

BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1 Pengertian Dasar Sistem

Sistem ini bisa diartikan sebagai prosedur-prosedur yang berhubungan, berkumpul untuk menyelesaikan suatu sasaran tertentu (Ii, 2012). Sistem dikatakan mempunyai karakter-karakter seperti di bawah ini

1. Didalam sistem terdapat komponen atau dikenal dengan bagian dari sistem. *sub system* elemen yang lebih kecil dari sistem, sedangkan *supra system* element yang lebih besar dari sebuah sistem.
2. Suatu sistem mempunyai batas sistem (boundary). Batas sistem disebut juga daerah yang membatasi suatu sistem dengan lingkungan sistem luarnya.
3. Suatu sistem mempunyai lingkungan luar (environment). Lingkungan luar dari batas sistem akan mempengaruhi operasi dari sistem, hal ini dapat juga menguntungkan ataupun merugikan sistem tersebut
4. Suatu sistem mempunyai penghubung (interface). Penghubung merupakan media perantara antar sub sistem. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya.
5. Suatu sistem mempunyai tujuan (goal). Didalam sebuah sistem tentu mempunyai tujuan, karna jika sistem tersebut tidak mempunyai tujuan

maka sistem tersebut di anggap tidak ada gunanya. sasaran dari sistem itu sangat menentukan sekali masukan yang di butuhkan oleh sistem.

2.1.2 Pengertian Dasar Informasi

Menurut Sutabri Tata didalam Susanto, Utami, & Hermanto, (2019) sistem informasi merupakan sebuah kebutuhan dalam pengelolaan transaksi harian yang dapat mendukung fungsi operasi yang bersidat manajerial yang bertujuan menyediakan laporan yang di perlukan.

Menurut Prasajo Diat Lantip, (2011), menyatakan bahwasanya informasi itu bisa juga disebut dengan data. artian dari data itu sendiri adalah data yang belum diolah belum bisa di gunakan untuk pengambilan keputusan.

Dalam manajemen, informasi merupakan data yang telah diproses sehingga mempunyai arti tertentu bagi penerimanya. Data itu berasal dari sebuah informasi, dan juga data itu sendiri menggambarkan sebuah kejadian, dan bisa di tarik kesimpulan bahwa data dan informasi itu saling berkaitan satu sama lain.

2.1.3 Konsep Dasar Sistem Informasi

Menurut Winarno didalam Murtadho & Wahid, (2016), disamping dapat membantu sebuah perusahaan, sebuah sistem informasi itu bisa menjadi faktor kompetitif yang paling utama.

Sistem informasi adalah sistem komposit buatan manusia yang terdiri dari orang, perangkat keras, perangkat lunak, dan sumber daya data. Tujuannya adalah untuk mengumpulkan, memproses, menyimpan, mengirim, dan memberikan

informasi dalam cara yang tepat waktu dan benar untuk mencapai manajemen, regulasi, dan kontrol kegiatan dalam organisasi (Huang, Huang, & Chu, 2019).

Sistem informasi membantu mengatur, memvisualisasikan, dan menganalisis data dan untuk mengkonversi data mentah menjadi informasi yang berguna dengan tujuan keseluruhan mendukung dan meningkatkan pengambilan keputusan. Komponen-komponen yang saling berhubungan dan bekerja sama untuk mengumpulkan, memproses, menyimpan dan menyebarkan informasi untuk mendukung pengambilan keputusan, koordinasi, pengendalian dan untuk memberikan gambaran aktivitas di dalam perusahaan/Organisasi (Siegert, Schlegel, Groß, & Bauernhansl, 2017). Berikut dijelaskan aktifitas dasar dari sistem:

1. *Input*, kegiatan pengelolaan dalam suatu sistem informasi yang diambil dari suatu organisasi atau lingkungan eksternal.
2. *Process*, mengubah sebuah proses ke dalam bentuk yang berguna.
3. *Output*, Memberikan informasi untuk orang yang akan menggunakannya.
4. *Feedback*, Evaluasi terhadap output yang dikembalikan ke pada anggota organisasi.

2.1.4 Komponen Sistem Informasi

Menurut Stmik & Surabaya, (2014), untuk mencapai sasaran didalam sebuah sistem informasi terdapat beberapa blok yang saling berinteraksi, diantaranya adalah:

1. Blok Masukan, data-data yang ada didalam sistem informasi diwakili oleh input.
2. Blok Model, Blok model ini terdiri dari kombinasi prosedur logika dan model matematik yang akan memanipulasi data *input* dan data yang tersimpan didasar data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.
3. Untuk mendapatkan hasil keluaran yang diinginkan maka diperlukan blok model yang terdiri dari peraduan prosedur logika dan bentuk matematik yang bisa mengolah data input dan data yang tersimpan di dalam data.
4. Blok Teknologi, disebut juga dengan kotak alat (toolbox) yang bisa menerima beberapa kegunaan, kegunaan tersebut adalah menerima masukan, menjalankan model, mempunyai output yang berguna untuk sistem secara menyeluruh.
5. Blok Basis DataBasis data (*database*) adalah sekumpulan data-data yang mempunyai keteriaktan yang satu dengan yang lainnya, terletak pada perangkat kersa komputer dan butuh perangkat lunak untuk mengolahnya. Data penting untuk disimpan di dalam sebuah database yang digunakan untuk penyediaan data.
6. Blok Kendali, pengendalian ini penting digunakan untuk menjamin suatu sistem informasi berjalan dengan semestinya.

