

BAB II

KAJIAN PUSTAKA

A. Kajian Teoritis

1. Sistem Pakar

Sistem Pakar atau yang dikenal sebagai *Knowledge Based* adalah aplikasi komputer yang memfasilitasi pengambilan keputusan atau penyelesaian masalah dalam bidang tertentu. Sistem ini beroperasi dengan menggunakan pengetahuan yang sebelumnya telah ditentukan oleh pakar yang berkompeten. Dalam pembuatannya, sistem pakar menyatukan berbagai aturan inferensi dengan basis pengetahuan yang disediakan oleh satu atau lebih pakar. Gabungan ini disimpan dalam basis data komputer dan dipakai dalam proses pengambilan keputusan untuk mengatasi masalah spesifik (Seppewali, Mulyo, and Riswan 2023).

Sistem pakar adalah metode untuk membuat sistem komputer yang menggunakan pengetahuan dari pakar sebagai sumber informasi. Sistem ini dirancang untuk memungkinkan komputer mengambil keputusan mirip dengan seorang pakar dalam bidang tertentu (Yuliyanti 2020). Sistem pakar merupakan aplikasi perangkat lunak yang menyimpan pengetahuan dari satu atau lebih ahli manusia dalam bidang tertentu. Program ini pertama kali diciptakan oleh ahli kecerdasan buatan pada era 1960-an dan 1970-an, dan mulai digunakan dengan keperluan bisnis pada dekade 1980-an. Pada dasarnya, sistem berbasis pengetahuan menggunakan seperangkat aturan untuk melakukan analisis informasi tentang masalah tertentu yang

diberikan oleh pengguna, serta melakukan analisis matematis terhadap masalah tersebut. Berdasarkan desainnya, sistem pakar juga dapat merekomendasikan serangkaian tindakan kepada pengguna untuk menyelesaikan masalah. Sistem ini menggunakan penalaran untuk mencapai kesimpulan (Syukriyanto, Suryadi, and Maulana 2023).

Berdasarkan penelitian diatas disimpulkan bahwa Sistem pakar adalah program komputer canggih yang dirancang untuk meniru kemampuan pengambilan keputusan seorang pakar di bidang tertentu. Sistem ini beroperasi menggunakan pengetahuan dan analisis yang telah disiapkan sebelumnya oleh pakar yang memiliki keahlian dalam bidang tersebut. Manfaat utama sistem pakar adalah meningkatkan efisiensi dan efektivitas pengambilan keputusan, memperluas akses ke keahlian pakar, mendokumentasikan dan melestarikan pengetahuan pakar, serta membantu dalam pelatihan pakar.

2. Diagnosa

Secara etimologi, kata "diagnosis" berasal dari bahasa Yunani, yaitu "gnosis" yang berarti pengetahuan, sedangkan "dia" berarti melalui. Secara terminologi, diagnosis merujuk pada proses menetapkan keadaan yang abnormal atau normal dari suatu kondisi dengan menggunakan dasar pemikiran dan pertimbangan ilmu pengetahuan. Diagnosis ini penting dalam berbagai bidang, terutama kedokteran, untuk menentukan kondisi pasien dan merencanakan perawatan yang tepat (Indra Maulana Yusup Kusumah, Linda Apriyanti 2022).

Diagnosis merupakan proses identifikasi mengenai suatu hal. Diagnosis digunakan dalam berbagai bidang seperti medis, ilmu pengetahuan, teknik, serta bisnis. Diagnosis utama adalah kondisi setelah pemeriksaan, terbukti menjadi alasan penetapan utama pasien dirawat dirumah sakit. Secara terminologi, diagnosis merupakan penetapan kondisi yang menyimpang atau normal berdasarkan pemikiran dan pertimbangan ilmu pengetahuan (Fuad, Adhiatma, and Ikhsan 2022).

Berdasarkan penelitian diatas disimpulkan bahwa diagnosis adalah proses penetapan suatu keadaan, baik normal maupun menyimpang, berdasarkan ilmu pengetahuan. Diagnosis penting dalam berbagai bidang, terutama kedokteran, untuk menentukan kondisi pasien dan merencanakan perawatan yang tepat. Diagnosis dilakukan dengan identifikasi dan analisis informasi yang relevan, seperti gejala, tanda, hasil pemeriksaan, dan riwayat kesehatan.

