

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terdahulu

Berdasarkan penelitian yang dilakukan oleh Fadhilah, dkk (2021) mengenai Implementasi Sistem Pakar Mendiagnosa Penyakit Penyakit *Gastritis* Dengan Menggunakan Metode *Teorema Bayes*, penelitian Fadhilah, dkk menyimpulkan bahwa proses mendiagnosa penyakit *gastiris* pada lambung manusia menggunakan metode *teorema bayes* diawali dengan proses penentuan penyakit dan gejala selanjutnya dilakukan proses perhitungan dengan memilih gejala-gejala yang dialami sehingga didapatkan nilai kesimpulan dari setiap penyakit untuk ditentukan yang terpilih berdasarkan nilai tertinggi.

Berdasarkan penelitian yang dilakukan oleh Febriani, dkk (2021) mengenai Penerapan Metode *Teorema Bayes* untuk Diagnosa Penyakit Telinga Hidung dan Tenggorokan (THT), penelitian Febriani, dkk menyimpulkan bahwa penerapan Metode *Teorema Bayes* untuk Diagnosa Penyakit Telinga Hidung dan Tenggorokan (THT) dengan metode pengujian sistem menggunakan *black-box testing* telah berjalan dengan baik dan bekerja sesuai dengan rancangan.

Berdasarkan penelitian yang dilakukan oleh Gea, dkk (2021) mengenai Sistem Pakar Mendiagnosa Penyakit *Anemia* Dengan Menggunakan Metode *Teorema Bayes* Berbasis *Web*, penelitian Gea, dkk menyimpulkan bahwa Sistem pakar mendiagnosa penyakit *Anemia* dapat mengetahui jenis penyakit *anemia*

berdasarkan gejala penyakit *anemia* yang diderita dengan menggunakan metode *Teorema Bayes*.

Berdasarkan penelitian yang dilakukan oleh Kudadiri (2021) mengenai Sistem Pakar Diagnosa Penyakit Tanaman Tomat yang Disebabkan Abu Vulkanik Sinabung Menerapkan Metode *Teorema Bayes*, penelitian Kudadiri menyimpulkan bahwa pendiagnosaan penyakit tanaman pada buah tomat yang disebabkan abu vulkanik sinabung dengan menggunakan *teorema bayes*. Metode *bayes* dapat diterapkan untuk mendiagnosa penyakit pada tanaman tomat dapat menghasilkan perhitungan manual sehingga proses diagnosa dapat dilakukan dengan cepat dan akurat.

Berdasarkan penelitian yang dilakukan oleh Rahmadsyah (2021) mengenai Sistem Pakar Diagnosa Penyakit Kulit Akibat Infeksi Jamur Menggunakan *Teorema Bayes*, penelitian Rahmadsyah menyimpulkan bahwa sistem yang telah dikembangkan menggunakan implementasi *teorema bayes* dapat digunakan sebagai alat bantu diagnosis awal penyakit kulit akibat infeksi jamur. Dari 31 data pasien yang telah dibandingkan antara sistem dan pakar, didapatkan 27 data pasien yang sesuai dan 4 data pasien yang tidak sesuai. Sehingga tingkat kinerja sistem berdasarkan hasil validasi tersebut, diperoleh persentase 87,1% data pasien yang sesuai, serta 12,9% data pasien yang tidak sesuai.

Berdasarkan penelitian yang dilakukan oleh Sagat dan Purnomo (2021) mengenai Sistem Pakar Diagnosa Penyakit Mata Menggunakan Metode *Teorema Bayes*, penelitian Sagat dan Purnomo menyimpulkan bahwa sistem pakar dengan metode *teorema bayes* yang telah dibuat dapat diimplementasikan dalam

membantu keperluan medis dalam mendiagnosa penyakit mata yaitu penyakit *hordeolum*, mata kering, *episkleritis*, *pterygium*, dan *katarak*. Dengan hasil validasi kesesuaian antara hasil pakar dan hasil sistem sebesar 96%.

II.2. Landasan Teori

Landasan teori dikutip dari beberapa jurnal penelitian terdahulu yang berkaitan dengan penelitian ini.

