

BAB II

DASAR TEORI

2.1. Tinjauan Pustaka

Penelitian sebelumnya yang dilakukan oleh Rudi (2020) yang berjudul “ Sistem Informasi Servis dan Penjualan *Sparepart* berbasis visual” sistem ini membahas tentang suatu sistem aplikasi yang dapat membantu dalam pengolahan data servis dan penjualan sparepart motor secara efektif dan efisien. Dengan menggunakan alat bantu komputer berbasis visual. Sistem aplikasi yang di bangun ini dapat memberikan kemudahan dalam proses pengolahan data servis dan penjualan sparepart motor pada bengkel [1].

Penelitian lain yang dilakukan oleh Adi Ahmad (2021) yang berjudul “ Perancangan Ssistem Informasi Servis kendaraan bermotor ” yang membahas tentang pembuatan sistem informasi manajemen atau teknologi informasi berbasis komputer yang penulis harapkan bisa memberikan pemecahan masalah terhadap segala proses pengolahan administrasi seperti pendataan pelanggan , pendataan transaksi servis dan penjualan sparepart serta pembuatan laporan – laporan yang di perlukan oleh pihak – pihak bersangkutan [2].

Penelitian lain yang dilakukan oleh Julianto Simatupang (2020) yang berjudul “ Implementasi Sistem Informasi Booking Servis Online Pada PT. Riau Argo Perkasa Berbasis Web” yang membahas tentang sistem informasi booking servis yang dapat menangani masalah pada pembookingan servis dan pemilihan jadwal. Sehingga pengaturan dapat dilakukan secara terpusat dan mempermudah pelanggan dalam mencari informasi servis dan melakukan pembookingan servis [3].

Penelitian lain yang dilakukan oleh Fajar Trisnadi (2020) yang berjudul “ Perancangan Aplikasi Pendaftaran Servis Motor Dan Penjualan Spare Part Pada Bengkel Jali” yang membahas tentang sistem yang dapat mendukung semua kegiatan yang dilakukan oleh Begkel Jali Motor agar menjadi lebih mudah khususnya dalam penjualan spare part dan servis kendaraan. Tujuan dari penelitian ini adalah merancang sistem yang masih manual agar terkomputerisasi dengan baik sehingga dapat mempermudah pihak Bengkel dalam mengolah data, pencarian data, meminimalisir terjadi redundansi data, mempermudah transaksi servis,

serta membuat laporan yang lebih efektif dan efisien dengan menggunakan Bahasa pemrograman JAVA dengan *MySQL* sebagai database [4].

Penelitian lain yang dilakukan oleh Beby Rohmah Udmi Annidah (2021) yang berjudul “Perancangan sistem informasi servis motor pada bengkel arif motor” yang membahas tentang sistem aplikasi servis serta pengeluaran sparepart yang dapat membantu proses keakuratan data servis serta membuat suatu sistem pengolahan data sparepart dan jasa servis yang dapat digunakan untuk meningkatkan kinerja pelayanan. Sistem ini mempunyai gambaran tentang sistem yang sedang berjalan dan mengetahui masalah yang dihadapi sebagai acuan pembuat sistem yang lebih efektif dan efisien, merancang aplikasi servis motor sebagai fasilitas yang dapat membantu proses penjualan *sparepart*, persediaan *sparepart*, dan jasa yang digunakan untuk servis dan mengimplementasikan sistem dengan menggunakan data yang ada pada bengkel [5].

“Sistem Informasi Servis Motor Berbasis Web” yang dibuat dapat memberikan solusi untuk pengolahan data servis motor secara praktis, cepat, akurat dan membuat pengolahan data menjadi tersusun dan tersimpan dengan baik. Sistem ini memberikan informasi data pelanggan yang melakukan pemesanan servis, data pemilik, data teknisi, data servis dan *sparepart*, data pelanggan, laporan keuangan yang akan dilaporkan kepada pemilik bengkel, dan data riwayat servis pelanggan yang sebelumnya pernah melakukan servis di bengkel motor Helios. Sistem ini dapat mempermudah admin maupun pemilik bengkel dalam proses pencarian data sehingga mempercepat proses pencarian data, serta menyimpan data dengan aman dan meminimalisir kesalahan dalam pengolahan data servis.

