

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terdahulu

Berdasarkan penelitian yang dilakukan oleh Budihartanti (2017) mengenai Sistem Pendukung Keputusan Dalam Penilaian Karyawan Dengan Menerapkan Metode *Simple Additive Weighting*, penelitian Budihartanti membuat aplikasi sistem pendukung keputusan dengan metode *Simple Additive Weighting* untuk penilaian karyawan dan memiliki hasil yaitu karyawan dengan nilai tertinggi dari hasil perhitungan metode *Simple Additive Weightin* yaitu 0.892.

Berdasarkan penelitian yang dilakukan oleh Bintang dan Sinaga (2021) mengenai Sistem Pendukung Keputusan Pengangkatan Jabatan Pimpinan Tinggi Pratama Dengan Metode *Simple Additive Weighting*, penelitian Bintang dan Sinaga membuat aplikasi sistem pendukung keputusan dengan metode *Simple Additive Weighting* untuk pengangkatan jabatan pimpinan dan memiliki hasil yaitu alternatif dengan nilai tertinggi dari hasil perhitungan metode *Simple Additive Weighting*.

Berdasarkan penelitian yang dilakukan oleh Hayati dan Aliyah (2019) mengenai Sistem Pendukung Keputusan Promosi Jabatan Menggunakan Metode *Simple Additive Weighting*, penelitian Hayati dan Aliyah membuat aplikasi sistem pendukung keputusan dengan metode *Simple Additive Weighting* untuk keputusan promosi jabatan dan memiliki hasil yaitu karyawan dengan nilai tertinggi dari hasil perhitungan metode *Simple Additive Weightin* yaitu 92.3.

Berdasarkan penelitian yang dilakukan oleh Kusuma, dkk (2019) mengenai Sistem Pendukung Keputusan Untuk Menentukan Jabatan Menggunakan Metode *Simple Additive Weighting*, penelitian Kusuma, dkk membuat aplikasi sistem pendukung keputusan dengan metode *Simple Additive Weighting* untuk keputusan menentukan jabatan dan memiliki hasil yaitu jika memiliki status nilai akhir perhitungan metode *Simple Additive Weighting* diatas 0.8.

Berdasarkan penelitian yang dilakukan oleh Milani dan Yunita (2019) mengenai Sistem Pendukung Keputusan Penilaian Pegawai Untuk Kenaikan Jabatan Dengan Metode *Simple Additive Weighting*, penelitian Milani dan Yunita membuat aplikasi sistem pendukung keputusan dengan metode *Simple Additive Weighting* untuk keputusan penilaian pegawai untuk kenaikan jabatan dan memiliki hasil yaitu karyawan dengan nilai tertinggi dari hasil perhitungan metode *Simple Additive Weightin* yaitu 0.96.

Berdasarkan penelitian yang dilakukan oleh Rahmatullah dan Hasugian (2020) mengenai Penerapan Metode *Analytical Hierarchy Process* Dan *Simple Additive Weighting* Untuk Pemilihan Karyawan Terbaik Setiap Jabatan Pada PT. Wahana Piranti Teknologi, penelitian Rahmatullah dan Hasugian membuat aplikasi sistem pendukung keputusan dengan metode *Simple Additive Weighting* untuk keputusan penilaian pegawai untuk kenaikan jabatan dan memiliki hasil yaitu karyawan dengan nilai tertinggi dari hasil perhitungan metode *Simple Additive Weightin* yaitu 0.9381.

II.2. Landasan Teori

Landasan teori dikutip dari beberapa jurnal penelitian terdahulu yang berkaitan dengan penelitian ini.

II.2.1. Penerapan

Penerapan (implementasi) adalah bermuara pada aktivitas, aksi, tindakan, atau adanya mekanisme suatu sistem. Implementasi bukan sekedar aktivitas, tetapi suatu kegiatan yang terencana dan untuk mencapai tujuan kegiatan. Penerapan (implementasi) adalah perluasan aktivitas yang saling menyesuaikan proses interaksi antara tujuan dan tindakan untuk mencapainya serta memerlukan jaringan pelaksana, birokrasi yang efektif. Berdasarkan pengertian-pengertian tersebut dapat disimpulkan bahwa kata penerapan (implementasi) bermuara pada aktifitas, adanya aksi, tindakan, atau mekanisme suatu system. Ungkapan mekanisme mengandung arti bahwa penerapan (implementasi) bukan sekedar aktifitas, tetapi suatu kegiatan yang terencana dan dilakukan secara sungguh-sungguh berdasarkan acuan norma tertentu untuk mencapai tujuan kegiatan. (Susilo, dkk, 2019: 94).

