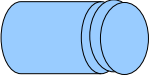
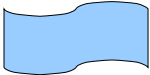
10. Rencana konversi.

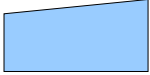
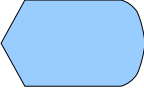
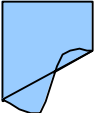

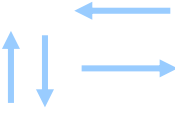

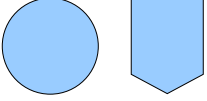
Berupa rencana untuk menerapkan sistem baru terhadap sistem lama.

Bagan Alir sistem merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Bagan alir sistem digambarkan dengan simbol-simbol sebagai berikut :

Tabel 2.2 Bagan Alir Sistem

NO	NAMA SIMBOL	SIMBOL	KETERANGAN
1.	Simbol Dokumen		Menunjukkan dokumen <i>input</i> dan <i>output</i> baik itu proses manual, mekanik, atau komputer
2.	Simbol Kegiatan Manual		Menunjukan pekerjaan manual
3.	Simbol Simpanan Offline		Menunjukkan file non-komputer yang diarsip urut angka (<i>numerical</i>), huruf (<i>alphabetical</i>), atau tanggal (<i>chronological</i>)

4.	Simbol Kartu Plong		Menunjukkan <i>input</i> dan <i>output</i> yang menggunakan kartu plong (<i>punched card</i>).
5.	Simbol Proses		Menunjukkan kegiatan proses dari operasi program computer
6	Simbol Operasi Luar		Menunjukkan operasi yang dilakukan di luar proses operasi komputer
7.	Simbol Pengurutan Offline		Menunjukkan proses urut data di luar proses komputer. operasi luar, menunjukkan operasi yang dilakukan di luar proses operasi komputer
8.	Simbol Pita Magnetik		Menunjukkan <i>input</i> dan <i>output</i> menggunakan pita <i>magnetic</i> .
9.	Simbol Hard Disk		Menunjukkan <i>input</i> dan <i>output</i> menggunakan <i>harddisk</i>
10.	Simbol Diskette		Menunjukkan <i>input</i> dan <i>output</i> menggunakan <i>diskette</i>
11.	Simbol Drum Magnetik		Menunjukkan <i>input</i> dan <i>output</i> menggunakan drum magnetic
12.	Simbol Pita Kertas Berlubang		Menunjukkan <i>input</i> dan <i>output</i> menggunakan pita kertas berlubang.

13.	Simbol Keyboard		Menunjukkan <i>input</i> yang menggunakan <i>on-line keyboard</i>
14.	Simbol Display		Menunjukkan <i>output</i> yang ditampilkan di monitor.
15.	Simbol Pita Kontrol		Menunjukkan penggunaan pita kontrol (<i>control tape</i>) dalam <i>batch control</i> total untuk pencocokan di proses <i>batch processing</i> .
16	Simbol Hubungan Komunikasi		Menunjukkan proses transmisi data melalui <i>channel</i> komunikasi.
17.	Simbol Garis Alir		Menunjukkan arus dari proses
18.	Simbol Penjelasan		Menunjukkan penjelasan dari suatu proses
19.	Simbol Penghubung		Menunjukkan penghubung ke halaman yang masih sama atau ke halaman yang lain

Sumber : (Jogiyanto HM, 2005 : 796-799)

Untuk mempermudah penggambaran suatu sistem yang ada atau sistem yang baru yang akan dikembangkan secara logika tanpa memperhatikan lingkungan fisik di mana data tersebut mengalir atau lingkungan fisik di mana data tersebut akan disimpan, maka digunakan Diagram Arus Data (DAD) atau *Data Flow Diagram* (DFD). Dalam menggambarkan sistem perlu dilakukan pembentukan simbol, berikut ini simbol-simbol yang sering digunakan dalam DAD :

1. *External entity* (kesatuan luar) atau *boundary* (batas sistem).