3. *Forward Chaining*

Forward chaining merupakan metode reasoning dalam penggunaan *inference engine* yang dapat dijelaskan secara logis sebagai penerapan berulang modus ponens, yaitu sekumpulan aturan inferensi dan argumen yang valid. Pendekatan ini memulai dengan data tambahan sampai mencapai tujuan atau kesimpulan yang diinginkan. Mesin inferensi yang menggunakan *forward chaining* mencari aturan-aturan inferensi dan mengaktifkan aturan tersebut ketika kondisi antecedent (IF -THEN clause) terpenuhi (Rofiqoh, Kurniadi, and Riansyah 2019).

Forward Chaining merupakan metode yang diterapkan pada sistem pengetahuan atau kecerdasan buatan untuk melakukan pemikiran atau pengambilan keputusan berdasarkan fakta-fakta yang sudah diketahui menuju kesimpulan yang lebih kompleks. Istilah “*forward chaining*” berarti “maju ke depan” atau “melangkah maju.” Di dalam sistem berbasis pengetahuan, langkah ini berawal berdasarkan data yang sudah ada, kemudian menggunakan prinsip-prinsip penalaran untuk secara bertahap memperoleh kesimpulan yang lebih kompleks. Metode ini sangat membantu dalam keadaan yang melibatkan beragam data dan kaidah, serta ketika perlu menganalisis pola dan hubungan di antara mereka guna memperoleh hasil spesifik (Faisal, Opitasari, and Mufti 2024).

Berdasarkan penelitian di atas disimpulkan bahwa *forward chaining* adalah metode proses berpikir dalam kecerdasan buatan yang bekerja dengan melacak fakta yang diketahui dan aturan logis untuk mencapai kesimpulan. Metode ini dimulai dari data awal dan secara bertahap menggabungkan aturan untuk mendapatkan hasil yang lebih kompleks. *Forward chaining* bermanfaat dalam situasi dengan banyak fakta dan aturan, di mana pola dan keterkaitan perlu ditemukan untuk mencapai tujuan tertentu.

4. *Waterfall*

Model air terjun, juga dikenal sebagai model waterfall, adalah metode klasik untuk pengembangan perangkat lunak yang menggabungkan metode pengembangan linier dan berurutan. Ini terdiri

dari lima hingga tujuh fase, masing-masing dengan tujuan dan tugas yang berbeda. Siklus hidup perangkat lunak dari pengirimannya digambarkan dalam setiap fase, dan hasil dari fase sebelumnya mengalir ke fase berikutnya (Fatimah and Nuryaningsih 2020).

Waterfall adalah proses suatu sistem yang dilakukan secara berurutan atau secara linear. Artinya, jika langkah satu tidak dilakukan, maka langkah kedua, ketiga, dan seterusnya tidak akan bisa dilakukan. Tahap ketiga hanya dapat dilakukan jika tahap pertama dan kedua sudah dilakukan (Hartono 2021).

Berdasarkan penelitian diatas disimpulkan bahwa model air terjun, juga dikenal sebagai waterfall, adalah teknik klasik pengembangan perangkat lunak yang menggunakan pendekatan linier dan berurutan. Metode ini terdiri dari lima hingga tujuh tahap yang berbeda, masing-masing dengan tujuan dan tanggung jawab tertentu. Setiap tahap menggambarkan siklus hidup perangkat lunak, mulai dari pengirimannya, dan hasil dari tahap sebelumnya menjadi input untuk tahap berikutnya. Dengan kata lain, proses dalam model air terjun dilakukan secara berurutan, setiap tahap harus diselesaikan sebelum melanjutkan. Misalnya, hanya tahap pertama dan kedua yang dapat dimulai sebelum tahap ketiga.

5. Laragon

Laragon adalah aplikasi yang mirip dengan XAMPP, tetapi didesain khusus untuk memenuhi kebutuhan pengembang PHP dengan penggunaan framework Laravel. Laragon menyediakan berbagai layanan

seperti Apache, MySQL, PHP Server, Memcached, Redis, Composer, Xdebug, PhpMyAdmin, Cmdr, dan lainnya. Aplikasi ini sangat cocok bagi pengembang PHP yang bekerja dengan Laravel, karena mempermudah proses pengembangan aplikasi (Permatasari 2023).

Laragon adalah sebuah aplikasi yang memungkinkan pengguna untuk mengaktifkan server komputer lokal atau stack pengembangan web. Laragon sangat berguna untuk memulai situs web dengan cepat dan memiliki performa yang tinggi. Selain itu, Laragon mendukung berbagai platform seperti PHP, Node.js, Python, dan Ruby, membuatnya fleksibel untuk berbagai proyek pengembangan (Pratama, Amalia, and Putra 2024).