2.1.5 Pengembangan Dan Perancangan Sistem Informasi

Didalam kegiatan ini yaitu melakukan perubahan dari sistem lama menjadi sebuah sistem yang baru secara semua atatu hanya sebagian dari sistem lama.

2.1.5.1 SDLC (*Software Development Life Cycle*)

Pemodelan yang bisa digunakan untuk melakukan pengembangan suatu system informasi disebut juga dengan SDLC atau *Software Development Life Cycle*. A.S Rosa dan Salahuddin M, (2011), berikut disebutkan tahapan yang teradapat pada SDLC secara umum yaitu

1. Inisiasi (*Initiation*)

Pada tahapan ini digunakan untuk pembuatan proposal suatu perangkat lunak.

2. Mengartikan beberapa konsep yaitu konsep biaya, rencana, lingkup sistem dan pembelajaran kemudahan pada sistem

3. Perencanaan (*Planning*)

Mengembangkan perencanaan dalam memanajemen suatu proyek atau rencana lainnya. perencanaan ini jug bisa menjadi dasar agar bisa mendapatkan sumber daya yang dibutuhkan.

4. Analisis kebutuhan (*requirements analysis*).

Kegiatan analisa untuk pemakai sistem atau disebut juga dengan user dan mengembangkan apa yang dibutuhkan oleh user.

5. Desain (*design*)

Mempresentasi kan apa yng dibutuhkan menjadi sesuatu yang lengkap, desain ini fokus untuk atas kebutuhan fungsi yang pada sistem tersebut.

6. Pengembangan (*development*)

Kegiatan ini adalah merubah desain menjadi sebuah sistem informasi yang lengkap. Pembuatan basis data dan meyiapkan file yang aka di uji serta

melakukan pembuatan kode program, memperbaiki serta menyusun kode program tersebut.

7. Integrasi dan pengujian (*integration and test*)

Memperlihatkan sebuah sistem yang bisa memenuhi kebutuhan yang dicantumkan di dalam dokumen fungsional, dan di validasi oleh staff penjamin kualitas kemudian menghasilkan sebuah laporan analisa pengujian.

8. Implementasi (*implementation*)

Pada tahapan ini dilakukan implementasi pada lingkungan produksi, dalam tahapan ini di jalan solusi dari masalah yang telah diidentifikasi oleh tahapan pengujian.

Terdapat beberapa model dari SDLC yang bisa digunakan, dan yang paling penting mempunyai kelemahan dan kelebihan di dalam setiap model yang ada pada SDLC. Untuk itu kita harus bisa mengenali tipe dan model apa yang paling sesuai dengan *customer* dengan sesuai juga dengan *developer*. Menurut A.S Rosa dan Salahuddin M, (2011) yang terdapat di dalam model SDLC bisa disebutkan seperti di bawah ini:

1. Model *Air terjun*

Model SDLC air terjun (*waterfall*) Biasanya juga dikenal dengan model sequensial atau disebut juga dengan alur hidup klasik. didalam model ini memberikan beberapa pedekatan yang umum digunakan adalah analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

2. Model *prototype*

Permulaan pada model ini adalah melakukan proses dikumpulkannya kebutuhan-kebutuhan yang di minta oleh pelanggan untuk perangkat lunak yang di buat. dan langsung membuat sebuah program awal atau prototype supaya pelanggan bisa membayangkan kebutuhan selanjutnya, program prototype ini biasanya program yang belum jadi sepenuhnya.

3. Model *rapid application Development* (RAD)

Model sangat cocok untuk waktu pengerjaan sistem yang pendek. Model di adaptasi dari model sebelumnya yaitu model waterfall, sifat dari model ini adalah incremental.

4. Model *iterative*

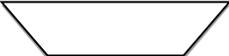
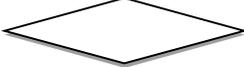
Pada model ini memadukan proses model waterfall dan iteratif pada prototype. pada setiap pertambahannya akan mengalami penambahan-penambahan fungsi.

Dan didalam menyusun penelitian ini, biasanya model air terjun banyak digunakan untuk perancangan sistem informasi berbasis web.

