

BAB II

LANDASAN TEORI

2.1 Sistem Perhitungan Penjualan PT Panca Patriot Prima

Sistem penjualan pada PT Panca Patriot Prima memiliki rumus perhitungan sendiri mengenai proses transaksi penjualan, rumus perhitungan tersebut diantaranya :

a. Rumus Penentuan Penjualan

Menetapkan berapa jumlah penjualan ayam *broiler* yang harus dilakukan.

$$\text{Jumlah penjualan} = \text{jumlah kg ayam} \times \text{harga ayam}$$

b. Rumus Nota Timbang

$$\text{Netto} = \text{total ekor ayam} \times \text{kg ayam} - \text{berat keranjang}$$

$$\text{Berat Kotor} = \text{total ekor ayam} \times \text{kg} + \text{berat keranjang}$$

$$\text{Bekker} = \text{jumlah penimbangan ayam ekor} + \text{jumlah ayam ekor} \times \text{jumlah y berat keranjang}$$

Keterangan :

Bekker = berat keranjang

y = total berat keranjang

1.2 Sistem Penjualan

Penjualan ialah sebuah transaksi dilakukan dua pihak atau lebih antara satu orang ke orang lain dengan pembayaran yang sah yang menghasilkan suatu pendapatan atau kita jabarkan lagi secara luas pengertian penjualan ialah sebuah aktivitas yang bertujuan untuk mencari pembeli / mempengaruhi konsumen agar

terdapat suatu transaksi yang dilakukan oleh kedua pihak dengan alat pembayaran yang sah dan saling menguntungkan bagi kedua pihak tersebut. Pengertian penjualan merupakan suatu aktivitas yang berinteraksi secara langsung dengan konsumen untuk memperoleh pesanan atau penjualan secara langsung (Rhonda Abrams).

Sistem penjualan itu sendiri adalah suatu sistem yang saling berhubungan (*penjual dengan calon pembeli*) melalui prosedur-prosedur serta sarana pendukung untuk menghasilkan sebuah informasi pemesanan dan terjadinya suatu keseriusan dalam bertransaksi.

Sistem Penjualan Barang diartikan sebagai suatu sistem penjualan, melalui prosedur-prosedur yang meliputi urutan kegiatan sejak diterimanya pesanan dari pembeli, pengecekan barang ada atau tidak ada dan diteruskan dengan pengiriman barang yang disertai dengan pembuatan faktur dan mengadakan pencatatan atas penjualan yang berlaku (Niswonger, 1999).

1.2.1 Jenis – Jenis Penjualan

Menurut Basu Swastha pada buku "Manajemen Penjualan", terdapat beberapa jenis penjualan yang biasa dikenal dalam masyarakat diantaranya adalah:

- *Trade Selling*, penjualan yang terjadi bilamana produsen dan pedagang besar mempersilahkan pengecer untuk berusaha memperbaiki distribusi produk mereka,. Hal ini melibatkan para penyalur dengan kegiatan promosi, peragaan, persediaan dan produk baru, jadi titik beratnya adalah para penjualan melalui penyalur bukan pada penjualan ke pembeli akhir.

- *Missionary Selling*, penjualan berusaha ditingkatkan dengan mendorong pembeli untuk membeli barang dari penyalur perusahaan.
- *Technical Selling*, berusaha meningkatkan penjualan dengan pemberian saran dan nasihat kepada pembeli akhir dari barang dan jasa.
- *New Business Selling*, berusaha membuka transaksi baru dengan membuat calon pembeli menjadi pembeli seperti halnya yang dilakukan perusahaan asuransi.
- *Responsive Selling*, setiap tenaga penjual diharapkan dapat memberikan reaksi terhadap permintaan pembeli melalui route driving and retaining.

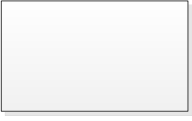


Jenis penjualan ini tidak akan menciptakan penjualan yang besar, namun terjalin hubungan pelanggan yang baik yang menjurus pada pembelian ulang.

1.3 Bagan Aliran Dokumen

Menurut Jogiyanto (2005 : 20) 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 *Flowchart*

No.	Simbol	Nama Simbol <i>Flowchart</i>	Fungsi

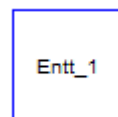
No.	Simbol	Nama Simbol <i>Flowchart</i>	Fungsi
1.		Dokumen	Untuk menunjukkan dokumen input dan output baik untuk proses manual, mekanik atau komputer.
2.		Proses Komputerisasi	Menunjukkan kegiatan dari operasi program komputer.
3.		Database	Untuk menyimpan data.
4.		Penghubung	Menunjukkan hubungan di 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 urutan angka.

1.4 Data Flow Diagram

Menurut Kendall (2003: 241), *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: 265), dalam memetakan *Data Flow Diagram*, terdapat beberapa simbol yang digunakan antara lain:

1. *External entity*

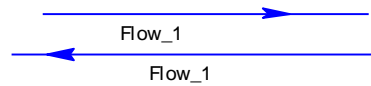
Suatu *external entity* atau entitas merupakan orang, kelompok, departemen, atau sistem lain di luar sistem yang dibuat dapat menerima atau memberikan informasi atau data ke dalam sistem yang dibuat.



