

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Yuliyana dan dkk, (2019) dengan judul “Sistem Pakar Diagnosa Penyakit Gigi Menggunakan Metode *Naive Bayes*” Diagnosis penyakit gigi dengan metode *Naive Bayes* dilakukan beberapa tahapan user melakukan *input* fakta gejala penyakit. Yang membedakan penelitian yang ada sistem pakar dengan menerapkan metode *naïve bayes*, sistem akan menghitung *probabilitas prior*, *likelihood*, dan *posterior*. Nilai terbesar dari perhitungan *posterior* akan dijadikan hasil diagnosa.

Achmad Syarifudin dan dkk, (2018) dengan judul “Sistem Pakar Diagnosis Penyakit Pada Tanaman Jagung Menggunakan Metode *Naive Bayes* Berbasis *Android*”. Yang membedakan penelitian yang ada sistem pakar dengan menerapkan metode *naïve bayes*. Sistem pakar pada penelitian ini menghasilkan tingkat akurasi sebesar 96%.

Yunia Ervinaeni dan dkk, (2019) dengan judul “Sistem Pakar Diagnosa Gangguan Hiperaktif Pada Anak Dengan Metode *Naive Bayes* Berbasis *Web*”. Sistem pakar diagnosa gangguan hiperaktif menggunakan metode *naive bayes* dapat memberikan hasil berupa presentase angka. Yang membedakan penelitian yang ada sistem pakar dengan menerapkan metode *naïve bayes*. Sistem pakar dengan menentukan hiperaktif pada anak dapat dibuat sebuah aplikasi berbasis *web* dan

menggunakan metode *naive bayes* dalam mendiagnosa seorang anak yang menderita gangguan hiperaktif.

Berdasarkan penelitian terdahulu, maka dilakukan penelitian selanjutnya yaitu “Sistem Pakar Mendiagnosa Penyakit *Fibroadenoma Mammae (FAM)* Menggunakan Metode *Naïve Bayes* Studi Kasus : RSUD. Delima”. Hasil penelitian ini adalah untuk mengetahui hasil metode *naive bayes* dalam mendiagnosa penyakit *Fibroadenoma Mammae (FAM)*. Sehingga bisa menjadi acuan atau tolak ukur para tenaga medis dalam hal meningkatkan nilai kepercayaan dengan pengujian metode untuk memperoleh nilai akurasi terhadap *Fibroadenoma Mammae (FAM)*.

II.2. Sistem

Sistem umumnya diartikan sebagai satu kesatuan yang utuh. Menurut Mulyanto (2009), sistem adalah kumpulan dari sub-sub *system* baik *abstrak* maupun fisik yang saling terintegrasi dan saling berkolaborasi untuk mencapai suatu tujuan tertentu. Al Fatta (2013) mendefinisikan sistem sebagai sekelompok elemen-elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan. Secara garis besar, sebuah sistem informasi terdiri atas tiga komponen utama. Ketiga komponen tersebut mencakup *software*, dan *brainware*. Ketiga komponen ini saling berkaitan satu sama lain. Dari definisi tersebut dapat disimpulkan bahwa sistem adalah kumpulan dari elemen-elemen yang saling berinteraksi dan saling berhubungan satu sama lainnya untuk mencapai suatu tujuan. (Muhdar Abdurahman: 2018)

II.3. Informasi

Informasi merupakan data atau fakta yang telah diproses sedemikian rupa, sehingga berubah bentuknya menjadi informasi. Di samping itu informasi dapat mengurangi ketidakpastian serta mempunyai nilai dalam keputusan karena dengan adanya informasi kita dapat memilih tindakan-tindakan dengan resiko yang paling kecil. Untuk menghasilkan kebijaksanaan dan keputusan yang baik diperlukan pengolahan data menjadi informasi yang relevan dengan masalah perusahaan yang sedang dihadapi. Dengan demikian data itu merupakan bahan mentah yang harus diproses lebih dahulu baru kemudian dapat digunakan.

