

BAB II

LANDASAN TEORI

2.1 Pengertian Mebel

Menurut Kamus Besar Bahasa Indonesia, mebel adalah perabot yang diperlukan, berguna, atau disukai, seperti barang atau benda yang dapat dipindah – pindah, digunakan untuk melengkapi rumah, kantor, dsb. Kursi, meja, dan lemari merupakan contoh-contoh sederhana dari mebel. Mebel berasal dari kata *movable*, yang artinya bergerak. Mebel juga sering disebut juga sebagai furnitur.

2.2 Pengertian Aplikasi

Aplikasi menurut kamus Komputer Eksekutif (1993:9) adalah masalah yang memakai teknik pemrosesan data. Aplikasi biasanya mengacu pada komputasi yang diinginkan, atau pemrosesan data. Sedangkan menurut Tirtobisono (1999:21), aplikasi adalah istilah yang digunakan untuk pengguna komputer bagi pemecahan masalah. Biasanya istilah aplikasi dipasangkan atau digabungkan dengan suatu perangkat lunak, sehinggal akan memberikan arti atau makna baru yaitu suatu program yang ditulis atau dibuat untuk menangani masalah tertentu.

2.3 Penjualan

2.3.1 Pengertian Penjualan

Penjualan menurut Nickels (1998:10), adalah proses dimana penjual memuaskan segala kebutuhan dan keinginan pembeli agar dicapai manfaat baik

bagi penjual maupun dari pembeli yang berkelanjutan dan menguntungkan kedua belah pihak. Penjualan merupakan sumber hidup suatu perusahaan, karena dari penjualan dapat diperoleh laba serta suatu usaha memikat konsumen yang diusahakan untuk mengetahui daya tarik konsumen sehingga dapat mengetahui hasil produk yang dihasilkan. Menurut Winardi (1996:176), penjualan adalah hasil yang dicapai sebagai imbalan jasa – jasa yang diselenggarakan yang dilakukan perniagaan transaksi dunia usaha.

2.3.2 Jenis Penjualan

Menurut Martin (2006), penjualan dapat dibedakan dan diidentifikasi dari perusahaannya, antara lain:

1. Penjualan Langsung, yaitu penjualan dengan mengambil barang dari supplier dan langsung dikirim ke pelanggan.
2. Penjualan Stok Gudang, yaitu penjualan barang dari stok yang telah tersedia di gudang.
3. Penjualan Kombinasi, yaitu penjualan dengan mengambil barang yang sebagian dari *supplier* dan sebagian dari stok yang tersedia di gudang.

Menurut Sumarni didalam bukunya Bauran Pemasaran dan Loyalitas Pelanggan (2003:321), penjualan dapat dibedakan menjadi:

1. Penjualan Langsung, yaitu suatu proses membantu dan membujuk satu atau lebih calon konsumen untuk membeli barang atau jasa atau bertindak sesuai ide tertentu dengan menggunakan komunikasi tatap muka.
2. Penjualan Tidak Langsung, yaitu bentuk presentase dan promosi barang dan jasa dengan menggunakan media tertentu seperti surat kabar, majalah, radio, televisi, papan iklan, brosur, dan lain-lain.

2.3.3 Sistem Penjualan

Sistem Penjualan adalah sekelompok unsur atau bagian yang saling berhubungan dan berfungsi secara bersama-sama sesuai tugas masing-masing untuk mencapai tujuan yang telah ditetapkan. Menurut Mc Leod (2001:5), sistem penjualan adalah suatu proses yang saling mendukung dalam usahanya untuk memenuhi kebutuhan pembeli dan bersama-sama mendapatkan kepuasan dan keuntungan.

Berikut contoh beberapa unsur atau bagian dalam sistem penjualan barang pada suatu perusahaan:

1. Bagian Penjualan
2. Bagian Gudang
3. Bagian Produksi

Dan untuk elemen atau bagian dalam sistem penjualan dari masing-masing perusahaan mungkin tidak akan sama. Hal ini disebabkan karena kebutuhan informasi yang berbeda-beda antara perusahaan yang satu dengan yang lainnya.

