

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Adapun penelitian terkait ini, peneliti akan membandingkan hasil penelitiannya dengan lima jurnal yaitu :

Berdasarkan penelitian oleh Ayu Octavia (2019) dengan judul penelitian "Sistem Pendukung Keputusan Untuk Menentukan Mutasi Karyawan dengan Menggunakan Metode Oreste (Studi Kasus: PDAM Tirta Deli Kab. Deli Serdang)" Mutasi karyawan ini dilakukan sehingga dapat tercapainya tujuan perusahaan yang optimal untuk menciptakan karyawan yang memiliki profesionalisme yang tinggi, sehingga memiliki daya adaptasi dan antisipasi yang fleksibel terhadap segala macam bentuk dan sifat perubahan yang sedang maupun akan terjadi.

Berdasarkan penelitian oleh Agustian Sinaga, dkk, (2018) dengan judul penelitian "Pemilihan Toko Handphone Terbaik Di Kota Pematangsiantar Menggunakan Metode Oreste" Tujuan Penelitian ini dilakukan untuk mengetahui toko *handphone* terbaik di kota Pematang siantar menurut konsumen yang diberikan kepada konsumen melalui *Kuesioner*. Penelitian ini menggunakan Sistem Pendukung Keputusan dengan metode Oreste. Tahap-tahap yang dilakukan dalam memilih toko *handphone* dimulai dengan menganalisa sistem yang sedang berjalan pada toko *handphone*, kemudian dari hasil analisa dilakukan pembobotan kriteria pada toko *handphone*, lalu dilakukan pembobotan alternatif pada toko *handphone*. Setelah itu dilakukan proses hasil peringkat bobot alternatif toko *handphone*.

Berdasarkan penelitian oleh Saifur Rohmad, dkk (2021) dengan judul “Sistem Pendukung Keputusan Perpanjangan Kontrak Kerja Karyawan Pada PT. Telkom Akses Reg IV Menggunakan Metode Oreste”. Dalam penentuan kontrak kerja karyawan juga sering muncul subjektifitas dari manager ataupun rekan kerja yang dapat mempengaruhi proses pengambilan keputusan tentang kontrak kerja karyawan. Kriteria yang menjadi bahan penilaian kontrak kerja karyawan antara lain appraisal, absensi, BPJS ketenagakerjaan, dan posisi karyawan. Hasil dari penelitian ini memperoleh nilai 0,90 pada validasi perhitungan Koefisien Korelasi Spearman. Sehingga sistem dikatakan layak untuk digunakan dalam penentuan kontrak kerja karyawan PT. Telkom Akses REG IV.

Berdasarkan penelitian oleh Mardiyah Lubis (2020) dengan judul penelitian “Sistem Pendukung Keputusan Pemilihan Kader Kesehatan Puskesmas Mandala Kecamatan Medan Tembung dengan Menggunakan Metode Oreste”. Puskesmas adalah suatu bagian terpenting dari pusat pelayanan kesehatan bagi masyarakat. Dimana masyarakat banyak yang mengantri untuk berobat di puskesmas, sehingga pihak puskesmas kesulitan untuk menanganin masyarakat yang mengantri untuk berobat. Maka puskesmas mengusulkan kepada masyarakat menjadi seorang kader dari puskesmas untuk menanganin pelayanan kesehatan dari masyarakat khususnya untuk sekitar puskesmas. Namun untuk pemilihan kader tersebut yang awalnya terbuka untuk umum, namun pemilihan kadernya tersebut dipilih diketahui oleh masyarakat yang disekitar puskesmas tersebut, hanya dilakukan sepihak saja atau orang terdekat, sehingga kecurangan dalam pemilihan tersebut sering terjadi. Maka solusi dalam pemilihan kader tersebut diperlukanlah suatu Sistem pendukung

keputusan dan menerapkan metode Oreste adalah metode pengembangan dari MADM yang berdasarkan suatu perangkingan dari setiap kriteria dan alternatif.

Sistem yang dibangun dapat membantu Pihak Puskesmas dalam mengambil suatu keputusan dalam pemilihan kader kesehatan puskesmas Mandala Kecamatan Medan Tembung yang tepat, efesiensi dan objektif. Dan hasil proses perangkingan Oreste dari setiap kriteria dan alternatif. Serta unitnya dapat mengadopsi *Besson Rank*, *Besson Rank* itu sendiri adalah pendekatan rata-rata dimana pada setiap kriteria yang sama.

