

BAB II

LANDASAN TEORI

2.1. Definisi simulasi

Simulasi adalah sebagai suatu model sistem dimana komponennya di presentasikan oleh prosesor-prosesor aritmetika dan logika yang dijalankan komputer untuk memperkirakan sifat-sifat dinamis system tersebut (Emshoff & Simun, 1970).

Simulasi adalah merupakan proses perencanaan model dari sistem nyata yang dilanjutkan dengan pelaksanaan eksperimen terhadap model untuk mempelajari perilaku sistem atau evaluasi strategi (Shannon, 1975).

Simulasi adalah tiruan sistem nyata yang dikerjakan secara manual atau komputer, yang kemudian di observasi dan disimpulkan untuk mempelajari karakterisasi sistem (Banks & Carson 1984).

Simulasi merupakan proses perencanaan model matematis atau logis sistem nyata, melakukan eksperimen terhadap model dengan menggunakan komputer untuk menggambarkan, menjelaskan dan memprediksi perilaku sistem (Hoover & Perry, 1990).

Simulasi didefinisikan sebagai sekumpulan metode dan aplikasi untuk menirukan atau mereprestasikan perilaku dari suatu sistem nyata, yang biasanya dilakukan pada komputer dengan menggunakan perangkat lunak tertentu (Law & Kelton, 1991)

Simulasi merupakan proses aplikasi membangun model dari sistem, melakukan eksperimen dengan model tersebut untuk menjelaskan perilaku sistem, mempelajari kinerja sistem atau untuk membangun sistem baru sesuai dengan kinerja yang diinginkan (Khosnevis, 1994).

2.1.1. Metode simulasi

Metode simulasi merupakan perancangan model dari suatu sistem nyata (Real) dan pelaksanaan eksperimen-eksperimen dengan model ini untuk tujuan memahami tingkah laku sistem atau untuk menyusun strategi (dalam suatu batas limit yang ditentukan oleh sebuah satu atau beberapa kriteria) sehubungan dengan operasi sistem tersebut. Metode simulasi dapat menjelaskan tingkah laku dari sebuah model matematika yang dibuat sesuai dengan karakter sistem yang asli sehingga seorang analis bisa mengambil kesimpulan tentang tingkah laku dari sistem dunia nyata (STIKOM Digital Library, 2001).

2.1.2. Kesimpulan

Simulasi adalah suatu teknik numerik untuk melakukan percobaan-percobaan pada suatu komputer digital, yang melibatkan bentuk-bentuk fungsi matematika dan logika tertentu untuk menjelaskan tingkah laku dan struktur suatu nyata yang kompleks. Simulasi dapat digunakan untuk merancang, menganalisa dan menilai suatu sistem.

2.2. Balik Modal

2.2.1. Definisi Modal Menurut (KBBI)

Menurut kamus besar bahasa Indonesia (KBBI) modal adalah:

- a. **Uang** yang dipakai sebagai pokok (induk) untuk berdagang, melepas uang yang dapat dipergunakan untuk menghasilkan sesuatu yang menambah kekayaan dan sebagainya.
- b. **Barang** yang digunakan sebagai dasar atau bekal untuk bekerja (berjuang dan sebagainya).

2.2.2. Jenis-jenis Modal

- c. **Asing** adalah modal dari bangsa (negara) asing yang ditanam dalam suatu negara dengan maksud memperoleh keuntungan yang cukup.
- d. **Bergerak** adalah modal yang dapat dipindah-pindahkan (seperti mobil, uang).
- e. **Campuran** adalah modal patungan
- f. **Dasar** adalah modal yang merupakan dasar utama ; Modal utama.
- g. **Dengkul** adalah modal yang tidak berupa uang atau harta, tetapi hanya berupa perkataan atau tenaga.
- h. **Ekuitas** adalah modal yang diperoleh melalui pasar modal.
- i. **Kerja** adalah bagian modal yang beredar dalam jangka waktu pendek, yaitu beredar dari uang menjadi persediaan barang piutang dan menjadi uang kembali.
- j. **Manusiawi** adalah bentuk modal yang berupa keterampilan dan kecakapan.

