

BAB II

LANDASAN TEORI

2.1 Sistem Informasi

Sistem adalah suatu kesatuan dari berbagai elemen berupa fisik maupun bukan fisik yang tertuju pada suatu kumpulan dengan saling berhubungan dan terdapat interaksi pada satu tujuan atau sasaran akhir dari sistem. Pengertian informasi adalah hasil dari pemrosesan data yang relevan dan terdapat manfaat bagi penggunaannya. Dari teori tersebut dapat disimpulkan bahwa sistem merupakan kumpulan dari *element* atau komponen yang terlihat secara fisik maupun *non* fisik dengan tujuan untuk membentuk suatu kumpulan yang berhubungan dan berinteraksi pada satu tujuan. Dan informasi merupakan suatu proses data secara relevan dan memiliki manfaat bagi *user*nya [9].

Sistem informasi merupakan sebuah alat dalam menyajikan informasi dengan sedemikian rupa agar dapat memiliki manfaat bagi penerima. Tujuannya untuk memberikan serangkaian informasi dalam memulai, perencanaan, organisasi, dan operasional sebuah perusahaan untuk proses pengambilan keputusan. Dari teori tersebut dapat disimpulkan bahwa sistem informasi adalah sebuah alat yang memberikan informasi dalam sebuah perencanaan, organisasi dan operasional perusahaan untuk pengambilan suatu keputusan [10]. Terdapat beberapa karakteristik informasi yang berkualitas menurut *Kroenke & Boyle* yaitu :

1. Akurat

Informasi yang memiliki kualitas berasal dari data yang benar, lengkap dan akurat berdasarkan hasil dari pengolahan data yang sesuai dengan harapan. Data akurat sangat berguna dalam bisnis dikarenakan data berkaitan dengan pengambilan suatu keputusan untuk strategi bisnis perusahaan. Pengambilan

keputusan dari sebuah informasi yang kurang akurat akan mendapatkan hasil yang tidak sesuai dengan harapan.

2. Tepat Waktu

Informasi yang memiliki kualitas membutuhkan sebuah data yang tepat waktu, dimana merujuk pada ketersediaan data ketika diperlukan untuk dapat digunakan pada kebutuhan tertentu. Informasi yang berkualitas berasal dari data yang diolah dengan cepat dan tepat agar memiliki manfaat. Dengan contoh ketika perusahaan memerlukan sebuah laporan bulanan, maka data akan diproses sesuai dengan data yang dihasilkan dari perusahaan dengan jangka waktu sebulan dan dapat diselesaikan dalam waktu yang singkat, dikarenakan laporan tersebut sangat menjadi pertimbangan *management* dalam melakukan pengambilan keputusan.

3. Relevan

Data dapat relevan baik dalam konteks maupun subjek. Relevansi data berdasarkan konteks yang merujuk pada data yang tersusun sesuai dengan penggunaan dalam bidang tertentu. Dengan contoh ketika karyawan pada bagian *payroll*, maka daftar data jam kerja pada setiap karyawan merupakan data yang relevan terkait dengan pekerjaan karyawan yang berada di bagian *payroll* tersebut. Akan tetapi ketika daftar data jam kerja diberikan kepada seluruh karyawan maka menjadi tidak relevan berdasarkan konteks. Relevan berdasarkan subjek merujuk pada data yang akan disusun sesuai dengan subjek yang terkait.

4. Cukup

Informasi yang memiliki kualitas juga bergantung pada data yang cukup. Dengan data yang cukup sesuai dengan keperluan dan tidak melebihi apa yang diperlukan ketika memproses data untuk menjadi informasi maka data itu berkualitas.

5. Sebanding dengan biaya

Data tidak gratis, oleh karena itu terdapat biaya dalam memproses data yaitu dalam pemeliharaan sistem untuk memproses data, membayar gaji karyawan yang mengelola data dan sebagainya. Dengan hal tersebut maka penggunaan data harus dengan bijak agar data yang dihasilkan dapat mengimbangi biaya yang dibutuhkan ketika memproses data tersebut [11].

2.2 Sistem Pencatatan

Sistem pencatatan adalah sebuah sistem yang berguna untuk membantu perusahaan dalam mengelola akuntansi dari perhitungan, pencatatan dan pelaporan keuangan yang lebih efektif dan efisien. Dengan adanya sistem ini, akuntansi yang rumit dan kompleks dapat menjadi lebih *simple* dengan adanya sistem otomatis dan akurat dalam mengelola data. Terdapat manfaat pada sistem pencatatan akuntansi yaitu untuk membuat *management* akuntansi perusahaan bekerja lebih efektif karena terintegrasi pada satu sistem yang dapat mempercepat proses akuntansi [12].