2.1.5.2 Flowchart

Menurut Puspitawati dan Sri Dewi Anggadani, (2011:105) Diketahui *flowchart* mempunyai bagian-bagian yang bisa disebut *sistem*, *document*, dan program *flowchart*. berikut dijelaskan simbol-simbol dari flowchart:

Tabel 2. 1 Simbol *Flowchart*

Simbol	Deskripsi
	<i>Document</i>
	<i>Input manual</i>
	<i>Terminator</i>
	<i>Manual process</i>
	Process
	Selection
	<i>Tape</i>
	<i>Display</i>
	<i>Different Pages Links</i>
	<i>File</i>
	<i>Pendefined Process</i>
	<i>One Page Links</i>
	<i>Database</i>
	<i>Files</i>



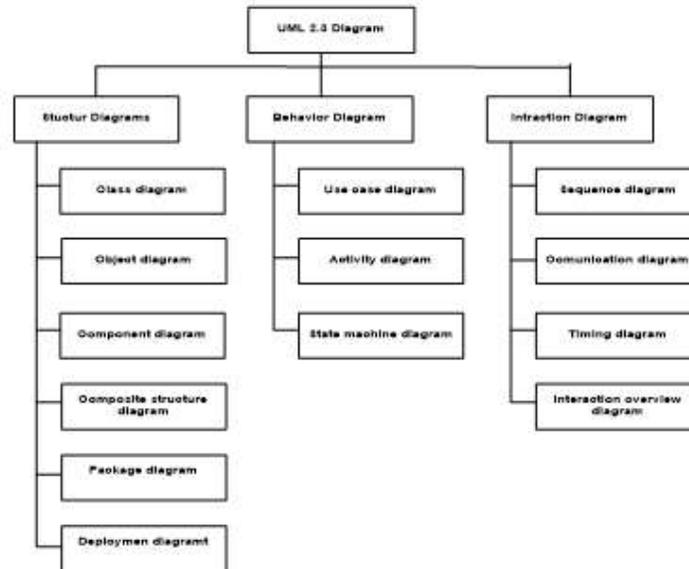
2.1.5.3 UML (*Unified Modeling Language*)

UML disebut juga dengan Unified Modeling Language adalah pemodelan dan bentuk komunikasi menggunakan diagram sebagai teknik untuk pembangunan programan berorientasi dengan object.

Penggunaan dari UML tidak terbatas pada metodologi tertentu saja, sebab kenyataannya UML paling banyak digunakan pada metodologi yang berorientasi pada object.

2.1.5.4 Diagram UML

Ada 13 macam dari diagram UML ini yang telah diketahui dikelompokkan menjadi tiga. Berikut Pembagian dari kategori-kategori tersebut:



Gambar 2.1 *Diagram UML*

Sumber: Rosa dan Shalahuddin (2011: 134)

Disini dijelaskan secara singkat struktur dari kategori-kategori diatas:

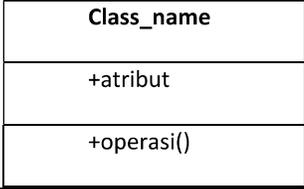
1. *Structure diagram* Adalah digram yang dikumpulkan dan digunakan untuk penggambaran struktur statis dari sistem yang telah dimodelkan.
2. *Behavior diagram* adalah penggambaran kelakuan dari sistem atau rangkaian perubahan yang ada pada sebuah sistem.
3. *Interaction diagram* adalah penggambaran interaksi dari suatu sistem terhadap sistem berbeda dan bagian dari sistem.

Penjelasan dari diagram yang digunakan peneliti dari diagram-digram diatas antara lain:

1. *Class Diagram*

Di dalam pendiskripsian metode yang terdapat di dalam suatu kelas perlu di perhatikan cohesion dan coupling. arti cohesion tersebut adalah ukuran seberapa dekat kaitan instruksi satu sama lain, dan arti dari coupling itu sendiri ukuran seberapa dekat kaitan instruksi satusama lain di dalam suatu kelas. A.S Rosa dan Salahuddin M, (2011).

Tabel 2. 2 Simbol *Diagram UML*

Simbol	Deskripsi
<p>class</p> 	Kelas-kelas yang terdapat pada struktur dari system
<p><i>Interface</i></p>  <p>Interface name</p>	Nama dari interface di dalam Object Oriented Programming
<p><i>association</i></p> 	Relasi-relasi antar kelas
<p>Asosiasi berarah / <i>directed association</i></p> 	Relasi-relasi antar kelas
<p>Generalisasi</p> 	relasi-relasi antar kelas (umum dan khusus)

Kebergantungan / <i>dependency</i> 	relasi-relasi antar kelas (kebergantunagn)
<i>aggregation</i> 	Relasi antaar kelas

Sumber: Rosa dan Shalahuddin (2011: 134)

2. *Use Case Diagram*

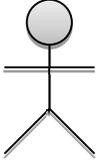
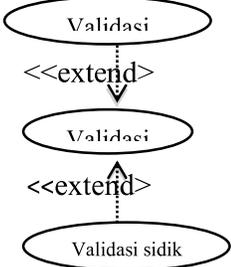
menggambarkan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat.

