

**SKRIPSI**

**SISTEM PAKAR DIAGNOSIS PENYAKIT PADA TANAMAN CABAI ,  
MENGUNAKAN METODE *FORWARD CHAINING* DAN *CERTAINTY*  
*FACTOR* BERBASIS WEB**



**HAMRIANI**

**16121241**

**PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS SEMBILANBELAS NOVEMBER KOLAKA  
2020**

HALAMAN PERSETUJUAN  
SKRIPSI

SISTEM PAKAR DIAGNOSIS PENYAKIT PADA TANAMAN CABAI  
MENGUNAKAN METODE *FORWARD CHAINING* DAN  
*CERTAINTY FACTOR* BERBASIS WEB

(Studi Kasus : Desa Nelumbu Kecamatan Mowewe)



Muh. Nurtanzis Sutoyo, S.Kom., M.Cs  
NIDN. 0921068401

## KATA PENGANTAR

Puji syukur kehadiran ALLAH SWT karena atas berkat, rahmat dan hidayahnya akhirnya penulis dapat menyelesaikan skripsi ini yang berjudul **“Sistem Pakar Diagnosis penyakit Pada Tanaman Cabai Dengan Menggunakan Metode *Forward Chaining* Dan *Certainty Factor* Berbasis WEB”** sebagai salah satu syarat untuk mendapatkan gelar Strata Satu (S1) pada Universitas SembilanBelas November Kolaka.

Penulis menyadari, berhasilnya studi dan penyusunan skripsi ini tidak terlepas dari bantuan berbagai pihak yang telah memberikan semangat dan do'a kepada penulis dalam menghadapi setiap tantangan, olehnya itu penulis menghaturkan Terimakasih sebesar-besarnya kepada pihak-pihak yang telah membantu penulis hingga dapat menyelesaikan skripsi ini.

Dalam penyajian skripsi penulis menyadari bahwa terdapat banyak kekurangan dalam isi proposal ini. Hal ini tidak terlepas dari kata-kata yang mungkin tidak berkenan dan perlu untuk diperhatikan kembali. Oleh karena itu penulis sangat mengharapkan kepada bapak/ibu untuk mengoreksi demi kesempurnaan penulisan.

Kolaka, Maret 2021

Hamriani

## DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PERSETUJUAN.....	ii
KATA PENGANTAR .....	iii
DAFTAR ISI.....	v
DAFTAR GAMBAR.....	vii
DAFTAR TABEL.....	vii
BAB. I    PENDAHULUAN	
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah .....	3
1.3. Tujuan Penelitian.....	3
1.4. Batasan Masalah .....	3
1.5. Manfaat Penelitian.....	3
BAB. II    TINJAUAN PUSTAKA	
2.1. Kajian Pustaka .....	4
2.2. Landasan Teori .....	6
2.2.1 Sistem .....	6
2.2.2. Sistem Pakar .....	9
2.2.3. Struktur Sistem Pakar .....	13
2.2.4. Diagnosis Penyakit .....	18
2.2.5. Metode <i>Certainty Factor</i> .....	19
2.2.6. <i>Forward Chaining</i> .....	20
2.2.7. Perancangan Sistem .....	21
2.2.8. <i>Data Flow Diagram (DFD)</i> .....	25
2.2.9. <i>Flowchart</i> .....	26
2.2.10. <i>Flowmap</i> .....	27
2.2.11. Pengertian <i>Web</i> , MySQL, PHP.....	28
2.2.12. Metode Pengembangan Sistem.....	30
2.2.13. Teknik pengujian sistem.....	32
BAB. III    METODE PENELITIAN	
3.1. Jadwal Penelitian .....	34
3.2. Teknik Pengumpulan Data .....	34
3.3. Metode Pengembangan Sistem.....	35
3.4. Analisis Kebutuhan .....	36
BAB. IV    HASIL DAN PEMBAHASAN	
4.1. Analisis Sistem .....	37
4.2. Metode <i>Forward Chaining</i> .....	37
4.3. Metode <i>Certainty Factor</i> .....	39
4.4. Perancangan Sistem.....	43
4.5. Perancangan Basis Data.....	45