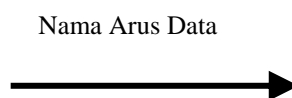
Setiap sistem pasti mempunyai batas sistem (*boundary*) yang memisahkan suatu sistem dengan lingkungan luarnya. Sistem akan menerima *input* dan menghasilkan *output* kepada lingkungan luarnya. Kesatuan luar (*external entity*) merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lain yang berada di lingkungan luarnya yang akan memberikan *input* serta menerima *output* dari sistem. (Jogiyanto HM, 2005 ;701)



Gambar 2.2 Contoh Notasi kesatuan luar

2. *Data flow* (arus data)

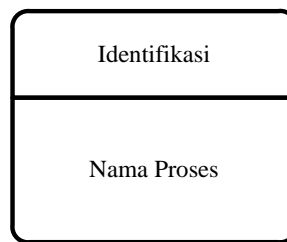
Arus data ini menunjukkan arus atau aliran data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem. (Jogiyanto HM, 2005 ;702)



Gambar 2.3 Contoh Notasi arus data

3. *Process* (proses)

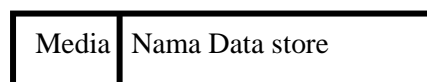
Suatu proses adalah kegiatan atau kerja yang dilakukan orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses. (Jogiyanto HM, 2005 ;705)



Gambar 2.4 Contoh Notasi proses

4. *Data store* (simpanan data).

Simpanan data pada DFD dapat disimbolkan dengan sepasang garis horizontal paralel yang tertutup disalah satu ujungnya. (Jogiyanto HM, 2005 ;707)



Gambar 2.5 Contoh Notasi simpanan data

2.7 Implementasi Sistem

Sistem telah dianalisa dan didesain secara rinci dan teknologi telah diseleksi dan dipilih. Tiba saatnya sekarang sistem untuk diimplementasikan (diterapkan). Tahap implementasi sistem merupakan tahap meletakkan sistem supaya siap untuk dioperasikan. Tahap implementasi sistem dapat terdiri dari langkah-langkah sebagai berikut:

1. Menerapkan rencana implementasi

Rencana implementasi merupakan kegiatan awal dari tahap implementasi sistem. Rencana implementasi dimaksudkan terutama untuk mengatur biaya dan waktu yang dibutuhkan selama tahap implementasi.

2. Melakukan Kegiatan Implementasi

Kegiatan implementasi dilakukan dengan dasar kegiatan yang telah direncanakan dalam rencana implementasi. Kegiatan-kegiatan yang dapat dilakukan dalam tahap implementasi ini adalah sebagai berikut :

a. Persiapan tempat dan instalasi perangkat keras dan perangkat lunak

Jika peralatan baru akan dimiliki, maka tempat atau ruangan untuk peralatan ini perlu dipersiapkan terlebih dahulu. Keamanan fisik dari tempat ini perlu juga dipertimbangkan. Sistem komputer yang besar membutuhkan tempat dengan lingkungan yang lebih harus diperhitungkan. Langkah selanjutnya setelah persiapan fisik tempat adalah menginstalasi perangkat keras yang sudah dikirim dan menginstalasi perangkat lunak yang sudah ada.

b. Pemrograman dan pengetesan sistem

Pemrograman merupakan kegiatan menulis kode program yang akan dieksekusi oleh komputer. Kode program yang ditulis oleh pemrogram harus berdasarkan dokumentasi yang disediakan oleh analis sistem hasil dari desain sistem secara rinci. Sebelum program diterapkan, maka program harus terlebih dahulu bebas dari kesalahan-kesalahan. Oleh sebab itu, program harus diuji untuk menemukan kesalahan-kesalahan yang mungkin dapat terjadi. Program

dites untuk tiap-tiap modul dan dilanjutkan dengan pengetesan untuk semua modul yang telah dirangkai.

c. Pengetesan sistem.

Pengetesan sistem biasanya dilakukan setelah pengetesan program. Pengetesan sistem dilakukan untuk memeriksa kekompakan antar komponen sistem yang diimplementasikan. Tujuan utama dari pengetesan sistem ini adalah untuk memastikan bahwa elemen-elemen atau komponen-komponen dari sistem telah berfungsi sesuai dengan yang diharapkan.

