

BAB II

LANDASAN TEORI

2.1 Sistem

Istilah sistem secara umum dapat didefinisikan sebagai kumpulan hal atau elemen yang saling bekerja sama atau yang dihubungkan dengan cara-cara tertentu sehingga membentuk satu kesatuan untuk melaksanakan suatu fungsi guna mencapai suatu tujuan. Sistem mempunyai karakteristik atau sifat tertentu, yaitu: Komponen Sistem, Batasan Sistem, Lingkungan Luar Sistem, Penghubung Sistem, Masukan Sistem, Keluaran Sistem, Pengolahan Sistem dan Sasaran Sistem (Edhy, 2003).

2.2 Aplikasi

Menurut Jogyanto (2010), teknologi yang canggih dari perangkat keras akan berfungsi bila instruksi-instruksi tertentu telah diberikan kepadanya. Instruksi-instruksi tersebut disebut dengan perangkat lunak (*software*). Perangkat lunak dapat diklasifikasikan ke dalam dua bagian besar, yaitu sebagai berikut :

1. Perangkat lunak sistem (*system software*), yaitu perangkat lunak yang mengoperasikan sistem komputernya. Perangkat lunak sistem dapat dikelompokkan lagi menjadi empat bagian yakni perangkat lunak sistem operasi (*operating system*), perangkat lunak sistem bantuan (*utility*), perangkat lunak bahasa (*language software*).
2. Perangkat lunak aplikasi (*application software*), yaitu program yang ditulis dan diterjemahkan oleh *language software* untuk menyelesaikan suatu aplikasi.

Berdasarkan uraian di atas, aplikasi peramalan permintaan merupakan sebuah program yang ditulis dan diterjemahkan oleh *language software* untuk menyelesaikan sebuah sistem peramalan, atau untuk melakukan sistem peramalan tertentu.

2.3 Permintaan

Menurut Gilarso (2007), dalam ilmu ekonomi istilah permintaan (*demand*) mempunyai arti tertentu, yaitu selalu menunjuk pada suatu hubungan tertentu antara jumlah suatu barang yang akan dibeli orang dan harga barang tersebut. Permintaan adalah jumlah dari suatu barang yang mau dan mampu dibeli pada berbagai kemungkinan harga, selama jangka waktu tertentu, dengan anggapan hal-hal lain tetap sama.

2.4 Faktor yang Mempengaruhi Permintaan

Menurut Sadono (2008), ada beberapa faktor yang mempengaruhi permintaan akan suatu barang selain harga barang itu sendiri. Faktor-faktor tersebut antara lain:

1. Harga barang-barang lain

Hubungan antara suatu barang dengan berbagai jenis-jenis barang lainnya dapat dibedakan kepada tiga golongan, yaitu:

a. Barang lain itu merupakan pengganti.

Sesuatu barang dinamakan barang pengganti kepada barang lain apabila ia dapat menggantikan fungsi barang lain tersebut. Sekiranya harga barang pengganti bertambah murah maka barang yang digantikannya akan mengalami pengurangan dalam permintaan.

b. Barang lain itu merupakan pelengkap.

Apabila sesuatu barang selalu digunakan bersama dengan barang lainnya, maka barang tersebut dinamakan barang pelengkap kepada barang lain tersebut. Kenaikan atau penurunan permintaan terhadap barang pelengkap selalu sejalan dengan perubahan permintaan barang yang digenapinya.

c. Barang yang tidak mempunyai kaitan sama sekali (barang netral).

Apabila dua macam barang tidak mempunyai hubungan yang rapat maka perubahan terhadap permintaan salah satu barang tersebut tidak akan mempengaruhi permintaan barang lainnya. Barang seperti ini dinamakan barang netral.

2. Perubahan pendapatan

Perubahan pendapatan pembeli selalu menimbulkan perubahan terhadap permintaan berbagai jenis barang.

3. Distribusi pendapatan

Distribusi pendapatan juga dapat mempengaruhi corak permintaan terhadap berbagai jenis barang. Sejumlah pendapatan masyarakat yang tertentu besarnya akan menimbulkan corak permintaan masyarakat yang berbeda apabila pendapatan tersebut diubah corak distribusinya.

