

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terdahulu

Berdasarkan penelitian yang dilakukan oleh Hummairoh, dkk (2021) mengenai Penerapan WASPAS Dalam Menentukan Kelayakan Penerima Bantuan Program Keluarga Harapan Kelurahan Sudirejo-I, Hummairoh, dkk menyimpulkan bahwa sistem ini dapat menghasilkan perhitungan yang akurat serta tepat sasaran. Pengujian metode berdasarkan perhitungan manual yang dilakukan oleh sistem terbukti sama dan sesuai dengan yang diharapkan. Dengan sistem ini maka akan menjadikan kinerja pihak Kota Medan Kelurahan Sudirejo I dalam menentukan bantuan menjadi lebih mudah dan resiko kecurangan menjadi lebih kecil.

Berdasarkan penelitian yang dilakukan oleh Laia, dkk (2021) mengenai Sistem Pendukung Keputusan Dalam Menentukan Lokasi Strategis Cabang Baru di UD. Ario Nias Selatan Menggunakan Metode WASPAS, Laia, dkk menyimpulkan bahwa berdasarkan dari metode *weighted aggregated sum product assesment* (WASPAS) dapat diterapkan untuk menentukan lokasi baru dari UD Ario Nias Selatan.

Berdasarkan penelitian yang dilakukan oleh Negroho dan Wahyuni (2021) mengenai Sistem Pendukung Keputusan Kelayakan Kenaikan Gaji Pegawai Menggunakan Metode WASPAS, Negroho dan Wahyuni menyimpulkan bahwa hasil perhitungan dengan metode WASPAS ada 5 pegawai yang dinyatakan Layak untuk kenaikan gaji, hal itu dikarenakan berdasarkan perhitungan dari metode dan

nilai yang lebih besar dari nilai 8, Adapun 5 pegawai yang layak ata kenaikan gaji yaitu Harris, Rusmanto, Manto, Koko dan Warman sedangkan yang tidak layak ada 15 pegawai..

Berdasarkan penelitian yang dilakukan oleh Novriandi dan Utami (2021) mengenai Sistem Pendukung Keputusan Kelayakan Pemberian Dana Pinjaman Pada Karyawan Menggunakan Metode WASPAS (Studi Kasus CV Vano Putra), Novriandi dan Utami menyimpulkan bahwa sistem pendukung keputusan ini menghasilkan nilai yang tepat dalam waktu yang cepat sehingga penerapan sistem ini sangat membantu perusahaan dalam menentukan keputusannya untuk memberikan dana pinjaman pada karyawan yang tepa.

Berdasarkan penelitian yang dilakukan oleh Siallagon dan Safrizal (2020) mengenai Sistem Pendukung Keputusan Menentukan Kelayakan Bantuan Ternak Oleh Dinas Ketahanan Pangan Dan Peternakan PROVSU Menggunakan Metode WASPAS, Siallagon dan Safrizal menyimpulkan bahwa penerapan metode WASPAS pada sistem pendukung keputusan dilakukan dengan cara memasukkan setiap data alternatif dan kriteria serta memberikan nilai pada setiap alternatif dan memberikan bobot pada kriteria lalu disinkronkan dengan tahapan-tahapan pada metode WASPAS Sehingga akan menghasilkan alternatif yang terlayak mendapatkan bantuan ternak menggunakan metode WASPAS.

II.2. Landasan Teori

Landasan teori peneliti kutip dari beberapa teori yang berkaitan dengan penelitian ini yaitu:

II.2.1. Sistem Pendukung Keputusan

Sistem pendukung keputusan atau *Decision Support System* yang merupakan salah satu contoh pemrograman yang berbasis pengetahuan, dalam menggunakan sumber daya individu secara intelektual dan kemampuan komputer untuk meningkatkan mutu memilih keputusan. Hal ini merupakan dari penggunaan komputer berbasiskan sistem pendukung yang berhadapan dengan masalah setengah tersusun. (Laia, dkk, 2021: 219).

