

BAB II
LANDASAN TEORI

2.1 Tinjauan Pustaka

Dalam penulisan skripsi ini peneliti menggali informasi dari beberapa penelitian sebelumnya sebagai bahan perbandingan, baik mengenai kekurangan atau kelebihan yang sudah ada. Selain itu, peneliti juga menggali informasi dari buku-buku maupun skripsi dalam rangka mendapatkan suatu informasi yang ada sebelumnya tentang teori yang berkaitan dengan judul yang digunakan untuk memperoleh landasan teori ilmiah.

Dibawah ini merupakan beberapa penelitian terkait, penelitian ini tidak lepas dari penelitian-penelitian terdahulu sebelumnya yang telah dilakukan sehingga menjadikan sebagai acuan atau pedoman dalam melakukan penelitian ini. Selain itu penelitian terkait bertujuan untuk menghindari anggapan kesamaan dengan penelitian ini.

Tabel 2.1 Penelitian Terkait

No	Nama Penelitian (Tahun)	Judul Penelitian	Hasil Penelitian
1	Hera Tria (2022)	Implementasi Sistem Penunjang Keputusan Penentuan Finalis Dalam Pemilihan Duta Menggunakan Metode Analytical Hierarchy Process (Studi Kasus : IIB DARMAJAYA)	Sistem yang dibangun untuk mempermudah pendaftaran dan juga menghasilkan laporan keputusan hasil dari perhitungan yang efektif dan objektif dalam pemilihan Muli Mekhanai Kota Bandar Lampung.

No	Nama Penelitian (Tahun)	Judul Penelitian	Hasil Penelitian
2	Hystenarti Hertiyana (2019)	Sistem Pendukung Keputusan Seleksi Pemilihan Perguruan Tinggi Menggunakan Metode TOPSIS	Membantu calon mahasiswa dalam seleksi pemilihan perguruan tinggi untuk melanjutkan studi. Diperoleh 7 kriteria diantaranya adalah Biaya (BY) yang diperoleh dari biaya perkuliahan, Nilai Akreditasi (NA) yang diperoleh dari nilai akreditasi perguruan tinggi, Prestasi (PS) yang diperoleh dari prestasi akademik dan non akademik, Pendidikan Dosen (PD) yang diperoleh dari pendidikan dosen, Fasilitas (FS) yang diperoleh dari fasilitas yang tersedia, Beasiswa (BS) yang diperoleh dari beasiswa yang ada pada perguruan tinggi dan UKM (UK) yang diperoleh dari unit kegiatan mahasiswa yang tersedia pada perguruan tinggi tersebut. Hasil dari penelitian ini dapat memenuhi tujuan karena hasil perhitungannya bisa lebih cepat.

No	Nama Penelitian (Tahun)	Judul Penelitian	Hasil Penelitian
3	Eny Maria dan Eko Junirianto (2021)	Sistem Pendukung Keputusan Pemilihan Bibit Karet Menggunakan Metode TOPSIS	<p>Sistem pendukung keputusan dapat melakukan pengambilan keputusan dengan menghitung nilai normalisasi dan perkalian bobot pada masing-masing kriteria dan subkriteria dari setiap alternatif sehingga dapat diberikan keputusan pemilihan bibit yang baik. Untuk bibit yang unggul adalah bibit dengan kriteria bentuk payung setengah lingkaran, posisi tangkai daun horizontal, bentuk tangkai daun lurus, warna daun hijau, bentuk daun oval, tulang daun menyirip dan ketebalan daun 0,21-0,24 mm. Berdasarkan hasil pengujian sistem bibit yang unggul adalah Bibit 1 dengan nilai preferensi 1. Hasil Akhir Aplikasi yang disesuaikan dengan perhitungan manual sesuai dengan formula TOPSIS maka diperoleh hasil yang sama, dan telah dikonfirmasi dengan Pakar yang mengetahui tentang pembibitan Karet terbaik dikonfirmasi bahwa Bibit1 yang terbaik diantara alternatif yang ada.</p>

No	Nama Penelitian (Tahun)	Judul Penelitian	Hasil Penelitian
4	Ramos Somya, Retantyo Wardoyo (2019)	Perancangan Sistem Pendukung Keputusan Seleksi Asisten Dosen Menggunakan Kombinasi Metode Profile Matching Dan TOPSIS Berbasis Web Service	pemilihan asisten dosen dapat dimodelkan dengan metode TOPSIS dan Profile Matching dengan teknologi Web Service. Profile Matching berguna untuk menangani parameter yang bukan didasarkan pada nilai maksimum (benefit) dan minimum (cost), tetapi berupa nilai ideal yang harus dipenuhi sesuai dengan persyaratan pengambil keputusan. Hasil dari Profile Matching kemudian dikombinasikan dengan metode TOPSIS untuk mendapatkan keluaran berupa ranking