4. Cita Rasa Masyarakat

Cita rasa mempunyai pengaruh yang cukup besar terhadap keinginan masyarakat untuk membeli barang-barang.

5. Jumlah Penduduk

Pertambahan penduduk tidak dengan sendirinya menyebabkan pertambahan permintaan tetapi biasanya pertambahan penduduk diikuti oleh perkembangan

dalam kesempatan kerja. Dengan demikian lebih banyak orang yang menerima pendapatan dan ini menambah daya beli dalam masyarakat. Pertambahan daya beli ini akan menambah permintaan.

6. Ekspektasi Tentang Masa Depan

Ramalan para konsumen bahwa harga-harga akan menjadi bertambah tinggi pada masa depan akan mendorong mereka untuk membeli lebih banyak pada masa kini, untuk menghemat pengeluaran pada masa yang akan datang. Sebaliknya, ramalan bahwa lowongan kerja akan bertambah sukar diperoleh dan kegiatan ekonomi akan mengalami resesi, akan mendorong orang lebih berhemat dalam pengeluarannya dan mengurangi permintaan.

2.5 Persediaan

Persediaan merupakan kebutuhan utama bagi suatu perusahaan, persediaan merupakan aset yang selalu dalam keadaan berputar dan secara terus menerus mengalami perubahan. Menurut Sartono (2010), persediaan pada umumnya merupakan salah satu jenis aktiva lancar yang jumlahnya cukup besar dalam suatu perusahaan. Hal ini mudah dipahami karena persediaan merupakan faktor penting dalam menentukan kelancaran operasi perusahaan. Ditinjau dari segi neraca persediaan adalah barang-barang atau bahan yang masih tersisa pada tanggal neraca, atau barang-barang yang akan segera dijual, digunakan atau diproses dalam periode normal perusahaan.

Menurut Kasmir (2008), persediaan merupakan sejumlah barang yang disimpan oleh perusahaan dalam suatu tempat (gudang). Persediaan merupakan cadangan perusahaan untuk proses produksi atau penjualan pada saat dibutuhkan.

Menurut Alexandri (2009), suatu aktiva yang meliputi barang-barang milik perusahaan dengan maksud untuk dijual dalam suatu periode usaha tertentu atau persediaan barang-barang yang masih dalam pengerjaan atau proses produksi ataupun persediaan bahan baku yang menunggu penggunaannya dalam proses produksi.

2.5.1 Fungsi Persediaan

Menurut Herjanto (2008), beberapa fungsi penting yang terdapat pada persediaan dalam memenuhi kebutuhan perusahaan, yaitu :

1. Memberikan pelayanan kepada pelanggan dengan tersedianya barang yang diperlukan.
2. Menghilangkan risiko keterlambatan pengiriman bahan baku atau barang yang dibutuhkan perusahaan.
3. Menghilangkan risiko jika material yang dipesan tidak baik sehingga harus dikembalikan.

2.5.2 Jenis-Jenis Persediaan

Menurut Sartono (2010), menerangkan bahwa jenis persediaan yang ada dalam perusahaan akan tergantung pada jenis perusahaan yaitu:

1. Perusahaan Jasa persediaan yang biasanya timbul seperti persediaan bahan pembantu atau persediaan habis pakai, yang termasuk didalamnya adalah kertas, karton, stempel, tinta, buku kwitansi, materai.
2. Perusahaan Manufaktur jenis persediaannya meliputi persediaan bahan pembantu, persediaan barang jadi, persediaan barang dalam proses dan persediaan bahan baku.

2.6 Peramalan

Menurut Aulia (2010), peramalan dilakukan untuk menentukan jumlah permintaan terhadap suatu produk dan merupakan langkah awal dari proses perencanaan dan pengendalian produksi. Dalam peramalan ditetapkan jenis produk apa yang diperlukan (*what*), jumlahnya (*how many*), dan kapan dibutuhkan (*when*).

