

## **BAB II**

### **KAJIAN PUSTAKA**

#### **A. Kajian Teoritis**

##### **1. Sistem Pakar**

Sistem pakar merupakan bagian dari ilmu komputer yang dirancang untuk mengadopsi proses berpikir dan pengambilan keputusan seperti yang dilakukan oleh manusia. Sistem ini mampu melaksanakan tugas yang umumnya dilakukan oleh seorang pakar dan menghasilkan output yang hampir sebanding dengan pemikiran manusia (Diana et al., 2023)

Pendapat lain yang dikemukakan oleh Asnawati, (2025) menjelaskan bahwa sistem pakar dimaksudkan untuk digunakan oleh orang-orang seperti masyarakat umum yang membutuhkan informasi untuk membuat keputusan tentang cara menyelesaikan suatu masalah. Sistem ini mengintegrasikan pengetahuan ahli ke dalam komputer berbasis pengetahuan untuk membantu menyelesaikan masalah dan menemukan solusi yang tepat.

Berdasarkan pendapat yang telah disampaikan sebelumnya, dapat disimpulkan bahwa sistem pakar adalah program komputer yang dirancang untuk meniru proses pengambilan keputusan yang umumnya dilakukan oleh seorang ahli di bidang tertentu. Sistem ini beroperasi dengan memanfaatkan pengetahuan serta analisis yang telah dirumuskan oleh pakar yang kompeten. Beberapa keuntungan utama

dari sistem pakar antara lain meningkatkan efisiensi dan efektivitas dalam proses pengambilan keputusan, memperluas akses terhadap keahlian para ahli, serta membantu dalam pendokumentasian dan pelestarian pengetahuan mereka.

## **2. Diagnosis Sindrom *Fear Of Missing Out* (FOMO)**

Pada dasarnya, FOMO adalah rasa takut dan cemas yang dirasakan seseorang ketika mereka merasa ada peristiwa menarik atau menyenangkan yang sedang atau akan terjadi di suatu tempat yang memicu keinginan kuat untuk berpartisipasi dalam peristiwa tersebut. FOMO lebih disebabkan oleh keinginan untuk mendapatkan manfaat atau kesenangan daripada khawatir tentang apa yang mungkin terlewatkan dari lingkungan sekitar (Carolina & Mahestu, 2020).

Menurut Nadzirah et al., (2022) bahwa meskipun FOMO sudah lama ada, perkembangan mereka masih sangat rendah karena teknologi belum secanggih sekarang. Saat ini, perkembangan teknologi dan media sosial, ditambah dengan kemudahan akses melalui smartphone, membuat individu yang kecanduan cenderung terus membawa ponsel ke mana pun mereka berada.

Kesimpulan dari pernyataan tersebut menunjukkan bahwa FOMO adalah suatu kondisi psikologis yang ditandai oleh rasa takut kehilangan pengalaman sosial yang dinilai menarik oleh lingkungan sekitar. Dorongan untuk memperoleh pengalaman yang dianggap berharga daripada kehilangan momen penting dari lingkungan sekitar mendorong fenomena

ini, dan penggunaan media sosial telah mempercepat dan menyebarkannya. Hal ini menyebabkan orang-orang yang mengalami FOMO cenderung menggunakan perangkat elektronik secara terus-menerus untuk menghindari perasaan tertinggal.

### 3. Forward Chaining

*Forward Chaining*, atau penalaran maju, merupakan metode inferensi yang dimulai dari data atau fakta yang telah tersedia. Proses ini dilakukan dengan mencocokkan fakta tersebut terhadap sejumlah aturan yang relevan, lalu menghasilkan informasi atau kesimpulan baru berdasarkan kecocokan tersebut. Proses ini terus berlangsung selama masih ada aturan yang relevan dengan fakta yang ada, atau sampai tercapai tujuan yang diinginkan. Metode ini disebut juga sebagai pendekatan yang digerakkan oleh data (*data-driven*), karena alur penelusurannya bergantung sepenuhnya pada informasi yang tersedia di awal (Amalia & Mahyuddin, 2023).