No	Nama Penelitian (Tahun)	Judul Penelitian	Hasil Penelitian
5	Agus Qurnia Candra (2021)	Sistem Pendukung Keputusan Pemberian Predikat Siswa Teladan Di SMPN2 Seputih Mataram Menggunakan Metode TOPSIS	Membuat sistem untuk digunakan dalam memberikan hasil perhitungan serta perancangan yang dibutuhkan oleh pihak sekolah, seleksi akan digunakan untuk menentukan siapa yang terbaik yang telah disesuaikan dengan kebutuhan dan kriteria yang telah ditentukan sebelumnya. Hasil pengujian dari sistem ini yaitu memberikan rekomendasi pemberian predikat siswa teladan, pengujian dilakukan untuk menguji keakuratan pada sistem dalam hal penilaian.

2.2 Sistem

Sistem adalah suatu kesatuan yang terdiri dari unsur, elemen, prosedur dan sub sistem yang saling berhubungan secara terorganisasi berdasarkan fungsinya, menjadi satu kesatuan untuk mencapai suatu tujuan tertentu [Jumroni, Suwita, & Tiara, 2021]

Teknologi Informasi (TI), atau dalam bahasa Inggris dikenal dengan istilah *Information technology (IT)* merupakan istilah umum untuk teknologi apa pun yang membantu manusia dalam membuat, mengubah, menyimpan, mengomunikasikan dan/atau menyebarkan informasi.

Menurut Kamus Besar Bahasa Indonesia (KBBI), sistem adalah perangkat unsur yang secara teratur saling berkaitan sehingga membentuk suatu totalitas. Sistem juga dapat diartikan sebagai susunan yang teratur dari pandangan, teori,

asas, dan sebagainya. KBBI juga mendefinisikan pengertian sistem sebagai sebuah metode.

Menurut Poerwadarminta Sistem adalah sejumlah bagian yang saling bekerja sama untuk bisa melakukan suatu maksud atau tujuan. Jika salah satu bagian tersebut rusak, proses pengerjaan tugas akan terhambat, dan akibatnya tujuan semakin sulit dicapai.

2.3 Sistem Pendukung Keputusan

Sistem Pendukung Keputusan adalah sebuah sistem yang dimaksudkan untuk mendukung para pengambil keputusan manajerial dalam situasi keputusan semi terstruktur. SPK berfungsi sebagai tambahan atau pendukung bagi pembuat keputusan, dapat memperluas pengetahuan dan kemungkinan, namun tidak menggantikan penilaian [Agarina, Miranti, & Sutedi, 2019].

2.3.1 Tahapan Sistem Pendukung Keputusan

1. Mendefinisikan masalah.
2. Mengumpulkan data atau informasi yang relevan yang saling berkaitan.
3. Pengolaha data dapat menajadi informasi dalam bentuk laporan tulisan atau grafik.
4. Menentukan alternatif berupa solusi yang dapat berbentuk dalam presentase.

2.3.2 Karakteristik Sistem Pendukung Keputusan

1. Mendukung pengambilan keputusan untuk membahas masalah-masalah terstruktur.
2. Output ditujukan bagi personil organisasi dalam semua tingkatan.
3. Mendukung disemua fase proses pengambilan keputusan: intelegensi, desain, dan pilihan.
4. Adanya interface manusia atau mesin, dimana manusia (user) tetap mengontrol proses pengambilan keputusan.

5. Menggunakan model-model matematis dan statistic yang sesuai dengan pembahasan.
6. Memiliki kemampuan dialog untuk memperoleh informasi sesuai dengan kebutuhan.
7. Memiliki subsistem-subsistem yang terintegrasi.
8. Pendekatan easy to use.
9. Kemampuan sistem untuk beradaptasi dengan cepat.

2.3.3 Kriteria Sistem Pendukung Keputusan

Sistem pendukung keputusan dirancang secara khusus untuk mendukung seseorang yang harus mengambil keputusan-keputusan tertentu. Berikut ini beberapa kriteria sistem pendukung keputusan:

1.4 Interaktif

Sistem pendukung keputusan memiliki *user interface* yang komunikatif sehingga pemakai dapat melakukan akses secara cepat ke data dan memperoleh informasi yang dibutuhkan.