Beberapa definisi peramalan menurut Santoso (2009) :

1. Perkiraan munculnya sebuah kejadian masa depan, berdasarkan data yang ada di masa lampau.
2. Proses menganalisis data historis dan data saat ini untuk menentukan *trend* di masa mendatang.
3. Proses estimasi dalam situasi yang tidak diketahui.
4. Pernyataan yang dibuat tentang masa depan.
5. Penggunaan ilmu dan teknologi untuk memperkirakan situasi di masa depan.
6. Upaya sistematis untuk mengantisipasi kejadian atau kondisi di masa depan.

Peramalan juga dapat diartikan sebagai seni dan ilmu untuk memperkirakan kejadian pada masa yang akan datang, sedangkan aktivitas peramalan merupakan suatu fungsi bisnis yang berusaha memperkirakan penjualan dan penggunaan produk sehingga produk-produk itu dibuat dalam kuantitas yang tepat (Gasperz, 2005).

2.6.1 Tahapan Peramalan

Ramalan-ramalan yang berguna bagi manajemen harus dianggap sebagai suatu proses yang sistemik. Dengan kata lain, suatu ramalan jangan dianggap sebagai suatu hal yang permanen atau statis. Kondisi kenyataan yang berubah-ubah

mengharuskan suatu ramalan untuk dikaji ulang, direvisi, dan didiskusikan. Oleh karena itu, Lincolin (2010) membagi tahapan peramalan sebagai berikut:

1. Pengumpulan data, tahap ini sangat penting dalam melakukan peramalan, tahapan ini biasanya merupakan tahap yang cukup rumit dan seringkali merupakan tahap yang paling kritis karena tahap-tahap berikutnya dapat dilakukan atau tidak tergantung pada relevansi data yang diperoleh.
2. Pengujian pola data
3. Pemilihan teknik peramalan, dalam tahap ini data yang telah diuji dilakukan pemilihan metode peramalan berdasarkan perilaku data yang dimiliki data tersebut.
4. Peramalan periode masa lalu, tahap ini data yang telah diuji dilakukan perhitungan peramalan berdasarkan teknik peramalan yang telah dipilih.
5. Akurasi peramalan, tahap selanjutnya merupakan menghitung kesalahan peramalan dengan menggunakan MSE, MAD, MAPE, MPE
6. Tahap yang akhir yaitu Peramalan periode berikutnya dan menggunakan hasilnya dalam pengambilan keputusan.

2.6.2 *Horizon Waktu Peramalan*

Peramalan biasanya diklasifikasikan berdasarkan *horizon* waktu masa depan yang dilingkupinya. *Horizon* waktu terbagi menjadi beberapa kategori menurut Heizer dan Render (2011) :

1. Peramalan jangka pendek

Peramalan ini meliputi jangka waktu hingga satu tahun, tetapi umumnya kurang dari tiga bulan.

2. Peramalan jangka menengah

Peramalan jangka menengah atau *intermediate* umumnya mencakup hitungan bulan hingga tiga tahun.

3. Peramalan jangka panjang

Peramalan ini umumnya untuk perencanaan masa tiga tahun atau lebih.

2.6.3 Pola Data Peramalan

Tahapan utama yang harus dilakukan sebelum melakukan peramalan adalah memilih suatu metode deret waktu (*time series*) yang tepat dengan mempertimbangkan jenis pola data, sehingga metode yang paling tepat dengan pola tersebut dapat digunakan untuk melakukan peramalan, menurut Lincolin (2010), pola data dapat dibedakan menjadi empat jenis, yaitu:

1. Pola horizontal (H), terjadi bilamana data relatif stabil dalam periode waktu tertentu.
2. Pola musiman (M), terjadi bilamana suatu data runtut waktu mengalami perubahan pola yang berulang setiap tahun.
3. Pola siklis (C), adanya fluktuasi seperti gelombang disekitar garis trend, dan berulang setiap dua tahun, tiga tahun, atau lebih.
4. Pola trend (T), terjadi bilamana data runtut waktu mengandung komponen jangka panjang dan menunjukkan pertumbuhan atau penurunan dalam data tersebut sepanjang suatu periode yang panjang.

