

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terdahulu

Berdasarkan penelitian yang dilakukan oleh Astuti dan Saragih (2020) mengenai Sistem Pendukung Keputusan Pemilihan Sekolah Terbaik Dengan Metode MOORA, Astuti dan Saragih menyimpulkan metode yang digunakan yaitu MOORA akan membantu dalam pengambilan keputusan yang menentukan nilai bobot dari setiap atribut.

Berdasarkan penelitian yang dilakukan oleh Hasibuan, dkk (2019) mengenai Perbandingan Metode MOORA Dan TOPSIS Dalam Penentuan Penerimaan Siswa Baru Dengan Pembobotan ROC Menggunakan GUI Matlab, Hasibuan, dkk menyimpulkan Nilai rata-rata prosentase sensitivitas yang diperoleh untuk metode MOORA (-1,61%) lebih besar dibandingkan metode TOPSIS (-7,96%). Sehingga dapat disimpulkan bahwa metode yang paling tepat yang bisa digunakan pada kasus ini adalah metode MOORA.

Berdasarkan penelitian yang dilakukan oleh Nisa (2020) mengenai Metode MOORA Dan WASPAS Untuk Pengambilan Keputusan Penentuan Prioritas Dalam Peningkatan Kualitas Mata Pelajaran, Nisa menyimpulkan dari hasil penggunaan Metode MOORA dan WASPAS dalam penentuan mata pelajaran yang menjadi prioritas utama dalam peningkatan kualitas maka didapatkan hasil keputusan mata pelajaran Bahasa Arab dan Matematika sebagai mata pelajaran yang menjadi prioritas utama dalam peningkatan kualitasnya.

Berdasarkan penelitian yang dilakukan oleh Mesran, dkk (2018) mengenai Sistem Pendukung Keputusan Pemilihan Peserta Jaminan Kesehatan Masyarakat (Jamkesmas) Menerapkan Metode MOORA, Mesran, dkk menyimpulkan Sistem pendukung keputusan dapat mengatasi permasalahan pemilihan peserta Jamkesmas menjadi lebih tersistem dan tepat pada masyarakat yang benar-benar membutuhkan.

Berdasarkan penelitian yang dilakukan oleh Amalia (2020) mengenai Sistem Pendukung Keputusan Pemilihan Objek Wisata Unggulan Menggunakan Metode MOORA, Amalia menyimpulkan hasil pengujian menunjukkan bahwa sistem ini sudah menghasilkan hasil yang cukup akurat. Hasil menunjukkan bahwa 45 data wisata dari total 153 data wisata pada tahap pengujian data manual dibandingkan dengan pengujian menggunakan SPK telah mencapai tingkat keberhasilan 99%.

II.2. Landasan Teori

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

II.2.1. Sistem Pendukung Keputusan (SPK)

Sistem Pendukung Keputusan (SPK) merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan, dan pemanipulasian data. Selain itu digunakan untuk membantu pengambilan keputusan dalam situasi semi terstruktur dan situasi yang tidak terstruktur, dimana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat. (Amalia, 2020 : 24).

Sistem Pendukung Keputusan (SPK) atau *Decision Support System* (DSS) adalah sebuah sistem yang mampu memberikan kemampuan pemecahan masalah maupun kemampuan pengkomunikasian untuk masalah dengan kondisi terstruktur dan tak terstruktur. Sistem ini digunakan untuk membantu pengambilan keputusan dalam situasi terstruktur dan situasi yang tidak terstruktur, dimana tak seorangpun tahu secara pasti bagaimana keputusan seharusnya dibuat. (Nisa, 2020 : 22).

II.2.1.1. Karakteristik Dan Kemampuan Sistem Pendukung Keputusan

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

1. Mendukung proses pengambilan keputusan, menitik beratkan pada *management by perception*.
2. Adanya tatap muka manusia/mesin di mana manusia (*user*) tetap memegang kendali proses pengambilan keputusan.
3. Mendukung pengambilan keputusan untuk membahas masalah terstruktur, semi terstruktur dan tak struktur.
4. Memiliki kapasitas dialog untuk memperoleh informasi sesuai dengan kebutuhan.
5. Memiliki subsistem-subsistem yang terintegrasi sedemikian rupa sehingga dapat berfungsi sebagai kesatuan item.

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. Intelligenci

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. (Panggabean, 2016 : 2).

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 semi terstruktur 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 system 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 *stand alone* yang digunakan oleh seorang pengambil keputusan pada satu lokasi atau didistribusikan di satu organisasi keseluruhan dan di beberapa organisasi sepanjang rantai persediaan. (Panggabean, 2016 : 2).

