

BAB II

LANDASAN TEORI

2.1 Sistem

Menurut Jogiyanto (2010) Sistem dapat didefinisikan dengan dua jenis pendekatan yaitu pendekatan komponen dan pendekatan prosedur. Jika didefinisikan dengan menggunakan pendekatan komponen sistem merupakan kumpulan dari beberapa komponen yang membentuk suatu kesatuan yang bergerak untuk suatu tujuan yang sama. Berikut contoh dari sistem yang didefinisikan secara komponen yaitu, sebuah sistem komputer yang terdiri dari komponen *software* dan *hardware* yang saling bekerja sama.

Kemudian penjelasan dari sistem apabila didefinisikan dengan menggunakan pendekatan prosedur yaitu sebuah sistem merupakan kumpulan dari beberapa prosedur yang memiliki tujuan tertentu. Contoh dari sistem jika didefinisikan menggunakan pendekatan prosedur yaitu, sebuah sistem akuntansi yang terdiri dari kumpulan-kumpulan prosedur akuntansi seperti prosedur penerimaan kas, prosedur pengeluaran kas, penjualan dan pembelian.

2.2 Informasi

Menurut Jogiyanto (2010) Informasi adalah sebuah data atau fakta-fakta yang telah diolah menjadi bentuk yang berguna untuk penggunanya. Dalam membentuk sebuah informasi terdapat tiga aturan yang digunakan yaitu relevan (*relevance*) yang dapat dilihat dari seberapa tepat informasi tersebut diberikan oleh penggunanya, lalu *timelines* yang dapat dilihat dari ketepatan waktu dari

informasi yang disajikan, dan yang terakhir adalah akurat (*accurate*) yang dapat dilihat dari tingkat kegunaan informasi tersebut bagi penggunanya.

2.3 Sistem Informasi

Menurut Jogiyanto (2010) Sistem Informasi adalah suatu sistem di dalam organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategis dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan

2.4 Access Point

Menurut Syamsudin (2010) *Access point* merupakan perangkat *wireless* yang berfungsi sebagai pusat akses jaringan yang biasa dikenal juga sebagai *wireless router*. *Access point* yang digunakan di Telkom ada jenis *indoor* dan *outdoor*. Perbedaan pada *Access Point* untuk *indoor* dan *outdoor* memiliki perbedaan yang sangat signifikan yaitu pada *outdoor* memiliki daya dan jangkauan radio yang lebih luas dibanding *indoor*.

Menurut Purbo (2006), dalam membangun sebuah *Internet Service Provider* (ISP) dengan menggunakan *access point* dapat digunakan untuk memberikan *service* pada *client wireless*. Pada dasarnya *access point* memiliki fungsi sebagai *bridge* antara jaringan *wireless* dan jaringan kabel LAN.

Access point memiliki prinsip kerja seperti *switch* atau *hub* yang digunakan untuk jaringan berbasis kabel. Dari kedua perangkat keras tersebut yang membedakan adalah dalam melakukan transmisi data *switch* dan *hub*

menggunakan kabel UTP, sedangkan *Access point* menggunakan gelombang radio pada medium udara.

2.5 Persediaan

Menurut Rangkuti (2007) Persediaan merupakan aktiva yang meliputi barang-barang milik perusahaan dengan maksud untuk dijual dalam suatu periode usaha tertentu, atau persediaan barang-barang yang masih dalam pengerjaan/proses produksi, ataupun persediaan bahan baku yang menunggu penggunaannya dalam suatu produksi. Persediaan merupakan salah satu unsur paling aktif dalam operasi perusahaan yang secara kontinu diperoleh, diubah, kemudian dijual kembali. Persediaan memiliki berbagai jenis, menurut Rangkuti (2007) jenis-jenis persediaan menurut fungsinya dibagi menjadi tiga, yaitu :

1. *Batch Stock/Lot Size Inventory*

Persediaan yang diadakan karena kita membeli atau membuat bahan-bahan atau barang-barang dalam jumlah yang lebih besar daripada jumlah yang dibutuhkan saat itu.