2.7 Uji Pola Data

Menurut Santoso (2009), Uji pola data pada intinya adalah menguji apakah sebuah data dapat dikatakan stasioner atau tidak. Jika pada data terdapat *trend*

atau ada komponen seasonal atau siklis, dikatakan bahwa data tidak dapat dikatakan stasioner. Namun sebaliknya, jika pada data tidak ada *trend*, seasonal ataukah siklis, maka data dapat dikatakan stasioner. Stasioneritas data penting untuk menentukan lebih jauh metode *forecasting* apa yang tepat dilakukan. Metode untuk data yang stasioner akan berbeda dengan metode *forecasting* apa yang tepat dilakukan. Metode untuk data yang stasioner akan berbeda dengan metode *forecasting* untuk data yang tidak stasioner.

Pada umumnya, jika sebuah data saling berkorelasi pada jarak waktu yang berdekatan, misalnya antara waktu t dengan waktu sebelumnya ($t-1$), maka dikatakan data mempunyai kecenderungan berotokolerasi. Besaran korelasi antara data ke t dan data ke $t-1$ cukup tinggi, kemudian menurun secara bertahap. Data demikian bisa diduga mempunyai unsur *trend* di dalamnya dan tidak bersifat random. Sebaliknya, data yang mempunyai korelasi antar waktu yang rendah serta tidak menunjukkan pola penurunan otokorelasi yang bertahap, pada data tersebut dapat dikatakan tidak ada unsur *trend*.

Pengujian stasioner data penting karena banyak teknik *forecasting* yang mensyaratkan data harus stasioner. Namun jika kegiatan *forecasting* yang dilakukan tidak mensyaratkan data harus stasioner (walaupun hal ini jarang), proses pengujian berikut ini tentu tidak perlu dilakukan. Pengujian stasioneritas data dapat dilakukan dengan dua cara yaitu dengan grafik atau dengan menghitung otokorelasi. Sebaliknya kedua cara dilakukan secara bersama-sama, karena saling melengkapi.

2.8 Single Exponential Smoothing

Menurut Santoso (2009), metode ini beranggapan bahwa semakin ‘jauh’ sebuah data terkini, semakin berkurang bobot data tersebut. Dengan demikian, jika

data terakhir adalah data tahun 2008, maka data tahun 2007 dinilai lebih penting dan diberi bobot lebih besar dalam upaya *forecasting* dibandingkan dengan data tahun 2006. Demikian pula, data tahun 2006 dianggap lebih berperan dalam prediksi dibanding data tahun 2005. Pada data terkini (misal tahun 2008), ada konstanta *smoothing* (α) untuk melakukan *forecast*. Pada data satu periode sebelum data terakhir (dalam hal ini tahun 2007), konstanta menjadi $\alpha(1-\alpha)$. Untuk data dua periode sebelum data terakhir (dalam hal ini tahun 2006), konstanta menjadi $\alpha(1-\alpha)^2$. Demikian seterusnya untuk n periode sebelum data terakhir.

Menurut Lincolin (2010), Pemulusan eksponensial merupakan prosedur yang mengulang perhitungan secara terus-menerus dengan menggunakan data terbaru. Metode ini didasarkan pada perhitungan rata-rata (pemulusan) data-data masa lalu secara eksponensial. Setiap data diberi bobot yang lebih besar. Bobot yang digunakan adalah α untuk data yang paling baru, $\alpha(1-\alpha)$ digunakan untuk data yang agak lama, $\alpha(1-\alpha)^2$ untuk data yang lebih lama lagi, dan seterusnya.

Dalam bentuk yang mulus, ramalan yang baru (untuk waktu $t+1$) dapat dianggap sebagai rata-rata yang diberi bobot terhadap data terbaru (pada waktu t) dan ramalan yang lama (untuk waktu t). Bobot α diberikan pada data terbaru, dan bobot $1-\alpha$ diberikan pada ramalan yang lama, di mana $0 < \alpha < 1$.

