

## BAB II TINJAUAN PUSTAKA

### 2.1. Pencapaian Terdahulu

Penelitian terhadap hasil penelitian terdahulu dilakukan untuk menghindari penelitian yang sama dan mencari metode yang akan dikembangkan untuk penelitian selanjutnya. Adapun pencapaian penelitian terdahulu sebagai berikut :

1. Penelitian yang dilakukan oleh Tonggam Panggabean yang melakukan penelitian berjudul "*Penerapan Metode Simple Additive Weighting (SAW) dalam pemberian reward bagi pegawai honorer menggunakan pembobotan Rank Order Centroid*". Penelitian ini mengambil sebagai responden adalah pegawai kontrak waktu tertentu (PKWT) pada instansi Badan Pengelola Keuangan dan Aset (BPKA) kabupaten Deli Serdang Propinsi Sumatera Utara. Pemberian reward kepada pegawai honorer ini diharapkan mampu bekerja maksimal untuk mewujudkan visi misi instansi yang bersangkutan. Pada penelitian ini kriteria yang digunakan adalah beban kerja, kehadiran, kedisiplinan dan etika kerja sedangkan pembobotan masing-masing kriteria menggunakan metode ROC. (Panggabean et al., 2021)
2. Penelitian oleh Karmalia Safitri berjudul "*Sistem Pendukung Keputusan Pemilihan Karyawan Berprestasi Dengan Menggunakan Metode Analytical Hierarchy Proses (Studi kasus : PT. Capella Dinamik Nusantara Takengon)*". Ada empat kriteria yang digunakan untuk menentukan karyawan berprestasi yaitu kejujuran, kedisiplinan, rajin, dan tanggung jawab yang diolah dengan menggunakan metode AHP sehingga diperoleh pegawai yang berprestasi pada PT. Capella Dinamik Nusantara Takengon.(Safitri et al., 2017)
3. Kajian Muliati Badaruddin (2019) berjudul "*Sistem Pendukung Keputusan Penilaian Kinerja Karyawan Menerapkan Kombinasi Metode Simple Additive Weighting (SAW) dengan Rank Order Centroid (ROC)*". Untuk menilai pegawai menggunakan kriteria kualitas kerja, disiplin, kerjasama, loyalitas, teguran. Empat kriteria sebagai benefit dan satu kriteria yaitu teguran bersifat sebagai cost. Lima kriteria ini pembobotannya disusun berdasarkan ROC.(Badaruddin, 2019)

4. Penelitian yang dilakukan oleh Ahmad Luthfi Rahman, Muhammad Hasbi dan Setiowati pada tahun 2020 berjudul *"Implementasi Sistem Pendukung Keputusan untuk Pemilihan Siswa Berprestasi di Sekolah Menengah Atas dengan Metode AHP dan TOPSIS"* (Rahman et al., 2020)
5. Penelitian yang dilakukan oleh Sufiandi, Andyansyah Putera Utama Siahaan dan Alfiandi pada tahun 2020 yang berjudul *"Sistem Pendukung Keputusan Pemilihan Pegawai Honorer Kelurahan Babura dengan Metode MFEP"*. (Supiyandi et al., 2020)

## **2.2. Tinjauan Teoritis**

Pada tinjauan teoritis akan di jelaskan beberapa hal yang terkait dan diuraikan dalam penelitian yang bertujuan untuk memberikan persepsi yang sama. Adapun hal-hal tersebut sebagai berikut :

### **2.2.1. KPPBC TMP A Jakarta**

KPPBC TMP A Jakarta adalah singkatan dari Kantor Pengawasan dan Pelayanan Bea dan Cukai Tipe Madya Pabean A merupakan salah satu instansi pemerintah yang merupakan bagian vertikal dari struktur organisasi Direktorat Jenderal Bea dan Cukai (DJBC). Dan DJBC adalah merupakan bagian dari Kementerian Keuangan Republik Indonesia.

