

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Berikut adalah beberapa jurnal penelitian terdahulu terkait judul penelitian skripsi ini :

Berdasarkan penelitian yang dilakukan oleh *Simamora (2018;500)* mengenai Sistem Pendukung Keputusan Pemilihan Mekanik Terbaik Menerapkan Metode WASPAS , Simamora menyimpulkan bahwa kriteria yang ditetapkan yaitu : Keahlian Servis, Kehadiran, Masa Jabatan dan Tanggung Jawab. Lalu, alternatif yang diberikan ada 4 Lakukan Perhitungan menggunakan WASPAS maka didapat hasil bahwa A3 memiliki perankingan yang lebih tinggi dengan nilai 2,12875 prioritas yang paling tinggi dijadikan mekanik terbaik dengan nama “Ardi”.

Berdasarkan penelitian yang dilakukan oleh *Elvina D Marbun (2018;27)*, dengan judul “Penerapan Metode Weight Aggretd Sum Product Assesment dalam Menentukan Tepung Terbaik Untuk Memproduksi Bihun” penelitian yang dilakukan oleh Elvina D Marbun pada tahun 2018 adalah sistem pendukung keputusan yang dapat memberikan informasi yang ringkas tentang dalam menentukan tepung terbaik untuk produksi bihun agar terciptanya mutu terbaik.

Penerapan dalam Metode Weight Aggretd Sum Product Assesment akan membantu dalam menentukan tepung terbaik untuk produksi bihun.

Berdasarkan penelitian yang dilakukan oleh *Arifin (2018;507-514)* mengenai Sistem Pendukung Keputusan Pemilihan Bibit mangga terunggul menerapkan metode SAW dan WASPAS, Arifin menyimpulkan bahwa kriteria ada 4 yaitu : daun, batang, cabang, bentuk pohon dari kriteria tersebut diberikan nilai bobot dari setiap kriteria setelah itu dihitung menggunakan metode WASPAS maka didapat hasil nilai tertinggi oleh A4 dengan nilai 0,661269 sedangkan perhitungan yang menggunakan metode SAW nilai tertinggi pada A1 dengan nilai 0,925.

Berdasarkan penelitian yang dilakukan oleh *Muhamman Ickhsan (2018;97)* dengan judul “ Sistem Pendukung Keputusan Kredit Usaha Rakyat Menggunakan Metode Weighted Aggretd Sum Product Assesment (WASPAS) “ tujuan penelitian ini adalah untuk menentukan kelayakan nasabah penerima Kredit Usaha Rakyat (KUR), ada beberapa kriteria-kriteria yang harus dipertimbangkan sebelum memberikan KUR kepada nasabah calon penerimaan KUR. Pada penelitian terdahulu sistem pendukung keputusan pemberian kredit usaha rakyat menggunakan metode AHP, dengan menggunakan metode WASPAS ini dapat menggunakan beberapa kriteria-kriteria yang ada diharapkan dapat membantu dalam menentukan kelayakan ppemberian kredit usaya rakyat kepada nasabah.

II.2. Uraian Teoristis

Adapun uraian teoristis sebagai berikut :

II.2.1 Sistem Pendukung Keputusan

Beberapa definisi pengambilan keputusan yang dikemukakan para ahli dijelaskan sebagai berikut:

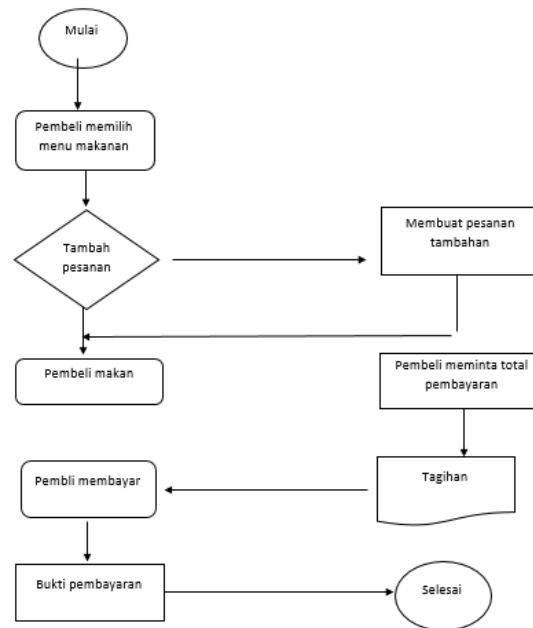
1. Menurut George R. Terry Mendefinisikan pengambilan keputusan adalah pemilihan alternatif perilaku (kelakuan) tertentu dari dua atau lebih alternatif yang ada.
2. Menurut S.P. Siagian mendefinisikan pengambilan keputusan adalah suatu pendekatan yang sistematis terhadap hakikat alternatif yang dihadapi dan mengambil tindakan yang menurut perhitungan merupakan tindakan yang paling tepat.
3. Menurut James A.F.Stoner mendefinisikan pengambilan keputusan adalah proses yang digunakan untuk memilih suatu tindakan sebagai cara pemecahan masalah.

Dari pengertian – pengertian diatas, dapat disimpulkan bahwa pengabilan keputusan merupakan suatu proses pemilihan alternatif terbaik dari beberapa alternatif secara sistematis untuk ditindak lanjuti (digunakan) sebagai suatu cara pemecahan masalah.

Adapun kateristik dari SPK, diantaranya adalah sebagai berikut :

1. Mendukung seluruh kegiatan organisasi
2. Mendukung beberapa keputusan yang saling berinteraksi
3. Dapat digunakan berulang kali dan bersifat konstan

4. Terdapat dua komponen utama, yaitu data dan mode;
5. Menggunakan baik data eksternal maupun internal
6. Menggunakan beberapa model kuantitatif



Gambar 2.1. SOP Rumah Makan Selat Malaka

II.2.2 Rumah Makan Selat Malaka

Rumah makan Selat Malaka adalah salah satu rumah makan seafood di kota Medan khususnya di Belawan. Rumah makan Selat Malaka berdiri sejak tahun 2019 yang didirikan oleh ibu Selly Sartika dan bapak Bagus Irianto. Salah satu tujuan rumah makan tersebut adalah untuk memanjakan lidah para pecinta makanan seafood dengan cita rasa ciri khas dari rumah makan Selat Malaka.

II.2.3. Menu Makanan Rumah Makan Selat Malaka

Seperti rumah makan pada umumnya, rumah makan selat malaka mempunyai berbagai macam menu makanan dan minuman. Dengan cita rasa ciri

khas dan dengan bahan-bahan yang berkualitas dan fresh. Yang membuat para konsumen puas pada menu makanan di rumah makan Selat Malaka. Menu makanan dan minuman pada rumah makan Selat Malaka dapat dilihat pada gambar 2.1.



Gambar 2.2. Menu Makanan Dan Minuman Pada Rumah Makan Selat Malaka

II.2.4. Metode WASPAS (*Weight Aggregated Sum Product*)

Metode *Weigh Aggrestated Sum Product Asesment* (WASPAS) adalah metode yang mencari prioritas pilihan lokasi yang paling sesuai dengan menggunakan cara pembobotan. Penggunaan metode ini merupakan kombinasi dari dua sumber yang dikenal dengan WMM, MCDM approaches dan model produk berat (WPM) pada awalnya memerlukan normalisasi linear dari elemen

hasil. Menggunakan metode WASPAS, kriteria kombinasi paling tertinggi dicari berdasarkan dua kriteria paling tertinggi. Kriteria pertama optimal, kriteria rata-rata keberhasilan sama dengan metode WSM. Pendekatan ini merupakan yang populer dan digunakan MCDM untuk pengambilan keputusan. (Maringan Sianturi, dkk, 2018)