Dengan demikian :

$$\text{Ramalan baru} = \alpha \times (\text{data baru}) + (1-\alpha) \times (\text{ramalan yang lama})$$

Secara sistematis, persamaan pemulusan eksponensial dapat ditulis:

$$\hat{Y}_{t+1} = \alpha Y_t + (1 - \alpha)\hat{Y}_t \dots\dots\dots(2.1)$$

dengan:

$$\hat{Y}_{t+1} = \text{nilai ramalan untuk periode berikutnya}$$

α = konstanta pemulusan ($0 < \alpha < 1$)

Y_t = data baru atau nilai Y yang sebenarnya pada periode t

\hat{Y}_t = nilai pemulusan lama atau rata-rata yang dimuluskan hingga periode $t-1$

Agar α dapat diinterpretasikan dengan baik, dapat diuraikan sebagai berikut :

$$\begin{aligned}\hat{Y}_{t+1} &= \alpha Y_t + (1 - \alpha) \hat{Y}_t \\ &= \alpha Y_t + \hat{Y}_t - \alpha \hat{Y}_t \\ &= \hat{Y}_t + \alpha (Y_t - \hat{Y}_t)\end{aligned}$$

2.9 Menghitung Kesalahan Peramalan

Menurut Heizer dan Render (2011), Akurasi keseluruhan dari setiap model peramalan dapat dijelaskan dengan membandingkan nilai yang diramal dengan nilai aktual atau lainnya yang sedang diamati. Jika F_t melambangkan peramalan pada periode t , dan A_t melamangkan permintaan aktual pada periode t , maka kesalahan peramalannya (deviasinya) adalah sebagai berikut:

$$\begin{aligned}\text{Kesalahan Peramalan} &= \text{Permintaan Aktual} - \text{Nilai Peramalan} \\ &= A_t - F_t\end{aligned}$$

Ada beberapa perhitungan yang bisa digunakan untuk menghtiung kesalahan peramalan total. Perhitungan ini dapat digunakan untuk memandingkan model peramalan yang berbeda, mengawasi peramalan, dan untuk memastikan peramalan berjalan dengan baik.

2.9.1 Mean Squared Error (MSE)

Menurut Lincolin (2010), Kesalahan rata-rata kuadrat atau *mean squared error* (MSE) merupakan metode alternatif dalam mengevaluasi suatu teknik peramalan. Setiap kesalahan atau residual dikuadratkan, kemudian dijumlahkan dan

dibagi dengan jumlah observasi. Pendekatan ini menghukum suatu kesalahan peramalan yang besar karena dikuadratkan. Pendekatan ini penting karena suatu teknik yang menghasilkan kesalahan yang moderat lebih kecil tetapi kadang-kadang menghasilkan kesalahan yang sangat besar, Persamaan 3.6 menunjukkan bagaimana cara menghitung MSE ini:

$$MSE = \frac{\sum_{t=1}^n (y_t - \hat{y}_t)^2}{n}$$

2.9.2 Mean Absolute Deviation (MAD)

Menurut Lincolin (2010), salah satu cara untuk mengevaluasi teknik peramalan adalah dengan menggunakan penjumlahan kesalahan absolut. Simpangan absolut rata-rata atau Mean Absolute Deviation (MAD) mengukur akurasi peramalan dengan merata-ratakan kesalahan peramalan (nilai absolutnya). MAD ini sangat berguna jika seorang analis ingin mengukur kesalahan peramalan dalam unit ukuran yang sama seperti data aslinya. Persamaan berikut menunjukkan bagaimana cara menghitung MAD.

$$MAD = \left| \frac{Y_t - \hat{Y}_t}{n} \right| \dots\dots\dots(2)$$

Dimana:

- t = Jumlah periode
- Y_t = Permintaan pada periode t
- \hat{Y}_t = Ramalan untuk periode t
- n = Total jumlah periode
- || = Nilai absolut