Terdapat 2 metode dalam sistem pencatatan yaitu metode perpetual dan metode periodik. Metode perpetual merupakan metode yang mencatat semua transaksi penjualan dan pembelian barang dagangan secara langsung pada akun persediaan, sedangkan metode periodik pencatatan transaksi pembelian dan penjualan barang dilakukan dengan terpisah. Pada metode perpetual suatu perusahaan tidak memerlukan perhitungan secara fisik ataupun *stock opname* untuk stok yang ada. Pada metode periodik suatu perusahaan dapat mengetahui besar persediaan yang ada di *inventory* sehingga daftar stok menjadi akurat [26].

2.3 Pembelian

Pembelian merupakan sebuah transaksi dari suatu usaha yang dilakukan untuk memenuhi kebutuhan barang atau jasa yang dibutuhkan berdasarkan kualitas dan kuantitas yang tersedia selama periode tertentu. Biasanya adanya pembelian dikarenakan untuk kebutuhan pribadi, kepentingan suatu produksi ataupun dijual kembali [13]. Dapat disimpulkan bahwa pembelian akan terjadi ketika terdapat transaksi oleh suatu usaha dalam menambah kebutuhan barang atau jasa sesuai dengan kuantitas dan kualitas yang ada.

2.4 Inventory

Persediaan sendiri yaitu kumpulan barang yang disimpan dan akan dijual oleh penjual pada masa tertentu, tergantung pada permintaan dari pelanggan di periode yang akan datang. Terdapat kategori persediaan terdiri dari barang baru, setengah proses produksi dan barang jadi untuk dijual atau dipasarkan [13]. Sistem *inventory* merupakan kumpulan dari aspek kebijakan dan pengendalian dalam memantau tingkat persediaan dan menentukan tingkat mana yang harus dijaga *available stock* harus diisi kembali dan beberapa banyak yang harus dipesan [14]. Persediaan sendiri merupakan aset yang dibeli dan diproses agar dapat menghasilkan suatu barang atau jasa. *Inventory* merupakan stok, *trading stock* ataupun *inventory stock*, akan tetapi *inventory* sendiri tidak termasuk dalam *capital assets*, *human resource*, dan uang [27].

2.5 Usaha Mikro, Kecil dan Menengah

Pengertian UMKM menurut UU No.20 Tahun 2008 yaitu sebuah usaha yang dikelola dan dimiliki oleh perorangan atau sekelompok kecil orang dan jumlah aset, hingga kekayaan maupun pendapatannya akan dibagi sesuai dengan bagiannya masing – masing [15].

- a. Usaha Mikro adalah usaha produktif milik orang perorangan dan/atau badan usaha perorangan yang memenuhi kriteria Usaha Mikro sebagaimana diatur dalam Undang - Undang ini.
- b. Usaha Kecil yaitu usaha ekonomi produktif yang berdiri sendiri, yang dilakukan oleh orang perorangan atau badan usaha yang bukan merupakan anak perusahaan atau bukan cabang perusahaan yang dimiliki, dikuasai, atau menjadi bagian baik langsung maupun tidak langsung dari Usaha Menengah atau Usaha Besar yang memenuhi kriteria Usaha Kecil sebagaimana dimaksud dalam Undang - Undang ini.
- c. Usaha Menengah merupakan usaha ekonomi produktif yang berdiri sendiri, yang dilakukan oleh orang perorangan atau badan usaha

yang bukan merupakan anak perusahaan atau cabang perusahaan yang dimiliki, dikuasai, atau menjadi bagian baik langsung maupun tidak langsung dengan Usaha Kecil atau Usaha Besar dengan jumlah kekayaan bersih atau hasil penjualan tahunan sebagaimana diatur dalam Undang - Undang ini [15].

2.6 Metode *Prototyping*

Metode *prototyping* yaitu sebuah metode pengembangan terhadap perangkat lunak, yang memiliki model fisik secara kerja sistem dan berfungsi untuk sebagai versi awalan dari sistem. Metode ini menghasilkan *prototype* sebuah sistem sebagai perantara pengembang dan *user* agar dapat berkomunikasi dalam kegiatan pengembangan sistem informasi [16].

