

## BAB II LANDASAN TEORI

### 2.1 Tinjauan Pustaka

Berdasarkan hasil yang didapatkan melalui kajian literatur terhadap penelitian terdahulu yang telah dilakukan. Maka penulis merujuk pada beberapa literatur yang cukup berkaitan dengan penelitian ini. Berikut ialah tabel kajian literatur pustaka yang menjadi referensi penulis, dapat dilihat pada tabel 2.1.

Tabel 2. 1 Tinjauan Pustaka

| No | Detail Referensi | Keterangan   |
|----|------------------|--|
| 1  | Judul            | Sistem Pendukung Keputusan Penerimaan Calon Siswa Baru pada SMK SMTI Bandar Lampung dengan Menggunakan Metode <i>Simple Additive Weighting</i> (SAW)   |
|    | Tahun            | 2017   |
|    | Metode           | <i>Simple Additive Weighting</i> (SAW)   |
|    | Permasalahan     | “Pada penerimaan calon siswa baru pada SMK SMTI Bandar Lampung dalam mengambil keputusan kurang objektif karena dalam pengambilan keputusan membutuhkan waktu yang lama, karena disebabkan oleh data calon siswa yang diolah cukup banyak dan waktu yang lama dan data terbatas” (Priandika and Wantoro, 2017).  |
|    | Hasil            | “Hasil dari penelitian ini adalah sebuah sistem pendukung keputusan penerimaan calon siswa baru menggunakan metode <i>Simple Additive Weighting</i> (SAW) yaitu pola perhitungan yang digunakan dengan penjumlahan terbobot dari ranting kinerja ada setiap alternatif pada semua atribut dan dapat membantu pihak sekolah sebagai pengambilan keputusan” (Priandika and Wantoro, 2017). |

Tabel 2.1 Tinjauan Pustaka (Lanjutan)

| No | Detail Referensi | Keterangan   |
|----|------------------|--|
| 2  | Judul            | Sistem Pendukung Keputusan Pemberian Penghargaan UMKM Skala Mikro Di Kabupaten Bandung Barat Menggunakan Metode <i>Analytic Hierarchy Process</i>  |
|    | Tahun            | 2021   |
|    | Metode           | <i>Analytic Hierarchy Process</i>  |
|    | Permasalahan     | “Penentuan prioritas UMKM yang terbaik perlu dilakukan, dengan menentukan prioritas UMKM yang terbaik. Jumlah UMKM yang banyak dan jumlah kriteria penilaian yang cukup banyak menyulitkan dalam melakukan penilaian untuk pemberian penghargaan dan bantuan terhadap UMKM yang terbaik” (Hadiana, Witanti and Sabrina, 2021). |
|    | Hasil            | “Sistem yang dibangun menggunakan metode Analytic Hierarchy Process (AHP) dengan data kriteria sebanyak 6 data diantaranya penilaian kuisioner, jumlah karyawan, aset, omset, Kualitas Produk, dan nilai kekatifan“ (Hadiana, Witanti and Sabrina, 2021).  |
| 3  | Judul            | Sistem Pendukung Keputusan Penentuan Prioritas UMKM di Kabupaten Bandung Barat Menggunakan TOPSIS  |
|    | Tahun            | 2018   |
|    | Metode           | <i>Technique for Order Preference by Similarity to Ideal Solution (TOPSIS)</i>   |
|    | Permasalahan     | “Menentukan pelaku UMKM yang produktif dapat membantu pemerintah dalam menyalurkan segala bantuannya dengan tepat sasaran” (Sumirah, Abdillah and Komarudin, 2018).  |
|    | Hasil            | “Penelitian ini telah membangun sistem yang mampu merekomendasikan pelaku usaha UMKM yang produktif dari segi omset, aset, jumlah produksi, jumlah permintaan, jumlah penjualan, tenaga kerja dan kualitas produk dengan bobot yang ditentukan oleh pengguna” (Sumirah, Abdillah and Komarudin, 2018).                         |

Tabel 2.1 Tinjauan Pustaka (Lanjutan)

