

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Tanaman hortikultura merupakan salah satu komoditi pertanian yang memiliki tingkat konsumsi cukup tinggi dimasyarakat. Berdasarkan data Ditjen Horti (2013), permintaan pasar terhadap produk hortikultura mencapai rata-rata pertumbuhan 11% atau mengalahkan komoditas pertanian lain (perkebunan dan pangan) yang hanya 7-8%. Ada sekitar 90 dari 323 jenis produk hortikultura yang telah dikomersialkan secara luas, dengan rincian 25 jenis sayuran, 26 jenis buahbuahan, 24 jenis tanaman hias dan 15 jenis tanaman biofarmaka. Buah-buahan memiliki potensi pengembangan lebih tinggi dibandingkan jenis lainnya, sehingga menarik minat masyarakat untuk melakukan budi daya tanaman penghasil buah. Semangka (*Citrullus vulgaris* L.) adalah tanaman hortikultura dari famili Cucurbitaceae yang diminati karena rasa buah yang manis dengan kandungan air yang cukup banyak, warna daging buah yang cerah, konsistensi yang remah, dan kandungan gizi yang baik bagi kesehatan. Menurut Solihah (2015), keberadaan asam amino sitrulin, kalium, air, vitamin C, vitamin A (karotenoid), dan vitamin K pada buah semangka terbukti efektif dimanfaatkan dalam pengobatan alternatif untuk penderita hipertensi. Kondisi lingkungan tumbuh yang beragam merupakan kendala dalam kegiatan budi daya tanaman semangka. Tanah sebagai media penanaman yang paling sering digunakan, terdiri dari banyak jenis dengan tingkat kualitas dan kesuburan tanah yang berbeda(Trizayuni et al., 2021)

Proses pembudidayaan tanaman semangka juga tidak terlepas dengan persoalan penyakit. Terbatasnya pengetahuan dan kurangnya pemahaman mengenai cara penanganan secara tepat mengidentifikasi penyakit tanaman semangka sering mengakibatkan pertumbuhan tanaman semangka kurang maksimal, sehingga hasil panen pun kurang memuaskan bahkan bisa mengakibatkan gagal panen. Kondisi ini merupakan persoalan yang sering dialami masyarakat, tidak hanya para petani tanaman semangka tetapi untuk siapa saja yang ingin melakukan budidaya tanaman semangka.

Kenyataan ini membutuhkan seorang pakar untuk mendiagnosis/mengidentifikasi dalam menentukan jenis penyakit dan cara pengendaliannya agar memperoleh tindakan pencegahan dan penanganan yang tepat. Namun demikian, keterbatasan jumlah pakar dan kondisi lokasi sering menjadi kendala. Untuk itu keberadaan sistem pakar bisa menjadi alternatif penting dalam mengatasi persoalan yang terjadi.

Pada Balai Benih Umum (BBU) melakukan 10 kali penanaman benih tanaman semangka dengan tingkat keberhasilan mencapai angka 80% dari 3000 benih tanaman semangka yang ditanam. Maka dapat disimpulkan jumlah tanaman semangka yang gagal berkisar 20%. Penyakit merupakan faktor utama gagalnya panen pada tanaman semangka di Balai Benih Umum (BBU) Desa Talikumain Tambusai.

Sistem Pakar adalah sistem yang menggunakan pengetahuan manusia yang dimasukkan ke dalam komputer untuk memecahkan masalah-masalah yang biasanya diselesaikan oleh pakar. Sistem pakar dapat mengumpulkan dan menyimpan

pengetahuan seorang pakar atau beberapa orang pakar dalam sebuah perangkat lunak Komputer. Selanjutnya pengetahuan tersebut direpresentasikan dalam format tertentu dan dihimpun dalam suatu basis pengetahuan (Hafiz Firdaus & Agung Susilo Yuda Irawan, 2023).

Berdasarkan latar belakang di atas maka penulis tertarik mengangkat penelitian dengan judul **“Sistem Pakar Diagnosis Penyakit Tanaman Semangka Menggunakan Metode *Case Based Reasoning* Berbasis Web”**.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan maka didapatkan rumusan masalah sebagai berikut:

1. Bagaimana membantu petani di Balai Benih Umum (BBU) Desa Talikumain Tambusai untuk mendeteksi penyakit pada tanaman semangka ?
2. Bagaimana menghasilkan aplikasi sistem pakar untuk mendeteksi penyakit pada tanaman semangka menggunakan metode *Case Based Reasoning* (CBR) di Balai Benih Umum (BBU) Desa Talikumain Tambusai ?
3. Bagaimana menerapkan metode *Case Based Reasoning* (CBR) dalam mendeteksi penyakit pada tanaman semangka ?

1.3 Ruang Lingkup Permasalahan

Adapun ruang lingkup permasalahan penelitian ini adalah sebagai berikut :

1. Penelitian ini hanya digunakan untuk membantu mendiagnosis penyakit tanaman semangka menggunakan Metode *Case Based Reasoning*.

2. Penyakit yang didiagnosis sebanyak 6 jenis penyakit, 20 gejala dan solusi perawatan.
3. Variabel yang digunakan adalah Layu Fusarium, Bercak Daun, *Altermaria Leaf Spot*, Busuk Pangkal Batang, Embun Tepungdan Embun Bulu.
4. Sistem pakar dibangun dengan menggunakan bahasa pemrograman PHP dan *database MYSQL*

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian dalam tugas akhir ini adalah :

1. Membantu masyarakat dan petanisemangka dalam melakukan mendeteksi penyakit yang menyerang tanaman semangka dan solusi penanganannya.
2. Menghasilkan suatu sistem aplikasi deteksi penyakit tanaman semangkamenggunakan metode *Case Based Reasoning* (CBR) di Balai Benih Umum (BBU) Desa Talikumain Tambusai
3. Menerapkan metode *Case Based Reasoning* (CBR) dalam mendeteksi penyakit pada tanaman semangka.

1.5 Manfaat Penelitian

- a. Manfaat bagi Instansi Balai Benih Umum Desa Talikumain Tambusai
 1. Untuk mempermudah dan mempercepat petani dalam melakukan diagnosis awal terhadap penyakit yang menyerang tanaman semangka.
 2. Sebagai informasi bagi masyarakat dan petani semangka mengenai permasalahan penyakit yang menyerang tanaman semangka.

b. Manfaat bagi Mahasiswa (Peneliti)

1. Untuk mengetahui bagaimana menganalisis dan membangun sistem pakar dengan menggunakan metode *Case Based Reasoning* dalam memberikan hasil diagnosa awal penyakit yang diderita beserta keterangan dan solusi yang benar dan tepat.
2. Dapat menambah wawasan tentang penyakit dan solusi pada tanaman semangka dan pengalaman terjun langsung ke perkebunan petani.