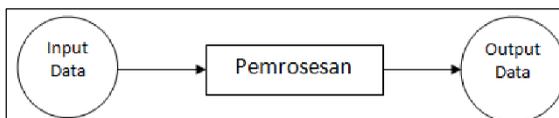
2.2. Landasan Teori

2.2.1. Sistem Informasi

Sistem adalah kumpulan dari bagian-bagian yang bekerja sama untuk mencapai tujuan yang sama. Bisa juga diartikan sekumpulan objek-objek yang saling berelasi dan berinteraksi serta hubungan antar objek bisa dilihat sebagai satu kesatuan yang dirancang untuk mencapai satu tujuan. Untuk memahami atau mengembangkan suatu sistem, maka perlu membedakan unsur-unsur dari sistem yang membentuknya.[6] Karakteristik tersebut adalah :

1. Batasan yaitu penggambaran dari suatu elemen yang termasuk didalam sistem dan mana yang diluar sistem.
2. Lingkungan yaitu segala sesuatu diluar sistem, lingkungan yang menyediakan asumsi, kendala, dan input terhadap suatu sistem.
3. Masukan yaitu sumber daya dari lingkungan yang dikonsumsi dan dimanipulasi oleh suatu sistem.
4. Keluaran yaitu sumber daya atau produk yang disediakan untuk lingkungan sistem oleh kegiatan dalam suatu sistem.
5. Komponen yaitu kegiatan-kegiatan atau proses yang mentransformasikan input menjadi bentuk setengah jadi.
6. Penghubung yaitu tempat dimana komponen atau sistem dan lingkungan bertemu atau berinteraksi.
7. Penyimpanan yaitu area yang dikuasai dan digunakan untuk penyimpanan sementara dan tetap dari informasi, energi, bahan baku, dan sebagainya.

Informasi dapat didefinisikan sebagai data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimaannya dan bermanfaat dalam pengambilan keputusan saat ini atau saat mendatang. Pengertian Sistem Informasi dapat didefinisikan sebagai suatu alat untuk menyajikan informasi dengan cara sedemikian rupa sehingga bermanfaat bagi penerimaannya. Sistem informasi berdasarkan konsep data digambarkan dalam gambar 2.1. sebagai berikut [6] :



Gambar 2.1. Konsep Sistem Informasi

2.2.2. Servis Motor

Setiap sepeda motor yang dioperasikan, pada akhirnya akan mengalami suatu keadaan dimana bagian-bagian dari sepeda motor tersebut (mesin, transmisi, rangka, dsb) mengalami kelelahan dan keausan sehingga mengurangi kinerjanya, diantaranya, tenaga mesin menurun, akselerasi lambat, bahan bakar boros, dan kemungkinan kerusakan berlanjut/merembet terhadap kerusakan komponen yang lainnya [7]. Apabila kondisi tersebut tidak ditanggulangi melalui perawatan berkala

kendaraan, maka kondisi tersebut akan meningkat ke arah kerusakan komponen yang bertambah parah dan membutuhkan dana yang cukup besar untuk mengembalikan sepeda motor pada kondisi semula.

Servis motor merupakan kegiatan perawatan berkala pada sepeda motor, dimana kegiatan ini meliputi [7] :

1. Memeriksa bagian-bagian sepeda motor untuk memastikan bagian tersebut masih berfungsi sebagaimana mestinya.
2. Membersihkan bagian yang kotor agar kotoran yang ada tidak merusak sistem.
3. Menyetel bagian yang berubah agar sesuai dengan spesifikasinya.
4. Memperbaiki atau mengganti komponen yang rusak atau aus.

Uraian rangkaian kegiatan yang dilakukan setiap melaksanakan servis sepeda motor adalah sebagai berikut [7]:

1. Bagian Mesin
 - a. Memeriksa dan merawat baterai.
 - b. Memeriksa dan mengganti oli pelumas mesin
 - c. Membersihkan saringan udara
 - d. Membersihkan saringan bahan bakar
 - e. Memeriksa dan menyetel busi
 - f. Membersihkan karburator
 - g. Menyetel katup
 - h. Menyetel campuran bahan bakar/putaran mesin
 - i. Menyetel kebebasan kopling
2. Bagian Kelistrikan
 - a. Memeriksa dan merawat baterai.
 - b. Memeriksa fungsi kelistrikan (bel, lampu tanda belok, lampu kepala, lampu rem, lampu indikator).
3. Bagian Chasis
 - a. Memeriksa dan menyetel gerak bebas rem.
 - b. Memeriksa, merawat dan menyetel gerak bebas rantai roda.
 - c. Memeriksa kecocokan poros kemudi.
 - d. Memeriksa kondisi ban dan menyetel tekanan angin ban.
 - e. Memeriksa dan mengencangkan baut-baut pengikat (baut rangka, baut pengikat mesin, tuas starter, tuas transmisi, dsb).