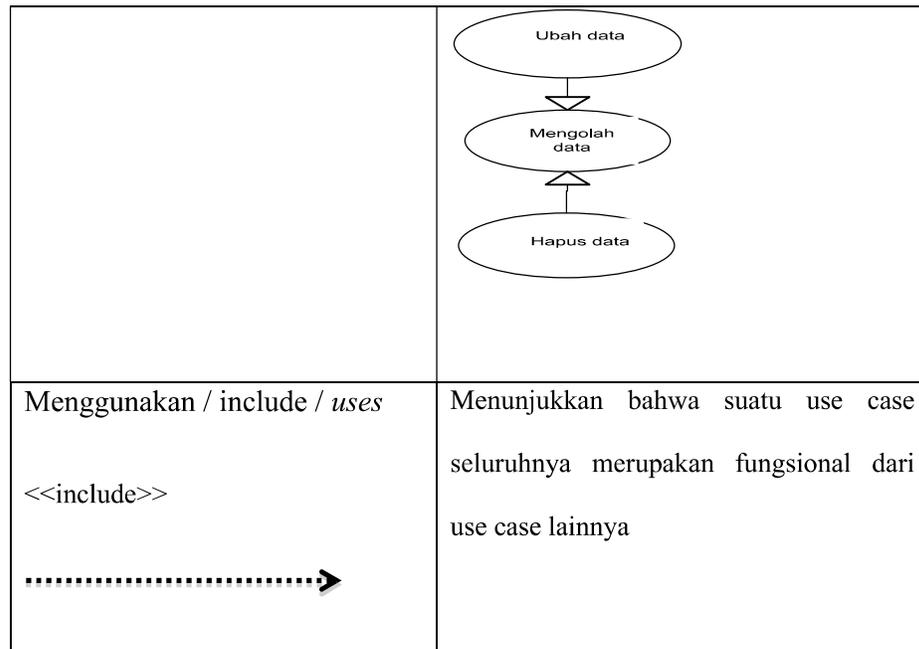
- a. Aktor diartikan sebagai orang, sistem lain, atau sebuah proses yang melakukan interaksi terhadap sistem dimana aktor ini berada di luar dari sistem itu sendiri.
- b. Begitu juga dengan *Use Case*, *Use case* adalah fungsi yang ada dan digunakan sebagai unit-unit dalam melakukan pertukaran antara unit dan aktor.

terdapat Simbol didalam *Use case Diagram*:

Tabel 2. 3 Simbol *Use Case Diagram*

Simbol	Deskripsi

<p><i>Use case</i></p> 	<p>sebagai unit yang bertukar pesan dengan unit atau aktor</p>
<p><i>actor</i></p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri</p>
<p>Asosiasi</p> 	<p>Dideskripsikan jalur komunikasi yang mampu berinteraksi dengan aktor</p>
	<p>Relasi use case tambahan berarti dapat berdiri sendiri tanpa mirip dengan prinsip inherit yang ada pada Object Oriented Programming</p>
<p><i>generalization</i></p> 	<p>Generalisasi hubungan dari umum ke khusus antara dua buah use case, misalnya:</p> <p>Arah panah mengarah pada <i>use case</i> yang menjadi generalisasi (umum)</p>



Sumber: Rosa dan Shalahuddin (2011: 134)

3. Activity Diagram

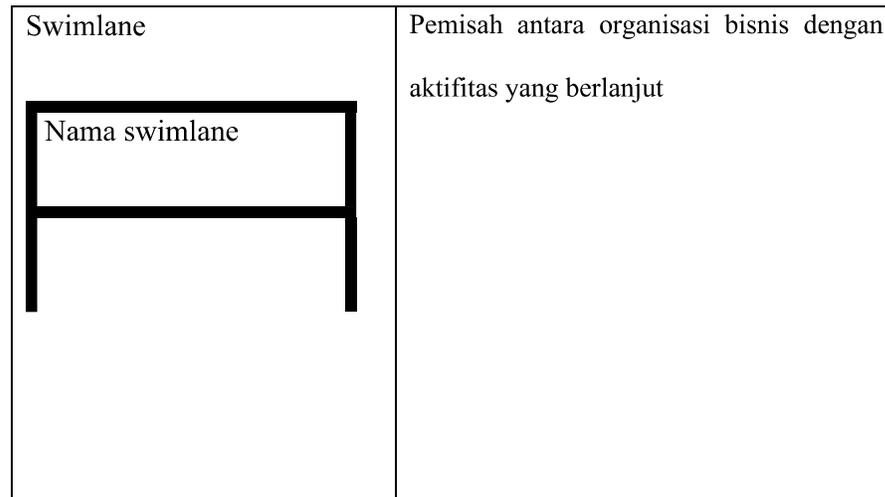
Diagram aktivitas diartikan sebagai Proses penggambaran aliran proses dari sebuah sistem atau porses bisnis. Didalam diagram aktifitas terdapat beberapa pedefinisian, hal tersebut adalah:

- a. Perancangan pada proses bisnis yang setiap urutan aktifitasnya di gambarkan proses bisnis sistem yang ddi deskripsikan.
- b. Setiap-setiap aktifitas itu dianggap memiliki rancangan antar muka tampilan yang dikelompokan menjadi urutan-urutan.
- c. Dari setiap aktifitas itu dianggap memerlukan pengujian-pengujian yang perlu di deskripsikan kasus ujinya.

