

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem Informasi

2.1.1 Pengertian Dasar Sistem

Secara sederhana, suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berintegrasi, saling tergantung satu sama lain, dan terpadu.[Tata Sutabri, 2012:10]

Contoh, sistem komputer dapat terdiri dari subsistem perangkat keras dan subsistem perangkat lunak. Masing-masing subsistem dapat terdiri dari subsistem-subsistem lagi atau terdiri dari komponen-komponen pendukung sistem itu sendiri. Subsistem perangkat keras dapat terdiri dari alat masukan, alat pemroses, alat keluaran dan media penyimpanan. Subsistem-subsistem yang ada saling berinteraksi dan saling berhubungan membentuk suatu kesatuan sehingga tujuan atau sasaran sistem tersebut dapat tercapai.

2.1.2. Karakteristik Sistem

Menurut Tata Sutabri [2012:13] Karakteristik sistem adalah model umum sebuah sistem yang terdiri dari input, proses, dan output. Hal ini merupakan konsep sebuah sistem yang sangat sederhana mengingat sebuah sistem juga memiliki karakteristik atau sifat-sifat tertentu., yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem.

Adapun karakteristik yang dimaksud adalah sebagai berikut : Komponen Sistem (*Component*), Batasan Sistem (*Boundary*), .Lingkungan Luar Sistem (*Environtment*), Penghubung Sistem (*Interface*), Masukan Sistem (*Input*), Keluaran Sistem (*Output*), Pengolah Sistem (*Process*), Sasaran Sistem (*Objective*).

2.2 Konsep Dasar Informasi

2.2.1. Pengertian Informasi

Menurut Tata Sutabri [2012:22] informasi adalah data yang telah diklasifikasi atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi mengolah data menjadi informasi atau tepatnya mengolah data dari bentuk tak berguna menjadi berguna bagi penerimanya. Sistem informasi bukan merupakan hal yang baru, yang baru adalah komputerisasinya.

Sebelum ada komputer, teknik penyaluran informasi yang memungkinkan manajer merencanakan serta mengendalikan operasi telah ada. Komputer menambahkan satu atau dua dimensi, seperti kecepatan, ketelitian, dan penyediaan data dengan volume yang lebih besar yang memberikan bahan pertimbangan yang lebih banyak untuk mengambil keputusan.

2.2.2. Kualitas Informasi

Menurut Tata Sutabri [2012:31] Informasi bernilai sempurna apabila pengambil keputusan dapat mengambil keputusan secara optimal dalam setiap hal, dan bukan keputusan yang “rata-rata” akan menjadi optimal dan untuk menghindari kejadian-kejadian yang akan mendatangkan kerugian. Namun demikian informasi yang sempurna mungkin memang tidak ada.

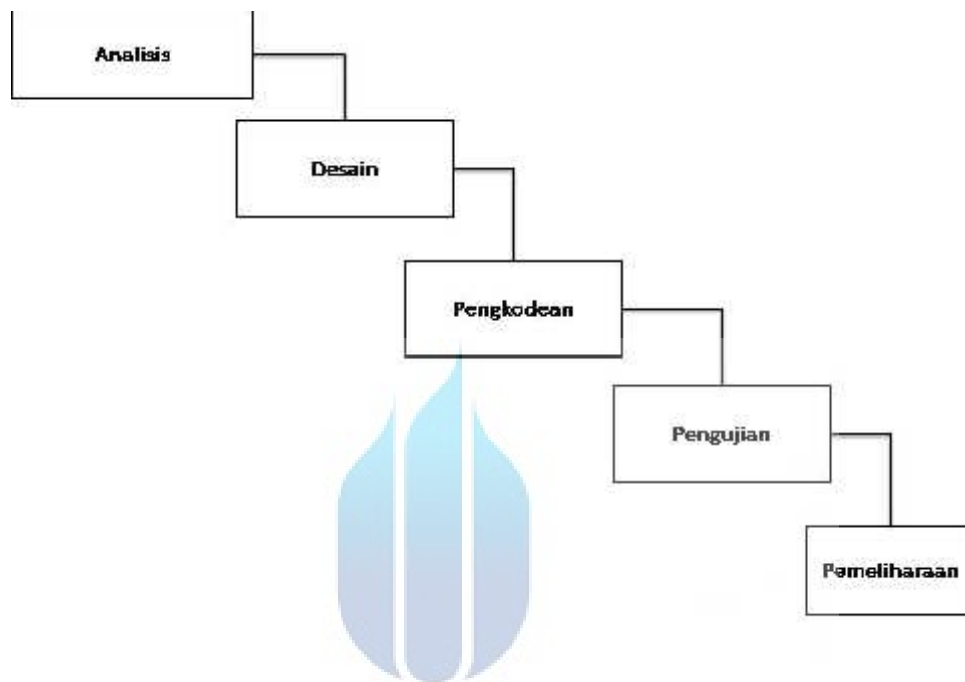
Dalam hal-hal demikian, perkiraan hasil sebelumnya mungkin dipengaruhi oleh informasi oleh informasi tambahan, meskipun informasi tersebut tidak memberikan kepastian. Informasi yang tidak sempurna sesungguhnya merupakan informasi dari uji petik (*sampling*). Informasi ini tidak sempurna karena lebih banyak mengandung perkiraan dan bukan suatu hal yang pasti.

