



BAB II
TINJAUAN PUSTAKA

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terdahulu

Berdasarkan penelitian yang dilakukan oleh Ahyarudin dan Topiq (2021) mengenai Sistem Pakar Diagnosa Penyakit Paru-Paru Pada Anak Dengan Metode *Naïve Bayes* Berbasis *Web*, penelitian Ahyarudin dan Topiq menggunakan metode *Naïve Bayes* untuk diagnosa penyakit Paru-Paru, sedangkan penelitian ini menggunakan metode *Naïve Bayes* untuk diagnosa penyakit Meningitis.

Berdasarkan penelitian yang dilakukan oleh Handoko dan Neneng (2018) mengenai Sistem Pakar Diagnosa Penyakit Selama Kehamilan Menggunakan Metode *Naïve Bayes* Berbasis *Web*, penelitian Handoko dan Neneng menggunakan metode *Naïve Bayes* untuk diagnosa penyakit Selama Kehamilan, sedangkan penelitian ini menggunakan metode *Naïve Bayes* untuk diagnosa penyakit Meningitis.

Berdasarkan penelitian yang dilakukan oleh Setiyani dan Prasetyaningrum (2021) mengenai Penerapan Metode *Naïve Bayes Classifier* Pada Sistem Pakar Diagnosa Penyakit Lambung, penelitian Setiyani dan Prasetyaningrum menggunakan metode *Naïve Bayes* untuk diagnosa penyakit Lambung, sedangkan penelitian ini menggunakan metode *Naïve Bayes* untuk diagnosa penyakit Meningitis.

Berdasarkan penelitian yang dilakukan oleh Yuliana, dkk (2018) mengenai Sistem Pakar Diagnosa Penyakit Ispa Menggunakan Metode *Naïve*

BayesClassifier Berbasis *Web*, penelitian Yuliana, dkk menggunakan metode *Naïve Bayes* untuk diagnosa penyakit ISPA, sedangkan penelitian ini menggunakan metode *Naïve Bayes* untuk diagnosa penyakit Meningitis.

Berdasarkan penelitian yang dilakukan oleh Yunita dan Widodo (2021) mengenai Sistem Pakar Diagnosa Gangguan Jiwa Menggunakan Metode *Naïve Bayes* Berbasis *Web*, penelitian Yunita dan Widodo menggunakan metode *Naïve Bayes* untuk diagnosa Gangguan Jiwa, sedangkan penelitian ini menggunakan metode *Naïve Bayes* untuk diagnosa penyakit Meningitis.

Berdasarkan penelitian Gea, dkk (2021) yang berjudul Sistem Pakar Mendiagnosa Penyakit Anemia Dengan Menggunakan Metode *Teorema Bayes* Berbasis *Web*, penelitian Gea, dkk menggunakan metode *Teorema Bayes* untuk diagnosa penyakit anemia, sedangkan penelitian ini menggunakan metode *teorema* untuk diagnosa penyakit Kardiomiopati Diabetik.

Berdasarkan penelitian Rahmadsyah (2021) yang berjudul Sistem Pakar Diagnosa Penyakit Kulit Akibat Infeksi Jamur Menggunakan *Teorema Bayes*, penelitian Rahmadsyah menggunakan metode *Teorema Bayes* untuk diagnosa penyakit kulit, sedangkan penelitian ini menggunakan metode *teorema* untuk diagnosa penyakit Kardiomiopati Diabetik.

Berdasarkan penelitian Sagat dan Purnomo (2021) yang berjudul Sistem Pakar Diagnosa Penyakit Mata Menggunakan Metode *Teorema Bayes*, penelitian Sagat dan Purnomo menggunakan metode *Teorema Bayes* untuk diagnosa penyakit mata, sedangkan penelitian ini menggunakan metode *teorema* untuk diagnosa penyakit Kardiomiopati Diabetik.

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 salah satu cabang dari *Artificial Intelligence* (AI) yang membuat penggunaan secara luas *knowledge* yang khusus untuk penyelesaian masalah tingkat manusia seperti yang biasanya dilakukan seorang pakar. Sistem pakar (*expert system*) adalah sitem informasi yang berisi pengetahuan dari pakar sehingga dapat digunakan untuk berkonsultasi. Pengetahuan dari pakar kemudian didalam sistem digunakan sebagai dasar oleh sistem pakar untuk menjawab pertanyaan (konsultasi). Kepakaran (*expertise*) adalah pengetahuan yang ektensif dan spesifik yang diperoleh melalui rangkaian pelatihan, membaca dan pengalaman. Pengetahuan membuat pakar dapat mengambil keputusan secara lebih baik dan lebih cepat dari pada non-pakar dalam memecahkan problem yang kompleks. (Yunita dan Widodo, 2021: 3).