Berdasarkan penelitian diatas disimpulkan bahwa laragon merupakan aplikasi server lokal yang dirancang untuk memudahkan developer PHP, khususnya yang menggunakan framework Laravel. Laragon menyediakan berbagai layanan seperti Apache, MySQL, PHP Server, dan alat bantu lainnya yang mempercepat proses pengembangan aplikasi. Keunggulan Laragon lainnya adalah kemudahan penggunaan, kecepatan, dan dukungan untuk berbagai platform seperti PHP, Node.js, Python, dan Ruby.

6. *Database*

Basis data atau *database* adalah sekumpulan informasi yang tersimpan secara sistematis dalam komputer, memungkinkan akses dan pemeriksaan melalui program untuk mendapatkan informasi. Perangkat lunak yang mengelola dan menjalankan kueri pada basis data disebut

sistem manajemen basis data (*Database Management System atau DBMS*). Konsep dasar dari basis data atau database adalah sekumpulan data yang membentuk berkas (*file*) yang saling terkait menurut aturan tertentu untuk menghasilkan data baru atau informasi (Billah, Ramdhani, and Muhima 2023).

Basis data merupakan sekumpulan data yang beragam dan saling terkait, disimpan bersama dalam suatu media tertentu dengan tujuan untuk memperoleh informasi dari kumpulan data tersebut (Permatasari 2023). Database sistem adalah sistem komputer yang dibuat untuk menyimpan data yang telah diproses atau informasi, serta memastikan informasi tersebut dapat diakses saat diperlukan. Secara pokok, basis data adalah wadah untuk menyimpan data yang memfasilitasi akses yang cepat dan mudah (Rizky, Rafieqah Nalar and Mahardika 2023).

Berdasarkan penelitian diatas disimpulkan bahwa database atau basis data adalah kumpulan informasi yang terstruktur dan terhubung secara sistematis dalam komputer. Informasi ini dapat diolah dan dimanipulasi dengan program komputer untuk menghasilkan informasi baru. Sistem manajemen basis data (DBMS) digunakan untuk mengelola dan mengakses data dalam database. Secara ringkas, database merupakan kumpulan data yang terorganisir dan mudah diakses untuk menyimpan dan mengelola informasi.

7. MySQL

MySQL merupakan pengembangan lebih lanjut dari proyek UNIREG yang dikerjakan oleh Michael Monty Widenius dan perusahaan perangkat lunak Swedia, TeX. MySQL adalah DBMS open source yang memiliki dua jenis lisensi, yaitu perangkat lunak bebas (Free Software) dan perangkat lunak terbatas (Shareware). Oleh karena itu, MySQL adalah server database gratis dengan lisensi GNU General Public License (GPL), memungkinkan penggunaannya untuk keperluan pribadi atau komersial tanpa harus membayar biaya lisensi (Kalsum Siregar et al. 2024).

MySQL adalah sebuah sistem basis data yang dapat mengakses database berbasis jaringan dan digunakan untuk menjalankan perintah-perintah SQL dalam mendukung manajemen data secara efisien. MySQL menyediakan berbagai fitur seperti kemampuan multithreaded, multi-user, dan sebagai sistem pengelola basis data SQL (DBMS). Selain itu, MySQL terkenal karena kecepatan dan keandalannya dalam menangani volume data besar. MySQL juga kompatibel dengan banyak sistem operasi, termasuk Windows, Linux, dan macOS, sehingga memudahkan integrasi dengan berbagai lingkungan pengembangan (Permatasari 2023).

Berdasarkan penelitian diatas disimpulkan bahwa MySQL adalah DBMS open source yang dikembangkan dari proyek UNIREG oleh Michael Monty Widenius dan perusahaan TeX, yang tersedia dengan lisensi GNU GPL hingga dapat dimanfaatkan tanpa biaya untuk tujuan pribadi maupun bisnis. Sistem basis data ini mendukung multithreaded,

multi-user, dan perintah SQL, terkenal karena kecepatan dan keandalannya dalam mengelola volume data besar. Selain itu, kompatibilitasnya dengan berbagai sistem operasi seperti Windows, Linux, dan macOS memudahkan integrasi dalam berbagai lingkungan pengembangan.

