

BAB II

TINJAUAN PUSTAKA

2.1 Landasan Teori

2.1.1 Website Penjualan

Menurut Nugroho (2016:718) *website* penjualan (*E-commerce*) adalah penyebaran, pembelian, penjualan, pemasaran barang dan jasa melalui sistem elektronik seperti internet atau televisi, www, atau jaringan komputer lainnya. *E-commerce* dapat melibatkan transfer dana elektronik, pertukaran data elektronik, sistem manajemen inventori otomatis, dan sistem pengumpulan

2.1.1.1 Website Statis

Web Statis adalah *web* yang isinya atau *content* tidak berubah-ubah. Maksudnya adalah isi dari dokumen *web* tersebut tidak dapat diubah secara cepat dan mudah. Ini karena teknologi yang digunakan untuk membuat dokumen *web* ini tidak memungkinkan dilakukan perubahan isi atau data. Teknologi yang digunakan untuk *web* statis adalah jenis *Client Side Scripting* seperti HTML, *Cascading Style Sheet* (CSS).

2.1.1.2 Website Dinamis

Web Dinamis adalah jenis *web* yang *content* atau isinya dapat berubah-ubah setiap saat. Untuk membuat *web dinamis* diperlukan beberapa komponen yaitu *Client Side Scripting* (HTML, *Javascript*, *Cascading Style Sheet*), *Server Side*

Scripting seperti PHP, program basis data seperti MySQL untuk menyimpan data-datanya.

2.1.2 Pemrograman Struktur

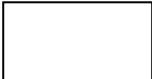
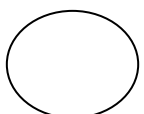
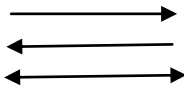

Menurut Jogiyanto (2010:53) dalam bukunya Analisis & Desain, menjelaskan bahwa: pendekatan terstruktur ini dimulai dari awal tahun 1970. Pendekatan terstruktur dilengkapi dengan alat-alat dan teknik-teknik yang dibutuhkan dalam pengembangan sistem, sehingga hasil akhir dari sistem yang dikembangkan akan didapatkan sistem yang strukturnya didefinisikan dengan baik dan jelas.

2.1.3 Alat Pengembang

2.1.3.1 DFD (*Data Flow Diagram*)

Menurut Santoso dan Nurmalina (2017:86) *Data Flow Diagram (DFD)* merupakan suatu diagram yang menggambarkan alir data dalam suatu entitas ke sistem atau sistem ke entitas. *DFD* juga dapat diartikan sebagai teknik grafis yang menggambarkan alir data dari input atau masukan menuju atau output. Istilah dalam bahasa indonesianya adalah diagram aliran data yang dapat dilihat pada tabel 2.

Tabel 2.1 Simbol *Data Flow Diagram (DFD)*

Yourdan / De Marco	Keterangan
Entitas Eksternal 	Entitas eksternal, dapat berupa orang atau unit terkait yang berinteraksi dengan sistem tetapi diluar sistem.
Proses 	Proses orang, unit yang mempergunakan atau melakukan transformasi data, komponen fisik tidak diidentifikasi.
Alir Data 	Alir data dengan arah khusus dari sumber ke tujuan
<u>Data Store</u> 	Penyimpanan data atau tempat data <i>direfer</i> oleh proses.



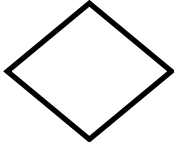

Sumber : Santoso, Nurmalina (2017 : 86)

2.1.3.2 ERD (*Entity Relationship Diagram*)

Menurut Santoso dan Nurmalina (2017 : 87) *Entity Relationship Diagram* adalah gambar atau diagram yang menunjukkan informasi dibuat, disimpan, dan digunakan dalam sistem bisnis.