- k. **Patungan** adalah modal yang diperoleh atas dasar pengumpulan uang bersama-sama untuk maksud tertentu.
- l. **Swasta** adalah modal yang ditanam oleh swasta untuk memperoleh keuntungan yang cukup.
- m. **Tetap** adalah modal yang tidak dapat dipindah-pindahkan (seperti gedung, tanah)
- n. **Tetap masyarakat** adalah modal yang jumlahnya cukup besar yang ditanamkan dalam sektor perhubungan dan angkutan dengan tujuan menghemat sumber produksi yang ada.

2.2.3. Masa Balik Modal

Jangka waktu yang diperlukan untuk memperoleh kembali modal yang ditanamkan dalam suatu investasi; jangka waktu pengembalian modal dihitung atas dasar nisbah (rasio) dan investasi awal terhadap arus kas masuk tahunan (payback period) (Glosarium, 2003).

2.2.4. Payback Period

Jangka waktu balik modal yang menggunakan payback period biasanya menghasilkan angka yang lebih menarik dibandingkan dengan ROI (Return On Investment). Hal ini dikarenakan akumulasi laba yang diperhitungkan biasanya mengabaikan unsur bunga pinjaman, pajak tahunan, depresiasi dan amortisasi. Payback period dimaknai saat ketika akumulasi EBITDA mencapai nilai Capex (capital expenditure, atau belanja modal).

Laba yang dimaksud dalam EBITDA merupakan perhitungan laporan Laba Rugi, bukan laporan arus kas. Perbedaannya: dalam laporan Laba rugi, seseorang harus memahami Biaya Modal dari transaksi atau produk (barang/jasa) yang dijual, jadi bukan biaya pembelian (belanja HPP) bulam berjalan.

Terkait Capex, perlu disepakati bersama bahwa belanja modal itu terkait unsur-unsur yang bisa didepresiasi (aset, termasuk biaya renovasi) dan diamortisasikan (misalnya biaya awal waralaba, biaya persiapan bisnisnya). Artinya, investor harus memahami bahwa Capex hanya sebagian dari total investasi yang dibutuhkan (Utomo Njoto, 2016).

2.2.5. Perhitungan Balik Modal

Istilah balik modal kerap disamakan dengan *break even point* (BEP) namun kenyataanya berbeda.

Beragam cara menghitung balik modal dipraktikan banyak pengusaha disesuaikan dengan sudut pandang masing-masing namun untuk sebuah usaha kecil yang sederhana perhitungan balik modal terbaik dapat ditentukan berdasarkan laba bersih yang diterima setiap bulan sebagai pengembalian modal investasi bisnis.

Jumlahnya akan menyesuaikan keinginan pemilik usaha. Semakin besar pengembalian modal itu maka semakin cepat balik modalnya. Dapat juga diilustrasikan seperti berikut :

Tabel 2.1. Ilustrasi perhitungan balik modal menggunakan ROI (Return On Investment)

Modal awal keseluruhan	Laba keseluruhan Bulan Pertama :
Rp. 30.000.000,- (tempat, bahan baku, dll)	<ul style="list-style-type: none"> ➤ Pemasukan : Rp. 5.000.000,- (dari hasil produksi) ➤ Pengeluaran : Rp. 3.000.000,- (gaji karyawan, bahan baku, listrik, dll)
	Laba bersih bulan pertama: Rp. 2.000.000,-
Keterangan : Kerugian dari modal awal Rp. 28.000.000,- (dari sini kita dapat memperkirakan balik modal yang dikeluarkan akan kembali dalam jangka waktu kurang lebih 15 bulan.	

2.3. Pengertian ROI (Return On Investment)

ROI (Singkatan bahasa Inggris : *Return On Investment*) atau ROR (Singkatan bahasa Inggris : *Rate of Return*) – dalam bahasa Indonesia Laba atas Investasi - adalah rasio uang yang diperoleh atau hilang pada suatu investasi, relatif terhadap jumlah uang yang diinvestasikan. Jumlah uang yang diperoleh atau hilang tersebut dapat disebut bunga atau laba/rugi. Investasi uang dapat dirujuk sebagai asset, modal, pokok, basis biaya investasi.

ROI biasanya dinyatakan dalam bentuk presentase dan bukan dalam nilai desimal. ROI tidak memberikan indikasi berapa lamanya suatu investasi. Namun

demikian, ROI sering dinyatakan dalam satuan tahunan atau disetahunkan dan sering juga dinyatakan untuk suatu tahun kalender atau fiskal.