Keuntungannya:

- a. Potongan harga pada harga pembelian
- b. Efisiensi produksi
- c. Penghematan biaya angkutan

2. *Fluctuation Stock*

Persediaan yang diadakan untuk menghadapi fluktuasi permintaan konsumen yang tidak dapat diramalkan

3. *Anticipation Stock*

Persediaan yang diadakan untuk menghadapi fluktuasi permintaan yang dapat diramalkan, berdasarkan pola musiman yang terdapat dalam satu tahun dan untuk menghadapi penggunaan, penjualan, atau permintaan yang meningkat.

2.6 *Reorder Point*

Menurut Martono (2013) *Reorder point* yaitu titik jumlah pemesanan kembali, dimana pesanan dilakukan apabila persediaan yang ada telah mencapai suatu titik tertentu. *Reorder Point* terjadi apabila jumlah persediaan yang terdapat di dalam stok berkurang terus. Dengan demikian kita harus menentukan berapa banyak batas minimal tingkat persediaan yang harus dipertimbangkan sehingga tidak terjadi kekurangan persediaan. Jumlah yang diharapkan tersebut dihitung selama dalam waktu tenggang, mungkin dapat juga ditambah dengan *safety stock* yang biasanya mengacu pada kemungkinan terjadinya kekurangan stok pada masa tenggang.

Berikut adalah persamaan dari Reorder point :