4.5.1	Struktur Tabel .....	45
4.5.2	Relasi Antar Tabel .....	48
4.6.	<i>Flowchart</i> .....	48
4.3.1.	<i>Flowchart Login</i> .....	48
4.3.2.	<i>Flowchart Menu Utama</i> .....	49
4.3.3.	<i>Flowchart Form Input Penyakit</i> .....	50
4.3.3.	<i>Flowchart Form Input Gejala</i> .....	50
4.3.3.	<i>Flowchart Input Rule</i> .....	51
4.3.3.	<i>Flowchart Diagnosis</i> .....	52
4.3.3.	<i>Flowchart User</i> .....	52
4.7.	Implementasi Sistem .....	53
4.8.	Pengujian Sistem .....	50
4.8.1.	Pengujian Sistem Pada Halaman Login Admin .....	60
4.8.2.	Pengujian Sistem Pada Halaman Utama Admin .....	61
4.8.3.	Pengujian Sistem Pada Halaman Data Penyakit .....	62
4.8.4.	Pengujian Sistem Pada Halaman Data Gejala .....	62
4.8.5.	Pengujian Sistem Pada Halaman Data Rule .....	63
4.6.	Pengujian Akurasi Sistem .....	63
BAB. V PENUTUP		
5.1.	Kesimpulan .....	65
5.2.	Saran .....	65

## DAFTAR PUSTAKA

## DAFTAR GAMBAR

Gambar 2.1	Arsitektur Sistem Pakar .....	13
Gambar 2.2	<i>Algoritma Forward Chaning</i> .....	20
Gambar 2.3	Tahapan Model <i>Waterfall</i> .....	31
Gambar 4.1	<i>Flowchart</i> Sistem .....	40
Gambar 4.2	Diagram Konteks .....	43
Gambar 4.3	Diagram Level 0 .....	44
Gambar 4.4	Diagram Level 1 Proses 1 .....	44
Gambar 4.5	Diagram Level 1 Proses 2 .....	45
Gambar 4.6	Diagram Level 1 Proses 3 .....	45
Gambar 4.7	Relasi Antar Tabel .....	48
Gambar 4.8	<i>Flowchart</i> Login Admin .....	48
Gambar 4.9	<i>Flowchart</i> Menu Utama .....	49
Gambar 4.10	<i>Flowchart Form Input</i> Penyakit .....	50
Gambar 4.11	<i>Flowchart Form Input</i> Gejala .....	51
Gambar 4.12	<i>Flowchart Form Input</i> Rule .....	51
Gambar 4.13	<i>Flowchart</i> Diagnosis .....	52
Gambar 4.14	<i>Flowchart Form User</i> .....	52
Gambar 4.15	Halaman Login .....	53
Gambar 4.16	Halaman Menu Utama Admin .....	54
Gambar 4.17	Halaman <i>Input</i> Penyakit .....	55
Gambar 4.18	Halaman <i>Input</i> Gejala .....	56
Gambar 4.19	Halaman <i>Input</i> Rule .....	57
Gambar 4.20	Halaman Diagnosis .....	58
Gambar 4.21	Halaman <i>User</i> .....	59
Gambar 4.22	Laporan Hasil Diagnosis .....	60