Simbol desain sistem ERD dapat dilihat pada tabel 3 berikut:

Tabel 2.2 Simbol *Entity Relationship Diagram*

Nama	Simbol	Keterangan
Entitas		Persegi panjang menyatakan himpunan entitas adalah orang, atau berada dimana data akan dikumpulkan
Atribut		Atribut merupakan informasi yang diambil tentang sebuah entitas
Relasi		Belah ketupat menyatakan himpunan relasi merupakan hubungan antar entitas
Link		Garis sebagai penghubung antar himpunan, relasi, dan himpunan entitas dengan atributnya.






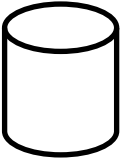



Sumber : Santoso, Nurmalina (2017 : 87)

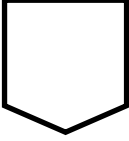
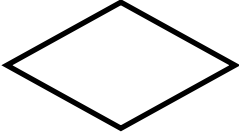
2.1.3.3 *FlowChart* (Bagan Alir)

Menurut Saputra (2017:106), *Flowchart* (Bagan Alir) merupakan suatu diagram yang menggambarkan alur kerja dari suatu sistem.

Pengertian diatas dapat disimpulkan bahwa *flowchart* (Bagan Alir) adalah sekumpulan simbol-simbol yang menunjukkan atau menggambarkan rangkaian kegiatan program dari awal hingga akhir. Lambang-lambang *flowchart* (Bagan Alir) yang digunakan ada 11 yaitu :

Tabel 2.3 Simbol *Flowchart* (Bagan Alir)

Nama	Simbol	Keterangan
Dokumen		Dokumen atau laporan: dokumen tersebut dipersiapkan dengan tulisan tangan, atau dicetak dengan komputer.
Multidokumen		Digambarkan dengan cara penumpukan simbol dokumen.
Pengertian <i>on-line</i>		Masukkan (<i>entry</i>) data melalui peralatan <i>on-line</i> seperti terminal/personal komputer.
Pemrosesan computer		Fungsi pemrosesan yang dilaksanakan dengan komputer, biasanya menghasilkan perubahan atas data atau informasi.
Proses Manual		Pelaksanaan pemrosesan yang dilaksanakan secara manual.
Disk Magnetis		Data disimpan secara permanen di dalam disk magnetis, dipergunakan untuk <i>file</i> utama (<i>master file</i>) dan <i>database</i> .
Penyimpanan <i>on-line</i>		Data disimpan di dalam file <i>on-line temporer</i> melalui media yang dapat diakses secara langsung, seperti <i>disk</i> .
Arus dokumen atau proses		Arah pemrosesan atau arus dokumen, arus yang normal berada dibawah dan mengarah ke kanan.
<i>On-page</i> conector		Menghubungkan arus pemrosesan di satu halaman yang sama.




Nama	Simbol	Keterangan
<i>Off-page</i> conector		Suatu penanda masuk dari, atau keluar ke halaman lain.
Keputusan		Langkah pengambilan keputusan, dipergunakan dalam sebuah program komputer bagan alir.

Sumber : Saputra (2017:106)

2.1.3.4 STD (*State Transition Diagram*)

Menurut Pressman (2002:354) *State Transition Diagram* (STD) menunjukkan bagaimana sistem bertingkah laku sebagai akibat dari kejadian eksternal. Untuk melakukannya, STD menunjukkan berbagai model tingkah laku (disebut *state*) sistem dan cara di mana transisi dibuat dari *state* satu ke *state* lainnya. STD berfungsi sebagai dasar bagi permodelan tingkah laku.

Tabel 2.4 Simbol *State Transition Diagram* (STD)

Notasi	Keterangan
	State, menyatakan status
	Transisi, menyatakan perubahan status
	Kondisi, penyebab perubahan keadaan (status), sedangkan aksi yang harus dilakukan ketika akan berubah keadaan (status)

Sumber : Pressman (2002:354)

2.1.3.5 Kamus Data (*Data Dictionary*)

Menurut Sutabri (2004:170) kamus data adalah katalog fakta tentang data dari kebutuhan-kebutuhan informasi dari suatu sistem informasi. Kamus data juga mempunyai suatu bentuk untuk mempersingkat arti dari simbol-simbol yang dijelaskan, yang disebut notasi.