2.4 Fleksibel

Sistem pendukung keputusan memiliki sebanyak mungkin *variable* alternatif-alternatif keputusan kepada pemakai.

3. Data Kualitas

Sistem pendukung keputusan memiliki kemampuan untuk menerima data kualitas yang dikuantitaskan yang sifatnya subjektif dari pemakainya, sebagai data masukan untuk pengolahan data. Misalnya terhadap kecantikan yang bersifat kualitas, dapat dikuantitaskan dengan pemberian bobot nilai seperti 75 atau 90.

4. Prosedur Pakar

Sistem pendukung keputusan mengandung surat prosedur yang dirancang berdasarkan rumusan formal atau juga berupa prosedur

kepakaran seseorang atau kelompok dalam menyelesaikan suatu bidang masalah dengan fenomena tertentu.

2.4 Program Keluarga Harapan

Program keluarga harapan atau disingkat PKH merupakan program pemberian bantuan sosial bermasyarakat kepada Keluarga Miskin (KM) yang telah ditetapkan sebagai penerima PKH. Sebagai upaya percepatan penanggulangan kemiskinan, sejak tahun 2007 Pemerintah Indonesia telah melaksanakan PKH. Program Perlindungan Sosial yang juga dikenal di dunia internasional dengan istilah *Conditional Cash Transfers* (CCT) ini terbukti cukup berhasil dalam menanggulangi kemiskinan yang dihadapi di negara-negara tersebut, terutama masalah kemiskinan kronis.

2.5 Metode Topsis

2.5.1 Pengertian Topsis

Metode TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) merupakan suatu bentuk metode pendukung keputusan yang didasarkan pada konsep bahwa alternatif yang terbaik tidak hanya memiliki jarak terpendek dari solusi ideal positif tetapi juga memiliki jarak terpanjang dari solusi ideal negative. TOPSIS menganut prinsip bahwa alternatif yang terpilih harus mempunyai jarak terdekat dari solusi ideal positif dan jarak terpanjang (terjauh) dari solusi ideal negatif dari sudut pandang geometris dengan menggunakan suatu jarak Euclidean (jarak antara kedua titik) untuk menentukan besaran kedekatan relatif dari suatu alternatif dengan solusi yang optimal [Darmawan, Amalia, & Rosiani, 2021].

2.5.2 Kelebihan dan Kekurangan Topsis

a. Kelebihan Topsis

1. Topsis memiliki konsep yang sederhana serta mudah untuk dipahami.

2. Topsis memiliki komputasinya efisien, suatu perhitungan komputasinya lebih efisien serta lebih cepat.
3. Topsis dapat dijadikan sebagai pengukur suatu kinerja alternative dan juga dapat sebagai alternative keputusan dalam sebuah bentuk output komputasi yang sederhana
4. Topsis dapat digunakan sebagai metode pengambilan keputusan yang lebih cepat.

b. Kekurangan Topsis

1. Topsis belum memiliki penentuan bobot prioritas yang menjadi suatu prioritas hitungan terhadap kriteria yang dapat berguna untuk meningkatkan validitas nilai bobot perhitungan kriteria. Dengan alasan tersebut maka metode ini dapat dikombinasikan dengan misalnya metode AHP agar dapat menghasilkan output atau suatu keputusan yang lebih maksimal.
2. Topsis belum memiliki adanya bentuk linguistic untuk suatu penilaian alternative terhadap kriteria. Biasanya bentuk linguistic tersebut dapat di interpretasikan dalam sebuah bilangan fuzzy.
3. Topsis belum memiliki sebuah mediator seperti hirarki yang jika diproses secara mandiri maka dalam ketepatan suatu pengambilan keputusan cenderung belum menghasilkan keputusan yang sempurna.

2.5.3 Tahapan Metode Topsis

Langkah-langkah yang dilakukan dalam menyelesaikan suatu permasalahan menggunakan metode Topsis (Maria, 2018) adalah sebagai berikut:

1. Membuat matriks keputusan yang ternormalisasi

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}}$$

Dimana: r_{ij} adalah hasil dari normalisasi matriks keputusan R dengan $i=1,2,\dots,m$; dan $j=1,2,\dots,n$

X_{ij} adalah nilai dari suatu alternatif (i) terhadap kriteria (j) dengan $i=1,2,\dots,m$; dan $j=1,2,\dots,n$