Adapun kualitas dari suatu informasi tergantung dari 3 (tiga) hal, yaitu informasi harus akurat (*accurate*), tepat waktu (*timelines*), dan relevan (*relevance*).

2.3 SDLC (System Developmet Life Cycle)

2.3.1. Metode Waterfall

Menurut [Rosa, 2011] model air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut didalam dari analisis, desain, pengodean, pengujian dan tahapan pendukung.



Gambar 2.1 Model Waterfall [Rosa a.s-m.Shalahuddin:2011]

1. Analisa kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

2. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap

selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

3. Pembuatan Kode Program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahapan ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahapan desain.

4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru.

Tahap pendukung atau pemeliharaan dapat mengurangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

2.4 Definisi UML (*Unified Modelling Language*)

Menurut [Prabowo Pudjo Widodo Herlawati:2011] UML (Unified Modeling Language), adalah satu set standar teknik diagram yang memberikan sistem proyek dari analisis melalui implementasi. Saat ini sebagian besar sistem berorientasi objek pendekatan analisis dan desain menggunakan UML untuk menggambarkan sistem berkembang. UML menggunakan satu set diagram yang berbeda untuk menggambarkan berbagai pandangan dari sistem berkembang.

Diagram di kelompokkan menjadi dua klasifikasi luas : struktur dan perilaku struktur diagram termasuk kelas, objek, paket, penyebaran, komponen ,dan komposit. UML tipe 2.0 adalah suatu metode terbuka yang digunakan untuk menspesifikasi, memvisualisasi, membangun, dan mendokumentasikan artifak-artifak dari suatu pengembangan sistem piranti lunak yang berbasis pada objek. Seperti bahasa-bahasa lainnya, UML mendefinisikan notasi dan syntax. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak dan UML 2 terdiri dari 13 jenis diagram resmi seperti yang tertulis berikut ini :

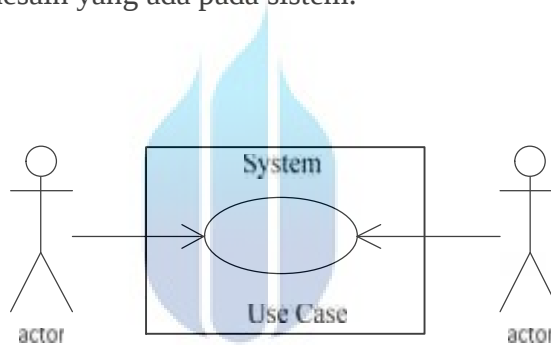
Tabel 2.1 Jenis-jenis Diagram UML

No	Diagram	Kegunaan
1	Activity	<i>Behaviour</i> procedural dan parallel
2	Class	Class, fitur, dan hubungan-hubungan
3	Communication	Interaksi antar objek penekanan pada jalur
4	Component	Struktur dan koneksi komponen
5	Composite structure	Dekomposisi <i>runtime</i> sebuah class
6	Deployment	Pemindahan artifak ke node
7	Interaction overview	Campuran <i>sequence</i> dan <i>activity</i> diagram
8	Object	Contoh konfigurasi dari contoh-contoh
9	Package	Struktur hirarki <i>compile-time</i>
10	Sequence	Interaksi antar objek; penekanan pada <i>sequence</i>
11	State machine	Bagaimana <i>even</i> mengubah objek selama aktif
12	Timing	Interaksi antar objek; penekanan pada timing
13	Use case	Bagaimana pengguna berinteraksi dengan sebuah system