## DAFTAR TABEL

Tabel 2.1	Penelitian Terdahulu .....	5
Tabel 2.2	Kombinasi <i>evidence anteseden</i> .....	20
Tabel 2.3	Contoh <i>ERD</i> .....	24
Tabel 2.4	Simbol-Simbol DFD .....	25
Tabel 2.5	Simbol-Simbol dalam <i>Flowchart</i> .....	26
Tabel 2.6	Simbol-Simbol <i>Flowmap</i> .....	27
Tabel 3.1	Jadwal Penelitian.....	34
Tabel 4.1	Tabel Penyakit.....	38
Tabel 4.2	Tabel Gejala .....	38
Tabel 4.3	Basis Pengetahuan.....	39
Tabel 4.4	Tabel Penyimpanan Data Penyakit .....	46
Tabel 4.5	Tabel Penyimpanan Data Gejala .....	46
Tabel 4.6	Tabel Penyimpanan Data Aturan .....	47
Tabel 4.7	Tabel Penyimpanan Data Diagnosis .....	47
Tabel 4.8	<i>Tabel Penyimpanan Data User</i> .....	48
Tabel 4.9	Pengujian Sistem Pada Halaman Login .....	60
Tabel 4.10	Pengujian Sistem Pada Halaman Utama Admin .....	61
Tabel 4.11	Pengujian Sistem Pada Halaman Data Penyakit .....	62
Tabel 4.12	Pengujian Sistem Pada Halaman Data Gejala.....	62
Tabel 4.13	Pengujian Sistem Pada Halaman Data <i>Rule</i> .....	63
Tabel 4.14	Tabel Data Pengujian Sistem Yang Sesuai dan Tidak Sesuai.....	64

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Di Kolaka Timur mulai banyak petani yang membudidayakan cabai merah atau cabai salah satunya banyak petani Mowewe Kolaka Timur yang membudidayakan tanaman cabai untuk rotasi dalam bercocok tanam atau menjadi mata pencaharian dari menanam cabai. Sebagian besar atau mayoritas penduduknya bermata pencaharian dengan cara bertani atau bercocok tanaman, banyak tanaman yang dapat hidup di Mowewe salah satunya adalah tanaman cabai, akan tetapi kendala di jumpai dalam proses penanaman cabai antara lain hama dan penyakit yang sering mengakibatkan gagal panen. Pada saat musim tertentu (musim hujan), produksi cabai berkurang. Salah satu penyebab produksi menurun adalah penyakit tanaman cabai yang diserang penyakit dapat menimbulkan gagal panen. Untuk itu diperlukan upaya pengendalian yang tepat agar tidak menggagalkan panen. Pengendalian tidak hanya saat serangan sudah ada, tetapi yang paling penting adalah tindakan mencegah agar penyakit tidak datang menyerang.

Masih banyak dari petani yang belum dapat menanam tanaman cabai sesuai dengan standar yang ada, yang bisa mengakibatkan terserangnya tanaman oleh penyakit, kebanyakan tidak mengetahui dengan pasti jenis penyakit yang sedang mengenai cabai. Kebanyakan petani akan menduga duga penyakit apa yang sedang menyerang tanaman cabai yang mengakibatkan para petani merugi, karena tanamannya mati, harga jual yang rendah dan hasil panen yang sedikit. Karena kurangnya pengetahuan petani dalam mengetahui jenis penyakit yang menyerang tanaman cabai dan bagaimana cara mengatasinya sesuai dengan ciri-ciri penyakit yang terdapat pada tanaman tersebut membuat terlambatnya proses diagnosis.

Cara perawatan yang salah atau tidak sesuai dengan standart yang ada, dapat mengakibatkan tanaman cabai rentan terkena hama penyakit dan dapat mengakibatkan hasil panen tidak bisa maksimal, hasil penjualan yang rendah dan



bahkan bisa mengakibatkan gagal panen. Hama utama pada tanaman cabai antara lain ulat hama, ulat grayak, kutu daun, thrips, tungau, lalat buah hama tersebut yang bisa mengakibatkan tanaman cabai terkena penyakit. Penyakit utama pada tanaman cabai antara lain adalah busuk buah, bercak daun, layu fusarium, penyakit virus.