Berdasarkan penelitian dari Arswendi Perdana, dkk (2022) dengan judul “Sistem Pendukung Keputusan Seleksi Penerima Beasiswa Berprestasi Pada Yayasan Pendidikan Jaya Untuk Anak Karyawan Dengan Menerapkan Metode Oreste dan ROC (*Rank Order Centroid*)” Untuk mengantisipasi agar tidak terjadinya kesalahan dalam pemberian beasiswa maka dibutuhkan sebuah Sistem Pendukung Keputusan. Dengan menerapkan metode Oreste dan ROC (*Rank Order Centroid*), dan dengan penjabaran latar belakang tersebut maka penulis tertarik untuk melakukan penelitian terhadap pengambil keputusan dalam penilaian kinerja staf tata usaha yang dimana masing-masing kriteria dan alternatif memiliki faktor nilai bobot yang pada akhir proses menghasilkan perangkingan.

Berdasarkan penelitian dari Alwendi dan Dasril Aldo (2020) dengan judul “Sistem Pendukung Keputusan Pemilihan Toko *Handphone* Terbaik Di Kota Padang sidimpuan Menggunakan Metode Oreste” Penelitian ini menggunakan Sistem Pendukung Keputusan dengan metode Oreste. Metode Oreste merupakan salah satu metode Sistem Pendukung Keputusan yang mengadopsi *Besson Rank*,

yakni pendekatan skala prioritas dari setiap indicator kriteria. Data yang digunakan sebagai alternatif adalah data toko *handphone* di Padang disimpulkan dengan sampel 5 toko dengan 5 kriteria penilaian, yaitu Harga, Kualitas pada Handphone, Pelayanan, Daya Tarik Konsumen dan Lokasi Toko. Tahap-tahap yang dilakukan dalam memilih toko *handphone* dimulai dengan menganalisa sistem yang sedang berjalan pada toko *handphone*, kemudian dari hasil analisa dilakukan pembobotan kriteria pada toko *handphone*, lalu dilakukan pembobotan alternatif pada toko *handphone*.

Sedangkan penelitian yang dilakukan peneliti menghasilkan Hasil penelitian ini diharapkan dapat mempermudah SMK Swasta PAB I Medan dalam menentukan siswa yang mendapatkan beasiswa dan Hasil penelitian di harapkan jadi bahan acuan dan referensi untuk penelitian selanjutnya dalam penerapan metode oreste dalam suatu sistem pendukung keputusan.

II.2. Landasan Teori

II.2.1 Sistem

Definisi sistem adalah “kumpulan dari bagian-bagian yang bekerja sama untuk mencapai tujuan yang sama.” Definisi sistem adalah “sekumpulan objek-objek yang saling berelasi dan berinteraksi serta hubungan antar objek bisa dilihat sebagai suatu kesatuan yang dirancang untuk mencapai suatu tujuan. Sistem adalah penggabungan dari bagian-bagian atau komponen-komponen yang terpisah-pisah dan disatukan menjadi satu rangkaian dan menjadi suatu fungsi yang baru (Aris, 2017)

Kata sistem berasal dari bahasa Latin (*systema*) dan bahasa Yunani (*sustema*) yang artinya adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi. Secara umum sistem adalah kumpulan dari beberapa bagian tertentu yang saling berhubungan secara harmonis untuk mencapai suatu tujuan tertentu. Elemen-elemen yang mewakili suatu sistem secara umum adalah masukan (*Input*) pengolahan (*Processing*) dan keluaran (*Output*). (Sutopo et al., 2016)

II.2.1.1. Karakteristik Sistem

Untuk memahami atau mengembangkan suatu sistem, maka perlu membedakan unsur-unsur dari sistem yang membentuknya. Berikut adalah karakteristik sistem yang dapat membedakan suatu sistem dengan sistem lainnya yaitu:

1. Batasan (*Boundary*) adalah Penggambaran dari suatu elemen atau unsur mana yang termasuk di dalam sistem dan mana yang di luar sistem.
2. Lingkungan (*Environment*) adalah Segala sesuatu di luar sistem, lingkungan yang menyediakan asumsi, kendala, dan input terhadap suatu sistem.
3. Masukan (*Input*) adalah Sumber daya (data, bahan baku, peralatan, energi) dari lingkungan yang dikonsumsi dan dimanipulasi oleh suatu sistem.
4. Keluaran (*Output*) adalah Sumber daya atau produk (informasi, laporan, dokumen, tampilan layer Komputer, barang jadi) yang disediakan untuk lingkungan sistem oleh kegiatan dalam suatu sistem.