2.4.1. Use Case Diagram

Menurut [Prabowo Pudjo Widodo Herlawati:2011] pengertian dari Use Case Diagram adalah “Diagram fungsional dalam bahwa mereka menggambarkan fungsi dasar dari sistem” ke dua deskripsi use case dan diagram use case di dasarkan pada persyaratan di identifikasi dan deskripsi diagram aktivitas deskripsi proses. Use Case menjelaskan cara yang sangat sederhana dari fungsi utama dari sebuah sistem dan bermacam-macam perbedaan dari pemakai akan saling berpengaruh. Use Case digunakan untuk menggambarkan fungsi dari sistem dan sebagai model dialog antara aktor dan sistem”.

Use case diagram menampilkan *actor*, *use case*, dan hubungan antara mereka. *Use case diagram* dapat sangat membantu bila kita sedang menyusun kebutuhan sistem, mengkomunikasikan rancangan dengan klien dan merancang test khusus untuk desain yang ada pada sistem.



Gambar 2.2 Use Case Model [Prabowo Pudjo Widodo Herlawati:2011]

Tabel 2.2 Simbol – simbol Use Case Diagram

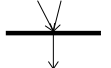
GAMBAR	NAMA	KETERANGAN
	<i>Actor</i>	Menspesifikasikan seseorang peran yang pengguna mainkan ketika aktor di dalamnya alternatif berinteraksi dengan <i>use case</i> .

	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung pada nya elemen yang tidak mandiri (<i>independent</i>).
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atas nya objek induk (<i>ancestor</i>).
	<i>Include</i>	Menspesifikasikan fungsi satu kasus penggunaan dengan yang lain bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
	<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
	<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemen nya (<i>sinergi</i>).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

2.4.2. Activity Diagram

Menurut [Prabowo Pudjo Widodo Herlawati :2011] Activity Diagram adalah “Activity Diagram yang digunakan untuk menggunakan tingkah laku proses bisnis satu orang dari obyek. Dalam beberapa kebiasaan bisa dilihat sebagai diagram aliran data yang sudah berpengalaman digunakan dalam conjunction dengan struktur analis. Tidak hanya itu, diagram activity dapat digunakan untuk menggambarkan segala macam dari level menengah aliran kerja bahwa involve beberapa perbedaan Use Case, untuk menjelaskan use case perorangan. Kesimpulannya untuk mengkhususkan detail dari metode perorangan”. Berikut ini adalah simbol-simbol yang sering digunakan pada saat pembuatan activity diagram :

Tabel 2.3 Simbol Activity Diagram

GAMBAR	NAMA	KETERANGAN
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antar muka saling berinteraksi satu sama lain
	<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Activity Final Node</i>	Bagaimana objek dibentuk dan di hancurkan
	<i>Join Node</i>	Digunakan untuk membawa kembali bersama-sama satu searus paralel atau kegiatan bersamaan (atau tindakan)

	<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran
	<i>Decision</i>	Digunakan untuk mewakili kondisi tes untuk memastikan bahwa aliran kontrol atau aliran objek hanya turun satu jalur
	<i>Swimlane</i>	Digunakan untuk memecah sebuah diagram aktivitas dalam baris dan kolom untuk menetapkan individu kegiatan (atau tindakan) kepada individu atau objek yang bertanggung jawab untuk melaksanakan kegiatan (atau tindakan)

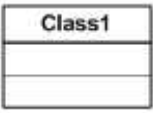
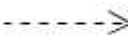
2.4.3. Class Diagram

Menurut [Prabowo Pudjo Widodo Herlawati 2011] Class Diagram adalah “ Model statis yang menunjukkan kelas dan hubungan antar kelas yang tetap konstan dalam sistem dari waktu ke waktu. Diagram kelas menggambarkan kelas yang meliputi perilaku dengan hubungan antara kelas. bagian berikut pertama menyajikan unsur-unsur diagram kelas ”. Diagram class bersifat statis, menggambarkan hubungan apa yang terjadi bukan apa yang terjadi jika mereka berhubungan.