| No    | Detail Referensi   | Keterangan   |
|-------|--|--|
| 4     | Judul  | Implementasi Sistem Pendukung Keputusan dengan Metode SAW dalam Pemilihan Guru Terbaik.  |
|       | Tahun  | 2020   |
|       | Metode   | <i>Simple Additive Weighting (SAW)</i>   |
|       | Permasalahan   | “Memiliki guru yang profesional merupakan suatu keharusan bagi sekolah dalam melaksanakan proses belajar mengajar yang bermutu, demikian pula dengan SMKN 1 Kadipaten. Oleh sebab itu, sekolah selalu mendorong peningkatan kualitas guru dengan cara memantau kerja guru dalam menerapkan tugasnya sehingga dapat mencapai standar kompetensi yang telah ditentukan” (Apriani, Krisnawati and Fitrisari, 2021). |
| Hasil | “Sistem yang dibangun menggunakan metode SAW dapat menentukan nilai bobot untuk setiap atribut, kemudian dilanjutkan dengan proses perankingan yang akan menyeleksi alternatif terbaik dari sejumlah alternatif yang ada. Dalam hal ini alternatif yang dimaksud adalah penentuan guru terbaik pada SMKN 1 Kadipaten“ (Apriani, Krisnawati and Fitrisari, 2021). |  |
| 5     | Judul  | Sistem Pendukung Keputusan Pemilihan Rumah Tinggal di Perumahan Menggunakan Metode SAW Studi Kasus Kota Samarinda.   |
|       | Tahun  | 2017   |
|       | Metode   | <i>Simple Additive Weighting (SAW)</i>   |
|       | Permasalahan   | “Tingginya permintaan konsumen dalam memiliki rumah atau hunian khususnya di kota Samarinda membuat konsumen harus teliti dalam memilih rumah yang sesuai dengan kriteria-kriteria yang diinginkan. Menyadari itu semua akhir-akhir ini banyak developer menawarkan berbagai alternative rumah dari mulai harga, lokasi, desain, maupun cara pembayaran” (Adianto <i>et al.</i> , 2017).                         |
| Hasil | “Penelitian ini menghasilkan aplikasi berbasis website guna membantu konsumen memilih perumahan yang diinginkan dengan menggunakan metode SAW” (Adianto <i>et al.</i> , 2017).   |  |

## 2.2 Data

Data merupakan tampilan dari fakta atau gambaran mengenai suatu objek atau kejadian. Data dinyatakan dengan nilai yang berbentuk angka, deret karakter atau simbol (Kusrini, 2007). Fungsi dari data yaitu digunakan sebagai bahan evaluasi dalam menanggapi suatu masalah tertentu dan memiliki fungsi untuk memecahkan sebuah masalah dan menentukan suatu kebijakan serta keputusan (Suaidah and Lathifah, 2021).

## 2.3 Informasi

Informasi adalah hasil pemrosesan data di mana data tersebut sudah diproses dan dipresentasikan menjadi suatu yang bermakna untuk pengambilan keputusan. Informasi juga dapat diartikan sebagai himpunan dari data yang nyata dengan satu atau beberapa orang dalam suatu waktu. Informasi sangat mungkin akan menjadi data dalam proses yang akan menghasilkan menjadi kurang berguna (Kusrini, 2007). Suatu informasi berguna bagi pembuat keputusan karena informasi bisa menurunkan ketidakpastian tentang hal yang sedang dipikirkan. Kualitas informasi memiliki karakteristik informasi yang berkualitas yaitu relevan, akurat, lengkap, tepat waktu, dapat dipahami dan dapat dibandingkan (Puspitasari and Budiman, 2021).

## 2.4 Sistem

Dikutip dari (Kusrini, 2007) menurut Hall Sistem adalah kumpulan dari komponen yang berfungsi untuk mengumpulkan, memproses, menyimpan, menganalisis dan menyebarkan informasi untuk tujuan menghasilkan keluaran (*output*). Syarat dari sebuah sistem adalah sistem harus dibentuk untuk menyelesaikan masalah yang terjadi, mempunyai elemen sistem yang sudah

ditentukan dan tujuan memiliki tujuan elemen yang jelas (Septilia and Styawati, 2020).

Berdasarkan (Diana, 2018) komponen sistem informasi terdiri dari masukan, proses dan keluaran. Masukan adalah proses pengambilan keputusan adalah data dan informasi. Data dapat berupa suatu keadaan, gambar, suara, huruf, angka atau Bahasa yang dapat digunakan sebagai bahan untuk melihat lingkungan, objek, kejadian ataupun suatu konsep (Fariyanto and Ulum, 2021). Proses adalah pengambilan keputusan merupakan Langkah-langkah yang diambil oleh seorang pengambilan keputusan untuk mendapatkan keputusan yang terbaik (Sri Wahyuni and Ayu Megawaty, 2021). Keluaran dari proses pengambilan keputusan adalah keputusan yang dipilih oleh seseorang pengambil keputusan, dimana keputusan ini tentunya merupakan keputusan terbaik.