Dalam penelitian lain yang dilakukan oleh Tri et al., (2025) bahwa dalam *Forward Chaining*, pernyataan *IF-THEN* biasanya digunakan untuk menulis aturan. Jika kondisi (*IF*) dari sebuah aturan terpenuhi oleh fakta awal, maka tindakan atau konsekuensi (*THEN*) tersebut akan dijalankan. Metode *Forward Chaining* menggunakan pendekatan berbasis aturan untuk menerapkan sistem pakar dalam penelitian ini. Metode ini memformat *IF* gejala *THEN* diagnosis. Metode yang terorganisir ini memastikan bahwa sistem dapat menganalisis gejala secara sistematis dan menghasilkan diagnosis yang akurat.

Berdasarkan hasil penelitian, dapat disimpulkan bahwa metode *Forward Chaining* merupakan suatu pendekatan penalaran yang bekerja secara maju dari fakta-fakta awal menuju suatu kesimpulan, dengan cara mencocokkan fakta yang ada terhadap aturan-aturan dalam basis pengetahuan. Proses ini menggunakan format *IF-THEN* untuk menyusun aturan, sehingga jika kondisi awal terpenuhi, maka konsekuensi akan dijalankan. Metode ini sangat sesuai diterapkan dalam sistem pakar karena memungkinkan sistem untuk menganalisis data atau gejala secara sistematis dan menghasilkan kesimpulan atau diagnosis yang akurat. Keunggulan dari *Forward Chaining* terletak pada kemampuannya dalam menangani informasi yang tidak lengkap serta meningkatkan ketepatan dalam pengambilan keputusan berbasis data yang tersedia.

#### **4. Certainty Factor**

Metode *Certainty Factor* (CF) adalah nilai yang digunakan mengevaluasi keyakinan profesional. Untuk menunjukkan besarnya kepercayaan, Shortliffe Buchanan memperkenalkan CF selama pembuatan sistem pakar MYCIN. Nilai kepastian fakta (CF) menunjukkan tingkat keyakinan terhadap suatu fakta atau aturan. Nilai tertinggi CF adalah + 1,0, yang berarti pasti benar atau notasi hubungan, dan nilai terendah CF adalah - 1,0, yang berarti pasti salah atau notasi hubungan. Nilai positif menunjukkan derajat keyakinan, sedangkan nilai negatif menunjukkan derajat ketidakpercayaan (Anggara et al., 2024). Berikut adalah bobot nilai CF yang ditunjukkan pada gambar 2.1.

Tabel 2. 1 Bobot Nilai CF

<i><b>Keterangan</b></i>	<i><b>Nilai</b></i>
Sangat Yakin	<b>1</b>
Yakin	<b>0,8</b>
Cukup Yakin	<b>0,6</b>
Sedikit Yakin	<b>0,4</b>
Tidak Tahu	<b>0,2</b>
Tidak	<b>0</b>

Nilai kepastian yang digunakan pada aturan dengan satu pernyataan pendukung (Ajisari & Prasetyaningsrum, 2024):

$$CF_{gejala} = CF(\text{user}) * CF(\text{pakar}) \quad (1)$$

Keterangan:

$CF_{gejala}$  = Gabungan dari nilai  $CF(\text{user})$  dan  $zCF(\text{pakar})$ .

$CF(\text{user})$  = nilai yang diperoleh dari user.

$CF(\text{pakar})$  = nilai yang diperoleh dari pakar.

Jika ada beberapa gejala, faktor keyakinan untuk penyakit dihitung dengan menggunakan persamaan (2) berikut:

$$CF_{combine} = CF_{old} + CF_{gejala} * (1 - CF_{old}) \quad (2)$$

Keterangan:

$CF_{combine}$  = nilai kombinasi/akhir.

$CF_{old}$  = nilai gabungan sebelumnya (dari gejala-gejala yang telah dihitung).

Selanjutnya, presentase hasil dihitung dengan persamaan (3) dari hasil persamaan di atas.

$$\text{Presentase} = CF_{penyakit} * 100 \quad (3)$$

Keterangan:

CFpenyakit = nilai hasil gabungan dari semua gejala terkait penyakit.

Presentase = tingkat keyakinan akhir dalam bentuk persen.

## 5. Website

Website merupakan sekumpulan halaman yang berisi informasi yang relevan, kadang-kadang disertai dengan gambar, audio, video, animasi, teks, atau kombinasi dari semua ini (Sriadhi, 2021). Website berfungsi sebagai media untuk menyampaikan informasi, berkomunikasi, serta mendukung berbagai aktivitas seperti pendidikan, bisnis, dan layanan publik. Dengan kemampuannya menyajikan konten secara interaktif dan real-time, website menjadi bagian penting dalam pengembangan sistem informasi berbasis web.