Prototyping memberikan versi awal yang dibangun sesuai dengan yang dibutuhkan oleh sistem informasi. *Feedback* dari *user* sangatlah berguna dalam metode *prototyping* dikarenakan metode ini akan melibatkan *user* dalam menggunakan *input*, *output*, model dan *user interface*. Metode ini digunakan untuk memvalidasi kebutuhan *user* dengan sistem yang dihasilkan [17].



Gambar 2.1 Metode *Prototyping*

Sumber : [17]

1. *Planning*

Tahapan awal yang dilakukan adalah membuat *planning* sistem yang akan dibuat dengan mengumpulkan data dengan melakukan observasi dan wawancara pada *owner* Aneka Motor untuk mengetahui *user requirement* apa saja yang diperlukan didalam sistem.

2. *Analysis*

Setelah mengumpulkan *user requirement* dari owner Aneka Motor, tahapan selanjutnya yaitu melakukan analisis pada masalah berdasarkan apa saja yang dibutuhkan. Pada tahap ini akan dilakukan penentuan fitur – fitur yang akan di masukan kedalam sistem sesuai dengan kebutuhan yang diperlukan.

3. *Design*

Tahapan selanjutnya yaitu membuat *design* ataupun gambaran dari sistem yang akan dibuat. Pada *design* yang dibuat sesuai dengan hasil analisis pada tahapan sebelumnya. Ketika telah melakukan *design* maka akan ditunjukkan kepada *owner* Aneka Motor dan meminta saran terkait gambaran sistem yang akan dibuat. Ketika terdapat masukan – masukan yang diberikan, maka akan dilakukan evaluasi kembali hingga hasil *design* sesuai dengan kebutuhan yang telah disetujui.

4. *System Prototype*

Ketika *design* telah dibuat, maka selanjutnya akan membuat sistem *prototype* dengan berdasarkan hasil *design* yang telah ditentukan. *Prototype* yang dibuat menggunakan *HTML* dan *MySQL* sebagai *databasenya*. Penulis menggunakan *html* dikarenakan dapat mudah untuk dipahami, bersifat *open source* secara gratis, dan tidak memerlukan *compiler* dalam menjalankan perintah.

5. *Implementation*

Setelah *prototype* telah berhasil dibuat maka tahapan selanjutnya yaitu melakukan implementasi sistem kepada *user* yaitu *owner* UMKM Aneka Motor. Pada tahapan penerapan sistem, *user* akan selalu di dampingi sampai dapat menguasai fitur – fitur dari keseluruhan sistem. Pada proses *prototype* ini hanya sampai dilakukannya *testing* untuk sistem yang dibuat.

2.7 Unified Modelling Language (UML)




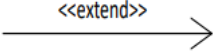
UML adalah suatu *tool*/model yang memiliki tujuan untuk membantu dalam perancangan pengembangan suatu *software* secara *object oriented*. UML sendiri memiliki standar penulisan yaitu sistem *blueprint*, yang meliputi proses bisnis, skema *database*, komponen sistem *software* [18].

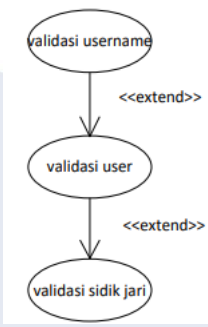

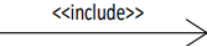
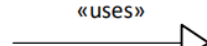
2.7.1 Use Case Diagram

Use Case Diagram merupakan diagram yang menggambarkan suatu interaksi antara *user* dan sistem. *Use case diagram* terdapat sebuah aktor dan interaksi yang dilakukan, aktornya dapat berupa manusia, sistem ataupun perangkat keras. *Use case* sangat membantu dalam penyusunan sebuah sistem yang akan dipakai dan merancang *test* untuk *design* sistem yang ada [19].