Sebuah class memiliki tiga area pokok :

- Nama, merupakan nama dari sebuah kelas.
- Atribut, merupakan properti dari sebuah kelas. Atribut melambangkan batas nilai yang mungkin ada pada obyek dari class.
- Operasi, adalah sesuatu yang bisa dilakukan oleh sebuah class atau yang dapat dilakukan oleh class lain terhadap sebuah class.

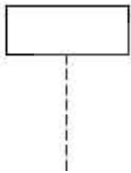
Tabel 2.4 Notasi pada Class Diagram

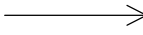
GAMBAR	NAMA	KETERANGAN
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
	<i>Generalization</i>	Merupakan semacam hubungan antara beberapa class.
	<i>Nary Association</i>	Upaya untuk menghin dari asosiasi dengan lebih dari 2 objek.
	<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
	<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

2.4.4. Sequence Diagram

Menurut [Prabowo Pudjo Widodo Herlawati:2011] Sequence Diagram adalah “Sequence Diagram mengilustrasikan obyek yang mengambil bagian dalam Use Case dan pesan yang terlewat diantara mereka untuk satu Use Case. Sebuah Sequence Diagram adalah model dinamis yang menunjukkan urutan eksplisit pesan yang lewat diantara objek dalam interaksi yang di definisikan”.

Tabel 2.5 Simbol – simbol yang terdapat pada Sequence Diagram

Simbol	Nama Simbol	Kegunaan
	Object	Object, merupakan instance dari sebuah class dan dituliskan tersusun secara horizontal.
	Actor	Actor, yang digunakan untuk menggambarkan pelaku atau pengguna dalam use case. Pelaku ini meliputi manusia atau sistem komputer atau subsistem lain yang memiliki metode untuk melakukan sesuatu.
	Lifeline	Lifeline, digunakan untuk mempresentasikan sebuah individu dalam interaksi dan hanya sebuah entitas interaksi.
	Activation	Activation dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah lifeline. Ditujukan untuk mengidentifikasikan sebuah obyek yang akan melakukan sebuah aksi.

	Message	Message , Digambarkan dengan anak panah horizontal antara Activation. Message mengidentifikasi komunikasi antara object.
---	---------	--

2.5 Database

Basis data (*Database*) adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis, sehingga data digunakan oleh suatu program komputer untuk memperoleh informasi dari basis data tersebut. Basis data adalah sekumpulan data yang terhubung satu sama lain secara logika dan suatu deskripsi data yang dirancang untuk memenuhi kebutuhan informasi suatu organisasi. [Verdi Yasin, 2012:274]

Menurut Fathansyah [2012:2-3] basis data (*Database*) dapat didefinisikan dalam sejumlah sudut pandang seperti

- Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak data dimanfaatkan kembali dengan cepat dan mudah.
- Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
- Kumpulan file/table/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

2.5.1 MySql

Menurut [Abdul Kadir 2010] MySQL termasuk jenis RDBMS (Relational Database Management System). Sehingga istilah seperti tabel, baris, dan kolom tetap digunakan dalam MySQL. Pada MySQL sebuah database mengandung satu beberapa tabel, tabel terdiri dari sejumlah baris dan kolom.

mysql dikembangkan oleh sebuah perusahaan Swedia bernama *mysql AB*,. dapat dilihat dan gratis, serta *server mysql* dapat dipakai tanpa biaya tapi hanya untuk kebutuhan nonkomersial.

Sementara distribusi *Windows mysql* sendiri dirilis secara *shareware*. Barulah pada Juni 2000 *mysql AB* mengumumkan bahwa sejak versi 3.23.19, *mysql* adalah *Software* bebas berlisensi GPL. Artinya, “*Source code mysql* dapat dilihat dan gratis, serta *server mysql* dapat dipakai tanpa biaya untuk kebutuhan apa pun.