ROI juga dikenal sebagai tingkat laba (*rate of profit*) atau hasil suatu investasi pada saat ini, masa lampau atau prediksi di masa mendatang. Atau bahasa sederhana ROI merupakan pengembalian keuntungan atas investasi (Wikipedia)

2.3.1. Pengertian ROI menurut para ahli

ROI (*Return On Investment*) adalah salah satu bentuk dari rasio profitabilitas yang dimaksudkan untuk mengukur kemampuan perusahaan dengan keseluruhan dana yang ditanamkandalam aktiva yang digunakan untuk operasinya perusahaan untuk menghasilkan keuntungan (S. Munawir (2007:89).

ROI (*Return On Investment*) merupakan kemampuan perusahaan untuk menghasilkan keuntungan yang akan digunakan untuk menutup investasi yang di keluarkan (Sutrisno, 2001:255).

ROI (*Return On Investment*) adalah rasio yang menunjukkan seberapa banyak laba bersih yang bisa diperoleh dari seluruh kekayaan yang dimiliki perusahaan (Suad Husnan & Enny Pudjiastuti (2004:74).

Rasio ini menunjukkan berapa persen diperoleh laba bersih bila diukur dari modal pemilik (Sofyan S. Harahap, 2007:305).

2.3.2. Faktor-faktor yang mempengaruhi *Return On Investment* (ROI)

ROI juga dapat dilihat dengan mengkombinasikan dua faktor, yaitu:

1. *Turnover* dari *operating assets* (Tingkat perputaran aktiva yang digunakan untuk operasi, yaitu kecepatan berputarnya *operating assets* dalam suatu periode tertentu).
2. *Profit Margin*, yaitu keuntungan operasi yang dinyatakan dalam prosentase dan jumlah penjualan bersih, *profit margin* ini mengukur tingkat keuntungan yang dapat dicapai oleh perusahaan dihubungkan dengan penjualan.

Besarnya ROI akan berubah kalau ada perubahan *profit margin* atau *assets turnover*, baik masing-masing atau kedua-duanya. Dengan demikian maka pimpinan perusahaan dapat menggunakan salah satu atau kedua-duanya dalam rangka usaha untuk memperbesar ROI. Usaha mempertinggi ROI dengan memperbesar *profit margin* adalah bersangkutan dengan usaha untuk mempertinggi efisiensi disektor produksi, penjualan dan administrasi. Usaha mempertinggi ROI dengan memperbesar *assets turn over* adalah kebijaksanaan investasi dana dalam berbagai aktiva, baik aktiva lancar maupun aktiva tetap.

2.3.3. Analisis *Return On Investment* (ROI)

Analisa *Return on Investment* (ROI) dalam analisa keuangan mempunyai arti yang sangat penting sebagai salah satu tehnik analisa keuangan yang bersifat menyeluruh (komprehensif). Analisa ROI ini sudah merupakan tehnik yang lazim

digunakan oleh pemimpin perusahaan untuk mengukur efektivitas dari keseluruhan operasi perusahaan.

Return On Investment itu sendiri adalah salah satu bentuk dari rasio profitabilitas yang dimaksudkan untuk dapat mengukur kemampuan perusahaan dengan keseluruhan dana yang ditanamkan dalam aktiva yang digunakan untuk operasinya perusahaan untuk menghasilkan keuntungan. Dengan demikian ratio ini menghubungkan keuntungan yang diperoleh dari operasinya perusahaan (*net operating income*) dengan jumlah investasi atau aktiva yang digunakan untuk menghasilkan keuntungan operasi tersebut (*net operating assets*).

Adapun rumus *Return On Investment* adalah sebagai berikut:

$$RETURN ON INVESTMENT = \frac{\text{Laba Bersih Sesuai Pajak}}{\text{Total Aktiva}} \times 100\%$$

(Sartono, 2010:123)

“Return On Investment (ROI) merupakan pengukuran kemampuan perusahaan secara keseluruhan dalam menghasilkan keuntungan dengan jumlah keseluruhan aktiva yang tersedia di dalam perusahaan (Sartono, 2010:123).