### a. *Framework*

*Framework* adalah alat bantu yang digunakan oleh para pembuat aplikasi untuk mempermudah dan mempercepat proses pembuatan serta pengembangan aplikasi. Didalamnya sudah tersedia fungsi – fungsi dasar yang sering dipakai, sehingga pekerjaan menjadi lebih cepat (Suprayogi et al., 2023).

*Framework* merupakan kerangka kerja dalam pembuatan aplikasi yang memberikan struktur teratur dan memudahkan pengembangan, terutama dalam kerja tim. Dalam membuat website, perlu menyesuaikan dengan pola kerja yang telah ditetapkan oleh *framework* yang digunakan (Putri & Hendryadi, 2025).

Berdasarkan pendapat diatas, dapat disimpulkan bahwa *framework* adalah kerangka kerja yang dirancang untuk mempermudah dan mempercepat proses pengembangan aplikasi, baik desktop, mobile, maupun web. Dengan menyediakan struktur yang terorganisir dan kumpulan perintah serta fungsi dasar, *framework* membantu pengembang menghasilkan kode yang rapi serta memfasilitasi kerja tim. Penggunaan *framework* juga menuntut penyesuaian terhadap pola kerja tertentu yang telah ditetapkan, terutama dalam pengembangan website.

b. *Laravel*

*Laravel* adalah framework PHP terbaik yang dikembangkan oleh Taylor Otwell. Ini adalah framework web berbasis PHP yang tidak berbayar dan open-source yang dimaksudkan untuk membuat aplikasi web yang menggunakan pola MVC (Sinlae et al., 2024).

*Laravel* adalah *framework* yang dibuat untuk mengoptimalkan penggunaan bahasa pemrograman PHP saat membuat aplikasi website. *Laravel* menggunakan pola arsitektur Model-View-Controller (MVC), yang memisahkan logika aplikasi dari antarmuka pengguna. dalam pendekatan ini, aplikasi diklasifikasikan menjadi tiga bagian pokok: Model berperan dalam mengelola data, View berfungsi untuk menampilkan informasi kepada pengguna, dan Controller bertugas mengatur alur interaksi di antara komponen-komponen tersebut. Pola struktur ini menjadikan proses pengembangan aplikasi lebih

terorganisir, efisien, serta memudahkan dalam pemeliharaan dan pengembangan di masa mendatang. (Sweetania & Herawati, 2022).

Dari pendapat di atas, dapat disimpulkan bahwa *Laravel* adalah framework PHP yang populer yang memanfaatkan pola Model View Controller (MVC) untuk membuat website atau sistem dan menyediakan kerangka kerja terstruktur.

c. Laragon

Laragon merupakan platform server lokal yang memungkinkan pengguna menjalankan berbagai aplikasi secara virtual di komputer. Dengan antarmuka grafis yang intuitif, Laragon menawarkan kemudahan dalam membangun dan mengelola proyek lokal. Keunggulan utamanya terletak pada kelengkapan fitur yang melebihi banyak aplikasi sejenis. Laragon juga mendukung pengembangan aplikasi berbasis teknologi modern seperti Node.js, MongoDB, Python (termasuk Django dan Flask), PostgreSQL, Ruby, Java, serta integrasi dengan Google Cloud Platform (Sitorus et al., 2024).

d. PHP

PHP adalah salah satu bahasa pemrograman yang digunakan untuk memproses data di sisi server dan mengembalikan hasilnya kepada browser dalam bentuk tampilan berbasis HTML. Singkatan dari PHP adalah Hypertext Preprocessor, dan bahasa ini memiliki keunggulan dalam memanipulasi basis data secara efisien dan praktis (Arif, 2023).

Penggunaan PHP memungkinkan pengembangan situs web yang bersifat dinamis, sehingga pengelolaan dan pemeliharaan konten menjadi lebih fleksibel, mudah, dan efisien. Dengan kemampuan ini, perubahan atau pembaruan pada sistem dapat dilakukan tanpa harus mengubah seluruh struktur halaman secara manual (Ery Hartati, 2022).

Dari pendapat di atas, dapat disimpulkan bahwa PHP adalah bahasa pemrograman server-side yang bersifat open source dan digunakan untuk membangun situs web dinamis. PHP memungkinkan integrasi dengan HTML serta mendukung pengolahan data di sisi server, sehingga memudahkan dalam pengembangan dan pemeliharaan situs web yang interaktif dan efisien.