Perangkat lunak ini bermanfaat untuk mengelola data dengan cara yang sangat flexibel dan cepat. Berikut adalah sejumlah aktivitas yang terkait dengan data yang didukung oleh perangkat lunak tersebut :

1. Menyimpan data ke dalam tabel
2. Menghapus data dalam tabel
3. Mengubah data dalam tabel
4. Mengambil data yang tersimpa dalam tabel
5. Memungkinan untuk memilih data tertentu yang diambil
6. Memungkinkan untuk melakukan pengaturan hak akses terhadap data.

Dan pada penggunaannya dalam *MySql* ini bisa menggunakan banyak tools, salah satunya *phpMyAdmin*.



Gambar 2.3 phpMyAdmin Localhost

2.5.2. XAMPP

XAMPP Adalah perangkat yang menggabungkan tiga aplikasi ke dalam satu paket, yaitu Apache, MySQL, Dan PHP MyAdmin. Dengan XAMPP pekerjaan Anda akan sangat dimudahkan, karena dapat menginstalasi dan mengkonfigurasi ketiga aplikasi tadi secara sekaligus dan otomatis.

XAMPP telah mengalami perkembangan dari waktu ke waktu. Versi yang terbaru adalah revisi dari edisi sebelumnya, sehingga lebih baik dan lebih lengkap. Aplikasi utama dalam paket Xampp setidaknya terdiri atas web server Apache, MySQL, FileZilla, Mercury, Tomcat.

2.5.3. PHP (Hypertext Preprocessor)

Menurut [Loka Dwiartara, 2013] PHP termasuk dalam open source product. Jadi Anda dapat merubah source code dan mendistribusikannya secara bebas. PHP juga diedarkan secara gratis. Anda bisa mendapatkannya secara gratis. PHP juga dapat berjalan di berbagai wweb server semisal IIS, Apache, PWS dan lain-lain. Adapun kelebihan-kelebihan dari PHP yaitu:

- PHP mudah dibuat dan kecepatan akses tinggi.
- PHP dapat berjalan dalam web server yang berbeda dan dalam sistem operasi yang berbeda pula. PHP dapat berjalan disistem operasi UNIX, Windows98, Windows NT dan Macintosh.
- PHP diterbitkan secara gratis.
- PHP juga dapat berjalan pada web server Microsoft Personal, Web Server, Apache, IIS, Xitami dan sebagainya.
- PHP adalah termasuk bahasa yang embedded (bisa ditempel atau diletakan dalam tag HTML).

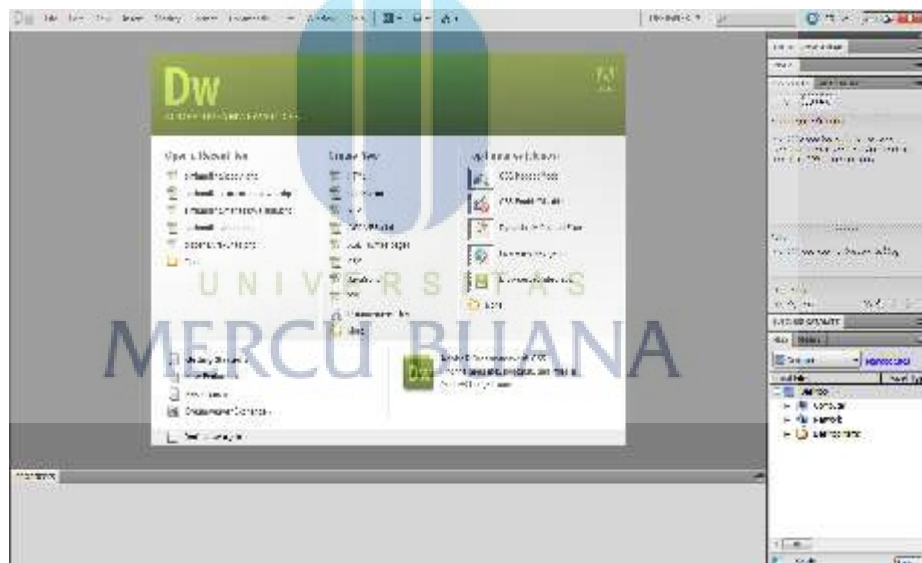
Sistem database yang didukung oleh PHP adalah:

- Oracle.
- Sybase.
- mSQL.
- MySQL.

- Solid.
- Generic ODBC.
- Postgres SQL.