Menurut Kamus Besar Bahasa Indonesia (KBBI) Penerapan adalah suatu pembuatan mempraktekkan suatu teori, metode, dan hal lain untuk mencapai tujuan tertentu dan untuk suatu kepentingan yang diinginkan oleh suatu kelompok atau golongan yang telah terencana dan tersusun sebelumnya. (Anshor dan Apandi, 2019: 133).

II.2.2. Metode *Simple Additive Weighting* (SAW)

Metode *Simple Additive Weighting* (SAW) sering juga dikenal istilah metode 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 rating alternatif yang ada. Metode ini mengharuskan pembuat keputusan menentukan bobot bagi setiap atribut atau kriteria.

Langkah metode SAW, sebagai berikut:

1. Memberikan nilai setiap alternatif (A_i) pada setiap kriteria atau atribut (C_j).
2. Memberikan nilai bobot (W)
3. Melakukan normalisasi matriks dengan cara menghitung nilai rating kinerja ternormalisasi (r_{ij}) dari alternatif (A_i) pada atribut (C_j) berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut keuntungan/benefit = Maksimum atau atribut biaya/cost = Minimum). Apabila berupa atribut keuntungan, maka nilai (X_{ij}) dari setiap kolom atribut dibagi dengan nilai Max (X_{ij}) dari setiap kolom, sedangkan jika berupa atribut biaya, maka nilai Min (X_{ij}) dari tiap kolom atribut dibagi dengan nilai (X_{ij}) setiap kolom.
4. Mengalikan matriks ternormalisasi (R) dengan nilai bobot (W).
5. Menentukan nilai preferensi untuk setiap alternatif (V_i) dengan cara menjumlahkan hasil kali antara matriks ternormalisasi (R) dengan bobot (W). Nilai (V_i) yang lebih besar mengidentifikasikan bahwa setiap alternatif A_i lebih terpilih. Formula ternormalisasi disajikan pada persamaan di bawah ini:

$$R_{ij} = \begin{cases} \frac{X_{ij}}{\max X_{ij}} & \text{jika } j \text{ adalah atribut keuntungan (benefit)} \\ \frac{\min X_{ij}}{X_{ij}} & \text{jika } j \text{ adalah atribut biaya (cost)} \end{cases} \dots\dots\dots(1)$$

Keterangan:

- a. Simbol R_{ij} adalah rating kinerja ternormalisasi dari alternatif A_i pada atribut C_j $i = 1, 2, \dots, m$ dan $j = 1, 2, \dots, n$. (m dan n merupakan banyaknya alternatif dan kriteria).
- b. Simbol X_{ij} adalah nilai rating kecocokan pada A_i dan C_j .
- c. Simbol $\max X_{ij}$ adalah nilai terbesar dari semua nilai rating kecocokan pada setiap kriteria.
- d. Simbol $\min X_{ij}$ adalah nilai terkecil dari semua rating kecocokan pada setiap kriteria.
- e. Atribut keuntungan adalah jika nilai terbesar dalam atribut tersebut merupakan nilai terbaik.
- f. Atribut biaya adalah jika nilai terkecil dalam atribut tersebut merupakan nilai terbaik.

Nilai preferensi untuk setiap alternatif (V_i) diberikan sebagai berikut:

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

Keterangan:

- a. Simbol V_i adalah rangking untuk setiap alternatif.
- b. Simbol n adalah banyaknya (jumlah) alternatif.
- c. Simbol W_j adalah nilai bobot dari setiap kriteria.
- d. Simbol R_{ij} adalah nilai rating kinerja ternormalisasi. (Milani dan Yunita, 2019: 32).

II.2.3. Sistem Pendukung Keputusan (SPK)

Sistem Pendukung Keputusan (SPK) merupakan suatu sistem yang menghasilkan suatu alternatif keputusan yang dapat dipergunakan sebagai alat bantu dalam mengambil keputusan. Metode SAW dipilih karena metode ini menentukan nilai bobot untuk setiap atribut, kemudian dilanjutkan dengan proses perangkingan yang akan menyeleksi alternatif terbaik dari sejumlah alternatif. (Budihartanti, 2019: 1).