Tujuan sistem penjualan adalah:

1. Mencatat dan mengkonfirmasi *order* penjualan dengan cepat dan akurat.
2. Memastikan bahwa konsumen menerima kiriman produk dan jasa tepat waktu, sesuai yang dijanjikan.
3. Menagih tepat waktu dan akurat, sehingga perputaran kas lebih cepat.
4. Mencatat dan mengelompokkan transaksi keuangan secara cepat dan akurat (ke dalam jurnal maupun ke buku besar).

2.4 Pengertian Web

Menurut Yuhefizar (2008:10), *website*, atau *world wide web* (www) adalah kumpulan halaman-halaman web yang mengandung informasi. Sedangkan menurut Rianto (2007), web adalah fasilitas *hypertext* yang mampu menampilkan data berupa teks, gambar, suara, animasi dan multimedia lainnya, dimana diantara data-data tersebut saling terkait dan berhubungan satu dengan yang lainnya. Untuk memudahkan dalam membaca data tersebut dibutuhkan sebuah *browser* seperti *Internet Explorer*, *Opera*, *Google Chrome* ataupun *Mozilla Firefox*.

Proses kerja saat pemanggilan halaman web terbagi menjadi dua bagian, yaitu proses pada sisi klien atau dikenal dengan istilah *Client Side* dan proses pada sisi *server* atau dikenal dengan istilah *Server Side*.

1. *Client Side Scripting*

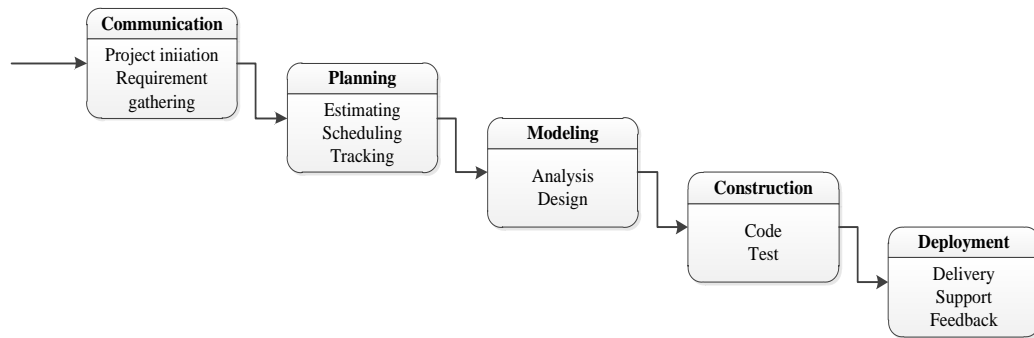
Client Side Scripting adalah bahasa pemrograman yang proses pengolahannya dilakukan di komputer pengunjung (*Client*). Ketika seseorang ingin melihat suatu website, maka website yang bertipe ini akan sepenuhnya di download dan diproses di komputer masing-masing sehingga siapapun yang melihat website kita bisa mengambil seluruh *script* yang ditulis oleh pembuatnya. Web yang menggunakan program ini biasanya website statis, yaitu yang tampilannya hampir tidak berubah, kecuali *website* tersebut dirombak pada halamannya (misal: web bertipe *.html). Contoh *Client Side Scripting* adalah HTML (*Hypertext Markup Language*), CSS (*Cascading Style Sheet*), JavaScript, XML (*Extensible Markup Language*).

2. *Server Side Scripting*

Server Side Scripting adalah kebalikan dari *Client Side Scripting* yaitu bahasa pemrograman web yang pengolahannya dilakukan di komputer *server* dan hasil pengolahannya dikirimkan ke komputer *client*/pengunjung dalam bentuk bahasa html, sehingga pengunjung tidak mengetahui *script* yang telah ditulis oleh pembuatnya. Web tipe ini bersifat dinamis yaitu bisa dirubah kontennya dengan mudah tanpa mengubah *script*. Bisa digunakan untuk berbagai kebutuhan web seperti *online shop*, forum, web sekolah, *blogging* dan berbagai kebutuhan lainnya. Contoh *Server Side Scripting* adalah: PHP, ASP, ASP. Net, Java Server Page.