Di Mowewe, tenaga tani yang ahli pada bidang penyakit tanaman cabai masih terbatas, baik dari segi jumlah dan waktu kerja. Dalam menyelesaikan serangan hama dan penyakit yang menyerang tidak sedikit dari petani melakukan kesalahan dalam mengatasi permasalahan yang dihadapi. Sehingga dengan pembuatan sistem pakar untuk mendiagnosis penyakit tanaman Cabai ini diharapkan bisa membantu petani untuk mengatasi permasalahan dengan memberikan solusi yang baik. Proses pembuatan sistem pakar ini adalah menggunakan metode *forward chaining* yang digunakan untuk melakukan peruntutan untuk mencocokkan fakta yang tersimpan dalam basis pengetahuan menggunakan beberapa aturan dan metode *certainty factor* yang digunakan untuk menyelesaikan masalah ketidakpastian data. Dalam hal ini sistem pakar dihadirkan sebagai alternatif pengganti seorang pakar, petani membutuhkan sebuah sistem pakar untuk meningkatkan kualitas dalam menanam dan juga bisa meningkatkan harga jual tanaman cabai dan WEB digunakan karena memiliki keunggulan seperti mudah dikembangkan, mudah untuk diakses, fleksibel dan informasi lebih mudah didistribusikan.

Penelitian Sumpala dan Sutoyo (2018) dengan Sistem Pakar Untuk Mendiagnosis Hama dan Penyakit Tanaman Kakao Menggunakan Metode *Forward Chaining* dan *Certainty Factor* menghasilkan sebuah sistem pakar yang dapat menganalisis jenis penyakit tanaman kakao dengan hasil pengujian tingkat keakuratan sebesar 99%, berdasarkan penjelasan diatas maka dibutuhkan sistem pakar untuk mendiagnosis penyakit tanaman cabai dengan harapan dapat memodelkan sistem pakar diagnosis penyakit tanaman cabai yang dapat memberikan informasi mengenai gejala, penyakit, dan cara pengendaliannya. penulis tertarik membuat sistem pakar dengan harapan pembuatan sistem pakar ini bisa berjalan dengan lancar serta bermanfaat bagi masyarakat petani tanaman

Cabai dalam melestarikan Cabai serta meningkatkan hasil panen masyarakat. Penelitian tersebut dilakukan penelitian mengenai ”Sistem Pakar Diagnosis Penyakit Tanaman Cabai Berbasis *Web* Menggunakan Metode *Forward chaining* dan *Certainty Factor*” dengan harapan dapat memodelkan sistem pakar diagnosis penyakit tanaman cabai yang dapat memberikan solusi secara tepat.

### 1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, maka dapat dirumuskan di dalam penelitian ini adalah apakah metode *Forward Chaing* dan *Certainty Factor* dapat digunakan untuk mendiagnosis penyakit pada tanaman cabai ?

### 1.3 Batasan Masalah

Agar tidak terlepas dari maksud dan tujuan dalam penyusunan laporan maka peneliti membatasi pokok permasalahan :

1. Sistem pakar ini hanya menyelesaikan masalah tentang penyakit pada tanaman cabai dan data akan di ambil dari Balai Desa Nelombu Kecamatan Mowewe Kabupaten Kolaka Timur.
2. Penyakit tanaman cambia yang didiagnosis ialah Antraknosa, Bercak Daun Serkospora dan Busuk daun Fitofthora

### 1.4 Tujuan Penelitian

Tujuan dari penelitian ini yaitu adalah untuk membangun aplikasi sistem pakar yang dapat mendiagnosis penyakit pada tanaman Cabai menggunakan metode *Forward Chaing* dan *Certainty Factor* berbasis web.