## **6. Database**

*Database* dapat diibaratkan sebagai wadah penyimpanan terstruktur yang berfungsi untuk menyimpan berbagai data secara sistematis agar mudah diakses dan dikelola. Data tersebut disimpan dalam media penyimpanan komputer dan diakses melalui perangkat lunak khusus. Dengan adanya database, proses pencarian, penyimpanan, serta pengelolaan data dapat dilakukan secara efisien, aman, dan terorganisir (Syahputri et al., 2023).

Tanpa perlu melakukan pengolahan data secara manual, pengguna kini dapat memanfaatkan DBMS. Sistem ini dilengkapi dengan fitur yang menjamin keamanan, konsistensi, dan keutuhan data. Dalam jumlah besar, data dapat disusun dan diakses melalui perangkat lunak bernama *Database*

*Management System* (DBMS) merupakan sistem yang dibuat secara khusus untuk menangani dan menyusun data. Istilah DBMS ini juga mengacu pada sistem yang mengelola basis data, yaitu kumpulan informasi yang saling berkaitan dan tersusun secara sistematis guna memenuhi kebutuhan informasi suatu organisasi, sebagaimana dijelaskan oleh (Kadir, 2020).

## 7. MySQL

MySQL merupakan sistem manajemen basis data (DBMS) yang bersifat open source dan memanfaatkan *Structured Query Language* (SQL) sebagai bahasa utama untuk pengolahan serta pengambilan data. Sistem ini banyak diminati karena antarmukanya yang user-friendly, kecepatan dalam menjalankan kueri, dan kemampuannya dalam menangani sistem dengan skala menengah. Selain itu, MySQL mendukung penggunaan multi-pengguna dalam jaringan, sehingga sangat cocok untuk pengelolaan basis data pada platform web maupun aplikasi daring. Sebagai perangkat lunak *open source*, MySQL dapat diunduh, dimodifikasi, dan dimanfaatkan secara gratis baik untuk keperluan individu maupun komersial. Kombinasi antara fleksibilitas dan kesesuaiannya dengan standar SQL menjadikan MySQL salah satu pilihan utama dalam manajemen data (Rusyadi, 2024).



MySQL merupakan database paling umum digunakan di situs web, dikembangkan menjadi cepat dan kecil, khusus untuk situs web. Sangat disukai untuk situs web yang ditulis dalam PHP karena PHP dan MySQL sangat bekerja sama (Silalahi, 2022).


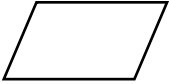
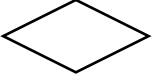
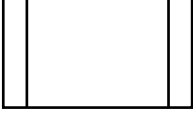

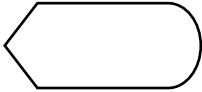

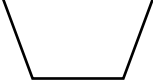
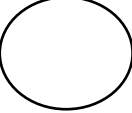
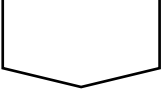
## 8. *Flowchart*

Menurut Tuasamu et al., (2023) menjelaskan bahwa bagan alur, juga dikenal sebagai *flowchart*, adalah teknik analitis bergambar yang digunakan untuk menjelaskan beberapa elemen sistem informasi secara jelas, ringkas, dan logis. Bagan alir mencatat bagaimana proses bisnis dilakukan dan bagaimana dokumen mengalir melalui organisasi. Gambar alir sistem, prosedur, dan pengendalian intern perusahaan digambarkan dalam *flowchart*.

*Flowchart* yaitu sebuah ilustrasi visual yang menyajikan tahapan-tahapan atau proses secara terstruktur dan berurutan dalam menjalankan suatu program. Dalam definisi lain, *flowchart* merupakan sebuah diagram yang menggunakan simbol-simbol grafis untuk merepresentasikan aliran dari suatu proses tertentu. Diagram ini menampilkan berbagai tahapan atau aksi yang digambarkan melalui simbol-simbol tersebut. Dengan kata lain, *flowchart* adalah gambaran visual dari urutan prosedur atau langkah-langkah dalam suatu program yang dirancang untuk mencapai tujuan (Listyoningrum et al., 2023).