## 2.5 Keputusan

Keputusan merupakan proses dalam memilih strategi atau aksi dalam pemecahan masalah tersebut (Rasyid Ridho *et al.*, 2020). Aksi memilih strategi yang dipercayai oleh pembuat keputusan akan memberikan solusi terbaik atas sesuatu yang digunakan dalam pengambilan keputusan (Kusrini, 2007). Kriteria memiliki ciri-ciri keputusan adalah :

1. Banyak pilihan,
2. Adanya kendala atau syarat,
3. Mengikuti suatu model tingkah laku, baik yang terstruktur maupun tidak terstruktur,
4. Banyak input/variabel,
5. Adanya faktor resiko,

6. Membutuhkan kecepatan, ketepatan dan keakuratan.

## 2.6 Sistem Pendukung Keputusan

### 2.6.1 Pengertian SPK

Sistem adalah kumpulan elemen yang saling berkaitan yang bertanggung jawab memproses masukan (*input*) sehingga menghasilkan keluaran (*output*). Sistem Pendukung Keputusan (SPK) atau *Decision Support Sistem* (DSS) adalah sistem informasi interaktif yang menyediakan informasi, permodelan dan manipulasi data (Niis Molo, Kelen and Rema, 2020). Sistem itu digunakan untuk membantu pengambilan keputusan dalam situasi yang semiterstruktur dan situasi yang tidak terstruktur, di mana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat (Kusrini, 2007). Sistem Pendukung Keputusan (SPK) merupakan sebuah aplikasi yang dapat memberikan beberapa solusi pemecahan masalah dengan kondisi permasalahan semi terstruktur dan tak terstruktur. Aplikasi dapat diterapkan untuk membantu user dalam pengambilan keputusan dalam situasi semi terstruktur dan situasi yang tidak terstruktur, dimana tidak ada satu orang pun tahu secara pasti bagaimana keputusan seharusnya dibuat (Ashari and Suaidah, 2020). SPK lebih bertujuan untuk mendukung manajemen dalam melakukan pekerjaan yang bersifat analitis dalam situasi yang kurang terstruktur dan dengan kriteria yang kurang jelas (Saraski *et al.*, 2022). SPK tidak dimaksudkan untuk melakukan otomatisasi pengambilan keputusan, tetapi memberikan perangkat interaktif yang memungkinkan pengambilan keputusan untuk melakukan berbagai analisis menggunakan model-model yang tersedia (Lannidya Prameswari, Astuti and Widya Ariestya, 2022). Sistem pendukung keputusan (Decision Support System) merupakan sistem informasi intraktif yang menyediakan

informasi, pemodelan, dan pemanipulasian data. Sistem ini digunakan untuk membantu pengambilan keputusan dalam situasi yang semiterstruktur dan situasi yang tidak terstruktur, dimana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat (Nur Cahya and Suaidah, 2021).

### 2.6.2 Tujuan SPK

Tujuan dari SPK adalah menurut (Kusrini, 2007) :

1. Membantu manajer dalam pengambilan keputusan atau masalah yang terjadi.
2. Memberikan dukungan atas pertimbangan manager dan bukan untuk menggantikan fungsi manajer.
3. Meningkatkan efektivitas keputusan yang diambil manager lebih dari pada perbaikan efisiensinya.
4. Kecepatan komputasi, komputer dapat memungkinkan para pengguna untuk menggunakan sistem dengan secara cepat dan biaya yang rendah.
5. Peningkatan produktivitas, membangun kelompok pengambilan keputusan terutama para user.

### 2.6.3 Langkah – Langkah Permodelan SPK

Saat melakukan permodelan dalam pembangunan SPK dilakukan langkah-langkah sebagai berikut menurut (Kusrini, 2007) :

#### 1. Studi Kelayakan (*Intelligence*)

Pada tahapan ini sasaran ditentukan dan dilakukan pencarian prosedur, pengumpulan data, identifikasi masalah, identifikasi kepemilikan masalah, klasifikasi masalah, hingga akhirnya terbentuk sebuah pernyataan masalah.

#### 2. Perancangan (*Design*)

Pada tahap ini akan dirancang model yang akan digunakan dan kriteria-kriteria yang ditentukan. Setelah itu, dicari alternatif model yang bisa menyelesaikan permasalahan tersebut. Kemudian tentukan variabel model.

### 3. Pemilihan (*Choise*)

Setelah tahapan *design* ditentukan berbagai alternatif model beserta variabelnya yang ditentukan. Tahap ini juga akan dilakukan pemilihan modelnya termasuk solusi dari model tersebut.

### 4. Membuat SPK

Setelah menentukan model, selanjutnya adalah mengimplementasikannya dalam aplikasi SPK.

## 2.7 UMKM

Menurut undang-undang no 20 tahun 2008 tentang Usaha Mikro, Kecil dan Menengah, UMKM merupakan usaha produktif orang perorangan dan ataupun badan usaha perorangan yang memenuhi kriteria usaha mikro, kecil dan menengah sebagaimana diatur dalam undang-undang. UMKM memiliki kedudukan yang sangat berarti dalam menggerakkan roda perekonomian khususnya di Provinsi Lampung. pengelolaan usaha ini dilakukan secara simpel sehingga lebih banyak menciptakan nilai ekonomi (Kurniawati and Ahmad, 2021). Keunggulan UMKM dalam bertahan dari badai krissi karena bermacam sebab, Awal, umumnya UMKM menciptakan barang konsumsi serta jasa yang dekat dengan kebutuhan publik. Kedua, UMKM tidak mengandalkan bahan baku import serta lebih menggunakan sumber daya local mulai dari bahan baku, sumber daya manusia serta perlengkapan kerjanya (Oktaviani, Ari Aldino and Arra Putri, 2022). Ketiga, rata-rata bisnis UMKM memakai modal relatif rendah.