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. Informasi juga disebut data yang diproses atau data yang memiliki arti. Informasi merupakan data yang telah diproses sedemikian rupa sehingga meningkatkan pengetahuann seseorang yang menggunakan. Para pembuat keputusan memahami bahawa informasi menjadi faktor kritis. Informasi dapat berupa data mentah, data tersusun, kapasitas, sebuah saluran informasi dan sebagainya. (Astia Firman dan dkk, 2016 : 29).

II.4. Sistem Informasi

Sistem Informasi adalah serangkaian *hardware*, *software*, data, manusia, dan prosedur yang bekerja bersama untuk memproduksi informasi. Prosedur adalah suatu instruksi atau serangkaian instruksi, yang diikuti oleh *user* untuk menyelesaikan kegiatannya. Kegunaan umum dari sistem informasi termasuk kategori dari sistem

informasi yang dapat digunakan oleh hampir semua bagian di dalam perusahaan. Sistem informasi terintegrasi digunakan oleh banyak bagian dan memfasilitasi berbagi informasi (*information Sharing*) dan komunikasi dalam perusahaan (Mas Ayoe Elhias, 2016 : 3)

II.5. Sistem Pakar

Sistem Pakar (*Expert System*) merupakan solusi AI bagi masalah pemrograman pintar (*Intelligent*). Profesor Edward Feigenbaum dari *Stanford University* yang merupakan *pionir* dalam teknologi sistem pakar mendefinisikan Sistem Pakar sebagai sebuah program komputer pintar (*intelligent computer program*) yang memanfaatkan pengetahuan (*knowledge*) dan prosedur inferensi (*inference procedure*) untuk memecahkan masalah yang cukup sulit hingga membutuhkan keahlian khusus dari manusia.

Dengan kata lain, Sistem Pakar adalah sistem komputer yang ditujukan untuk meniru semua aspek (*emulates*) kemampuan pengambilan keputusan (*decision making*) seorang pakar. Sistem Pakar memanfaatkan secara maksimal pengetahuan khusus layaknya seorang pakar untuk memecahkan masalah.

Pakar atau ahli (*expert*) didefinisikan sebagai seseorang yang memiliki pengetahuan atau keahlian khusus yang tidak dimiliki oleh kebanyakan orang. Seorang pakar dapat memecahkan masalah yang tidak mampu dipecahkan kebanyakan orang. Dengan kata lain, dapat memecahkan suatu masalah dengan lebih efisien namun bukan berarti lebih mudah. Pengetahuan yang dimuat ke dalam Sistem

Pakar dapat berasal dari seorang pakar atau pun pengetahuan yang berasal dari buku, jurnal, majalah, dan dokumentasi yang dipublikasikan lainnya, serta orang yang memiliki pengetahuan meskipun bukan ahli. Istilah Sistem Pakar (*expert system*). Sering disinonimkan dengan sistem berbasis pengetahuan (*knowledge-based system*) atau Sistem Pakar berbasis pengetahuan (*knowledge based expert system*). (B. Herawan Hayadi, 2018:2).

II.5.1. Manfaat Sistem Pakar

Sistem pakar sangat populer karena sangat banyak kemampuan dan manfaat yang diberikannya, di antaranya :

1. Meningkatnya produktivitas, karena sistem pakar dapat bekerja lebih cepat dari pada manusia.
2. Membuat seseorang yang awam bekerja seperti layaknya seorang pakar.
3. Meningkatkan kualitas, dengan memberi nasehat yang konsisten dan mengurangi kesalahan.
4. Mampu menangkap pengetahuan dan kepakaran seseorang.
5. Memudahkan akses pengetahuan seorang pakar.
6. Bisa digunakan sebagai media pelengkap dalam pelatihan. Pengguna pemula yang bekerja dengan sistem pakar akan menjadi lebih berpengalaman karena adanya fasilitas penjelas yang berfungsi sebagai guru.