Tabel 2.5 Simbol *Data Dictionary* (Kamus Data)

Notasi	Arti
=	Terdiri dari, terbentuk dari, sama dengan
+	Dan
()	Optional
{ }	Iterasi/pengulangan, missal 1 {...} 10
[]	Pilih satu dari beberapa alternative (pilihan) missal : [AIBICIDI]
**	Komentar
@	Identifier suatu data store
	Pemisah dalam bentuk []

Sumber : Sutabri (2004:170)

2.1.4 Metode Pengembang Perangkat Lunak

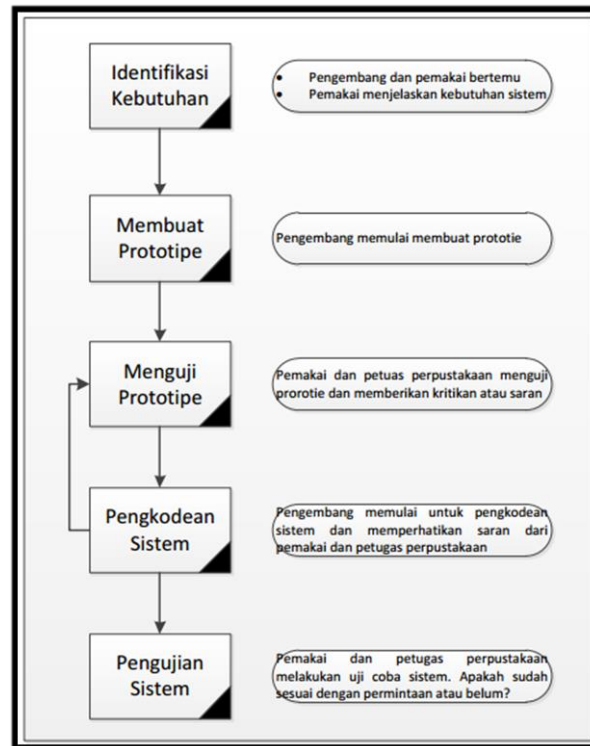
Secara prinsip metode pembangunan perangkat lunak bertujuan untuk membantu menghasilkan perangkat lunak yang berkualitas. Metode pengembangan perangkat lunak (atau disebut juga model proses atau paradigm rekayasa perangkat lunak) adalah suatu strategi pengembangan yang memadukan proses, metode, dan perangkat (*tools*).

Metode-metode pengembangan perangkat lunak secara umum ada 4 diantaranya yaitu metode *sekuensial linier* atau *waterfall*, *prototype*, *RAD*, *Model Evolutionary Development*. Dalam pembuatan website ini Model yang digunakan dalam proses pengembangan untuk membangun sistem aplikasi ini yaitu metode *prototype*.

2.1.4.1 Prototype

Menurut Afrina (2012:431), *Prototype* merupakan metode pengembangan sistem yang menggunakan pendekatan untuk membuat suatu program dengan cepat dan bertahap sehingga

segera dapat dievaluasi oleh pemakai. Berikut tahapan-tahapan dalam metode *prototyping* seperti terlihat pada gambar 1.