2.5 Metode Pengembangan SDLC (*Systems Development Life Cycle*)

Menurut Pressman (2014), nama lain dari Model *Waterfall* adalah Model Air Terjun dan kadang dinamakan siklus hidup klasik (*classic life cycle*), dimana hal ini menyiratkan pendekatan yang sistematis dan berurutan (sekuensial) pada pengembangan perangkat lunak. Pengembangan perangkat lunak dimulai dari spesifikasi kebutuhan pengguna dan berlanjut melalui tahapan-tahapan perencanaan (*planning*), pemodelan (*modelling*), konstruksi (*construction*), serta penyerahan sistem perangkat lunak ke para pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan berkelanjutan pada perangkat lunak yang dihasilkan.



Gambar 2.1 Model pengembangan *Waterfall* (Pressman, 2014)

Gambar 2.1 menunjukkan tahapan umum dari model proses *waterfall*.

Model ini disebut dengan *waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan. Akan tetapi, Pressman (2014) memecah model ini meskipun secara garis besar sama dengan tahapan-tahapan model *waterfall* pada umumnya.

Model ini merupakan model yang paling banyak dipakai dalam *Software Engineering*. Model ini melakukan pendekatan secara sistematis dan urut mulai dari level kebutuhan sistem lalu menuju ke tahap *Communication*, *Planning*, *Modeling*, *Construction*, dan *Deployment*.

Berikut ini adalah penjelasan dari tahap-tahap yang dilakukan di dalam Model *Waterfall* menurut Pressman (2014) :

a. *Communication*

Langkah pertama diawali dengan komunikasi kepada konsumen/pengguna. Langkah awal ini merupakan langkah penting karena menyangkut pengumpulan informasi tentang kebutuhan konsumen/pengguna.

b. *Planning*

Setelah proses *communication* ini, kemudian menetapkan rencana untuk pengerjaan *software* yang meliputi 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 ke sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat *coding*. Proses ini berfokus pada rancangan struktur data, arsitektur *software*, representasi *interface*, dan detail (algoritma) prosedural.

d. *Construction*

Construction merupakan proses membuat kode (*code generation*). *Coding* atau pengkodean merupakan penerjemahan desain dalam bahasa yang bisa dikenali oleh komputer. *Programmer* akan menerjemahkan transaksi yang diminta oleh *user*. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu *software*, artinya penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan *testing* terhadap sistem yang telah dibuat. Tujuan *testing* adalah menemukan kesalahan-kesalahan terhadap sistem tersebut untuk kemudian bisa diperbaiki.

e. *Deployment*

Tahapan ini bisa dikatakan final dalam pembuatan sebuah *software* atau sistem. Setelah melakukan analisis, desain dan pengkodean maka sistem yang sudah jadi akan digunakan *user*. Kemudian *software* yang telah dibuat harus dilakukan pemeliharaan secara berkala.

2.6 Kebutuhan Perangkat Lunak

Dalam menentukan kebutuhan perangkat lunak, yang pertama perlu harus diperhatikan setelah definisi dari kebutuhan perangkat lunak, adalah jenis dari kebutuhan tersebut seperti apakah produk atau proses. Keseluruhan proses tersebut dapat menjelaskan perbedaan antara kebutuhan sistem dan perangkat lunak.

2.7 Analisis dan Desain Perangkat Lunak

Analisis sistem atau perangkat lunak dilakukan dengan tujuan untuk dapat mengidentifikasi dan mengevaluasi permasalahan yang terjadi dan kebutuhan yang diharapkan, sehingga dapat diusulkan perbaikannya.

Perancangan desain perangkat lunak merupakan penguraian suatu sistem informasi yang utuh ke dalam bagian komputerisasi yang dimaksud, mengidentifikasi dan mengevaluasi permasalahan, menentukan kriteria, menghitung konsistensi terhadap kriteria yang ada, serta mendapatkan hasil atau tujuan dari masalah tersebut serta mengimplementasikan seluruh kebutuhan operasional dalam membangun aplikasi.

Tahap analisis merupakan tahap yang kritis dan sangat penting, karena kesalahan di dalam tahap ini juga akan menyebabkan kesalahan di tahap selanjutnya. Dalam tahap analisis sistem terdapat langkah-langkah dasar yang harus dilakukan oleh analis sistem sebagai berikut:

1. *Identify*, yaitu mengidentifikasi masalah.
2. *Understand*, yaitu memahami kerja dari sistem yang ada.
3. *Analyze*, yaitu menganalisis sistem.
4. *Report*, yaitu membuat laporan hasil analisis.