2.9.3 Mean Absolute Percentage Error (MAPE)

Menurut Lincolin (2010), Kadang kala lebih bermanfaat jika kita menghitung kesalahan peramalan dengan menggunakan secara *persentase* ketimbang nilai absolutnya. Persentase kesalahan absolut rata-rata atau MAPE dihitung dengan menemukan kesalahan absolut setiap periode, kemudian membaginya dengan nilai observasi pada periode tersebut, dan akhirnya merata-ratakan persentase absolut ini. Pendekatan ini sangat berguna jika ukuran variabel peramalan merupakan faktor penting dalam mengevaluasi akurasi peramalan tersebut. MAPE memberikan petunjuk seberapa besar kesalahan peramalan dibandingkan dengan nilai sebenarnya dari series tersebut. MAPE juga dapat digunakan untuk memperbandingkan akurasi dari teknik yang sama atau berbeda pada dua series yang berbeda. Berikut ini menunjukkan bagaimana cara menghitung MAPE:

$$MAPE = \frac{\sum_{t=1}^n \left| \frac{Y_t - \hat{Y}_t}{Y_t} \right|}{n}$$

2.9.4 Mean Percentage Error (MPE)

Menurut Lincolin (2010), kadang kala perlu juga untuk menentukan apakah suatu metode peramalan bisa atau tidak (secara konsisten tinggi atau rendah). Presentase kesalahan rata-rata atau *mean percentage error* (MPE) digunakan dalam kasus seperti ini. MPE dihitung dengan cara menemukan kesalahan setiap periode, kemudian membaginya dengan nilai sebenarnya pada periode tersebut, dan kemudian merata-ratakan persentase kesalahan tersebut. Jika menghasilkan persentase mendekati nol. Jika hasil persentase negatifnya cukup

besar, maka metode peramalan tersebut menghasilkan hasil ramalan yang terlalu tinggi, demikian sebaliknya.

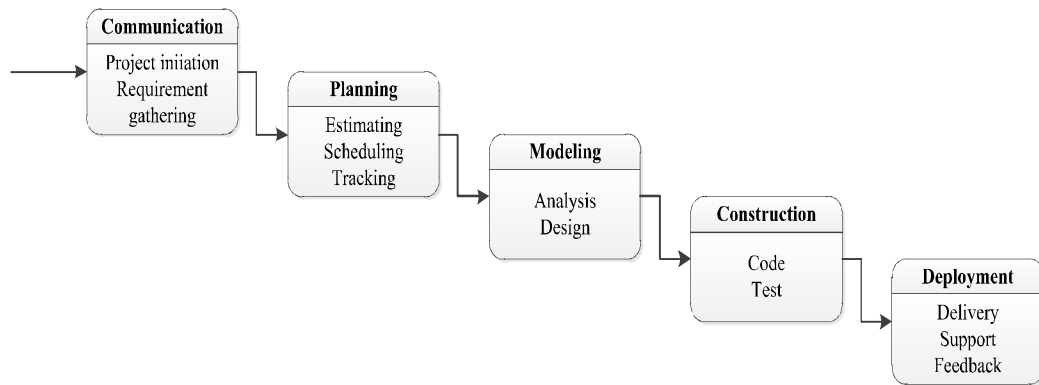
$$MPE = \frac{\sum_{t=1}^n \left| \frac{Y_t - \hat{Y}_t}{y_t} \right|}{n}$$

Keputusan kita dalam memilih suatu teknik peramalan sebagian tergantung pada apakah teknik tersebut menghasilkan kesalahan yang bisa dengan kecil atau tidak. Tentu saja sangat logis dan realistis jika kita menginginkan teknik peramalan yang menghasilkan kesalahan terkecil. Empat cara pengukuran akurasi peramalan yang dibahas di muka digunakan untuk tujuan berikut :

- a. Perbandingan akurasi dari dua teknik peramalan yang berbeda
- b. Pengukuran kegunaan atau reliabilitas suatu teknik peramalan
- c. Pencarian teknik peramalan yang optimal

2.10 *System Development Life Cycle (SDLC)*

Menurut Pressman (2015), Model *System Development Life Cycle (SDLC)* ini biasa disebut juga dengan model *waterfall* atau disebut juga *classic life cycle*. Adapun pengertian dari SDLC ini adalah suatu pendekatan yang sistematis dan berurutan. Tahapan-tahapannya adalah communication, perencanaan (*planning*), pemodelan (*modeling*), konstruksi (*construction*), serta penyerahan sistem perangkat lunak ke para pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan berkelanjutan pada perangkat lunak yang dihasilkan. Model *System Development Life Cycle (SDLC)* ditunjukkan pada gambar 2.1 berikut.