Sistem pendukung keputusan adalah suatu sistem yang bisa memberikan pemecahan masalah dengan kondisi semi terstruktur dan tak terstruktur. Perlu diketahui bahwa disini sistem pendukung keputusan merupakan *tool* pendukung, bukan sebagai *tool* pengambil keputusan. Tujuan spk adalah menyediakan informasi, membimbing, memberikan taksiran atau perkiraan, dan mengarahkan user atau pengguna informasi agar dapat melakukan pengambilan keputusan dengan baik. (Hummairoh, dkk, 2021: 333).

II.2.1.1. Karakteristik Utama Sistem Pendukung Keputusan

Dari pengertian Sistem Pendukung Keputusan maka dapat ditentukan karakteristik antara lain:

1. Sistem yang berbasis komputer.

2. Dipergunakan untuk membantu para pengambil keputusan.
3. Melalui cara simulasi yang interaktif.
4. Dimana data dan model analisis sebagai komponen utama. (Apriani, 2019: 11).

II.2.1.2. Proses Pengambilan Keputusan

Pengambilan keputusan meliputi beberapa tahap dan melalui beberapa proses, pengambilan keputusan meliputi empat tahap yang saling berhubungan dan berurutan. Empat proses tersebut adalah:

1. *Inteligenci*

Tahap ini merupakan proses penelusuran dan pendekatan dari lingkup problematika serta proses pengenalan masalah. Data masukan diperoleh, diproses, dan diuji dalam rangka mengidentifikasi masalah.

2. *Design*

Tahap ini merupakan proses menemukan dan mengembangkan alternatif. Tahap ini meliputi proses untuk mengerti masalah, menurunkan solusi dan menguji kelayakan solusi.

3. *Choice*

Pada tahap ini dilakukan proses pemilihan di antara berbagai alternatif tindakan yang mungkin dijalankan. Tahap ini meliputi pencarian, evaluasi, dan rekomendasi solusi yang sesuai untuk variabel hasil pada alternatif yang dipilih.

4. *Implementation*

Tahap implementasi adalah tahap pelaksanaan dari keputusan yang telah diambil. Pada tahap ini perlu disusun serangkaian tindakan yang terencana, sehingga hasil keputusan dapat dipantau dan disesuaikan apabila diperlukan perbaikan. Dalam hal ini, model Simon juga menggambarkan kontribusi Sistem Informasi Manajemen (SIM) dan Ilmu Manajemen / *Operations Research* (IM / OR) terhadap proses pengambilan keputusan. (Apriani, 2019: 11).

II.2.1.3. Karakteristik Sistem Pendukung Keputusan

Karakteristik dan kapabilitas kunci dari Sistem Pendukung Keputusan adalah sebagai berikut:

1. Dukungan untuk pengambilan keputusan, terutama pada situasi semiterstruktur dan tak terstruktur.
2. Dukungan untuk semua level manajerial, dari eksekutif puncak sampai manajer lini.
3. Dukungan untuk individu dan kelompok.
4. Dukungan untuk semua keputusan independen dan sekuensial.
5. Dukungan di semua fase proses pengambilan keputusan: inteligensi, desain, pilihan, dan implementasi.
6. Dukungan pada berbagai proses dan gaya pengambilan keputusan.
7. Kemampuan Sistem beradaptasi dengan cepat dimana pengambilan keputusan dapat menghadapi masalah-masalah dengan cara mengadaptasi sistem terhadap kondisi-kondisi perubahan yang terjadi.

8. Pengguna merasa seperti rumah. *User-friendly*, kapabilitas grafis yang kuat, dan sebuah bahasa interaktif yang alami.
9. Peningkatan terhadap keefektifan pengambilan keputusan (akurasi, timelines, kualitas) dari pada efisiensi (biaya).
10. Pengambil keputusan mengontrol penuh semua langkah proses pengambilan keputusan dalam memecahkan masalah.
11. Pengguna akhir dapat mengembangkan dan memodifikasi sistem sederhana.
12. Menggunakan model-model dalam penganalisisan situasi pengambilan keputusan.
13. Disediakkannya akses untuk berbagai sumber data, format, dan tipe, mulai dari Sistem informasi geografis (GIS) sampai sistem berorientasi objek.
14. Dapat dilakukan sebagai alat standalone yang digunakan oleh seorang pengambil keputusan pada satu lokasi atau didistribusikan di satu organisasi keseluruhan dan di beberapa organisasi sepanjang rantai persediaan. (Apriani, 2019: 11).