7. Meningkatkan kemampuan untuk menyelesaikan masalah karena sistem pakar mengambil sumber pengetahuan dari banyak pakar. (B. Herawan Hayadi, 2018:2).

II.5.2. Ciri –Ciri Sistem Pakar

Menurut B. Herawan Hayadi (2018), ciri–ciri sistem pakar adalah sebagai berikut :

1. Terbatas pada domain keahlian tertentu
2. Dapat memberikan penalaran untuk data yang tidak pasti
3. Dapat mengemukakan rangkaian alasan yang diberikannya dengan cara yang dapat dipahami.
4. Berdasarkan pada kaidah atau *rule* tertentu.
5. Dirancang untuk dapat dikembangkan secara bertahap.
6. Pengetahuan dan mekanisme *inferensi* jelas terpisah.
7. Keluarannya bersifat anjuran
8. Sistem dapat mengaktifkan kaidah secara searah oleh pemakai

II.5.3.Konsep Umum Sistem Pakar

Sistem pakar terdiri dari beberapa konsep yang harus dimiliki. Konsep dasar dari suatu sistem pakar adalah sebagai berikut :

1. Keahlian

Adalah suatu pengetahuan khusus yang diperoleh dari latihan, belajar dan pengetahuan. Pengetahuan dapat berupa fakta, teori, aturan, strategi global untuk memecahkan masalah.

2. Ahli (*Expert*)

Melibatkan kegiatan mengenali dan memformulasikan permasalahan, memecahkan masalah secara cepat dan tepat, menerangkan pemecahannya, belajar dari pengalaman, merestrukturisasi pengetahuan, memecahkan aturan serta menentukan relevansi.

3. Mentransfer Keahlian (*Transferring Expertise*)

Adalah proses pentransferan keahlian dari seorang pakar kedalam komputer agar dapat digunakan oleh orang lain yang bukan pakar. Pengetahuan tersebut ditempatkan kedalam sebuah komponen yang dinamakan basis pengetahuan.

4. Menyimpulkan Aturan (*Inferencing Rule*)

Merupakan kemampuan komputer yang telah diprogram, penyimpulan ini dilakukan oleh mesin *inferensi* yang meliputi prosedur tentang penyelesaian masalah.

5. Peraturan (*Rule*)

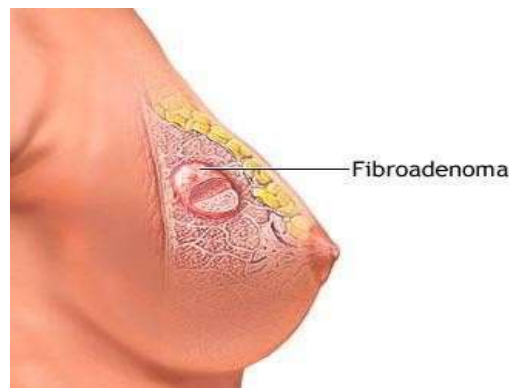
Diperlukan karena mayoritas dari sistem pakar bersifat *rule-based systems*, yang berarti pengetahuan disimpan dalam bentuk peraturan.

6. Kemampuan menjelaskan (*Explanation Capability*)

Adalah karakteristik dari sistem pakar yang memiliki kemampuan menjelaskan atau memberi saran mengapa tindakan dianjurkan atau tidak dianjurkan. (B. Herawan Hayadi, 2018: 5).

II.6. Fibroadenoma Mammae (FAM).

FAM merupakan tumor jinak payudara yang paling sering ditemui pada wanita muda dan dewasa, yaitu pada wanita 3 dekade pertama kehidupan. Frekuensi FAM yang paling tinggi adalah pada wanita yang berumur 20-25 tahun. Tumor ini ditemukan 2 kali lebih sering pada orang kulit hitam, pasien dengan kadar hormon tinggi (remaja dan wanita hamil), dan pasien yang mendapatkan terapi hormon estrogen. Penderita FAM memiliki risiko 2 kali lebih besar untuk menderita kanker payudara dikemudian hari dibandingkan wanita yang tidak menderita FAM.