1.6 Metode Pengumpulan Data

Metode yang dilakukan penulis untuk mengumpulkan data-data dalam penelitian ini diantaranya :

1. Observasi

Observasi adalah suatu metode mengumpulkan data dengan melakukan pengamatan langsung pada objek yang diteliti. Metode ini bertujuan untuk dapat mengetahui langsung bagaimana alur kerja yang terjadi pada objek yang diteliti.

2. Wawancara

Proses wawancara dilakukan langsung kepada Balai Benih Umum tentang penyakit tanaman semangka untuk mendapatkan jenis penyakit, gejala, dan nilai kepastian dari metode *Case Based Reasoning* pada setiap gejala untuk menentukan jenis penyakit.

3. Studi Pustaka

Studi pustaka dilakukan untuk mendapatkan teori serta konsep yang mendukung dalam penelitian dan berkaitan dengan masalah yang diangkat dalam penelitian. Hal dipelajari dalam studi pustaka antara lain definisi sistem pakar, penggunaan *Case*

Based Reasoning, dan metode yang digunakan untuk kasus mendiagnosis penyakit tanaman semangka dengan membaca buku, jurnal, dan referensi yang terkait sehingga memudahkan dalam menyelesaikan permasalahan yang ada.

1.7 Sistematika Penulisan

Sistematika penulisan Tugas Akhir ini disusun untuk memberikan gambaran umum tentang penelitian yang dijalankan. Sistematika penulisan Tugas Akhir ini adalah sebagai berikut :

BAB 1. PENDAHULUAN

Berisi uraian tentang latar belakang penelitian, rumusan masalah, ruang lingkup masalah, tujuan dan manfaat penelitian, metode pengumpulan data dan sistematika penulisan.

BAB 2. LANDASAN TEORI

Bagian ini membahas teori atau gambaran umum serta kebutuhan sistem yang berkaitan dengan sistem pakar (*expert system*) serta metode *Case Based Reasoning*.

BAB 3. METODE PENELITIAN

Bab ini akan membahas mengenai identifikasi masalah, pengumpulan data, studi literatur, analisis dan perhitungan metode *Case Based Reasoning*, perancangan sistem, implementasi serta pengujian sistem.

BAB 4. ANALISIS DAN PERANCANGAN

Berisi pembahasan tentang metode analisis yang akan digunakan, analisis sistem, tahapan pencarian solusi memakai UML dan perancangan sistem.

BAB 5. IMPEMANTASI DAN PENGUJIAN

Bab ini berisi penjelasan mengenai implementasi sistem pakar penentuan jenis penyakit, lingkungan implementasi, batasan implementasi, analisis hasil, pengujian sistem, hasil pengujian dan kesimpulan pengujian yang telah dirancang pada bab sebelumnya.

BAB 6. KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan tentang aplikasi yang telah dibuat, dan saran untuk pihak lain yang ingin mengembangkan aplikasi ini atau memiliki masalah yang sama.

BAB 2

LANDASAN TEORI

2.1 Sistem

Sistem merupakan sebagai satu kesatuan komponen yang satu sama lain saling berhubungan untuk mencapai suatu hasil yang diharapkan secara optimal sesuai dengan tujuan (Purwaningsih et al., 2022). Pengertian sistem adalah sebagai berikut:

1. Sistem adalah suatu kebulatan keseluruhan yang kompleks atau terorganisir, suatu himpunan atau perpaduan hal-hal atau bagian-bagian yang membentuk suatu kebulatan atau keseluruhan yang kompleks atau utuh.
2. Sistem merupakan himpunan komponen yang saling berkaitan yang Bersama-sama berfungsi untuk mencapai suatu tujuan.
3. Sistem merupakan sehimpunan komponen atau subsistem yang terorganisasikan dan berkaitan sesuai dengan rencana untuk mencapai tujuan tertentu.

Kata sistem berasal dari bahasa Yunani yang mengandung arti “Systema” yang berarti kesatuan atau kumpulan. Ditinjau dari perkataan kata, sistem berarti sekumpulan objek yang bekerja bersama-sama untuk menghasilkan kesatuan metode, prosedur atau teknik yang digabungkan dan diatur sedemikian rupa sehingga menjadi satu kesatuan yang berfungsi untuk mencapai tujuan.

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai komponen-komponen, batasan sistem, lingkungan luar sistem, penghubung, masukan, keluaran pengolah dan sasaran atau tujuan (Effendy et al, 2023).

1. Komponen (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem.

2. Batasan sistem (*Boundary*)

Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan dan menunjukkan ruang lingkup dari sistem tersebut.

3. Lingkungan luar sistem (*Environment*)

Lingkungan luar sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan juga merugikan.

4. Penghubung sistem (*Interface*)

Penghubung sistem merupakan media yang menghubungkan antara suatu subsistem dengan yang lainnya. Melalui penghubung ini kemungkinan sumber-sumber daya mengalir dari suatu subsistem ke subsistem yang lainnya.

5. Masukan sistem (*Input*)

Masukan sistem adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan dan masukan sinyal *maintenance* input adalah energi yang dimasukkan supaya sistem tersebut dapat berjalan.

6. Keluaran sistem (*Output*)

Keluaran sistem adalah energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran dapat merupakan masukan untuk subsistem yang lain.

7. Pengolahan sistem (*Process*)

Pengolahan sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan merubah masukan menjadi keluaran.

8. Sasaran sistem (*Objective*)

Suatu sasaran sistem mempunyai tujuan atau sasaran, kalau sistem tidak mempunyai sasaran maka sistem tidak akan ada. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya. Sasaran sangat berpengaruh pada masukan dan keluaran yang dihasilkan.

2.2 Pakar

Pakar adalah seseorang yang memiliki, melibatkan, atau menampilkan keterampilan atau pengetahuan khusus yang diperoleh dari pelatihan atau pengalaman. Istilah keahlian sering digunakan dalam arti relatif dalam penelitian pendidikan. Keahlian dapat didefinisikan dalam hal kinerja luar biasa dalam suatu domain; penelitian kinerja ahli menyelidiki kinerja unggul secara konsisten dari individu yang unggul dalam tugas perwakilan dalam domain itu (Fadhilah et al., 2023).

Pakar atau biasa juga disebut sebagai ahli pada bidangnya, ialah seseorang yang dianggap sebagai sumber terpercaya atas teknik atau keahlian tertentu yang

dimilikinya serta keampuannya untuk menilai dan memberikan pandangan atau pendapat serta memutuskan sesuatu isu atau masalah dengan benar, dengan baik dan terpercaya sesuai dengan aturan atau kaidah dalam bidang tertentu (Firdaus, 2020).