2.2.3. Bengkel Motor

Bengkel adalah tempat di mana seseorang teknisi melakukan pekerjaannya melayani jasa perbaikan dan perawatan kendaraan. Bengkel umum kendaraan bermotor adalah bengkel umum yang berfungsi untuk membetulkan, memperbaiki, dan merawat kendaraan bermotor agar tetap memenuhi persyaratan teknis dan layak jalan . Hal ini guna memenuhi tuntutan PP No. 44 Tahun 1993 tentang Kendaraan dan pengemudi pasal 126, 127, 128, dan 129, dinyatakan bahwa setiap kendaraan bermotor harus memenuhi persyaratan teknis dan kelayakan kendaraan bermotor. Bengkel merupakan suatu usaha jenis wirausaha kecil dan menengah yang bergerak dalam bidang jasa pelayanan perbaikan, baik itu sepeda motor atau mobil. Usaha bengkel sepeda motor adalah usaha yang melakukan perbaikan sepeda motor agar dapat kembali berjalan dengan baik sesuai dengan keinginan pemilik atau bentuk asli dari sepeda motor tersebut [8].

2.2.4. Aplikasi Berbasis Web

Aplikasi berbasis *Web* merupakan suatu perangkat lunak komputer yang dikodekan dalam bahasa pemrograman yang didukung oleh perangkat lunak seperti *HTML*, *JavaScript*, *CSS*, *Ruby*, *Python*, *PHP*, *Java* dan bahasa pemrograman lainnya. Aplikasi web adalah sebuah program yang disimpan di server dan dikirim melalui internet dan diakses melalui antarmuka browser. Dapat disimpulkan aplikasi web merupakan aplikasi yang diakses menggunakan teknologi browser dengan melalui jaringan internet atau intranet [9].

Berikut beberapa komponen dalam aplikasi berbasis web antara lain :

1. *PHP*

PHP adalah singkatan dari *PHP Hypertext Preprocessor*, yang merupakan sebuah bahasa scripting yang terpasang pada *HTML*. Sebagian besar sintaks mirip dengan bahasa C, Java dan Perl, ditambah beberapa fungsi *PHP* yang spesifik.

2. *JavaScript*

JavaScript adalah bahasa yang digunakan untuk membuat program yang digunakan agar dokumen *HTML* yang ditampilkan dalam browser menjadi lebih interaktif, tidak sekedar indah saja. *JavaScript* memberikan beberapa fungsionalitas ke dalam halaman *web*.

3. *HTML dan CSS*

HTML adalah singkatan dari *HyperText Markup Language*. *HTML*

merupakan *file* teks yang ditulis menggunakan aturan-aturan kode tertentu untuk kemudian disajikan ke *user* melalui suatu aplikasi *web browser*. *CSS* adalah sebuah pemrograman atau boleh dibilang *script* yang mengendalikan beberapa komponen (*tag html*) dalam sebuah *website* sehingga tampilan akan menjadi lebih terstruktur, rapi, dan seragam.

4. *MySQL*

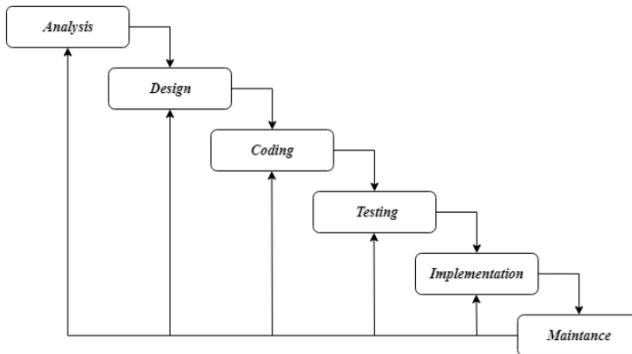
MySQL adalah suatu sistem manajemen basis data relasional (*RDBMS - Relational Database Management Sistem*) yang merupakan salah satu jenis *database server* terkenal dan banyak digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengelolaan datanya.

