

BAB II

LANDASAN TEORI

2.1 Pengenalan Proses Produksi di Tianjin Sunshine Plastic Co., Ltd

Tianjin Sunshine Plastic Co., Ltd merupakan perusahaan manufaktur yang memproduksi plastik. Plastik yang dijual ke *customer* dikemas dalam satuan roll yang biasa disebut dengan *Slit Roll* (SR), seperti pada gambar 1. *Customer* dapat memesan SR dengan tipe film, ketebalan, panjang dan lebar sesuai dengan keinginan.



Gambar 2.1 *Slit Roll* yang Dijual ke *Customer*

2.1.1 Jenis Barang

Ada 5 jenis barang yang ada di Tianjin Sunshine Plastic Co., Ltd (Amaliyah dan Harvianto, 2014), yaitu:

1. *Raw Material*

Raw material (RM) adalah bahan baku yang digunakan untuk memproduksi plastik. RM yang paling banyak digunakan adalah biji plastik. Masing-masing biji plastik mempunyai tipe berbeda. Total RM yang dimiliki Tianjin

Sunshine Plastic Co., Ltd adalah 53 jenis RM.

2. *Jumbo Roll (Semi Finished Goods)*

RM diolah menjadi plastik. Plastik yang diproduksi adalah plastik yang berukuran besar yang lebarnya mencapai 6 meter. Plastik ini disebut dengan *jumbo roll* (JR). JR masih merupakan barang setengah jadi yang nantinya akan diolah lagi.



(Trias Sentosa Tbk., 2013)

Gambar 2.2 *Jumbo Roll*

3. *Slit Roll (Finished Goods)*

Slit Roll (SR) adalah barang jadi dan nantinya dijual kepada *customer*. SR diperoleh dari JR yang telah melalui proses pemotongan. JR dipotong menjadi SR sesuai dengan ukuran yang dipesan oleh *customer*. SR dibedakan menjadi 2, yaitu:

a. *SR Free stock*

SR yang belum terikat oleh pesanan *customer*.

b. *SR SO Stock*

SR yang sudah terikat oleh pesanan *customer*. Saat *customer* memesan, pesanan tersebut dicatat dalam SO oleh bagian *sales*, dan diberi nomor

SO. Oleh karena itu SR yang sudah terikat pesanan dicantumkan nomor SOnya.

c. *Raw Recycle*

Raw Recycle (RR) adalah barang yang dapat didaur ulang. Dalam proses produksi ada sisa-sisa plastik yang tidak terpakai atau tidak terpotong. Sisa-sisa tersebut disebut RR. JR atau SR yang cacat/kualitasnya jelek juga dimasukkan kedalam kategori RR.