8. PHP

PHP adalah bahasa pemrograman web yang berbasis server side yang digunakan untuk mengembangkan situs web dinamis dengan konten yang dapat berubah sesuai kondisi tertentu, serta memungkinkan interaksi dan umpan balik dari pengguna (Permatasari 2023). PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman yang umum digunakan untuk menciptakan halaman web yang dinamis. Ini memungkinkan kita untuk berinteraksi dengan basis data, mengirim formulir, dan melakukan banyak hal lainnya yang membuat situs web kita lebih interaktif (MRizky 2020).

PHP merupakan bahasa pemrograman skrip yang telah didesain untuk pengembangan web, khususnya dalam pembuatan halaman web dinamis. PHP saat ini merupakan bahasa pemrograman skrip yang paling banyak digunakan. PHP sering digunakan untuk pemrograman website, meskipun juga dapat digunakan untuk keperluan lain. PHP adalah salah satu bahasa *server-side scripting* yang disiapkan di antara kode HTML untuk membuat halaman web yang dinamis (Rizky, Rafieqah Nalar and Mahardika 2023).

Berdasarkan penelitian tersebut, dapat ditarik kesimpulan bahwa PHP (Preprocessor Hypertext) ialah bahasa pemrograman yang digunakan

untuk web server-side yang digunakan buat mengembangkan situs daring dinamis dan interaktif, memungkinkan interaksi dengan basis data dan pengiriman formulir. Sebagai scripting language yang paling banyak digunakan, PHP sering disisipkan di antara bahasa HTML untuk menciptakan situs web yang responsif dan sesuai kondisi tertentu. Selain pemrograman web, PHP juga dapat digunakan untuk berbagai aplikasi lain, menjadikannya bahasa yang fleksibel dan serbaguna.

9. HTML

HTML adalah salah satu bentuk markup atau kode. HTML bukanlah sebuah bahasa pemrograman, tetapi lebih merupakan sebuah bahasa yang sederhana. Jadi, bahasa yang dipakai dalam proses pembuatan website dikenal sebagai *Hypertext Markup Language* (HTML) (Pratama, Amalia, and Putra 2024).

HTML (*HyperText Markup Language*) adalah bahasa yang digunakan untuk membuat halaman web. Ini seperti kerangka dasar yang mengatur bagaimana halaman web kita akan terlihat dan berperilaku. Dalam bahasa yang lebih sederhana, HTML adalah apa yang membuat teks, gambar, dan tautan muncul di situs web (MRizky 2020).

Berdasarkan penelitian diatas disimpulkan bahwa *HyperText Markup Language*, atau HTML, adalah bahasa umum yang dipakai untuk mendesain situs daring yang berfungsi sebagai kerangka dasar yang mengatur tampilan dan perilaku halaman tersebut. Meskipun bukan bahasa pemrograman, HTML adalah markup yang memungkinkan teks, gambar,

dan tautan muncul di situs web. HTML merupakan elemen penting dalam pembuatan website, memberikan struktur dan format dasar untuk konten web.

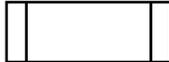
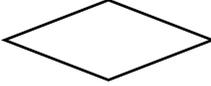
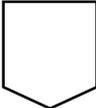
10. *Flowchart*

Flowchart adalah representasi grafis dari alur kerja atau proses yang disajikan dalam bentuk diagram. Tujuannya adalah untuk memvisualisasikan langkah-langkah dan keputusan dalam suatu sistem atau prosedur. *Flowchart* menggunakan simbol-simbol grafis untuk merepresentasikan aktivitas, keputusan, dan arus kontrol dalam suatu proses (Suhadi, Samperura, and Riza 2024).

Flowchart adalah metode untuk menggambarkan alur proses sebuah operasi. *Flowchart* dibuat untuk memvisualkan tahapan penyelesaian masalah secara sederhana, terstruktur, rapi, dan jelas dengan menggunakan simbol-simbol standar. Pembuatan *flowchart* harus memperhatikan penggunaan simbol yang tepat (Siregar, Devinta Ayu 2020).