$$ROP = d \times LT + SS$$

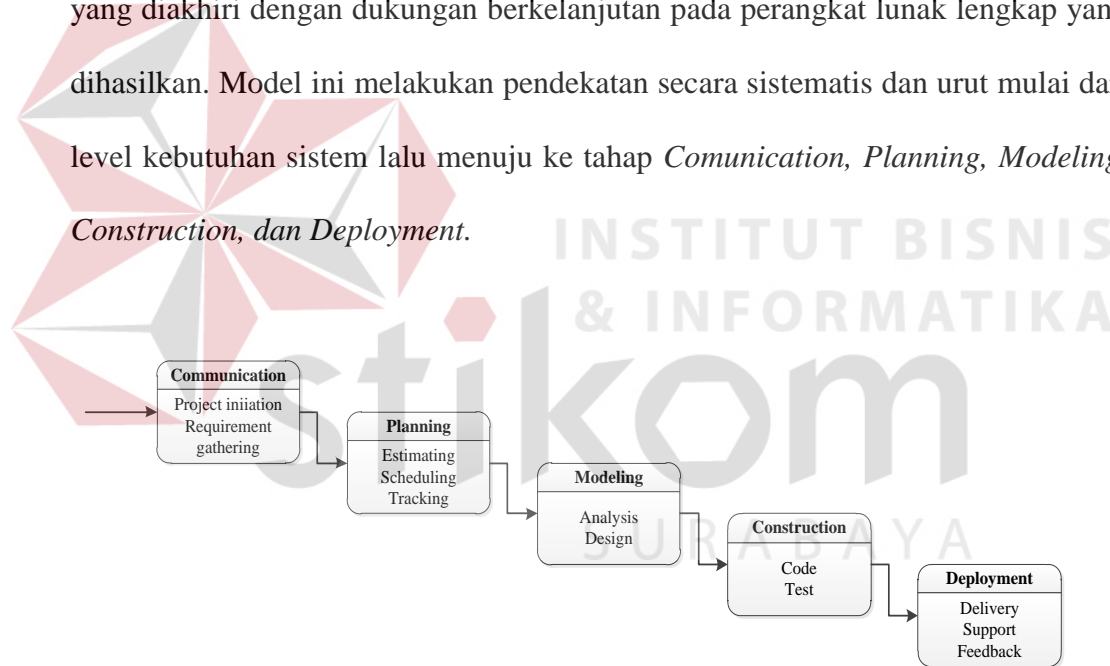
d = kebutuhan konstan

LT = Lead Time

SS (*Safety Stock*) = batas titik aman persediaan

2.7 System Development Life Cycle (SDLC)

Menurut Pressman (2010) model SDLC biasa disebut juga dengan model *waterfall* atau disebut juga *classic life cycle*. Adapun pengertian dari SDLC ini adalah suatu pendekatan yang sistematis dan berurutan pada pengembangan perangkat lunak. SDLC memiliki tahapan-tahapan yang dimulai dengan spesifikasi kebutuhan pengguna dan berlanjut melalui tahapan-tahapan perencanaan (*planning*), pemodelan (*modeling*), konstruksi (*construction*), serta penyerahan sistem perangkat lunak ke para pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan berkelanjutan pada perangkat lunak lengkap yang dihasilkan. Model ini melakukan pendekatan secara sistematis dan urut mulai dari level kebutuhan sistem lalu menuju ke tahap *Communication, Planning, Modeling, Construction, dan Deployment*.



Gambar 2.1 Pengembangan Model SDLC (Pressman, 2010)

Model ini banyak digunakan dalam *Software Engineering*. Pada Gambar 2.1 menunjukkan tahapan umum dari model waterfall karena setiap tahapnya harus menunggu tahap sebelumnya selesai dan dijalankan dengan berurutan.

Berikut ini adalah penjelasan dari tahap-tahap yang dilakukan di dalam Model *waterfall* menurut Pressman :

a. *Communication*

Langkah pertama diawali dengan komunikasi kepada pengguna. Langkah awal ini bertujuan untuk pengumpulan informasi tentang kebutuhan pengguna.

b. *Planning*

Setelah proses *communication* dilanjutkan dengan tahap selanjutnya yaitu *planning*. Langkah ini dilakukan untuk merencanakan tugas-tugas teknis yang akan dilakukan, resiko yang mungkin terjadi, sumber yang dibutuhkan, hasil yang akan dibuat, dan jadwal pengerjaan.

c. *Modeling*

Pada proses *modeling* ini menerjemahkan syarat kebutuhan perangkat lunak kedalam bentuk desain yang dapat diperkirakan sebelum dilakukan *coding*.

Pada tahap ini yang umum dikerjakan adalah rancangan struktur data, arsitektur *software*, representasi *interface*, dan *detail* (algoritma) prosedural.

d. *Construction*

Pada tahap *construction* merupakan proses pengkodean atau yang biasa disebut *coding*. *Coding* atau pengkodean merupakan penerjemahan desain dalam bahasa yang bisa dikenali oleh komputer. Seseorang yang bertanggung jawab atas proses ini ada *programmer*. *Programmer* akan melakukan *coding* sesuai kebutuhan yang diminta oleh pengguna.

e. *Deployment*

Pada tahap ini dapat dikatakan final dalam pembuatan sebuah sistem. Setelah melakukan analisis, desain dan pengkodean maka sistem yang sudah jadi akan digunakan pengguna.

2.8 Testing

Menurut Romeo (2003) testing adalah proses pemantapan kepercayaan akan kinerja program atau sistem sebagaimana yang diharapkan. Testing *Software* adalah proses mengoperasikan *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 sebagai berikut:

1. Untuk melakukan testing kesesuaian suatu komponen terhadap desain *White Box Testing*.
2. Untuk melakukan testing kesesuaian suatu komponen terhadap spesifikasi *Black Box Testing*.

2.8.1 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 diasosiasikan 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 direpresentasikan dengan *flowgraph*.

2.8.2 *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 fungsional testing. *Black box* testing berfokus pada kebutuhan fungsional pada *software*, berdasarkan pada spesifikasi kebutuhan dari *software*.

Dengan adanya *black box* testing, perancang *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.