Tabel 2. 2 Simbol Flowchart

BAGAN	NAMA	FUNGSI
	<i>Flow</i>	Menunjukkan arah alur dari proses dalam program.
	<i>Terminator</i>	Menandai titik awal atau akhir dari suatu program.

	<i>Process</i>	Menggambarkan suatu aktivitas atau operasi yang dilakukan.
	<i>Input / Output</i>	Menunjukkan proses pemasukan atau pengeluaran data.
	<i>Decision</i>	Menyatakan kondisi yang menentukan percabangan alur proses.
	<i>Predefine Process</i>	Mengindikasikan sebuah sub-rutin atau prosedur yang telah ditentukan sebelumnya.
	<i>Preparation</i>	Melambangkan proses inialisasi atau pemberian nilai awal.
	<i>Display</i>	Menyatakan perangkat atau media untuk menampilkan keluaran.
	<i>Document</i>	Menunjukkan data masukan atau keluaran dalam bentuk dokumen.
	<i>Manual Operation</i>	Menggambarkan aktivitas yang dilakukan secara manual.
	<i>On Page Connector</i>	Penghubung antar bagian dalam halaman yang sama.
	<i>Off page Connector</i>	Menghubungkan bagian alur yang terletak di halaman berbeda.

Sumber : Zalukhu et al., (2023)

## 9. *Unified Modeling Language (UML)*




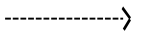
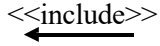
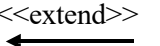
Dalam dunia pengembangan sistem yang berorientasi objek (object-oriented development), *Unified Modeling Language (UML)* merupakan alat bantu yang sangat krusial. *Unified Modeling Language (UML)* adalah bahasa pemodelan visual standar yang umum dipakai dalam rekayasa perangkat lunak untuk mendeskripsikan, menjabarkan, merancang, dan mendokumentasikan berbagai elemen dalam suatu sistem. Kelebihan UML terletak pada konsistensi notasinya yang mudah dipahami oleh berbagai pihak yang terlibat, seperti analis sistem, perancang, pengembang, hingga pengguna akhir atau klien (Puspita & Astriani, 2024).

Dalam penelitian lain yang dilakukan oleh Binangkit et al., (2023) *Unified Modeling Language (UML)* merupakan bahasa standar yang digunakan secara luas dalam mendeskripsikan, merancang, dan membangun sistem perangkat lunak. UML berperan sebagai metode pengembangan sistem berbasis objek serta menjadi sarana penting dalam menunjang proses perancangan sistem tersebut. Dalam penerapannya, UML menyediakan berbagai jenis diagram sebagai alat bantu, seperti use case diagram, activity diagram, sequence diagram, dan class diagram, yang masing-masing memiliki fungsi tersendiri dalam pemodelan sistem. Kerangka penyusun UML terdiri dari beberapa class.

a. Use case diagram

Use case diagram berguna untuk menguraikan, mengilustrasikan, dan mencatat perilaku suatu sistem. Mereka menjelaskan apa yang dapat dilakukan oleh aktor saat sistem beroperasi dan persyaratan yang dari sudut pandang aktor atau pengguna sistem, harus dipenuhi oleh sistem (Rura & Ardiansyah, 2024).

Tabel 2. 3 Simbol Use Case Diagram

No	Simbol	Nama	Keterangan
1		Aktor	Peran eksternal yang berinteraksi dengan sistem, biasanya digambarkan sebagai aktor.
2		<i>Use Case</i>	Fitur atau layanan sistem yang digunakan aktor, menggambarkan fungsi yang disediakan oleh sistem untuk memenuhi kebutuhan pengguna.
3		<i>Association</i>	Hubungan antara aktor dan use case.
4		Generalisasi	Hubungan pewarisan antara aktor yang bersifat umum dan khusus.
5		<i>Include</i>	Menandakan use case yang selalu dijalankan oleh use case lainnya.
6		<i>Extend</i>	Menggambarkan fitur tambahan yang diaktifkan apabila kondisi tertentu terpenuhi.



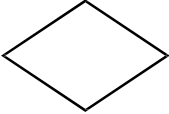


Sumber : Puspita et al., (2025)

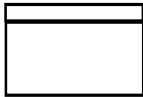
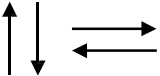
b. Activity diagram

Activity Diagram merupakan jenis diagram yang berfungsi untuk menggambarkan urutan aktivitas atau proses dalam suatu sistem.

