

BAB II

LANDASAN TEORI

2.1 Penjualan

Menurut Himayati (2008), “penjualan adalah suatu transaksi yang bertujuan untuk mendapatkan suatu keuntungan, dan merupakan suatu jantung dari suatu perusahaan. Penjualan bisa dilakukan dengan jasa atau barang, baik kredit maupun tunai”.

Penjualan memiliki tujuan yaitu penjualan untuk mendapatkan laba atau keuntungan untuk perusahaan dari barang yang dibuat oleh produsen dilakukan dengan pengelolaan atau manajemen yang baik. Pelaksanaannya, penjualan tidak akan berjalan tanpa peran dari pelaku yang berperan di dalamnya contohnya pedagang, agen distributor, dan pemasaran

Kunci keberhasilan dalam melakukan mengelola tenaga penjualan adalah menyeleksi tenaga penjualan yang baik. Proses penting yang harus dilakukan dalam proses pengelolaan organisasi adalah *staffing* atau seleksi personil. Hal tersebut berlaku bagi organisasi lainnya seperti toko – toko, perusahaan, tenaga penjualan, maupun kantor – kantor. Terdapat tiga tahap utama dalam seleksi personil tenaga penjualan:

- a. Tentukan jumlah dan persyaratan jenis calon yang dibutuhkan.
- b. Memberi batasan dalam penerimaan lamaran dari para pelamar.
- c. Pilih hanya yang memenuhi persyaratan sesuai yang ditentukan.

2.2 Sales Promotion

Menurut Kotler (2005) *sales promotion* adalah “berbagai kumpulan alat-alat insentif, yang sebagian besar berjangka pendek, yang dirancang untuk merangsang pembelian produk atau jasa tertentu dengan lebih cepat dan lebih besar oleh konsumen atau pedagang”.

Selain itu, pendapat *sales promotion* oleh Utami (2008) adalah “dorongan jangka pendek untuk pembelian atau penjualan suatu produk atau jasa”.

Definisi – definisi diatas dapat disimpulkan bahwa *sales promotion* merupakan *tools* atau alat – alat insentif yang digunakan untuk mendorong pembelian atau penjualan suatu jasa atau produk menjadi lebih besar dan lebih cepat yang sebgain besar untuk jangka pendek.

2.2.1 Tujuan Sales Promotion

Tujuan *sales promotion* bersumber pada tujuan komunikasi pemasaran. Tujuan ini dijabarkan dengan tujuan pemasaran yang lebih mendasar, yang dirancang untuk produk tertentu. Menurut Kotler dan Armstrong dalam buku Prinsip-prinsip Pemasaran (2008) tujuan dari *sales promotion* bervariasi sangat luas.

- a. Penjual bisa menggunakan promosi konsumen untuk meningkatkan penjualan jangka pendek atau membangun pangsa pasar jangka panjang.
- b. Tujuan promosi dagang mencakup : membuat pengecer mendagangkan produk baru dan memberi ruang lebih banyak untuk persediaan, membuat mereka membeli di muka.

- c. Tujuannya meliputi: mendapatkan lebih banyak dukungan armada penjualan untuk produk sekarang atau produk baru atau mendapatkan wiraniaga untuk mencari langganan baru.

Berdasarkan yang dikemukakan oleh Kotler dan Armstrong dapat disimpulkan bahwa tujuan yang ingin dicapai oleh penjual melalui *sales promotion* ialah dapat meningkatkan penjualan, mendorong pembelian konsumen dan mendapatkan pelanggan baru.

2.2.2 Alat – alat *sales promotion*

Menurut Kotler (2005) menyatakan bahwa alat-alat *sales promotion* terdiri dari:

1. Sampel

Sejumlah kecil produk yang ditawarkan kepada konsumen untuk dicoba.

2. Kupon

Sertifikat yang memberi pembeli potongan harga untuk pembelian produk tertentu.

3. Tawaran pengembalian uang (rabat)

Tawaran untuk mengembalikan sebagian uang pembelian suatu produk kepada konsumen yang mengirimkan “bukti pembelian” ke perusahaan manufaktur.

4. Paket Harga (transaksi potongan harga/diskon)

Menawarkan kepada konsumen penghematan dari harga biasa suatu produk.

5. Premium (hadiah pemberian)

Barang yang ditawarkan secara cuma-cuma atau dengan harga sangat miring sebagai insentif untuk membeli suatu produk

6. Program frekuensi

Program yang memberikan imbalan yang terkait dengan frekuensi dan intensif konsumen membeli produk atau jasa perusahaan tersebut.