## 2.8 UPTD PLUT KUMKM Provinsi Lampung

UPTD PLUT KUMKM Provinsi Lampung merupakan kependekan dari Unit Pelaksana Teknis Dinas Pusat Layanan Usaha Terpadu Koperasi dan Usaha Mikro Kecil dan Menengah. Dasar hukum pembentukan UPTD PLUT-KUMKM adalah dalam Peraturan Gubernur Lampung Nomor 44 tahun 2018 tentang Pembentukan Organisasi dan Tata Kerja Unit Pelaksana Teknis Daerah pada Dinas Koperasi dan UMKM Provinsi Lampung. Tugas UPTD PLUT-KUMKM adalah sesuai dengan Peraturan Gubernur Lampung Nomor 44 tahun 2018 adalah :

1. Melaksanakan operasional pelayanan kepada masyarakat dalam bidang Koperasi dan Usaha Mikro, Kecil dan Menengah.
2. Menyusun rencana teknis operasional di bidang pengkajian, pengembangan dan menyelenggarakan Pendidikan dan diklat bagi KUMKM dan masyarakat.

## 2.9 Metode SAW

### 2.9.1 Pengertian SAW

*Simple Additive Weighting* (SAW) sering juga dikenal istilah metode penjumlahan terbobot. Konsep metode SAW adalah mencari penjumlahan terbobot dari ranting kinerja pada setiap alternatif pada semua atribut. Metode SAW membutuhkan proses normalisasi matriks keputusan ( $X$ ) ke suatu skala yang dapat diperbandingkan dengan semua ranting alternatif yang ada (Pahu, Rizkia Putri and Renaldo, 2018). Metode SAW harus menentukan bobot bagi setiap atribut. Nilai total untuk alternative diperoleh dengan menjumlahkan seluruh hasil perkalian antara ranting dan bobot tiap atribut (Diana, 2018).

Diberikan persamaan normalisasi :

$$= \begin{cases} \frac{x_{ij}}{\max_i x_{ij}} & \text{jika } j \text{ adalah atribut keuntungan (benefit)} \\ \frac{\min_i x_{ij}}{x_{ij}} & \text{jika } j \text{ adalah atribut biaya(cost)} \end{cases} \dots (1)$$

Dengan  $r_{ij}$  adalah ranting kinerja ternormalisasi dari alternatif  $A_i$  pada atribut  $C_j$ ;  $i=1,2,\dots,m$  dan  $j=1,2,\dots,n$ . nilai preferensi untuk setiap alternatif ( $V_i$ ) diberikan persamaan ke dua :

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

Keterangan :

$V_i$  = nilai preferensi

$W_{ij}$  = bobot rangking

$R_{ij}$  = ranting kinerja ternormalisasi

### 2.9.2 Tahapan SAW

Tahapan dari metode SAW menurut (Adianto *et al.*, 2017) adalah :

1. Menentukan kriteria-kriteria yang akan dijadikan patokan untuk pengambilan keputusan.
2. Menentukan rating pencocokan setiap alternatif pada setiap kriteria.
3. Membuat matriks keputusan berdasarkan kriteria, kemudian melakukan normalisasi matriks berdasarkan persamaan yang disesuaikan dengan jenis atribut sehingga diperoleh matriks ternormalisasi.

4. Hasil akhir diperoleh dari proses perankingan yaitu penjumlahan dari perkalian matriks ternormalisasi dengan nilai bobot sehingga diperoleh nilai terbesar yang dipilih sebagai alternative terbaik sebagai solusi.

#### 2.10 Website

*Website* merupakan sebuah kumpulan halaman yang berisikan dokumen visual seperti teks, gambar, suara, animasi dan video di dalamnya yang menggunakan *hypertext transfer protocol* dengan media internet dan dapat diakses dengan menggunakan *browser* (Setiawan and Ramdany, 2019).