### **2.2.2. Capaian Kinerja Pegawai (CKP)**

Capaian Kinerja Pegawai atau CKP adalah alat atau tools bagi organisasi untuk mengetahui tingkat keberhasilan pegawai dalam menjalankan tugasnya. CKP berupa nilai capaian dari sasaran kinerja pegawai yang dibuat setiap awal tahun anggaran. Setiap karyawan harus menyiapkan beberapa indikator terukur secara bersamaan sasaran kinerja pegawai (SKP) yang akan dicapai pada akhir tahun dan akan dilaporkan progressnya pada setiap periode misalnya setiap bulan, triwulan, per semester dan pertahun. Pengukuran nilai indikator dapat berupa index atau prosentase sesuai formulasinya.

Setiap pegawai harus membuat SKP masing-masing secara berjenjang mulai dari yang tertinggi sampai terendah. SKP bawahan harus mendukung SKP atasannya agar terdapat harmonisasi pencapaian tujuan bersama.

Untuk mengelola CKP ini, telah menggunakan sistem informasi yang di sebut e-Performace Kemenkeu.

### **2.2.3. Nilai Prilaku Pegawai**

Selain kinerja, setiap pegawai juga akan dinilai prilakunya. Skema penilaian adalah 360 derajat, atasan langsung, peers dan bawahan akan memberikan nilai. Jumlah peers yang menilai ada dua orang yang dipilih secara random by system. Penilaian diberikan pada skala 1 sampai dengan 100. Penilaian tidak dibedakan berdasarkan jabatan. Unsur yang dinilai sebagai berikut :

1. Integritas;
2. Kerjasama;
3. Komunikasi;
4. Orientasi pada hasil;
5. Pelayanan publik;
6. Pengembangan diri dan orang lain;
7. Mengelola perubahan; dan
8. Mengambil keputusan.

Pengelolaan nilai prilaku juga sudah dilakukan menggunakan sistem informasi yang sama dengan nilai CKP yaitu pada aplikasi e-Performance Kemenkeu.

### **2.2.4. Absen**

Setiap hari kerja pegawai wajib melalukan absen kecuali mendapat ijin atau cuti. Absen sehari minimal dua kali yaitu yaitu pada waktu pagi hari sebagai jam masuk kerja dan absen sore hari untuk jam keluar kantor. Jam masuk kantor pada pukul 7.30 WIB sedangkan jam pulang kantor pukul 17.00 WIB. Pegawai yang masuk kantor melewati pukul 7.30 disebut terlambat datang dan akan tercatat berjenjang setiap 30 menit sebagai TL1, TL2, TL3 dan TL4. Begitu pula pada saat pulang kantor sebelum pukul 17.00 disebut pulang sebelum waktunya atau PSW.

Pulang sebelum waktunya juga tercatat secara berjenjang setiap 30 menit yaitu PSW1, PSW2, PSW3 dan PSW4.

Pegawai yang tercatat terlambat datang atau pulang sebelum waktunya akan dikenai sanksi pemotongan tunjangan kinerja sesuai tingkatannya. Berikut adalah tabel pemotongan pencatatan absensi pegawai :

*Tabel 2. 1 Potongan Sanksi Absen*

<b>Keterlambatan</b>	<b>Jam</b>	<b>Potongan Sanksi</b>
TL1	07.31 sd 08.01	0,5%
TL2	08,01 sd 08.31	1%
TL3	08.31 sd 09.01	1,25%
TL4	>= 09.01	2,5%
Pulang sebelum waktu		
PSW1	16.31 sd < 17.00	0,5%
PSW2	16.01 sd < 16.31	1%
PSW3	15.31 sd < 16.01	1,25%
PSW4	< 15.31	2,5%

Dalam 1 hari pegawai wajib melakukan 2 kali absen, jika dalam satu hari tidak melakukan absen maka akan dikenakan sanksi pemotongan total 5%.

Kegiatan absensi pegawai dilakukan menggunakan sistem informasi yang disebut MyCeisa yang diinstall pada masing-masing handphone pegawai dan terintegrasi dengan aplikasi CEHRIS.