Gambar 1. *Fibroadenoma mammae*

FAM merupakan tumor jinak payudara yang paling sering terjadi pada wanita usia kurang dari 25 tahun. Pada populasi barat, FAM ditemukan pada 7-13% pasien yang menjalani pemeriksaan payudara, sedangkan di *Shanghai* kurang lebih 1 dari 350 wanita di diagnosis menderita FAM sebelum usia 60 tahun. Insiden FAM pada

wanita menurun sesuai dengan peningkatan usia, dan sebagian menurun saat *menopause*. Berdasarkan laporan dari *New South Wales Breast (NSWB) Cancer Institute*, FAM umumnya terjadi pada wanita berusia 21- 25 tahun, kurang dari 5% terjadi pada usia di atas 50 tahun, sedangkan prevalensinya lebih dari 9% populasi wanita terkena FAM.

Hasil survey awal yang peneliti lakukan terhadap 10 pasien wanita yang berkunjung dan didiagnosa menderita FAM di poliklinik spesialis Bedah Umum RSUD Bengkalis, didapat informasi bahwa 10 orang pasien berusia rata-rata antara 15- 35 tahun mengeluh adanya benjolan di payudara, 4 orang mengatakan adanya riwayat keluarga yang pernah menderita kanker payudara, 3 orang mengatakan tidak pernah memberikan ASI (air susu ibu) kepada anak nya, 3 orang mengatakan dirinya menggunakan alat kontrasepsi hormon. Dari 10 pasien wanita yang berkunjung dan berobat dokter melakukan pemeriksaan fisik dan pemeriksaan penunjang seperti *ultrasonografi (USG)* dan *Fine Needle Aspiration Cytology (FNAC)*, setelah menjalani pemeriksaan selanjutnya pasien dianjurkan untuk tindakan operasi. Peningkatan angka kejadian FAM dari tahun ketahun serta adanya kekhawatiran benjolan pada payudara disalah artikan sebagai kanker. Perlu diketahui bahwa tidak semua benjolan pada payudara bersifat ganas dan juga tidak semua benjolan harus mendapatkan tindakan pembedahan. Oleh karena itu perlu adanya pembahasan tentang FAM dan faktor-faktor apa saja yang menyebabkan kejadian FAM. Berdasarkan uraian di latar belakang dan fenomena tersebut peneliti tertarik untuk melakukan penelitian tentang Faktor-Faktor Yang Menyebabkan Kejadian

Fibroadenoma Mammae (FAM) Pada Pasien Wanita Yang Berkunjung Di Poliklinik Spesialis Bedah Umum RSUD Bengkalis Tahun 2017. (Alini, Lise Widya : 2018)

Menurut organisasi amal pendukung kanker, laki-laki masih memiliki jaringan payudara. Inilah yang membuat mereka tetap rentan mengalami risiko tumor payudara maupun kanker payudara. Hingga pubertas, jaringan payudara pada anak laki-laki dan perempuan adalah sama. Keduanya memiliki sedikit jaringan payudara di belakang puting dan are atau daerah kulit yang lebih gelap di sekitar puting. Jaringan payudara tersebut terdiri dari beberapa tabung kecil atau saluran yang dikelilingi oleh jaringan lemak, jaringan ikat, pembuluh darah dan pembuluh limfatik. Cancer Research UK melaporkan sekitar 390 pria didiagnosis dengan kanker payudara setiap tahunnya. Sementara itu, jumlah wanita yang didiagnosis menderita kanker payudara adalah 54.800 orang. Lebih lanjut, pria yang memiliki kadar estrogen sangat tinggi juga memiliki risiko lebih besar terkena kanker. Dimana kanker payudara itu sendiri yang menjadi salah satu penyebabnya ialah pembesaran dari adanya *Fibroadenoma mammae*. (Assyifa R.A Isa : 2018).