2.3 Sistem Pakar

Sistem pakar (expert system) merupakan sistem yang berusaha untuk mengadopsi kemampuan atau pengetahuan manusia ke dalam komputer, agar komputer dapat bekerja dalam menyelesaikan suatu masalah seperti layaknya seorang pakar atau seseorang yang mempunyai keahlian dalam bidang tertentu, yaitu pakar yang mempunyai knowledge atau kemampuan khusus yang tidak diketahui dan dimiliki oleh orang lain. Sistem pakar merupakan cabang dari *Artificial Intelligence* (AI) (Sinaga, 2021).

Sistem pakar merupakan salah satu cabang dari *Artificial Intelligence* (AI) yang didefinisikan sebagai suatu program komputer yang memanfaatkan sekumpulan pengetahuan (*knowledge*) untuk domain tertentu yang diharapkan dapat memberikan solusi/jawaban dari suatu masalah dengan mencoba meniru proses pengambilan keputusan oleh seorang pakar dalam suatu bidang keahlian tertentu (Brilliant & Nizamiyati, 2022).

Sistem pakar merupakan suatu program komputer cerdas yang memanfaatkan pengetahuan (*knowledge*) dan prosedur inferensi untuk memecahkan masalah yang cukup sulit sehingga dibutuhkan seorang yang ahli untuk memecahkannya.

Pengetahuan merupakan sebuah kekuatan yang dapat menyelesaikan suatu masalah yang ditemui sehari-hari (Harahap & Sinaga, 2020).

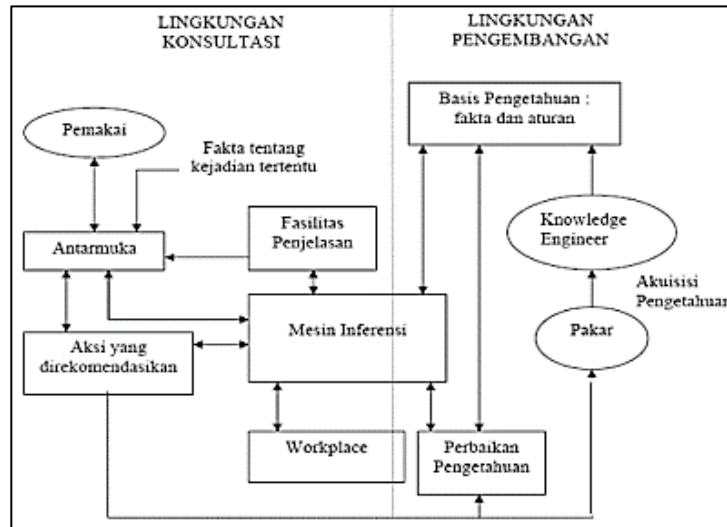
2.3.1 Struktur Sistem Pakar

Struktur sistem pakar terdiri dari beberapa tahapan yang meliputi:

Membuat basis pengetahuan yang mampu menampung data gejala-gejala dan penyebab penyakit tanaman semangka. Membangun sistem pengetahuan untuk menganalisis suatu masalah tertentu yang selanjutnya akan mencari jenis penyakit.

Merancang antarmuka pemakai yang dapat menjangkau semua kebutuhan user tanpa mempersulit atau membingungkan user dalam menggunakan sistem. Dengan adanya tahapan analisis sistem, dapat ditentukan kebutuhan sistem sehingga dapat mendiagnosis penyakit tanaman semangka dengan menggunakan metode *Case Based Reasoning*.

Sistem pakar ini juga memungkinkan pakar untuk melakukan hal-hal seperti login, memasukkan data gejala dan jenis penyakit, mencocokkan data jenis penyakit berdasarkan gejalanya, mencocokkan data yang ada pada basis pengetahuan, memasukkan data solusi atau mengupdatenya, dan melakukan logout, sedangkan user hanya mencocokkan data gejala dan melihat hasil diagnosis. Berikut gambar strukturnya :



Gambar 2.1 Struktur Sistem Pakar(Kurniawati, 2021)

Adapun komponen yang terdapat dalam struktur sistem pakar ini adalah :

a. *Knowledge Base* (Basis Pengetahuan)

Basis Membuat basis pengetahuan yang mampu menampung data gejala-gejala dan penyebab penyakit tanaman semangka.

b. *Inference Engine* (mesin Inferensi)

Merupakan otak dari sebuah sistem pakar dan dikenal juga dengan sebutan *control structure* (struktur kontrol) atau *rule interpreter* (dalam sebuah sistem pakar berbasis kaidah) *Forward Chaining*.

c. *Working Memory*

Berguna untuk menyimpan fakta yang dihasilkan oleh inference engine dengan penambahan parameter berupa derajat kepercayaan atau dapat juga

dikatakan sebagai global database dari fakta yang digunakan oleh rule-rule yang ada.

d. *Explanation Facility*

Menyediakan kebenaran dari solusi yang dihasilkan kepada user (*reasoningchain*).

e. *Knowledge Acquisition Facility*

Meliputi proses pengumpulan, pemindahan dan perubahan dari kemampuan pemecahan masalah seorang pakar atau sumber pengetahuan terdokumentasi keprogram sebuah komputer, dengan tujuan untuk memperbaiki atau mengembangkan basis pengetahuan.

f. *User Interface*

Mekanisme untuk memberi kesempatan kepada user dan sistem pakar untuk berkomunikasi antar muka, menerima informasi dari pemakai dan mengubahnya ke dalam bentuk yang dapat diterima oleh system (Kurniawati, 2021).

2.4 Manfaat dan Kekurangan Sistem Pakar

Menurut Naufal Rifqi & Agus Iskandar (2023), sistem pakar mempunyai manfaat dan kekurangan yang diberikannya, diantaranya :

2.4.1 Manfaat Sistem Pakar

1. Memberikan pengambilan keputusan yang lebih baik, karena sistem pakar memberikan jawaban yang konsisten dan logis dari waktu ke waktu.

Jawaban yang diberikan logis karena alasan logikanya dapat diberikan oleh sistem pakar dalam proses konsultasi.

2. Memberikan solusi tepat waktu, kadang kala seorang manajer membutuhkan jawaban dari pakar, tetapi pakar yang dibutuhkan tidak berada ditempat, sehingga keputusan menjadi terlambat. Dengan sistem pakar, jawaban yang dibutuhkan oleh pengambil keputusan selalu tersedia setiap saat dibutuhkan.
3. Menyimpan pengetahuan di organisasi, pengetahuan pakar merupakan hal yang penting dan kadang kala pengetahuan ini akan hilang jika pakar keluar atau telah pensiun dari perusahaan. Dengan sistem pakar, pengetahuan dari pakar dapat disimpan di sistem pakar dan tersedia terus selama dibutuhkan.