5. Komponen (*Component*) adalah Kegiatan-kegiatan atau proses dalam sistem yang mentransformasikan input menjadi bentuk setengah jadi (*output*). Komponen ini bisa merupakan subsistem dari sebuah sistem.
6. Penghubung (*Interface*) adalah Tempat di mana komponen atau sistem dan lingkungannya bertemu atau berinteraksi.
7. Penyimpanan (*Storage*) adalah Area yang dikuasai dan digunakan untuk penyimpanan sementara dan tetap dari informasi, energi, bahan baku, dan sebagainya. Penyimpanan merupakan suatu media penyangga di antara komponen tersebut bekerja dengan berbagai tingkatan yang ada dan memungkinkan komponen yang berbeda dari berbagai data yang sama (Aris, 2017)

II.2.2 Sistem Pendukung Keputusan

Sistem pendukung keputusan adalah sistem informasi berbasis komputer yang interaktif, dengan cara mengolah data dengan berbagai model untuk memecahkan masalah-masalah yang tidak terstruktur sehingga dapat memberikan informasi yang bisa digunakan oleh para pengambil keputusan dalam membuat sebuah keputusan. Dalam sebuah sistem pendukung keputusan, sumber daya intelektual yang dimiliki seseorang dipadukan dengan kemampuan komputer untuk membantu meningkatkan kualitas dari keputusan yang diambil. Pengambilan keputusan merupakan sebuah proses memilih sebuah tindakan diantara beberapa alternatif yang ada, sehingga tujuan yang diinginkan dapat tercapai (Omni Alfina: 2019)

Sistem Pendukung Keputusan adalah konsep spesifik sistem yang menghubungkan komputerisasi informasi dengan para pengambil keputusan sebagai pemakainya. Dengan demikian Sistem Pendukung Keputusan adalah seperangkat elemen yang digabungkan satu dengan yang lainnya saling bekerja sama untuk menghasilkan satu kesatuan di dalam pencapaian suatu tujuan bersama. Sistem pendukung keputusan adalah suatu sistem berbasis komputer yang menghasilkan berbagai alternatif keputusan untuk membantu manajemen dalam menangani berbagai permasalahan yang terstruktur ataupun tidak terstruktur dengan menggunakan data dan model. Untuk menghasilkan keputusan yang baik di dalam sistem pendukung keputusan, perlu didukung oleh informasi dan fakta-fakta yang berkualitas antara lain :

1. Aksebilitas

Atribut ini berkaitan dengan kemudahan mendapatkan informasi, informasi akan lebih berarti bagi si pemakai kalau informasi tersebut mudah didapat, karena akan berkaitan dengan aktifitas dari nilai informasinya.

2. Kelengkapan

Atribut ini berkaitan dengan kelengkapan isi informasi, dalam hal ini isi tidak menyangkut hanya volume tetapi juga kesesuaian dengan harapan si pemakai sehingga sering kali kelengkapan ini sulit diukur secara kuantitatif

3. Ketelitian

Atribut ini berkaitan dengan tingkat kesalahan yang mungkin di dalam pelaksanaan pengolahan data dalam jumlah (volume) besar. Dua tipe kesalahan yang sering terjadi yaitu berkaitan dengan perhitungan.

4. Ketepatan

Atribut ini berkaitan dengan kesesuaian antara informasi yang dihasilkan dengan kebutuhan pemakai. Sama halnya dengan kelengkapan, ketepatan sangat sulit diukur secara kuantitatif.

5. Ketepatan Waktu

Kualitas informasi juga sangat ditentukan oleh ketepatan waktu penyampaian dan aktualisasinya. Misal informasi yang berkaitan dengan perencanaan harian akan sangat berguna kalau disampaikan setiap dua hari sekali.

6. Kejelasan

Atribut ini berkaitan dengan bentuk atau format penyampaian informasi. Bagi seorang pimpinan, informasi yang disajikan dalam bentuk grafik, histogram, atau gambar biasanya akan lebih berarti dibandingkan dengan informasi dalam bentuk kata-kata yang panjang.