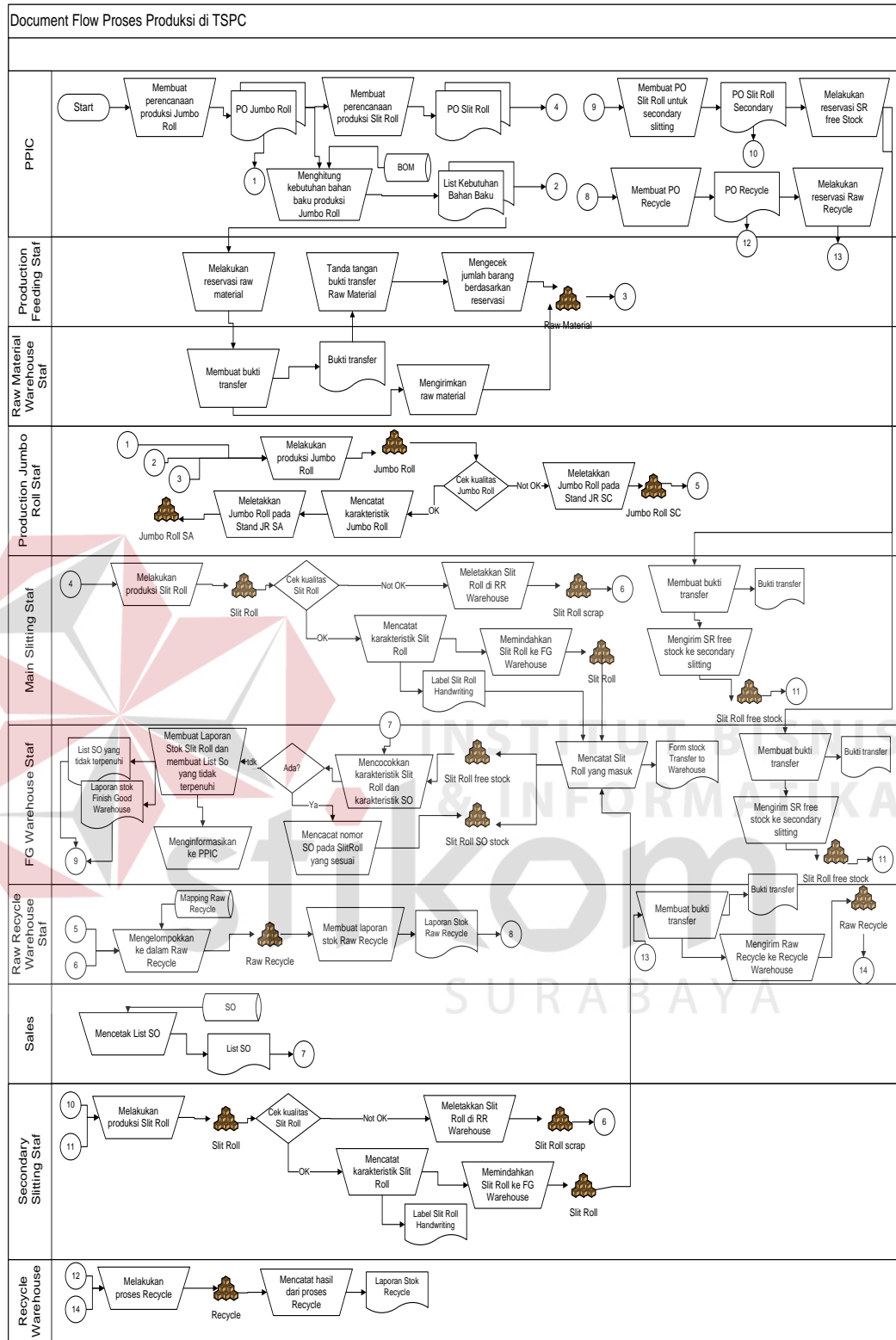
d. *Recycle*

Recycle (RC) adalah RR yang sudah diproses melalui proses *recycle*. Barang jenis RC ini dapat digunakan kembali untuk memproduksi JR.

2.1.2 Warehouse

Ada 5 jenis *warehouse* yang ada di Tianjin Sunshine Plastic Co., Ltd (Amaliyah dan Harvianto, 2014), yaitu:

1. *RM warehouse*, digunakan untuk menyimpan RM.
2. *Stand JR*, digunakan untuk menyimpan JR. Ada 46 *stand JR SA* dan 1 *stand JR SC* yang ada di Tianjin Sunshine Plastic Co., Ltd.
3. *Finished goods warehouse*, digunakan untuk menyimpan SR, baik *SR free stock* maupun *SR SO stock*.
4. *RR warehouse*, digunakan untuk menyimpan RR.
5. *RC warehouse*, digunakan untuk proses *recycle* dan menyimpan material hasil proses *recycle*.



Gambar 2.4 Document Flow Proses Produksi di TSPC Co., Ltd.

Staf PPIC membuat perencanaan produksi JR. Perencanaan produksi JR diperoleh melalui rapat PPIC tiap akhir bulan. Dari hasil rapat diperoleh PO (*Process Order*) JR untuk bulan depan. Setelah PO JR didapat, Staf PPIC membuat perencanaan produksi untuk SR. JR dipotong menjadi beberapa bagian untuk menjadi SR. PPIC menentukan kombinasi pemotongan JR menjadi SR menggunakan aplikasi *Calculator Normal*. Aplikasi tersebut berfungsi untuk menentukan kombinasi potong JR menjadi SR. Hasil dari aplikasi *Calculator Normal* disebut PO SR.