2.5.4. Adobe Dreamweaver

Adobe Dreamweaver adalah aplikasi desain dan pengembangan web yang menyediakan editor WYSIWYG (What You See Is What You Get) visual (lebih dikenal sebagai *Design view*) dan kode editor dengan fitur standar seperti syntax highlighting, code completion, dan code collapsing.. Versi Dreamweaver ini adalah Dreamweaver CS6. Versi ini memiliki performa lebih baik dan memiliki tampilan yang memudahkan anda untuk membuat dan mengelola halaman web, termasuk diantaranya dalam hubungannya dengan bahasa pemrograman PHP dan database MySQL.



Gambar 2.4 AdobeDreamweaver

2.6 Definisi Yang Berhubungan

2.6.1. Teori Persediaan (*Inventory*)

Pengendalian persediaan merupakan fungsi manajerial yang sangat penting, karena persediaan phisik banyak perusahaan melibatkan investasi rupiah

terbesar dalam pos aktiva lancar. Bila perusahaan menanamkan terlalu banyak dananya dalam persediaan, menyebabkan biaya penyimpanan yang berlebihan, dan mungkin mempunyai “opportunity cost” (dana dapat ditanamkan dalam investasi yang lebih menguntungkan). Demikian pula, bila perusahaan tidak mempunyai persediaan yang mencukupi, dapat mengakibatkan biaya-biaya dari terjadinya kekurangan bahan.

Persediaan (inventory) menurut T. Hani Handoko (2011:333) adalah segala sesuatu atau sumber daya-sumber daya organisasi yang disimpan dalamantisipasi dalam pemenuhan permintaan.

Persediaan menurut Prof. Dr. Sofjan Assauri (2009:237) menyatakan :

Persediaan adalah suatu aktiva yang meliputi barang-barang milik perusahaan dengan maksud untuk dijual dalam suatu periode usaha atau persediaan barang – barang yang masih dalam proses pengerjaan atau proses produksi ataupun persediaan bahan baku yang masih menunggu penggunaanya dalam proses produksi

Inventory merupakan kata lain dari persediaan barang, istilah persediaan menunjukkan barang-barang yang dimiliki oleh suatu perusahaan. Persediaan dapat mengambil berbagai bentuk yang tergantung pada jenis usaha yang ditekuni oleh perusahaan yang bersangkutan.

Dalam perusahaan yang bergerak dalam bidang jasa, persediaan bahan baku merupakan salah satu unsur yang paling aktif dalam operasional perusahaan yang secara continue digunakan dalam kegiatan memproduksi barang sehari-hari.

2.6.2 Definisi Manajemen Persediaan

Manajemen persediaan adalah kemampuan suatu perusahaan dalam mengatur dan mengelola setiap kebutuhan barang baik barang mentah, barang setengah jadi, dan barang jadi agar selau tersedia baik dalam kondisi pasar yang stabil dan fluktuasi. Untuk mewujudkan persediaan terlaksana secara baik dan stabil maka pihak perusahaan harus menerapkan konsep manajemen persediaan (*inventory management*) yang realistis dan dapat diterima oleh berbagai pihak.

2.6.3. Jenis-jenis Persediaan

Menurut T. Hani Handoko (2011:334) persediaan barang menurut jenisnya dibagi menjadi 3 (tiga) yaitu :

1. Persediaan barang mentah (*raw material*), yaitu persediaan barang-barang yang berwujud seperti baja, kayu, dan komponen-komponen lainnya yang digunakan dalam proses produksi.
2. Persediaan komponen-komponen rakitan (*prucashedcpart component*), yaitu persediaan barang-barang yang terdiri komponen-komponen yang diperoleh dari perusahaan lain, dimana secara langsung dapat dirakit menjadi suatu produk.
3. Persediaan bahan pembantu atau penolong (*supplies*), yaitu persediaan barang-barang yang diperlukan dalam proses produksi, tetapi tidak merupakan bagian atau komponen barang jadi.
4. Persediaan barang dalam proses (*work in process*), yaitu persediaan barang – barang yang merupakan keluaran dari tiap-tiap bagian dalam proses produksi atau yang telah diolah menjadi suatu bentuk, tetapi masih perlu diproses lebih lanjut menjadi barang jadi.
5. Persediaan barang jadi (*Finished goods*), yaitu persediaan barang –barang yang telah selesai diproses atau diolah dalam pabrik dan siap untuk dijual atau dikirim kepada pelanggan.