### 1.5 Manfaat Penelitian

Manfaat dari penelitian ini antara lain:

1. Membantu masyarakat, petani atau penyuluh dalam mendapatkan informasi penyakit tanaman Cabai yang ada di Mowewe.
2. Membantu masyarakat, petani atau penyuluh tanaman cabai dalam mendapatkan informasi penanganannya berdasarkan gejala yang terlihat.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Kajian Pustaka**

Susilo (2018) dengan judul Sistem Pakar Metode *Forward Chaining* Dan *Certainty Factor* Untuk Mengidentifikasi Penyakit Pertusis Pada Anak dan hasil yang didapat Dengan adanya sistem pakar untuk mengidentifikasi penyakit pertusis pada anak berbasis web dalam mendiagnosis dengan menggunakan metode *Forward Chaining* dan *Certainty Factor* dengan tingkat keyakinan pertusis pada anak sebesar 97% .

Yuwono, dkk (2017) dengan judul Penerapan Metode Forward Chaining Dan Certainty Factor Pada Sistem Pakar Diagnosis Hama Anggrek Coelogyne Pandurata dan hasil yang di dapat Dengan adanya sistem Pakar dengan menggunakan *Forward Chaining* dan *Certainty Factor* dapat membantu para pembudidaya dan juga penghobi dalam mendapatkan informasi tentang hama yang menyerang tanaman anggrek Coelogyne Pandurata, dan tidak diharuskan untuk berkonsultasi langsung dengan para pakar anggrek dengan presentase 93,0736 %.

Sumpala dan Sutoyo (2018) dengan judul Sistem Pakar Untuk Mendiagnosis Hama dan Penyakit Tanaman Kakao Menggunakan Metode Forward Chaining dan Certainty Factor dan hasil yang di dapat sebuah sistem pakar yang dapat menganalisis jenis penyakit tanaman kakao dengan hasil pengujian tingkat keakuratan sebesar 99%.

Tabel 2.1 Penelitian Terdahulu

No	Peneliti	Judul	Metode	Hasil
1	Susilo (2018)	Sistem Pakar Metode <i>Forward Chaining</i> Dan <i>Certainty Factor</i> Untuk Mengidentifikasi Penyakit Pertusis Pada Anak	<i>Forward Chaining, Ceratinty Factor</i>	Dengan adanya sistem pakar untuk mengidentifikasi penyakit pertusis pada anak berbasis web dalam mendiagnosis dengan menggunakan metode <i>Forward Chaining</i> dan <i>Certainty Factor</i> dengan tingkat keyakinan pertusis pada anak sebesar 97%
2	Yuwono, dkk (2017)	Penerapan Metode <i>Forward Chaining</i> Dan <i>Certainty Factor</i> Pada Sistem Pakar Diagnosis Hama Anggrek Coelogyne Pandurata	<i>Forward Chaining Dan Certainty Factor</i>	Dengan adanya sistem Pakar dengan menggunakan <i>Forward Chaining</i> dan <i>Certainty Factor</i> dapat membantu para pembudidaya dan juga penghobi dalam mendapatkan informasi tentang hama yang menyerang tanaman anggrek Coelogyne Pandurata, dan tidak diharuskan untuk berkonsultasi langsung dengan para pakar anggrek dengan presentase 93,0736 %.
3	Sumpala dan Sutoyo (2018)	Sistem Pakar Untuk Mendiagnosis Hama dan Penyakit Tanaman Kakao Menggunakan Metode <i>Forward Chaining</i> dan <i>Certainty Factor</i>	<i>Forward Chaining Dan Certainty Factor</i>	Menghasilkan sebuah sistem pakar yang dapat menganalisis jenis penyakit tanaman kakao dengan hasil pengujian tingkat keakuratan sebesar 99%

Pada kasus yang sekarang ini yang akan dilakukan penelitian terbaru yaitu **Sistem Pakar Diagnosis Penyakit Pada Tanaman Cabai Menggunakan Metode *Forward Chaining* dan *Certainty factor* berbasis web**”. Penelitian ini membahas pembuatan program aplikasi untuk mengidentifikasi penyakit tanaman

Cabai menggunakan metode *forward chaining* dan *certainty factor*, dan menggunakan bahasa pemrograman PHP dan MySQL digunakan sebagai dasar pembentukan basis pengetahuan program aplikasi ini Hasil atau luaran dari aplikasi ini memberikan nilai probabilitas kepastian hama dan penyakit pada tanaman cabai, dan kemudian hipotesanya dipilih dengan nilai terbesar.