Untuk memproduksi JR, staf PPIC harus menghitung kebutuhan bahan baku untuk proses produksi berdasarkan dari PO JR yang sudah ada. Kebutuhan bahan baku meliputi komponen RM dan jumlahnya. Data tersebut diperoleh dari data *Bill of Material* (BOM) yang sudah disimpan oleh PPIC. Berdasarkan dokumentasi *user requirement* dari pihak TSPC, rumus untuk menghitung kebutuhan bahan baku adalah:

$$Qty \text{ yang dibutuhkan} = Qty \text{ BOM} \times \frac{\text{panjang JR}}{1000} \times \frac{\text{lebar JR}}{1000} \dots (1)$$

Rumus 1. Menghitung Kebutuhan Bahan Baku

Berdasarkan perhitungan tersebut, diperoleh *list* kebutuhan bahan baku yang diberikan kepada *production feeding staff* dan *production JR staff*.

Production feeding staff melakukan reservasi RM kepada RM *warehouse staff* berdasarkan *list* kebutuhan bahan baku yang diberikan oleh staf PPIC. RM *warehouse staff* membuat bukti transfer kemudian mengirimkan RM beserta bukti *transfERNYA* kepada *production feeding staff* untuk dicek dan ditandatangani

sebagai persetujuan bahwa barang yang dipesan sudah diterima. RM siap digunakan untuk proses produksi oleh *production JR staff*.

Production JR staff melakukan proses produksi JR berdasarkan PO JR dan *list* kebutuhan bahan baku yang diberikan oleh staf PPIC. Setelah JR terbentuk, JR di cek kualitasnya. Jika tidak memenuhi syarat, maka JR tersebut diletakkan pada *stand* JR SC. Jika kualitas JR memenuhi syarat, maka *production JR staff* mencatat karakteristik JR. Contoh karakteristik yang dicatat adalah sebagai berikut:

Tabel 2.1 Karakteristik JR

Description	Value
<i>Density</i> (g/Cm ³)	0.91
<i>Film Code</i>	AF30-20
<i>Film Type</i>	AF30
<i>Film Width</i> (milimeter)	6620
<i>Grammage</i>	18.2
<i>Inside</i>	CO
<i>Length</i> (Meter)	25264
<i>Month/Year of Production</i>	104
<i>Outside</i>	CO
<i>Product Line</i>	A
<i>Roll Number</i>	GA84A
<i>Sequence Number</i>	10
<i>Speed</i> (M/Sec)	200
<i>Status</i>	SA
<i>Thickness Film</i> (Micron)	20
<i>Treatment</i>	COCO
<i>Weight</i> (Kg)	3043.9

(Amaliyah dan Harvianto, 2014)

Setelah karakteristik JR dicatat, JR tersebut diletakkan pada *stand* JR SA. JR SA dipotong menjadi SR berdasarkan PO SR yang sudah dibuat oleh PPIC. SR yang sudah terbentuk di cek kualitasnya. Jika kualitas tidak memenuhi syarat, maka akan diletakkan ke RR *warehouse*. Jika kualitas memenuhi syarat, maka

main slitting staff akan menulis label. Label yang ditulis meliputi: tipe film, panjang, lebar, ketebalan, berat, kode produksi atau disebut juga dengan Nomor *Roll*, *grade*, *criteria grade*, nama *customer*, nomor SO, dan nomor SO *Item*. Nomor *Roll* terdiri dari 10 digit. Aturan pemberian Nomor *Roll* adalah sebagai berikut:

1 Production Month		2 Production Line		3 Production Date		4 Production Year		5 JR Number		6 MS Position		7 MS Slitting		8 SS Position		9 SS Slitting		10
Code	Meaning	Code	Meaning	Code	Meaning	Code	Meaning	Code	Meaning	Code	Meaning	Code	Meaning	Code	Meaning	Code	Meaning	berapa kali terpotong dr MS
T	January	A	Line 1	1	1	1	xxxx1	1	1	1	1	1	1	0	MS	1	1	
U	February			2	2	2	xxxx2	2	2	2	2	2	2	1	1	2	2	
Y	March			3	3	3	xxxx3	3	3	3	3	3	3	2	2	3	3	
A	April			4	4	4	xxxx4	4	4	4	4	4	4	3	3	4	4	
B	May			5	5	5	xxxx5	5	5	5	5	5	5	4	4	5	5	
C	June			6	6	6	xxxx6	6	6	6	6	6	6	5	5	6	6	
D	July			7	7	7	xxxx7	7	7	7	7	7	7	6	6	7	7	
E	August			8	8	8	xxxx8	8	8	8	8	8	8	7	7	8	8	
F	September			9	9	9	xxxx9	9	9	9	9	9	9	8	8	9	9	
G	October			A	10	0	xxxx0	A	10	A	10	A	10	9	9	A	10	
K	November			B	11			B	11	B	11			A	10			
L	December			C	12			C	12	C	12							
				D	13			D	13									
				E	14			E	14									
				F	15			F	15									
				G	16			G	16									
				H	17			H	17									
				J	18			J	18									
				K	19			K	19									
				L	20			L	20									
				M	21													
				N	22													
				P	23													
				Q	24													
				R	25													
				S	26													
				T	27													
				W	28													
				X	29													
				Y	30													
				Z	31													

(Amaliyah dan Harvianto, 2014)

Gambar 2.5 Aturan Pemberian Nomor *Roll* pada SR

Label tersebut ditempel pada SR. Kemudian SR dipindahkan ke *finished goods warehouse*.

Finished goods warehouse staff mencatat SR yang masuk ke *finished goods warehouse* pada *form stock transfer to warehouse*. SR yang masuk ke *finished goods warehouse* dibedakan menjadi SR *free stock* dan SR *SO Stock*.

SR *free stock* dapat berubah menjadi SR *SO Stock* saat staf dari departemen *sales* memberikan *list SO* yang harus dipenuhi kepada *finished goods warehouse*. *Finished goods warehouse staff* mencari dan mencocokkan karakteristik SR dan karakteristik SO. Jika SR yang dicari tersedia, maka *finished goods warehouse staff* menambahkan nomor SO tersebut ke SR yang sesuai, dan status SR tersebut berubah menjadi SR *SO Stock*. Jika SR yang dicari tidak tersedia, maka *finished goods warehouse staff* membuat laporan SR yang ada saat ini dan membuat *list SO* yang belum terpenuhi kemudian menginformasikannya ke PPIC.

Berdasarkan laporan dari pihak *finished goods warehouse* mengenai SO yang belum terpenuhi dikarenakan tidak adanya SR *free stock* yang karakteristiknya memenuhi SO tersebut, PPIC membuat PO untuk *secondary slitting*. Untuk membuat PO *secondary slitting*, PPIC mencari SR *free stock* di *main slitting* atau *finished goods warehouse* yang tipe film, ketebalan, *density*, *grade*, dan *criteria grade* yang sama dengan SO yang belum terpenuhi. Kemudian menentukan kombinasi potong dengan bantuan *calculator normal*. Setelah PO *secondary slitting* dibuat, PPIC memberikan PO tersebut ke *secondary slitting staff* dan melakukan reservasi SR *free stock* kepada *main slitting* atau *finished goods warehouse staff* untuk dikirim ke ruang *secondary slitting*. *Secondary staff* melakukan proses produksi SR sesuai dengan PO yang dibuat PPIC. Proses produksi yang dilakukan sama dengan proses di *main slitting*. Jika SR mempunyai

kualitas yang jelek, maka akan dijadikan SR *scrap* dan diletakkan di RR *warehouse*.