Menurut A.S Rosa dan Salahuddin M, (2011) Didalam penggunaannya diagram aktifitas memiliki beberapa simbol. Dabawah ini di jelaskan beberapa dari simbol tersebut:

Tabel 2. 4 Simbol *Activity Diagram*

Simbol	Deskripsi
Status awal 	Pengawalan dari suatu aktifitas dari sistem
Aktivitas 	Segala aktifitas yang di lakukan system
Percabangan 	Diartikan sebagai cabang untuk berbagai pilihan yang lebih dari satu
Penggabungan / <i>joint</i> 	Diartikan sebagai suatu aktifitas yang digabungkan dengan aktifitas lainnya
Status Akhir 	Akhir dari proses yang dilakukan oleh system



Sumber: Rosa dan Shalahuddin (2011: 134)

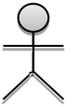
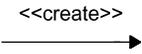
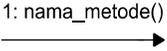
4. *Sequence Diagram*

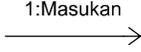
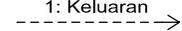
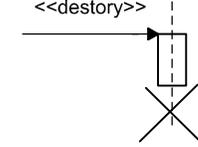
di gambarkan kegiatan object di dalam use case untuk menggambarkan waktu hidup sebuah object dan dikirimkan dan di terima oleh beberapa object. Berapa banyak digaram sequence yang digambarkan mengikuti berapa banyak use case yang telah di gambar..

Menurut A.S Rosa dan Salahuddin M, (2011) Didalam penggunaannya, terdapat beberapa simbol *digram sequence*, simbol-simbol tersebut adalah:

Tabel 2. 5 Simbol *Sequence Diagram*

Simbol	Deskripsi
--------	-----------

<p>aktor</p> <p>atau</p>   <p>tanpa waktu aktif</p>	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri</p>
<p>Garis hidup</p> 	<p>Diartikan sebagai kehidupan suatu objek</p>
<p>Objek</p> 	<p>Diartikan objek yang berinteraksi pesan</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi pesan</p>
<p>Pesan tipe create</p> 	<p>Menyatakan vsuatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>
<p>Pesan tipe call</p> 	<p>Diartikan sebagai object yang memanggil operasi pada object lain atau dirinya sendiri</p>
<p>Pesan tipe <i>send</i></p>	<p>Diartikan sebagai pengiriman data oleh</p>

	suatu object ke object lainnya
<p>Pesan tipe <i>return</i></p> 	Diartikan sebagai object yang telah menjalankan suatu operasi dan menhsaikan kilas balik ke object tertentu
<p>Pesan tipe <i>destroy</i></p> 	Diartikan sebagai obeect yang mengakhiri object lain

Sumber: Rosa dan Shalahuddin (2011: 138)

2.1.6 Preventive Maintenance

Bedasarkan devinisi (Sudrajat, Ating. 2011) didalam (A. S. Gunawan, Setiawan, & Legirian, 2017) *Preventive maintenance* (PM) diartikan sebagai kegiatan pemeliharaan pada mesin produksi dengan penjadwalan tertentu yang bertujuan untuk melancarkan proses produksi yang sedang berlangsung. Adapun tujuan yang terdapat dari PM yaitu mencegah terjadinya kerusakan, serta mendeteksi kerusakan yang akan terjadi, dan bisa menemukan kerusakan yang tersembunyi.

Penelitian sebelumnya pernah dilakukan oleh Zhang, Liu, Lin, & Wu, (2020), menjelaskan bahwa Aktivitas perawatan mesin dianggap bisa

meningkatkan Efisiensi mesin meningkat. Mesin akan dipulihkan ke status awal saat aktivitas pemeliharaan ini dilakukan.

Menurut (S. Pandi, H. Santosa, 2014) Pentingnya kegiatan PM untuk menghadapi fasilitas produksi yang tergolong dalam critical unit, dan golongan critical unit itu sendiri terjadi apabila:

- a. Dapat membayakan keselamatan dan kesehatan pekerja.
- b. Pengaruh terhadap product yang dihasilkan.
- c. Kemacetan terhadap seluruh proses produksi.
- d. Harga fasilitas yang relatif besar dan juga mahal.

kegiatan preventive maintenance ini dapat dibedakan menjadi dua, yang pertama adalah routine maintenance dan yang kedua adalah periodic maintenance:

- a. Routine diartikan adalah perawatan atau pemeliharaan tersebut dilakukan dengan setiap waktu tertentu, contohnya setiap hari.
- b. Periodic diartikan Adalah Pengerjaan perawatan atau pemeliharaan yang di kerjakan dengan kurun waktu tertentu , contohnya seminggu sekali atau pun sebulan sekali.