Table 2.1 Simbol *Use Case Diagram*

Sumber : [20]

Simbol	Keterangan
<p>Use Case</p> 	Merupakan sistem dari berbagai unit yang dapat memiliki interaksi antar unit / aktor dengan menggunakan awalan pada frase yaitu <i>use case</i> .
<p>Aktor / Actor</p> 	Orang, proses ataupun sistem secara terhubung untuk berinteraksi dengan sistem informasi. Simbol aktor yang merupakan gambar orang akan tetapi tidak dapat dipastikan selalu orang asli.
<p>Asosiasi / Association</p> 	Simbol komunikasi antar aktor dan <i>use case</i> yang saling memiliki interaksi.
<p>Ekstensi / Extend</p> 	Merupakan sebuah ekstensi dari sebuah <i>use case</i> dalam memasukan <i>optional behavior</i> . Contoh :

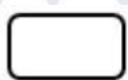
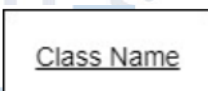
	
<p>Generalisasi / Generalized</p> 	<p>Hubungan pada generalisasi dan spesialisasi yaitu merupakan dua buah <i>use case</i> yang dimana fungsi dari satunya adalah fungsi yang umum.</p>
<p>Include / Uses</p>  	<p>Relasi <i>use case</i> yang ditambahkan ke sebuah <i>use case</i>, dimana <i>user</i> yang ditambahkan membutuhkan sebuah <i>use case</i> untuk menjalankan fungsinya.</p>






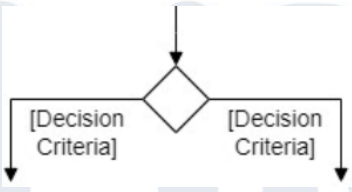
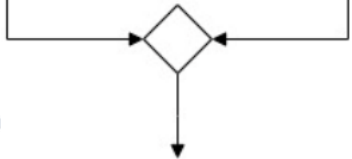
2.7.2 Activity Diagram

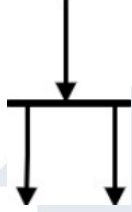
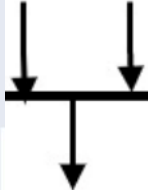
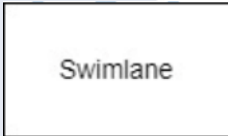
Activity Diagram yaitu sebuah gambaran dari aktivitas proses bisnis pada suatu objek. kebiasaan yang bisa dilihat dari diagram dengan menggunakan dengan struktur analisis. *Activity Diagram* dapat berfungsi sebagai penggambaran segala perilaku dari aliran kerja dan juga aktivitas yang terkait [21].

Tabel 2.2 Simbol *Activity Diagram*

Sumber : [21]

Simbol	Keterangan
	<p><i>Activity</i> :</p> <ul style="list-style-type: none"> - Berfungsi untuk perwakilan serangkaian tindakan. - Memberi label sesuai nama.
	<p><i>Object Node</i> :</p> <ul style="list-style-type: none"> - Berfungsi dalam menampilkan objek yang

	berhubungan dengan gabungan objek. - Memberi label sesuai nama.
	<i>Control Flow :</i> - Menunjukkan urutan dari eksekusi.
	<i>Object Flow :</i> - Menunjukkan aliran pada suatu objek ke aktivitas lainnya.
	<i>Initial Node :</i> - Menggambar awal pada kegiatan yang akan dilakukan
	<i>Final – Activity Node :</i> - Berfungsi dalam pemberhentian pada semua aliran <i>control</i> dan objek yang berhubungan pada suatu kegiatan / tindakan
	<i>Final – Flow Node :</i> - Berfungsi dalam menghentikan aliran <i>control</i> atau objek tertentu.
	<i>Decision Node :</i> - Berfungsi dalam kondisi dimana sebagai perwakilan untuk memastikan pada setiap aliran <i>control</i> atau objek memiliki turun satu jalur.
	<i>Merge Node :</i> - Berfungsi untuk menyatukan kembali keputusan yang berbeda yang dibuat menggunakan <i>decision node</i> .
	<i>Fork Node :</i> - Berfungsi sebagai pembagi antar perilaku menjadi sekumpulan aktivitas parallel ataupun bersamaan.

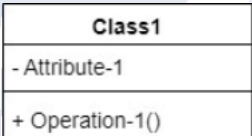
	
	<p><i>Join Node :</i></p> <ul style="list-style-type: none"> - Berfungsi dalam penyatuan dari serangkaian <i>activity</i> parallel ataupun bersamaan.
	<p><i>Swimlane :</i></p> <ul style="list-style-type: none"> - Berfungsi dalam memecahkan <i>activity diagram</i> menjadi baris dan kolom untuk menetapkan aktivitas individu.