#### 2.11 PHP

PHP (*Hypertext Preprocessor*) yaitu salah satu bahasa pemrograman web *server-side* yang bersifat *open source* yang terintegrasi dengan HTML dan berada pada *server* (Setiawan and Ramdany, 2019). PHP pertama kali dibuat oleh Rasmus Lerdoft pada tahun 1995. Pada waktu itu PHP bernama *Form Interpreted (FI)*, yang wujudnya berupa sekumpulan script yang digunakan untuk mengelola data formulir dari web. Berikut kelebihan bahasa pemrograman PHP :

1. Banyaknya web server yang mendukung bahasa pemrograman PHP sehingga konfigurasi semakin mudah.
2. Pengembangan bahasa pemrograman PHP tergolong lebih mudah karena banyak programmer yang membantu dalam mengembangkan maupun menggunakannya.
3. Relative mudah untuk dipahami, karena materi-materi untuk mempelajari PHP sudah banyak.
4. Bahasa pemrograman PHP juga dapat disisipkan ke dalam HTML

5. Cocok digunakan untuk pemrograman web dinamis, walau bisa juga untuk membuat program computer lainnya.
6. PHP merupakan bahasa pemrograman bersifat open source, sehingga dapat digunakan di banyak operasi komputer dan tentunya gratis.

## 2.12 Laravel

Laravel adalah sebuah *framework* PHP yang menggunakan konsep MVC (*Model View Controller*) untuk membuat website yang powerful (Luthfi, 2017). Sekarang versi dari Laravel sudah masuk ke dalam versi 9. Prinsip Laravel adalah lebih kepada pembuatan website dengan cepat, sederhana dan fleksibel. Ada beberapa fitur di dalam Laravel yaitu *Migration* untuk melakukan modifikasi skema *database*, *Eloquent* untuk melakukan fungsi *query* dalam *database*, *Artisan* berfungsi sebagai perintah untuk berinteraksi dengan Laravel dan *Blade Template Engine* yang memberikan pengalaman yang baru dalam menuliskan kode program pada *view* (Lestari and Savitri Puspaningrum, 2021).

## 2.13 Unified Modeling Language


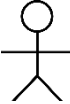


*Unified Modeling Language* atau UML merupakan mekanisme standar industri untuk memvisualisasikan, menentukan, membangun dan mendokumentasikan sistem perangkat lunak. UML telah menjadi standar resmi untuk kebutuhan seperti pemodelan data, pemodelan bisnis, dan pengembangan *real time* (Rosa and Salahuddin, 2018).

### 2.13.1 Use Case Diagram

*Usecase Diagram* merupakan permodelan untuk menggambarkan *behavior* sistem informasi yang akan dibuat (Rosa and Salahuddin, 2018). *Usecase* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem

informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Berikut symbol-simbol yang akan digunakan dalam menggambarkan *Usecase Diagram*, dapat dilihat pada tabel dibawah ini :

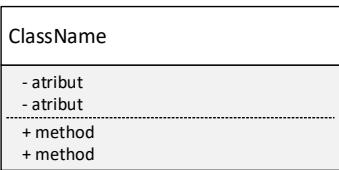




Tabel 2. 2 *Usecase Diagram*

| No | Simbol  | Deskripsi   |
|----|---|---|
| 1  |    | <i>Usecase</i><br>Fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>usecase</i> .               |
| 2  |    | <i>Actor</i><br>Actor seseorang/sesuatu yang berinteraksi dengan yang akan dibuat. Diluar sistem informasi, biasanya dinyatakan menggunakan kata benda.   |
| 3  |    | <i>Asosiasi/association</i><br><i>Asosiasi/association</i> merupakan komunikasi antara <i>actor</i> dan <i>usecase</i> yang berpartisipasi pada <i>usecase</i> atau <i>usecase</i> memiliki interkasi dengan <i>actor</i> . |
| 4  |  | <i>Generalisasi (Generalization)</i><br>Merupakan hubungan (umum – khusus) antara dua buah <i>usecase</i> dimana fungsi yang satu adalah fungsi yang lebih umum   |
| 5  | --<<include>>-->  | <i>Include</i><br><i>Include</i> berarti <i>usecase</i> yang ditambahkan akan dipanggil saat <i>usecase</i> tambahan dijalankan.  |
| 6  | ---<<extend>>---  | <i>Extend</i><br><i>Extend</i> merupakn <i>usecase</i> tambahan ke sebuah <i>usecase</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>usecase</i> tambahan itu.  |

### 2.13.2 Class Diagram

*Class Diagram* adalah sebuah diagram yang menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem(Rosa and Salahuddin, 2018). Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Class Diagram*, dapat dilihat pada tabel dibawah ini :

Tabel 2. 3 *Class Diagram*

| No | Simbol  | Deskripsi  |
|----|---|--|
| 1  |    | <i>Class</i> adalah Blok-blok pembangun pada pemrograman berbasis objek yang terdiri dari tiga bagian yaitu class, atribut dan method.   |
| 2  |    | <i>Interface</i> atau yang disebut dengan antar muka.  |
| 3  |    | Asosiasi ( <i>Asociation</i> )<br>Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan simbol.   |
| 4  |  | Asosiasi Berarah ( <i>Directed Association</i> )<br>Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan simbol. |
| 5  |  | Generalisasi adalah relasi antar kelas dengan makna generalisasi spesialisasi (umum khusus)  |