Berdasarkan JR SC dan SR *scrap* yang diletakkan di RR *warehouse*, petugas RR *warehouse* harus memilah-milah berdasarkan jenis RR dan membuat laporan stoknya untuk diserahkan kepada PPIC. Jika dirasa stok RR sudah terlalu penuh, maka petugas PPIC membuat PO *recycle* untuk mengolah RR menjadi RC. PO *recycle* tersebut diserahkan kepada *recycle warehouse staff* dan melakukan reservasi RR kepada *raw recycle staff* untuk diserahkan kepada pihak *recycle warehouse*. Setelah PO *recycle* dan RR diterima, *recycle staff* melakukan proses *recycle* dan mencatatnya.

2.2 Gudang (Warehouse)

Gudang adalah bangunan yang dipergunakan untuk menyimpan barang dagangan. Gudang adalah suatu tempat yang digunakan untuk menyimpan barang baik berupa RM, barang *work in process* atau *finished goods* (Warman, 2004).

Menurut Warman (2004), jenis gudang dibedakan menjadi:

a. Gudang operasional

Gudang dimana bahan baku disimpan. Dalam gudang ini dapat pula disimpan barang setengah jadi, atau suku bagian atau barang dalam proses.

b. Gudang perlengkapan

Dapat berupa bengkel tambahan yang diletakkan dekat proses produksi untuk menyediakan perkakas kerja, barang pelumas atau barang lainnya yang diperlukan dalam proses produksi.

c. Gudang pemberangkatan

Merupakan ruang penyimpanan dari bagian pengiriman, dimana barang-barang itu disimpan sebelum diberangkatkan dari pabrik.

d. Gudang musiman

Dalam industri tertentu terkadang diperlukan sediaan barang yang harus disimpan dalam jumlah banyak sehingga harus menyewa ruangan.

2.3 Warehouse Management System

Menurut Muehlbauer (2006), *Warehouse Management System* (WMS) adalah sebuah konsep pergudangan untuk mengelola barang-barang yang ada di gudang. Manajemen pergudangan dirancang bertujuan untuk mengontrol kegiatan pergudangan atau logistik. Yang diharapkan dari pengontrolan ini adalah terjadinya pengurangan biaya-biaya yang ada pada gudang atau logistik, pengambilan dan pemasukan barang ke gudang yang efektif dan efisien, serta kemudahan dan keakuratan informasi *stock* barang di gudang.

Ada dua elemen kunci dari operasi sistem manajemen gudang:

1. Karyawan gudang (operator) menggunakan komputer untuk mencatat pekerjaan secara *real time*. Kuncinya di sini adalah secara *real time*. Kegiatan yang dilakukan oleh operator dicatat ke dalam komputer segera, tidak ditulis di atas kertas dan dicatat dalam komputer nanti
2. *Barcode* digunakan untuk mengurangi jumlah informasi yang harus dimasukkan langsung oleh operator ke terminal komputer. Operator melakukan *scanning* label yang terdapat *barcode* dimana *barcode* tersebut merepresentasikan *item numbers*, *lot numbers*, kuantitas dan banyak lagi yang memungkinkan untuk menghilangkan penekanan tombol.

Menggunakan teknologi *barcode* secara *real time* memungkinkan komputer untuk memvalidasi pekerjaan yang dilakukan oleh operator, memberikan konfirmasi positif jika proses berhasil dilakukan, dan memberikan konfirmasi *error* jika terjadi kesalahan (Muehlbauer, 2006).

2.4 Pergerakan Barang (*Goods Movements*)

Pergerakan barang (*goods movements*) disebut juga dengan *material movement* atau *stock movement* adalah suatu peristiwa yang menyebabkan perubahan pada stok. Pergerakan barang tersebut meliputi:

1. *Goods Receipt*

Adalah penerimaan barang yang terjadi karena adanya penerimaan barang dari *vendor* atau dari hasil produksi. *Goods receipt* mengarah pada peningkatan stok di gudang.

2. *Goods Issue*

Adalah pergerakan barang yang terjadi karena adanya proses penarikan material, konsumsi bahan, atau pengiriman barang ke pelanggan. *Goods issue* mengarah pada penurunan stok di gudang.