7. Kontes, undian, dan permainan

Kegiatan promosi yang memberikan konsumen peluang untuk memenangkan sesuatu seperti uang tunai, perjalanan atau barang entah dengan keberuntungan atau dengan usaha ekstra.

8. Imbalan berlangganan

Uang tunai atau hadiah lain yang ditawarkan bagi penggunaan suatu produk atau jasa perusahaan.

9. Pengujian gratis

Mengundang calon pembeli menguji-coba produk tanpa biaya dengan harapan mereka akan membeli.

10. Garansi produk

Janji eksplisit atau implisit penjual bahwa produk tersebut akan bekerja sebagaimana telah ditentukan.

11. Promosi bersama

Dua atau lebih *merk* perusahaan bekerja sama dengan kupon, pengembalian uang, dan kontes untuk meningkatkan daya tarik.

12. Promosi – silang

Menggunakan suatu *merk* untuk mengiklankan *merk* lain yang tidak bersaing.

13. *point of purchase*

Display atau peragaan yang berlangsung ditempat pembayaran atau penjualan.

2.3 Sistem

Menurut Herlambang dan Tanuwijaya (2005) definisi sistem dapat dibagi menjadi dua pendekatan, yaitu pendekatan secara prosedur dan pendekatan secara komponen. Berdasarkan pendekatan prosedur, sistem didefinisikan sebagai kumpulan dari beberapa prosedur yang mempunyai tujuan tertentu. Berdasarkan pendekatan komponen, sistem merupakan kumpulan dari komponen-komponen yang saling berkaitan untuk mencapai tujuan tertentu.

Menurut Kristanto (2003), terdapat dua kelompok pendekatan di dalam mendefinisikan sistem, yaitu yang menekankan pada prosedurnya dan yang menekankan pada komponen atau elemennya.

1. Pendekatan sistem yang lebih menekankan pada prosedur, mendefinisikan sistem sebagai berikut: “Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul, bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu”. (Herlambang dan Tanuwijaya, 2005).
2. Pendekatan sistem yang merupakan jaringan kerja dari prosedur, lebih menekankan urutan-urutan operasi didalam sistem. Prosedur didefinisikan oleh Neuschel R. F. sebagai berikut: “Suatu prosedur adalah suatu urutan-urutan operasi klerikal (tulis-menulis), biasanya melibatkan beberapa orang di dalam satu atau lebih departemen, yang diterapkan untuk menjamin penanganan yang seragam dari transaksi-transaksi bisnis yang terjadi”. (Herlambang dan Tanuwijaya, 2005).
3. Pendekatan yang lebih menekankan pada elemen atau komponennya mendefinisikan sistem sebagai berikut: “Sistem adalah kumpulan dari elemen-

elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. (Herlambang dan Tanuwijaya, 2005).

2.4 Aplikasi

Menurut Jogiyanto (2005), “aplikasi adalah program yang ditulis dan diterjemahkan oleh *language software* untuk menyelesaikan suatu aplikasi tertentu”. Dari referensi Jogiyanto (2005) tersebut dapat disimpulkan bahwa aplikasi adalah perangkat lunak yang diubah ke dalam komputer yang terdapat berbagai perintah digunakan untuk menyelesaikan berbagai pekerjaan atau tugas – tugas tertentu misalnya penambahan data, penerapan, dan penggunaan.