Berdasarkan penelitian diatas telah ditemukan bahwa *flowchart* ialah gambaran alur kerja maupun proses secara terperinci. Tujuannya adalah untuk memvisualisasikan langkah-langkah, keputusan, dan arus kontrol dalam suatu sistem atau prosedur. Dengan menggunakan simbol-simbol grafis standar, *flowchart* membantu memahami proses dengan lebih mudah dan efektif. Symbol yang digunakan terdapat pada tabel 2.1 sebagai berikut :

Tabel 2.1 Simbol-simbol Flowchart

BAGAN	NAMA	FUNGSI
	Terminator	Simbol untuk permulaan (start) atau akhir (stop) dari suatu kegiatan
	Flow	Simbol yang mengaitkan satu sama lain
	Preparation	Simbol untuk menyiapkan penyimpanan yang akan diproses di dalam storage
	Proses	Simbol yang menunjukkan bahwa komputer melakukan pengolahan
	Input/output data	Simbol yang menunjukkan proses di mana data dimasukkan atau dikeluarkan dari sistem
	Sub Program	Simbol untuk pelaksanaan suatu bagian (sub-program)/prosedure
	Decision	Dalam alur proses, simbol ini digunakan untuk menunjukkan titik di mana keputusan harus dibuat
	On Page Connector	Simbol yang memungkinkan proses dimasukkan atau digabungkan dalam lembar atau halaman yang sama
	Off Page Connector	Salah satu fungsi simbol ini adalah untuk menunjukkan bahwa diagram

		flowchart menuju halaman berikutnya
	Comment	Tempat untuk membuat komentar tentang suatu tindakan

Sumber : Siregar and Hambali 2020

11. UML

Bahasa pemodelan UML digunakan untuk menspesifikasikan, mengilustrasikan, membangun, serta mencatat sistem perangkat lunak dengan pendekatan berorientasi objek (Firmansah, Rohman, and Farlina 2020). UML merupakan bahasa standar yang digunakan untuk merancang, penyimpanan dan visualisasi grafik sistem perangkat lunak. Dengan menggunakan UML, pengembang perangkat lunak dapat lebih banyak bekerja sama dan bekerja sama selama tahap perencanaan dan pengembangan. Ini juga membantu mereka memahami struktur dan perilaku sistem (Faisal, Opitasari, and Mufti 2024).

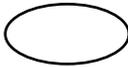
UML (*Unified Modelling Language*) merupakan metode pemodelan visual yang digunakan merancang sistem berorientasi objek. Diagram UML yang sering digunakan meliputi *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Class Diagram*, *Statemachine Diagram* dan *Component Diagram*. Pada tahap analisis kebutuhan, UML digunakan untuk visualisasi, menentukan ruang lingkup, serta mendokumentasikan artefak sistem secara efektif yang bermanfaat untuk berbagai pemangku kepentingan dalam suatu aplikasi (Marthiawati et al. 2024).

Berdasarkan penelitian diatas disimpulkan bahwa UML (*Unified Modeling Language*) adalah bahasa pemodelan standar yang digunakan untuk menyusun, menyimpan, dan menggambarkan sistem perangkat lunak secara visual dengan pendekatan berorientasi objek. UML memungkinkan pengembang perangkat lunak untuk bekerjasama secara efektif dalam perencanaan, pengembangan, serta pemahaman struktur perilaku sistem, dengan diagram seperti *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Class Diagram*, *Statemachine Diagram*, dan *Component Diagram*. Visualisasi, penentuan ruang lingkup, dan dokumentasi artefak sistem adalah semua fungsi UML, yang berguna bagi pihak yang berkepentingan dengan aplikasi.

1. *Use Case Diagram*

Diagram *Use Case* menggambarkan aktivitas-aktivitas yang dilakukan oleh suatu sistem dari perspektif eksternal. Use case digunakan untuk menggambarkan fungsi dasar dari sebuah sistem informasi. Use case mendeskripsikan cara sistem bisnis berinteraksi dengan lingkungannya (Indriyani et al. 2019). Berikut simbol *use case* diagram pada tabel 2.2.

Tabel 2.2 Simbol Use Case Diagram

No	Simbol	Nama	Keterangan
1.		Actor	simbol yang menunjukkan fungsi alat, sistem, atau individu dalam komunikasi dengan use case.
2.		<i>Use Case</i>	simbol yang menunjukkan bagaimana aktor berinteraksi dan abstrak dengan use case

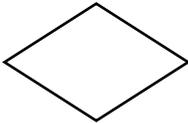
3.		Association	Simbol yang menggambarkan abstraksi dari hubungan antara aktor dan <i>use case</i> .
4.		Generalist	Simbol yang menunjukkan spesialisasi aktor agar dapat berpartisipasi dalam <i>use case</i> .
5		Include	Simbol yang menunjukkan bahwa suatu <i>use case</i> sepenuhnya merupakan bagian dari fungsionalitas <i>use case</i> lainnya.
6		Extend	Simbol yang menunjukkan bahwa suatu <i>use case</i> menambahkan fungsionalitas ke <i>use case</i> lainnya jika kondisi tertentu terpenuhi.