Sumber: Afrina (2012:431)

Gambar 2.1 Metode *Prototyping*

Adapun tahapan – tahapan *prototyping* menurut gambar 1 adalah sebagai berikut:

a. Identifikasi kebutuhan pemakai

Tahap ini adalah tahap awal yaitu mengidentifikasi kebutuhan pemakai. Pada tahap ini akan dilakukan komunikasi antara pengembang dan pemakai untuk mengidentifikasi kebutuhan-kebutuhan yang diperlukan untuk membangun sistem

b. Membangun *Prototyping*

Setelah mendapatkan data dari berbagai sumber, langkah selanjutnya membangun prototyping awal, sebagai gambar sistem yang akan di bangun.

c. Menguji *Prototyping*

Setelah membangun prototyping, maka gambaran sistem yang akan dibangun dievaluasi dengan tujuan untuk mendapatkan saran dan masukan dari pemakai. Jika pada tahap ini masih ada revisi maka harus dilakukan perbaikan.

d. Pengkodean system

Dalam tahap ini prototyping yang sudah disepakati selanjutnya diterjemahkan ke dalam bahasa pemrograman yang sesuai yaitu PHP dan database menggunakan MySQL.

e. Pengujian system

Setelah selesai di bangun dengan pemrograman, selanjutnya dilakukan proses pengujian sistem menggunakan metode *blackbox* dilakukan oleh pemakai.

Adapun kelebihan dan kekurangan metode *Prototype* menurut

Fathiah (2014 : 3) Kelebihanyaitu :

- 1.) *User* dapat melihat secara langsung perkembangan sistem seiring dengan permintaannya,
- 2.) *Developer* belajar langsung mengenai kebutuhan sistem dari customer/user.

- 3.) Hasil produk yang lebih akurat (lebih sesuai dengan permintaan *user*).
- 4.) Desain sistem lebih fleksibel.
- 5.) Interaktif dengan adanya simulasi *prototype*
- 6.) Untuk pengembangan lebih lanjut (jika terjadi perubahan), *developer* hanya perlu mengubah *prototype*.
- 7.) Jika *customer* sudah "puas", *prototype* dibuat menjadi sistem secara sempurna untuk dijadikan 'Final Product'.

Kekurangan dari metode *Prototype* yaitu :

- 1.) Proses bisa jadi berlanjut terus menerus tanpa henti(mengikuti keinginan *customer*).
- 2.) Bisa jadi *customer* malah menginginkan *prototype* sistem dikirim.
- 3.) Reputasi yang buruk sebagai sebuah metode yang bersifat "Quick-and-Dirty".
- 4.) Kemungkinan perawatan secara keseluruhan bisa saja terabaikan.
- 5.) Pengembangan yang berlebihan untuk *prototype*.

2.1.5 Teknik Pengujian Perangkat Lunak

Menurut Mustaqbal (2015 : 31) merupakan suatu proses pelaksanaan suatu program dengan tujuan menemukan suatu kesalahan. Suatu kasus test yang baik adalah apabila test tersebut mempunyai kemungkinan menemukan sebuah kesalahan yang tidak terungkap.

Suatu test yang sukses adalah bila test tersebut membongkar suatu kesalahan yang awalnya tidak ditemukan.

Dalam teknik pengujian perangkat lunak terdapat dua jenis teknik yaitu *White Box* dan *Black Box*. Pada penelitian ini dicoba terapkan pengujian dengan menggunakan teknik *Black Box testing*.

2.1.5.1 Blackbox Testing

Menurut Mustaqbal, dkk, (2015 : 34), *Black Box Testing* berfokus pada spesifikasi fungsional dari perangkat lunak. *Tester* dapat mendefinisikan kumpulan kondisi *input* dan melakukan pengetesan pada spesifikasi *fungsional* program.

Black Box Testing bukanlah solusi alternatif dari *White Box Testing* tapi lebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh *White Box Testing*.

Black Box Testing cenderung untuk menemukan hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (*interface errors*).
3. Kesalahan pada struktur data dan akses basis data.
4. Kesalahan performansi (*performance errors*).
5. Kesalahan inisialisasi dan terminasi.