2.2.5. **Rekayasa Perangkat Lunak**

Perangkat lunak (*Software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model sistem, dan cara penggunaannya. Rekayasa perangkat lunak (RPL) sebagai penerapan suatu pendekatan yang sistematis, disiplin dan terkuantifikasi atas pengembangan, penggunaan dan pemeliharaan perangkat lunak. RPL sendiri adalah suatu disiplin ilmu yang membahas semua aspek produksi perangkat lunak, mulai dari tahap awal yaitu analisa kebutuhan pengguna, menentukan spesifikasi dari kebutuhan pengguna, desain, pengkodean, pengujian sampai pemeliharaan sistem setelah digunakan. RPL merupakan pembangunan dengan menggunakan prinsip atau konsep rekayasa dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomi yang dipercaya dan bekerja secara efisien menggunakan mesin [10].

RPL lebih fokus pada praktik pengembangan perangkat lunak dan mengirimkan perangkat lunak yang bermanfaat kepada pelanggan (*customer*). Adapun ilmu komputer lebih fokus pada teori dan konsep dasar perangkat komputer. Rekayasa perangkat lunak lebih fokus pada bagaimana membuat perangkat lunak yang memenuhi kriteria berikut [10]:

1. Dapat terus dipelihara setelah perangkat lunak selesai dibuat seiring berkembangnya teknologi dan lingkungan (*maintainability*).
 - a. Dapat diandalkan dengan proses bisnis yang dijalankan perubahan yang terjadi (*dependability robust*)
 - b. Kemampuan untuk dipakai sesuai dengan kebutuhan (*usability*) Dari kriteria di atas maka perangkat lunak yang baik adalah perangkat lunak yang dapat memenuhi kebutuhan pelanggan (*customer*) atau *user* (pemakai perangkat lunak) atau berorientasi pada pelanggan atau pemakai perangkat lunak, bukan berorientasi pada pembuat atau pengembang perangkat lunak.
 - c. Efisien dari segi sumber daya dan penggunaan.
Menurut Pressman (2012) Metode *Waterfall* adalah model klasik yang bersifat sistematis, berurutan dalam membangun software. Model air terjun (*Waterfall Model*) memiliki pendekatan klasik dalam pengembangan perangkat lunak yang menggambarkan metode pengembangan linier dan berurutan. Ini terdiri dari lima hingga tujuh fase, setiap fase didefinisikan oleh tugas dan tujuan yang berbeda, di mana keseluruhan fase menggambarkan siklus hidup perangkat lunak hingga pengirimannya. Setelah fase selesai, langkah pengembangan selanjutnya mengikuti dan hasil dari fase sebelumnya mengalir ke fase berikutnya [11].



Gambar 2.2 Metode *Waterfall*

1. Analisis (*analysis*)
 Analisis merupakan tahap awal dalam pembangunan sistem. Tahapan ini lebih difokuskan pada analisis kebutuhan. Pada tahap ini pengembangan sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut.
2. Desain (*design*)
 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 pengkodean. 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. Pengkodean (*coding*)
 Pengkodean merupakan pengimplementasian hasil desain ke dalam kode atau bahasa yang dimengerti oleh mesin komputer dengan menggunakan bahasa pemrograman tertentu.
4. Pengujian (*testing*)
 Pengujian merupakan proses yang berfokus pada bagian internal perangkat lunak secara logis untuk memastikan bahwa setiap pernyataan telah diuji, dan pada bagian eksternal fungsi, dimana diberlakukan pengujian untuk menemukan kesalahan (*error*) dan memastikan bahwa masukan (*input*) yang ditentukan akan

- memberikan hasil yang diharapkan.
5. Implementasi (*implementation*)
Implementasi merupakan penerapan sistem informasi *servis* motor pada bengkel motor apakah sudah sesuai dengan kebutuhan bengkel.
 6. Pemeliharaan (*maintance*)
Tahap pemeliharaan merupakan tahap akhir dari pengembangan sistem. Pemeliharaan juga dilakukan dengan tujuan melakukan penyesuaian dan perbaikan pada sistem tersebut.

Keunggulan Model pendekatan pengembangan software metode *waterfall* adalah pencerminan kepraktisan rekayasa , yang bisa membuat kualitas software tetap terjaga. Jenis model yang bersifat lengkap sehingga proses pemeliharannya lebih mudah. Karena struktur logis dari model, kesalahan konseptual seringkali dapat dihindari. Model ini mengarah pada dokumentasi teknis yang luas, yang merupakan kelegaan bagi programmer dan pengembang baru dan juga berguna dalam tahap pengujian. Kemajuan proyek dapat dipantau menggunakan tonggak sejarah. Total biaya dapat diperkirakan dengan akurasi relatif jika tidak ada konflik [11].