2. Membuat matriks keputusan yang ternormalisasi terbobot, ditujukan dengan rumus

$$y_{ij} = w_i r_{ij}$$

Dimana:

y_{ij} adalah elemen dari matriks keputusan yang ternormalisasi terbobot Y w_j adalah bobot kriteria ke-j

r_{ij} adalah elemen dari matriks keputusan yang ternormalisasi R

3. Menentukan matriks solusi ideal positif dan solusi negative. Solusi ideal positif dinotasikan A^+ sedangkan solusi negatif dinotasikan A^- . Solusi ideal positif (dari hasil tiap kriteria diambil dari nilai Y terbesar) Berikut ini adalah rumus

$$A^+ = (y_1^+, y_2^+, \dots, y_n^+);$$

Dimana:

y_j^+ adalah: $\max y_{ij}$, jika j adalah atribut keuntungan, $\max y_{ij}$, jika j adalah atribut biaya Solusi ideal negatif (dari hasil tiap kriteria diambil dari nilai Y terkecil). Dengan rumus:

$$A^- = (y_1^-, y_2^-, \dots, y_n^-);$$

Dimana:

y_j^- adalah: $\min y_{ij}$, jika j adalah atribut keuntungan, $\min y_{ij}$, jika j adalah atribut biaya

4. Menentukan jarak antara nilai setiap alternatif dengan matriks solusi ideal positif dan negatif. Jarak adalah alternatif A_i dengan solusi positif dirumuskan sebagai:

$$s_i^+ = \sqrt{\sum_{j=1}^n (y_{ij} - y_j^+)^2}; i = 1, 2, \dots, m$$

Jarak adalah alternatif A_i dengan solusi ideal negatif dirumuskan sebagai:

$$s_i^- = \sqrt{\sum_{j=1}^n (y_{ij} - y_i^-)^2}; i = 1, 2, \dots, m$$

- Menentukan nilai preferensi untuk setiap alternatif. Kedekatan relatif dari alternatif A+ dengan solusi ideal A- dipresentasikan dengan rumus:

$$C_i = \frac{S_i^-}{S_i^- + S_i^+}$$

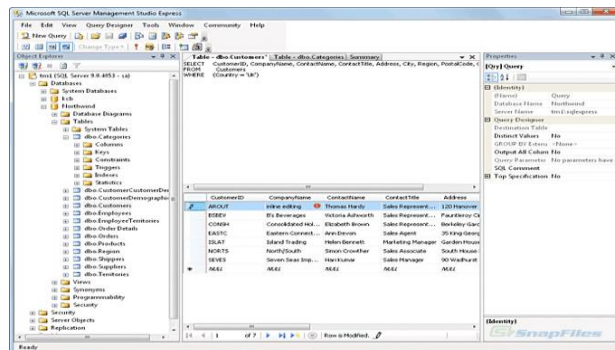
Dengan $0 < C_i < 1$ dan $i = 1, 2, 3, \dots, m$

Meranking alternatif. Alternatif dapat diranking berdasarkan urutan C_i . Maka dari itu, alternatif terbaik adalah salah satu yang berjarak terpendek terhadap solusi ideal dan berjarak terjauh dengan solusi ideal negatif.

2.6 Alat-Alat Pengembangan Sistem

2.6.1 MySQL

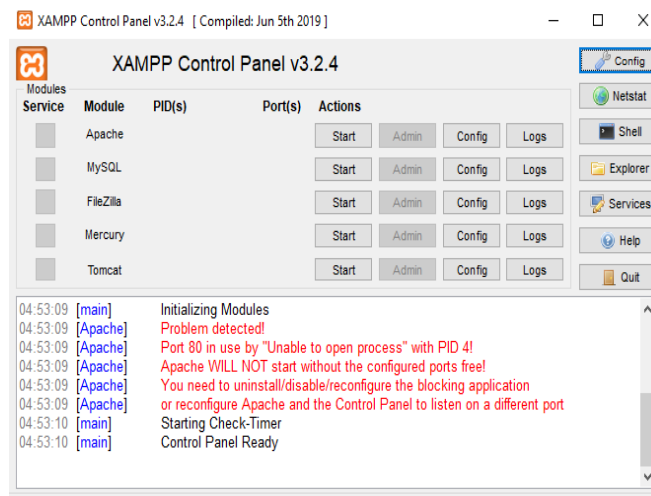
MySQL pada awalnya diciptakan pada tahun 1979, oleh Michael "Monty" Widenius, seorang programmer komputer asal Swedia. Monty mengembangkan sebuah sistem database sederhana yang dinamakan UNIREG yang menggunakan koneksi low-level ISAM *database engine* dengan indexing. MySQL merupakan sistem manajemen database yang bersifat relational. Artinya, data yang dikelola dalam database yang akan diletakkan pada beberapa tabel yang terpisah sehingga manipulasi data akan jauh lebih cepat [Novendri, Saputra, & Firman, 2019].