II.2.1. Sistem Pakar

Sistem pakar adalah suatu aplikasi komputer yang ditujukan untuk membantu pengambilan keputusan atau pemecahan persoalan dalam bidang yang spesifik. Sistem ini bekerja dengan menggunakan pengetahuan dan metode analisis yang telah didefinisikan terlebih dahulu oleh pakar yang sesuai dengan bidang keahliannya. (Febriani, dkk, 2021:329).

Sistem Pakar (*Expert System*) adalah salah satu cabang dari AI yang membuat pengguna secara luas *knowledge* yang khusus untuk penyelesaian masalah tingkat manusia yang pakar. Seorang pakar adalah orang yang mempunyai keahlian dalam bidang tertentu yaitu pakar yang mempunyai *knowledge* atau kemampuan khusus yang orang lain tidak mengetahui atau mampu dalam bidang yang dia miliki. (Kudadiri dan Hasibuan, 2021:194).

II.2.2. Diagnosa

Diagnosa merupakan tahapan dan hasil dari diagnosis suatu penyakit yang diderita oleh penderita. Menurut Kamus Besar Bahasa Indonesia (KBBI),

Diagnosis adalah penentuan jenis penyakit dengan cara memeriksa atau meneliti gejala-gejalanya. Secara umum diagnosa adalah upaya untuk mengetahui atau mengidentifikasi suatu penyakit atau masalah kesehatan yang diderita atau dialami seseorang pasien atau penderita. Diagnosis adalah menentukan sebab malfungsi dalam situasi kompleks yang didasarkan pada gejala-gejala yang teramati, diantaranya medis, elektronis, mekanis, dan diagnosis perangkat lunak. (Ayudia, 2020:383).

II.2.3. Penyakit

Penyakit adalah suatu keadaan tidak normal dari tubuh maupun psikis yang sifatnya objektif karena masing-masing memiliki parameter tertentu. Penyakit adalah istilah medis yang digambarkan sebagai gangguan dalam fungsi tubuh yang menghasilkan berkurangnya kapasitas. Penyakit terjadi ketika keseimbangan dalam tubuh tidak dapat dipertahankan. Keadaan sakit terjadi pada saat seseorang tidak lagi berada dalam kondisi sehat yang normal. (Sukorini, 2017:9).

II.2.4. Chicken Guinea

Chicken Guinea atau yang biasa disebut *Chikungunya* adalah salah satu penyakit yang ditularkan melalui perantara nyamuk yaitu nyamuk *Aedes*. Seluruh wilayah Indonesia pernah melaporkan kasus *Chikungunya*. Jumlah kasus penyakit ini pada tahun 2019 adalah 5.042 kasus. Gejala penyakit yang ringan menyebabkan banyak penderita yang tidak mencari pengobatan sehingga tidak

terdata. Penyakit ini disebabkan oleh infeksi virus *Chikungunya*, virus RNA yang masuk ke dalam *Famili Togaviridae*. Gejala *Chikungunya* bervariasi dari tidak bergejala sampai timbul gejala 2-12 hari setelah tergigit nyamuk *Aedes aegypti* dan *Ae.albopictus* yang terinfeksi. Gejala berupa demam, nyeri otot dan sendi yang parah. Terkadang dapat disertai dengan sakit kepala, mual, muntah, dan kemerahan. Pada umumnya dapat sembuh sempurna namun beberapa penderita masih mengalami nyeri sendi yang dapat berlangsung kronis dan mengalami komplikasi pada mata, saluran cerna, saraf, dan jantung. Belum ada vaksin maupun obat antiviral untuk penyakit ini, penanganan bersifat suportif. Wisatawan diminta untuk melindungi diri dari gigitan nyamuk *Aedes* di sepanjang hari dari pagi hingga sore. Selain Indonesia, wisatawan yang berkunjung ke Kepulauan Karibia, Amerika Tengah, Amerika Selatan, SubSahara Afrika, Asia Tenggara, dan Subbenua India juga berisiko terhadap *Chikungunya*. (Damayanti dan Yanti, 2020:235).