II.2.3.1. Tujuan Sistem Pendukung Keputusan (SPK)

Tujuan dari Sistem Pendukung Keputusan adalah sebagai berikut:

1. Membantu manajer dalam pengambilan keputusan atas masalah semi terstruktur.
2. Memberikan dukungan atas pertimbangan manajer dan bukannya di maksudkan untuk menggantikan manajer.
3. Meningkatkan efektivitas keputusan yang diambil manajer lebih daripada perbaikan efisiensinya.
4. Kecepatan komputasi. Komputer memungkinkan para pengambil keputusan untuk melakukan banyak komputasi secara cepat dengan biaya yang rendah.
5. Peningkatan produktivitas. Membangun suatu kelompok pengambil keputusan, terutama para pakar, bisa sangat mahal. Pendukung terkomputerisasi bisa mengurangi ukuran kelompok dan memungkinkan para anggotanya untuk berada di berbagai lokasi yang berbeda-beda (menghemat biaya perjalanan). Selain itu, produktivitas staf pendukung (misalnya analisis keuangan dan hukum) bisa di tingkatkan. Produktivitas juga bisa di tingkatkan menggunakan

peralatan optimasi yang menentukan cara terbaik untuk menjalankan sebuah bisnis.

6. Dukungan kualitas. Komputer bisa meningkatkan kualitas keputusan yang di buat. Sebagai contoh, semakin banyak data yang di akses, makin banyak juga alternatif yang bisa di evaluasi. Analisis resiko bisa di lakukan dengan cepat dan pandangan dari para pakar (beberapa dari mereka berada di lokasi yang jauh) bisa dikumpulkan dengan cepat dan dengan biaya yang lebih rendah. Keahlian bahkan bisa di ambil langsung dari sebuah sistem komputer melalui metode kecerdasan tiruan. Dengan komputer, para pengambil keputusan bisa melakukan simulasi yang kompleks, memeriksa banyak scenario yang memungkinkan, dan menilai berbagai pengaruh secara cepat dan ekonomis. Semua kapabilitas tersebut mengarah kepada keputusan yang lebih baik.
7. Berdaya saing. Manajemen dan pemberdayaan sumber daya perusahaan. Tekanan persaingan menyebabkan tugas pengambilan keputusan menjadi sulit. Persaingan di dasarkan tidak hanya pada harga, tetapi juga pada kualitas, kecepatan, kustomasi produk, dan dukungan pelanggan. Organisasi harus mampu secara sering dan cepat mengubah mode operasi, merestrukturisasi ulang proses dan struktur, memberdayakan karyawan, serta berinovasi. Teknologi pengambilan keputusan bisa menciptakan pemberdayaan yang signifikan dengan cara memperbolehkan seseorang untuk membuat keputusan yang baik. secara cepat, bahkan jika mereka memiliki pengetahuan yang kurang.
8. Mengatasi keterbatasan kognitif dalam pemrosesan dan penyimpanan. Otak manusia memiliki kemampuan yang terbatas untuk memproses dan menyimpan

informasi. Orang-orang kadang sulit mengingat dan menggunakan sebuah informasi dengan cara yang bebas dari kesalahan. (Apriani, 2019: 12).

II.2.3.2. Karakteristik Sistem Pendukung Keputusan (SPK)

Sistem pendukung keputusan dirancang secara khusus untuk mendukung seseorang dalam mengambil keputusan tertentu. Menurut Ada beberapa karakteristik sistem pendukung keputusan, yaitu:

1. Interaktif dapat melakukan akses secara cepat ke data dan memperoleh informasi yang dibutuhkan. Fleksibel Sistem Pendukung Keputusan (SPK) memiliki sebanyak mungkin variable masukan, kemampuan untuk mengolah dan memberikan keluaran yang menyajikan alternatif-alternatif keputusan kepada pemakai.
2. Data kualitas Sistem Pendukung Keputusan (SPK) memiliki kemampuan menerima data kualitas yang dikuantitaskan yang sifatnya subyektif dari pemakainya sebagai data masukan untuk pengolahan data. Misalnya: penilaian terhadap kecantikan yang bersifat kualitas, dapat dikuantitaskan dengan pemberian bobot nilai seperti 75 atau 90.
3. Prosedur Pakar Sistem Pendukung Keputusan (SPK) mengandung suatu prosedur yang dirancang berdasarkan rumusan formal atau juga beberapa prosedur kepakaran seseorang atau kelompok dalam menyelesaikan suatu bidang masalah dengan fenomena tertentu. (Apriani, 2019: 12).