### **2.2.5. Kewajiban Pelaporan Harta**

Untuk menciptakan pengelolaan negara yang baik para pejabat negara wajib melaporkan harta kekayaan setiap tahun kepada Komisi Pemberantasan Korupsi (KPK). Pejabat negara yang wajib lapor harta dimulai 1 Januari - 31 Maret untuk posisi harta tahun sebelumnya. Misalnya perolehan harta pada tahun 2022, dilaporkan pada 1 Januari - 31 Maret 2023 tahun berikutnya. Pelaporan harta kekayaan oleh pejabat pengelola negara ini disebut LHKPN. Kewajiban dan mekanisme pelaporan harta ini diatur oleh Peraturan KPK nomor 2 tahun 2020.

Kewajiban melaporkan harta khusus untuk pegawai Kementerian Keuangan diwajibkan kepada seluruh pegawai. Pegawai kementerian ini wajib melaporkan harta kepada Inspektorat Jenderal Kementerian Keuangan dengan nama Alpha. Bagi pejabat yang sudah melaporkan harta kepada KPK (LHKPN) karena aplikasi pelaporan LHKPN dan Alpha sudah terintegrasi antara KPK dan Kemenkeu, maka pejabat negara tersebut secara otomatis sudah tidak wajib mengisi Alpha. Dan bagi pegawai yang tidak wajib mengisi LHKPN maka pegawai

tersebut wajib mengisi Alpha. Kewajiban pegawai kemenkeu dalam melakukan pengisian Alpa diatur dalam Keputusan Menteri Keuangan No 227/PMK.09/2021.

Kepada seluruh pegawai kemenkeu, waktu pengisian baik LHKPN maupun Alpha untuk tahun 2022 adalah dilakukan mulai 1 Januari - 28 Februari 2023.

#### **2.2.6. Sistem Pendukung Keputusan**

Dalam kehidupan sehari-hari selalu melakukan pengambilan keputusan untuk tujuan tertentu, mulai dari keputusan yang sederhana sampai yang kompleks. Keputusan dapat berupa kegiatan memilih dari beberapa alternatif pilihan yang tersedia. Untuk itu diperlukannya sistem pendukung keputusan agar kegiatan pengambilan keputusan menjadi lebih optimal sesuai tujuannya.

Dalam sistem pendukung keputusan terbagi menjadi empat tahapan (Pribadi et al., 2020) yaitu :

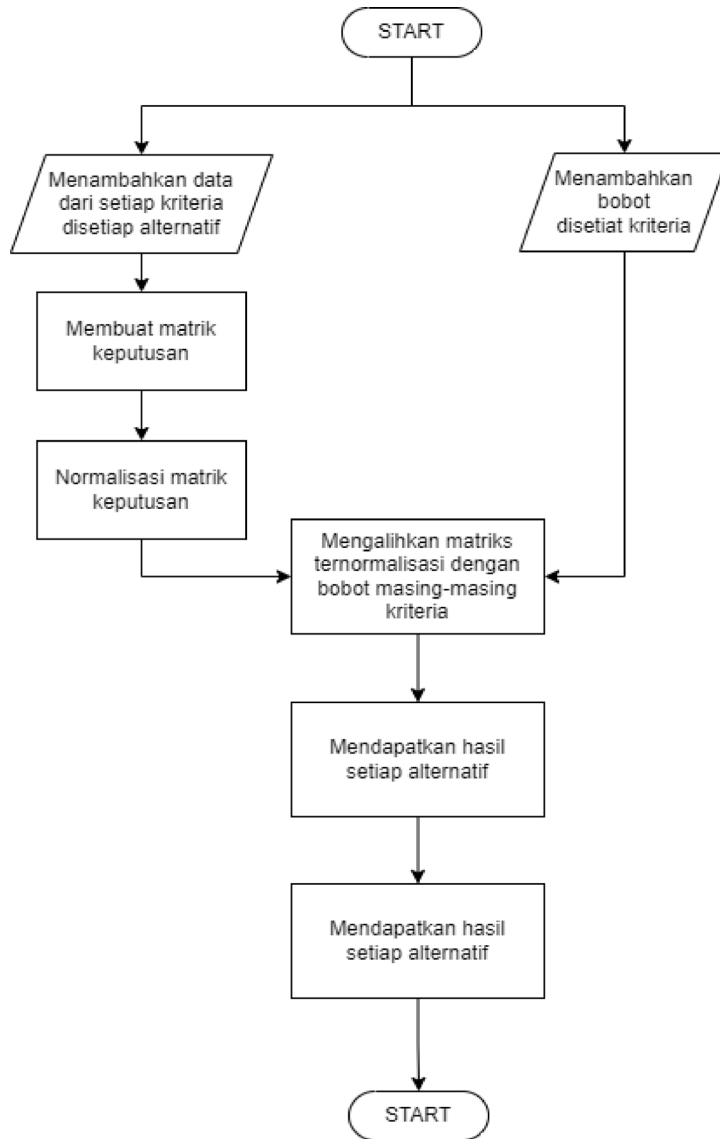
1. Tahapan pemahaman (*intelligence phase*)
2. Tahapan perancangan (*design phase*)
3. Tahapan pemilihan (*choice phase*)
4. Tahapan implementasi (*implementation phase*)