2.4.2 Kekurangan Sistem Pakar

1. Dalam Sistem pakar hanya dapat menangani pengetahuan yang konsisten, sistem pakar dirancang dengan aturan-aturan yang hasilnya sudah pasti dan konsisten sesuai dengan alur di diagram pohonnya. Untuk pengetahuan yang cepat berubah-ubah dari waktu ke waktu, maka *knowledge base* di sistem pakar harus selalu diubah (*perbarui*), yang tentu cukup merepotkan.
2. Sistem pakar tidak dapat menangani hal yang bersifat *judgement*, sistem pakar memberikan hasil yang pasti, sehingga keputusan akhir pengambilan keputusan jika melibatkan kebijaksanaan dan institusi masih tetap di tangan manajemen.
3. Format *knowledge base* sistem pakar terbatas, *knowledge base* pada sistem pakar berisi aturan-aturan (*rules*) yang ditulis dalam bentuk statemen if-then.

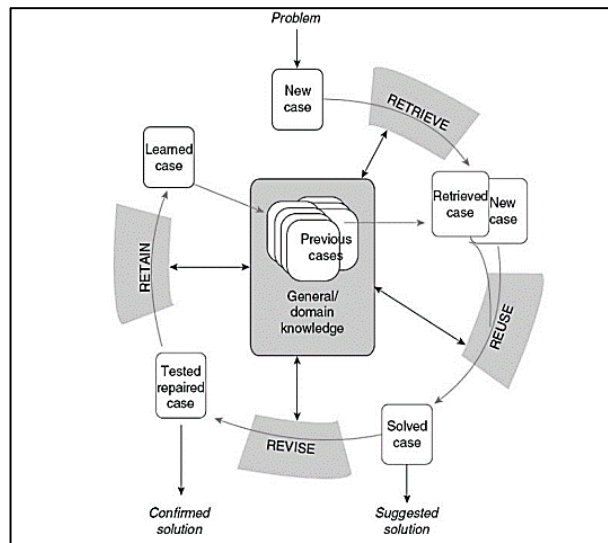
2.5 Metode *Case Based Reasoning*

Penelitian yang dilakukan penulis dalam bentuk studi kasus. Metode yang digunakan peneliti dalam kasus ini adalah metode eksperimental dengan membuat perangkat lunak case based reasoning. Pengumpulan data dalam penelitian ini berasal dari pengamatan langsung dilapangan serta hasil wawancara dengan yang ahli atau pakar semangka. *SDLC (System Development Life Cycle)* digunakan untuk analisis dan perancangan perangkat lunak.

Berikutnya tahap desain yaitu melakukan desain sistem yakni desain Output, desain Input, desain Database, desain teknologi dan desain model. Tahap produksi/pembuatan yaitu melakukan pembuatan sistem menggunakan PHP sebagai Bahasa Pemrograman dengan memanfaatkan database MySQL. Tahap terakhir adalah melakukan tahap pengujian, dimana seluruh perangkat lunak, program tambahan dan semua program yang terlibat dalam pembangunan sistem diuji untuk memastikan sistem dapat berjalan dengan semestinya.

Pada penalaran *Case Base Reasoning*, suatu basis kasus berisi kasus-kasus dengan solusi yang telah dicapai. Untuk menemukan solusi dari sebuah kasus baru yang diberikan, sistem akan mencari kasus-kasus dalam basis kasus yang memiliki tingkat kesamaan yang paling tinggi. *CBR* mengumpulkan kasus sebelumnya yang hampir sama dengan masalah yang baru dan berusaha untuk memodifikasi solusi agar sesuai dengan kasus yang baru (Aamodt & Plaza, 2020).

Ide dasar dari *CBR* adalah asumsi bahwa permasalahan yang serupa mempunyai solusi serupa. *CBR* terdiri dari atas empat langkah utama, yaitu: *Retrieve*, *Reuse*, *Revise*, dan *Retain*.



Gambar 2.2 Siklus Case Based Reasoning(Wahanani et al., 2020)

2.5.1 Kelebihan Dan Kekurangan Metode Case Based Reasoning

Menurut Esthi et., al (2024) Adapun kelebihan dan kekurangan metode *Case Based Reasoning* dalam proses penelitian adalah :

a, Kelebihan

1. Dapat memecahkan permasalahan dengan mudah.
2. Semakin banyak pengalaman, sistem akan menjadi lebih pintar.

Sehingga dapat memecahkan permasalahan dengan mudah..

b. Kekurangan

1. Tidak menjamin solusi terbaik atau optimum karena penalaran ini berdasarkan kasus terdahulu, jika solusi terdahulu salah maka tahapan revise sangat dibutuhkan untuk mengurangi tingkah kesalahan.
2. Semakin banyak pengalaman/*case*, proses pencarian semakin lama karena harus membandingkan dengan *case* yang paling mirip.

2.6 Bahasa Pemrograman

Dalam perancangan suatu sistem, dibutuhkan beberapa bahasa pemrograman perancangan sistem agar analisis dan hasil yang ingin dicapai dapat mencapai sebuah hasil yang maksimal.

2.6.1 Hypertext Preprocessor(PHP)

PHP (Hypertext Preprocessor) itu bahasa pemrograman berbasis web. Jadi, *PHP* adalah bahasa program yang digunakan untuk membuat aplikasi berbasis web. *PHP* termasuk bahasa program yang bisa berjalan di sisi *server*, atau sering disebut *Side Server Language*. Jadi, program yang dibuat dengan kode *PHP* tidak bisa berjalan kecuali dia dijalankan pada *serverweb*, tanpa adanya *server web* yang terus berjalan tidak akan bisa dijalankan. *PHP* merupakan bahasa pemrograman yang ditunjuk untuk membuat aplikasi web. Ditinjau dari pemrosesannya, *PHP* tergolong berbasis *server side*. Artinya, pemrosesan dilakukan di *server*. Hal ini berkebalikan dengan bahasa seperti *JavaScript*, yang pemrosesannya dilakukan di sisi *client (client side)* (Anggraini et al., 2020).

PHP merupakan singkatan dari *PHP (Hypertext Preprocessor)* yang digunakan sebagai bahasa *script server-side* dalam pengembangan web yang disisipkan pada

dokumen *HTML*. PHP merupakan *software open source* yang disebar dan dilisensikan secara gratis serta dapat didownload secara bebas dari situs resminya <http://www.php.net> (Khoirula Umami & Mubarak, 2024). Kelebihan dari PHP, yaitu :

- a. Bahasa pemrograman *PHP* adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
- b. *Web server* yang mendukung PHP dapat ditemukan dimana-mana dari mulai apache, IIS, Lighttpd, nginx, hingga Xitami dengan konfigurasi lebih mudah.
- c. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan *developer* yang siap membantu pengembangan.
- d. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.

PHP adalah bahasa *open source* yang dapat digunakan di beberapa mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui *console* serta juga dapat menjalankan perintah-perintah sistem.

2.6.2 Cascading Style Sheets(CSS)

CSS (Cascading Style Sheet) adalah bahasa pemrograman untuk memberikan tampilan desain yang akan digunakan pada web seperti warna, *font*, *outline*, *background*, menyesuaikan tampilan website dengan ukuran layar, dsb. *CSS* digunakan pada pembuatan website ini adalah untuk berkolaborasi dengan *HTML* agar dapat menghasilkan tampilan *website* yang menarik (Sari et al., 2022).