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. (Panggabean, 2016 : 3).

II.2.1.5. Multi-Criteria Decision Making (MCDM)

Multi-Criteria Decision Making (MCDM) adalah suatu metode pengambilan keputusan untuk menetapkan alternatif terbaik dari sejumlah alternatif berdasarkan beberapa kriteria tertentu. Kriteria biasanya berupa ukuran-ukuran, aturan-aturan atau standar yang digunakan dalam pengambilan keputusan. Secara umum dapat dikatakan bahwa MCDM menyeleksi alternatif terbaik dari sejumlah alternatif. Terdapat beberapa fitur umum yang digunakan dalam MCDM, yaitu :

1. Alternatif, alternatif adalah obyek-obyek yang berbeda dan memiliki kesempatan yang sama untuk dipilih oleh pengambil keputusan.
2. Atribut, atribut sering juga disebut sebagai kriteria keputusan.
3. Konflik antar kriteria, beberapa kriteria biasanya mempunyai konflik antara satu dengan yang lainnya, misalnya kriteria keuntungan (*benefit*) akan mengalami konflik dengan kriteria biaya (*cost*).
4. Bobot keputusan, bobot keputusan menunjukkan kepentingan relatif dari setiap kriteria, $W = (w_1, w_2, w_3, w_4, \dots, w_n)$.
5. Matriks keputusan, suatu matriks keputusan X yang berukuran $m \times n$, berisi elemen x_{ij} yang merepresentasikan rating dari alternatif A_i ($i = 1, 2, \dots, m$) terhadap kriteria C_j ($j = 1, 2, \dots, n$). (Hasibuan, dkk, 2019 : 463).

II.2.2. Pelajar

Pelajar merupakan aset utama yang menyumbang terhadap pembangunan modal insan dan kesejahteraan negara. Justeru, aktiviti pengajaran dan pembelajaran memberi fokus kepada empat domain utama, yaitu kognitif (pengetahuan), psikomotor (kemahiran fizikal), afektif dan sosial untuk melihat pencapaian hasil pembelajaran pelajar. Bagi membentuk pelajar menjadi insan kamil yang seimbang, domain afektif dan sosial perlu dalam menentukan masa depan sesebuah negara atau masyarakat. Jelaslah bahawa, peranan sistem pendidikan penting dalam menyediakan negara dengan modal insan berkualiti, memiliki sahsiah terpuji, berkemahiran tinggi dalam pelbagai bidang serta mampu berfikir secara kreatif dan kritis khususnya untuk mengisi arus pemodenan negara. (Sidik, dkk, 2020 : 70).

II.2.3. Beasiswa

Beasiswa adalah pembiayaan yang tidak bersumber dari pendanaan sendiri ataupun orang tua, akan tetapi diberikan oleh pemerintah, perusahaan swasta, universitas, serta lembaga pendidikan yang dapat memberikan kesempatan untuk meningkatkan kapasitas sumber daya manusianya melalui pendidikan. Beasiswa tersebut diberikan kepada siswa yang berhak menerima berdasarkan klasifikasi, kualitas, dan kompetensi. Tujuan pemberian beasiswa adalah untuk meringankan beban biaya pendidikan siswa. (Kurniawan, dkk, 2017 : 13).

II.2.4. Sekolah

Sekolah merupakan salah satu lembaga pendidikan formal tempat dimana anak menggali dan mengembangkan potensi yang ada pada dirinya. Dengan adanya pendidikan di sekolah diharapkan siswa tidak hanya menguasai materi pelajaran tetapi sekolah juga dituntut untuk dapat membentuk dan membina kepribadian siswa agar sesuai dengan nilai-nilai dan norma-norma yang berlaku dalam masyarakat. Dalam upaya mengoptimalkan kepribadian peserta didik, maka sekolah perlu menanamkan tata tertib terhadap siswa. Keberadaan tata tertib sekolah berfungsi sebagai pedoman bagi siswa dalam berperilaku di lingkungan sekolah. (Surya dan Wahyu, 2020 : 59).