2.8 Teknik Pengujian Sistem

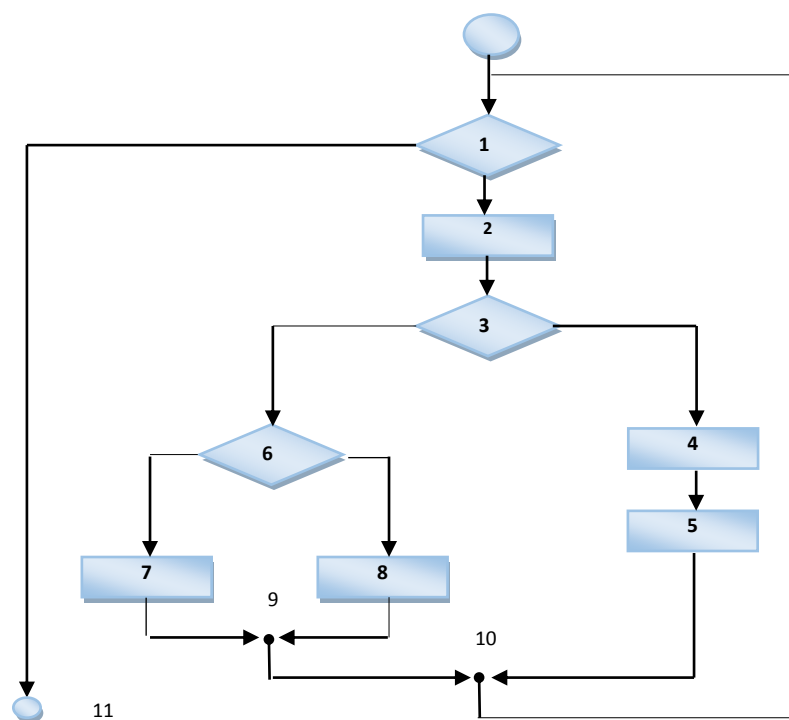
2.8.1 White Box

Pengujian perangkat lunak adalah elemen kritis dari jaminan kualitas perangkat lunak dan merepresentasikan kajian pokok dari spesifikasi, desain dan pengkodean.

Pengujian sistem / perangkat lunak memiliki sejumlah aturan yang berfungsi sebagai sasaran pengujian, diantaranya adalah sebagai berikut :

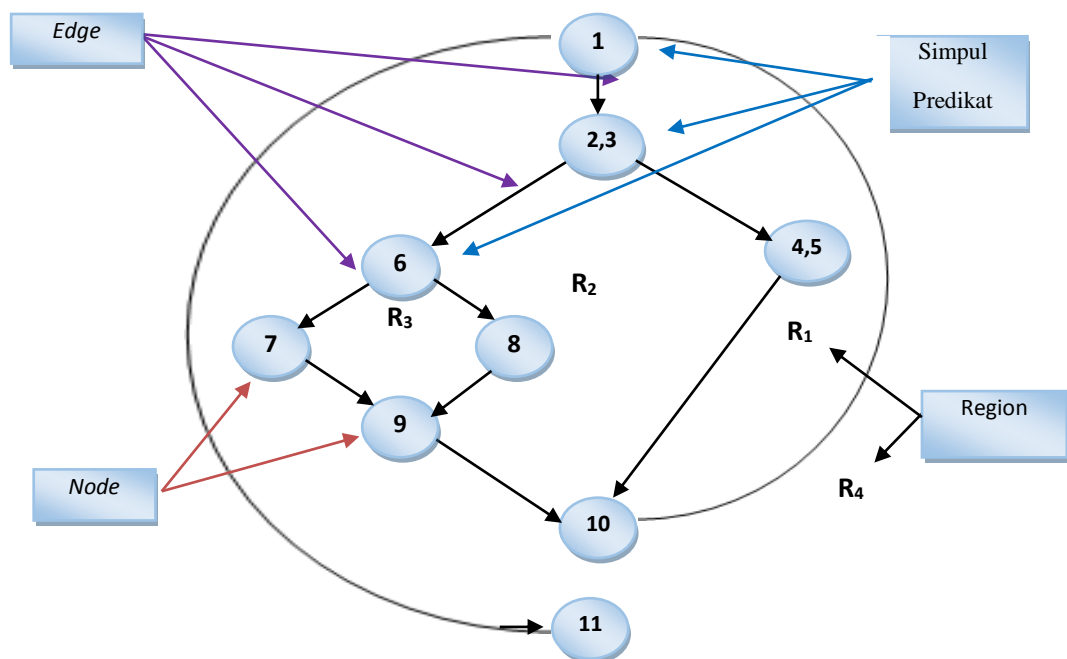
1. Pengujian adalah proses eksekusi suatu program dengan maksud menemukan kesalahan.
2. *Test case* yang baik adalah *test case* yang memiliki probabilitas tinggi untuk menemukan kesalahan yang belum pernah ditemukan sebelumnya.
3. Pengujian yang sukses adalah pengujian yang mengungkap semua kesalahan yang belum pernah ditemukan sebelumnya.