### 2.13.3 Activity Diagram

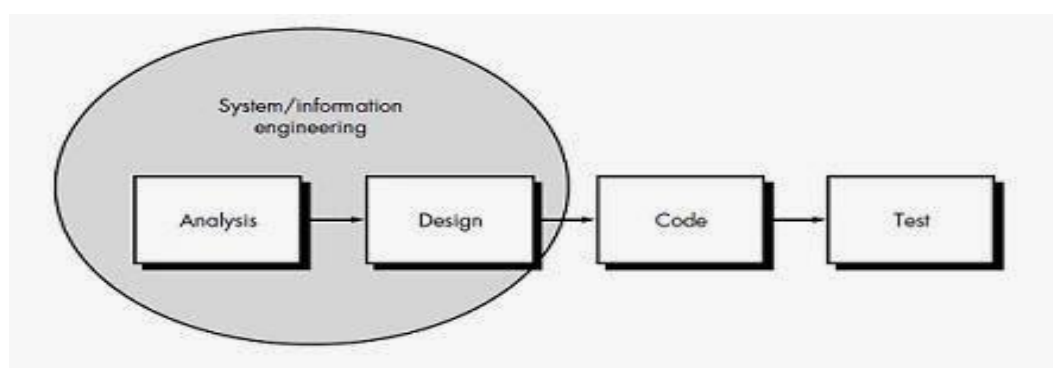
*Activity* diagram adalah diagram yang menggambarkan *workflow* atau aliran kerja dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak (Rosa and Salahuddin, 2018). Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Activity Diagram*, dapat dilihat pada tabel dibawah ini :

Tabel 2. 4 *Activity Diagram*

| No | Simbol | Deskripsi  |
|----|--------|--|
| 1  | ●      | Status awal aktivitas sistem, sebuah awal proses dari sebuah sistem.   |
| 2  | ▭      | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.   |
| 3  | ◇      | Percabangan ( <i>Decision</i> ) merupakan asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.      |
| 4  | ▭      | Pengabungan ( <i>Join</i> ) merupakan asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu. |
| 5  | ●      | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.                             |

#### 2.14 *SDLC Waterfall*

Model SDLC atau air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linier*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, dan pengujian (Rosa and Salahuddin, 2018). *Waterfall* adalah model pengembangan paling handal dan paling lama digunakan Cocok untuk sistem *software* dengan kompleksitas rendah (*predictable*). Pengerjaan *project* sistem akan terjadwal dengan baik dan mudah dikontrol. Berikut adalah ilustrasi model *Waterfall* dapat dilihat pada gambar di bawah ini.

Gambar 2. 1 Proses Model *Waterfall*

Sumber (Rosa and Salahuddin, 2018)

Berikut adalah penjelasan dari tahapan metode *Waterfall* :

1. *Analysis*

Pada tahap ini pengembang sistem dibutuhkan sesuatu komunikasi yang bertujuan buat memahami aplikasi yang diperlukan pengguna serta batas aplikasi. Data ini umumnya bisa diperoleh lewat wawancara, *survey* ataupun dialog. Pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan (Rosa and Salahuddin, 2018).

2. *Design*

Pada proses desain, dilakukan penerjemahan ketentuan kebutuhan ke suatu perancangan desain perangkat lunak yang bisa diperkirakan sebelum dibuatnya proses pengkodean (*coding*). Proses ini berfokus pada struktur informasi, arsitektur perangkat lunak, representasi interface, serta perinci algoritma prosedural. Desain sistem biasanya dibuat dan digunakan seperti *Flowchart*, *Mind Map*, ERD, dan UML (Rosa and Salahuddin, 2018).

3. *Code*

Pada tahap ini terjadi proses menerjemahkan perancangan desain ke bentuk yang bisa dipahami oleh mesin, dengan memakai kode kode bahasa pemrograman. Kode program yang dihasilkan masih berbentuk modul- modul kecil yang nantinya hendak digabungkan pada tahap selanjutnya (Rosa and Salahuddin, 2018).

4. *Test*

Peguajian ini berfokus pada perangkat lunak secara dari segi logis dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini digunakan



untuk meminimalisir kesalahan atau error dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan (Rosa and Salahuddin, 2018).

### 2.15 *Black Box*

*Black Box* adalah teknik pengujian *software* yang berfokus pada spesifikasi fungsional dari *software*. Kelebihan penggunaan *Black Box* adalah (1) pengujian tidak perlu memiliki pengetahuan tentang bahasa pemrograman tertentu; (2) pengujian dilaksanakan dari *point of view* bagi *user*, untuk membantu melihat *error* atau *bug*. Saat ini terdapat beberapa teknik *Black Box* salah satunya adalah *Boundary Value Analysis* (BVA) adalah teknik dalam *Black Box* yang berfokus pada proses masukan dengan menguji nilai batas atas dan bawah disaat melakukan proses masukan (Snadhika Jaya, 2018).