Semakin tinggi rasio ini, semakin baik keadaan perusahaan.

2.3.4. Fungsi analisis *Return On Investment* (ROI)

- a. Sebagai salah satu kegunaan yang prinsipil ialah sifatnya yang menyeluruh. Apabila perusahaan sudah menjalankan praktek akuntansi yang baik maka manajemen dengan menggunakan teknik analisa ROI dapat mengukur efisiensi penggunaan modal yang bekerja, efisiensi produksi dan efisiensi bagian penjualan.
- b. Apabila perusahaan mempunyai data industri sehingga dapat diperoleh ratio industri, maka dengan analisa ROI dapat dibandingkan efisiensi penggunaan modal pada perusahaannya dengan perusahaan lain yang sejenis, sehingga dapat diketahui apakah perusahaannya berada dibawah, sama, atau diatas rata-ratanya. Dengan demikian akan dapat diketahui dimana kelemahannya dan apa yang sudah kuat pada perusahaan tersebut dibandingkan dengan perusahaan lain yang sejenis.
- c. Analisa ROI juga dapat digunakan untuk mengukur efisiensi tindakan-tindakan yang dilakukan oleh divisi atau bagian, yaitu dengan mengalokasikan semua biaya dan modal kedalam bagian yang bersangkutan. Arti pentingnya mengukur *rate of return* pada tingkat bagian adalah untuk dapat membandingkan efisiensi suatu bagian dengan bagian yang lain didalam perusahaan yang bersangkutan.
- d. Analisa ROI juga dapat digunakan untuk mengukur profitabilitas dari masing-masing produk yang dihasilkan oleh perusahaan. Dengan menggunakan "*product cost system*" yang baik, modal dan biaya dapat dialokasikan kepada berbagai-bagai produk yang dihasilkan oleh

perusahaan yang bersangkutan, sehingga dengan demikian akan dapat dihitung profitabilitas dari masing-masing produk.

- e. ROI selain berguna untuk keperluan control, juga berguna untuk keperluan perencanaan. Misalnya ROI dapat digunakan sebagai dasar untuk pengambilan keputusan kalau perusahaan akan mengadakan ekspansi.

2.3.5. Kelemahan analisis *Return On Investment* (ROI)

- a. Salah satu kelemahan yang prinsipil ialah kesukarannya dalam membandingkan *rate of return* suatu perusahaan dengan perusahaan lain yang sejenis, mengingat bahwa kadang-kadang praktek akuntansi yang digunakan oleh masing-masing perusahaan tersebut adalah berbeda-beda. Perbedaan metode dalam penilaian berbagai-bagai aktiva antara perusahaan yang satu dengan perusahaan yang lain, perbandingan tersebut akan dapat memberi gambaran yang salah.
- b. Kelemahan lain dari tehnik analisa ini adalah terletak pada adanya fluktuasi nilai dari uang (daya belinya). Suatu mesin atau perlengkapan tertentu yang dibeli dalam keadaan inflasi nilainya berbeda dengan kalau dibeli pada waktu tidak ada inflasi, dan hal ini akan berpengaruh dalam menghitung *investment turnover* dan *profit margin*.
- c. Dengan menggunakan analisa *rate of return* atau *return on investment* saja tidak akan dapat digunakan untuk mengadakan perbandingan antara dua perusahaan atau lebih dengan mendapatkan kesimpulan yang memuaskan.

2.3.6. Pengaruh *Profit Margin* terhadap *Return On Investment (ROI)*

Rasio laba usaha dengan penjualan penjualan neto berkaitan dengan total aktiva yang digunakan untuk mencapai *sales revenue*. Rasio laba usaha dengan penjualan bersifat komplementer (pelengkap) dengan rasio laba bersih dengan *Return On Investment*.

Berdasarkan teori-teori tersebut diatas dengan demikian dapatlah dikatakan bahwa *profit margin* dimaksudkan untuk mengetahui efisiensi perusahaan dengan melihat kepada besar kecilnya laba usaha dalam hubungannya dengan *sales*, sedangkan “*operating turnover*” dimaksudkan untuk mengetahui efisiensi perusahaan dengan melihat kepada kecepatan perputaran *operating assets* dalam suatu periode tertentu.