II.2.3.3. Komponen Sistem Pendukung Keputusan (SPK)

Aplikasi sistem pendukung keputusan bisa terdiri dari beberapa subsistem, yaitu:

1. Subsistem Manajemen Data Subsistem manajemen data memasukkan satu *database* yang berisi data yang relevan untuk suatu situasi dan dikelola oleh perangkat lunak yang disebut sistem manajemen *database* DBMS (*Data Base Management System*). Subsistem manajemen data bisa diinterkoneksi dengan data *warehouse* perusahaan, suatu repositori untuk data perusahaan yang relevan dengan pengambilan keputusan.
2. Subsistem Manajemen Model Merupakan paket perangkat lunak yang memasukan model keuangan, statistik, ilmu manajemen, atau model kuantitatif lain yang memberikan kapabilitas analitik dan manajemen perangkat lunak yang tepat. Bahasa-bahasa pemodelan untuk membangun model-model kustom juga dimasukkan. Perangkat lunak itu sering disebut sistem manajemen basis model (MBMS). Komponen tersebut bisa dikoneksikan ke penyimpanan korporat atau eksternal yang ada pada model.
3. Subsistem Antarmuka Pengguna Pengguna berkomunikasi dengan dan memerintahkan sistem pendukung keputusan melalui subsistem tersebut. Pengguna adalah bagian yang dipertimbangkan dari sistem. Para peneliti menegaskan bahwa beberapa kontribusi unik dari sistem pendukung keputusan berasal dari interaksi yang intensif antara komputer dan pembuat keputusan.
4. Subsistem Manajemen Berbasis-Pengetahuan Subsistem tersebut mendukung semua subsistem lain atau bertindak langsung sebagai suatu komponen independen dan bersifat opsional. Selain memberikan intelegensi untuk memperbesar pengetahuan si pengambil keputusan, subsistem tersebut bisa diinterkoneksi dengan *repository* pengetahuan perusahaan (bagian dari

sistem manajemen pengetahuan), yang kadang-kadang disebut basis pengetahuan organisasional. (Apriani, 2019: 12).

II.2.3.4. Manfaat Sistem Pendukung Keputusan (SPK)

Sistem Pendukung Keputusan (SPK) dapat memberikan berbagai manfaat dan keuntungan. Manfaat yang dapat diambil dari SPK adalah:

1. SPK memperluas kemampuan pengambil keputusan dalam memproses data atau informasi bagi pemakainya.
2. SPK membantu pengambilan keputusan lebih cepat serta hasilnya dapat diandalkan.
3. SPK dapat menghasilkan solusi dengan keputusan, dan juga mampu menyajikan berbagai alternatif pemecahan masalah.

II.2.3.5. Tahapan Pengambilan Keputusan

Tahapan rancang bangun SPK terdiri dari:

1. Identifikasi tujuan rancang bangun, yang bertujuan untuk menentukan arah dan sasaran yang hendak dicapai dalam pembuatan suatu sistem pendukung keputusan.
2. Perancangan pendahuluan untuk merumuskan kerangka dan ruang lingkup sistem pendukung keputusan serta persyaratan tujuan yang mesti dipenuhinya, memilih konsep-konsep, menganalisis dan mengaplikasi model pembuatan keputusan yang relevan dengan tujuan sistem yang akan dibangun. Perancangan sistem, yang diawali dengan analisis sistem untuk merumuskan spesifikasi sistem pendukung keputusan dilanjutkan dengan perancangan

konfigurasi sistem, beserta perangkat keras dan perangkat lunak pendukungnya. (Apriani, 2019: 13).

II.2.4. Promosi

Promosi adalah suatu aktivitas komunikasi yang dilakukan oleh seseorang atau suatu perusahaan dengan masyarakat luas, dimana tujuannya adalah untuk memperkenalkan sesuatu (barang/ jasa/ merek/ perusahaan) kepada masyarakat dan sekaligus mempengaruhi masyarakat luas agar membeli dan menggunakan produk tersebut. Promosi adalah kegiatan yang dilakukan oleh perusahaan yang menonjolkan keistimewaan-keistimewaan produknya yang membujuk konsumen sasaran agar membeli. Mengacu pada pengertian promosi di atas tujuan promosi pada umumnya secara keseluruhan melakukan promosi adalah agar meningkatkan angka penjualan dan keuntungan usaha. (Syahputra, 2019: 84).

II.2.5. Kenaikan Jabatan

Kenaikan jabatan adalah penghargaan yang diberikan atas prestasi kerja dan loyalitas karyawan terhadap perusahaan. Karyawan merupakan sumber daya yang sangat penting untuk menentukan keberhasilan suatu satuan kerja. Upaya meningkatkan kualitas kinerja seorang karyawan yaitu adanya kenaikan jabatan karyawan kontrak menjadi karyawan tetap. (Anshor dan Apandi, 2019: 131).