Dalam menyusun skema pengujian menggunakan skala *Guttman*. Menurut Skala Guttman merupakan skala kumulatif yang digunakan untuk menjawab sebuah pertanyaan yang tegas seperti jawaban “Ya-Tidak”, “Benar-Salah”, “Berhasil-Gagal”. Skala Guttman akan mengukur dan menghitung suatu dimensi dari sebuah variable yang memiliki banyak multi dimensi. Untuk mengukur suatu variabel pada aspek *Functionality* yang akan diteliti maka menggunakan instrumen penelitian skala likert (Sugiyono Prof, 2010). Untuk dapat menilai hasil perhitungan pengujian sistem maka akan menggunakan retang kriteria menurut (Sugiyono Prof, 2010) yaitu :

Tabel 2. 5 Rentang Kriteria

| No | Rentang Kriteria | Kriteria           |
|----|------------------|--------------------|
| 1  | 0% - 20 %        | Sangat Tidak Layak |
| 2  | 21% - 40 %       | Tidak Layak        |
| 3  | 41% - 60 %       | Ragu-Ragu          |
| 4  | 61% - 80 %       | Layak              |
| 5  | 81% - 100 %      | Sangat Layak       |

### 2.16 ISO 25010

ISO/IEC 25010 adalah standar model kualitas dari sebuah sistem dan perangkat lunak yang menggantikan ISO/IEC 9126 tentang *software engineering* (Lamada, Miru and Amalia, 2020). Dikutip dari (P. Miguel, Mauricio and Rodríguez, 2014), Menurut IEE *Standard Glossary of Software Enggunnering Terminology* kualitas produk software didefinisikan sebagai :

1. Se jauh mana suatu sistem, komponen atau proses memenuhi persyaratan yang ditentukan.
2. Se jauh mana suatu sistem, komponen atau proses memenuhi kebutuhan atau harapan pengguna.

Karakteristik ISO/IEC 25010 ini dibagi menjadi beberapa karakteristik antara lain yaitu :

#### 2.16.1 *Functional Suitability*

Karakteristik se jauh mana perangkat lunak mampu menyediakan fungsi yang memenuhi kebutuhan yang dapat digunakan dalam kondisi dan keadaan tertentu . *Functional Suitability* memiliki sub karakteristik yaitu :

1. *Functional completeness*, sejauh mana rangkaian fungsi mencakup semua tugas dan tujuan pengguna yang ditentukan.
2. *Functional Correctness*, sejauh mana produk atau sistem memberikan hasil yang benar dengan tingkat presisi yang dibutuhkan.
3. *Functional Appropriateness*, sejauh mana fungsi memfasilitasi pencapaian tugas dan tujuan tertentu.

#### 2.16.2 *Reliability*

Sejauh mana suatu sistem, produk atau komponen melakukan fungsi tertentu dalam kondisi tertentu untuk jangka waktu tertentu. Karakteristik ini terdiri dari sub karakteristik berikut :

1. *Maturity*, sejauh mana sistem, produk, atau komponen beroperasi dan dapat diakses saat diperlukan untuk digunakan.
2. *Availability*, sejauh mana sistem, produk atau komponen beroperasi dan dapat diakses saat diperlukan untuk digunakan.
3. *Fault Tolerance*, sejauh mana sistem, produk atau komponen beroperasi sebagaimana dimaksud meskipun ada kesalahan perangkat keras atau perangkat lunak.
4. *Recoverability*, sejauh mana dalam hal gangguan atau kegagalan, produk atau sistem dapat memulihkan data yang terpengaruh secara langsung dan membangun kembali keadaan sistem yang diinginkan.

#### 2.16.3 *Performance Efficiency*

Karakteristik ini mewakili kinerja relative terhadap jumlah sumber daya yang akan digunakan dalam kondisi yang dinyatakan. Karakteristik ini terdiri dari sub-karakteristik berikut :

1. *Time Behaviour*, sejauh mana respon dan waktu pemrosesan serta tingkat input suatu produk atau sistem saat menjalankan fungsinya untuk memenuhi persyaratan.
2. *Resource Utilization*, sejauh mana jumlah dan jenis sumber daya yang digunakan oleh produk atau sistem ketika menjalankan fungsinya.
3. *Capability*, sejauh mana batas maksimum parameter produk atau sistem memenuhi persyaratan.