Dalam sistem pendukung keputusan banyak metode yang digunakan untuk mengambil keputusan. Metode yang akan digunakan disesuaikan dengan situasi dalam mengambil keputusannya. Adapun beberapa metode tersebut adalah :

- a. Metode *fuzzy logic*;
- b. Metode SAW (*Simple Additive Weighting*);
- c. Metode TOPSIS (*Technique for Order by Similarity Ideal Solution*);
- d. Metode AHP (*Analytic Hierarchy Process*);
- e. Dan masih banyak lagi metode yang dapat dipakai.

*Simple Additive Weighting (SAW)* adalah salah satu metode dalam sistem pendukung keputusan. Metode ini disebut juga metode penjumlahan terbobot. Konsep dari metode ini adalah menjumlahkan secara terbobot dari rating kinerja yang ada pada semua atribut.

Algoritma atau langkah-langkah metode SAW (Saputra & Aprilian, 2020) sebagai berikut :



Gambar 2. 1 Algoritma SAW

A. Mengidentifikasi kriteria yang akan menjadi acuan untuk pengambilan keputusan;

$$C_i = C_1, C_2, C_3, \dots, C_i \quad (2-1)$$

B. Menentukan penilaian kesesuaian setiap alternatif untuk setiap kriteria ( $C_i$ );

$$C_i = x_i \quad (2-2)$$

C. Menentukan matriks keputusan berbasis kriteria ( $C_{ij}$ )

$$C_{ij} = x_{ij} \quad (2-3)$$

D. Normalisasikan matriks berdasarkan persamaan yang dicocokkan dengan jenis atribut, apakah atributnya berjenis benefit atau cost. Jadi kita mendapatkan matriks dinormalisasi  $r_{ij}$  dengan rumus normalisasi sebagai berikut :

- a) Jika suatu kriteria memiliki atribut benefit, maka semakin tinggi nilainya semakin bagus. Untuk menghitung nilai normalisasi dengan rumus :

$$r_{ij} = \frac{x_{ij}}{\max_{ij}} \quad (2-4)$$

Keterangan :

$r_{ij}$  = Normalisasi untuk atribut benefit;

$X_{ij}$  = Nilai rating dari kriteria  $C_{ij}$ ;

Max  $x_{ij}$  = Nilai max dari nilai rating  $C_{ij}$ ;

- b) Jika suatu kriteria memiliki atribut cost, maka semakin kecil nilainya semakin bagus. Untuk menghitung nilai normalisasi dengan rumus :

$$r_{ij} = \frac{\min x_{ij}}{x_{ij}} \quad (2-5)$$

Keterangan :

$r_{ij}$  = Normalisasi untuk atribut cost;

$X_{ij}$  = Nilai rating dari kriteria  $C_{ij}$ ;

Min  $x_{ij}$  = Nilai min dari nilai rating  $C_{ij}$ ;

E. Hasil akhir berasal dari proses pemeringkatan yaitu jumlah perkalian matriks yang dinormalisasi  $r_{ij}$  dengan vector bobot ( $w_j$ .) Dan hasilnya diperingkat dari nilai terbesar sampai terkecil, dengan rumus :

$$V_i = \sum_{j=1}^n w_j r_{ij} \quad (2-6)$$

Keterangan :