Sistem pakar merupakan cabang dari *Artificial Intelligence* (AI) yang cukup tua karena sistem ini mulai dikembangkan pada pertengahan 1960. Sistem pakar yang muncul pertama kali adalah *General Purpose Problem Solver* (BPS) yang dikembangkan oleh Newel dan Simon. Sistem Pakar adalah sistem yang berusaha mengadopsi pengetahuan para pakar dalam menyelesaikan permasalahan berbasis sistem komputer. Menurut Turban dan Aronson “sistem pakar merupakan sistem yang menggunakan pengetahuan manusia yang dimasukkan kedalam komputer untuk memecahkan permasalahan yang biasanya diselesaikan oleh

pakar. Dibawah ini merupakan komponen yang dibutuhkan dalam membangun sistem pakar, antara lain adalah:

1. Memori Kerja (*Working Memory*)
2. Antar Muka Pengguna (*User Interface*)
3. Mekanisme Inferensi (*Inference Machine*)
4. Basis Pengetahuan (*Knowledge Base*). (Yuliana, dkk, 2018: 130).

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 gejalagejala yang teramati, diantaranya medis, elektronis, mekanis, dan diagnosis perangkat lunak. (Ayudia, 2020: 383).

II.2.3. Penyakit

Penyakit adalah kondisi abnormal tertentu yang secara negatif memengaruhi struktur atau fungsi sebagian atau seluruh tubuh suatu mahluk hidup, dan bukan diakibatkan oleh cedera eksternal apa pun. Penyakit juga dikenal sebagai kondisi medis yang berhubungan dengan gejala dan tanda klinis tertentu. Suatu penyakit dapat disebabkan oleh faktor-faktor eksternal seperti patogen atau oleh disfungsi internal. Sebagai contoh, disfungsi internal sistem imun dapat

menghasilkan berbagai penyakit yang berbeda, di antaranya berbagai bentuk defisiensi imun, hipersensitivitas, alergi, dan penyakit autoimun. Penyakit merupakan keadaan tidak normal dalam tubuh sehingga mengganggu aktivitas sehari-hari. Penyakit terjadi bila ada perubahan dari nilai rata-rata normal yang telah ditetapkan sehingga diagnosisanya ditentukan oleh medis. (Sukroni, 2017: 4).

II.2.4.Meningitis

Meningitis merupakan infeksi yang menyerang sistem saraf pusat (SSP), terutama menyerang anak pada usia kurang dari 2 tahun, dengan puncak angka kejadian pada usia 6-18 bulan. Angka kejadian meningitis diperkirakan 1-2 juta kasus terjadi dalam setahun dengan mortalitas pasien berkisar antara 2%-30% diseluruh dunia. Insidensi kasus meningitis bakterial pada anak-anak usia 20 tahun), secara berturut-turut sebesar 20,6 dan 10 per 100.000 per tahun (Wall et al., 2014). Kasus meningitis bakteria di Indonesia sekitar 158/100.000 kasus setiap tahun, diakibatkan oleh *Haemophilus influenzae* type b 16/100.000 dan bakteria lainnya 67/100.000, bila dibandingkan dengan negara maju angka tersebut lebih tinggi. Pasien meningitis yang bertahan hidup berisiko mengalami komplikasi. Komplikasi utamanya terjadi karena terdapat kerusakan pada otak. Pasien yang bertahan hidup dari meningitis dapat mengalami gangguan saraf. Meningitis dapat disembuhkan dengan pengobatan yang tepat, namun penyakit ini akan menimbulkan masalah kesehatan lainnya, yaitu gangguan penglihatan sebagian, gangguan pendengaran, dan acquired brain injury. Hilangnya penglihatan sebagian terjadi karena meningitis dapat merusak saraf yang bertanggung jawab untuk penglihatan (saraf optik), sehingga menyebabkan

kebutaan pada salah satu dari kedua mata. Banyak orang mengalami gangguan penglihatan sementara karena pembengkakan saraf optik setelah meningitis. Penyakit lanjutan dari meningitis lainnya adalah acquired brain injury juga dapat menyebabkan perubahan yang lebih halus. Otak membutuhkan waktu lebih dari 20 tahun untuk berkembang sepenuhnya. Sedangkan apabila mengalami gangguan pendengaran, penderita akan mengalami 5 kali lebih memiliki gangguan pendengaran yang signifikan, bahkan 2,4% pasien mengalami gangguan pendengaran bilateral yang memerlukan implan koklea. (Hadning, dkk, 2020: 10).