2.1.6.1 Standar Perawatan

Inspeksi yang dilakukan secara periode atau dikenal dengan secara terencana bisa dilakukan secara efektif dengan cara perusahaan itu memiliki standar perawatan, adapun standar perawatan itu perlu karena:

1. Beragam kegiatan jenis aktifitas perawatan.
2. Memerlukan waktu untuk dapat menguasai teknik perawatan.

3. Kegiatan perawatan di anggap masih kurang efektif di dalam kegiatan produksi.

Ada beberapa hal yang harus distandarisasi, antara lain:

1. Desain peralatan, peralatan yang sama di anggap sebagai standar desain.
2. Performa peralatan bisa di terapkan sebagai standar dari performa peralatan ketika beroperasi.
3. Material bisa di tetapkan sebagai acuan dari alat atau komponen yang di gunakan.

Pengtesan bisa menjadi standar ketika peralatan di pasang, untuk melihat peralatan berkerja dengan baikik.

2.1.6.2 Perencanaan Perawatan

Dalam memlakukan kegiatan preventive maintenance yang optimal, perusahaan perlu melakukan perencenan yang matang. Disni disebutkan beberapa tipe dari rencana perawatan:

1. Tahunan perawatan.
2. Bulanan perawatan.
3. Mingguan perawatan.
4. *Big Repair*.

Pemahaman tentang rencana perawatan sangat perlu dilakukan sebab tanpa adanya pemahan tentang kondisi peralatan tersebut agak rumit dalam melakukan perencanaan. Prioritas untuk setiap perawatan itu tidak perlu sama. prioritas ini

dapat berubah sewaktu-waktu sesuai dengan rencana produksi maupun penempatan peralatan yang baru.

Berikut disebutkan beberapa program yang bisa digunakan untuk melakukan rencana perawatan, beberapa langkah-langkahnya adalah sebagai berikut:

1. Pilih pekerjaan yang dibutuhkan.
 - a. Peraturan perusahaan.
 - b. Standar perawatan.
 - c. Note kerusakan.
 - d. Rencana tahunan sebelumnya.
 - e. *Job order* dari rantai produksi.
2. Seleksi pekerjaan yang akan dilaksanakan.
3. Hitung interval perawatan tentatif.
4. hitung jadwal pekerjaan, waktu dan biaya perawatan
5. Pengaturan kerja.

Mempersiapkan rencana perawatan bulanan:

- a. Tentukan prioritas pekerjaan.
- b. Perkirakan biaya dan tenaga kerja yang diperlukan.
- c. Seimbangkan beban kerja dan persiapkan penjadwalan.

Didalam pekerjaan yang besar, diperlukan perawatan yang besar pula disebabkan peralatan harus dimatikan dalam kurun waktu tertentu, dapat dikatakan jika perawatan yang besar memerlukan biaya yang besar. Overhaul diartikan sebagai biaya yang besar, oleh karena itu harus dideskripsikan secara detail. sekecil

apapun harus di data di setiap langkah dalam melakukan perawatan. terdapat 4 hal yang harus di perhatikan dalam overhaul, hal tersebut adalah:

1. Identifikasi masalah.
2. Detail adminsitratif yang pasti.
3. Implementasi proyek perawatan.
4. Kontrol kemajuan proyek perawatan.

2.1.6.3 Catatan Perawatan

Mencatat setiap aktifitas dari perawatan adalah kegiatan perawatan yang wajib dilakukan, kegiatan ini dianggap sulit untuk dilakukan, tapi penting sekali untuk dilakukan. Pencatatan kegiatan perawatan tidaklah terstandarisasi, tergantung pada perusahaan itu sendiri, setiap perusahaan mempunyai bentuk catatan masing-masing.

Kebijakan perusahaan atau level managerial ikut mengambil andil dalam implementasi perawatan itu sendiri, berikut beberapa catatan perawatan yang dibuthkan: Penggunaan sistem yang sudah terkomputerisasi lebih digunakan dalam melakukan catatan perawata, hal ini dapatmeudahkan penganalisaan pada kerusakan dalam jumlah yang banyak. tahapan pertama untuk dapat menerapkan langkah ini yaitu mensederhanakan prosedur administratif.

2.1.6.4 Pengendalian Suku Cadang

Tujuan dari pengendalian suku cadang antara lain:

1. Meningkatkan *reliabilitas* peralatan dan memperpanjang umur pakai peralatan melalui pemilihan secara teliti pada proses pembelian dan penyimpanan *sparepart*.
2. Menjamin ketersediaan *spare part* pada saat dibutuhkan sehingga dapat meminimasi waktu perawatan.
3. Mengurangi persediaan, biaya pemesanan dan biaya penyimpanan melalui proses pencanaan pengendalian *inventory*.