$V_i$  = hasil penjumlahan terbobot matrik ternormalisasi;

$r_{ij}$  = Normalisasi matrik keputusan;

$w_j$  = Nilai bobot;

F. Nilai  $V_i$  yang terbesar menunjukkan bahwa opsi  $A_i$  lebih baik

### 2.2.7. Web Server

*Web Server* adalah perangkat lunak yang berjalan di sisi server. bertugas terima permintaan (*request*) berupa halaman *web* melalui HTTP atau HTTPS. Browser mendeteksi dari klien dan mengembalikan (*respons*) hasilnya berupa halaman *web* yang biasanya dalam format dokumen HTML. (solichin, 2005)

*Web server* yang banyak digunakan antara lain :

- a) *Apache web server*;
- b) *Internet Information Service*;
- c) *Xitami web server*;
- d) *Sun Java System web server*;

### **2.2.8. HTML CSS dan Javascript**

Hampir semua situs web memakai *HTML* dan *CSS*. *Browser* akan menerima *HTML* dan *CSS* dari *web server* yang menampung situs tersebut. *Web browser* menafsirkan kode *HTML* dan *CSS* Sebagian besar halaman *web* juga menyertakan konten tambahan seperti gambar, audio, video, atau animasi dan beberapa situs juga menggunakan *JavaScript* atau *Flash* ke *browser*

Situs web sederhana sering ditulis hanya menggunakan *HTML* dan *CSS*. Situs web yang kompleks khususnya yang diperbarui secara berkala dan menggunakan manajemen konten sistem (*CMS*), alat *blogging*, atau perangkat lunak *e-commerce* – sering menggunakan teknologi di-*web server*, tapi teknologi ini sebenarnya digunakan untuk menghasilkan *HTML* dan *CSS* yang kemudian dikirim ke browser. Jadi, jika situs menggunakan teknologi ini dapat menggunakan *HTML* baru dan Pengetahuan *CSS* untuk mengambil lebih banyak kontrol atas tampilan situs. Situs yang lebih kompleks seperti ini mungkin menggunakan database untuk menyimpan data, dan pemrograman bahasa seperti *PHP*, *ASP.Net*, *Java*, atau *Ruby* di-*server web*. (Duckett, n.d.)

### **2.2.9. PHP**

*PHP* merupakan bahasa pemrograman yang dapat berproses pada *server-side scripting* sedangkan *HTML* berjalan pada *client-side*. Sehingga sintak *PHP* tidak akan terlihat tidak seperti pada *HTML*

### **2.2.10. MySQL**

Database adalah sekumpulan data yang disimpan secara terstruktur sehingga dapat dengan mudah dianggil kembali pada saat diperlukan. Sedangkan



*DBMS* atau *database* manajemen sistem adalah perangkat lunak yang digunakan dalam memanajemen *database* salah satu contohnya adalah *MySQL*.

### **2.2.11. XAMPP**

*XAMPP* adalah sebuah paket perangkat lunak yang bersifat *cross platform* yang terdiri dari *Apache*, *MySQL*, *PHP* dan *Perl* sesuai dengan singkatan dari namanya. Program aplikasi *XAMPP* bertindak sebagai *server* lokal untuk menerima data dari berbagai jenis data *website*.

### **2.2.12. Visual Studio Code (VS Code)**

*VS code* adalah salah satu *text editor* yang digunakan untuk menuliskan serangkaian kode atau program untuk membuat sebuah aplikasi. *VS code* biasa digunakan untuk *developer PHP* dan dapat menggunakan *framework Code Igniter* atau *Laravel*

### **2.2.13. Metodologi Pengembangan**

Dalam rekayasa perangkat lunak, rekayasa sistem dan sistem informasi, metodologi pengembangan dikenal sebagai siklus hidup pengembangan aplikasi adalah proses merancang, membangun, menguji, dan menerapkan sistem informasi. Konsep siklus hidup pengembangan sistem berlaku untuk beberapa konfigurasi perangkat keras dan perangkat lunak, karena suatu sistem dapat berupa perangkat keras saja, perangkat lunak saja atau kombinasi keduanya. (Dalle et al., 2020)

Metodologi pengembangan adalah fase pengembangan umum dari sistem informasi. Metodologi pengembangan adalah tahapan pengembangan sistem informasi yang memiliki tahapan pengembangan sebuah sistem informasi sebagai berikut :



Gambar 2. 2 Metodologi Pengembangan

### 1. Perencanaan

Tahap untuk merencanakan membuat atau mengembangkan sistem informasi yang telah ada. Pada tahap ini merencanakan semua aspek yang ada baik waktu maupun biaya.