II.2.1.4. Komponen-Komponen Sistem Pendukung Keputusan

Sistem Pendukung Keputusan terdiri dari empat subsistem, yaitu:

1. Manajemen Data, meliputi basis data yang berisi data-data yang relevan dengan keadaan dan dikelola oleh perangkat lunak yang disebut dengan *Database Management System* (DBMS).

2. Manajemen Model berupa sebuah paket perangkat lunak yang berisi model-model *financial*, *statistic*, *management science*, atau model kuantitatif, yang menyediakan kemampuan analisa dan perangkat lunak manajemen yang sesuai.
3. Subsistem Dialog atau Komunikasi, merupakan subsistem yang dipakai oleh *user* untuk berkomunikasi dan member perintah (menyediakan *user interface*).
4. Manajemen *knowledge* yang mendukung subsistem lain atau sebagai komponen yang berdiri sendiri. (Apriani, 2019: 11).

II.2.1.5. Komponen Besar Sistem Pendukung Keputusan

Secara umum DSS dibangun oleh tiga komponen besar yaitu:

1. *Database Management*.
2. *Model Base*.
3. *Software System/User Interface*. (Apriani, 2019: 11).

II.2.2. Kelayakan

Studi Kelayakan merupakan studi kelayakan suatu proyek atau usaha yang dapat menjadi investasi untuk dilaksanakan. Terdapat lima tujuan dari studi kelayakan adalah sebagai berikut:

1. Menghindari Resiko/Dampak.

Resiko adalah bahaya, akibat atau konsekuensi yang dapat terjadi akibat sebuah proses yang sedang berlangsung atau kejadian yang akan datang. Dalam bidang asuransi, resiko dapat diartikan sebagai suatu keadaan ketidakpastian, di mana

jika terjadi suatu keadaan yang tidak dikehendaki dapat menimbulkan suatu kerugian.

2. Memudahkan Perencanaan Usaha.

Rencana usaha adalah pernyataan formal atas tujuan berdirinya sebuah bisnis, serta alasan mengapa pendirinya yakin bahwa tujuan tersebut dapat dicapai, serta strategi atau rencana-rencana apa yang akan dijalankan untuk mencapai tujuan tersebut.

3. Memudahkan Pelaksanaan.

Pelaksanaan diartikan sebagai suatu usaha atau kegiatan tertentu yang dilakukan untuk mewujudkan rencana atau program dalam kenyataannya.

4. Memudahkan Pengawasan.

Pelaksanaan diartikan sebagai suatu usaha atau kegiatan tertentu yang dilakukan untuk mewujudkan rencana atau program dalam kenyataannya.

5. Memudahkan Pengendalian.

Pengendalian adalah pengukuran dan perbaikan terhadap pelaksanaan kerja bawahan, agar rencana-rencana yang telah dibuat untuk mencapai tujuan-tujuan perusahaan dapat diselenggarakan. (Harari, dkk, 2021: 2).

II.2.3. Jaminan

Jaminan merupakan aset pihak peminjam yang diberikan kepada pemberi pinjaman jika peminjam tidak dapat mengembalikan pinjamannya maka aset yang dijaminan dapat dimiliki oleh orang yang memberikan pinjaman. Jaminan merupakan aset pihak peminjam yang diberikan kepada pemberi pinjaman jika

peminjam tidak dapat mengembalikan pinjamannya maka aset yang dijaminkan dapat dimiliki oleh orang yang memberikan pinjaman. (Hidayani dan Pohan, 2020: 204).