2.6.4. Fungsi Persediaan

Menurut prof. Dr. Sofjan Assauri (2009:239) persediaan yang terdapat dalam perusahaan dapat dibedakan menurut beberapa cara. Dilihat dari fungsinya, persediaan terbagi 3 (tiga) diantaranya adalah sebagai berikut :

1. Fungsi Decoupling

Fungsi *Decoupling* adalah persediaan yang memungkinkan perusahaan dapat memenuhi permintaan langganan tanpa tergantung pada supplier. Persediaan bahan mentah diadakan agar perusahaan tidak akan sepenuhnya tergantung pada pengadaannya dalam hal kuantitas dan waktu pengiriman.

2. Fungsi Economis Lot Sizing

Persediaan *Lot Size* ini perlu mempertimbangkan penghematan-penghematan atau potongan pembelian, biaya pengangkutan per unit menjadi lebih murah dan sebagainya. Hal ini disebabkan karena perusahaan melakukan pembelian dalam kuantitas yang lebih besar dibandingkan dengan biaya-biaya yang timbul karena besarnya persediaan.

3. Fungsi Antisipasi

Apabila perusahaan menghadapi fluktuasi permintaan yang diperkirakan dan diramalkan berdasarkan pengalaman atau data-data masa lalu, yaitu permintaan musiman. Dalam hal ini perusahaan dapat mengadakan persediaan musiman (*seasonal inventories*). Disamping itu, perusahaan juga sering menghadapi ketidakpastian jangka waktu pengiriman dan permintaan akan barang-barang selama periode tertentu. Dalam hal ini perusahaan memerlukan persediaan ekstra yang disebut persediaan pengaman (*safety stock inventories*).

2.7 *Economic Order Quantity (EOQ)*

Metode EOQ merupakan teknik pengendalian persediaan yang klasik atau tertua yang paling sederhana, model ini diperkenalkan oleh F. W. Harris pada tahun 1915. Metode ini bertujuan untuk meminimisasi biaya total (keseluruhan) dan untuk mendapatkan hasil persediaan ekonomis dengan melakukan efisiensi biaya.

2.7.1. Variabel dan Asumsi dalam EOQ

Menurut Irham Fahmi (120:2012) bentuk variabel dalam *Economical Order Quantity* yang terlihat jelas yaitu:

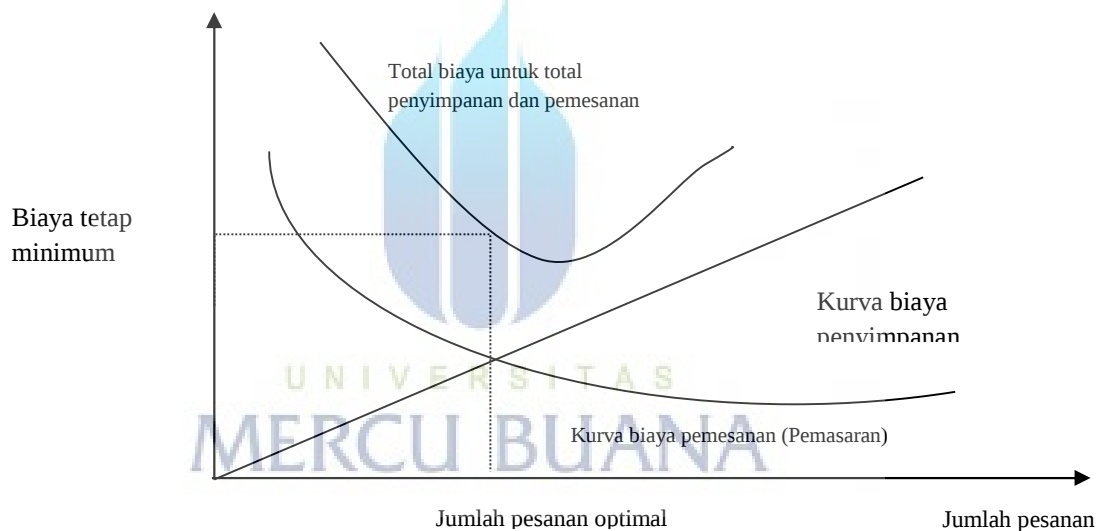
- a. *Total Cost* atau biaya total. Merupakan keseluruhan biaya yang dikeluarkan dalam suatu masa yang terjadi.
- b. *Ordering Cost* atau biaya pesanan. Merupakan biaya keseluruhan biaya yang dikeluarkan selama dalam proses pembelian.
- c. *Carrying cost* atau biaya penyimpanan. Merupakan biaya-biaya yang

dikeluarkan sehubungan dengan penyimpanan.