Berikut langkah-langkah Metode WASPAS sebagai berikut, (Maringan Sianturi, dkk, 2018) :

1. Menentukan Normalisasi Matrix dalam Pengambilan Keputusan

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1a} \\ x_{21} & x_{22} & \dots & x_{2a} \\ \dots & \dots & \dots & \dots \\ x_{m1} & x_{m2} & \dots & x_{m3} \end{bmatrix}$$

Jika nilai maksimal dan minimal telah ditentukan maka persamaan sebagai berikut :

Untuk Kriteria benefit :

$$X_{ij} = \frac{x_{ij}}{\max x_{ij}}$$

Untuk Kriteria cost :

$$X_{ij} = \frac{\min x_{ij}}{x_{ij}}$$

2. Menghitung Nilai Normalisasi Matrix dan Bobot WASPAS dalam Pengambilan Keputusan

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

Keterangan :

0.5 = Ketetapan

Q_i = Nilai dari Q ke i

$X_{ij}w$ = Perkalian nilai X_{ij} dengan bobot w

Benefit = Jika nilai terbesar adalah terbaik

Cost = Jika nilai terkecil adalah terbaik

II.2.5. Android 2.2

Menurut (Ahmad, 2015;190-200), Android adalah sebuah *platform* pertama yang betul-betul terbuka dengan pengembangannya dan komperhensif untuk perangkat *mobile*, semua perangkat lunak yang ada difungsikan menjalankan sebuah *device mobile* tanpa memikirkan kendala kepemilikan yang menghambat inovasi pada teknologi *mobile*. Dalam definisi lain, Android merupakan subset perangkat lunak untuk perangkat *mobile* yang meliputi sistem operasi, *middleware*, dan aplikasi inti yang diliris oleh Google. Sedangkan Android *SDK* (*Software Developmen Kit*) menyediakan *tools* dan API yang diperlukan untuk mengembangkan aplikasi pada *platform* Android dengan menggunakan bahasa pemrograman Java.

Aplikasi *Android* ditulis dalam bahasa pemrograman *Java*, yaitu kode java yang terkomplikasi bersama-sama dengan data dan *file-file* sumber yang dibutuhkan oleh aplikasi yang digabungkan oleh *apps tools* menjadi paket aplikasi Android, sebuah file yang di tandai dengan akhiran *.apk*. *file* inilah yang didistribusikan sebagai aplikasi dan diinstal pada *handset Android*. File ini diunduh oleh pengguna ke perangkat *mobile* mereka. Semua kode dijadikan satu

file.apl, dan kemudian kita sebut sebagai sebuah aplikasi (Ahmad : 2015 : 190-200)

II.2.5. Database

1. PHP

PHP adalah sebuah bahasa pemrograman yang berjalan dalam sebuah web-server (*server side*). PHP diciptakan oleh programmer Unix dan Perl yang bernama Rasmus Lerdoft pada bulan Agustus-September 1994. Pada awalnya, Rasmus mencoba menciptakan sebuah *script* dalam *website* pribadinya dengan tujuan untuk memonitor siapa saja yang pernah mengunjungi *website*-nya. PHP dapat berjalan pada semua jenis sistem operasi, antara lain pada Linux dan varian Unix (HP-UX, Solaris dan OpenBSD), pada Ms Windows, Mac dan masih banyak lagi. Selain itu PHP juga dapat berjalan pada beberapa *web-server* antara lain Apache, Microsoft IIS, personal webserver, Netscape dan dan Iplanet Server, Caudium, Xitami, Omnihttpd dan masih banyak lagi. Sistem kerja dari PHP diawali dengan permintaan yang berasal dari halaman *website* oleh *browser*. Berdasarkan URL atau alamat *website* dalam jaringan *internet*, *browser* akan menemukan sebuah alamat dari *webserver*, mengidentifikasi halaman yang dikehendaki, dan menyampaikan segala informasi yang dibutuhkan oleh *webserver*. (Edwina, dkk. 2017;184)