2.5 Android

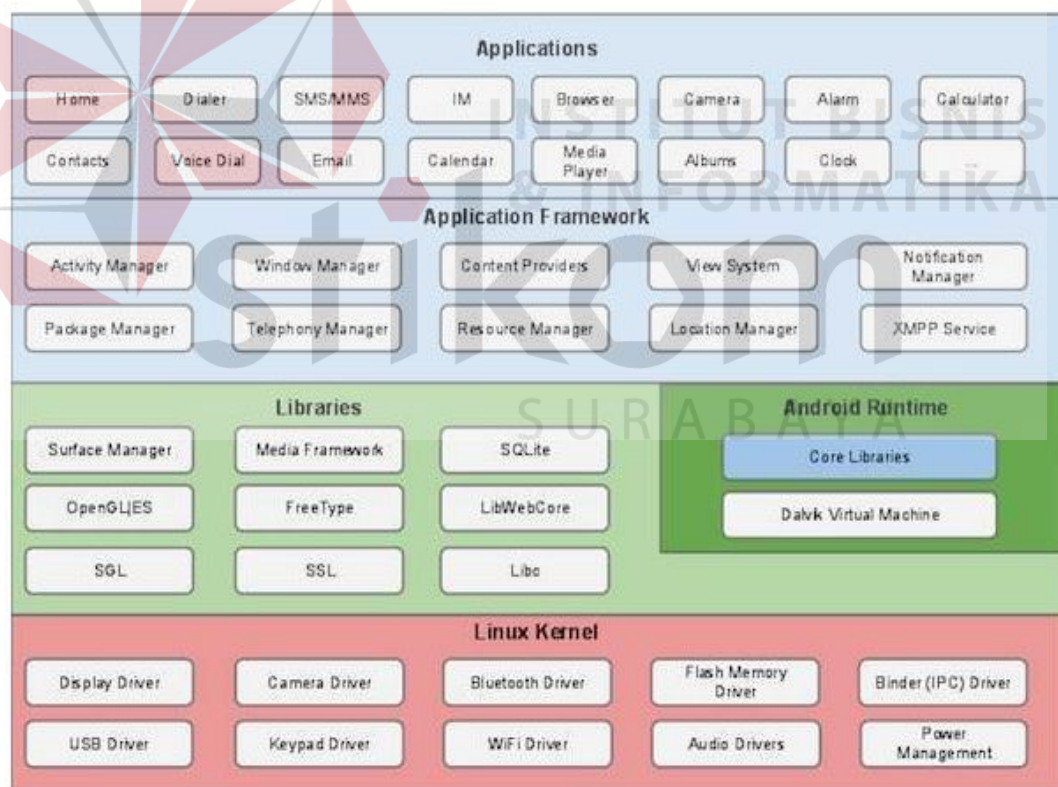
Android menurut Sfaat (2011), adalah sistem operasi untuk *mobile phone* atau telepon seluler yang berbasis linux. Sistem operasi ini sama seperti sistem operasi lainnya seperti Symbian yang digunakan oleh nokia, iOS digunakan oleh Apple dan Blackberry. Kelebihan utama Android adalah tidak berbayar dan *open source*, hal ini yang membuat *smartphone* Android memiliki harga yang relatif lebih murah dibandingkan dengan Blackberry atau iPhone walaupun perangkat keras (*hardware*) yang dimiliki lebih baik. Android memiliki beberapa fitur diantaranya: *WiFi hotspot, multi-touch, multitasking, GPS, Java support*, mendukung jaringan (*GSM, CDMA, LTE, dan WiMax*) dan Android memiliki kemampuan telepon selular pada umumnya.

Perilisan perdana Android, 5 November 2007, *Open Handset Alliance* dengan Android mendukung untuk pengembangan standar terbuka untuk perangkat

seluler. Pada Google, Google merilis kode – kode Android dengan menggunakan lisensi dari Apache, Apache merupakan lisensi perangkat lunak dan standar terbuka perangkat seluler.

2.5.1 Arsitektur Android

Arsitektur android terdiri dari 5 Layer diantaranya adalah *linux kernel*, *Android runtime*, *Library*, *Application Frameworks*, *application* dan *widget*. Gambar 2.3.1 memberikan gambaran komponen – komponen dalam arsitektur sistem operasi Android.



Gambar 2.1 Arsitektur Android

Menurut safaat (2011), secara garis besar arsitektur Android dapat dijelaskan dan digambarkan sebagai berikut:

1. *Applications dan Widgets*

Applications dan *Widgets* ini adalah *layer* dimana kita berhubungan dengan aplikasi saja, dimana biasanya kita download aplikasi kemudian kita lakukan instalasi dan jalankan aplikasi tersebut. *Layer* terdapat aplikasi inti termasuk *e-mail client*, program SMS, kalender, peta, *browser*, kontak, dan lain – lain. Semua aplikasi ditulis menggunakan bahasa pemrograman java.

2. *Application Frameworks*

Applications Frameworks ini adalah *layer* dimana para pembuat aplikasi melakukan pengembangan/pembuatan aplikasi yang akan dijalankan di sistem operasi Android, karena pada *layer* inilah aplikasi dapat dirancang dan dibuat, seperti *content-providers* yang berupa SMS dan panggilan telepon. Komponen – komponen yang termasuk dalam *Applications Frameworks* adalah sebagai berikut:

- a. *Views*
- b. *Content provider*
- c. *Resource manager*
- d. *Notification Manager*
- e. *Activity Manager*

3. *Libraries*

Libraries ini adalah *layer* dimana fitur – fitur Android berada, biasanya para pembuat aplikasi mengakses *libraries* untuk menjalankan aplikasinya. Berjalan

diatas *kernel*, *Layer* ini meliputi berbagai *library C/C++* inti seperti *Libc* dan *SSL*, serta:

- a. *Libraries* media untuk pemutaran media audio dan video.
- b. *Libraries* untuk manajemen tampilan.
- c. *Libraries Graphics* mencakup *SGL* dan *OpenGL* untuk grafis 2D dan 3D.
- d. *Libraries SQLite* untuk dukungan *database*.
- e. *Libraries SSL* dan *WebKit* terintegrasi dengan *web browser* dan *security*.
- f. *Libraries Live Webcore* mencakup *modern web browser* dengan *engine embedded web view*.
- g. *Libraries 3D* yang mencakup implementasi *OpenGL ES 1.0 API's*.