Hasil akhir dari pencampuran kedua efisiensi *profit margin* dan *operating assets turnover* menentukan tinggi rendahnya *earning power*. Oleh karena itu makin tingginya tingkat *profit margin* atau *operating assets turnover* masing-masing atau keduanya akan mengakibatkan naiknya *earning power*.

Seperti halnya yang telah dikemukakan **S. Munawir** (2007:89) bahwa: “Besarnya *Return On Investment* akan berubah kalau ada perubahan *Profit Margin* atau *Asset Turn Over*, baik masing-masing atau keduanya.”

2.4. Pengertian Aplikasi

Aplikasi berasal dari kata *application* yang artinya penerapan, lamaran, penggunaan. Secara istilah aplikasi adalah program siap pakai yang dibuat untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain dan dapat digunakan oleh sasaran yang dituju.

Aplikasi bisa dikatakan suatu perangkat lunak yang siap pakai dengan menjalankan instruksi-instruksi dari pengguna (user), aplikasi banyak diciptakan untuk membantu berbagai keperluan, seperti: membuat laporan, percetakan, pencatatan dan lain lain.

Aplikasi adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu. Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari (user) pengguna (Menurut Kamus besar bahasa Indonesia).

Aplikasi merupakan penggunaan dalam suatu komputer, instruksi (instruction) atau pertanyaan (statement) yang disusun sedemikian rupa sehingga komputer dapat memproses input menjadi output (Menurut Jagiyanto, 2005).

Aplikasi adalah perangkat lunak yang digunakan untuk tujuan tertentu, seperti mengolah dokumen, mengatur Windows &, permainan (game) dan sebagainya (Rachman Hakim, 2003).

2.4.1. Jenis-jenis Aplikasi

a. Word Processing

Word Processing adalah program yang dapat dipakai untuk menyunting naskah. Contoh : Microsoft Word, Lotus Ami Pro dan WordPerfect.

b. Desktop Publishing

Merupakan program yang mengatur tata letak cetakan pada suatu naskah sehingga siap untuk dicetak. Contoh : Ventura Publisher, Page Maker.

c. Program Spreadsheet

Merupakan program-program yang digunakan untuk mengolah data secara berkolom. Contoh : Microsoft Excell, Lotus Improv.

d. Database Management System

Salah satu kegunaan komputer didalam organisasi adalah untuk menyimpan data dalam jumlah besar. Dari data ini dapat dihasilkan berbagai informasi. Untuk menyimpan, mengolah data, dan kemudian menghasilkan informasi, diperlukan program yang disebut dengan program database management system (disingkat DBMS) dan sering disebut dengan program database saja. Contoh : Paradox, Foxpro, Microsoft Access, Approach.

e. Graphics

Salah satu kebutuhan pengguna adalah membuat gambar. Untuk itu ia bisa menggunakan program yang khusus digunakan untuk membuat gambar atau graphics. Seseorang yang tidak pintar menggambar dengan tangan, dapat membuat gambar yang bagus di komputer, karena gambar

dikomputer mudah diubah dan diolah. Contoh : Corel Draw, Stanford Graphics, Visio.

f. Program Akuntansi

Aplikasi yang juga banyak dipakai dalam dunia bisnis adalah aplikasi yang berhubungan dengan keuangan dan akuntansi. Contoh : DacEasy Accounting, Pacioli 2000, PeachTree Accounting.

g. Program Statistik

Program statistik merupakan program yang digunakan untuk melakukan penelitian yang berhubungan dengan analisis statistik. Contoh : SAS, SPSS, Statisca.

h. Communication

Communication merupakan program yang digunakan untuk berkomunikasi dengan pemakai komputer lain. Contoh : Carbon Copy, DataFax, Procomm Plus, CrossTalk.

i. Multimedia

Multimedia merupakan software yang digunakan untuk menghubungkan komputer dengan peralatan multimedia seperti kamera video, kamera digital, video player. Contoh : Microsoft Video.

j. Game

Game merupakan program untuk permainan. Contoh : Flight Simulator, Baseball, Mobile Legend, vainglori.

k. Antivirus

Antivirus merupakan program yang digunakan untuk mendeteksi dan menghilangkan virus yang tertular pada komputer yang sedang dipakai.