2. MYSQL

MYSQL adalah sistem manajemen *database* SQL yang bersifat *Open Source* dan paling populer saat ini. Sistem *Database* MySQL mendukung beberapa fitur seperti *multithreaded*, *multi-user*, dan *SQL database management system* (DBMS). Database ini dibuat untuk keperluan sistem *database* yang cepat, handal dan mudah digunakan. MySQL dan PHP

merupakan sistem yang saling terintegrasi. Maksudnya adalah pembuatan *database* dengan menggunakan sintak PHP dapat di buat. Sedangkan input yang di masukkan melalui aplikasi web yang menggunakan *script serverside* seperti PHP dapat langsung dimasukkan ke database MySQL yang ada di *server* dan tentunya *web* tersebut berada di sebuah *web server*. (*Edwina, dkk. 2017;184*)

3. Javascript

Javascript adalah bahasa *scripting* yang digunakan untuk membuat aplikasi *web*, sifatnya *client-side* sehingga dapat diolah langsung di *browser* tanpa harus terhubung ke *server* terlebih dahulu. Walaupun namanya menggunakan kata “*Java*”, Javascript tidak berhubungan dengan bahasa pemrograman java, meskipun keduanya memiliki kemiripan dalam hal *syntax* yang meniru bahasa C. (*Edwina, dkk. 2017;184*)

4. Java

Java adalah Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di Sun Microsystems saat ini merupakan bagian dari Oracle dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana serta dukungan rutin-rutin aras bawah yang minimal. Aplikasi-aplikasi berbasis java umumnya dikompilasi ke dalam *p-code (bytecode)* dan dapat dijalankan pada berbagai Mesin Virtual Java (JVM). Java merupakan bahasa pemrograman yang bersifat umum/nonspesifik (*general purpose*), dan secara khusus didisain untuk memanfaatkan dependensi implementasi seminimal mungkin. Karena fungsionalitasnya yang memungkinkan

aplikasi java mampu berjalan di beberapa *platform* sistem operasi yang berbeda, java dikenal pula dengan slogannya, “Tulis sekali, jalankan di mana pun”. Saat ini java merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis *web*. (*Edwina, dkk. 2017;185*)

II.2.7. Basis Data

Untuk mengetahui apa yang dimaksud dari basis data, ada baiknya terlebih dahulu mengetahui apa yang dimaksud dengan data. Data merupakan fakta mengenai obyek, organisasi, dan lain sebagainya. Data dinyatakan dengan nilai (angka, deretan, karakter dan simbol). Dengan demikian basis data adalah kumpulan file yang saling berhubungan satu sama lain yang disimpan dalam media penyimpanan elektronik.

1. Integritas Basis Data

Integritas basis data merupakan salah satu indikator tentang kualitas dari basis data yang dimiliki. Semakin tinggi integritasnya, maka kualitas basis data semakin tinggi. Terdapat beberapa jenis aturan integritas (*integrity constraints*) yang dapat digunakan untuk mengatur definisi dan modifikasi terhadap basis data sehingga menjamin integritas data, yaitu :

a. Entity integrity constraints (aturan integritas entitas)

Aturan integritas entitas merupakan aturan yang diterapkan dengan cara mendeklarasikan *primary key* untuk setiap entitas agar dijamin tidak ada baris-baris yang berelasi memiliki nilai yang sama.

b. Domain constraints (aturan domain)

Domain merupakan kumpulan tipe data dan jangkauan nilai yang diperbolehkan pada atribut sebuah relasi. Definisi domain dari sebuah atribut akan mencakup : tipe, data,

panjang, format, jangkauan, nilai yang memungkinkan, keunikan dan kemungkinan atau *null*.

c. *Referensi integrity constrains* (aturan integritas refensial)