Kelemahan model *waterfall* ini adalah lambatnya proses pengembangan perangkat lunak. Dikarenakan proses yang satu tidakbisa diloncat-loncat maka dari itu model ini sangat memakan waktu dalam mengembangkannya. Kelemahan yang lain kinerja tidak optimal dan efisien. Konflik, bug, dan kesalahan program terkadang menyebabkan kenaikan biaya dan waktu yang cukup lama. Hal yang sama berlaku jika klien tidak puas. Spesifikasi yang awalnya dibuat seringkali sulit untuk dipahami oleh klien karena lebih abstrak dari pada apa yang seharusnya dilakukan oleh perangkat lunak [11].

2.2.6. Flowchart

Flowchart merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Biasanya mempengaruhi penyelesaian masalah yang khususnya perlu dipelajari dan dievaluasi lebih lanjut [12]. Berikut adalah simbol-simbol yang terdapat pada *flowchart* seperti pada tabel dibawah ini :

1. *Flow Direction Symbols* (Simbol Penghubung/alur)

Simbol yang digunakan untuk menghubungkan antara symbol yang satu dengan yang lainnya. Simbol ini juga disebut connecting line, simbol tersebut adalah :

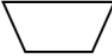
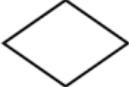
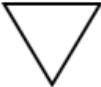
Tabel 2.1 Simbol Penghubung *Flowchart*

No	NAMA	SIMBOL	KETERANGAN
1.	<i>Arus / Flow</i>		Untuk menyatakan arus suatu proses
2.	<i>Communication link</i>		Untuk menyatakan bahwa adanya transisi suatu data atau informasi dari suatu lokasi ke lokasi lainnya
3.	<i>Connector</i>		Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman / lembaran sama
4.	<i>Offline Connector</i>		Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman atau lembaran yang berbeda

2. *Processing Symbols* (Simbol Proses)

Simbol yang menunjukkan jenis operasi pengolahan dalam suatu proses / prosedur. Symbol – symbol tersebut adalah :

Tabel 2.2 Simbol *Flowchart*

No	NAMA	SIMBOL	KETERANGAN
1.	Proses		Sebuah fungsi pemrosesan yang dilaksanakan oleh komputer biasanya menghasilkan perubahan terhadap data atau informasi
2.	Simbol manual		Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer (manual)
3.	<i>Decision / Logika</i>		Untuk menunjukkan suatu kondisi tertentu, dgn dua kemungkinan, YA / TIDAK
4.	<i>Predefined Process</i>		Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal
5.	Terminal		Untuk menyatakan permulaan atau akhir suatu program
6.	<i>Offline Storage</i>		Untuk menunjukkan bahwa data dalam symbol ini akan disimpan ke suatu media tertentu
7.	<i>Manual Input</i>		Untuk memasukkan data secara manual dengan menggunakan online keyword

3. *Input / Output Symbols* (Simbol Input – output)
 Simbol yang menunjukkan jenis peralatan yang digunakan sebagai media *input* atau *output*. Simbol – simbol tersebut adalah :

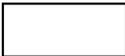
Tabel 2.3 *Input Output* simbol

No	NAMA	SIMBOL	KETERANGAN
1.	<i>Input / Output</i>		Untuk menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya
2.	<i>Disk Storage</i>		Untuk menyatakan input berasal dari disk atau output disimpan ke disk
3.	<i>Document</i>		Untuk menyetak dokumen

2.2.7. *Entity Relationship Diagrams (ERD)*

ERD adalah representasi grafik dari data untuk organisasi atau untuk area bisnis, menggunakan entitas sebagai kategori data dan relationship untuk asosiasi antar entitas. Dari pengertian di atas dapat diambil kesimpulan bahwa ERD adalah model yang merepresentasikan data dalam entitas dan hubungan antar entitas secara jelas yang akan digunakan untuk membangun basis data [13].