Diagram ini berperan penting dalam membantu pengembang memahami serta memodelkan alur kerja atau proses bisnis secara sistematis dan terstruktur (Pranoto et al., 2024). Selain itu, diagram ini menggambarkan bagaimana setiap alur dimulai, kemungkinan kondisi yang muncul, serta bagaimana aktivitas tersebut diselesaikan.

Tabel 2. 4 Simbol Activity Diagram

No	Simbol	Nama	Keterangan
1		Status Awal	Melambangkan titik permulaan dalam sebuah diagram aktivitas yang menunjukkan dimulainya proses.
2		Aktivitas	Menunjukkan suatu tindakan atau proses yang dilakukan oleh sistem, biasanya diawali dengan kata kerja.
3		<i>Decision</i>	Menyatakan titik percabangan dalam alur proses, di mana terdapat lebih dari satu opsi aktivitas yang dapat dipilih.
4		<i>Join</i>	Mengilustrasikan proses penggabungan dua atau lebih aktivitas menjadi satu alur kegiatan.
5		Status Akhir	Merepresentasikan titik penyelesaian dalam diagram aktivitas yang menandakan berakhirnya proses sistem.

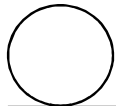
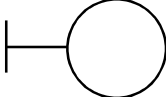
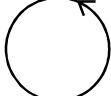
6		<i>Swimlane</i>	Menyediakan representasi visual untuk membedakan peran atau tanggung jawab antar entitas dalam suatu organisasi berdasarkan aktivitas yang dilakukan.
7		<i>Line connector</i>	Menunjukkan hubungan atau keterkaitan antara elemen-elemen dalam diagram alur aktivitas.

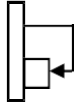
Sumber : Ramdany, (2024)

### c. Sequence diagram

Sequence diagram adalah jenis diagram UML yang menunjukkan bagaimana objek berinteraksi dalam suatu sistem. Perangkat lunak dalam ini biasanya digunakan untuk memvisualisasikan aliran pesan antar objek. Memahami perilaku dan interaksi suatu sistem lebih mudah dengan diagram ini (Mulyati et al., 2024).

Tabel 2. 5 Simbol Sequence Diagram

No	Simbol	Nama	Keterangan
1		<i>Entity Class</i>	Simbol yang menggambarkan sistem sebagai landasan dalam menyusun basis data, merepresentasikan proses utama dalam pemodelan data.
2		<i>Boundary Class</i>	Simbol yang menangani komunikasi antar lingkungan sistem, sebagai jembatan informasi antara sistem internal dan eksternal, sebagai jembatan informasi antara sistem internal dan eksternal.
3		<i>Control Class</i>	Simbol yang bertanggung jawab terhadap class objek yang berisi logika, entitas

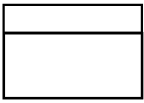
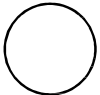

			pengelola operasi atau fungsi dalam sistem.
4		<i>Recursive</i>	Simbol yang melambangkan pesan untuk dirinya, menandakan pemanggilan metode internal dalam objek yang sama.



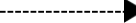

Sumber : Rasiban et al., (2024)

#### d. Class diagram

Class diagram merepresentasikan relasi antar kelas serta memberikan gambaran terperinci mengenai setiap kelas yang ada dalam rancangan suatu sistem. Diagram ini juga menjelaskan peran, tanggung jawab, dan aturan yang dimiliki oleh masing-masing entitas, yang secara langsung memengaruhi cara kerja sistem secara keseluruhan.. Oleh karena itu, class diagram merupakan representasi visual dari struktur sistem program berdasarkan jenis-jenis yang dibentuk. Class diagram juga berfungsi sebagai alur jalan database ke sistem yang akan dibangun atau dibuat.

Tabel 2. 6 Simbol Class Diagram

No	Simbol	Nama	Keterangan
1		<i>Class</i>	Struktur utama untuk memodelkan objek dalam sistem.
2		<i>Interface</i>	Merupakan representasi antarmuka dalam pemrograman berorientasi objek, yang harus diimplementasikan oleh kelas.
3		<i>Association</i>	Hubungan umum antar kelas, sering disertai multiplicity untuk menunjukkan jumlah

			objek yang terlibat dalam relasi.
4		<i>Direct Association</i>	Menunjukkan hubungan antara satu kelas dengan kelas lainnya yang memanfaatkannya; hubungan ini sering disertai dengan multiplicity.
5		<i>Generalisasi</i>	Mengilustrasikan relasi pewarisan antar kelas, dari yang bersifat umum ke yang lebih spesifik (spesialisasi).
6		<i>Dependency</i>	Menandakan adanya ketergantungan satu kelas terhadap kelas lainnya dalam sistem.
7		<i>Aggregation</i>	Menjelaskan relasi bagian-keseluruhan (whole-part) antara kelas-kelas dalam sistem.