Pengujian *White Box*, adalah metode pengujian yang menggunakan struktur kontrol desain prosedur untuk memperoleh *test case*. Dengan menggunakan metode *white box*, perekayasa sistem dapat melakukan *test case* yang memberikan jaminan bahwa semua jalur independen pada suatu modul telah digunakan paling tidak satu kali, menggunakan semua keputusan logis pada sisi *true* dan *false*, mengeksekusi semua *loop* pada batasan mereka dan pada batas operasional mereka, dan menggunakan stuktur data internal untuk menjamin validitasnya. Pengujian *Basis Path* adalah teknik pengujian *white box* yang diusulkan pertama kali oleh Tom McCabe. Metode *basis path* ini memungkinkan desainer *test case* mengukur kompleksitas logis dari desain prosedural dan menggunakannya sebagai pedoman untuk menetapkan basis set dari jalur eksekusi. (Roger S. Pressman, 2002 : 536).



Gambar 2.6 Bagan alir

Bagan alir digunakan untuk menggambarkan struktur kontrol program. Dan untuk menggambarkan grafik alir, harus memperhatikan representasi desain prosedural pada bagan alir. Pada gambar dibawah ini, grafik alir memetakan bagan alir tersebut ke dalam grafik alir yang sesuai (dengan mengasumsikan bahwa tidak ada kondisi senyawa yang diisikan di dalam diamond keputusan dari bagan alir tersebut). Masing-masing lingkaran, yang disebut *simpul* grafik alir, merepresentasikan satu atau lebih statemen prosedural. Urutan kotak proses dan permata keputusan dapat memetakan simpul tunggal. Anak panah tersebut yang disebut *edges* atau *links*, merepresentasikan aliran kontrol dan analog dengan anak panah bagan alir. *Edge* harus berhenti pada suatu simpul, meskipun bila simpul tersebut tidak merepresentasikan statemen prosedural. (Roger S. Pressman, 2002 : 536).



Gambar 2.7 Grafik Alir

- *Node* adalah lingkaran yang merepresentasikan satu atau lebih statemen prosedural.
- *Edge* adalah anak panah pada grafik alir.
- *Region* adalah area yang membatasi *edge* dan *node*
- Simpul Predikat adalah simpul atau *node* yang berisi kondisi yang ditandai dengan dua atau lebih *edge* yang berasal darinya.

Dari gambar *flowgraph* di atas didapat :

Path 1 = 1– 11

Path 2 = 1– 2 – 3 – 4 – 5 – 10– 1–11

Path 3 = 1– 2 – 3 – 6 – 8 – 9 – 10– 1 – 11

Path 4 = 1– 2 – 3 – 6 – 7 – 9–10–1–11

Path 1,2,3,4 yang telah didefinisikan diatas merupakan *basis set* untuk diagram alir.