Gambar 2.1 Pengembangan menggunakan Model *Waterfall* (Pressman, 2015)

Penjelasan-penjelasan SDLC Model *Waterfall*, adalah sebagai berikut:

a. Communication

Sebelum memulai pembuatan sistem, sangatlah penting untuk melakukan komunikasi dan berkolaborasi dengan konsumen serta pemegang kepentingan sistem yang akan dibuat. Hal ini dimaksudkan untuk mengerti kebutuhan proyek yang akan dibuat dan untuk mengumpulkan segala kebutuhan yang membantu dalam menentukan fungsi *software* yang akan dibuat.

b. Planning

Proses perencanaan dalam pembuatan proyek akan dituliskan ke dalam *software project plan*. Perencanaan tersebut mendefinisikan tugas teknis yang akan dilakukan, resiko yang mungkin muncul, sumber daya yang akan dibutuhkan, produk yang dihasilkan, rencana kerja, dan estimasi waktu pengerjaan *project*.

c. Modeling

Pada proses *modeling* ini menerjemahkan syarat kebutuhan ke sebuah model analisis dan desain *software* yang dapat diperkirakan sebelum dibuat *coding*. Proses ini berfokus pada rancangan struktur data, arsitektur *software*,

representasi interface, dan detail (algoritma) prosedural. Tahapan ini menghasilkan dokumen yang disebut *software requirement*.

d. Construction

Construction merupakan proses membuat kode. *Coding* atau pengkodean merupakan penerjemahan desain dalam bahasa yang bisa dikenali oleh komputer. Programmer akan menerjemahkan transaksi yang diminta oleh *user*. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu software, artinya penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem yang telah dibuat. Tujuan testing adalah menemukan kesalahan-kesalahan terhadap sistem tersebut untuk kemudian bisa diperbaiki.

e. Deployment

Tahapan ini bisa dikatakan final dalam pembuatan sebuah aplikasi atau sistem. Setelah melakukan analisis, desain dan pengkodean maka sistem yang sudah jadi akan digunakan *user*. Kemudian aplikasi yang telah dibuat harus dilakukan pemeliharaan secara berkala. Dalam tahapan ini penulis tidak melakukan aktifitas *deployment* dikarenakan keterbatasan waktu penelitian penulis.

2.11 Analisis Sistem

Menurut Jogiyanto (2010), analisis sistem adalah penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dan dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikannya.

Menurut Al-fatta (2007), analisis sistem merupakan teknik pemecahan masalah yang menguraikan bagian-bagian komponen dengan mempelajari seberapa bagus bagian-bagian komponen tersebut bekerja dan berinteraksi untuk mencapai suatu tujuan. Analisis sistem memiliki tujuan utama yaitu untuk memahami dan mendokumentasikan kebutuhan bisnis dan kebutuhan sistem.

2.12 Perancangan Sistem

Menurut Jogyanto (2010) Perancangan Sistem dapat didefinisikan sebagai penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi. Perancangan sistem bertujuan untuk memenuhi kebutuhan para pemakai sistem dan memberikan gambaran yang jelas dari rancang bangun yang dibuat.

Menurut O'Brien dan Marakas (2010) menjelaskan bahwa perancangan sistem adalah sebuah kegiatan merancang dan menentukan cara mengolah sistem informasi dari hasil analisa sistem sehingga dapat memenuhi kebutuhan dari pengguna termasuk diantaranya perancangan user interface, data dan aktivitas proses.

2.13 Database

Menurut Connolly dan Begg (2010), Basis Data atau *Database* adalah kumpulan relasi-relasi logis dari data (dan deskripsi data) yang dapat digunakan bersama dan dibuat untuk memperoleh informasi yang di butuhkan oleh perusahaan. *Database* adalah relasi data logis yang terdiri dari *entity*, *attribute*, dan *relationship* dari informasi organisasi/perusahaan. Kegunaan dari *database* adalah

menghilangkan *redundancy* data, keterbatasan akses data, meningkatkan keamanan, *multiple User*, *independensi* data (kebebasan data).