Aturan integritas refensial adalah aturan yang mengatur kebenaran referensi dari suatu obyek ke obyek yang lain dalam basis data. Aturan integritas refensial akan menjamin bahwa setiap kolom kunci tamu dari tabel yang merujuk harus tepat sama dengan nilai dalam kolom kunci primer dari tabel yang ditunjuk, atau jika tidak akan bernilai *null*.

d. *Assertions* (pernyataan)

Merupakan sebuah aturan yang berlaku pada saat operasi *insert/update/delete* dan dapat melibatkan beberapa tabel yang bertujuan untuk membuat *database* tetap pada kondisi yang diinginkan.

e. *Trigger* (pemicu)

Trigger adalah aturan yang akan mengeksekusi perintah secara otomatis sebagai akibat sampingan dari proses modifikasi dalam *database*.

2. Normalisasi

Normalisasi merupakan cara untuk melakukan pendekatan dalam membangun desain logika basis data relational yang secara tidak langsung berkaitan dengan model data, tetapi dengan menggunakan sejumlah aturan atau kriteria standart untuk menghasilkan struktur tabel yang normal. Pada dasarnya desain logika basis data relasional dapat menggunakan prinsip normalisasi maupun transformasi dari model E-R ke bentuk fisik.

Dalam perspektif normalisasi sebuah basis data dikatakan baik jika setiap tabel yang membentuk basis data sudah berada dalam keadaan normal.

Berikut adalah bentuk-bentuk normalisasi :

a. Bentuk tidak normal

Bentuk ini merupakan kumpulan data yang akan direkam, tidak ada keharusan mengikuti format tertentu, dapat saja tidak lengkap dan terduplikasi. Data dikumpulkan apa adanya sesuai dengan keadaannya.

b. Bentuk normal tahap pertama (1NF)

Bentuk normal tahap pertama (1NF) terpenuhi jika tidak ada *record* yang duplikat dan tidak memiliki atribut berniali banyak dalam tabel tersebut

c. Bentuk normal tahan kedua (2NF)

Bentuk normal kedua (2NF) terpenuhi jika tidak pada sebuah tabel semua atribut yang tidak termasuk dalam *primary key* memiliki ketergantungan fungsional pada *primary key* secara utuh

d. Bentuk normal tahap ketiga (3NF)

Sebuah tabel dikatakan memenuhi bentuk normal tahap ketiga (3NF), jika untuk setiap ketergantungan fungsional dengan notasi $X \rightarrow A$, dimana A mewakili semua atribut tunggal di dalam tabel yang tidak ada di dalam X, maka :

- X haruslah *superkey* pada semua tabel tersebut
- Atau A merupakan bagian dari *primary key* pada tabel tersebut.

e. Bentuk normal tahap keempat dan kelima

Penerapan aturan normalisasi sampai bentuk ketiga sudah memadai untuk menghasilkan tabel kualitas baik. Namun demikian, terdapat pula bentuk norma keempat (4NF) dan kelima (5NF). Bentuk normal keempat berkaitan dengan sifat ketergantungan banyak nilai (*multivalued dependency*) pada suatu tabel yang merupakan pengembangan dari

ketergantungan fungsional. Adapun bentuk normal kelima merupakan nama lain dari *Project Join Normal Form* (PJNF).

- f. Boyce Code Normal Form (BCNF)
 - Memenuhi 1NF
 - Relasi harus bergantung fungsi pada atribut *super key*.

II.2.8 Eclipse IDE

Menurut (Lengkong, dkk, 2015; 21), *Eclipse* adalah sebuah IDE (*Integreted Development Enviroment*) untuk mengembangkan perangkat lunak dan dapat dijalankan semua *platform* (*platform independent*). *Eclipse* pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan *open source* , yang berarti setiap orang boleh melihat kode pemograman perangkat lunak ini. selain itu, kelebihan dari *eclipse* yang membuat populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan kompenen yang dinamakan *plug-in*.