Sumber : Ramdany (2024)

## B. Kajian Empiris

Dalam menyusun penelitian ini, penulis merujuk pada beberapa jurnal yang memiliki keterkaitan dengan topik yang dibahas. Oleh karena itu, penulis mengangkat tiga hasil penelitian terdahulu yang dianggap relevan sebagai landasan referensi:

1. Pada penelitian yang dilakukan oleh Sholeha et al., (2023) Pengembangan sebuah sistem pakar yang dirancang untuk mendeteksi gangguan psikologis pada usia remaja.. Sistem ini bertujuan untuk membantu mendiagnosis gangguan kesehatan mental pada remaja di Kalimantan Selatan menggunakan metode *Forward Chaining* dan *Certainty Factor*. Sistem dirancang berdasarkan data dari seorang psikolog dan mengidentifikasi

tingkat depresi (normal, ringan, sedang, berat) melalui gejala yang dipilih pengguna. Hasil validasi menunjukkan bahwa sistem memiliki tingkat akurasi sebesar 86,67%, di mana dari 15 data uji, 13 di antaranya sesuai dengan hasil diagnosis pakar. Dengan demikian, sistem ini berfungsi sebagai alat deteksi dini yang efisien dan praktis untuk mengidentifikasi permasalahan kesehatan mental pada remaja.

2. Penelitian serupa dilakukan oleh Fernando et al., (2024) yang mengembangkan sistem pakar untuk diagnosis gangguan kepribadian dengan menggunakan metode *Forward Chaining* dan *Certainty Factor*, dengan tujuan membantu masyarakat melakukan diagnosis secara mandiri tanpa harus berkonsultasi langsung dengan psikolog yang membutuhkan biaya besar. Sistem ini dirancang untuk mengidentifikasi 10 jenis gangguan kepribadian berdasarkan 79 gejala yang dianalisis, dengan nilai kepastian yang diperoleh dari pakar untuk meningkatkan akurasi hasil diagnosis. Berdasarkan pengujian menggunakan data rekam medis dari 10 pasien, sistem menunjukkan tingkat akurasi 98,1% dalam mendiagnosis gangguan kepribadian antisosial dan akurasi di atas 90% untuk jenis gangguan lainnya. Hasil ini menunjukkan bahwa sistem yang dikembangkan mampu memberikan kontribusi signifikan dalam deteksi dini gangguan kepribadian dan mengurangi beban biaya psikologis.
3. Penelitian oleh Agave & Ulum, (2023) yang berjudul “Diagnosa Penyakit Ibu Hamil Menggunakan Metode *Forward Chaining* Berbasis Website” bertujuan untuk mengatasi permasalahan kurangnya informasi dan