II.7. Patofisiologi

Fibroadenoma mammae biasanya ditemukan pada kuadran luar atas, merupakan *lobus* yang berbatas jelas, mudah digerakkan dari jaringan di sekitarnya. Pada gambaran *histologi* menunjukkan *stroma* dengan *poliferasi fibroblast* yang mengelilingi kelenjar dan rongga *kistik* yang dilapisi *epitel* dengan bentuk berbeda dan ukuran yang berbeda. Pembagian *Fibroadenoma* berdasarkan *histologi*, yaitu :

1. *Fibroadenoma Pericanaliculare*

Yaitu kelenjar berbentuk bulat dan lonjong dilapisi epitel selapis atau beberapa lapis.

2. *Fibroadenoma Intracanalicular*

Yaitu jaringan ikat mengalami poliferasi lebih banyak sehingga kelenjar berbentuk panjang-panjang (tidak teratur) dengan lumen yang sempit atau menghilang.

Fibroadenoma biasanya berdiameter 1 – 5 cm dan biasanya bila dibuang diameternya mencapai 2 – 4 cm, bila diameter mencapai 10 - 15 cm muncul *Giant Fibroadenoma*. Tampilan pada wanita muda biasanya terdapat masa yang terasa sedangkan pada wanita yang lebih tua ditemukan kepadatan *mamografis*. Pertumbuhan tumor dirangsang oleh hormon dan regresi terjadi *pasca menopause* (Assyifa R.A Isa : 2018).

II.8. Metode *Naïve Bayes*

Naive Bayes adalah metode untuk mengklasifikasi *probabilitas* sederhana yang didasarkan pada *Teorema Bayes*. Dalam *Teorema Bayes* dikombinasikan dengan “*Naive*” yang berarti dalam atribut dengan sifat bebas (*independent*).

Metode *naïve bayes* merupakan pendekatan statistik untuk melakukan inferensi induksi pada persoalan klasifikasi. Perhitungan *naive bayes* dapat dilakukan dengan langkah berikut ini :

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

Keterangan

X = Data dengan *class* yang belum diketahui.

H = Hipotesis data X merupakan suatu *class* spesifik

P(H|X) = Probabilitas hipotesis H berdasarkan kondisi X (Posteriori prob)

P(H) = Probabilitas hipotesis H (Prior prob)

P(X|H) = Probabilitas X berdasarkan kondisi tersebut.

P(X) = Probabilitas dari X.

II.9. Database

Database adalah sekumpulan tabel-tabel yang saling berelasi, relasi tersebut bisa ditunjukkan dengan kunci dari tiap tabel yang ada. Satu *database* menunjukkan satu kumpulan data yang dipakai dalam satu lingkup perusahaan dan instansi (Gellysa Urva, Helmi Fauzi Siregar : 2015).

Database diartikan sebagai kumpulan data yang terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dimanipulasi, diambil dan dicari secara tepat. Selain berisi data, *database* juga berisi metadata. Metadata adalah data yang menjelaskan tentang struktur data itu sendiri. Sebagai contoh, dapat memperoleh informasi tentang nama-nama kolom dan tipe data yang ada pada sebuah tabel. Data nama kolom dan tipe yang ditampilkan tersebut disebut metadada (Dwiny Meidelfi, dkk, 2018).

II.10. *Hypertext Preprocessor (PHP)*

PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan web dan dapat ditanamkan pada sebuah skripsi *HTML*. Bahasa *PHP* dapat dikatakan menggambarkan beberapa bahasa pemrograman seperti C, Java, dan Perl serta mudah untuk dipelajari. *PHP* merupakan bahasa *scripting server – side*, dimana pemrosesan datanya dilakukan pada sisi *server*. Sederhananya, serverlah yang akan menerjemahkan skrip program, baru kemudian hasilnya akan dikirim kepada *client* yang melakukan permintaan.