## 2.2 Landasan Teori

### 2.2.1 Sistem

Bagi suatu lembaga yang berkembang perancangan sistem memiliki peranan yang sangat penting karena dengan perancangan sistem yang tepat maka tujuan dari lembaga tersebut akan mudah dicapai. Untuk mengetahui lebih dalam tentang sistem berikut ini ada beberapa pengertian tentang sistem dari para ahli yaitu sebagai berikut:

Menurut Abdul Kadir (2003) menyatakan bahwa sistem merupakan sekumpulan elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan. Sedangkan menurut McLeod (Yakub 2012) Sistem adalah sekelompok elemen-elemen yang terintegrasi dengan tujuan yang sama untuk mencapai tujuan.

Menurut Tata Sutabri (2012) mendefenisikan bahwa sistem merupakan suatu kesatuan yang terdiri dari dua atau lebih komponen atau subsistem yang berinteraksi untuk mencapai tujuan. Sedangkan menurut Fredick H. Wu, sistem adalah “beroperasi dan berinteraksi dengan lingkungannya untuk mencapai sasaran tertentu”.

Berdasarkan pendapat yang di kemukakan di atas ditarik kesimpulan bahwa pengertian sistem adalah suatu kelompok yang dapat beroprasi dan berinteraksi baik fisik maupun non fisik dan menyelesaikan masalah dan mencapai suatu tujuan tertentu.

Menurut Jogyanto HM (2005).Menjelaskan bahwa karakteristik sistem yang dimiliki sebuah sistem adalah sebagai berikut :

### 1. Komponen Sistem

Suatu sistem terdiri dari sejumlah komponen-komponen yang saling berinteraksi, bekerjasama membentuk satu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem setiap sistem tidak peduli betapapun kecilnya, selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses system secara keseluruhan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem setiap sistem tidak peduli betapapun kecilnya, selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses system secara keseluruhan.

### 2. Batas Sistem

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan atau dengan lingkungan luarnya.

### 3. Lingkungan Luar Sistem

Segala sesuatu diluar dari batas sistem yang mempengaruhi proses dari suatu sistem. Lingkungan luar sistem ini dapat bersifat menguntungkan atau merugikan. Lingkungan luar yang menguntungkan harus di pelihara dan dijaga agar tidak hilang pengaruhnya, sedangkan lingkungan luar yang merugikan harus di musnahkan dikendalikan agar tidak mengganggu operasi sistem.

#### 4. Penghubung Sistem

Merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Untuk membentuk satu kesatuan, sehingga memungkinkan sumber-sumber daya mengalir dari subsistem yang satu ke subsistem yang lainnya. Dengan kata lain output dari suatu subsistem akan menjadi input dari subsistem yang lainnya.

#### 5. Masukan Sistem

Merupakan energi yang dimasukkan kedalam sistem. Masukan dapat berupa masukan Perawatan (*Maintenance Input*) adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Masukan sinyal (*Signal Input*) adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh sistem komputer, program adalah maintenance input yang digunakan untuk mengoperasikan komputernya dan data adalah sinyal input untuk diolah menjadi informasi.

#### 6. Keluaran Sistem

Merupakan hasil dari energi yang diolah oleh sistem. Misalnya untuk sistem komputer, keluaran yang berguna adalah informasi yang dikeluarkan oleh komputer, dan keluaran yang tidak berguna yang dikenal sebagai sisa pembuangan adalah panas yang dikeluarkan oleh komputer.