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

UML (*Unified Modeling Language*) 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;108-110).

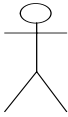
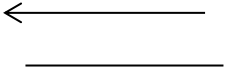
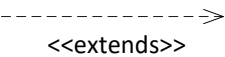
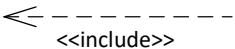
Adapun bagian-bagian dari UML (*Unified Modeling Language*) adalah sebagai berikut :

a. Use Case Diagram

Use Case Diagram merupakan pemodelan untuk kelakuan (Behavior) 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;108-109).

Tabel II.1. Simbol Use Case Diagram

Gambar	Nama	Keterangan
	<i>Use Case</i>	<i>Use Case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, yang dinyatakan dengan menggunakan kata kerja.
	<i>Actor</i>	<i>Actor</i> adalah <i>Abstraction</i> dari orang atau 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	Asosiasi 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 data.
	<i>Extends</i>	<i>Extends</i> merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.
	<i>Include</i>	<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

(Sumber : Ade Hendini, 2016;108-109)

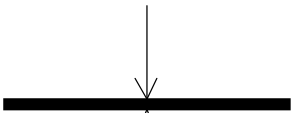
b. Activity Diagram

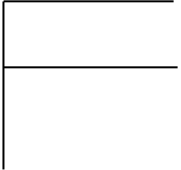
Activity Diagram adalah diagram yang menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang

dilakukan oleh aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Jadi dengan kata lain, diagram ini menunjukkan bagaimana aktivitas-aktivitas tersebut bergantung satu sama lain. *Activity Diagram* menggambarkan *Work Flow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis (Ade Hendini, 2016;109-110).

Simbol-simbol yang digunakan dalam *Activity Diagram* dapat dilihat pada Tabel II.2.

Tabel II.2. Simbol Activity Diagram

Gambar	Nama	Keterangan
	<i>Initial State / Start State</i>	<i>Start Point</i> diletakkan pada pojok kiri atas dan merupakan awal aktivitas.
	<i>Action State</i>	menggambarkan suatu proses atau kegiatan bisnis.
	<i>Transition</i>	digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	<i>Control Flow</i>	digunakan untuk menunjukkan kegiatan.
	<i>Decision</i>	Menggambarkan pilihan untuk pengambilan keputusan <i>true</i> atau <i>false</i> .
	<i>Final State</i>	Digunakan untuk mengakhiri aktivitas.

	Swimlane	Merupakan pembagian <i>Activity Diagram</i> untuk menunjukkan siapa melakukan apa.
---	----------	--

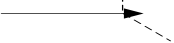
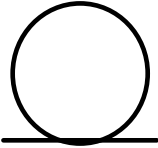
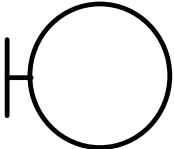
(Sumber : Ade Hendini, 2016;108-109)

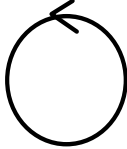
c. Sequence Diagram

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek (Ade Hendini, 2016;110).

Simbol-simbol yang digunakan dalam *Sequence Diagram* dapat dilihat pada Tabel II.3.

Tabel II.3. Simbol Sequence Diagram

Gambar	Nama	Keterangan
	<i>Activation</i>	Mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivasi sebuah operasi.
	<i>Lifeline</i>	garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .
	<i>Message</i>	simbol mengirim pesan antar <i>class</i> .
	<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>	Merupakan berisi kumpulan kelas yang menjadi <i>interfaces</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan <i>form entry</i> dan <i>form cetak</i>

	<i>Control class</i>	Merupakan suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas.
---	----------------------	--

(Sumber : Ade Hendini, 2016;110)

d. Class Diagram

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.

Simbol-simbol yang digunakan dalam *Class Diagram* dapat dilihat pada Tabel II.4.

Tabel II.4. Simbol 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 actor
	<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.

	<i>Dependency</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