Setelah tahap analisis sistem selesai dilakukan, maka analisis sistem telah mendapatkan gambaran dengan jelas apa yang harus dikerjakan. Tiba waktunya sekarang bagi analisis sistem untuk memikirkan bagaimana membentuk sistem tersebut, tahap ini disebut desain sistem atau perangkat lunak.

2.8 Konstruksi Perangkat Lunak

Pada tahap ini ialah melakukan konversi hasil desain ke sistem informasi yang lengkap melalui tahapan *coding* atau pengkodean termasuk bagaimana, membuat basis data dan menyiapkan prosedur kasus pengujian, mempersiapkan berkas atau *file* pengujian, pengkodean, pengompilasian, memperbaiki dan membersihkan program serta melakukan peminjaman pengujian. *Construction* ini memiliki beberapa tahapan secara umum.

Desain implementasi yang digunakan, bahasa pemrograman yang digunakan, kualitas dari implementasi yang dilakukan, proses pengetesan dan integritas. Dalam proses pengimplementasian penelitian ini, digunakan bahasa pemrograman dan beberapa aplikasi pendukung yaitu:

a. PHP

PHP merupakan singkatan dari “*PHP: Hypertext Preprocessor*”, adalah sebuah bahasa *script* berjenis *server side* yang menyatu dengan HTML. Sintaks dan perintah-perintah yang dimasukkan akan sepenuhnya dijalankan dan dikerjakan oleh server dan disertai pada halaman HTML biasa. PHP bertujuan untuk membuat aplikasi-aplikasi yang dijalankan di atas teknologi web. Dalam hal ini, aplikasi pada umumnya akan memberikan hasil pada *web browser*, tetapi prosesnya secara keseluruhan dijalankan dan dikerjakan di *web server*.

b. Adobe Dreamweaver

Adobe Dreamweaver adalah aplikasi desain dan pengembangan *web* yang menyediakan *editor visual* dan kode editor dengan fitur standar seperti *syntax*, *code completion*, dan *code collapsing* serta fitur lebih canggih seperti *real-time syntax checking* dan *code introspection* untuk menghasilkan petunjuk kode untuk membantu pengguna dalam menulis kode. Tata letak tampilan pada Adobe Dreamweaver memungkinkan pengguna dengan cepat membuat tata letak dan manipulasi elemen HTML. Adobe Dreamweaver memiliki fitur browser yang terintegrasi untuk melihat halaman web yang dikembangkan di jendela pratinjau program sendiri agar konten memungkinkan untuk terbuka di web browser yang telah terinstal.

c. MySQL

MySQL adalah sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya; SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

2.9 Uji Coba Perangkat Lunak

Uji coba perangkat lunak meliputi verifikasi yang dinamis dari tingkah laku sebuah perangkat lunak yang diwakili oleh beberapa contoh kasus uji coba. Kasus uji coba tersebut dilakukan dengan memberikan masukan kepada perangkat lunak agar muncul tingkah laku/reaksi yang diharapkan, begitu pula sebaliknya. Dalam uji coba perangkat lunak, yang pertama kali diperhatikan adalah fundamental dari uji coba perangkat lunak tersebut. Di dalamnya dijelaskan mengenai terminologi dari uji coba terkait, kunci masalah dari uji coba, dan hubungan uji coba tersebut dengan aktifitas lainnya di dalam perangkat lunak tersebut. Kedua, yang perlu diperhatikan adalah tingkatan dari uji coba. Di dalamnya dijelaskan tentang target dari uji coba dan tujuan dari uji coba tersebut. Ketiga, yang perlu diperhatikan adalah teknik dari uji coba. Di dalamnya meliputi uji coba berdasarkan intuisi dan pengalaman dari seorang *tester*, diikuti oleh teknik berdasarkan spesifikasi, teknik berdasarkan kode, teknik berdasarkan kesalahan, teknik berdasarkan penggunaan, dan teknik dasar yang relatif tergantung dari aplikasi tersebut. Keempat, yang perlu diperhatikan adalah pengukuran dari uji coba terkait. Di dalamnya dijelaskan bahwa pengukuran tersebut dikelompokkan menjadi dua, yakni yang berhubungan dengan evaluasi ketika uji coba dilakukan serta ketika uji coba selesai dilakukan. Kelima, yang perlu diperhatikan adalah proses uji coba itu sendiri, yang berisi tentang pertimbangan praktis dan aktifitas uji coba.