3. *Stock Transfer*

Adalah pemindahan material dari suatu lokasi penyimpanan ke lokasi penyimpanan yang lain. *Stock transfer* ini dapat terjadi dalam *plant* yang sama atau *plant* lain.

4. *Transfer Posting*

Transfer posting adalah istilah umum untuk *stock transfer* yang menyebabkan perubahan pada tipe stok atau kategori stok dari material. Contoh dari *transfer*

posting adalah *transfer posting material to material* dan *release material* dari proses *quality inspection* (Murray dan Akhtar, 2016).

2.5 Persediaan (*Inventory*)

Menurut Pujawan dan Mahendrawati (2010), persediaan (*inventory*) didefinisikan sebagai sumber daya yang disimpan untuk memenuhi permintaan saat ini maupun yang akan datang.

Persediaan dapat diklasifikasikan dengan berbagai cara, antara lain:

1. Berdasarkan bentuknya, persediaan bisa diklasifikasikan menjadi bahan baku (RM), barang setengah jadi (*work in proses*), dan produk jadi (*finished product*).
2. Berdasarkan fungsinya, persediaan bisa dibedakan menjadi:
 - a. *Pipeline/transit inventory*, persediaan ini muncul karena *lead time* pengiriman dari suatu tempat ke tempat lain.
 - b. *Cycle stock*, ini adalah persediaan akibat motif memenuhi skala ekonomi, Persediaan ini mempunyai siklus tertentu.
 - c. Persediaan pengaman (*safety stock*), fungsinya adalah sebagai perlindungan terhadap ketidakpastian permintaan maupun pasokan.
 - d. *Anticipation stock* adalah persediaan yang dibutuhkan untuk mengantisipasi kenaikan permintaan akibat sifat musiman dari permintaan terhadap produk.
3. Persediaan juga bisa diklasifikasikan berdasarkan sifat ketergantungan kebutuhan antara satu item dengan item lainnya, yaitu:
 - a. *Dependent demand* : Item-item yang kebutuhannya bergantung pada item yang lain. *Dependent demand* merupakan komponen atau bahan baku

untuk membuat *independent demand* yang diwujudkan dalam BOM (*Bill of Material*).

- b. *Independent demand* : Item tidak bergantung pada kebutuhan item yang lain.

2.6 Sistem Informasi

Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. 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. Sedangkan berdasarkan pendekatan komponen, sistem merupakan kumpulan dari komponen-komponen yang saling berkaitan untuk mencapai tujuan tertentu.

Menurut West Churman dalam buku yang ditulis oleh Krismiaji (2002), “sebuah sistem dapat didefinisikan sebagai serangkaian komponen yang dikoordinasikan untuk mencapai serangkaian tujuan”. Informasi adalah data yang telah diproses menjadi bentuk yang memiliki arti bagi penerima dan dapat berupa fakta, suatu nilai yang bermanfaat. Jadi ada suatu proses transformasi data menjadi suatu informasi yaitu *input*, proses, *output*.

Menurut Herlambang dan Tanuwijaya (2005), data adalah fakta-fakta atau kejadian-kejadian yang dapat berupa angka-angka atau kode-kode tertentu. Data masih belum mempunyai arti bagi penggunanya. Untuk dapat mempunyai arti data diolah sedemikian rupa sehingga dapat digunakan oleh penggunanya. Hasil pengolahan data inilah yang disebut sebagai informasi. Secara ringkas, Informasi

adalah data yang telah diolah dan mempunyai arti bagi penggunanya. Sehingga sistem informasi dapat didefinisikan sebagai prosedur-prosedur yang digunakan untuk mengolah data sehingga dapat digunakan oleh penggunanya. Data merupakan *raw material* untuk suatu informasi. Perbedaan informasi dan data sangat relatif tergantung pada nilai gunanya bagi manajemen yang memerlukan. Suatu informasi bagi level manajemen tertentu bisa menjadi data bagi manajemen level di atasnya, atau sebaliknya. Sedangkan pengertian Sistem informasi itu sendiri adalah suatu sistem terintegrasi yang mampu menyediakan informasi yang bermanfaat bagi penggunanya.