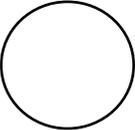
Sumber : Indriyani et al. 2019

2. Activity Diagram

Pada dasarnya diagram *activity* sering digunakan oleh *flowchart*. *Activity* diagram menggambarkan tentang alur aktivitas dalam program atau sistem, ini mencakup proses awal dari sistem tersebut, keputusan-keputusan yang bisa terjadi selama jalannya proses, serta bagaimana sistem berakhir (Tina 2022). Simbol *activity* diagram disajikan pada tabel 2.3.

Tabel 2.3 Simbol Activity Diagram

Gambar	Nama	Keterangan
	Status Awal	Simbol yang menunjukkan suatu diagram <i>activity</i> memiliki status awal.
	Aktivitas	Simbol aktivitas dalam sistem umumnya dimulai dengan kata kerja.
	Percabangan/ Decision	Simbol percabangan di mana terdapat opsi untuk melakukan lebih dari satu aktivitas.

Gambar	Nama	Keterangan
	Penggabungan / <i>Join</i>	Simbol penggabungan adalah saat lebih dari satu jalur aktivitas digabungkan menjadi satu.
	Status Akhir	Simbol yang menunjukkan diagram <i>activity</i> memiliki status akhir yang dilakukan sistem.
	<i>Swimlane</i>	Simbol yang memisahkan unit bisnis yang bertanggung jawab atas aktivitas yang sedang berlangsung.

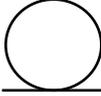
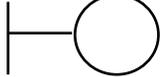
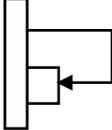
Sumber : Tina 2022

3. *Sequence Diagram*

Sequence diagram menunjukkan pesan yang lewat di antara objek untuk use case tertentu dari waktu ke waktu. *Sequence diagram* mengilustrasikan objek-objek yang berpartisipasi di dalam suatu use case. Namun, diagram interaksi adalah salah satu jenis diagram dinamis, yang menggambarkan mekanisme pelaksanaan, pesan yang dikirim, waktu pelaksanaan diagram ini diatur berdasarkan urutan waktu, dan elemen proses operasi diurutkan dari kiri ke kanan sesuai urutan waktu terjadinya pesan (Indriyani et al. 2019). Simbol *sequence diagram* disajikan pada tabel 2.4.

Tabel 2.4 Simbol *Sequence Diagram*

Gambar	Nama	Keterangan
	Actor	Orang atau divisi yang terlibat dalam suatu sistem
	Object Lifeline	Menyatakan kehidupan suatu objek
	Message	Menyatakan arah tujuan Antara Object Lifeline

	Message/return	Menyatakan arah kembali dalam satu Object Lifeline
	Message/return	Menyatakan arah tujuan Antara Object Lifeline
	Activation	Menyatakan objek dalam keadaan aktif dan berintraksi
	<i>Entity Class</i>	Gambar sistem yang digunakan sebagai landasan untuk penyusunan basis data
	<i>Boundary Class</i>	Manajemen komunikasi sistem-lingkungan
	<i>Control Class</i>	Bertanggung jawab atas objek dengan logika kelas
	<i>Recursive</i>	Simbol yang melambangkan pesan untuk dirinya.
	<i>Life Line</i>	Komponen terhubung dengan objek melalui garis putus.

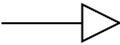
Sumber : Mufid 2024

4. *Class Diagram*

Class diagram adalah gambaran visual dari rangkaian sistem program dalam berbagai bentuk kelas yang di bentuk. *Class diagram* bagaimana alur database berjalan dalam sistem. Serta memvisualkan struktur utama sistem dari perspektif hubungan antar kelas yang digunakan untuk merancang sistem (Tina 2022). Simbol *class diagram* disajikan pada tabel 2.5.