Gambar 2.1 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 2.2. Simbol *Data Flow*

3. *Process*

Suatu proses dimana beberapa tindakan atau sekelompok tindakan dijalankan.



Gambar 2.3. Simbol *Process*

4. *Data Store*

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



Gambar 2.4. Simbol *Data Store*

1.5 Entity Relation Diagram

Entity relationship diagram (ERD) adalah gambaran pada sistem dimana di dalamnya terdapat hubungan antara *entity* beserta relasinya. *Entity* merupakan sesuatu yang ada dan terdefinisikan di dalam suatu organisasi, dapat abstrak dan nyata. Untuk setiap *entity* biasanya mempunyai *attribute* yang merupakan ciri *entity* tersebut. *Attribute* yaitu uraian dari entitas dimana mereka dihubungkan atau dapat dikatakan sebagai *identifier* atau *descriptors* dari entitas.

Entity Relationship Diagram ini diperlukan agar dapat menggambarkan hubungan antar *entity* dengan jelas, dapat menggambarkan batasan jumlah *entity* dan partisipasi antar *entity*, 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*

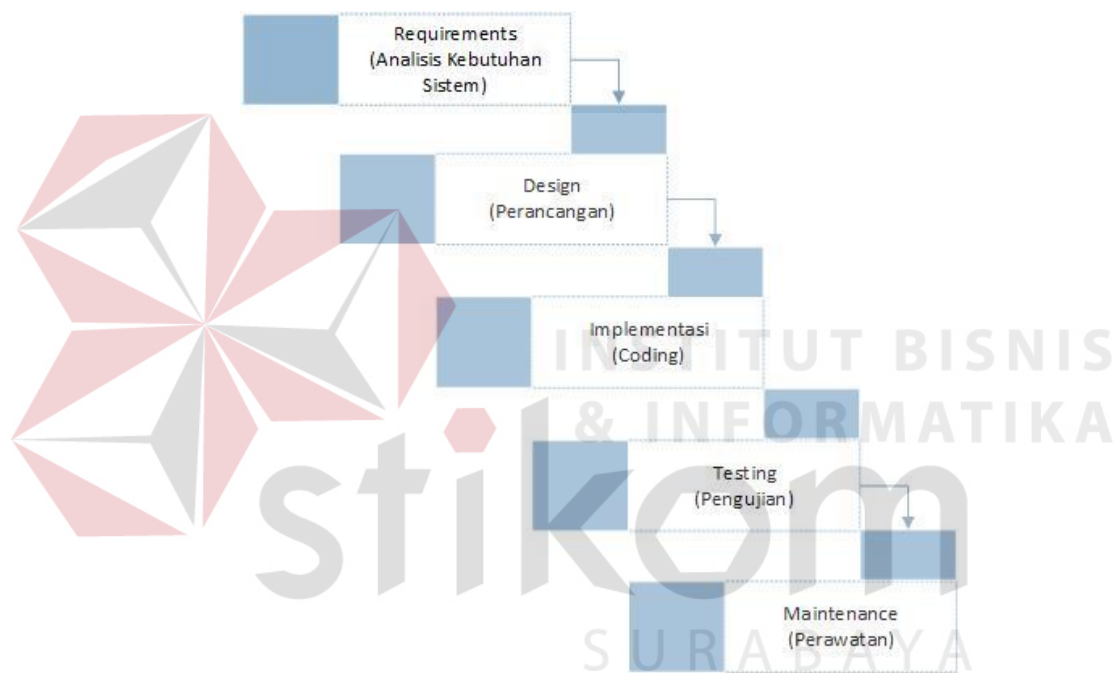
Physical Data Model (PDM) adalah jenis model data yang menggambarkan hubungan antar tabel secara fisikal.

1.6 System Development Life Cycle (SDLC)

Menurut Pressman (2001), 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 *Requirements* (analisis sistem),

Analysis(analisis kebutuhan sistem), *Design* (perancangan), *Coding* (implementasi), *Testing* (pengujian) dan *Maintenance* (perawatan).

Model eksplisit pertama dari proses pengembangan perangkat lunak, berasal dari proses-proses rekayasa yang lain. Model ini memungkinkan proses pengembangan lebih terlihat. Hal ini dikarenakan bentuknya yang bertingkat ke bawah dari satu fase ke fase lainnya, model ini dikenal dengan model *waterfall*, seperti terlihat pada gambar 6 berikut.