Tabel 2.4 Simbol *Entity Relationship Diagram*

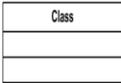
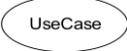
No	NAMA	SIMBOL	KETERANGAN
1.	<i>Entity / Entitas</i>		Suatu kumpulan objek atau sesuatu yang dapat dibedakan atau dapat didefinisikan

2.	<i>Relationship</i>		Hubungan yang dapat terjadi antara satu entitas atau lebih
3.	<i>Atribut</i>		Karakteristik dari entitas atau Relationship yang menyediakan penjelasan detail entitas atau relation
4.	<i>Link</i>		Baris sebagai penghubung antara himpunan, relasi dan himpunan entitas dan atributnya

2.2.8. *Unified Modelling Language (UML)*

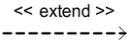
Unified Modelling Language (UML) adalah alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual . Selain itu merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem software yang terkait dengan objek. Penggunaan UML bertujuan untuk mengidentifikasi bagian-bagian yang termasuk dalam lingkup sistem didalam aplikasi, mendokumentasikan hasil analisa dan desain serta untuk menggambarkan sebuah sistem *software*. Model UML yang dipakai dalam pengembangan aplikasi ini antara lain *Use Case Diagram* dan *Activity Diagram*. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek [14]. Notasi pada UML dapat dilihat pada Tabel 2.5.

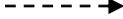
Tabel 2.5 Notasi pada UML

No	Notasi	Penjelasan
1.		<p>Actor adalah segala sesuatu yang berinteraksi dengan sistem aplikasi komputer. Biasanya yang dilakukan actor adalah memberikan informasi pada sistem dan\atau memerintahkan sistem untuk melakukan sesuatu.</p>
2.		<p>Class merupakan pembentuk utama dari sistem berorientasi obyek karena class menunjukkan kumpulan obyek yang memiliki atribut dan operasi yang sama. Class memiliki 3 bagian, Nama class, Atribut, dan Operasi.</p>
3.		<p>Interface merupakan kumpulan operasi tanpa implementasi dari suatu class. Implementasi operasi interface dijabarkan oleh operasi dalam class.</p>
4.		<p>Use case menjelaskan urutan kegiatan yang dilakukan actor dan sistem untuk mencapai suatu tujuan tertentu. Use case hanya menjelaskan apa yang akan dilakukan oleh actor dan sistem, bukan bagaimana actor dan sistem melakukan kegiatan tersebut.</p>
5.		<p>Interaction digunakan untuk menunjukkan baik aliran pesan atau informasi antar obyek maupun hubungan antar obyek. Biasanya interaction dilengkapi dengan teks bernama operation signature yang tersusun dari nama operasi, parameter yang dikirim dan tipe parameter yang dikembalikan.</p>
6.		<p>Package adalah kontainer atau wadah konseptual yang digunakan untuk mengelompokkan elemen-elemen dari sistem yang sedang dibangun, sehingga bisa dibuat model yang lebih sederhana. Tujuannya adalah untuk mempermudah</p>

		penglihatan dari model yang sedang dibangun.
7.		<i>Note</i> digunakan untuk memberikan keterangan dan komentar tambahan dari suatu elemen sehingga bisa langsung terlampir dalam model. Note ini bisa ditempelkan ke semua elemen notasi yang lain.
8.		<i>Dependency</i> merupakan relasi yang menunjukkan bahwa perubahan pada salah satu elemen memberi pengaruh pada elemen lain. Elemen yang ada di bagian tanda panah adalah elemen yang tergantung pada elemen yang ada di bagian tanpa tanda panah.
9.		<i>Include</i> menunjukkan bahwa suatu bagian dari elemen memicu eksekusi bagian dari elemen lain (yang ada pada tanda panah).

Tabel 2.6 Notasi Penghubung pada UML

No	Notasi	Penjelasan
1.		<i>Extend</i> menunjukkan bahwa suatu bagian dari elemen bisa disisipkan ke dalam elemen yang lain.
2.		<i>Association</i> menggambarkan navigasi antar class, berapa banyak obyek lain yang bisa berhubungan dengan satu obyek, dan apakah suatu class menjadi bagian dari class lainnya.