### 2. Analisis Kebutuhan

Ini adalah langkah untuk membangun sebuah sistem informasi yaitu analisis kebutuhan. Tahap ini adalah tahap menjawab pertanyaan “apa” dan “mengapa”, yaitu sistem seperti apa yang akan dibuat dan mengapa sistem itu dibuat ada kebutuhan apa sistem itu dibuat. Makanya tahap ini disebut tahap analisa kebutuhan, jangan sampai membuat sistem informasi tetapi tidak mengetahui untuk apa sistem ini dibuat atau membuat sistem informasi yang tidak dibutuhkan.

### 3. Perancangan

Tahap ini adalah design atau perancangan yaitu tahap untuk penggambaran dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam

satu kesatuan utuh sistem informasi. Jadi kalau sudah mengetahui kebutuhannya apa, lalu digambarkan kebutuhan tersebut menjadi sebuah perancangan. Perancangan ini dapat berupa data alir diagram untuk menggambarkan aliran data dalam sistem yang akan dibuat, rancangan databasenya, sketsa tampilan dari sistemnya seperti apa?

#### 4. Penerapan

Setelah perancangan dibuat adalah tahap penerapan atau pembuatan sistemnya sesuai perancangan yang sudah dibuat itu. Dalam tahap ini yang dilakukan adalah membuat koding aplikasi sistem informasinya.

#### 5. Pengujian

Tahap berikutnya adalah tahap pengujian sistem. Pada tahap ini, program atau perangkat lunak sistem informasi yang dibuat sepenuhnya diuji yaitu menguji kualitas perangkat lunak yang dibuat apakah masih ada kesalahan atautkah sudah berjalan dengan baik. Juga melakukan pengujian terhadap pengguna, apakah sistem yang dibuat sudah sesuai dengan pengguna.

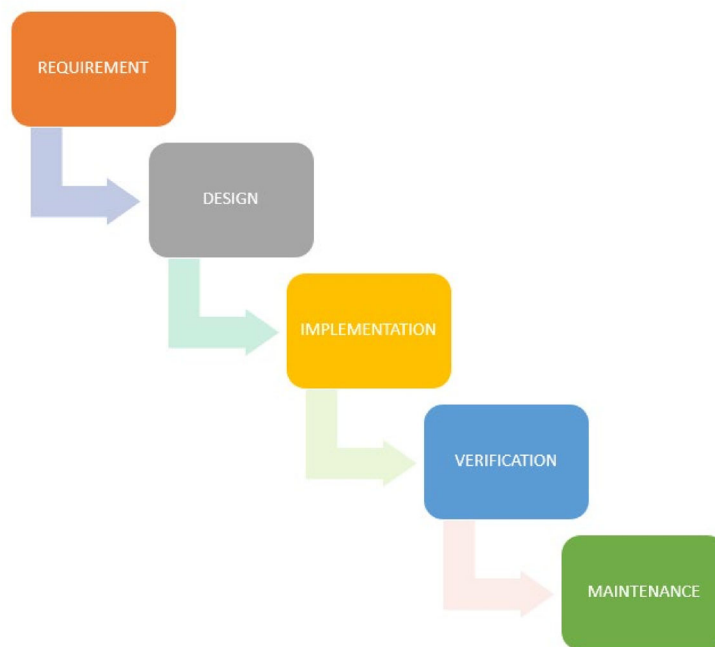
#### 6. Pemeliharaan)

Tahap yang terakhir adalah tahap pemeliharaan yaitu tahap dimana dalam operasionalisasi sistem informasi perlu dilakukan pemeliharaan sistem. Tidak menutup kemungkinan sistem akan berubah dan berkembang sesuai kebutuhan. Sehingga sistem yang telah dibuat tidak menjadi sistem yang mati yang tidak berubah dan berkembang.