CSS atau singkatan dari *Cascading Style Sheet* adalah suatu aturan untuk mengatur tampilan dari *website* sehingga tampilan dalam *web* lebih terstruktur. *CSS*

sendiri bukanlah bahasa pemrograman, CSS lebih seperti konfigurasi tampilan dari suatu tag pada *website*. CSS dapat merubah *text*, warna, *background* dan posisi dari suatu *tag* (Marlina et al., 2021).

CSS adalah singkatan dari *Cascading Style Sheet* yaitu dokumen *web* yang berfungsi mengatur elemen *HTML* dengan berbagai *property* yang tersedia sehingga dapat tampil dengan berbagai gaya yang diinginkan. Sebagian orang menganggap CSS bukan termasuk salah satu bahasa pemrograman karena memang strukturnya yang sederhana, hanya berupa kumpulan-kumpulan aturan yang mengatur *style* elemen *HTML* (Permatasari & Suhendi, 2020).

2.6.3 Java Script

Java Script merupakan salah satu bahasa pemrograman atau dalam hal ini bahasa *script* populer yang banyak digunakan untuk membuat suatu halaman *website* yang berinteraksi dengan pengguna dan juga dapat merespons *event* yang terjadi pada halaman. *Java Script* merupakan perekat antara halaman-halaman *website*. *Java Script* merupakan bahasa sisi klien yang didesain pada browser komputer, bukan pada server. *Java Script* awalnya dikenal dengan nama *LiveScript*. Sejarah *Java Script* bermula ketika Brendan Eich di Netscape pertama kali mengembangkannya pada tahun 1995 dengan terintegrasi pada Netscape Navigator 2.0 (Puwata, 2022).

2.7 Alat Bantu Perancangan Program

Perancangan suatu sistem membutuhkan beberapa alat bantu perancangan sistem agar analisis dan hasil dapat tercapai dengan maksimal.

2.7.1 Data

Secara etimologis data merupakan bentuk jamak dari kata “datum” yang berasal dari bahasa latin yang berarti “Sesuatu yang diberikan”. Data dapat berarti suatu fakta yang bisa digambarkan dengan kode, simbol, angka dan lain-lain (M.Makbul, 2021).

Menurut Kamus Besar Bahasa Indonesia pengertian data adalah keterangan yang benar dan nyata yang dapat dijadikan dasar kajian. Sedangkan Pribadi sendiri memiliki arti manusia sebagai perseorangan (diri manusia atau diri sendiri), sehingga dapat disimpulkan bahwa data pribadi merupakan keterangan yang benar dan nyata yang dimiliki oleh manusia sebagai perseorangan (Satria, 2022).

2.7.2 Basis Data (*Database*)

Basis data atau database, berasal dari kata basis dan data, adapun pengertian dari kedua pengertian tersebut adalah Basis: dapat diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul. Data: representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan peristiwa, konsep, keadaan, dan sebagainya yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya.(Cairina&Limega Candrasa, 2022).

Database adalah suatu kumpulan data yang berhub-ungan secara logika dan secara deskripsi dari data-data yang dirancang untuk memen-uhi kebutuhan informasi dalam suatu organisasi (Alfia, 2020)

2.7.3 Personal Home Page (PHP)

PHP adalah bahasa program yang digunakan untuk membuat aplikasi berbasis *web*. PHP termasuk bahasa program yang bisa berjalan di sisi *server*, atau sering

disebut *Side Server Language*. Jadi, program yang dibuat dengan kode PHP tidak bisa berjalan kecuali dijalankan pada *serverweb*, tanpa adanya *server web* yang terus berjalan tidak akan bisa dijalankan. PHP merupakan bahasa pemrograman yang ditunjuk untuk membuat aplikasi *web*. Ditinjau dari pemrosesannya, PHP tergolong berbasis *server side*. Artinya, pemrosesan dilakukan di *server*. Hal ini berkebalikan dengan bahasa seperti *Java Script*, yang pemrosesannya dilakukan di sisi klient (*client side*) (Anggraini et al., 2020).

PHP merupakan singkatan dari PHP (*Hypertext Preprocessor*) yang digunakan sebagai bahasa *script server-side* dalam pengembangan web yang disisipkan pada dokumen HTML. PHP merupakan *software open source* yang disebar dan dilisensikan secara gratis serta dapat didownload secara bebas dari situs resminya <http://www.php.net> (Sahi, 2020). Kelebihan dari PHP, yaitu :

- a. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
- b. *Web server* yang mendukung PHP dapat ditemukan dimana-mana dari mulai apache, IIS, Lighttpd, nginx, hingga Xitami dengan konfigurasi lebih mudah.
- c. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan *developer* yang siap membantu pengembangan.
- d. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.

PHP adalah bahasa *open source* yang dapat digunakan di beberapa mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah sistem.

2.7.4 MYSQL

MySQL merupakan suatu jenis *database server* yang sangat terkenal. MySQL termasuk jenis RDBMS (*Relational Database Management System*). MySQL mendukung bahasa pemrograman PH, bahasa permintaan yang terstruktur, karena pada penggunaannya SQL memiliki beberapa aturan yang telah distandarkan oleh asosiasi yang bernama ANSI. MySQL merupakan RDBMS (*Relational Database Management System*) *server*. RDBMS adalah program yang memungkinkan pengguna *database* untuk membuat, mengelola, dan menggunakan data pada suatu model *relational*. Dengan demikian, tabel-tabel yang ada pada *database* memiliki relasi antara satu tabel dengan tabel lainnya. Menurut (Hermiati et al., 2021) Ada beberapa keunggulan dari MySQL yaitu :

1. Cepat, handal dan mudah dalam penggunaannya. MySQL lebih cepat tiga sampai empat kali dari pada *database server* komersial yang beredar saat ini, mudah diatur dan tidak memerlukan seseorang yang ahli untuk mengatur administrasi pemasangan MySQL.
2. Didukung oleh berbagai bahasa *Database Server* MySQL dapat memberikan pesan *Error* dalam berbagai bahasa seperti Belanda, Portugis, Spanyol, Inggris, Perancis, Jerman, dan Italia.

3. Mampu membuat tabel berukuran sangat besar. Ukuran maksimal dari setiap tabel yang dapat dibuat dengan MySQL adalah 4 GB sampai dengan ukuran file yang dapat ditangani oleh sistem operasi yang dipakai.
4. Lebih murah MySQL bersifat *open source* dan didistribusikan dengan gratis tanpa biaya untuk UNIX platform, OS/2 dan *Windows Platform*. Melekatnya integrasi PHP dengan MySQL. Keterikatan antara PHP dengan MySQL yang sama-sama *Software Open-Source* sangat kuat, sehingga koneksi yang terjadi lebih cepat jika dibandingkan dengan menggunakan *database server* lainnya. Modul MySQL di PHP telah dibuat Built-in sehingga tidak memerlukan konfigurasi tambahan pada File konfigurasi PHP ini..