II.2.6. *Hyper Text Markup Language (HTML)*

HTML adalah "bahasa" untuk membuat dokumen yang bisa diakses lewat *web*. HTML memuat dua bagian yakni isi dan pemformat. Isi adalah materi yang akan disampaikan, sedangkan bagian pemformat berwujud "*tag*" untuk menjadikan isi bisa diterjemahkan sebagai paragraf, ganti baris, *heading*, *unordered list*, *ordered list*, *table*. Dokumen HTML merupakan dokumen teks biasa yang bisa dibuka dengan pengedit teks seperti Notepad. Bila dibuka dengan Notepad akan terlihat bahwa yang tersaji di *web browser* sebenarnya sudah ada di sana. Tetapi di Notepad terlihat ada tambahan di sana-sini berupa "*tag*" sehingga di *browser* terlihat seperti itu. Anatomi dokumen HTML terdiri dari dua bagian, yakni HEAD dan BODY. Bagian HEAD biasanya memuat informasi tentang atribut dari dokumen itu. Sedangkan bagian BODY memuat informasi utama yang akan dikomunikasikan lewat *web*. Adanya link dalam dokumen HTML menjadikan dokumen ini bisa menunjuk ke dokumen-dokumen HTML lain baik di *folder* yang sama, di *server* yang sama, maupun di *server* lain. Inilah alasan mengapa dokumen ini disebut hiperteks. Satu dokumen HTML juga bisa "memuat" dan menampilkan *file-file* lain seperti gambar, audio, Word, PDF, dan sebagainya. (Huzaeni, dkk, 2019: 140).

Hypertext Markup Language atau HTML adalah bahasa yang digunakan pada dokumen *web*. Dengan kata lain, HTML merupakan bahasa pemrograman berupa *tag-tag* yang digunakan untuk membuat dan mengatur penampilan dari halaman *website*. (Irmaniawan, 2020: 44).

II.2.7. *Hypertext Preprocessor* (PHP)

PHP singkatan dari *Hypertext Preprocessor* merupakan bahasa pemrograman yang skript bersifat *Open Source*. Program ini bersifat *server side*, artinya tanpa adanya server yang berjalan disisinya script program PHP tidak dapat dijalankan PHP adalah *script* yang ditanamkan dalam HTML untuk membuat halaman *website* dinamis yang berkerja secara otomatis dan berfungsi sebagai pengolahan data pada *server* di mana *script* tersebut di jalankan. (Huzaeni, 2019: 140).

PHP adalah bahasa standar yang digunakan dalam *website*. PHP adalah bahasa program yang berbentuk skrip yang diletakan di dalam *server web*. (Irawan, 2020: 46).

II.2.8. Normalisasi

Normalisasi merupakan salah satu teknik dalam desain basis data untuk memproduksi tabel-tabel yang memiliki atribut yang tidak tepat sesuai dengan kebutuhan sistem. Ciri-ciri tabel yang ternormalisasi adalah:

1. Jumlah atribut bersifat minimal, sesuai dengan kebutuhan data.
2. Atribut yang berhubungan dekat (disebut *functional dependency*) ditempatkan dalam tabel yang sama.
3. Redundansi (kemunculan duplikasi data yang tidak diharapkan) bersifat minimal dengan memanfaatkan *foreign key*.

Manfaat yang diperoleh dari hasil proses normalisasi ini adalah basis data menjadi mudah diakses, data mudah dikelola, dan meminimalkan tempat penyimpanannya. Manfaat normalisasi adalah sebagai berikut:

1. Normalisasi memastikan bahwa setiap kolom milik tabel yang tepat yang mana telah ditetapkan dan bukan tabel lain.
2. Normalisasi menghilangkan penyimpanan informasi yang redundan, ini menyederhanakan logika aplikasi, karena developer basisdata tidak perlu memikirkan mengenai beberapa salinan bagian informasi yang sama.
3. Normalisasi memastikan bahwa basisdata memiliki satu kolom di satu tempat, dengan satu nama, dengan satu nilai, pada satu waktu. Proses normalisasi disebutkan terdapat beberapa tahapan dalam normalisasi dimulai dari tahap paling ringan (1NF) hingga paling ketat (5NF). Biasanya hanya sampai pada tingkat 3NF atau BCNF karena sudah cukup memadai untuk menghasilkan tabel-tabel yang berkualitas baik. (Sujatmoko, dkk, 125).