2.1.6 Penelitian Terdahulu

Hasil penelitian terdahulu digunakan sebagai pedoman dasar, acuan, pertimbangan, maupun perbandingan bagi penelitian terbaru yang

sejenis, adapun penelitian terdahulu yang penulis gunakan seperti pada tabel 2.6 berikut:

Tabel 2.6 Hasil Penelitian Terdahulu

No	Judul	Penulis	Hasil
1	Perancangan Sistem Informasi Penjualan Online Studi Kasus Tokoku Jurnal SIMETRIS, Vol 7 No 2 November 2016 ISSN: 2252-4983	Fauyhi Eko Nugroho	Menghasilkan sebuah rancangan sistem informasi penjualan online berbasis web dengan menerapkan <i>object oriented programming</i> (OOP). metode yang digunakan dalam penelitian ini adalah metode <i>Rational Unified Process</i> (RUP) dengan tujuan untuk membangun sistem yang dapat membantu Tokoku dalam meningkatkan penjualannya.
2	Perancangan Sistem Informasi Penjualan Berbasis Web Dalam Memasarkan Mobil Bekas Citec Journal, Vol. 3, No. 1, November 2015 - Januari 2016 ISSN: 2354-5771	Sandy Kosasi	Menghasilkan sebuah rancangan sistem informasi penjualan berbasis web dengan menerapkan <i>object oriented programming</i> (OOP). metode yang digunakan dalam penelitian ini adalah metode ICDM (<i>Internet Commerce Development Methodology</i>) dengan tujuan untuk membangun sistem yang memudahkan konsumen dan tidak lagi bergantung kepada lokasi serta mudah memperluas pangsa pasar.
3	Perancangan Aplikasi Penjualan Berbasis Web Dengan Metode Prototyping Pada CV Khatulistiwa Jurnal e-Proceeding of Engineering Fakultas Rekayasa industri Vol.2, No.1 April 2015	Izharyan Iqbal, R.Wahyu Witjaksono, M.Teguh Kurniawan	Menghasilkan sebuah rancangan aplikasi penjualan berbasis web dengan menerapkan <i>object oriented programming</i> (OOP) yaitu <i>use case</i> . metode yang digunakan dalam penelitian ini adalah metode <i>prototyping</i> dengan tujuan untuk merancang sebuah aplikasi yang dapat meningkatkan penjualan dan perluasan <i>marketing</i> produk.

No	Judul	Penulis	Hasil
	ISSN : 2355-9365		

Dari penelitian terdahulu dan yang dilakukan penulis sekarang dapat disimpulkan bahwa memiliki sebuah perbedaan dan persamaan perbedaannya adalah penelitian 1,2, dan 3 menggunakan pemodelan UML seperti *use case*, sedangkan penulis menggunakan pemodelan struktur seperti DFD dan ERD. Lalu dari segi metode, penulis menggunakan metode *prototype*, sedangkan penelitian 1 menggunakan metode RUP (*Rational Unified Process*) dan penelitian 2 menggunakan metode ICDM (*Internet Commerce Development Methodology*).

Persamaannya adalah penulis dan penelitian 3 sama-sama menggunakan metode *prototype* dan penelitian 1,2, dan 3 memiliki tujuan yang sama seperti penulis, yaitu membangun penjualan online berbasis *web*.

2.2 Objek Penelitian

2.2.1 Sejarah Perusahaan

perusahaan PT Kumala Jaya Lestari berdiri sejak tahun 1990, yang dulu merupakan Kumala Motor dan kemudian berubah menjadi PT Kumala Jaya Lestari pada tahun 2001 dan bergerak dibidang penjualan Spare Part mobil, perusahaan melayani penjualan untuk masyarakat dan bengkel-bengkel kecil. Sejak Beroperasi PT Kumala Jaya Lestari

berkomitmen memberi pelayanan yang terbaik untuk masyarakat dengan produk yang sesuai standar.