Gambar 2.5. *System Development Life Cycle* (SDLC) Model Waterfall

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

a. *Requirement* (Analisis Kebutuhan Sistem)

Pada tahap awal ini dilakukan analisa guna menggali secara mendalam kebutuhan yang akan dibutuhkan. Kebutuhan ada bermacam-macam seperti halnya kebutuhan informasi bisnis, kebutuhan data dan kebutuhan user itu sendiri. Kebutuhan itu sendiri sebenarnya dibedakan menjadi tiga jenis

kebutuhan. Pertama tentang kebutuhan teknologi. Dari hal ini dilakukan analisa mengenai kebutuhan teknologi yang diperlukan dalam pengembangan suatu sistem, seperti halnya data penyimpanan informasi / *database*. Kedua kebutuhan informasi, contohnya seperti informasi mengenai visi dan misi perusahaan, sejarah perusahaan, latar belakang perusahaan. Ketiga, Kebutuhan *user*. Dalam hal ini dilakukan analisa terkait 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. Dalam tahap ini diperhitungkan biaya yang akan dikeluarkan seperti biaya implementasi, *testing* dan *maintenance*.

b. *Design* (Perancangan)

Selanjutnya, hasil analisa kebutuhan sistem tersebut akan dibuat sebuah *design database*, DFD, ERD, antarmuka pengguna / *Graphical User Interface (GUI)* dan jaringan yang dibutuhkan untuk sistem. Selain itu juga perlu dirancang struktur datanya, arsitektur perangkat lunak, detail prosedur dan karakteristik tampilan yang akan disajikan. Proses ini menterjemahkan kebutuhan sistem ke dalam sebuah model perangkat lunak yang dapat diperkirakan kualitasnya sebelum memulai tahap implementasi.

c. *Implementation* (*Coding*)

Rancangan yang telah dibuat dalam tahap sebelumnya akan diterjemahkan ke dalam suatu bentuk atau bahasa yang dapat dibaca dan diterjemahkan oleh komputer untuk diolah. Tahap ini juga dapat disebut dengan tahap implementasi, yaitu tahap yang mengkonversi hasil perancangan sebelumnya ke dalam sebuah bahasa pemrograman yang dimengerti oleh komputer.

Kemudian komputer akan menjalankan fungsi-fungsi yang telah didefinisikan sehingga mampu memberikan layanan-layanan kepada penggunanya.

d. *Testing* (Pengujian)

Pengujian program dilakukan untuk mengetahui kesesuaian sistem berjalan sesuai prosedur ataukah tidak dan memastikan sistem terhindar dari *error* yang terjadi. *Testing* juga dapat digunakan untuk memastikan kevalidan dalam proses *input*, sehingga dapat menghasilkan *output* yang sesuai. Pada tahap ini terdapat 2 metode pengujian perangkat yang dapat digunakan, yaitu: metode *black-box* dan *white-box*. Pengujian dengan metode *black-box* merupakan pengujian yang menekankan pada fungsionalitas dari sebuah perangkat lunak tanpa harus mengetahui bagaimana struktur di dalam perangkat lunak tersebut. Sebuah perangkat lunak yang diuji menggunakan metode *black-box* dikatakan berhasil jika fungsi-fungsi yang ada telah memenuhi spesifikasi kebutuhan yang telah dibuat sebelumnya. Pengujian dengan menggunakan metode *white-box* yaitu menguji struktur internal perangkat lunak dengan melakukan pengujian pada algoritma yang digunakan oleh perangkat lunak.

e. *Maintenance* (Perawatan)

Tahap terakhir dari metode SDLC ini adalah *maintenance*. Pada tahap ini, jika sistem sudah sesuai dengan tujuan yang ditentukan dan dapat menyelesaikan masalah pada koperasi, maka akan diberikan kepada pengguna. Setelah digunakan dalam periode tertentu, pasti terdapat penyesuaian atau perubahan sesuai dengan keadaan yang diinginkan, sehingga membutuhkan perubahan terhadap sistem tersebut. Tahap ini dapat pula diartikan sebagai tahap penggunaan perangkat lunak yang disertai dengan perawatan dan perbaikan.

Perawatan dan perbaikan suatu perangkat lunak diperlukan, termasuk didalamnya adalah pengembangan, karena dalam prakteknya ketika perangkat lunak digunakan terkadang masih terdapat kekurangan ataupun penambahan fitur-fitur baru yang dirasa perlu.

1.7 Aplikasi

Aplikasi adalah sebuah perangkat lunak yang menjadi *front end* dalam sebuah sistem yang digunakan untuk mengolah data menjadi suatu informasi yang berguna orang-orang dan sistem yang bersangkutan. (Sri Widianti : 2000). Istilah aplikasi berasal dari bahasa Inggris “*application*” yang berarti penerapan, lamaran ataupun penggunaan. Sedangkan secara istilah, pengertian aplikasi adalah suatu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju.

Menurut Whitten, Perancangan Sistem adalah “Proses dimana keperluan pengguna dirubah ke dalam bentuk paket perangkat lunak dan atau kedalam spesifikasi pada computer yang berdasarkan pada sistem informasi” (Kristanto, 1994)

Menurut kamus komputer eksekutif, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputansi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan. Adapun penggolongan mengenai aplikasi, sebagai berikut :

- a. Aplikasi *enterprise*
- b. Aplikasi infrastruktur
- c. Aplikasi informasi kerja
- d. Aplikasi media dan hiburan
- e. Aplikasi Pendidikan
- f. Aplikasi pengembangan media
- g. Aplikasi Rekayasa Produk