Cyclomatic complexity digunakan untuk mencari jumlah *path* dalam satu *flowgraph*. Dapat dipergunakan rumusan sebagaiberikut :

1. Jumlah region grafik alir sesuai dengan *cyclomatic complexity*.
2. *Cyclomatic complexity* $V(G)$ untuk grafik alir dihitung dengan rumus:

$$V(G) = E - N + 2 \quad \dots\dots\dots (2.1)$$

Dimana:

E= jumlah *edge* pada grafik alir

N = jumlah *node* pada grafik alir

Jalur 3 : 1 – 2 – 3 – 6 – 8 – 9 – 10 – 1 – 11

Jalur 4 : 1 – 2 – 3 – 6 – 7 – 9 – 10 – 1 – 11

Jalur 1, 2, 3, dan 4 yang ditentukan di atas terdiri dari sebuah basis set untuk grafik alir pada gambar 2.9 Bagaimana kita tahu banyaknya jalur yang dicari? Komputasi kompleksitas siklomatis memberikan jawaban. Fondasi kompleksitas siklomatis adalah teori grafik, dan memberi kita metrik perangkat lunak yang sangat berguna. Kompleksitas dihitung dalam salah satu dari tiga cara berikut :

1. Jumlah region grafik alir sesuai dengan kompleksitas siklomatis.
2. Kompleksitas siklomatis, $V(G)$, untuk grafik alir G ditentukan sebagai $V(G) = E - N + 2$ di mana E adalah jumlah *edge* grafik alir dan N adalah jumlah simpul grafik alir.
3. Kompleksitas siklomatis, $V(G)$, untuk grafik alir G juga ditentukan sebagai $V(G) = P + 1$, dimana P adalah jumlah simpul predikat yang diisikan dalam grafik alir G .

Pada gambar 2.8 grafik alir, kompleksitas siklomatis dapat dihitung dengan menggunakan masing-masing dari algoritma yang ditulis di atas :

1. Grafik alir mempunyai 4 region
2. $V(G) = 11 \text{ edge} - 9 \text{ simpul} + 2 = 4$
3. $V(G) = 3 \text{ simpul yang diperkirakan} + 1 = 4$

Dengan demikian, kompleksitas siklomatis dari grafik alir pada gambar 2.9 adalah 4. Yang lebih penting, nilai untuk $V(G)$ memberi kita batas atas untuk

jumlah jalur independen yang membentuk basis set, dan implikasinya, batas atas jumlah pengujian yang harus didesain dan dieksekusi untuk menjamin semua statemen program.

2.8.2 Black Box

Pengujian *Black-Box* berusaha menemukan kesalahan dalam kategori :

- pada struktur data (pengaksesan basis data)
- Kesalahan Fungsi tidak benar atau hilang
- Kesalahan antar muka
- Kesalahan inisialisasi dan akhir program
- Kesalahan performasi.

Pengujian ini berfokus pada persyaratan fungsional perangkat lunak dan merupakan komplemen dari pengujian *White-Box*. Hal tersebut dapat dicapai melalui :

- a. Pengujian *Graph-based* : dimulai dengan membuat grafik sekumpulan *node* yang mempresentasikan objek (misal *New File*, Layar baru dengan atributnya), link (hubungan antar objek), *node-weight* (misal nilai data tertentu seperti atribut layar, perilaku), dan link-weight (karakteristik suatu link, misal menu select).
- b. *Equivalence Partitioning* : membagi domain *input* untuk pengujian agar diperoleh kelas-kelas kesalahan (misal kelompok data karakter, atau atribut yang lain).
- c. Analisis Nilai Batas : pengujian berdasarkan nilai batas domain *input*.