#### 2.16.4 *Usability*

Sejauh mana produk atau sistem dapat digunakan oleh pengguna tertentu untuk mencapai tujuan tertentu dengan efektivitas, efisiensi dan kepuasan dalam penggunaannya. Karakteristik ini terdiri dari sub-karakteristik berikut :

1. *Appropriateness recognizability*, sejauh mana pengguna dapat mengenali apakah suatu produk atau sistem sesuai untuk kebutuhan mereka.
2. *Learnability*, sejauh mana produk atau sistem dapat digunakan oleh pengguna tertentu untuk mencapai tujuan tertentu dengan pembelajaran menggunakan produk atau sistem dengan efektivitas, efisiensi, kebebasan dari risiko yang didapat dan tingkat kepuasan dalam ranah penggunaan tertentu.
3. *Operability*, sejauh mana produk atau sistem memiliki atribut yang membuatnya mudah dioperasikan dan dikendalikan.
4. *User Error Protection*, sejauh mana sistem melindungi pengguna dari membuat kesalahan.

5. *User Interface Aesthetics*, sejauh mana antarmuka pengguna memungkinkan interaksi yang menyenangkan dan memuaskan bagi pengguna.
6. *Accessibility*, sejauh mana produk atau sistem dapat digunakan oleh orang-orang dengan jangkauan karakteristik dan kemampuan terluas untuk mencapai tujuan tertentu dalam ranah penggunaan tertentu.

#### 2.16.5 *Security*

Sejauh mana suatu produk atau sistem melindungi informasi dan data sehingga orang atau produk atau sistem lain memiliki tingkat akses data yang sesuai dengan jenis dan tingkat otorisasi mereka. Karakteristik ini terdiri dari sub-karakteristik berikut :

1. *Confidentiality*, sejauh mana produk atau sistem memastikan bahwa data hanya dapat diakses oleh mereka yang berwenang untuk memiliki akses.
2. *Integrity*, sejauh mana sistem, produk atau komponen mencegah akses tidak sah masuk ke dalam atau memodifikasi program atau *database*.
3. *Non-repudiation*, sejauh mana tindakan dapat dibuktikan telah terjadi sehingga peristiwa atau Tindakan tidak dapat dihindari.
4. *Accountability*, sejauh mana Tindakan suatu entitas dapat dilacak secara unik ke entitas tersebut.
5. *Authenticity*, sejauh mana identitas subjek atau sumber daya dapat dibuktikan sebagai yang diklaim.

#### 2.16.6 *Compatibility*

Sejauh mana suatu produk, sistem atau komponen dapat bertukar informasi dengan produk, sistem atau komponen lain dan menjalankan fungsi yang diperlukan

sambal berbagai lingkungan perangkat keras atau perangkat lunak yang sama.

Berikut karakteristik ini terdiri dari berikut :

1. *Co-existence*, sejauh mana suatu produk dapat melakukan fungsi yang diperlukan secara efisien sambal berbagai lingkungan dan sumber daya yang sama dengan produk lain, tanpa dampak yang merugikan pada produk lain.
2. *Interoperability*, sejauh mana dua atau lebih sistem produk atau komponen dapat bertukar informasi dan menggunakan informasi yang telah dipertukarkan.

#### 2.16.7 *Maintainability*

Karakteristik ini mewakili tingkat efektivitas dan efisiensi dimana produk atau sistem dapat dimodifikasi untuk memperbaikinya atau menyesuaikannya dengan perubahan lingkungan dan persyaratan. Karakteristik ini terdiri dari sub karakteristik yaitu :

1. *Modularity*, sejauh mana sistem atau program komputer terdiri dari komponen-komponen sedemikian rupa sehingga perubahan pada satu komponen memiliki dampak minimal pada komponen lainnya.
2. *Reusability*, sejauh mana asset dapat digunakan di lebih satu sistem.
3. *Analysability*, tingkat efektivitas dan efisiensi yang memungkinkan untuk menilai dampak pada produk atau sistem dari perubahan yang dimaksudkan untuk satu atau lebih bagiannya atau untuk mendeteksi suatu produk untuk kekurangan atau penyebab kegagalan untuk mengidentifikasi bagian untuk dimodifikasi.

4. *Modifiability*, sejauh mana suatu produk atau sistem dapat dimodifikasi secara efektif dan efisien tanpa menimbulkan kekurangan atau menurunkan kualitas produk yang ada.
5. *Testability*, tingkat efektivitas dan efisiensi dengan kriteria pengujian yang dapat ditetapkan untuk sistem produk atau komponen dan pengujian dapat dilakukan untuk menentukan apakah kriteria tersebut telah terpenuhi.

#### 2.16.8 *Portability*

Tingkat efektifitas dan efisiensi dimana sistem produk atau komponen dapat ditransfer dari satu perangkat keras, lunak atau operasional dan penggunaan lainnya. Karakteristik ini terdiri dari sub yaitu :

1. *Adaptability*, sejauh mana produk atau sistem dapat secara efektif dan efisien diadaptasi untuk perangkat.
2. *Installability*, tingkat efisiensi dimana produk atau sistem dapat berhasil dapat berhasil dipasang dan dilepas di lingkungan tersebut.
3. *Replaceability*, sejauh mana suatu produk dapat menggantikan sistem yang sudah ada.