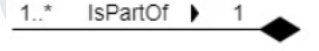
2.7.3 Class Diagram

Class Diagram merupakan kumpulan model yang menunjukkan hubungan antar *class* dengan sifat konstan dari waktu ke waktu dalam suatu sistem. *Class diagram* menggambarkan *class* yang terdapat status dan perilaku yang berhubungan antar *class* [21]. Simbol dalam *class diagram* memiliki arti masing – masing yaitu :

Tabel 2.3 Simbol *Class Diagram*

Sumber : [21]

Simbol	Keterangan
	<p><i>A Class :</i></p> <ul style="list-style-type: none"> - Terdapat orang, tempat , atau hal yang diperlukan suatu sistem dalam penyimpanan informasi - Memiliki nama yang tertulis menggunakan <i>bold</i> dan berada di kompartemen atas - Memiliki <i>attribute</i> yang berada di kompartemen tengah - Memiliki operasi yang berada di kompartemen bawah

<p>attribute name / derived attribute name</p>	<p><i>An Attribute :</i></p> <ul style="list-style-type: none"> - Menggambarkan keadaan dari suatu objek - Dapat menurunkan dari atribut lain
<p>operation name ()</p>	<p><i>An Operation :</i></p> <ul style="list-style-type: none"> - Menggambarkan suatu tindakan . fungsi yang bisa dilakukan oleh <i>class</i> - Dapat mengklasifikasi sebagai <i>constructor, update operation, query.</i>
	<p><i>An Association :</i></p> <ul style="list-style-type: none"> - Terdapat hubungan antar class dengan <i>class</i> itu sendiri. - Memberikan label dengan menggunakan kata kerja atau peran untuk mewakili hubungan. - Berisi beberapa simbol – simbol multiplisitas, yang dapat mewakili waktu minimum atau maksimum dari suatu <i>instance class.</i>
	<p><i>A Generalization :</i></p> <ul style="list-style-type: none"> - Menggambarkan hubungan dari beberapa <i>class.</i>
	<p><i>An Aggregation :</i></p> <ul style="list-style-type: none"> - Mewakili hubungan antara beberapa <i>class</i> dengan class itu sendiri - Merupakan bentuk khusus dari sebuah asosiasi
	<p><i>A Composition :</i></p> <ul style="list-style-type: none"> - Mewakili hubungan fisik dari beberapa <i>class</i> dan <i>class</i> itu sendiri - Merupakan bentuk khusus dari sebuah asosiasi

2.8 SAW

Metode SAW sering juga dikenal dengan nama penjumlahan terbobot. Konsep dasar pada metode *Simple Additive Weighting* disarankan untuk melakukan penyelesaian masalah penyeleksian dari sistem pengambilan keputusan secara

multi proses. Konsep dasar SAW yaitu mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut. Metode SAW juga membutuhkan proses normalisasi matriks keputusan (X) pada suatu skala yang dapat membandingkan dengan semua rating alternatif yang ada [22]. SAW merupakan istilah dari metode penjumlahan terbobot yang dimana mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut [22]. Terdapat rumus dalam melakukan normalisasi matrix keputusan :

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\text{Max}_i(x_{ij})} & \text{Jika } j \text{ merupakan atribut keuntungan (Benefit)} \\ \frac{\text{Min}_i(x_{ij})}{x_{ij}} & \text{Jika } j \text{ merupakan atribut biaya (Cost)} \end{cases}$$

Keterangan :

- r_{ij} = Nilai rating kinerja ternormalisasi
- x_{ij} = Nilai atribut yang dimiliki dari setiap kriteria
- Max_{ij} = Nilai Terbesar dari tiap kriteria
- Min_{ij} = Nilai Terkecil dari tiap kriteria
- Benefi* = Jika nilai tertinggi adalah terbaik
- Cost* = Jika nilai terendah adalah terbaik

Langkah – Langkah yang digunakan dalam menyelesaikan metode *simple additive weighting* yaitu :

- 1) Menentukan alternatif.
- 2) Menentukan kriteria yang akan dijadikan acuan dalam pengambilan keputusan .
- 3) Memberikan nilai rating kecocokan setiap alternatif pada setiap kriteria.
- 4) Menentukan bobot preferensi atau tingkat kepentingan (W) setiap kriteria.

$$W = [W_1 \ W_2 \ W_3 \ \dots \ W_j]$$

- 5) Membuat tabel rating kecocokan dari setiap alternatif pada setiap kriteria.
- 6) Membuat matrik keputusan yang dibentuk dari tabel rating kecocokan dari setiap alternatif pada setiap kriteria (Cj) yang sudah ditentukan, dimana $I = 1, 2 \dots m$ dan $j = 1, 2 \dots n$