Model metodologi pengembangan, terdapat beberapa model dalam pengembangan seperti :

1. *Waterfall Model,*
2. *Prototype Model,*
3. *Rapid Application Development Model,*
4. *Evolutionray Software Process Models,*
5. *Agile Model,*
6. *Fountain Model,*
7. *Syncronize and Stabilize Models*

*Waterfall Model* adalah salah satu model dalam melakukan pengembangan perangkat lunak. *Waterfall* merupakan model pengembangan klasik yang menggambarkan pengembangan perangkat lunak secara linier dan berurutan. Disebut *waterfall* karena bentuknya seperti aliran pada air terjun. Ada lima tahapan pada *waterfall model* yaitu analisis kebutuhan, perancangan, penerapan, pengujian dan pemeliharaan. Setiap process disajikan terpisah, misalnya tahap perancangan dapat dilakukan setelah tahap analisa kebutuhan selesai, kalau tahap analisa kebutuhan belum selesai maka tahap perancangan tidak dapat dilakukan.



Gambar 2. 3 Waterfall

### 2.2.13. Perancangan Sistem

Perancangan sistem menurut *yakub*, “perancangan suatu sistem dapat didefinisikan sebagai perancangan, perencanaan dan pembuatan sketsa, atau sebagai pengorganisasian beberapa elemen terpisah menjadi satu kesatuan ” (Yakub, 2012)

Tujuan perancangan sistem ini adalah

1. Untuk memenuhi kebutuhan pemakai sistem

2. Pastikan gambar yang jelas dan desain yang sempurna kepada pemrogram komputer dan ahli-ahli teknik lainnya yang terlibat.

Untuk mencapai tujuan ini, analisis harus mencapai sasaran berikut :

1. Desain sistem harus bermanfaat, mudah dipahami dan mudah digunakan nantinya.
2. Perancangan sistem harus mendukung tujuan utama.
3. Perancangan sistem harus efisien dan efektif untuk dapat mendukung pengolahan transaksi, pelaporan manajemen, dan mendukung keputusan manajemen,
4. Perancangan sistem harus dapat mempersiapkan rancang bangun yang terinci untuk masing-masing komponen sistem.

Personil yang terlibat :

1. Spesialis pengendalian
2. Personil penjaminan kualitas
3. Spesialis komunikasi data
4. Pemakai sistem

Perancangan sistem secara umum

1. Perancangan *output*,
2. Perancangan *input*
3. Perancangan *proses*
4. Perancangan *database*
5. Perancangan *kontrol*
6. Perancangan *jaringan*
7. Perancangan komputer

Perancangan sistem dengan *UML*, *UML* singkatan dari *Unified Modeling Language*. Juga *UML* menjadi cara untuk membuat pengembangan aplikasi lebih mudah kontinu. *UML* adalah seperangkat alat yang digunakan untuk eksekusi abstraksi dari sistem berorientasi objek atau perangkat lunak berorientasi objek. (Fajar, 2020)

*UML* dapat didefinisikan sebagai suatu bahasa standar visualisasi, perancangan dan pendokumentasian sistem atau dikenal juga sebagai bahasa standar penulisan cetak biru sebuah perangkat lunak.

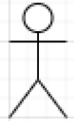
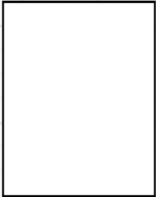
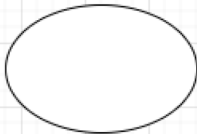
*UML* juga dimaksudkan untuk memfasilitasi pengembangan perangkat lunak (RPL) memenuhi semua kebutuhan pengguna secara efisien, lengkap dan akurat. Hal itu termasuk skalabilitas, daya tahan, keamanan dan lain lain.

Berapa banyak *UML* yang umum digunakan, seperti :

1. *Use Case Diagram*,

Yaitu jenis diagram *UML* yang menggambarkan interaksi antara sistem dan aktor. Sebuah *Use Case* dapat menggambarkan sifat interaksi antara si pengguna sistem dengan sistem. *Use Case* mudah dipelajari. Bagian yang valid diperlukan pada tahap awal menggambarkan Tindakan aktor melalui tindakan sistem itu sendiri seperti yang ditemukan.