II.2.9. Database

Teknik desain *database* merupakan cara dalam membuat rancangan *database*. Pada saat membuat *database*, terdapat beberapa hal yang perlu diperhatikan untuk menunjang keberhasilan dan keefektifan suatu *database*. Keberhasilan dan keefektifan suatu database dapat dilihat berdasarkan kemudahan dalam pengolahan data yang baik dan terstruktur, serta sinkronisasinya terhadap jenis *database* yang akan digunakan. *database* sendiri dapat diartikan sebagai kumpulan dari sebuah fakta dalam bentuk data yang disimpan secara terstruktur dan dirancang sesuai dengan kebutuhan dari *database* tersebut. Desain *database* merupakan suatu proses untuk menentukan rincian dari isi dan pengaturan data. (Wibawa, dkk, 2021: 191).

II.2.10. *My Structure Query Language (MySQL)*

MySQL (*My Structured Query Language*) atau yang biasa dibaca mal-sekuel adalah sebuah program pembuat dan pengelola *database* atau yang sering disebut dengan DBMS (*Database Management System*), sifat dari DBMS ini adalah open source. (Imaniawan, 2020: 45).

MYSQL merupakan salah satu *database* yang bersifat *open source* yang populer di dunia, dimana saat ini digunakan lebih dari 100 juta pengguna diseluruh dunia MYSQL turunan dari SQL (*Structured Query Language*) yang sering digunakan oleh para *programmer* dalam pembuatan program. Sistem *database* MYSQL selain mudah memahami sintaks dan pengaksesan juga di dukung beberapa fitur seperti *Multithreaded*, *multi-user* dan SQL *database* managemen sistem. (Huzaeni, dkk, 2019: 141).

II.2.11. *Unified Modeling Language(UML)*

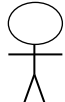
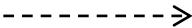
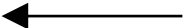
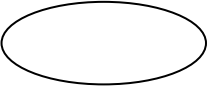
Unified Modeling Language (UML) adalah sebuah bahasa untuk menentukan, visualisasi, kontruksi, dan mendokumentasikan *artifact* (bagian dari informasi yang digunakan atau dihasilkan dalam suatu proses pembuatan perangkat lunak).

1. *Use Case Diagram*

Use Case Diagram merupakan pemodelan untuk melakukan (*behavior*) sistem informai yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Berikut adalah simbol-simbol yang ada pada

diagram *use case*. *Use case diagram* dapat digambarkan dengan sumber-sumber pada Tabel II.1.

Tabel II.1. Simbol Use Case

Notasi	Keterangan	Simbol
<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .	
<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).	
<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).	
<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.	
<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.	
<i>Association</i>	Apa yang mnghubungkan antara objek satu dengan objek lainnya.	
<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.	
<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .	
<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemennya (<i>sinergi</i>).	
<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi	

(Sumber: Pitrawati dan Sanjaya, 2021: 157)

2. Diagram Aktivitas (*Activity Diagram*)

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Perlu diperhatikan bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Berikut adalah simbol-simbol yang ada pada diagram aktivitas. *Activity diagram* dapat digambarkan dengan simbol-simbol seperti pada tabel II.2.

Tabel II.2. Simbol *Activity Diagram*

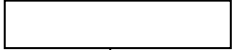
Notasi	Keterangan	Simbol
<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.	
<i>Action</i>	<i>State</i> dari sistem yang mencerminkan eksekusi dari suatu aksi.	
<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.	
<i>Activity Final</i>	Bagaimana objek dibentuk dan dihancurkan	
<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran	

(Sumber: Pitrawati dan Sanjaya, 2021: 157)

3. Diagram Urutan (*Sequence Diagram*)

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence Diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.3.

Tabel II.3. Simbol *Sequence Diagram*

Nama	Keterangan	Gambar
<i>Lifeline</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.	
<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.	
<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.	

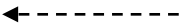
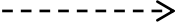
(Sumber: Pitrawati dan Sanjaya, 2021: 157)

4. *Class Diagram* (Diagram Kelas)

Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Berikut adalah simbol-simbol yang ada pada diagram Kelas. *Class diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.4.

Tabel II.4. *Class Diagram*

Notasi	Keterangan	Gambar
<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).	
<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.	
<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.	

<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.	
<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.	
<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri	
<i>Assocation</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya	

(Sumber: Pitrawati dan Sanjaya, 2021: 157)