Adapun pengertian lain *PHP* adalah *akronim* dari *Hypertext Preprocessor*, yaitu suatu bahasa pemrograman berbasis kode – kode (*script*) yang digunakan untuk mengolah suatu data dan mengirimkannya kembali ke *web browser* menjadi kode *HTML*”. Pada prinsipnya *server* akan bekerja apabila ada permintaan dari *client*. Dalam hal ini *client* menggunakan kode-kode *PHP* untuk mengirimkan permintaan ke server. 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*.

Selanjutnya *webserver* akan mencari berkas yang diminta dan menampilkan isinya di *browser*. *Browser* yang mendapatkan isinya segera

menerjemahkan kode *HTML* dan menampilkannya. Pada prinsipnya sama dengan memanggil kode *HTML*, namun pada saat permintaan dikirim ke *web-server*, *web-server* akan memeriksa tipe *file* yang diminta user. Jika tipe *file* yang diminta adalah *PHP*, maka akan memeriksa isi script dari halaman *PHP* tersebut. Apabila dalam *file* tersebut tidak mengandung script *PHP*, permintaan user akan langsung ditampilkan ke *browser*, namun jika dalam *file* tersebut mengandung script *PHP*, maka proses akan dilanjutkan ke modul *PHP* sebagai mesin yang menerjemahkan script-script *PHP* dan mengolah *script* tersebut, sehingga dapat dikonversikan ke kode-kode *HTML* lalu ditampilkan ke *browser user*. (Astria Firman, dkk, 2016).

II.11. MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data *SQL* (DBMS) yang *multithread*, dan *multi-user*. *MySQL* adalah implementasi dari sistem manajemen *basisdata* relasional (RDBMS). *MySQL* dibuat oleh *TcX* dan telah dipercaya mengelola sistem dengan 40 buah *database* berisi 10.000 tabel dan 500 di antaranya memiliki 7 juta baris. (Adis Lena Kusuma Ratna, 2014)

MySQL adalah *Relational Database Management System (RDBMS)* yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Dimana setiap orang bebas untuk menggunakan *MySQL*, namun tidak boleh dijadikan produk turunan yang bersifat komersial. *MySQL* sebenarnya merupakan turunan salah satu

konsep utama dalam database sejak lama, yaitu *SQL (Structured Query Language)*. *SQL* adalah sebuah konsep pengoperasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu sistem *database (DBMS)* dapat diketahui dari cara kerja *optimizer-nya* dalam melakukan proses perintah-perintah *SQL*, yang dibuat oleh *user* maupun program-program aplikasinya. Sebagai *database server*, *MySQL* dapat dikatakan lebih unggul dibandingkan *database server* lainnya dalam *query* data. Hal ini terbukti untuk *query* yang dilakukan oleh *single user*, kecepatan *query MySQL* bisa sepuluh kali lebih cepat dari *PostgreSQL* dan lima kali lebih cepat dibandingkan *Interbase*. (Adis Lena Kusuma Ratna, 2014).

II.12. Normalisasi

Menurut Indrajani (2015 : 11) Normalisasi adalah suatu teknik dengan pendekatan *bottom-up* yang digunakan untuk membantu mengidentifikasi hubungan. Dimulai dari menguji hubungan, yaitu *functional dependencies* antara atribut. Tujuan utama normalisasi adalah mengidentifikasi kesesuaian hubungan yang mendukung data untuk memenuhi kebutuhan data perusahaan. Terdapat lima bentuk normal yang biasa digunakan yaitu :

1. *First Normal Form (1 NF)*

Sudah tidak ada *repeating group* yaitu pengulangan yang terjadi pada beberapa atribut atau kolom dalam sebuah tabel, dan juga setiap atribut harus bernilai tunggal.

2. *Second Normal Form (2 NF)*

Untuk menjadikan tabel normal tingkat ke 2 maka sudah 1NF dan setiap atribut yang bukan *primary key* sepenuhnya secara *funksional* tergantung pada semua atribut pembentuk *primary key*.