II.2.5. Metode Teorema Bayes

Metode *Teorema bayes* dikemukakan oleh seorang pendeta Presbyterian Inggris pada tahun 1763 yang bernama Thomas Bayes ini kemudian disempurnakan Laplace. *Teorema bayes* digunakan untuk menghitung probabilitas terjadinya suatu peristiwa berdasarkan pengaruh yang didapat dari hasil observasi. Disamping ini metode *bayes* memanfaatkan data sampel yang diperoleh dari populasi juga memperhitungkan suatu distribusi awal yang disebut distribusi prior. (Fadhilah, dkk, 2021:2).

Teori keputusan *Bayes* merupakan pendekatan statistik yang fundamental dalam pengenalan pola. Pendekatan ini didasarkan kuantifikasi *trade-off* antara berbagai keputusan klasifikasi dengan menggunakan peluang dan ongkos yang ditimbulkan dalam keputusan-keputusan tersebut. *Teorema Bayes* adalah sebuah teori kondisi peluang yang memperhitungkan peluang sebuah kejadian (hipotesis) yang bergantung pada kejadian lain (bukti). Metode *Teorema Bayes* merupakan teknik yang digunakan untuk melakukan analisis dalam pengambilan keputusan terbaik dari sejumlah alternatif. *Teorema Bayes* juga merupakan metode yang baik didalam mesin inferensi berdasarkan data pelatihan, dengan menggunakan peluang bersyarat sebagai dasarnya. dengan rumusan seperti pada persamaan 1.

$$P(H|E) = \frac{P(E|H) \cdot P(H)}{P(E)} \dots\dots\dots(1)$$

Dimana:

$P(H|E)$: peluang dari hipotesis H_i jika diberikan bukti E .

$P(E|H)$: peluang dari bukti E , jika hipotesis H_i benar.

$P(H)$: peluang dari hipotesis H_i (berdasarkan hasil sebelumnya).

$P(E)$: peluang bukti E . (Rahmadsyah dan Akbar, 2021:39).

Studi Kasus:

Dalam sistem pakar penyakit tomat coba didiagnosa dengan kriteria-kriteria yang berdasarkan data yang ada. Adapun kriteria penyakit tanaman tomat sebagai berikut:

Tabel II.1. Gejala Penyakit Tanaman Tomat

Kode	Gejala	Penyakit Layu Fusarium	Penyakit Layu Bakteri
G1	Mula-mula helaian daun bagian-bawah melipat dan menggulung		☐
G2	Daun mengalami perubahan warna menjadi kuning dan layu	☐	☐
G3	Bercak pada daun berukuran 3-5 mm yang terdapat bintik-bintik hitam	☐	☐
G4	Pinggir daun terlihat membusuk dan basah	-	☐
G5	Berubahnya warna rimpang dan berbau tidak sedap	☐	☐
G6	Rimpang menjadi busuk	☐	☐

Berdasarkan tabel II.1 maka dipresentasikan seperti *rule* dibawah ini Refresentasi ini dimanfaatkan untuk mendapatkan pencarian dan penentu kesimpulan diagnosa. Dibawah ini adalah pembahasannya:

Rule 1: IF G1 = True AND G2 = True AND G3 = True AND G4 = True AND G5 = True AND G6 = True, THEN P = Layu Bakteri

Rule 2: IF G2 = True AND G3 = True AND G5 = True AND G6 = True, THEN P = Layu Fusarium

Berdasarkan tabel II.1. yaitu tabel gejala maka didapat nilai hipotesa dan dapat dilihat pada tabel dibawah sebagai berikut:

Tabel II.2. Nilai Hipotesa

Kode	Gejala	Penyakit Layu Fusarium	Penyakit Layu Bakteri
G1	Mula-mula helaian daun bagian bawah menggulung	0	0.2
G2	Daun mengalami perubahan warna menjadi kuning dan layu	0.4	0.2

G3	Bercak pada daun berukuran 3-5 mm yang terdapat bintik-bintik hitam	0.2	0.4
G4	Pinggir daun terlihat membusuk dan basah	0	0.4
G5	Berubahnya warna rimpang dan berbau tidak sedap	0.6	0.6
G6	Rimpang menjadi busuk	0.8	0.8

Adapun analisa terhadap sistem pakar yang dibangun merupakan *rule* yang menerapkan metode *teorema bayes*. Berdasarkan tabel II.2. dapat dilihat gejala dari penyakit tanaman tomat, maka setelah itu penulis memasukkan bobot atau terminologi kepastian dari gejala yang ada. Terminologinya dapat dilihat dibawah ini:

Tabel II.3. Terminologi Kepastian	
Terminologi Kepastian	A/B
Pasti	1.0
Hampir Pasti	0.8
Cukup Pasti	0.6
Kurang Pasti	0.4
Tidak Tahu	0.2
Tidak Pasti	0

Contoh soal: para petani tomat akhir-akhir ini mengeluh akibat serangan penyakit dari abu vulkanik sinabung terhadap tanaman tomat mereka. Akan tetapi mereka tidak tahu penyakit apa yang disebabkan abu vulkanik para petani melakukan diagnosa terhadap gejala yang menyerang tanaman tomat sebagai berikut:

$$G1 = E1 = 0.2 \text{ P}(E|H1) \quad G2 = E2 = 0.2 \text{ P}(E|H2) \quad G3 = E3 = 0.4 \text{ P}(E|H3) \quad G4 = E4 =$$

$$0.4 P(E|H4)G5 = E5 = 0.6 P(E|H5)G6 = E6 = 0.8 P(E|H6)$$

Berdasarkan nilai *evidence* diatas maka didapat hasil:

$$\begin{aligned} P[H]1 &= \frac{H1}{H1 + H2 + H3 + H4 + H5 + H6} \\ &= \frac{0}{0 + 0.4 + 0.2 + 0 + 0.6 + 0.8} \\ &= \frac{0}{2} \\ &= 0 \end{aligned}$$

$$\begin{aligned} P[H]2 &= \frac{H2}{H1 + H2 + H3 + H4 + H5 + H6} \\ &= \frac{0.4}{0 + 0.4 + 0.2 + 0 + 0.6 + 0.8} \\ &= \frac{0.4}{2} \\ &= 0.2 \end{aligned}$$

$$\begin{aligned} P[H]3 &= \frac{H3}{H1 + H2 + H3 + H4 + H5 + H6} \\ &= \frac{0.4}{0 + 0.4 + 0.2 + 0 + 0.6 + 0.8} \\ &= \frac{0.4}{2} \\ &= 0.2 \end{aligned}$$

$$\begin{aligned} P[H]4 &= \frac{H4}{H1 + H2 + H3 + H4 + H5 + H6} \\ &= \frac{0}{0 + 0.4 + 0.2 + 0 + 0.6 + 0.8} \\ &= \frac{0}{2} \\ &= 0 \end{aligned}$$

$$\begin{aligned} P[H]5 &= \frac{H5}{H1 + H2 + H3 + H4 + H5 + H6} \\ &= \frac{0.6}{0 + 0.4 + 0.2 + 0 + 0.6 + 0.8} \\ &= \frac{0.6}{2} \\ &= 0.3 \end{aligned}$$

$$\begin{aligned}
 P[H]6 &= \frac{H6}{H1 + H2 + H3 + H4 + H5 + H6} \\
 &= \frac{0.8}{0 + 0.4 + 0.2 + 0 + 0.6 + 0.8} \\
 &= \frac{0.8}{2} \\
 &= 0.4
 \end{aligned}$$

Setelah nilai $P(H_i)$ diketahui, probabilitas hipotesis H tanpa memandang *Evidence* apapun, maka langkahselanjutnya adalah:

$$\begin{aligned}
 P[H|E] i &= \frac{P[E|H]i * P[H]i}{\sum_{n=1}^n P[E|H]i * P[H]i} \\
 &= \frac{P[E|H]i * P[H]i}{P[E|H]1 * P[H]1 + P[E|H]2 * P[H]2 + n} \\
 &= P(H1) * P(E|H1) + P(H2) * P(E|H2) + P(H3) * P(E|H3) + P(H4) * P(E|H4) + \\
 &P(H5) * P(E|H5) + P(H6) * P(E|H6) \\
 &= (0.2 * 0) + (0.2 * 0.2) + (0.4 * 0.1) + (0.4 * 0) + (0.6 * 0.3) + (0.8 * 0.4) \\
 &= 0 + 0.04 + 0.04 + 0 + 0.18 + 0.32 \\
 &= 0.58
 \end{aligned}$$