#### 7. Pengolah Sistem

Merupakan bagian yang memproses masukan untuk menjadi keluaran yang diinginkan. Misalnya bagian produksi yang mengubah bahan baku menjadi barang jadi, bagian akuntansi yang mengolah data transaksi menjadi laporan keuangan.

#### 8. Tujuan Sistem

Setiap sistem mempunyai tujuan ataupun sasaran dan mempengaruhi input yang dibutuhkan dan output yang dihasilkan. Dengan kata lain suatu sistem akan dikatakan berhasil kalau pengoperasian sistem itu mengenai sasaran atau tujuan. Sistem yang tidak mempunyai sasaran, maka operasi sistem tidak ada gunanya.

### 2.2.2 Sistem Pakar

Sistem Pakar adalah sistem computer yang mengemulasi kemampuan kepakaran manusia. Kata mengemulasi diartikan lebih kuat dari simulasi yang berarti bahwa sistem pakar diharapkan mampu bertindak sebagaimana yang dilakukan pakar manusia dalam melakukan penalaran untuk memberikan suatu justifikasi/kesimpulan. Dalam melakukan penalaran, sistem pakar banyak memanfaatkan pengetahuan yang disimpan oleh sistem sebagai basis pengetahuan untuk menyelesaikan masalah pada tingkatan sebanding dengan pakar manusia (sri hartati dan sri iswanti 2003 )

Sistem pakar adalah aplikasi berbasis komputer untuk menyelesaikan suatu masalah sehari-hari sebagaimana halnya seperti yang dilakukan oleh seorang pakar (Kusrini, 2008). Pengetahuan dari seorang pakar tersebut kemudian diterapkan/ditirukan kedalam aplikasi komputer. Kemudian, aplikasi komputer tersebut dapat digunakan oleh pengguna untuk berkonsultasi. Sistem pakar dibentuk oleh 2 komponen utama yaitu basis pengetahuan (*knowledge based and inference engine*). Basis pengetahuan adalah proses penyimpanan pengetahuan seorang pakar ke dalam memori komputer. Sedangkan mesin inferensi itu sendiri merupakan otak dari aplikasi sistem pakar. Bagian ini menuntun pengguna untuk memasukkan data-data untuk kemudian diproses sehingga diperoleh suatu kesimpulan.

Sistem pakar adalah sistem cerdas berbasis komputer digunakan dalam penyelesaian masalah yang hanya bisa dilakukan oleh ahli/pakar pada suatu bidang. Dengan sistem ini masyarakat umum dapat melakukan perhitungan layaknya seorang pakar (Kusumadewi, 2003). Terdapat dua bagian penting dari sistem pakar meliputi lingkungan konsultasi dan lingkungan pengembang. Lingkungan pengembang digunakan oleh pengembang sistem untuk membangun komponen dan memperkenalkan pengetahuan ke dalam basis pengetahuan. Untuk lingkungan konsultasi berguna untuk melakukan konsultasi sehingga memperoleh pengetahuan dari sistem pakar layaknya dari seorang pakar (Sutojo, 2011).



Sistem pakar merupakan sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli. Sistem pakar biasanya digunakan untuk melakukan interpretasi dan analisa, diagnosis, dan membantu pengambilan keputusan (Oktaviani, 2012).

Sistem pakar sebagai sistem berbasis komputer yang menggunakan pengetahuan, fakta, dan teknik penalaran dalam memecahkan masalah yang biasanya hanya dapat dipecahkan oleh seorang pakar dalam bidang tersebut (Hartati dan Iswanti, 2008).

Sistem Pakar (*expert system*) adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli. Sistem pakar yang baik dirancang agar dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja dari para ahli. Dengan sistem pakar ini, orang awam pun dapat menyelesaikan masalah yang cukup rumit yang sebenarnya hanya dapat diselesaikan dengan bantuan para ahli. Bagi para ahli, sistem pakar ini juga akan membantu aktivitasnya sebagai asisten yang sangat berpengalaman (Kusumadewi, 2003).