Gambar 2.1 MySQL

2.6.4 XAMPP

Fungsinya sebagai server yang berdiri sendiri (localhost), yang terdiri dari program Apache HTTP Server, database MySQL, dan penerjemah bahasa yang ditulis dalam pemrograman PHP dan Perl [Mearaj et al., 2019]. Nama XAMPP adalah singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU General Public License dan gratis.



Gambar 2.4 XAMPP

2.6.5 Visual Studio Code

Visual Studio Code (VS Code) ini adalah sebuah teks editor ringan dan handal yang dibuat oleh Microsoft untuk sistem operasi multiplatform, artinya tersedia juga untuk versi Linux, Mac, dan Windows. Teks editor ini secara langsung mendukung bahasa pemrograman JavaScript, Typescript, dan Node.js, serta bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang via marketplace Visual Studio Code (seperti C++, C#, Python, Go, Java, dst). Banyak sekali fitur-fitur yang disediakan oleh Visual Studio Code, diantaranya Intellisense, Git Integration, Debugging, dan fitur ekstensi yang menambah kemampuan teks editor [Permana & Romadhon, 2019].

2.7 Metode Pengembangan Sistem

2.7.1 Metode RUP (*Rational Unified Process*)

Metode penelitian yang dikembangkan menggunakan pendekatan *Rational Unified Process* (RUP) yaitu pengembangan aplikasi secara iterasi atau berulang. RUP merupakan proses yang terdapat pada perancangan perangkat lunak dengan pendefinisi yang lebih baik dan penstrukturan yang baik. Metode RUP terdiri dari empat tahapan yaitu: (1) *inception*/permulaan, (2) *elaboration*/perluasan atau perencanaan, (3) *construction*/konstruksi, dan (4) *transition*/transisi.

1. *Inception*/Permulaan

Pada tahap *inception* dilakukannya pembuatan bisnis model dari sistem serta dirumuskannya kebutuhan dari sistem. Pengambilan data dapat dilakukan dengan wawancara.

2. *Elaboration*/Perluasan

Pada tahap *inception* dilakukannya pembuatan bisnis model dari sistem serta dirumuskannya kebutuhan dari sistem. Pengambilan data dapat dilakukan dengan wawancara.

3. *Construction*/Konstruksi

Pada tahap ini mengubah desain sistem ke dalam sistem yang jadi. Dengan menggunakan use case diagram pada tahap perancangan sistem terbentuk menjadi program yang berorientasi objek dan dibangun dengan menggunakan konsep MVC (*Model View Controller*).

4. *Transition*/Transisi

Pada tahap ini dilakukan pengujian terhadap logika atau fungsional dari sistem yang dibuat. Pengecekan dilakukan tidak terdapat error pada sistem. Untuk pengujian sendiri biasa digunakan whitebox testing, usability testing maupun blackbox testing.

5. UML (*Unified Modeling Language*)

Unified Modeling Language atau biasa disebut UML merupakan sebuah pemodelan berorientasi objek dan desain. Pemodelan ini pada awalnya digunakan oleh Grady Booch, Jim Rumbaugh, dan

Ivars Jacobson pada tahun 1990[11] . UML memiliki beberapa tahap dalam pemodelan sistem yang pertama use case diagram, activity diagram, sequence diagram dan class diagram.

6. *BlackBox Testing*

Unified Modeling Language atau biasa disebut UML merupakan sebuah pemodelan berorientasi objek dan desain. Pemodelan ini pada awalnya digunakan oleh Grady Booch, Jim Rumbaugh, dan Ivars Jacobson pada tahun 1990. UML memiliki beberapa tahap dalam pemodelan sistem yang pertama use case diagram, activity diagram, sequence diagram dan class diagram.

2.7.2 **Unified Model Language (UML)**

UML (*Unified Modelling Language*) adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung [Rosa and Shalahuddin, 2019].

Tujuan penggunaan UML adalah untuk memodelkan suatu sistem yang menggunakan konsep berorientasi objek dan menciptakan Bahasa pemodelan yang dapat digunakan oleh manusia maupun mesin [Purwati and Rahardi, 2018].