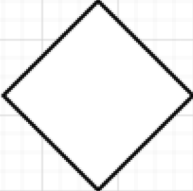

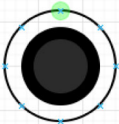
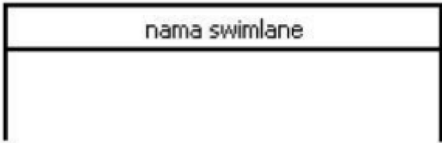
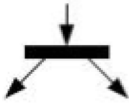

Tabel 2. 2 Simbol *Use Case Diagram*

No..	Gambar	Nama	Keterangan
1.		Actor	Mendefinisikan sekumpulan peran yang dilakukan pengguna dalam interaksi <i>use case</i>
2.	<code>&lt;&lt;include&gt;&gt;</code> ----->	Include	Secara eksplisit menyatakan sumber <i>use case</i>
3.	<code>&lt;&lt;extend&gt;&gt;</code> <-----	extend	Menunjukkan bahwa <i>use case</i> target memperluas perilaku usecase sumber di beberapa titik
4.	_____	Association	Apa yang menghubungkan satu obyek dengan objek lainnya
5.		sistem	Menentukan paket yang berisi sistem terbatas
6.		<i>Use case</i>	Deskripsi urutan tindakan yang ditunjukkan oleh sistem yang memberikan hasil terukur bagi operator

2. *Activity Diagram*,

atau dalam bahasa Indonesia berarti diagram aktifitas, merupakan sebuah diagram yang dapat memodelkan berbagai proses yang terjadi di dalam sistem. Sebagai sekumpulan proses yang mengontrol sistem dan digambarkan secara vertikal. Diagram aktivitas adalah contoh diagram *UML* dalam mengembangkan usecase.

Tabel 2. 1 Simbol Activity Diagram

Simbol	Uraian
Status awal 	Keadaan awal operasi sistem, keadaan awal diagram aktivitas
Aktivitas 	Aktivitas yang dilakukan oleh sistem biasanya diawali dengan kata kerja
Percabangan / decision 	Assosiasi percabangan dimana jika ada pilihan aktifitas lebih dari penggabungan
Penggabungan atau join 	Assosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi akhir
Status akhir 	Keadaan akhir dilakukan oleh sistem. Diagram aktivitas memiliki keadaan akhir
Swimlane 	Sebuah organisasi bisnis yang bertanggung jawab untuk kegiatan tertentu
Fork 	Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel
Join 	Digunakan untuk menunjukkan kegiatan yang digabungkan




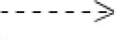

### 3. Class Diagram

Ini disebut jenis diagram structural karena menggambarkan apa yang harus dimiliki oleh berbagai komponen dalam sistem yang dimodelkan. Berbagai komponen ini dapat mewakili kelas yang dapat diprogram, objek utama, atau

interaksi antara kelas dan objek. Kelas itu sendiri adalah istilah yang menjelaskan sekelompok objek yang semuanya memainkan peran serupa dalam sistem.

Kelompok objek ini terdiri dari property struktural yang mendefinisikan apa yang bisa menurut kategori.

Tabel 2. 2 Simbol Class Diagram

No	Gambar	Nama	Keterangan
1.		generalisation	Hubungan dimana obyek anak (keturunan) berbagi perilaku dan struktur data objek di atas obyek induk (ancestor)
2		Navy association	Mencoba menghindari menghubungkan ke lebih dari dua objek
3		Class	Himpunan dari obyek-obyek yang memiliki atribut dan fungsi yang sama
4		Collaboration	Deskripsi urutan tindakan yang ditunjukkan oleh sistem yang memberikan hasil terukur bagi aktor
5		Realization	Operasi yang benar benar dilakukan oleh suatu obyek
6		dependency	Hubungan di mana perubahan anggota independent mempengaruhi anggota yang bergantung pada anggota non-independent
7		Assosiation	Apa yang menghubungkan suatu objek dengan objek lain