2.7.5 Hypertext Markup Language (HTML)

HTML (Hyper Text Markup Language) adalah bahasa standar yang digunakan untuk menampilkan halaman web. Yang bisa dilakukan dengan *HTML* yaitu mengatur tampilan dari halaman web dan isinya, mengatur tabel dan halaman web, mempublikasikan halaman web secara *online*, membuat *form* yang digunakan untuk menangani *registrasi* dan transaksi *via web*, menambahkan objek-objek seperti citra, audio, video, animasi, java applet dalam halaman web, serta menampilkan area gambar di browser.

2.7.6 XAMPP

XAMPP adalah perangkat lunak sumber terbuka yang digunakan untuk menciptakan lingkungan pengembangan web di dalam komputer lokal. XAMPP merupakan singkatan dari "X" (untuk berbagai sistem operasi), "Apache" (web server),

"MySQL" (sistem manajemen basis data), "PHP" (bahasa pemrograman), dan "Perl" (bahasa skrip). Perangkat ini memberikan platform komprehensif bagi pengembangan dan pengujian aplikasi web pada komputer pengguna sebelum diimplementasikan pada server online. XAMPP memungkinkan pengguna untuk dengan mudah mengaktifkan dan mengelola server web Apache, MySQL, PHP, dan Perl (Muntasir et al., 2023).

Mengenal bagian XAMPP yang biasa digunakan pada umumnya :

1. Htdocs adalah folder tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas PHP, HTML, dan script lain.
2. PHP MyAdmin merupakan bagian untuk mengelola bagian basis data MYSQL yang ada dikomputer. Untuk membukanya buka *browser* lalu ketik alamat <https://localhost/phpmyadmin> maka akan muncul sebuah halaman PHPMyAdmin.
3. Kontrol Panel yang berfungsi untuk mengelola layanan (*service*) XAMPP, seperti menghentikan (*stop*) layanan ataupun memulai (*start*).

2.8 UML (*Unified Modeling Language*)

UML merupakan sebuah bahasa spesifikasi standar yang memvisualisasi berdasarkan gambar untuk menspesifikasikan, membangun dan mendokumentasikan dari sebuah sistem pengembangan *software*. UML ada notasi grafik yang didukung oleh metode tunggal, yang membantu dalam pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek.

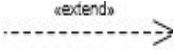
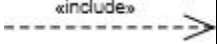
Bagian dan diagram adalah sebuah untuk penggambaran suatu permasalahan atau solusi dari sebuah model. Diantaranya :

2.8.1 Use Case Diagram

Usecase adalah teknik untuk memodelkan fungsional sistem, *usecase* mendeskripsikan interaksi tipikal antara para pengguna sistem dengan sistem itu sendiri. Didalam pembuatan *usecase*, secara garis besar terdapat dua macam pembacaan aliran informasi, yaitu aktor.

Tabel 2.1 Use Case Diagram

	<i>ACTOR</i> Orang proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari <i>actor</i> adalah gambar orang, biasanya dinyatakan menggunakan kata benda di awal <i>frase</i> nama <i>actor</i>.
	<i>USE CASE</i> Fungsionalitas yang disediakan sistem sebagai unitunit yang saling bertukar pesan antar unit atau actor biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama use case.
	<i>ASOSIASI/ASSOCIATION</i> Komunikasi antara <i>actor</i> dan <i>usecase</i> yang berpartisipasi pada <i>usecase</i> atau <i>usecase</i>.

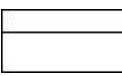
	EKSTENSI/EXTEND Relasi use case tambahan ke sebuah <i>usecase</i> dimana <i>usecase</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang di tambahkan.
	GENERALISASI/GENERALIZATION Hubungangeneralisasidan spesialisasi(umumkhusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
	MENGGUNAKAN/INCLUDE Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsional atau sebagai syarat dijalankan <i>use case</i> ini.

Sumber : Indah Nurlita & Reni Anggraini, 2023

2.8.2 Class Diagram

Class Diagram menunjukan *property* dan operasi sebuah *class* dan Batasan - Batasan yang terdapat dalam hubungan-hubungan objek.

Tabel 2.2 Class Diagram

GAMBAR	NAMA	KETERANGAN
	<i>Generalization</i>	Hubungan dimana objek semangka(<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 actor
	<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri

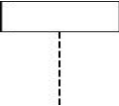
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objeklainnya
---	--------------------	--

Sumber : Indah Nurlita & Reni Anggraini, 2023

2.8.3 SequenceDiagram

Sequence diagram digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian untuk menghasilkan output tertentu.

Tabel 2.3 *Sequence Diagram*

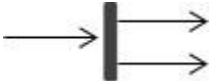
GAMBAR	NAMA	KETERANGAN
	<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

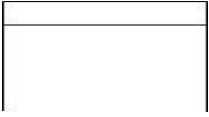
Sumber : Indah Nurlita & Reni Anggraini, 2023

2.8.4 Activity Diagram

Activity Diagram adalah sebuah teknik untuk menggambarkan logika prosedural, proses bisnis, dan jalur kerja. Dalam beberapa hal, diagram ini memakai peran mirip sebuah diagram alir.

Tabel 2.4 Activity Diagram

	<i>STATUS AWAL/INITIAL</i> Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
	<i>AKTIVITAS/ ACTIVITY</i> Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	<i>PERCABANGAN / DECISION</i> Asosiasi percabangan dimana lebih dari satu aktivitas digabungkan menjadi satu.
	<i>PENGGABUNGAN/ JOIN</i> Asosiasi penggabungan dimana lebih dari satu aktivitas lebih dari satu.
	<i>STATUS AKHIR/ FINAL</i> Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status satu.

	SWIMLINE Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
---	--

Sumber : Indah Nurlita & Reni Anggraini, 2023

2.9 Penelitian Terdahulu

Agar proses penelitian dan pemrograman berjalan sesuai dengan tujuan yang diinginkan, maka perlu beberapa perbandingan dengan penelitian sebelumnya.