Tabel 2.5 Simbol *Class Diagram*

Gambar	Nama	Keterangan			
<table border="1" data-bbox="459 1861 628 1995"> <tr> <td>Nama_kelas</td> </tr> <tr> <td>+attribut</td> </tr> <tr> <td>+operasi()</td> </tr> </table>	Nama_kelas	+attribut	+operasi()	Kelas	Kelas pada struktur sistem
Nama_kelas					
+attribut					
+operasi()					

	Interface	Sama seperti konsep antarmuka dalam pemrograman berorientasi objek.
	Association	Hubungan antar kelas dalam arti umum, asosiasi biasanya juga dilengkapi dengan Multiplicity.
	Directed Association	Hubungan antar kelas dengan makna satu kelas menggunakan atau dipakai oleh kelas lain, asosiasi biasanya juga dilengkapi dengan multiplicity.
	Generalisasi	Hubungan antarkelas dengan konsep generalisasi-spesialisasi (umum khusus).
	Dependency	Hubungan antarkelas yang menunjukkan ketergantungan antara satu kelas dengan kelas lainnya.
	Aggregation	Hubungan antarkelas yang menggambarkan konsep keseluruhan-bagian (whole-part).

Sumber : Tina 2022

B. Kajian Empiris

Penelitian sebelumnya tentang sistem pakar diagnosis penyakit pada hewan berbasis website telah banyak digunakan, dan penelitian sebelumnya diperlukan sebagai bahan referensi dalam bentuk jurnal ilmiah. Untuk penelitian sistem pakar diagnosis penyakit pada hewan bebek yang dilakukan dengan metode Forward Chaining, diperlukan beberapa penelitian dari sumber yang cocok untuk menjelaskan sistem pakar diagnosis pada hewan berbasis website.

Pada penelitian yang dilakukan Putra, Fadlil, and Umar (2021) dengan judul Analisis Metode *Forward Chaining* Pada Sistem Pakar Diagnosa Penyakit Hewan Sapi Berbasis Android menghasilkan sebuah sistem pakar yang dapat diakses oleh pengguna smartphone kapan saja dan di mana saja dan

dapat membantu mendiagnosa gejala penyakit pada sapi. Sistem pakar ini juga mampu menjelaskan penyebab dan solusi penyakit sapi.

Pada penelitian kedua yang dilakukan Faisal, Opitasari, and Mufti (2024) judul Sistem Pakar Pendiagnosa Penyakit Mata Dengan Metode *Forward Chaining* menghasilkan sistem yang dirancang untuk membantu pengguna dalam melakukan diagnosa penyakit mata dengan akurasi tinggi. Sistem pakar ini mengadopsi metode *forward chaining* dan dianalisis menggunakan *Unified Modeling Language* (UML), termasuk diagram *use case*, diagram *activity*, diagram *sequence*, dan diagram *class*, diagram ini menggambarkan secara detail fungsi-fungsi yang ada dalam sistem yang akan dikembangkan.

Pada penelitian ketiga yang dilakukan Rofiqoh, Kurniadi, and Riansyah (2019) dengan judul Sistem Pakar Menggunakan Metode *Forward Chaining* Untuk Diagnosa penyakit Hewan Sistem Pakar Menggunakan Metode *Forward Chaining* Untuk Diagnosa Penyakit Tanaman Karet menghasilkan sistem yang dapat mengidentifikasi penyakit pada tanaman karet berdasarkan gejala yang tersedia dalam database, serta memberikan bantuan yang efektif kepada petani dalam konsultasi mengenai kemungkinan penyakit yang mempengaruhi tanaman karet mereka. Dengan peningkatan pengetahuan dan perhatian petani terhadap perkebunan karet, diharapkan produksi karet dapat meningkat.

Pada penelitian keempat yang dilakukan Desi (2019) dengan judul Penerapan Metode *Forward Chaining* Untuk Sistem Pakar Diagnosa Penyakit