4. Android Runtime

Layer yang membuat aplikasi Android dapat dijalankan dimana dalam prosesnya menggunakan implementasi Linux. *Dalvik Cirtual Machine* (DVM) merupakan mesin yang membentuk dasar kerangka aplikais Android. Di dalam Android Runtime dibagi menjadi dua bagian, yaitu:

- a. *Core Libraries* : Aplikasi Android dibangun dalam bahasa java, sementara *Dalvik* sebagai mesin *virtualnya* mesinnya bukan *Virtual Machine Java*, sehingga diperlukan sebuah *libraries* yang berfungsi untuk menerjemahkan bahasa *Java/C* yang ditangani oleh *Core Libraries*.
- b. *Dalvik Virtual Machine* : Mesin *virtual* berbasis *register* yang dioptimalkan untuk menjalankan fungsi – fungsi secara efisien, dimana merupakan pengembangan yang mampu membuat *Linux kernel* untuk melakukan *threading* dan manajemen tingkat rendah.

5. *Linux Kernel*

Linux Kernel adalah *layer* dimana inti dari sistem operasi dari Android itu berada. Berisi *file – file* sistem yang mengatur sistem *processing, memory, resource, drivers*, dan sistem – sistem operasi Android lainnya. *Linux kernel* yang digunakan Android adalah *Linux Kernel release 2.6*.

2.5.2 Android Software Development Kit (SDK)

Menurut Safaat (2011), Android SDK adalah tools API (*Application programming Interface*) yang diperlukan untuk mengembangkan aplikasi pada *platform* Android yang menggunakan bahasa pemrograman Java. Android merupakan *subset* perangkat lunak untuk ponsel yang meliputi sistem operasi, *middleware* dan aplikasi kunci yang di-*release* oleh Google.

Saat ini disediakan Android SDK sebagai alat bantu dan API dikembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java. Sebagai *Platform* Android aplikasi netral, Android memberi kesempatan untuk membuat aplikasi yang dibutuhkan yang bukan merupakan aplikasi bawaan *Handphone/Smartphone*.

Adapun beberapa fitur menurut Safaat (2011) yang tersedia di Android adalah sebagai berikut ini:

1. Kerangka aplikasi : memungkinkan penggunaan dan penghapusan komponen yang tersedia.
2. *Dalvik* Mesin *Virtual* : mesin *virtual* dioptimalkan untuk perangkat telepon seluler.
3. Grafik : grafik di 2D dan grafis 3D berdasarkan *libraries* OpenGL.

4. SQLite : untuk penyimpanan data.
5. Mendukung media : audio, video, dan berbagai format gambar (MPEG4, H.264, AAC, AMR, JPG, PNG, GIF)
6. GSM, Bluetooth, EDGE, 3G, 4G dan WiFi (tergantung perangkat keras)
7. Kamera, *Global Positioning System* (GPS), Kompas, NFC (*Near Field Communication*), dan *Accelerometer* (tergantung perangkat keras).

2.5.3 Versi Android

1. Android versi 1.1

Pada 9 Maret 2009, Google merilis Android versi 1.1. Android versi ini dilengkapi dengan pembaruan estetis pada aplikasi jam *alarm*, *voice search* (pencarian suara), pengiriman pesan dengan *Gmail*, dan pemberitahuan *e-mail*.

2. Android versi 1.5 (CupCake)

Pada pertengahan Mei 2009, Google kembali merilis telepon seluler dengan menggunakan Android dan SDK dengan versi 1.5 (*CupCake*). Terdapat beberapa pembaruan untuk penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton video dengan modus kamera, mengunggah video ke *youtube* dan gambar ke *Picasa* langsung dari telepon, dukungan Bluetooth A2DP, kemampuan terhubung secara otomatis ke *headset Bluetooth*, animasi layar, dan *keyboard* pada layar yang dapat disesuaikan dengan sistem.

3. Android versi 1.6 (Donut)

Donut (Versi 1.6) dirilis pada bulan september dengan menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan indikator baterai

dan kontrol *applet VPN*. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus, kamera, *camcorder* dan galeri yang diintegrasikan, *CDMA / EVDO*, *802.1x*, *VPN*, *Gestures* dan *Text-to-speech engine*, kemampuan *dial* kontak, teknologi *text to change speech* (tidak tersedia pada semua ponsel) pengadaan resolusi VWGA.