7. Fleksibilitas

Atribut ini berkaitan dengan tingkat adaptasi dari informasi yang dihasilkan terhadap kebutuhan berbagai keputusan yang akan diambil dan terhadap sekelompok pengambil keputusan yang berbeda. (Arief Budiman, dkk : 2020)

II.2.3. Tujuan Sistem Pendukung Keputusan

Sistem Pendukung Keputusan tidak dimaksudkan untuk pengambilan keputusan, tetapi memberikan perangkat interaktif yang memungkinkan pengambil keputusan untuk melakukan berbagai analisis menggunakan model-model yang

tersedia. Menurut Arief Budiman, dkk (2020) Tujuan dari sistem pendukung keputusan adalah:

1. Membantu manajer dalam pengambilan keputusan atas masalah semi terstruktur.
2. Memberikan dukungan atas pertimbangan manajer dan bukannya dimaksudkan untuk menggantikan fungsi manajer
3. Meningkatkan efektifitas keputusan yang diambil manajer lebih daripada perbaikan efesiensinya
4. Komputer memungkinkan para pengambil keputusan untuk melakukan banyak komputasi secara cepat dengan biaya yang rendah
5. Peningkatan produktifitas
6. Komputer bisa meningkatkan kualitas keputusan yang dibuat.
7. Berdaya saing
8. Mengatasi keterbatasan kognitif dalam pemrosesan dan penyimpanan.

II.2.4 Metode Oreste

Metode Oreste adalah salah satu metode pengambilan keputusan multi kriteria atau yang lebih dikenal dengan istilah Multi Criteria Decision Making (MCDM) digunakan untuk menyelesaikan permasalahan dengan kriteria yang bertentangan dan tidak sepadan (Serafim Opricopic 2006). Metode ini berfokus pada peringkat dan pemilihan dari sekumpulan alternatif kriteria yang saling bertentangan untuk mengambil keputusan untuk mencapai keputusan akhir.

Oreste memungkinkan untuk peringkat percobaan dalam perintah lengkap atau dalam urutan parsial dengan mempertimbangkan *incomparability*. Metode Oreste merupakan metode dalam sistem pendukung keputusan yang mampu mengolah data ordinal atau data yang berbentuk peringkat yang sulit diolah dengan metode lain. Metode Oreste sudah di implementasikan ke dalam aplikasi Sanna yang dapat menghasilkan peringkat dari beberapa alternatif dari yang terbaik sampai yang terjelek. Tujuan dari metode ini adalah untuk menemukan struktur preferensi global seperangkat alternatif A, yang mencerminkan evaluasi alternatif pada setiap kriteria dan preferensi antara kriteria. Oreste hanya memperhitungkan peringkat alternatif dan kriteria, yang sangat cocok untuk dipergunakan memecahkan masalah yang berhubungan dengan data ordinal. Metode ini tidak memerlukan kuantifikasi bobot kriteria ataupun alternatif nilai kinerja, hanya peringkat urut mereka saja. Salah satu proses dalam metode Oreste adalah Besson-rank yaitu proses pemberian *ranking* untuk sejumlah kriteria atau alternatif berdasarkan tingkat kepentingannya.

Langkah-langkah perhitungan dengan metode Oreste adalah sebagai berikut:

1. Jadikan Dalam Bentuk Ordinal (*Besson – Rank*) Jika ada nilai yang sama, maka rangkingnya dicari rata-ratanya. Dari hasil tersebut, urutkan dari besar kekecil. Rangking nilai alternatif dari kriteria terbesar diberi nilai 1, dan untuk nilai selanjutnya di urutkan berdasarkan nilai yang menjadi urutan selanjutnya.

2. Mencari *Distance Score*

Mencari *Distance Score* dengan cara menghitung setiap pasangan alternatif-kriteria sebagai nilai "jarak" untuk posisi yang ideal dan ditempati oleh alternatif terbaik untuk kriteria yang paling penting menggunakan rumus:

$$D(a, c_j) = [\frac{1}{2} rc_j R + \frac{1}{2} rc_j(a) R]^{1/R}$$

Keterangan:

$D(c_j, a) = \text{Distance Score}.$

$rc_j = \text{Besson} - \text{rank kriteria } j.$

$rc_j(a) = \text{Besson} - \text{rank alternatif dalam kriteria } j.$

$R = \text{Koefisien (default = 3)}.$

3. Buatlah Hasil *Distance Rank* menjadi *Global Rank* Yaitu dengan mengurutkan hasil dari *Distance Rank* dalam bentuk *Ascending* (kecil ke besar).
4. Penjumlahan *Global Rank* Jumlahkan semua alternatif dalam kriteria dalam satu baris pada setiap kolom (yang sering dipanggil *Summary*).
5. Didapatkanlah hasil akhir (*Ascending*) Hasil *Summary* pada *Global Rank* hasilnya di urutkan. Data dengan nilai *Summary* terkecil merupakan data prioritas utama (peringkat pertama). (Ayu Octavia : 2019)