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1j} \\ \vdots & & & \vdots \\ x_{i1} & x_{i2} & \dots & x_{ij} \end{bmatrix}$$

- 7) Melakukan normalisasi matrik keputusan dengan cara menghitung nilai rating kinerja ternormalisasi (r_{ij}) dari alternatif A_i pada kriteria C_j
- 8) Hasil dari nilai rating kinerja ternormalisasi (r_{ij}) membentuk matrik ternormalisasi (R)

$$R = \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1j} \\ \vdots & & & \vdots \\ r_{i1} & r_{i2} & \dots & r_{ij} \end{bmatrix}$$

- 9) Hasil akhir nilai preferensi (V_i) diperorel dari penjumlahan dari perkalian elemen baris matrik ternormalisasi (R) dengan bobot preferensi (W) yang bersesuaian elemen kolom matrik (W).

$$V_i = \sum_{j=1}^n w_j r_{ij}$$

Hasil perhitungan nilai V_i yang lebih besar mengindikasikan bahwa alternatif A_i merupakan alternatif terbaik [23].

Proses perhitungan SAW untuk melakukan analisa supplier. Langkah awal yaitu melakukan penentuan bobot pada tiap kriteria. Kriteria dalam SAW Aneka Motor yaitu kecepatan *transportasi*, tingkat diskon, pelayanan, garansi, keaslian barang, dan tempo pembayaran. Dimana kriteria ini sudah *fixed* dan tidak dapat diganti.

Tabel 3.2 Tabel Penentuan Bobot Setiap Kriteria

Kode Kriteria	Nama Kriteria	Attribut Kriteria	Bobot Kriteria
C1	Kecepatan Transportasi	Cost	0.2
C2	Tingkat Diskon	Benefit	0.2
C3	Pelayanan	Benefit	0.1
C4	Garansi	Benefit	0.1

C5	Keaslian Barang	Benefit	0.2
C6	Tempo Pembayaran	Benefit	0.2
Total			1

2.9 HTML

HTML (Hypertext Markup Language) merupakan sebuah bahasa yang berguna untuk memberikan deskripsi struktur dari sebuah halaman web. *HTML* berfungsi dalam mempublikasikan dokumen *online*. Statement dasar dari sebuah *HTML* yaitu *tags*. Sebuah tag digambarkan sebuah kurung siku (<>). *Tags* akan diberikan sebuah dokumen atau bagian dari dokumen akan dibuat pasangan. Terdiri dari *tag* pembuka dan penutup [24]. Pada saat ini bahasa pemrograman *HTML* masih terus berkembang dikarenakan pengguna internet setiap harinya selalu berkembang. Seiring perkembangan *HTML* pun berkembang dari masa ke masa dan untuk pertama kalinya *HTML* diciptakan oleh IBM pada tahun 1980. Untuk versi dari perkembangan *HTML* yaitu :

1. *HTML* Versi 1.0, kemampuan yang dimiliki hanya terdapat *heading*, *paragraph*, *hypertext*, *list*, cetak tebal dan cetak miring pada *text*.
2. *HTML* Versi 2.0, terdapat penambahan kualitas yang terletak untuk menampilkan suatu *form* pada dokumen.
3. *HTML* Versi 3.0, terdapat penambahan fasilitas baru seperti *figure* yang dapat meletakkan gambar dan table serta terdapat rumus – rumus matematika.
4. *HTML* Versi 4.0, terdapat banyak perubahan seperti perintah pada tabel, *image*, *link*, *text*, *meta*, *form* dan lainnya.
5. *HTML* Versi 5.0, terdapat tampilan baru dari penggabungan antara *CSS*, *HTML* dan *JavaScript* [28].

2.10 PHP

PHP merupakan bahasa pemrograman yang dijalankan melalui halaman web, yang digunakan untuk mengolah informasi di internet. Sedangkan pengertian lain dari *PHP* yaitu bahasa pemrograman *web server-side* yang bersifat *open source*. *PHP* sendiri merupakan *script* yang bisa menyatu dengan *HTML* [24]. Pada umumnya, *HTML* digunakan untuk membangun sebuah *website* yang membuat *PHP* bukanlah bahasa pemrograman yang diutamakan. Dengan adanya *HTML* dan *CSS* membuat *website* lebih statis dan *php* menjadi bahasa pelengkap yang digunakan untuk menghasilkan *website* yang lebih interaktif. Dalam *PHP*, *Cookie* dan *Session* dapat digunakan dalam menyimpan informasi pengguna seperti fungsi *session_start()* digunakan untuk memulai sebuah *session* dan fungsi *setcookie()* untuk memulai *cookie* yang ada. Pada saat *cookie* dimulai maka akan menyimpan *username* dan *password user* ketika menggunakan *browser* sehingga tidak harus mengisi ulang ketika melakukan *login* halaman *website* [28].