Tabel 2.5 Penelitian Terdahulu

No	Penulis dan tahun	Judul	Metode	Hasil
1	Iskandar Dinata dan Sofa Sofiana (2022)	Implementasi Diagnosis Penyakit Pada Tanaman Anggur Menggunakan Metode CBR (<i>Case Based Reasoning</i>) Berbasis Website	<i>Case Based Reasoning</i>	Berdasarkan penelitian yang telah dilakukan dan pembahasan yang telah dijabarkan sebelumnya, maka didapatkan beberapa point kesimpulan sebagai berikut sistem pakar ini telah memberikan informasi terkait solusi dalam pendiagnosian penyakit tanaman anggur dikebun Azura Grape dengan berdasarkan gejala-gejala yang dialami sehingga membantu user/petani dalam mengidentifikasi jenis penyakit yang terjadi. Dengan

				penggunaan metode case based reasoning dan nearest neighbor similarity sistem pakar ini menghasilkan diagnosis lebih efektif karena berdasarkan data yang telah tersimpan dalam kasus lama yang dibandingkan tingkat kemiripannya dengan kasus/gejala-gejala baru yang dialami pada penyakit tanaman anggur dikebun Azura Grape. Berdasarkan hasil pengujian dari data gejala-gejala baru yang dialami dikebun anggur Azura Grape terdapat beberapa nilai dari masing-masing jenis penyakit anggur, yang pada akhir perhitungan terdapat 1 jenis penyakit dengan presentase paling besar yaitu penyakit peronospora dengan nilai 66,66%.
2	(Nasyuha et al., 2023)	Sistem Pakar Dalam Mendiagnosis Penyakit	<i>Case Based Reasoning</i>	Ada tiga jenis penyakit yang berbeda dengan total gabungan 15 gejala dalam sistem pakar untuk

		Leishmaniasis Menerapkan Metode Case-Based Reasoning (CBR)	mendiagnosis leishmaniasis. Tahap pertama dalam menangani kondisi tersebut sejak dini adalah memberikan solusi, yang bisa dilakukan di rumah. untuk membantu individu mengenali atau mendiagnosis leishmaniasis. Untuk memeriksa data gejala pasien saat menggunakan sistem pakar, metode Case-Based Reasoning telah berhasil melakukannya. Pasien teridentifikasi menderita cutaneous leishmaniasis berdasarkan temuan analisis yang menunjukkan nilai kemiripan maksimum sebesar 73% untuk suatu kondisi leishmaniasis. Oleh karena itu, diharapkan dengan hasil diagnosis ini, masyarakat dapat lebih mudah mengidentifikasi penyakit setelah penelitian sistem pakar untuk
--	--	--	---

				mendeteksi leishmaniasis, yang akan membantu pasien dan dokter dalam membuat penilaian terkait tindakan selanjutnya
3	(Dona et al., 2021)	Sistem pakar diagnosis penyakit jantung Dengan metode case based reasoning (cbr)	<i>Case Based Reasoning</i>	Dari hasil analisis terhadap percobaan implementasi sistem yang telah dilakukan dapat diambil beberapa kesimpulan mengenai Sistem Pakar Diagnosis Penyakit Jantung dengan Metode Case Based Reasoning (CBR) sebagai berikut: 1. Aplikasi sistem pakar diagnosis penyakit jantung dapat digunakan oleh semua orang yang ingin melakukan diagnosis penyakit jantung yang dialaminya. 2. Sistem ini dapat mengefisiensikan waktu dan mengatasi kurangnya pengetahuan pengguna terhadap gangguan penyakit jantung sehingga menghemat biaya konsultasi ke dokter spesialis jantung. 3. Dapat mengetahui penyakit jantung

				sejak dini sebelum dilakukan pengobatan lebih lanjut dengan dokter spesialis penyakit jantung. 4. Dapat diperoleh hasil yang sama antara perhitungan manual dengan perhitungan menggunakan sistem. 5. Hasil diagnosis yang dilakukan akan menunjukkan jenis penyakit jantung yang dialami dan berapa persen penyakit yang diderita serta solusi pengobatan sementara yang dapat dilakukan untuk penyakit yang diderita sebelum konsultasi lebih lanjut dengan dokter spesialis penyakit jantung.
4	(Nur et al., 2022)	Sistem Pakar Deteksi Penyakit Bawang Merah dengan Metode Case Based Reasoning	<i>Case Based Reasoning</i>	Dari tahapan penelitian serta berdasarkan proses analisis yang telah dilakukan, sehingga dapat diambil kesimpulan bahwa sistem pakar ini mampu menelusuri gejala yang telah dipilih oleh user berdasarkan kondisi bawang merah yang diamati.

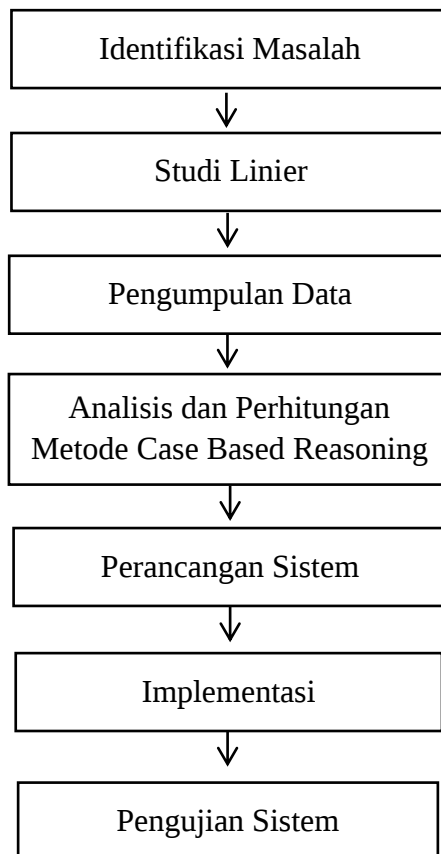
				Hasil identifikasi penyakit bawang dengan gejala: Umbi membusuk, Terdapat bercak melekok pada daun berwarna putih atau kelabu, Terdapat bercak merupai cincin berwarna ungu kemerahan, kemudian ujung daun mengering bahkan patah, Umbi mengering berwarna gelap. Didapatkan hasil identifikasi berupa Lalat Penggorok Daun dengan nilai 0%, Ulat Bawang 0%, Trips 0%, Ulat Tanah 0%, Layu Fusarium 33%, Bercak Ungu 100%, Antraknosa 0%, Virus Mozaik Bawang 0%, Bercak Daun 0%. Berdasarkan hasil perbandingan terhadap seluruh similarity penyakit pada bawang merah, maka dapat disimpulkan bahwa jenis penyakit yang dialami adalah penyakit Bercak Ungu dengan nilai similarity 100%
5	(Nasyuha et	Sistem Pakar	Case	Berdasarkan hasil perancangan dan pembahasan

	al., 2023)	Untuk Mendeteksi Penyakit Mata Menerapkan Metode Case Based Reasoning	<i>Based Reasoning</i>	pada perancangan sistem pakar diagnosis penyakit Mata maka dapat membuat sebuah sistem pakar dengan menggunakan metode Case Based Reasoning sehingga dapat mendeteksi penyakit Mata yang diderita Pasien; Memberikan informasi berupa gejala-gejala dan jenis penyakit yang terdapat pada pasien serta penanganannya berdasarkan penalaran para pakar dengan menggunakan aplikasi berbasis web. Serta melakukan diagnosis awal untuk mengetahui gejala awal penyakit untuk memastikan apakah pasien menderita penyakit Mata tanpa memerlukan konsultasi terhadap dokter. Dari perhitungan berdasarkan analisis data dengan membuat kemiripan gejala awal dengan gejala kasus maka dapat dihasilkan penyakit ablasio retina sebesar 33,97 %.
--	------------	---	------------------------	---