II.2.5. Website

Website merupakan sebuah media informasi yang ada di internet. Website tidak hanya dapat digunakan untuk penyebaran informasi saja melainkan bisa digunakan untuk membuat toko *online*. *Website* adalah kumpulan dari halaman-

halaman situs, yang biasanya terangkum dalam sebuah domain atau subdomain, yang tempatnya berada di dalam *World Wide Web* (WWW) di *Internet*. Sebuah halaman web adalah dokumen yang ditulis dalam format HTML (*Hyper Text Markup Language*), yang hampir selalu bisa diakses melalui HTTP, yaitu protokol yang menyampaikan informasi dari *server website* untuk ditampilkan kepada para pemakai melalui *web browser*. Semua publikasi dari *website-website* tersebut dapat membentuk sebuah jaringan informasi yang sangat besar (Yunita Trimarsiah : 2017).

II.2.6. PHP

PHP adalah bahasa pelengkap HTML yang memungkinkan dibuatnya aplikasi dinamis yang memungkinkan adanya pengolahan data dan pemrosesan data. Semua *sintax* yang diberikan akan sepenuhnya dijalankan pada *server* sedangkan yang dikirimkan ke *browser* hanya hasilnya saja. Kemudian merupakan bahasa berbentuk *script* yang ditempatkan dalam *server* dan diproses di *server*. Hasilnya akan dikirimkan ke *client*, tempat pemakai menggunakan *browser*. PHP dikenal sebagai sebuah bahasa *scripting*, yang menyatu dengan *tag-tag HTML*, dieksekusi di *server*, dan digunakan untuk membuat halaman web yang dinamis seperti halnya *Active Server Pages* (ASP) atau *Java Server Pages* (JSP). PHP merupakan sebuah *software Open Source* (Reza Hermiati : 2021).

II.2.7 MySQL

MySQL merupakan suatu jenis *database server* yang sangat terkenal. MySQL termasuk jenis RDBMS (*Relational Database Manajement System*). MySQL mendukung bahasa pemrograman PHP, bahasa permintaan yang terstruktur, karena pada penggunaannya SQL memiliki beberapa aturan yang telah distandarkan oleh asosiasi yang bernama ANSI. MySQL merupakan RDBMS (*Relational Database Management System*) server. RDBMS adalah program yang memungkinkan pengguna database untuk membuat, mengelola, dan menggunakan data pada suatu model relational. Dengan demikian, tabel-tabel yang ada pada database memiliki relasi antara satu tabel dengan tabel lainnya. Beberapa keunggulan dari MySQL yaitu :

- a. Cepat, handal dan mudah dalam penggunaannya. MySQL lebih cepat tiga sampai empat kali dari pada *database server* komersial yang beredar saat ini, mudah diatur dan tidak memerlukan seseorang yang ahli untuk mengatur administrasi pemasangan MySQL.
- b. Didukung oleh berbagai bahasa *Database Server* MySQL dapat memberikan pesan *Error* dalam berbagai bahasa seperti Belanda, Portugis, Spanyol, Inggris, Perancis, Jerman, dan Italia.
- c. Mampu membuat tabel berukuran sangat besar. Ukuran maksimal dari setiap tabel yang dapat dibuat dengan MySQL adalah 4 GB sampai dengan ukuran *file* yang dapat ditangani oleh sistem operasi yang dipakai.
- d. Lebih murah MySQL bersifat *open source* dan didistribusikan dengan gratis tanpa biaya untuk UNIX *platform*, OS/2 dan *Windows Platform*. Melekatnya

integrasi PHP dengan MySQL. Keterikatan antara PHP dengan MySQL yang sama-sama *Software Open-Source* sangat kuat, sehingga koneksi yang terjadi lebih cepat jika dibandingkan dengan menggunakan *database server* lainnya. Modul MySQL di PHP telah dibuat *Built-in* sehingga tidak memerlukan konfigurasi tambahan pada *File* konfigurasi *Php ini* (Reza Hermiati : 2021).

II.2.8 UML (*Unified Modelling Language*)

Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. (Ade Hendini : 2016).