Setelah seluruh nilai $P(H_i|E)$ diketahui, maka jumlahkan seluruh nilai bayesnya dengan rumus sebagai berikut:

$$\begin{aligned}
 \sum Bayes &= Bayes1 * H1 + Bayes2 * H2 + Bayes3 * H3 + Bayes4 * H4 + \\
 &Bayes5 * H5 + Bayes6 * H6 \\
 k=1 \\
 &= 0*0 + 0.068*0.4 + 0.068*0.2 + 0*0 + 0.310*0.6 + 0.551*0.8 \\
 &= 0 + 0.027 + 0.014 + 0 + 0.186 + 0.441 \\
 &= 0.668
 \end{aligned}$$

Berdasarkan kesimpulan perhitungan diatas maka dapat dipastikan tanaman tomat diprediksi terserang penyakit layu fusarium dengan nilai 66.8%.

II.2.6. Web

Situs *Web* merupakan salah satu media yang dapat digunakan untuk memberikan layanan informasi. *Web Browser* adalah aplikasi perangkat lunak yang digunakan untuk mengambil dan menyajikan sumber informasi *Web*. Sumber informasi *Web* diidentifikasi dengan *Uniform Resource Identifier* (URL) yang dapat terdiri dari halaman *Web*, *video*, gambar, ataupun konten lainnya. (Pakpahan dan Halawa, 2020:111).

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. 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.2.9. My Structure Query Language (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.2.8. Normalisasi

Normalisasi merupakan salah satu teknik dalam desain basis data untuk memproduksi tabel-tabel yang memiliki atribut yang tidak tepat sesuai dengan kebutuhan sistem. Ciri-ciri tabel yang ternormalisasi adalah:

1. Jumlah atribut bersifat minimal, sesuai dengan kebutuhan data.
2. Atribut yang berhubungan dekat (disebut *functional dependency*) ditempatkan dalam tabel yang sama.
3. Redundansi (kemunculan duplikasi data yang tidak diharapkan) bersifat minimal dengan memanfaatkan *foreign key*.

Manfaat yang diperoleh dari hasil proses normalisasi ini adalah basis data menjadi mudah diakses, data mudah dikelola, dan meminimalkan tempat penyimpanannya. Manfaat normalisasi adalah sebagai berikut:

1. Normalisasi memastikan bahwa setiap kolom milik tabel yang tepat yang mana telah ditetapkan dan bukan tabel lain.
2. Normalisasi menghilangkan penyimpanan informasi yang redundan, ini menyederhanakan logika aplikasi, karena *developer* basis data tidak perlu memikirkan mengenai beberapa salinan bagian informasi yang sama.
3. Normalisasi memastikan bahwa basis data memiliki satu kolom di satu tempat, dengan satu nama, dengan satu nilai, pada satu waktu. Proses normalisasi disebutkan terdapat beberapa tahapan dalam normalisasi dimulai dari tahap paling ringan (1NF) hingga paling ketat (5NF). Biasanya hanya sampai pada tingkat 3NF atau BCNF karena sudah cukup memadai untuk menghasilkan tabel-tabel yang berkualitas baik. (Sujatmoko, dkk, 2020:125).

II.2.9. Unified Modeling Language (UML)

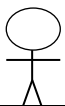
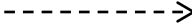
UML yaitu satu kumpulan konvensi permodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem perangkat lunak yang terkait dengan objek. UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan sistem yang besar dan kompleks. UML tidak hanya digunakan dalam 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.4.

Tabel II.4. 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> .

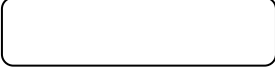
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

(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.5.

Tabel II.5. Simbol *Activity Diagram*

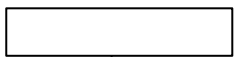
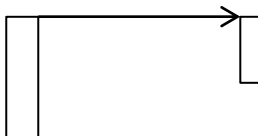
Gambar	Nama	Keterangan
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Action</i>	<i>State</i> dari sistem yang mencerminkan eksekusi dari suatu aksi.
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Activity Final</i>	Bagaimana 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.6.

Tabel II.6. 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.

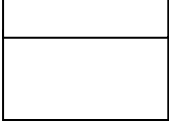
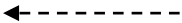
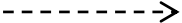
(Sumber:Andikos, 2019:39)

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)