II.2.5. Metode *Naïve Bayes*

Naïve Bayes merupakan salah satu metode yang dapat digunakan untuk mengklasifikasikan data. *Bayesian classification* merupakan pengklasifikasian statistik yang dapat digunakan untuk memprediksi probabilitas keanggotaan suatu class. Teorema ini dikemukakan oleh ilmuwan Inggris, Thomas Bayes, dengan memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya. *Teorema Bayes* dikombinasikan dengan “*Naïve*” yang berarti setiap atribut/variable bersifat bebas (*independent*). *Naïve Bayes* dapat dilatih dengan efisien dalam pembelajaran terawasi (*supervised learning*). Keuntungan dari pengklasifikasi adalah bahwa hanya membutuhkan sejumlah kecil data pelatihan untuk memperkirakan para-meter (sarana dan varians dari variable) yang diperlukan untuk klasifikasi. Karena variabel independen diasumsikan, hanya variasi dari variabel untuk masing-masing kelas harus ditentukan, bukan seluruh matriks kovarians. Dalam prosesnya, *Naïve Bayes* mengasumsikan bahwa ada atau tidaknya suatu fitur pada suatu kelas tidak berhubungan dengan ada atau

tidaknya fitur lain dikelas yang sama. Pada saat klasifikasi, pendekatan *bayes* akan menghasilkan label kategori yang paling tinggi probabilitasnya. (Handoko dan Neneng, 2021: 52).

Naïve Bayes adalah metode klasifikasi yang berdasarkan probabilitas dan *Teorema Bayesian* dengan asumsi bahwa setiap variabel X bersifat bebas atau berdiri sendiri dan tidak ada kaitannya dengan variabel lainnya. Metode NBC menempuh dua tahap dalam proses klasifikasi teks, yaitu tahap pelatihan dan tahap klasifikasi. Probabilitas adalah kemungkinan terjadinya suatu peristiwa antara 0 s/d 1. Pada tahap pelatihan dilakukan proses analisis terhadap sampel data yang dapat menjadi representasi dokumen. Perhitungan perbandingan antara term pada data *testing* dengan setiap kelas yang ada dapat dilakukan dengan persamaan 1.

$$P(a_i|v_j) = \frac{nc+m.p}{n+m} \dots\dots\dots (II.1)$$

Keterangan:

n : Jumlah term pada data latih dimana $v = v_j$

nc : Jumlah term dimana $v = v_j$ dan $a = a_j$

p : Probabilitas setiap kelas dalam data latih

m : Jumlah term pada data uji. (Yuliana, dkk, 2018: 4).

II.2.6. Metode *Teorema Bayes*

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 2.

$$P(H|E) = \frac{P(E|H)*P(H)}{P(E)} \dots\dots\dots (II.2)$$

Dimana:

$P(H|E)$: Probabilitas H benar jika diberikan evidence E.

$P(E|H)$: Probabilitas muncul evidence E, jika diketahui hipotesis H benar.

$P(H)$: Probabilitas hipotesis H (menurut hasil sebelumnya) tanpa memandang evidence apapun.

$P(E)$: Probabilitas evidence E. (Gea, dkk, 2021: 324).

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

HTML adalah "bahasa" untuk membuat dokumen yang bisa diakses lewat *web*. HTML memuat dua bagian yakni isi dan pemformat. Isi adalah materi yang akan disampaikan, sedangkan bagian pemformat berwujud "*tag*" untuk menjadikan isi bisa diterjemahkan sebagai paragraf, ganti baris, *heading*, *unordered list*, *ordered list*, *table*. Dokumen HTML merupakan dokumen teks biasa yang bisa dibuka dengan pengedit teks seperti Notepad. Bila dibuka dengan Notepad akan terlihat bahwa yang tersaji di *web browser* sebenarnya sudah ada di sana. Tetapi di Notepad terlihat ada tambahan di sana-sini berupa "*tag*" sehingga

di *browser* terlihat seperti itu. Anatomi dokumen HTML terdiri dari dua bagian, yakni HEAD dan BODY. Bagian HEAD biasanya memuat informasi tentang atribut dari dokumen itu. Sedangkan bagian BODY memuat informasi utama yang akan dikomunikasikan lewat *web*. Adanya *link* dalam dokumen HTML menjadikan dokumen ini bisa menunjuk ke dokumen-dokumen HTML lain baik di folder yang sama, di *server* yang sama, maupun di *server* lain. Inilah alasan mengapa dokumen ini disebut hiperteks. Satu dokumen HTML juga bisa "memuat" dan menampilkan *file-file* lain seperti gambar, audio, Word, PDF, dan sebagainya. (Huzaeni, dkk, 2019: 140).