Contoh : McAfee VirusScan, Norton Antivirus, AVG.

2.5. Pengertian Bootstrap

Bootstrap adalah framework HTML, CSS dan Javascript yang sangat populer untuk membangun web. (Anonim, 2011)

Framework bootstrap merupakan *framework* ataupun alat yang digunakan untuk membuat situs web responsif secara cepat, mudah dan gratis. *Bootstrap* terdiri dari CSS dan HTML untuk menghasilkan *Grid, Layout, Table, Form, Navigation* dan lain-lain. *Bootstrap* terdapat *Jquery Plugins* untuk menghasilkan UI yang menarik seperti *Transitions, Modal, Dropdown, Tooltip, Button* dan *Carousel*. Bootstrap dapat di download di situs resminya www.getbootstrap.com, didalam folder bootstrap terdapat tiga folder. Yaitu folder css, img dan js.

Dalam *bootstrap* terdapat struktur grid, yaitu struktur dua dimensi yang terdiri dari sumbu horizontal dan sumbu vertikal, sehingga akan tersusun kolom dan baris seperti halnya kolom pada media cetak surat kabar. Struktur kolom dalam bootstrap secara default terdiri dengan lebar grid 940px dan jumlah kolom maksimal adalah 12 kolom pada layar desktop. Struktur grid pada *bootstrap* mempunyai kelebihan yaitu sintak yang mudah dan memiliki fitur responsif, yang berarti layout yang dibuat dengan bootstrap dapat beradaptasi secara

otomatis terhadap ukuran layar mobile device mulai dari desktop komputer, tablet sampai dengan *smartphone*. (Mochamad, 2015)

2.5.1. Keunggulan Bootstrap

Bootstrap memiliki keunggulan dan juga kekurangan antara lain :

- a. Waktu pembuatan yang lebih cepat : Karena elemen-elemen yang biasa ada dalam sebuah website pada umumnya sudah dibuatkan *class*-nya oleh *Bootstrap*, jadi kita tinggal memanggilnya dalam *theme*.
- b. *Template* yang menggunakan *Bootstrap* lebih rapi : Mungkin ini akan sedikit relatif, karena kebiasaan masing-masing *developer* membuat *code* pasti berbeda-beda. Namun pada dasarnya, karena *Bootstrap* sudah terstruktur, untuk modifikasi *class* dan penambahan *class* memang sebaiknya mempertahankan kerapihan *code* yang sudah ada lebih dulu.
- c. *Template* yang menggunakan *Bootstrap* lebih ringan.
- d. Responsive dan tidak responsive : Ya, dengan sedikit modifikasi *template*, kita bisa memilih antara mengaktifkan sifat responsive dan tidak responsive.
- e. Dokumentasi Lengkap : twitter *Bootstrap* memiliki keunggulan dalam hal dokumentasinya, sehingga pengembang tidak perlu repot-repot untuk mempelajari. Dan hal ini menjadikan framework *Bootstrap* lebih dikenal daripada framework lainnya.

2.5.2. Kelamahan Bootstrap

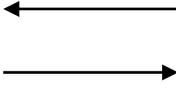
Mudah dikenali : Itulah salah satu hal buruk ketika sesuatu telah menjadi terkenal, semakin banyak orang yang mudah mengenali hal tersebut. tidak terkecuali dengan Bootstrap. mungkin bagi para pengembang front end website akan cukup merasa familiar ketika melihat sebuah template yang dibuat dengan Bootstrap. Dan akan terlihat sedikit seragam dengan template lain yang dibuat dengan Bootstrap, hal ini akan menjadi sedikit permasalahan ketika kita dituntut untuk membuat sebuah template yang berbeda dari kebanyakan. (Shah, 2015)

2.6. Data Flow Diagram (DFD)

Data Flow Diagram atau yang sering disebut juga Diagram Alur Data (DAD) adalah diagram yang digunakan untuk memodelkan sistem secara logic. Seperti halnya dengan alur dokumen, Diagram alur data pun dapat digunakan baik pada tahap analisis maupun tahap desain, namun kecenderungan diagram ini lebih cocok digunakan untuk tahap desain karena dengan diagram tersebut batasan ruang lingkup sistem terlihat sangat jelas sehingga pekerjaan pengembangan sistem yang dilakukan dapat lebih focus. Menurut Kristanto (2007), empat symbol dasar yang digunakan dalam DFD adalah :

Tabel 2.2. Simbol-simbol *Data Flow Diagram* (DFD)

Simbol	Fungsi
Entitas 	Diagram untuk menggambarkan suatu entitas eksternal, yang dapat mengirim data atau menerima data dari sistem.