Menurut Agus Sartono (2012:120) ada tiga asumsi dasar dalam penggunaan model *Economic Order Quantity* yaitu :

- a. Tingkat penjualan dapat diperkirakan
- b. Penggunaan bahan yang konstan
- c. Pemesanan dapat dilakukan seketika
- d. Pengiriman dapat dilakukan dengan cepat

Berdasarkan keterangan diatas maka dapat kita simpulkan bahwa EOQ adalah suatu bentuk usaha dari pihak manajemen perusahaan khususnya bagian persediaan dan produksi untuk selalu menciptakan kondisi dan situasi yang seimbang dan selalu stabil dalam berbagai kondisi. Secara matematis rumus EOQ dapat dikreativitaskan menjadi berbagai bentuk.



Gambar 2.5 Biaya Total sebagai fungsi jumlah pesanan

Sumber : Irham Fahmi 2012 : 121

2.7.2. Formula *Economic Order Quantity* (EOQ)

Menurut Joel Siegel dan Jae K. Shim (2012:120) menyatakan bahwa model *Economic Order Quantity* (EOQ) merupakan model matematik yang menentukan jumlah barang yang harus dipesan untuk memenuhi permintaan yang diproyeksikan, dengan biaya persediaan yang diminimalkan.

Adapun rumus untuk menghitung economic order quantity adalah :

$$EOQ = \frac{\sqrt{2(D)(OC)}}{CC}$$

Keterangan :

- EOQ : *Economic Order Quantity*
 D : Permintaan tahunan (*demand*)
 OC : Biaya Pemesanan (*ordering cost*)
 CC : Biaya penyimpanan (*carrying cost*)

2.7.3. Model EOQ dengan titik pemesanan ulang

Tenggang waktu antara pemesanan dilakukan diterima disebut sebagai tenggang waktu pemesanan ulang (*reorder lead time*). Konsep jarak waktu yang dibutuhkan untuk menerima pesanan diasumsikan konstan. Pemesanan sekarang harus dibuat sebelum saat dimana tingkat persediaan menjadi nol.karena permintaan suatu barang mengkonsumsi persediaan sementara pesanan sedang dikirim, Pemesanan harus dilakukan selama masih terdapat cukup persediaan dalam stok masih memenuhi permintaan selama periode tenggang waktu. Tingkat persediaan ini disebut dengan titik pemesanan ulang.

a. Biaya Pemesanan

Biaya persediaan kedua yaitu hanya melakukan pemesanan, total biaya pemesanan umumnya mencakup beberapa atau seluruh hal –hal berikut:

- Biaya pemerosesan suatu pemesanan, termasuk seluruh pencatatan.
- Biaya transportasi untuk mengangkut pesanan dari pemasok
- Biaya menurunkan pesanan dan menempatkannya dalam persediaan.
- Gaji pegawai yang terlibat dalam proses pemesanan.
- Seluruh perlengkapan yang digunakan dalam pemesanan, termasuk formulir, perangko, telepon.

b. Biaya Pemeliharaan (*Carrying Cost*)

Biaya pemeliharaan juga dikenal dengan biaya penyimpanan, dilambangkan dengan CC adalah biaya untuk pemeliharaan dalam persediaanya. Total biaya pemeliharaan umumnya mencakup beberapa atau seluruh dari hal-hal berikut :

- a. Biaya penyimpanan langsung (sewa,pemanas, lampu, perawatan, keamanan, penanganan, pencatatan, tenaga kerja, dan lain-lain dalam gudang tersebut)
- b. Laba investasi yang ditangguhkan (barang dalam persediaan tidak menghasilkan laba), seperti : bunga atas investasi dalam persediaan, keusangan produk, penyusutan, pajak, asuransi.