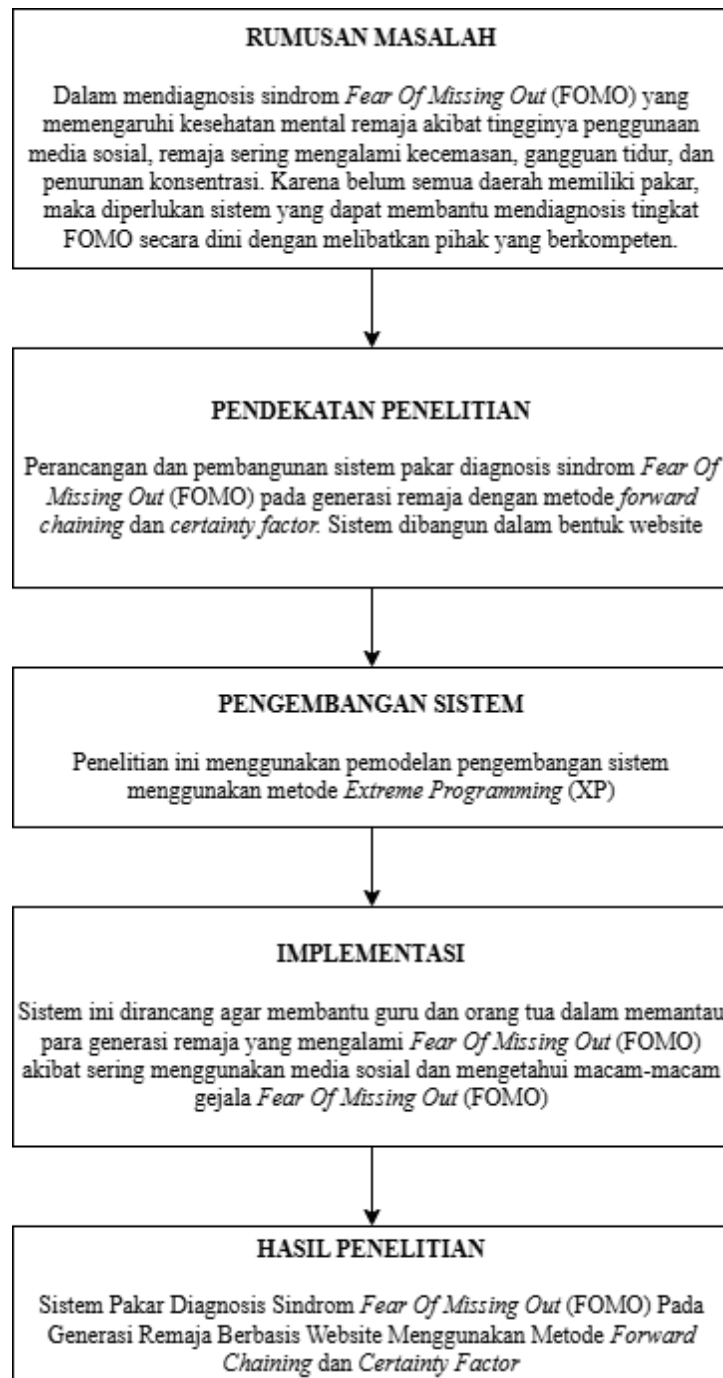
keterlambatan diagnosis penyakit pada ibu hamil, yang sering berujung pada meningkatnya Angka Kematian Ibu (AKI). Sistem pakar yang dikembangkan berbasis metode Forward Chaining dengan dukungan teknik pencarian Depth First Search, yang dikendalikan oleh data gejala yang diperoleh dari wawancara langsung dengan dua pakar medis. Pengembangan sistem dilakukan menggunakan metode Extreme Programming dengan pemrograman PHP native. Basis pengetahuan yang dibangun terdiri dari 6 jenis penyakit, 30 gejala, dan 6 solusi, serta dirumuskan dalam bentuk aturan (rule) dan pohon keputusan. Hasil pengujian menunjukkan bahwa sistem mampu melakukan diagnosis dini dengan akurasi mencapai 80%, menunjukkan performa sistem yang cukup baik dalam membantu proses identifikasi dini penyakit kehamilan.

Dari beberapa sumber literatur diatas, maka penggunaan metode yang tepat dalam proses pembangunan sistem sangat diperlukan. Dalam proses pengembangan, peneliti menerapkan metode *Extreme Programming* dengan arsitektur Model-View-Controller berbasis framework Laravel, serta mengintegrasikan algoritma *Forward Chaining* dan *Certainty Factor* yang dianggap sesuai untuk membangun sistem pakar diagnosis sindrom *Fear Of Missing Out* (FOMO). Walaupun sistem ini mampu memberikan diagnosis secara cepat, hasil yang diperoleh tetap memerlukan verifikasi oleh pakar, terutama apabila tingkat kepastian berada di bawah ambang batas yang ditentukan, guna memastikan ketepatan penanganan dan meminimalkan risiko kesalahan. Secara keseluruhan, temuan dalam penelitian ini mengindikasikan

bahwa sistem pakar berbasis web memiliki potensi signifikan dalam menyajikan informasi yang terstruktur, mempercepat proses pengambilan keputusan, serta memberikan alternatif solusi diagnosis yang terarah.

### C. Kerangka Berfikir

Penjelasan berikut menyajikan kerangka berpikir yang digunakan dalam pelaksanaan penelitian ini.



Gambar 2. 1 Gambar Kerangka Berfikir