II.2.3.1. Sistem Jaminan Tertutup

Sistem pengaturan hukum jaminan terbuka adalah sistem pengaturan bahwa orang dapat mengadakan perjanjian apapun juga tentang jaminannya sesuai dengan kesepakatan bersama dengan saksi dan jaminan berupa gadai. (Hidayani dan Pohan, 2020: 204).

II.2.3.2. Sistem Jaminan Terbuka

Sistem pengaturan hukum jaminan terbuka adalah sistem pengaturan bahwa orang dapat mengadakan perjanjian apapun juga tentang jaminannya sesuai dengan Kitab Undang-Undang Hukum Perdata dan tidak menentang ketertiban umum yang berlaku dalam masyarakat. (Hidayani dan Pohan, 2020: 204).

II.2.4. Pinjaman

Pinjaman adalah pemberian sejumlah uang dari suatu pihak (lembaga keuangan, seseorang atau perusahaan) kepada pihak lain (seseorang atau perusahaan) yang mewajibkan pinjamannya untuk melunasi dalam jangka waktu tertentu dengan jumlah bunga yang disepakati bersama. Dalam menghimpun SHU, maka koperasi simpan pinjam biasanya memperoleh keuntungan dari jasa dan atau bagi hasil yang diberikan oleh anggota dalam kegiatan pembiayaan atau pinjaman modal usaha yang di kerjasamakan dengan anggota koperasi. (Mbulu, 2019: 167).

II.2.4.1. Pinjaman Langsung

Pinjaman langsung adalah fasilitas pinjaman uang oleh penyedia jasa keuangan yang beroperasi dengan menyediakan kantor jasa peminjaman. Pinjaman langsung akan cair jika persyaratan dan jaminan terpenuhi. Alasan peminjam melakukan pinjaman langsung adalah menginginkan informasi yang lebih rinci mengenai lokasi, pekerja dan kejelasan tambahan. (Nuhayati, dkk, 2021: 74).

II.2.4.2. Pinjaman Online

Pinjaman *Online* adalah fasilitas pinjaman uang oleh penyedia jasa keuangan yang beroperasi secara *online*. Penyedia pinjaman *online* tersebut dikenal dengan sebutan *fintech*. Pinjaman *online* yang langsung cair dan tanpa jaminan merupakan solusi alternatif bagi masyarakat yang membutuhkan dana tunai tanpa harus mengajukannya secara tatap muka. Alasan peminjam melakukan pinjaman *online* adalah kesulitan dalam mengakses layanan keuangan formal dengan berbagai persyaratan administrasi yang harus dipenuhi. Persyaratan administrasi pinjaman online relatif lebih mudah jika dibandingkan dengan pinjaman layanan keuangan formal terlebih lagi pada situasi pandemi covid-19. (Nuhayati, dkk, 2021: 74).

II.2.5. Metode *Weight Aggregated Sum Product Assessment* (WASPAS)

Metode *Weighted Aggregated Sum Product Assessment* (WASPAS)” merupakan kombinasi dari pendekatan MCDM yang diketahui yaitu model jumlah tertimbang (*Weighted Sum Model*/WSM) dan model produk tertimbang (WPM) pada awalnya membutuhkan normalisasi linier dari elemen matriks keputusan

dengan menggunakan dua persamaan. Metode WASPAS digunakan untuk memecahkan berbagai masalah seperti di pembuatan keputusan, evaluasi alternatif dan seterusnya. WASPAS merupakan pengambilan keputusan yang memiliki kemampuan mencari solusi ideal positif dan solusi ideal negatif dalam memecahkan permasalahan yang ada. (Negroho dan Linda, 2021: 2). Berikut langkah-langkah metode WASPAS:

1. Menentukan normalisasi matrix dalam pengambilan keputusan

$$X = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \dots & \dots & \dots & \dots \\ x_{m1} & x_{m2} & \dots & x_{mn} \end{pmatrix} \dots\dots\dots(1)$$