3. *Third Normal Form (3 NF)*

Tabel sudah 2NF dan tidak memiliki *transitive dependencies*, *Transitive dependency* adalah ketika ada atribut yang secara tidak langsung tergantung pada *primary key*.

4. *Fourth Normal Form (4 NF)*

Relasi berada pada bentuk normal keempat apabila memenuhi syarat BCNF dan tidak mempunyai *multivalued dependency*.

5. *Fifth Normal Form (5 NF)*

Tabel bentuk normal kelima sering disebut *PJNF (Projection Join Normal Form)*, penyebutan *PJNF* karena untuk suatu relasi akan berbentuk normal kelima kata dari *belter* yang disebut dapat dipecah atau diproyeksikan menjadi beberapa tabel dan dari proyeksi-proyeksi dapat disusun kembali (*join*) menjadi tabel yang sama dengan keadaan semula.

II.13. *Unified Modeling Language (UML)*

Menurut Windu Gata (2015) Hasil pemodelan pada OOAD terdokumentasikan dalam bentuk *Unified Modeling Language (UML)*. *UML* 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. *UML* saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem.

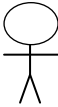
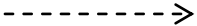
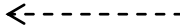
Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis *UML* adalah sebagai berikut (Urva. Dkk, 2015) :

1. *Use case* Diagram

Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dapat dikatakan *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Adapun simbol-simbol yang digunakan dalam *use case* diagram terdapat pada tabel II.1 :

Tabel II.1. Simbol *Use Case*

Gambar	Keterangan
	<i>Use Case</i> menggambarkan fungsionalitas yang

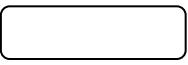
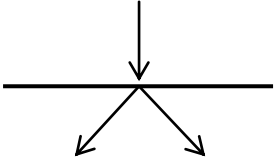
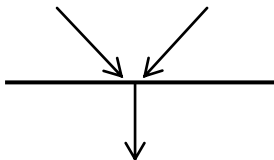
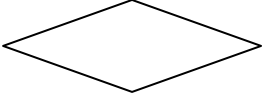
	disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktif, yang dinyatakan dengan menggunakan kata kerja.
	Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>use case</i> , tetapi tidak memiliki control terhadap <i>use case</i> .
	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 mengidikasikan aliran data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengidinkasikan bila aktor berinteraksi secara pasif dengan sistem.
	<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.
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

(Sumber :Urva, dkk; 2015)

2. Diagram Aktivitas (*Activity Diagram*)

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Adapun simbol-simbol yang terdapat pada activity diagram terdapat pada tabel II.2:

Tabel II.2. Simbol Activity Diagram

Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	<i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> , <i>false</i> .
	<i>Swimlane</i> , pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.

(Sumber :Urva, dkk; 2015)

3. *Class Diagram* (Diagram Kelas)

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. *Class diagram* secara khas meliputi: Kelas (*Class*), Relasi, *Associations*, *Generalization* dan *Aggregation*, Atribut (*Attributes*), Operasi (*Operations/Method*), *Visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *multiplicity* atau *kardinaliti* yang dapat dilihat pada tabel dibawah ini :

Tabel II.3. Multiplicity Class Diagram

Multiplicity	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

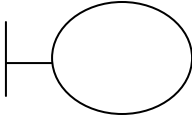
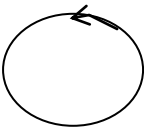
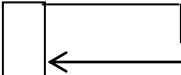
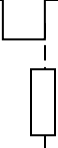
(Sumber :Urva, dkk; 2015

4. Diagram Urutan (*Sequence Diagram*)

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram* yaitu :

Tabel II.4. Simbol Sequence Diagram

Gambar	Keterangan
	<i>EntityClass</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> , berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan <i>form</i> cetak.
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	<i>Message</i> , simbol mengirim pesan antar <i>class</i> .
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i> , <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .

(Sumber :Urva, dkk; 2015)