BAB 3

METODOLOGI PENELITIAN

Metodologi penelitian adalah cara yang digunakan dalam memperoleh berbagai data untuk diproses menjadi informasi yang lebih akurat sesuai permasalahan yang akan diteliti. Metodologi yang digunakan dalam penelitian tugas akhir yang berjudul "Perancangan Sistem Pakar Diagnosis Penyakit Tanaman Semangka Dengan Menggunakan Metode *Case Based Reasoning* Berbasis Web" adalah:



Gambar 3.1 Kerangka Penelitian

Dalam metodologi penelitian dijabarkan tahapan-tahapan yang dilakukan dalam penelitian. Metodologi penelitian terdiri dari beberapa tahapan yang terkait secara sistematis. Tahapan ini diperlukan untuk memudahkan dalam melakukan penelitian.

Tahapan yang dilakukan dalam penelitian adalah sebagai berikut :

3.1 Identifikasi Masalah

Langkah pertama dalam mengembangkan sistem pakar ini adalah dengan mengidentifikasikan permasalahan yang akan diteliti, tahap ini dilakukan dengan menemukan permasalahan yang akan diteliti sehingga akan mempermudah data ditahap berikutnya.

3.2 Studi Linier

Studi linier dilakukan untuk mendapatkan teori serta konsep yang mendukung dalam penelitian dan berkaitan dengan masalah yang diangkat dalam penelitian. Hal dipelajari dalam studi linier antara lain definisi sistem pakar, penggunaan *Case Based Reasoning*, dan metode yang digunakan untuk kasus mendiagnosis penyakit menular pada anak dengan membaca buku-buku, jurnal-jurnal, artikel-artikel dan referensi yang terkait sehingga memudahkan dalam menyelesaikan permasalahan yang ada.

3.3 Pengumpulan Data

Pada tahap ini dilakukan pengumpulan data tentang aplikasi sistem pakar penyakit tanaman semangka, Semua tahap pada proses pengumpulan data-data tersebut diperoleh dari wawancara dan studi pustaka.

3.3.1 Wawancara (*Interview*)

Proses wawancara dilakukan langsung kepada Balai Benih Umum tentang penyakit tanaman semangka untuk mendapatkan jenis penyakit, gejala, dan nilai kepastian pada setiap gejala untuk menentukan jenis penyakit.

3.3.2 Studi Pustaka (*Library Research*)

Studi pustaka dilakukan untuk mendapatkan teori serta konsep yang mendukung dalam penelitian dan berkaitan dengan masalah yang diangkat dalam penelitian. Hal dipelajari dalam studi pustaka antara lain definisi sistem pakar, penggunaan *Case Based Reasoning*, dan metode yang digunakan untuk kasus mendiagnosis penyakit menular pada anak dengan membaca buku-buku, jurnal-jurnal, artikel-artikel dan referensi yang terkait sehingga memudahkan dalam menyelesaikan permasalahan yang ada.

3.4 Analisis dan Perhitungan Metode *Case Based Reasoning*

Pada bagian ini analisis dilakukan terhadap data dan permasalahan yang telah dirumuskan. kemudian merancang sebuah sistem yang dapat menjawab permasalahan dan kendala yang ada. Adapun analisis yang dilakukan adalah:

Similarity adalah pendekatan untuk mencari kasus dengan menghitung kedekatan antara kasus baru dengan kasus lama, yaitu berdasarkan pada pencocokan nilai dari sejumlah fitur yang ada. Adapun rumus untuk melakukan perhitungan kedekatan antara dua kasus adalah sebagai berikut. $\text{Similarity (problem, case) =}$

Rumus dasar *Case Based Reasoning*:

$$\frac{S1*W1+S2*W2+.....+Sn*Wn}{W1 + W2 ++ Wn} \dots\dots\dots(1)$$

S = similarity (nilai kemiripan) yaitu 1 (sama) dan 0 (beda)

W = weight (bobot yang diberikan)

3.5 Perancangan Sistem

Setelah tahapan analisis selesai dilakukan, maka tahapan selanjutnya adalah perancangan sistem. Tahapan perancangan sistem terdiri dari:

1. Perancangan struktur menu yang akan digunakan pada sistem yang dibangun.
2. Tahapan rancangan database beserta atribut yang dibutuhkan.
3. Tahapan rancangan *user interface* atau antar muka pengguna pada sistem yang dibangun.

3.6 Implementasi

Setelah melakukan tahap analisis sistem, maka pada tahap ini diimplementasikan dalam bahasa pemrograman komputer (*coding*). Untuk mengimplementasikan aplikasi ini dibutuhkan perangkat pendukung, perangkat tersebut berupa perangkat lunak dan perangkat keras dengan spesifikasi sebagai berikut:

a. Perangkat lunak

Dalam pembuatan dan penerapan aplikasi Berbasis Web ini dibutuhkan perangkat lunak yang menunjang pembuatannya adalah sebagai berikut:

1. Bahasa pemrograman : PHP, HTML, CSS

2. Editor : Visual Studio Code
 3. Web server : Xampp
 4. Database Management System (DBMS): MySQL.
 5. Browser : Google Chrome
 6. Sistem Operasi : Windows 10
- b. Perangkat keras

Perangkat keras yang akan digunakan dalam pembuatan sistem adalah:

1. Processor Intel Core i5 Processor (2.50GHz)
2. Memory 4 GB
3. Harddisk berkapasitas 118 GB
5. Terkoneksi dengan Internet

3.7 Pengujian Sistem

Pengujian dilakukan pada saat aplikasi akan dijalankan. Tahap pengujian dilakukan untuk dijadikan ukuran bahwa sistem berjalan sesuai dengan tujuan. Tujuan dari pengujian adalah mencari kesalahan atau error sesuai dengan kriteria yang ditetapkan, manfaat dari pengujian ini adalah agar jika aplikasi telah dijalankan dan digunakan tidak terjadi sebuah kesalahan atau tidak bermasalah. tahapan yang dilakukan dalam pengujian adalah :

1. Pengujian *Black Box*

Pengujian *black box* berfokus pada persyaratan atau kebutuhan fungsional perangkat lunak yang dibuat. Pengujian black-box berusaha menemukan kesalahan dalam kategori sebagai berikut:

- a. Fungsi-fungsi yang tidak benar atau hilang
- b. Kesalahan interface
- c. Kesalahan dalam struktur data atau akses database eksternal
- d. Kesalahan kinerja
- e. Inisialisasi dan kesalahan terminasi

.