<p>Aliran Data</p> 	<p>Digunakan untuk menunjukkan aliran data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.</p>
<p>Proses</p> 	<p>Digunakan untuk menunjukkan kegiatan atau kerja yang dilakukan orang, mesin atau komputer.</p>
<p>Simpanan Data</p> 	<p>Digunakan untuk menunjukkan simpanan dari data.</p>

2.7. Entity Relationship Diagram (ERD)

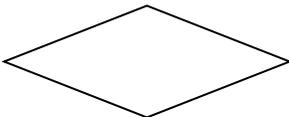
Menurut Kristanto (2007), ER-Model dapat didefinisikan suatu model untuk menjelaskan hubungan antar data dalam basis data. Pada ER-Model, data yang ada didunia nyata ditransformasikan dengan memanfaatkan sejumlah perangkat konseptual menjadi sebuah diagram data, yang umum disebut “*Entity Relationship Diagram*”. Menurut Kristanto (2007), komponen-komponen pembentuk model adalah :

1. Entitas, yaitu individu yang mewakili suatu yang nyata dan dapat dibedakan dari sesuatu yang lain.
2. Atribut, yaitu hal yang mendeskripsikan karakteristik dari entitas.
3. Relasi yang menunjukkan adanya hubungan diantara sejumlah entitas.

4. Kardinalitas yang menunjukkan jumlah maksimal entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain, yang dapat berupa :
- Satu ke satu (*One to One*)
 - Satu ke banyak (*One to Many*) atau banyak ke satu (*Many to One*)
 - Banyak ke banyak (*Many to Many*)

Menurut Kristanto (2007) ER-Model berisi komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang merepresentasikan seluruh fakta dari “dunia nyata” dapat digambarkan dengan lebih sistematis dengan menggunakan “*Entity Relationship Diagram*” (ERD). Notasi yang digunakan dalam ER-D Menurut Kristanto (2007) adalah :

Tabel 2.3. Notasi Simbolik ER-D (*Entity Relationship Diagram*)

Simbol	Fungsi
Persegi Panjang 	Menyatakan himpunan entitas
Lingkar atau Elips 	Menyatakan atribut yang berfungsi sebagai <i>key</i> diberi garis bawah.
Belah Ketupat 	Menyatakan himpunan
Garis	Sebagai penghubung antara himpunan

	relasi dengan himpunan entitas dan himpunan entitas dengan atributnya.
Kardinalitas 1 dan 1 (Satu ke Satu) 1 dan N (Satu ke Banyak) N dan N (Banyak ke Banyak)	Menyatakan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain.

2.8. Black-Box Testing

Menurut pressman (2010, p495) *Black-Box testing* berfokus pada persyaratan fungsional perangkat lunak yang memungkinkan *engineers* untuk memperoleh set kondisi *input* yang sepenuhnya akan melaksanakan persyaratan fungsional untuk sebuah program. *Black-Box testing* berusaha untuk menemukan kesalahan dalam kategori berikut:

1. Fungsi yang tidak benar atau fungsi yang hilang
2. Kesalahan antarmuka
3. Kesalahan dalam struktur data atau akses *database* eksternal
4. Kesalahan perilaku (*behavior*) atau kesalahan kinerja
5. Inisialisasi dan pemutusan kesalahan

Tes ini dirancang untuk menjawab beberapa pertanyaan-pertanyaan berikut ini:

1. Bagaimana validitas fungsional diuji?
2. Bagaimana perilaku dan kinerja sistem diuji?
3. Apa kelas *input* akan membuat kasus uji yang baik?

4. Apakah sistem *sensitive* terhadap nilai input tertentu?
5. Bagaimana batas-batas kelas data yang terisolasi?
6. Kecepatan dan volume data seperti apa yang dapat ditolerir sistem?
7. Efek apakah yang akan menspesifikasikan kombinasi data dalam sistem operasi?