4. Android versi 2.0/2.1 (Eclair)

Pada 3 September 2009 kembali diluncurkan ponsel Android dengan versi 2.0/2.1 (*Eclair*), perubahan yang dilakukan adalah pengoptimalan *hardware*, peningkatan *Google Maps* 3.1.2, perubahan *user interface* dengan *browser* baru dan dukungan *HTML 5*, daftar kontak yang beri, dukungan *flash* untuk kamera 3,2 MP, *digital zoom*, dan *Bluetooth* 2.1. Agar dapat bergerak cepat dalam persaingan perangkat generasi yang berikutnya, Google melakukan investasi dengan mengadakan kompetisi aplikasi *mobile* terbaik (*Killer apps*-aplikasi unggulan). Kompetisi ini berhadiah \$25,000 bagi setiap pengembang aplikasi terpilih. Kompetisi ini diadakan selama dua tahap yang tiap tahapnya dipilih 50 aplikasi terbaik.

Dengan semakin berkembangnya dan bertambahnya jumlah *handset* android, semakin banyak pihak ketiga yang berminat untuk menyalurkan aplikasi mereka dalam sistem operasi Android. Aplikasi yang terkenal yang diubah ke dalam sistem operasi Android adalah Shazam, Backgrounds, dan WeatherBug. Sistem operasi Android dalam situs internet juga dianggap penting untuk menciptakan aplikasi Android asli, contohnya MySpace dan Facebook.

5. Android versi 2.2

Pada 20 Mei 2010, Android 2.2 (Froyo) diluncurkan. Perubahan – perubahan umum terhadap versi – versi sebelumnya antara lain dukungan Adobe Flash 10.1, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, integrasi V8 *javascript engine* yang dipakai Google Chrome yang mempercepat kemampuan *rendering* pada *browser*, pemasangan aplikasi dalam SDCard, kemampuan WiFi *Hotspot portable*, dan kemampuan *auto update* dalam aplikasi Android *Market*.

6. Android versi 2.3 (Gingerbread)

Pada 6 Desember 2010, Android versi 2.3 (*Gingerbread*) diluncurkan. Perubahan – perubahan umum yang didapat dari Android versi ini antara lain peningkatan kemampuan permainan (*gaming*), peningkatan fungsi *copy paste*, layar *user interface* yang sudah didesain ulang, dukungan format video VP8 dan WebM, efek audio baru (*reverb*, *equalization*, *headphone virtualization*, dan *bass boost*), dukungan kemampuan *Near Field Communication* (NFC), dan dukungan jumlah kamera yang lebih dari satu.

7. Android versi 3.0/2.1 (Honeycomb)

Android *Honeycomb* dirancang khusus untuk tablet. Android versi ini mendukung ukuran layar yang lebih besar. *User Interface* pada *Honeycomb* juga berbeda karena sudah didesain untuk *tablet*. *Honeycomb* juga mendukung *multi processor* dan juga akselerasi perangkat keras (*hardware*) untuk grafis. *Tablet* pertama kali yang dibuat dengan menjalankan *Honeycomb* adalah Motorola Xoom.

8. Android versi 4.0 (ICS : Ice Cream Sandwich)

Pada tanggal 19 oktober 2011, diperkenalkannya Android versi 4.0 yang membawa fitur *Honeycomb* untuk *smartphone*, menambahkan fitur baru termasuk membuka kunci dengan pengenalan wajah, jaringan data, pemantauan penggunaan dan kontrol data, kontak jaringan sosial terpadu, perangkat tambahan fotografi, pencarian *email* secara *offline*, dan berbagi informasi dengan menggunakan NFC. Ponsel pertama yang menggunakan sistem operasi ini adalah Samsung Galaxy Nexus.

9. Android versi 4.1 (Jelly Bean)

Android *Jelly Bean* yang diluncurkan pada acara *Google I/O* membawa sejumlah keunggulan dan fitur baru. Adapun penambahan fitur baru diantaranya yaitu meningkatkan *input keyboard*, desain baru fitur pencarian, *UI* yang baru dan pencarian melalui *Voice Search* yang lebih cepat. *Google Now* yang juga menjadi bagian diperbarui pun tak ketinggalan. *Google Now* memberikan informasi yang tepat pada waktu yang tepat pula. Salah satu kemampuannya adalah dapat mengetahui informasi cuaca, lalu lintas, atau pun hasil pertandingan olahraga. Sistem operasi Android Jelly Bean 4.1 muncul pertama kali dalam produk tablet Asus, yaitu Google Nexus 7.