Salah satu aktivitas pengendalian suku cadang adalah mengklasifikasikan suku cadang. Tindakan ini merupakan langkah awal dari pengendalian melalui upaya mengidentifikasi status persediaan yang terbaru dan mengklasifikasikan secara terstruktur. Proses pengklasifikasian dibagi menjadi beberapa bagian, antara lain:

1. Suku cadang operasional seperti alat ukur.
2. Suku cadang perawatan yang dinilai penting dan peralatan yang dinilai kurang penting.

Pengendalian suku cadang juga sangat berhubungan erat dengan metode penyimpanan. Metode tersebut diklasifikasikan menjadi:

1. Metode penyimpan permanen, dilakukan dengan cara mengidentifikasi variasi dari persediaan,
 - a. metode *order point*, sebaiknya digunakan untuk suku cadang yang jumlahnya kecil dengan konsumsi pemakaian yang stabil. Kuantitas pemesanan yang tetap dipesan ketika persediaan telah berada di titik *order point* dan pengiriman selanjutnya menyusul saat tingkat *stock minimum*.

- b. Metode *double bin*, pemesanan sekaligus dengan 2 kontainer dimana kontainer yang satu untuk *quantity order* dan yang satu untuk menjaga stik minimum persediaan. Metode ini sebaiknya digunakan untuk suku cadang dengan kuantitas yang besar.
 - c. Metode *package*, pemesanan suku cadang dilakukan jika *box* atau kemasan suku cadang telah terbuka.
 - d. Metode *batch issue*, pada metode ini setiap batch produksi memiliki standar jumlah suku cadang yang diperlukan. Seseorang dapat meminta suku cadang yang sebelumnya sudah memiliki jumlah standar pemesanan terhadap suku cadang.
 - e. Metode *fixed quantity ordering*, kuantitas persediaan maksimum ditetapkan sekecil mungkin dan *order* dilakukan ketika sebuah suku cadang sudah digunakan. Jumlah suku cadang yang disimpan stabil dan tetap. Metode ini sebaiknya digunakan untuk suku cadang yang mahal.
 - f. Metode *fixed interval ordering*, metode ini dilakukan dengan melakukan pemesanan secara konstan dan tetap. Misalnya, jarak antara pemesanan setiap 6 bulan atau setiap satu tahun dan jumlah pemesanan relatif besar.
2. Metode berdasarkan kontrak spesial dengan *supplier*
- a. Metode pengiriman parsial berdasarkan kontrak harga per-unit, harga unit dari setiap *part* ditetapkan melalui rata-rata kuantitas barang yang

digunakan selama periode pemesanan, biasanya pemesanan dikirim dengan jumlah *batch* yang kecil.

Sistem deposit, pada sistem ini, gudang memberikan efek pinjaman kepada *supplier*. Jadi material sudah dikirim dahulu ke gudang untuk dideposit dan ketika ada barang yang digunakan maka pembayaran dilakukan.

2.2 Teori Khusus

2.2.1 WEB

Menurut Nugroho & Saleh, (2012) Web atau bisa disebut juga dengan Word Wide Web, adalah sebuah tempat yang paling menonjol di dalam dunia internet, penggunaan hypertext adalah awal dari dari web, dan semakin berkembangnya teknologi hioertext ini berganti dengan www. Kemampuan hipertext ini berbagai macam, mampu menyimpan gambar, video suara dan bahkan bisa berkembang kearah multimedia.

Menurut Sidik, (2012) *Word Wide Web(WWW)*, Web ini digunakan oleh pemakai yang terkoneksi dengan internet yang pada awalnya digunakan sebagai sarana atau tempat mencari informasi yang dibutuhkan dengan cara memasukan link yang disediakan di dalam dokumen-dokumen web yang ada pada browser. *Web* banyak di gunakan oleh perusahaan-perusahaan sebagai strategi teknologi informasinya, sebab web dapat memudahkan pemakai untuk berkomunikasi dengan pemakai internet yang lain. hal itu disebabkan oleh:

1. Informasi yang mudah di akases
2. Lebh mudah mengkonfigurasi *server*

3. Mudah untuk mendistribusikan informasi

web bisa di gunakan pada sistem operasi mana saja, hal ini disebabkan oleh penggunaan standar dokumen yang ada pada web.

2.2.2 HTML5

Seiring berkembang nya waktu sekarang *HTML* telah berubah menjadi *HTML5* di dalam membangun sebuah web, diaman *HTML5* ini mempunyai kelebihan dibanding versi yang sebelumnya. salah satunya menyederhanakan sintak-sintak dari *HTML* yang terdahulu. Setelah revisi lima kali dari *HTML* dan masih dalam tahap perkembangan. Kedepannya standar dari pemograman web akan di gantikan oleh *HTML5*. Didalam *HTML5* juga terdapat beberapa elemen *figure, video, audio, source, embedded, canvas*, dan elemen-elemen lainnya yang berkaitan dengan *multimedia*.