Hypertext Markup Language atau HTML adalah bahasa yang digunakan pada dokumen web. Dengan kata lain, HTML merupakan bahasa pemrograman berupa *tag-tag* yang digunakan untuk membuat dan mengatur penampilan dari halaman website. (Irmanawan, 2020: 44).

II.2.8. Hypertext Preprocessor (PHP)

PHP adalah bahasa standar yang digunakan dalam *website*. PHP adalah bahasa program yang berbentuk skrip yang diletakan di dalam *server web*. (Irawan, 2020: 46).

PHP singkatan dari Hypertext Preprocessor merupakan bahasa pemogram yang skript bersifat *Open Source*. Program ini bersifat *server side*, artinya tanpa adanya *server* yang berjalan disisinya script program PHP tidak dapat dijalankan PHP adalah *script* yang ditanamkan dalam HTML untuk membuat halaman

website dinamis yang berkerja secara otomatis dan berfungsi sebagai pengolahan data pada *server* di mana *script* tersebut di jalankan. (Huzaeni, 2019: 140).

II.2.9. My Structure Query Language (MySQL)

MYSQL merupakan salah satu *database* yang bersifat *open source* yang populer di dunia, dimana saat ini digunakan lebih dari 100 juta pengguna diseluruh dunia MYSQL turunan dari SQL (*Structured Query Language*) yang sering digunakan oleh para *programmer* dalam pembuatan program. Sistem *database* MYSQL selain mudah memahami sintaks dan pengaksesan juga di dukung beberapa fitur seperti *Multithreaded*, multi-user dan SQL *database* managemen sistem. (Huzaeni, dkk, 2019: 141).

MySQL (*My Structured Query Language*) atau yang biasa dibaca mai-se-kuel adalah sebuah program pembuat dan pengelola *database* atau yang sering disebut dengan DBMS (*Database Management System*), sifat dari DBMS ini adalah *open source*. (Imaniawan, 2020: 45).

II.2.10. Unified Modeling Language(UML)

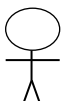
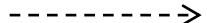
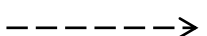
UML yaitu satu kumpulan konvensi permodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem perangkat lunak yang terkait dengan objek. UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan sistem yang besar dan kompleks. UML tidak hanya digunakan dalam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

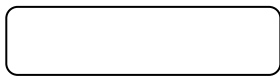
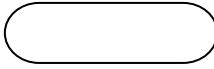
		terukur bagi suatu <i>actor</i> .
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemennya (sinergi).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

(Sumber: Andikos, 2019: 39)

2. Diagram Aktivitas (*Activity Diagram*)

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* dapat digambarkan dengan simbol-simbol seperti pada tabel II.2.

Tabel II.2. Simbol *Activity Diagram*

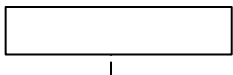
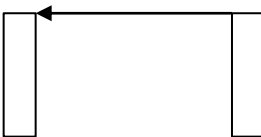
Gambar	Nama	Keterangan
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Activity Final</i>	Bagaimana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

(Sumber: Andikos, 2019: 39)

3. Diagram Urutan (*Sequence Diagram*)

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence Diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.3.

Tabel II.3. Simbol *Sequence Diagram*

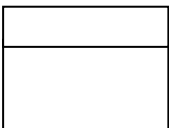
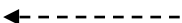
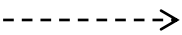
Gambar	Nama	Keterangan
	<i>Lifeline</i>	Objek entity, antarmuka yang saling berinteraksi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.

(Sumber: Andikos, 2019: 39)

4. *Class Diagram* (Diagram Kelas)

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain. *Class diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.4.

Tabel II.4. Class Diagram

Gambar	Nama	Keterangan
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
	<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
	<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
	<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri
	<i>Assocation</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

(Sumber: Andikos, 2019: 39)