10. Android versi 4.4 (Kitkat)


Google mengumumkan Android 4.4 KitKat pada 3 September 2013, dengan tanggal rilis 31 Oktober 2013. Fitur terbaru dari android versi 4.4 ini adalah pembaruan antarmuka dengan bar status dan navigasi transparan pada layar depan, optimasi kinerja pada perangkat dengan spesifikasi lebih rendah, kerangka kerja pencetakan, NFC *Host Card Emulation* sebagai *Emulator*

Smart Card, *WebViews* berbasis *Chromium*, perluasan Fungsionalitas bagi layanan pendengar notifikasi, API umum untuk mengembangkan dan mengelola *text messaging client*, kemampuan untuk menentukan aplikasi SMS standar, kerangka kerja baru untuk transisi UI, kerangka kerja akses penyimpanan untuk mengambil konten dan dokumen dari sumber lain, *sensor batching*, *step detector*, dan *Counter API*, peningkatan tampilan mode layar penuh, tombol perangkat lunak dan status bar bisa diakses dari tepi dengan cara menggesek, penyeimbang audio, pemantauan audio, dan peningkatan suara audio, perekaan aktifitas layar yang terintegrasi, infra merah, peningkatan API *accessibility*, mesin *virtual* eksperimental baru, ART, dukungan *Bluetooth Message Access Profile* (MAP).

2.6 Bagan Alir Dokumen

Menurut Jogiyanto (2005) Bagan alir dokumen (*document flowchart*) atau di sebut juga bagan alir formulir (*form flowchart*) atau *paperwork flowchart* merupakan bagan (*charts*) yang menunjukkan alir (*flow*) didalam program atau prosedur sistem secara logika dapat didefinisikan sebagai bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem dengan menggunakan simbol seperti pada tabel berikut:

Tabel 2.1 Simbol-Simbol *Flowchart*

No.	Simbol	Nama Simbol <i>Flowchart</i>	Fungsi
1.		Dokumen	Untuk menunjukkan dokumen input dan output baik untuk

No.	Simbol	Nama Simbol <i>Flowchart</i>	Fungsi
			proses manual, mekanik atau komputer.
2.		Proses Komputerisasi	Menunjukkan kegiatan dari operasi program komputer.
3.		Database	Untuk menyimpan data.
4.		Penghubung	Menunjukkan hubungan di halaman yang sama.
5.		Penghubung Halaman Lain	Menunjukkan hubungan di halaman lain.
6.		Terminator	Menandakan awal/akhir dari suatu sistem.
7.		Decision	Menggambarkan logika keputusan dengan nilai <i>true</i> atau <i>false</i> .
8.		Kegiatan Manual	Untuk menunjukkan pekerjaan yang dilakukan secara manual.
9.		Simpanan Offline	Untuk menunjukkan file non-komputer yang diarsip urut angka.

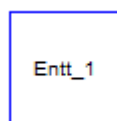
2.7 Data Flow Diagram

Menurut Kendall (2003), *Data Flow Diagram (DFD)* menggambarkan pandangan sejauh mungkin mengenai masukan, proses dan keluaran sistem, yang berhubungan dengan masukan, proses, dan keluaran dari model sistem yang dibahas. Serangkaian diagram aliran data berlapis juga bisa digunakan untuk merepresentasikan dan menganalisis prosedur-prosedur mendetail dalam sistem. Prosedur-prosedur tersebut yaitu konseptualisasi bagaimana data-data berpindah di dalam organisasi, proses-proses atau transformasi dimana data-data melalui, dan apa keluarannya. Jadi, melalui suatu teknik analisa data terstruktur yang disebut *Data Flow Diagram*, penganalisis sistem dapat merepresentasi proses-proses data di dalam organisasi.

Menurut Kendall (2003), dalam memetakan *Data Flow Diagram*, terdapat beberapa simbol yang digunakan antara lain:

1. *External entity*

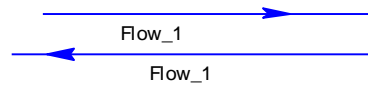
Suatu *external entity* atau entitas merupakan orang, kelompok, departemen, atau sistem lain di luar sistem yang dibuat dapat menerima atau memberikan informasi atau data ke dalam sistem yang dibuat.



Gambar 2.2 Simbol *External Entity*

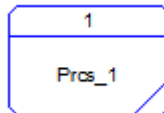
2. *Data Flow*

Data Flow atau aliran data disimbolkan dengan data tanda panah. Aliran data menunjukkan arus data atau aliran data yang menghubungkan dua proses atau *entitas* dengan proses.