2.2.3 CSS3

CSS atau *Cascading Style Sheet* biasanya digunakan pembangunan komponen di dalam web yang membuat tampilan dari web menjadi rapi atau terstruktur juga seragam. Penggunaan *CSS* ini akan meringankan web karna di pisahkan dari konten utama., perbedaannya akan sangat jelas dibandingkan dengan wweb yang tidak menggunakan *CSS*.(Agus Saputra, 2012).

Pada saat ini *CSS* sudah berkembang sampai versi ke tiga, dimana sudah jelas ada peningkatan di setia versi yang di keluarkan. Agus Saputra,(2012:28) menjelaskan disini disebutkan bahwa ada beberapa peningkatan:

1. *CSS1*, *CSS1*, terlihat masih kuno, dan hanya bisa dikembangkan dan digunakan untuk formatting bertipe dokumen html..
2. *CSS2*, *CSS2*, peningkatan sudah mulai menggunakan font, table-layout, dan berbagai media printer.

CSS telah mengembangkan versinya yang menjadi *CSS3*, banyak peningkatan dibagian animasi yang paling mencolok dari peningkatan ini

2.2.4 PHP

Menurut Elisa, Yana, & Noor, (2012), PHP memiliki kemampuan memproses data yang dinamis sehingga bahasa pemrograman ini cocok digunakan untuk sistem yang berbasis web. Menurut Aditya, (2011) Ada beberapa keunggulan dari PHP yang bisa kita lihat di dalam pemrograman web, keunggulan tersebut adalah:

1. Dalam Pemrograman PHP sering disebut juga dengan bahasa script yang tidak terdapat komilasi dalam penggunaanya.
2. Banyak dukungan *PHP* dan bisa di tempatkan dimana-mana seperti *apache*, *IIS*, *lightpd* dan *Xitami* dengan konfigurasi yang mudah.
3. Dalam sisi pengembangan lebih mudah, Banyaknya support dalah sis pengembangan disebabkan banyak developer yang siap memebantu dalam pengembangannya.
4. Dari segi pemahaman, banyak refferensi yang bisa digunakan didalam pemrograman *PHP*.

PHP juga dapat di gunakan di dalam (*linux, Unix, Macintosh, Windows*) yang bisa dijalankan di dalam console melalui perintah-perintah sistem.

2.2.5 MySQL

Menurut Dan, (2012) Merupakan program database yang sangat cepat dalam menerima dan mengirimkan data, MySQL bisa gunakan oleh beberapa pengguna dengan mmengikuti perintah standar yang disebut SQL (Structure Query Language) dan bisa digunakan sebagai client ataupun server.

Menurut Agus Saputra, (2012) Didalam *MySQL* alasan-alasan yang sering di pakai oleh programmer menggunakan MySQL adalah:

1. *open source*.
2. Penggunaan bahasanya adalah *SQL*,
3. Dari segi performa *database*-nya sangat cepat dan stabil.
4. *Easy of use*.
5. Adanya dukungan dari (*group*) pengguna *MySQL*.
6. Bisa digunakan di berbagai system operasi yang berbeda.
7. Bisa digunakan secara bersamaan di dalam waktu yang sama.

2.2.6 NodeJS

Menurut Pada, Teknik, & Polsri, (2019) Node.js disini menggunakan bahasa pemogramman javascript yang telah di desain untuk mengembangkan sebuah apilkasi. NOde.js dapat berjalan di sistem operasi windows, Linux dan Maac OS

tanpa memerlukan migrasi kode progra, selain itu nodejs bisa berjalan di sisi servel layaknya PHP, Ruby dan sebagainya.

2.2.7 XAMPP

Menurut Aditya, (2011), *XAAMP* merupakan aplikasi dari beberapa program yang mendukung banyak sistem operasi, didalam *XAMMP* terdapat program apache *HTTP server*, *MySQL database*, kemudian menggunakan bahasa pemograman PHP dan Perl. *XAMPP* ini terdapat progam *GNU* atau dikenal dengan (*General Public License*) bebas, *XAMPP* ini mempunyai halaman *web* yang dinamis dan mudah digunakan, *XAMPP* ini sendiri merupakan kumpulan dari beberapa program yaitu *Apache*, *MySQL*, *PHP* dan *Perl*.

2.2.8 PLC (Programable Logic Controller)

Menurut Gunawan & Prawoto, (2013) PLC adal sebuah peralatan kontrol di dalam dunia industri, dimana plc mempunyai kehandalan diantaranya sangat mudah di program dan di aplikasikan, penggunaan wirring yang lebih sedikit, pencarian masalah lebih sederhana, penggunaan daya yang lebih rendah, dan mudah dalam memodifikasi sistem. Merseth (1991) dalam (B. Gunawan & Prawoto, 2013) menyatakan bahwa Didalam suatu PLC bisa menggantikan beberapa rangkaian kontrol logic. I/O yang terdapat di PLC memungkinkan untuk interfacing lansung dengan proses yang sebenarnya.