Menurut Kusrini (2007), pengertian Basis Data adalah kumpulan data yang saling berelasi. Data sendiri merupakan fakta mengenai obyek, orang, dan lain-lain. Data dinyatakan dengan nilai (angka, deretan karakter, atau *symbol*).

2.14 Data Flow Diagram (DFD)

Menurut Jogiyanto (2010) *data flow diagram* merupakan diagram yang menggunakan notasi simbol untuk menggambarkan arus data sistem. DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika dan menjelaskan arus data dari mulai input hingga output data.

Menurut Jogiyanto (2010) ada beberapa simbol digunakan pada DFD untuk mewakili:

1. Kesatuan Luar (*External Entity*)

Kesatuan luar (*external entity*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi, atau sistem lain yang berada pada lingkungan luarnya yang memberikan input atau menerima output dari sistem.

2. Arus Data (*Data Flow*)

Arus Data (*data flow*) di DFD diberi simbol suatu panah. Arus data ini mengalir di antara proses, simpan data dan kesatuan luar. Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.

3. Proses (*Process*)

Proses (*process*) menunjukkan pada bagian yang mengubah input menjadi output, yaitu menunjukkan bagaimana satu atau lebih input diubah menjadi beberapa

output. Setiap proses mempunyai nama, nama dari proses ini menunjukkan apa yang dikerjakan proses.

4. Simpanan Data (*Data Store*)

Data Store merupakan simpanan dari data yang dapat berupa suatu file atau database pada sistem komputer.

2.15 *Testing*

Menurut Romeo (2003), testing adalah proses pemantapan kepercayaan akan kinerja program atau sistem sebagaimana yang diharapkan. *Testing Software* merupakan proses pengoperasian *Software* dalam suatu kondisi yang dikendalikan untuk verifikasi, mendeteksi *error* dan validasi. Verifikasi adalah pengecekan atau pengetesan entitas-entitas, termasuk *Software*, untuk pemenuhan dan konsistensi dengan melakukan evaluasi hasil terhadap kebutuhan yang telah ditetapkan.

Validasi adalah melihat kebenaran sistem apakah proses yang telah dituliskan sudah sesuai dengan apa yang dibutuhkan oleh pengguna. Deteksi *error* adalah *testing* yang berorientasi untuk membuat kesalahan secara intensif, untuk menentukan apakah suatu hal tersebut tidak terjadi. *Test case* merupakan suatu tes yang dilakukan berdasarkan pada suatu inisialisasi, masukan, kondisi ataupun hasil yang telah ditentukan sebelumnya. Adapun kegunaan dari *test case* ini, adalah untuk melakukan testing kesesuaian suatu komponen terhadap spesifikasi *Black Box Testing*.

2.15.1 *White-Box Testing*

Menurut Romeo (2003), *white box* testing adalah suatu metode desain *test case* yang menggunakan struktur kendali dari desain prosedural. Seringkali *white*

box testing di asosiasikan dengan pengukuran cakupan tes, yang mengukur persentase jalur-jalur dari tipe yang dipilih untuk dieksekusi oleh *test cases*. *White box testing* dapat menjamin semua struktur internal data dapat dites untuk memastikan validasinya.

Cakupan pernyataan, cabang dan jalur adalah suatu teknik *white box testing* yang menggunakan alur logika dari program untuk membuat *test cases* alur logika adalah cara dimana suatu bagian dari program tertentu dieksekusi saat menjalankan program. Alur logika suatu program dapat di representasi-kan dengan *flowgraph*.

2.15.2 Black-Box Testing

Menurut Romeo (2003), *Black box testing* dilakukan tanpa adanya suatu pengetahuan tentang detail struktur internal dari sistem atau komponen yang dites, juga disebut sebagai *functional testing*. *Black box testing* berfokus pada kebutuhan fungsional pada *Software*, berdasarkan pada spesifikasi kebutuhan dari software.

Dengan adanya *black box testing*, pereayasa *Software* dapat menggunakan kebutuhan fungsional pada suatu program. *Black box testing* dilakukan untuk melakukan pengecekan apakah sebuah *Software* telah bebas dari *error* dan fungsi-fungsi yang diperlukan telah berjalan sesuai dengan yang diharapkan.