Unified Modelling Language (UML) merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem *software* yang terkait dengan objek. UML merupakan salah satu alat bantu yang sangat handal dalam bidang pengembangan sistem berorientasi objek karena UML menyediakan Bahasa pemodelan visual yang memungkinkan pengembang sistem membuat *blue print* atas visinya dalam bentuk yang baku. UML berfungsi sebagai jembatan dalam mengkomunikasikan beberapa aspek dalam sistem melalui jumlah elemen grafis yang bisa dikombinasikan menjadi. (Omni Alfina : 2019)

Unified Modeling Language (UML) biasa digunakan untuk :

- a. Menggambarkan batasan sistem dan fungsi - fungsi sistem secara umum, dibuat dengan *use case* dan *actor*.

- b. Menggambarkan kegiatan atau proses bisnis yang dilaksanakan secara umum, dibuat dengan *interaction diagrams*.
- c. Menggambarkan representasi struktur *static* sebuah sistem dalam bentuk *class diagrams*.
- d. Membuat model behavior “yang menggambarkan kebiasaan atau sifat sebuah sistem” dengan *state transition diagrams*.
- e. Menyatakan arsitektur implementasi fisik menggunakan *component and development*.
- f. Menyampaikan atau memperluas *fungsi* dengan *stereotypes*. (Omni Alfina : 2019)

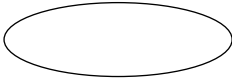
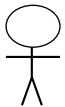
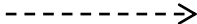
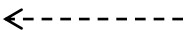
Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem. Dalam membangun perancangan sistem dengan alat bantu perancangan *Unified Modeling Language* (UML) ada beberapa tahapan yang akan dilakukan, yaitu sebagai berikut :

II.2.8.1. Use Case Diagram

Use case diagram merupakan pemodelan untuk sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam

sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut (Ade Hendini, 2016 : 108). Simbol-simbol yang digunakan dalam *use case* diagram dapat dilihat pada tabel II.1 dibawah ini :

Tabel II.1 Simbol Use Case

Gambar	Keterangan	Deskripsi
	<i>Use case</i>	Menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i> .
	Aktor	Sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem.
	Asosiasi	Penghubung antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.
	Asosiasi	Penghubung antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i>	Merupakan di dalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i>	Merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

(Sumber : Ade Hendini, 2016)

II.2.8.2. Class Diagram

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. *Class diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class diagram* secara khas meliputi : Kelas (*Class*), Relasi *Associations*, *Generalization* dan *Aggregation*, atribut (*Attributes*), operasi (*operation/method*) dan *visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *Multiplicity* atau *Cardinality* (Ade Hendini, 2016 : 111).

Tabel II.2. Multiplicity Class Diagram

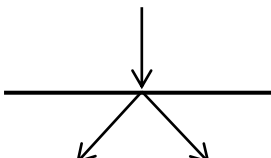
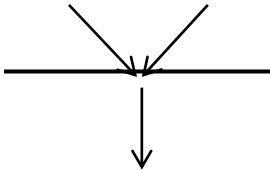
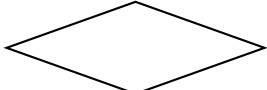
Multiplicity	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Ade Hendini, 2016 : 110)

II.2.8.3. Activity Diagram

Activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram* dapat dilihat pada tabel II.3 dibawah ini:

Tabel II.3. Simbol Activity Diagram

Gambar	Keterangan	Deskripsi
	<i>Start point</i>	Diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i>	Akhir aktifitas.
	<i>Activites</i>	Menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan).	Digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	<i>Join</i> (penggabungan)	Digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i>	Menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> , <i>false</i> .
	<i>Swimlane</i>	Pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.

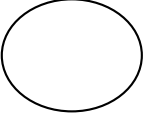
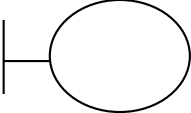
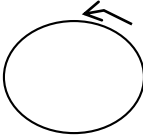
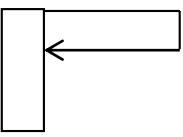
(Sumber : Ade Hendini, 2016 : 109)

II.2.8.4. Sequence Diagram

Sequence diagram menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar

objek. Simbol-simbol yang digunakan dalam *sequence diagram* dapat dilihat pada tabel II.4 dibawah ini :

Tabel II.4. Simbol *Sequence Diagram*

Gambar	Keterangan	Deskripsi
	<i>Entity Class</i>	Merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i>	Berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan <i>form</i> cetak.
	<i>Control class</i>	Suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	<i>Message</i>	Simbol mengirim pesan antar <i>class</i> .
	<i>Recursive</i>	Menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i>	Mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	<i>Lifeline</i>	Garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .

(Sumber : Ade Hendini, 2016 : 110)