Gambar 2.3 Simbol *Data Flow*

3. *Process*

Suatu proses dimana beberapa tindakan atau sekelompok tindakan dijalankan.

Gambar 2.4 Simbol *Process*

4. *Data Store*

Data store adalah simbol yang digunakan untuk melambangkan proses penyimpanan data.

Gambar 2.5 Simbol *Data Store*

2.8 *Entity Relationship Diagram*

Menurut Whitten (2007), ERD adalah model data yang menggunakan beberapa notasi untuk menggambarkan data dalam konteks entitas dan hubungan yang dideskripsikan oleh data tersebut.

Entity Relationship Diagram ini diperlukan agar dapat menggambarkan hubungan antar entitas dengan jelas, dapat menggambarkan batasan jumlah entitas dan partisipasi antar entitas, mudah dimengerti pemakai dan mudah disajikan oleh perancang *database*. Untuk itu, *entity relationship diagram* dibagi menjadi dua jenis model, yaitu:

1. *Conceptual Data model*

Conceptual Data model (CDM) adalah jenis model data yang menggambarkan hubungan antar tabel secara konseptual.

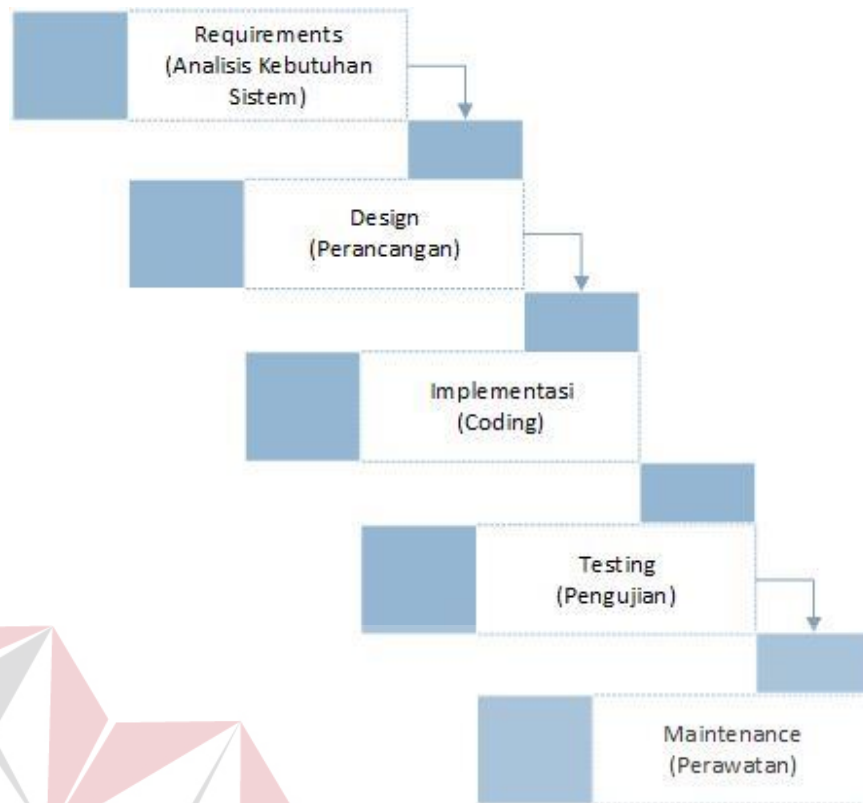
2. *Physical Data Model*

Physical Data Model (PDM) adalah jenis model data yang menggambarkan hubungan antar tabel secara fisik.

2.8 *System Development Life Cycle*

Menurut Pressman (2015), Model *System Development Life Cycle* (SDLC) ini kebanyakan disebut dengan model *classic life* atau disebut juga dengan *waterfall*. Pengertian dari SDLC adalah suatu pendekatan yang sistematis dan berurutan. SDLC memiliki beberapa tahapan yaitu *Requirements* (analisis sistem), *Analysis* (analisis kebutuhan sistem), *Design* (perancangan), *Coding* (implementasi), *Testing* (pengujian) dan *Maintenance* (perawatan).

Model eksplisit pertama dari kegiatan pengembangan perangkat lunak, berasal dari proses – proses rekayasa. Model ini dapat memberi penglihatan proses pengembangan. Karena, modelnya yang bertingkat dari satu tahap ke tahap yang lainnya berurutan ke bawah, model ini disebut dengan nama model *waterfall* seperti terlihat pada gambar 2.6 berikut.