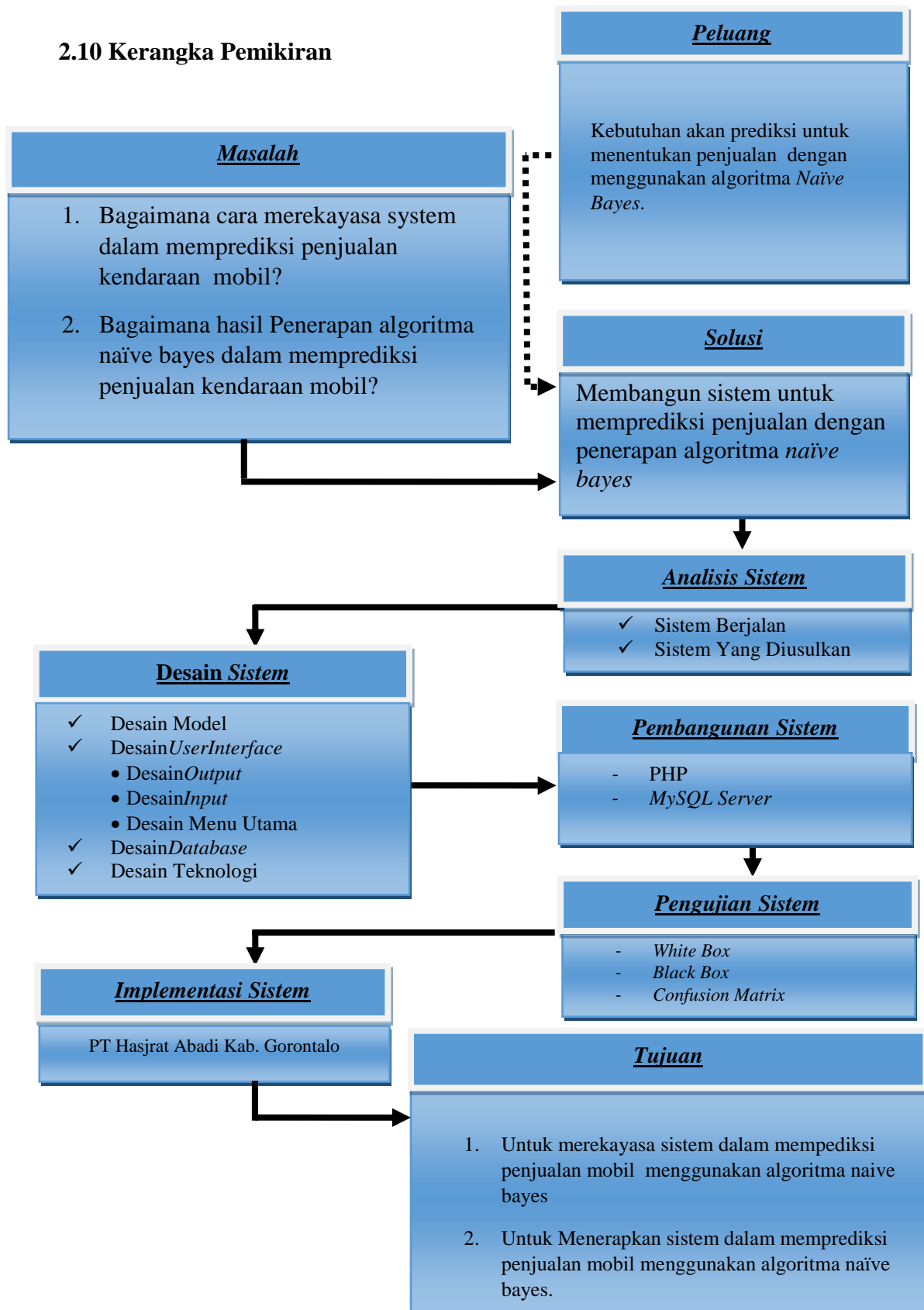
- d. Pengujian Perbandingan : disebut juga pengujian *back-to-back* yang diterapkan pada pada suatu versi perangkat lunak atau perangkat lunak redundan untuk memastikan konsistensinya.

2.9 Tools Pendukung

Tabel 2.3 Tools Pendukung

No	Nama Tools	Keterangan
1	PHP	Bahasa pemograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah situs web dan bisa digunakan bersamaan dengan HTML.
2	Mysql Server	software sistem manajemen basis data <i>SQL</i> atau <i>DBMS</i> yang <i>multi thread</i> dan <i>multi user</i> .

2.10 Kerangka Pemikiran



Gambar 2.8 Kerangka Pemikiran