Sebuah sistem terintegrasi atau sistem manusia-mesin, untuk menyediakan informasi untuk mendukung operasi, manajemen dalam suatu organisasi. Sistem ini memanfaatkan perangkat keras dan perangkat lunak komputer, prosedur manual, model manajemen dan basis data.

2.7 Agile Software Development

Menurut Proboyekti (2008), kata *Agile* berarti bersifat cepat, ringan, bebas bergerak, waspada. Kata ini digunakan sebagai kata yang menggambarkan konsep model proses yang berbeda dari konsep model-model proses yang sudah ada. Konsep *Agile Software Development* dicetuskan oleh Kent Beck dan 16 rekannya. Dalam *Agile Software Development* interaksi dan personel lebih penting dari pada proses dan alat, *software* yang berfungsi lebih penting daripada dokumentasi yang lengkap, kolaborasi dengan klien lebih penting dari pada negosiasi kontrak, dan sikap tanggap terhadap perubahan lebih penting daripada mengikuti rencana. Namun demikian, sama seperti model proses yang lain, *Agile Software Development* memiliki kelebihan dan tidak cocok untuk semua jenis proyek,

produk, orang dan situasi. *Agile Software Development* memungkinkan model proses yang toleransi terhadap perubahan kebutuhan sehingga perubahan dapat cepat ditanggapi. Namun di sisi lain menyebabkan produktifitas menurun.

Menurut *Agile Alliance* dalam artikel yang ditulis oleh Proboyekti (2008), 12 prinsip ini adalah bagi mereka yang ingin berhasil dalam penerapan *Agile Software Development*:

1. Kepuasan klien adalah prioritas utama dengan menghasilkan produk lebih awal dan terus menerus.
2. Menerima perubahan kebutuhan, sekalipun diakhir pengembangan.
3. Penyerahan hasil/*software* dalam hitungan waktu dua minggu sampai dua bulan.
4. Bagian bisnis dan pembangun kerja sama tiap hari selama proyek berlangsung
5. Membangun proyek dilingkungan orang-orang yang bermotivasi tinggi yang bekerja dalam lingkungan yang mendukung dan yang dipercaya untuk dapat menyelesaikan proyek.
6. Komunikasi dengan berhadapan langsung adalah komunikasi yang efektif dan efisien
7. *Software* yang berfungsi adalah ukuran utama dari kemajuan proyek
8. Dukungan yang stabil dari sponsor, pembangun, dan pengguna diperlukan untuk menjaga perkembangan yang berkesinambungan
9. Perhatian kepada kehebatan teknis dan desain yang bagus meningkatkan sifat *Agile*
10. Kesederhanaan penting

11. Arsitektur, kebutuhan dan desain yang bagus muncul dari tim yang mengatur dirinya sendiri
12. Secara periodik tim evaluasi diri dan mencari cara untuk lebih efektif dan segera melakukannya.

Extreme Programming adalah salah satu model pengembangan dari *Agile Software Development*. *Extreme Programming* menggunakan pendekatan berorientasi objek. Ada empat tahapan dalam model pengembangan *Extreme Programming*, yaitu:

1. Aktifitas perencanaan: pengumpulan *user stories* dari klien yang klien tetapkan prioritasnya. Setiap *story* ditetapkan harga dan lama pembangunan, jika terlalu besar, *story* dapat dipecah menjadi beberapa *story* yang lebih kecil.
2. Aktifitas desain: berprinsip: sederhana. Memanfaatkan kartu *Class-Responsibility-Collaborator* (CRC) untuk identifikasi dan mengatur *class-class* di konsep *object oriented*. Jika ditemui kesulitan, *prototype*. Lakukan *refactoring*, yaitu mengembangkan desain dari program setelah ditulis.
3. Aktifitas pengkodean: siapkan *unit test* sebelum pengkodean dipakai sebagai fokus pemrogram untuk membuat program. *Pair programming* dilakukan untuk *real time program solving* dan *real time quality assurance*.
4. Aktifitas pengujian: menggunakan *unit test* yang dipersiapkan sebelum pengkodean (Proboyekti, 2008).