2.2.2 Visi dan Misi

Visi PT Kumala Jaya Lestari

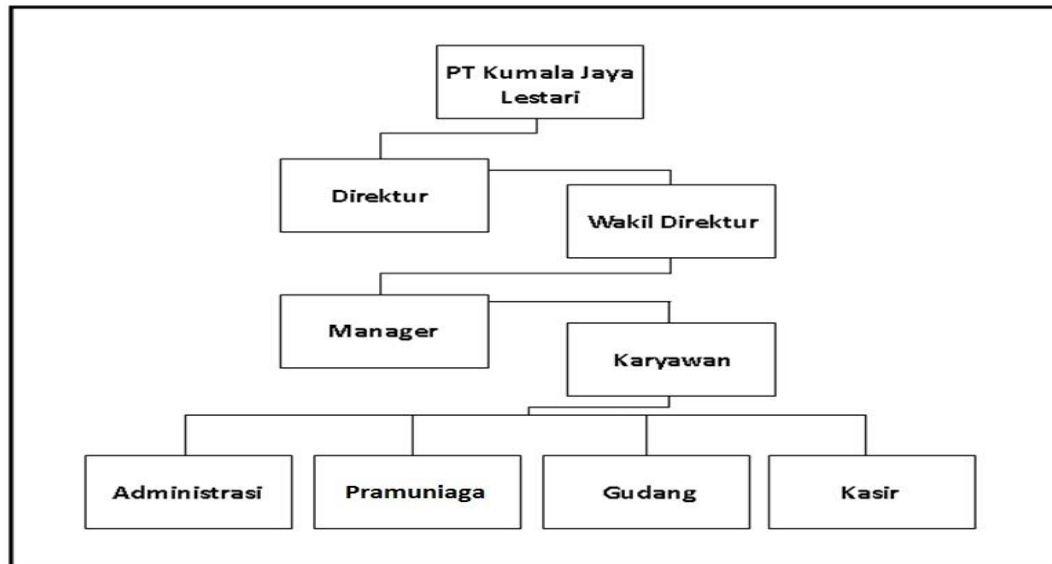
Adapun visi dari PT Kumala Jaya Lestari yaitu menjadikan perusahaan perdagangan spare part terbaik dan terlengkap yang dikelola secara professional.

Misi PT Kumala Jaya Lestari

Adapun misi dari PT Kumala Jaya Lestari yaitu selalu memberikan pelayanan yang terbaik bagi konsumen dan berusaha untuk menjadi perusahaan penyedia spare part mobil yang terdepan. Menyediakan spare part lengkap, berkualitas, dan layanan jual yang baik serta mengembangkan variasi produk yang inovatif untuk melayani dan memuaskan pelanggan.

2.2.3 Struktur Organisasi

Adapun struktur organisasi pada PT Kumala Jaya Lestari dilihat pada gambar 2.2



Sumber : PT Kumala Jaya Lestari

Gambar 2.2 Struktur Organisasi

Berdasarkan struktur organisasi gambar 2.2 divisi atau bagian yang terlibat langsung dalam sistem yang dibangun yaitu divisi kasir, divisi gudang, divisi pramuniaga, divisi administrasi, dan manager.

2.2.4 Tugas Wewenang

1. Direktur dan Wakil Direktur

Tugas dari direktur dan wakil direktur adalah bertanggung jawab atas semua kegiatan yang ada pada perusahaan

2. Manager

Tugas dari manager adalah mengerjakan dan mengawasi serta ikut bertanggung jawab atas seluruh kegiatan di dalam perusahaan.

3. Divisi Administrasi

Tugas dari divisi administrasi adalah mengurus nota hutang piutang, mendata barang masuk, mengurus barang retur, menginput harga, serta mendata absensi karyawan.

4. Divisi Pramuniaga

Tugas dari divisi pramuniaga adalah melayani para pelanggan yang datang ke perusahaan.

5. Divisi Gudang

Tugas dari divisi gudang adalah mengecek stok barang yang ada di perusahaan.

6. Divisi Kasir

Tugas dari divisi kasir adalah menerima pembayaran dari pelanggan serta mendata barang yang terjual.