3.		<p>Generalization menunjukkan hubungan antara elemen yang lebih umum ke elemen yang lebih spesifik. Dengan generalization, class yang lebih spesifik (subclass) akan menurunkan atribut dan operasi dari class yang lebih umum (superclass). Notasi ini digunakan pada konsep inheritance.</p>
4.		<p>Realization menunjukkan hubungan bahwa elemen yang ada di bagian tanpa tanda panah akan merealisasikan apa yang dinyatakan oleh elemen yang ada di bagian dengan panah.</p>

Berikut dibawah ini adalah Model *UML* yang dipakai dalam pengembangan aplikasi ini antara lain *Use Case Diagram*, activity diagram, dan flowchart :

1. *Use Case Diagram*

Use Case Diagram mendeskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem yang akan dibuat untuk mengetahui fungsi apa saja yang ada di dalam sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

2. *Activity Diagram*

Activity diagram merupakan diagram yang digunakan untuk menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis.

3. *Flowchart*

Flowchart adalah suatu bagan yang mempunyai arus menggambarkan langkah-langkah penyelesaian suatu masalah. Flowchart merupakan cara penyajian dari suatu algoritma.

2.2.9. **Rekayasa Web**

Rekayasa Web adalah sebuah aplikasi yang menggunakan pendekatan sistematis, disiplin, dan terukur untuk pengembangan, operasi dan pemeliharaan aplikasi berbasis *Web (Web-based applications)*. Sebagai catatan, kebanyakan literatur tentang rekayasa *web* mengacu

kepada situs *web*, aplikasi berbasis *web*, sistem berbasis *web*, dan aplikasi *web*. Sistem berbasis *web* melibatkan suatu campuran antara (*print publishing*) penerbitan cetakan dan pengembangan perangkat lunak, antara pemasaran dan komputasi, anatara komunikasi internal dan relasi *eksternal*, dan antara seni dan teknologi [15].

2.2.10. Basis Data

Basis data terdiri dari dua kata yaitu basis dan data, basis dapat dikatakan gudang, markas, atau tempat berkumpul. Sedangkan sata dapat diartikan representasi dari fakta dunia yang mewakili sebuah obyek (manusia, peristiwa, barang, keadaan dsb) yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi atau kombinasinya. Elmasri menyampaikan bahwa istilah basis data lebih dibatasi pada arti implisit yang khusus mempunyai beberapa pengertian, yaitu [16] :

1. Basis data merupakan kumpulan data dari berbagai sumber yang secara logika mempunyai arti implicit. Sehingga apabila data terkumpul secara acak dan tanpa mempunyai arti, tidak dapat disebut basis data.
2. Basis data dapat digunakan oleh beberapa pemakai dan beberapa aplikasi yang sesuai dengan kepentingan pemakai.
3. Basis data merupakan penyajian suatu aspek dari dunia nyata (*real word* atau *miniworld*). Misalnya basis data perbankan, perpustakaan, pertanahan, perpajakan, dan lain-lain.
4. Basis data perlu dirancang, dibangun dan data dikumpulkan untuk suatu tujuan tertentu.

SQL (*Structured Query Language*) adalah sebuah bahasa yang dipergunakan untuk mengakses data dalam basis data relasional dengan tujuan membuat *query* dan memanipulasi *database*, menulis dan menjalankan *query* di SQL adalah bagian dari pembahasan *database relasional*. SQL terdiri dari 2 bahasa *Data Definition Language* (DDL) dan *Data Manipulation Language* (DML). Implementasi DDL dan DML berbeda untuk tiap sistem manajemen basis data (SMBD), namun secara umum implementasi tiap bahasa ini memiliki bentuk standar yang ditetapkan.

1. *Data Definition Language* (DDL)
DDL adalah struktur basis data yang menggambarkan skema

basis data secara keseluruhan dan didesain dengan bahasa khusus. Digunakan untuk kepentingan penciptaan database, tabel, hingga penghapusan database atau tabel, contohnya : *Create Database, Create Table, Drop Table, Alter Table*.

2. Data Manipulation Language (DML)

DML adalah bentuk bahasa basis data yang digunakan untuk melakukan manipulasi dan pengambilan data pada suatu baris data. Contohnya : *Select, Delete, Insert, Update*.

Dari beberapa pengertian tersebut, dapat diambil kesimpulan bahwa sistem basis data merupakan sistem terkomputerisasi yang berujuan untuk memelihara data yang telah diolah dan mempercepat proses saat dibutuhkan. Dengan kata lain basis data merupakan media untuk menyimpan data agar dapat diakses dengan mudah dan cepat. Kebutuhan basis data dalam sistem informasi meliputi memasukkan, menyimpan, mengambil kembali data dan membuat laporan berdasarkan data yang telah disimpan.