2.8 Unified Modeling Language (UML)

Menurut Nugroho (2005), pemodelan visual adalah proses penggambaran informasi-informasi secara grafis dengan notasi-notasi baku yang telah disepakati sebelumnya. Notasi-notasi baku sangat penting demi suatu alasan komunikasi. Dengan notasi-notasi pemodelan yang bersifat baku komunikasi yang baik akan terjalin dengan mudah antar anggota tim pengembang sistem/perangkat lunak dan antara anggota tim pengembang dengan para pengguna (*end-user*). Untuk melakukan pemodelan sistem/perangkat lunak, dalam buku ini kita akan menggunakan notasi-notasi *Unified Modeling Language* (UML) yang akan kita gambarkan secara elektronik (dengan bantuan komputer) lewat sarana perangkat lunak. Dengan pemodelan menggunakan UML ini, pengembang dapat melakukan:

1. Tinjauan umum bagaimana arsitektur sistem secara keseluruhan.
2. Penelaahan bagaimana objek-objek dalam sistem saling mengirim pesan (*message*) dan saling bekerjasama satu sama lain.
3. Menguji apakah sistem/perangkat lunak sudah berfungsi seperti yang seharusnya.
4. Dokumentasi sistem/perangkat lunak untuk keperluan-keperluan tertentu di masa yang akan datang.

Menurut Haviluddin dalam (Nugroho, 2011) membangun *software* menggunakan pendekatan teknologi objek memberikan beberapa keuntungan, antara lain: memungkinkan penggunaan kembali objek yang ada (*reusable*), memungkinkan *software* yang baru dengan konstruksi yang lebih besar, *software* berorientasi objek secara umum lebih mudah dimodifikasi dan dirawat karena sebuah objek dapat dimodifikasi tanpa banyak berpengaruh pada objek yang lain.

Haviluddin dalam (Nugroho, 2011) juga mengatakan bahwa UML dipandang masih mempunyai kekurangan terutama dalam meng-*generate* kode program secara komplit. Hal ini karena kurangnya cara memodelkan aspek kelakuan internal perangkat lunak untuk dipetakan ke dalam kode program. Seperti yang kita ketahui, diagram UML yang dapat menghasilkan kode hanyalah diagram *class*, namun itupun hanya sebatas kerangka kodenya saja dan tidak bisa meng-*generate* badan programnya.

2.8.1 Diagram-diagram *Unified Modeling Language* (UML)

Menurut Sholiq (2010), UML menyediakan beberapa diagram visual yang menunjukkan berbagai aspek dalam sistem. Banyaknya diagram tersebut dimaksudkan untuk memberikan gambaran yang terintegrasi terhadap sistem yang akan dibangun. Beberapa diagram yang disediakan dalam UML antara lain:

1. Diagram *use case* bisnis
2. Diagram *use case*
3. Diagram aktivitas
4. Diagram sekuensial
5. Diagram kolaborasi
6. Diagram kelas
7. Diagram *statechart*
8. Diagram komponen
9. Diagram *deployment*

2.9 Black Box Testing

Menurut Romeo (2003), *black box testing* merupakan testing yang dilakukan tanpa pengetahuan detail struktur internal dari sistem atau komponen yang dites. *Black box testing* juga disebut sebagai *behavioral testing*, *specification-based testing*, *input/output testing* atau *functional testing*. *Black box testing* berfokus pada kebutuhan fungsional pada *software*, berdasarkan pada spesifikasi kebutuhan dari *software*.

Kategori *error* yang akan diketahui dengan menggunakan *black box testing* adalah:

1. Fungsi yang hilang atau tidak benar.
2. *Error* dari interface.
3. *Error* dari struktur data atau akses *external database*.
4. *Error* dari kinerja atau tingkah laku sistem.
5. *Error* dari inisialisasi dan terminasi.