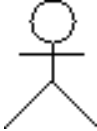

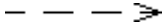
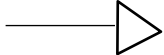
Berikut ini merupakan penjelasan tentang masing-masing diagram yang ada pada UML (*Unified Modelling Language*):

2.7.2.1 *Use Case Diagram*

Merupakan gambaran atau representasi dari interaksi yang terjadi antara sistem dan lingkungannya. *Use case* adalah teknik penemuan kebutuhan perangkat lunak yang dikenalkan pertama kali dalam metode pendekatan berbasis objek yang dikembangkan oleh Jacobson dan kawankawan pada tahun 1990an. Saat ini, diagram use case menjadi salah satu tipe dari Diagram *Unified Modeling Language* (UML) berbasis tingkah laku. Menurut Alistair Cockburn, use case mendeskripsikan

tingkah laku sistem di berbagai kondisi dan bagaimana sistem tersebut menanggapi permintaan pengguna.






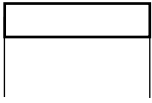
Tabel 2.2 Simbol *Use Case Diagram*

No	Simbol	Nama simbol	Deskripsi
1		<i>Usecase</i>	Menggambarkan fungsionalitas dari suatu sistem, sehingga pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun.
2		Aktor	Menggambarkan orang atau sistem yang menyediakan atau menerima informasi dari sistem atau menggambarkan pengguna <i>software</i> aplikasi (<i>user</i>).
3		Asosiasi <i>/assosiation</i>	Komunikasi antara actor dan usecase yang berpartisipasi pada usecase dan memiliki interaksi dengan actor
4		Ekstend / <i>extend</i>	Relasi usecase tambahan ke sebuah usecase, dimana usecase yang ditambahkan dapat berdiri sendiri walaupun tanpa usecase tambahan
5		Generalisasi	Hubungan generalisasi dengan spesialisasi (umum-khusus) antara dua buah usecase dimana fungsi yang satu merupakan fungsi yang lebih umum dari lainnya

2.7.2.2 Activity Diagram

Diagram aktivitas atau *activitydiagram* menggambarkan *workflow* (alirankerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak.

Tabel 2.3 Simbol *Activity Diagram*

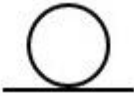
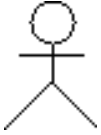
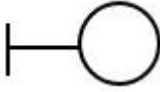


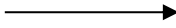
No	Simbol	Nama simbol	Deskripsi
1		Status awal	Status awal aktifitas sistem. Sebuah diagram aktifitas memiliki sebuah status awal
2		Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
3		Percabangan/ <i>decision</i>	Asosiasi percabangan dimana jika ada pilihan aktifitas lebih dari satu
4		Pengabungan/ <i>join</i>	Asosiasi pengabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
5		Status akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
6		<i>Swimlane</i>	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktifitas yang terjadi

2.7.2.3 Sequence Diagram

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

[Mandasari et al., 2016].

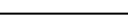

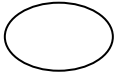
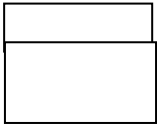

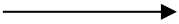
Tabel 2.4 Simbol-Simbol Sequence Diagram

No	Simbol	Nama Simbol	Deskripsi
1		<i>Entity Class</i>	Menggambarkan hubungan yang akan dilakukan
2		Aktor	Menggambarkan orang atau sistem yang menyediakan atau menerima informasi dari sistem atau menggambarkan pengguna <i>software</i> aplikasi (<i>user</i>).
3		<i>Boundary Class</i>	Menggambarkan sebuah gambaran dari foem
4		<i>Control Class</i>	Menggambarkan penghubung antara boundary dengan tabel
5		<i>A focus of control & a life line</i>	Menggambarkan tempat mulai dan berakhirnya message
6		<i>A message</i>	Menggambarkan pengiriman pesan

2.7.2.4 Class Diagram

Class diagram menggambarkan struktur sistem dari sudut pandang definisi kelas-kelas yang nantinya dibuat menciptakan sistem [Mandasari et al., 2016]

Tabel 2.5 Simbol-Simbol *Class Diagram*

No	Simbol	Nama Simbol	Deskripsi
1		<i>Generalization</i>	Hubungan dimana objek anak berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk.
2		<i>Navy Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor.
4		<i>Class</i>	Himpunan dari objek-objek yang terbagi atribut operasi yang sama.
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek
6		<i>Depedency</i>	Himpunan dimana perubahan yang terjadi pada suatu elemen mandiri akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.