II.2.5. Metode *Multi-Objective Optimization On The Basic Of Ratio Analysis* (MOORA)

Metode Moora adalah *multiobjectif* sistem yang mengoptimalkan dua atau lebih atribut yang saling bertentangan secara bersamaan. Metode ini diterapkan untuk memecahkan masalah dengan perhitungan matematika yang kompleks. Metode MOORA banyak diaplikasikan dalam bidang seperti bidang manajemen, bangunan, kontraktor, desain jalan, dan ekonomi. Metode ini memiliki tingkat selektifitas yang baik dalam menentukan suatu alternatif. Pendekatan yang dilakukan MOORA didefinisikan sebagai suatu proses secara bersamaan guna mengoptimalkan dua atau lebih yang saling bertentangan pada beberapa kendala. MOORA merupakan pengembangan dari *Multiple Criteria Decision Making* (MCDM) yang menggunakan 1 atau lebih kriteria penilaian dalam pengambilan

keputusan. Didalam metode MOORA terdiri dari dua fase, yaitu fase pendekatan titik referensi dan pendekatan sistem rasio yang memungkinkan pengukuran kriteria baik dan kurang dalam proses memilih alternatif dari serangkaian alternatif.

Metode MOORA memiliki 5 tahapan proses yaitu :

1. Menentukan tujuan, mengidentifikasi atribut dan mengevaluasi atribut tersebut.
2. Menentukan nilai matriks keputusan.
3. Menentukan normalisasi matriks.

Penyebut atau denominator, pilihan terbaik adalah akar kuadrat dari jumlah kuadrat dan setiap alternative per atribut. Maka dirumuskan sebagai berikut :

$$X_{ij} = X_{ij} / \sqrt{\sum_{i=1}^m x_{ij}^2} \dots \dots \dots (1)$$

Keterangan :

X_{ij} : Nilai dari alternatif i pada kriteria j

i : 1, 2, ..., m sebagai banyaknya alternatif

j : 1, 2, ..., m sebagai banyaknya kriteria

X_{i*j} : bilangan tidak berdimensi yang termasuk dalam interval [0, 1] mewakili nilai normalisasi dari alternatif i pada kriteria j.

4. Menentukan optimalisasi atribut.

Untuk optimasi Multi-Objektif, kinerja yang dinormalisasi ditambahkan dalam kasus maksimalisasi (untuk atribut yang menguntungkan) dan dikurangi dalam kasus minimalisasi (untuk atribut non menguntungkan). Saat atribut bobot dimasukkan, maka dapat dirumuskan sebagai berikut :

$$Y_i = \sum_{j=1}^g W_i \cdot X_{ij} - \sum_{j=g+1}^n W_i \cdot X_{ij} \dots \dots \dots (2)$$

Keterangan :

J : 1, 2, ..., g adalah jumlah tipe kriteria yang dimaksimalkan.

I : 1, 2, ..., g+1, g+2, ..., n adalah jumlah tipe kriteria yang diminimalkan.

Y_i : nilai dari penilaian yang telah dinormalisasi dari alternatif i terhadap semua kriteria.

X_{ij} : nilai dari alternatif pada kriteria j.

5. Perangkat nilai Y_i .

Nilai Y_i tergantung dari total maksimal dan minimal dalam matriks keputusan, bias positif atau negatif. Perangkingan nilai Y_i dapat dijadikan hasil keputusan. (Nisa, 2020 : 23).

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).

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

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).

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).

II.2.8. MySQL

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).

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).

II.2.9. Unified Modeling Language (UML)

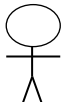
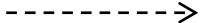

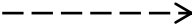
UML yaitu satu kumpulan konvensi permodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem perangkat lunak yang terkait dengan objek. UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan sistem yang besar dan kompleks. UML tidak hanya digunakan dalam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

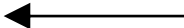


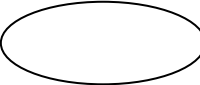

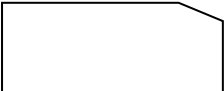
Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut :

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.


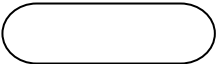



	<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari use case sumber pada suatu titik yang diberikan.
	<i>Association</i>	Apa yang mnghubungkan antara objek satu dengan objek lainnya.
	<i>System</i>	Menspesifikan paket yang menampilkan sistem secara terbatas.
	<i>Use 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 perilaku 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*

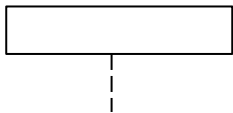

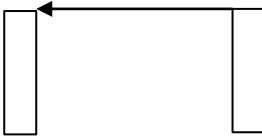
Gambar	Nama	Keterangan
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari aksi.
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Activity Final</i>	Bagaimana objek 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.





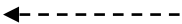
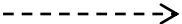
(Sumber : Andikos, 2019 : 39)

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 <i>actor</i>
	<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 memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
_____	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

(Sumber : Andikos, 2019 : 39)