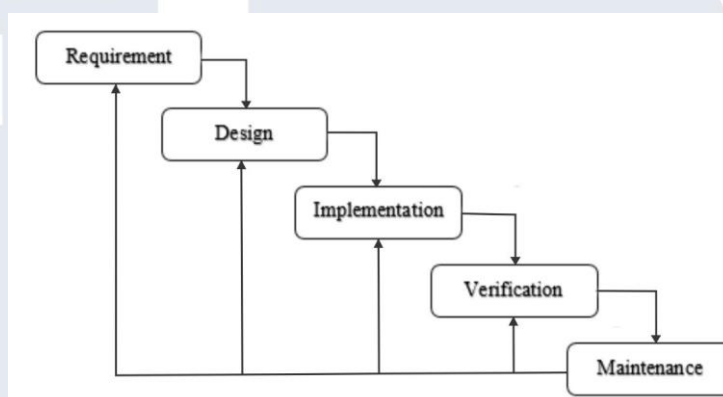
2.11 XAMPP

XAMPP sebuah *software* yang mendukung banyak operating system. Berfungsi sebagai *server* yang berdiri sendiri / *localhost*, yang terdapat *Apache HTTP Server*, *MySQL database*, dan penerjemah bahasa dengan bahasa pemrograman *PHP* dan *Perl*. Program ini terdapat dalam *GNU general public license* dan merupakan suatu *web server* yang mudah digunakan untuk memberikan tampilan halaman web secara dinamis [24]. *XAMPP* sendiri bersifat *open source* dan dapat digunakan sebagai *tool* pembantu dalam mengembangkan aplikasi yang berbasis *PHP*. Dimana *XAMPP v3.2.2* digunakan pada *windows* memiliki perangkat lunak seperti *MySQL*, *FileZilla*, *Mercury* dan *Tomcat*. Dalam pembuatan *website* menggunakan *XAMPP* dan *MySQL* sangat diperlukan sebuah penyimpanan basis data [28].

2.12 Waterfall

Metode waterfall merupakan metode yang biasanya dinamakan dengan *Classic Life Cycle* dikarenakan menggambar pendekatan secara sistematis dan juga berurutan sesuai dengan pengembangan perangkat lunak. Dimulai dari adanya spesifikasi

kebutuhan user dan selanjutnya melalui tahap – tahap *planning, modelling, contruction, deployment* yang diakhir dengan support pada software yang dihasilkan. Model pada pengembangan ini memiliki sifat linear dimulai dari tahap pemeliharaan dan tahapan berikutnya tidak akan dilaksanakan sebelum tahapan sebelumnya diselesaikan [29]. Terdapat tahapan metode waterfall seperti berikut :



Gambar 2.2 Metode *Waterfall*

Sumber : [29]

1. *Requirement* , merupakan tahapan yang diperlukan dalam pengembang sistem yang bertujuan dalam memahami *software* yang diharapkan oleh *user* dan batasan *software* tersebut. Informasi dapat dilakukan wawancara, diskusi dan survei secara langsung dan akan dianalisa untuk mendapatkan *user requirement*.

2. *Design* , merupakan tahapan membuat design sistem yang membantu *hardware* dalam mendefinisikan arsitektur sistem secara keseluruhan.

3. *Implementation* , merupakan integrasi pada tahap selanjutnya untuk mengembangkan dan menguji fungsionalitas suatu sistem yang dibuat atau disebut dengan unit *testing*.

4. *Verification* , merupakan tahapan verifikasi dan pengujian apakah sistem secara keseluruhan dapat memenuhi persyaratan sistem.

5. *Maintance* , ini merupakan tahapan akhir dari metode *waterfall* yang dimana *software* telah jadi dan dijalankan, termasuk memperbaiki kesalahan yang tidak ditemukan pada tahapan sebelum [29].