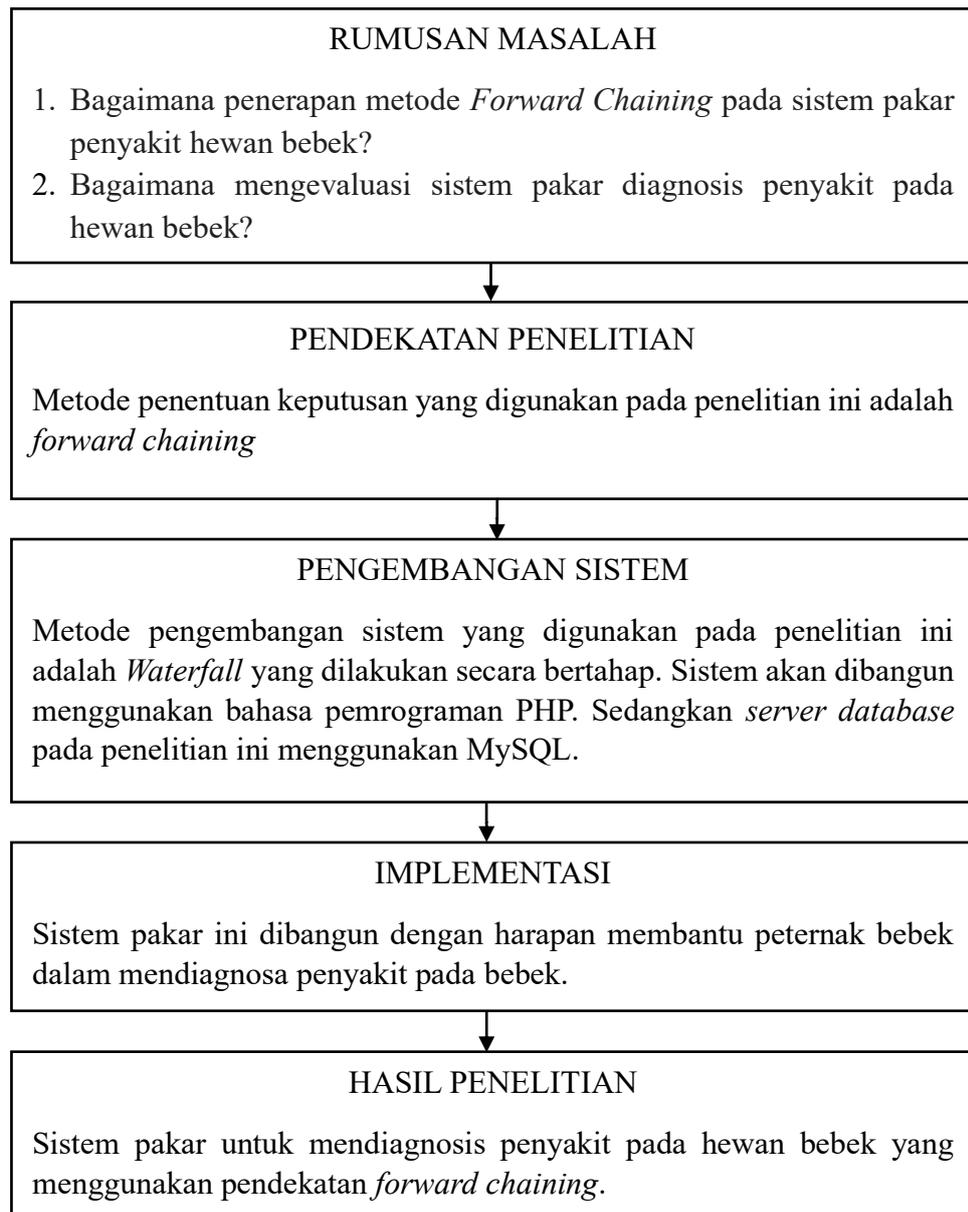
Rabies Pada Manusia dapat mengidentifikasi dan mendiagnosa penyakit rabies pada manusia berdasarkan gejalanya, memungkinkan penanganan yang cepat dan tepat.

Pada penelitian kelima yang dilakukan Seppewali, Mulyo, and Riswan (2023) dengan judul Sistem Pakar Diagnosa Kerusakan Motor Suzuki Smash Titan 115 Cc Menggunakan Metode *Forward Chaining* dapat disimpulkan berdasarkan deskripsi serta uraian yang telah disampaikan: 1. aplikasi pakar ini dirancang guna memberikan bantuan kepada pengguna dalam mengenali dan menyelesaikan masalah di kendaraan Suzuki Smash Titan 115 cc dengan cepat dan simpel, dengan tujuan meningkatkan efisiensi waktu dan mengurangi biaya perawatan. 2. Sistem ini efektif beroperasi menggunakan metode pencarian terbaik pertama dan pendekatan pelacakan maju.

Berdasarkan penelitian diatas menyimpulkan bahwa sistem pakar diagnosa penyakit menggunakan metode *forward chaining* terbukti efektif dalam membantu mendiagnosis penyakit pada hewan, tanaman, dan manusia. Sistem ini memiliki potensi untuk diterapkan secara luas dalam berbagai bidang untuk meningkatkan kesehatan dan kesejahteraan.

C. Kerangka Berfikir

Kerangka berpikir pada penelitian ini meliputi rumusan masalah, pendekatan penelitian, pengembangan sistem, mengimpletasikan sistem, hasil dan pembahasan. Kerangka berpikir penelitian ini dapat dilihat pada gambar 2.1



Gambar 2.1 Kerangka Berpikir