Keterangan:

m: nomor urutan alternatif

n: nomor urutan kriteria

X: matriks keputusan

2. Langkah membuat matrix normalisasi

- a. Jika menggunakan benefit

$$x_{ij} = x_{ij} / (\max_i x_{ij}) \dots\dots\dots(2)$$

- b. Jika menggunakan cost

$$x_{ij} = (\min_i x_{ij}) / x_{ij} \dots\dots\dots(3)$$

Dimana:

X_{ij} = Nilai kinerja alternatif sehubungan kriteria j

$\max_i X_{ij}$ = Nilai maksimum dari setiap kolom dan baris

$\min_i X_{ij}$ = Nilai minimum dari setiap kolom dan baris

Kriteria keuntungan(benefit): jika nilai terbesar adalah terbaik

Kriteria biaya(cost): jika nilai terkecil adalah terbaik

3. Normalisasi

$$Q_i = 0,5 \sum_{j=1}^n x_{ij} w_j + 0,5 \prod_{j=1}^n (x_{ij}) w_j = 1 \dots \dots \dots (4)$$

Dimana:

0,5 = Ketetapan

Q_i = Nilai Akhir dari Alternatif

W_j = Bobot yang ditentukan

II.2.6. *Hypertext Preprocessor (PHP)*

PHP merupakan singkatan dari “*Hypertext Preprocessor*”. PHP adalah sebuah bahasa *scripting* yang terpasang pada HTML. Sebagian besar sintaknya mirip dengan bahasa pemrograman C, Java, ASP dan Perl ditambah beberapa fungsi PHP yang Spesifik dan mudah dimengerti. PHP digunakan untuk membuat tampilan *web* menjadi lebih dinamis, dengan PHP anda bisa menampilkan atau menjalankan beberapa *file* dalam 1 file dengan cara di *include* dan *require*. PHP itu sendiri sudah dapat berinteraksi dengan beberapa *database* walaupun dengan kelengkapan yang berbeda yaitu seperti DBM, MySQL, Oracle. (Rahmasari, 2019: 415).

PHP adalah bahasa pemrograman yang sering disisipkan ke dalam HTML. PHP sendiri berasal dari kata *Hypertext Preprocessor*. Sejarah PHP pada awalnya merupakan kependekan dari *Personal Home Page* (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama *Form Interpreted* (FI), yang wujudnya berupa sekumpulan skrip yang

digunakan untuk mengolah data formulir dari *web*. Bahasa pemrograman ini menggunakan sistem *server-side*. *Server-side programming* adalah jenis bahasa pemrograman yang nantinya *script*/program tersebut akan dijalankan/diproses oleh *server*. Kelebihannya adalah mudah digunakan, sederhana, dan mudah untuk dimengerti dan dipelajari. (Sugijanto, dkk, 2020: 2).

Beberapa kelebihan dari PHP sebagai berikut.

1. Kesederhanaan, mudah dipelajari karna banyak referensi serta bisa membuat *website* dinamis.
2. PHP bersifat *open source*, karena bersifat *open source* PHP mudah didapatkan dan tersedia secara versi-versi baru dalam jangka waktu yang cepat.
3. Stabilitas dan Kompatibilitas, PHP stabil di berbagai sistem operasi seperti linux dan Macs selain itu PHP juga terintegrasi secara baik dengan berbagai macam *webserver* termasuk 2 yang paling populer yaitu IIS dan Apache.
4. Kemampuan proses cepat dalam menampilkan halaman *web* serta mampu berintraksi dengan banyak *database*.

Dalam penulisan sintak kode PHP diawali dengan tanda “” di akhir blok kodenya. Pada setiap baris instruksi program di akhir dengan tanda titik koma “.” artinya walaupun menuliskan lebih dari satu baris tetapi akan di anggap satu baris. (Huzaeni, 2019: 140).

Contoh PHP:

```
<?php echo"Hello World";?>
```

II.2.7. *Hyper Text Markup Language (HTML)*

HTML ialah kepanjangan dari *Hypertext Markup Language*. Definisi HTML adalah bahasa yang digunakan untuk menulis halaman *web*. fungsi utama HTML ialah memberi perintah pada *browser* untuk melakukan manipulasi tampilan melalui *tag-tag* yang ditulis dalam HTML. (Rahmasari, 2019: 415).