2.13 Penelitian Terdahulu

No	Jurnal	Hasil Penelitian	Kontribusi dalam Penelitian
1	<p>Judul Jurnal : Rancang Bangun Aplikasi Chatbot Sebagai Media Pencarian Informasi Anime Menggunakan Regular Expression Pattern Matching</p> <p>Nama Jurnal : ULTIMATICS, Vol. IX, No. 1</p> <p>Tahun : 2017.</p> <p>Penulis : David Domarco, Ni Made Satvika Iswari.</p>	<p>Penelitian ini menghasilkan aplikasi chatbot yang merupakan media interaktif dalam mendapatkan informasi seputar anime berbasis teks menggunakan regular expression pattern matching telah berhasil dirancang dan dibangun dengan nama Reikobot.</p>	<p>Penelitian ini menjadi acuan dalam melakukan perancangan sistem yang dibuat.</p>
2	<p>Judul Jurnal : Rancang Bangun Aplikasi UMN Library Catalog Menggunakan Metode Rocchio Relevance Feedback.</p> <p>Nama Jurnal : ULTIMA InfoSys, Vol. IX, No. 1.</p> <p>Tahun : 2018</p> <p>Penulis : Marcel Bonar Kristanda, Seng Hansun, Albert.</p>	<p>Penelitian ini Aplikasi UMN Library Catalog untuk pencarian bukupada koleksi perpustakaan UMN dengan metode roccchio relevance feedback berhasil dirancang dan dibangun dengan bagian arsitektur yang meliputi database, API, platformAndroid dan dapat mengeluarkan hasil pencarian yang lebih relevan.</p>	<p>Penelitian ini sebagai acuan dalam proses rancang bangun sebuah sistem.</p>
3	<p>Judul Jurnal : Rancang Bangun Sistem Informasi Pengelolaan Tugas</p>	<p>Penelitian ini membuat Sistem informasi ini menghasilkan sistem yang mampu membantu dosen dalam mengelola</p>	<p>Penelitian ini sebagai referensi dalam pembuatan rancang bangun sebuah sistem.</p>

	<p>Mahasiswa Pada Matakuliah Pemrograman Sistem Bergerak Berbasis Web.</p> <p>Nama Jurnal : Jurnal Vocational Teknik Elektronika dan Informatika Vol. 9, No. 1.</p> <p>Tahun : 2021</p> <p>Penulis : Pitri yenti, Geovanne Farell</p>	<p>tugas mahasiswa pada mata kuliah Pemrograman Sistem Bergerak. Dengan bahasa pemrograman PHP dan databaseMySQL.</p>	
4	<p>Judul Jurnal : RANCANG BANGUN SISTEM INVENTORI PENGENDALIAN STOK BARANG BERBASIS JAVA PADA PT KALIBESAR ARTAH PERKASA.</p> <p>Nama Jurnal : Jurnal SITECH, Vol 3, No 1.</p> <p>Tahun : 2020</p> <p>Penulis : Tuti Handayani, Ahmad Hapip Furqon, Supriyono.</p>	<p>Penelitian ini menghasilkan Sistem yang dapat menunjang inventori pengendalian stok barang pada PT. Kalibesar Artah Perkasa yaitu dengan membuat sebuah perogram aplikasi pengendalian stok barang berbasis Java, sehingga dapat memudahkan dalam pengolahan data dan pengendalian stok barang.</p>	<p>Penelitian ini membantu dalam pengetahuan dalam rancang bangun sistem inventory dalam suatu perusahaan.</p>
5	<p>Judul Jurnal : Sistem Pendukung Keputusan Pemilihan Smartphone dengan Menerapkan Metode</p>	<p>Penelitian ini bertujuan untuk merancang sistem pendukung keputusan pemilihan smartphone dengan menerapkan metode simple additive</p>	<p>Penelitian ini membantu sebagai referensi dalam perancangan metode SAW.</p>

	<p>Simple Additive Weighting (SAW).</p> <p>Nama Jurnal : Jurnal Sistem Informasi Volume 4.</p> <p>Tahun : 2017.</p> <p>Penulis : Harsiti , Henri Aprianti.</p>	<p>weighting (SAW), sehingga dapat memberikan solusi terhadap konsumen untuk memilih smartphone. Hasil dari penelitian ini adalah menghasilkan sebuah aplikasi sistem pendukung keputusan pemilihan smartphone yang dapat membantu konsumen melakukan pemilihan smartphone sesuai dengan keinginan dan kebutuhan berdasarkan kriteria yang telah ditentukan.</p>	
--	---	--	--