Gambar 2.6 *System Development Life Cycle (SDLC) Model Waterfall*

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

a. *Requirement* (Analisis Kebutuhan Sistem)

Pada tahap pertama ini melakukan analisa bertujuan untuk mencari tahu secara detil mengenai kebutuhan apa saja yang dibutuhkan. Macam – macam kebutuhan seperti kebutuhan informasi bisnis, kebutuhan data dan kebutuhan user itu sendiri. Kebutuhan itu sendiri dibedakan menjadi tiga jenis. Pertama, kebutuhan teknologi. Dari kebutuhan tersebut dilakukan analisa untuk kebutuhan teknologi yang dibutuhkan dalam mengembangkan suatu sistem, sama seperti data penyimpanan informasi / *database*. Kedua kebutuhan informasi, contoh seperti informasi mengenai identitas pegawai. Ketiga, Kebutuhan *user*. Untuk hal ini dilakukan analisa yang berhubungan dengan kebutuhan *user* dan kategori *user*. Dari analisa yang telah disebutkan di atas,

terdapat satu hal lagi yang tidak kalah pentingnya dalam tahap analisa di metode SDLC, yaitu analisa biaya dan resiko. Untuk tahap ini dilakukan perhitungan seluruh biaya yang akan dibutuhkan seperti biaya untuk *implementasi*, *testing*, dan *maintenance*.

b. *Design* (Perancangan)

Tahap selanjutnya, dari hasil analisa kebutuhan sistem tersebut akan digunakan untuk sebuah design database, DFD, ERD, *user interface* (antarmuka pengguna) atau *Graphical User Interface* (GUI) dan jaringan sesuai yang dibutuhkan untuk sistem. Struktur data juga perlu dirancang, arsitektur perangkat lunak, detil prosedur, dan karakteristik tampilan yang akan disajikan. Proses ini menterjemahkan kebutuhan sistem ke dalam model perangkat lunak yang nantinya dapat diperkirakan kualitasnya terlebih dahulu sebelum dilanjutkan ke tahap implementasi sistem.

c. *Implementation* (Coding)

Pada tahap sebelumnya rancangan yang telah dibuat akan diterjemahkan ke dalam suatu bahasa atau bentuk yang nantinya dapat dibaca dan diterjemahkan oleh komputer untuk proses olah. Pada tahap ini dapat disebut sebagai tahap implementasi, merupakan tahap untuk mengubah hasil perancangan sebelumnya menjadi sebuah bahasa pemrograman yang dapat dimengerti oleh komputer. Lalu, komputer akan menjalankan fungsi – fungsi yang telah dibuat sehingga komputer dapat memberikan *output* atau keluaran kepada penggunanya.

d. *Testing* (Pengujian)

Tahap ini dilakukan pengujian program bertujuan untuk mengetahui sesuai atau tidaknya sistem berjalan mengikuti prosedur dan memastikan sistem tidak mengalami *error*. Testing juga digunakan untuk mengetahui valid atau tidaknya proses mulai dari input sampai menjadi *output* yang sesuai. Untuk tahap ini ada 2 jenis metode pengujian sistem yaitu metode *black-box* dan *white-box*. Untuk pengujian dengan metode *black-box*, merupakan merupakan pengujian fokus terhadap fungsionalitas perangkat lunak tanpa perlu mengetahui struktur perangkat lunak yang dilakukan pengujian tersebut. Perangkat lunak yang dilakukan pengujian dengan menggunakan metode *black-box* ini dikatakan berhasil apabila fungsi – fungsi yang ada telah sesuai dengan spesifikasi kebutuhan yang telah dibuat sebelumnya. Untuk pengujian dengan metode *white-box*, yaitu menguji struktur internal perangkat lunak dengan cara melakukan pengujian pada algoritma yang digunakan oleh perangkat lunak.

e. *Maintenance* (Perawatan)

Tahap akhir dari SDLC *waterfall* ini adalah *maintenance* atau perawatan. Untuk tahap ini, apabila sistem sudah sesuai dengan tujuan yang telah ditentukan dan dapat menyelesaikan masalah yang terjadi pada perusahaan, kemudian akan diberikan kepada pengguna. Setelah sistem digunakan oleh pengguna, kemudian pada periode tertentu, akan diperlukan penyesuaian atau perubahan untuk mengikuti keadaan yang diinginkan, untuk itu dibutuhkan perubahan sistem tersebut. Tahap ini bisa dikatakan sebagai tahap oenggunaan perangkat lunak yang juga didampingi dengan perawatan dan perbaikan. Proses perawatan dan perbaikan diperlukan yang termasuk didalamnya melakukan pengembangan

perangkat lunak. Pengembangan diperlukan karena saat digunakan, perangkat lunak terkadang masih terdapat kekurangan maupun penambahan fitur – fitur yang sangat diperlukan.