2.9.1 Tingkatan Uji Coba Perangkat Lunak

Menurut Romeo (2003) terdapat tiga tingkatan testing secara umum, yaitu:

1. *Unit Testing*: testing penulisan kode-kode program dalam satuan unit terkecil secara individual.

Praktik *Unit Testing* secara umum:

- a. Tujuan: konfirmasi bahwa modul telah dikode dengan benar.
- b. Pelaku: biasanya *programer*.
- c. Apa yang dites:
 1. Fungsi (*Black Box*).
 2. Kode (*White Box*).
 3. Kondisi ekstrim dan batasan-batasan.
- d. Kapan selesai: biasanya saat *programer* telah merasa puas dan tidak diketahui lagi kesalahan.
- e. Alat bantu: tidak biasa digunakan.
- f. Data: biasanya tidak didata.

2. *System Testing*: proses testing pada sistem terintegrasi untuk melakukan verifikasi bahwa sistem telah sesuai spesifikasi.

Praktik *System Testing* secara umum:

- a. Tujuan: merakit modul menjadi suatu sistem yang bekerja. Dan menentukan kesiapan untuk melakukan *Acceptance Test*.
- b. Pelaku: pemimpin tim atau grup tes.
- c. Apa yang dites:
 1. Kebutuhan dan fungsi sistem.

2. Antar muka sistem.
- d. Kapan selesai: biasanya bila mayoritas kebutuhan telah sesuai dan tidak ada kesalahan mayor yang ditemukan.
- e. Alat bantu:
 1. Sistem pustaka dan pustaka *test case*.
 2. *Generator*, komparator dan simulator *data testing*.
- f. Data:
 1. Data kesalahan yang ditemukan.
 2. *Test case*.
 3. *Acceptance Testing*: testing formal yang dilakukan untuk menentukan apakah sistem telah memenuhi kriteria penerimaan dan memberdayakan pelanggan untuk menentukan apakah sistem dapat diterima atau tidak.
Praktik *Acceptance Testing* secara umum:
 - a. Tujuan: mengevaluasi kesiapan untuk digunakan.
 - b. Pelaku: pengguna akhir atau agen.
 - c. Apa yang dites:
 1. Fungsi mayor.
 2. Dokumentasi.
 3. Prosedur.
 - d. Kapan selesai: biasanya bila pengguna telah merasa puas atau tes berjalan dengan lancar / sukses.
 - e. Alat bantu: komparator.
 - f. Data: formalitas dokumen.

2.9.2 Pengujian *Black Box*

Menurut Pressman (2002: 532), pengujian *black box* adalah proses eksekusi suatu program dengan maksud menemukan kesalahan. Menurut Pressman (2002: 577), teknik pengujian *black box* adalah yang paling lazim selama integrasi. Pengujian *black box* digunakan untuk memperlihatkan bahwa fungsi – fungsi perangkat lunak adalah operasional bahwa *input* diterima dengan baik dan *output* dihasilkan dengan tepat.

2.9.3 *Web Application Performance Test*

Pada penelitian ini, *application testing* pada aplikasi yang dibuat yaitu dengan menggunakan tools WAPT (*Web Application Performance Test*) dengan menggunakan jenis simulasi *stress testing*. WAPT (*Web Application Performance Testing*) adalah sebuah *tool software testing* yang digunakan untuk pengujian tingkat *stress* pada suatu aplikasi web. *Stress testing* adalah pengujian yang dilakukan untuk mengetahui kemampuan perangkat lunak dalam menangani kondisi yang tidak normal (mencakup kuantitas/volume). Dari segi sistem, salah satunya adalah memastikan aplikasi web yang kita luncurkan bisa menangani pengakses aplikasi tersebut sejumlah yang kita targetkan. Untuk menentukan jumlah pengakses bukanlah hal mudah, tapi kita bisa melakukan pengujian sebelum aplikasi diluncurkan, dengan melakukan *stress testing*. Dengan hasil test tersebut, kita bisa mengetahui performa aplikasi web kita dan bisa memperkirakan dengan infrastruktur yang kita miliki sekarang apakah layanan akan berfungsi dengan baik atau tidak saat sistem diluncurkan untuk diakses oleh *user*.