HTML adalah singkatan dari *Hypertext Markup Language*. HTML memungkinkan seorang *user* untuk membuat dan menyusun bagian paragraf, *heading*, *link* atau tautan, dan *blockquote* untuk halaman *web* dan aplikasi. HTML bukanlah bahasa pemrograman, dan itu berarti HTML tidak punya kemampuan untuk membuat fungsionalitas yang dinamis. Sebagai gantinya, HTML memungkinkan *user* untuk mengorganisir dan memformat dokumen, sama seperti Microsoft Word. (Sugijanto, dkk, 2020: 2).

Contoh tag HTML:

<HTML>

<HEAD>

<TITLE>tag html</HTML>

</HEAD>

<BODY>

.....

</BODY>

</HTML>

II.2.8. MySQL

MySQL merupakan *database engine* atau *server database* yang mendukung bahasa *database* pencarian SQL. MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread*, *multi-user*. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL. (Sugijanto, dkk, 2020: 2).

Definisi MySQL merupakan *software* RDMS (*Relational Database Management System*) yang dapat mengelola *database* dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak pengguna dan dapat melakukan suatu proses secara sinkron atau bersamaan. (Rahmasari, 2019: 415).

Perintah dasar MySQL:

1. Create Database

Perintah ini untuk membuat database baru.

2. Create Table

Perintah ini untuk membuat tabel baru.

3. Insert Into

Perintah ini untuk menyimpan data baru.

4. Select

Perintah ini untuk melakukan pencarian.

5. Delete

Perintah ini untuk menghapus isi data pada tabel.

6. Update

Perintah ini untuk mengubah isi data pada tabel.

7. Alter Table

Perintah ini untuk mengganti tipe data pada tabel. (Rahmasari, 2019: 415).

II.2.9. *Unified Modeling Language (UML)*

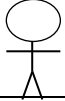
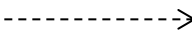
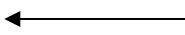
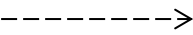
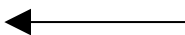
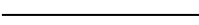
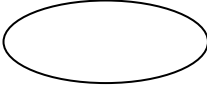
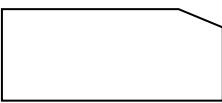
Unified Modeling Language (UML) yaitu satu kumpulan konvensi permodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem perangkat lunak yang terkait dengan objek. UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan sistem yang besar dan kompleks. UML tidak hanya digunakan dalam proses pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan. (Andikos, 2019: 39).

Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

1. *Use Case Diagram*

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case diagram* dapat digambarkan dengan sumber-sumber pada Tabel II.1.

Tabel II.1. Simbol Use Case

Gambar	Nama	Keterangan
	<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>	Menspesifikan 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 (sinergi).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

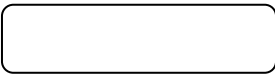
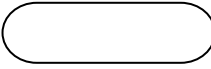
(Sumber: Andikos, 2019: 39)

2. Diagram Aktivitas (Activity Diagram)

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin

terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* dapat digambarkan dengan simbol-simbol seperti pada tabel II.2.

Tabel II.2. Simbol Activity Diagram

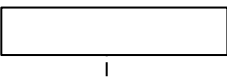
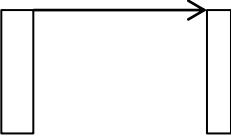
Gambar	Nama	Keterangan
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Activity Final</i>	Bagaimana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

(Sumber: Andikos, 2019: 39)

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

Gambar	Nama	Keterangan
	<i>Lifeline</i>	Objek entity, 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.
---	----------------	---

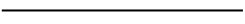
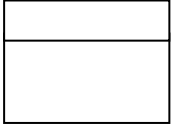
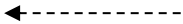
(Sumber: Andikos, 2019: 39)

4. *Class Diagram* (Diagram Kelas)

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek.

Class diagram menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.4.

Tabel II.4. *Class Diagram*

Gambar	Nama	Keterangan
	<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: Andikos, 2019: 39)