Menurut Kusumadewi (2003) konsep dasar sistem pakar mengandung keahlian, ahli, pengalihan keahlian, inferensi, aturan, dan kemampuan menjelaskan. Keahlian adalah suatu kelebihan penguasaan pengetahuan dibidang tertentu yang diperoleh dari pelatihan, membaca atau pengalaman.

Contoh bentuk pengetahuan yang termasuk keahlian adalah:

1. Fakta-fakta pada lingkup permasalahan tertentu.
2. Teori-teori pada lingkup permasalahan tertentu.
3. Prosedur-prosedur dan aturan-aturan berkenaan dengan lingkup permasalahan tertentu.
4. Strategi-strategi global untuk menyelesaikan masalah.
5. *Meta-knowledge* (pengetahuan tentang pengetahuan).

Ciri-ciri sistem pakar menurut Kusumadewi (2003), antara lain:

1. Memiliki fasilitas informasi yang handal.
2. Mudah dimodifikasi.
3. Dapat digunakan dalam berbagai jenis komputer.
4. Memiliki kemampuan untuk belajar beradaptasi.

Menurut Dewi (2015), banyak manfaat yang dapat diambil dengan adanya sistem pakar, antara lain:

1. Membuat seseorang yang awam dapat bekerja seperti layaknya seorang pakar.
2. Menyimpan pengetahuan dan keahlian para pakar.
3. Meningkatkan output dan produktivitas.
4. Meningkatkan kualitas.
5. Mampu mengambil dan melestarikan keahlian para pakar (terutama yang termasuk keahlian langka).
6. Memiliki kemampuan untuk mengakses pengetahuan.
7. Memiliki kemampuan memecahkan masalah yang kompleks.
8. Menghemat waktu dalam pengambilan keputusan.

Disamping memiliki beberapa keuntungan, sistem pakar juga memiliki beberapa kelemahan, antara lain:

1. Biaya yang diperlukan untuk membuat dan memeliharanya sangat mahal.
2. Sulit dikembangkan. Hal ini tentu saja erat kaitannya dengan ketersediaan pakar dibidangnya.
3. Sistem pakar tidak 100% bernilai benar (Kusumadewi, 2003)

Sistem pakar terdiri dari 2 bagian pokok, yaitu lingkungan pengembang (*development environment*) dan lingkungan konsultasi (*consultation environment*). Lingkungan pengembangan digunakan sebagai pembangun sistem pakar baik dari segi pembangunan komponen maupun basis pengetahuan. Lingkungan konsultasi digunakan oleh seseorang bukan ahli untuk berkonsultasi.

Komponen-komponen yang ada pada sistem pakar terdapat:

1. Antar Muka Pengguna

Antarmuka pengguna merupakan mekanisme yang digunakan oleh pengguna dan sistem pakar untuk berkomunikasi. Antarmuka menerima informasi dari pemakai dan mengubahnya ke dalam bentuk yang dapat diterima oleh sistem. Pada bagian ini terjadi dialog antara program dan pemakai, yang memungkinkan sistem pakar menerima instruksi dan informasi (*input*) dari pemakai, juga memberikan informasi (*output*) kepada pemakai.

2. Motor inferensi (*inference engine*).

Program yang berisi metodologi yang digunakan untuk melakukan penalaran terhadap informasi-informasi dalam basis pengetahuan dan blackboard serta digunakan untuk memformulasikan konklusi. Ada 2 elemen utama dalam motor inferensi, yaitu: *Interpreter*: mengeksekusi item-item agenda yang terpilih dengan menggunakan aturan-aturan dalam basis pengetahuan yang sesuai. *Scheduler*:

3. Blackboard

Blackboard merupakan area kerja memori tempat pendeskripsian masalah yang diberikan oleh data input, digunakan juga untuk perekaman hipotesis dan keputusan sementara. Ada 3 tipe keputusan yang dapat direkam, yaitu:

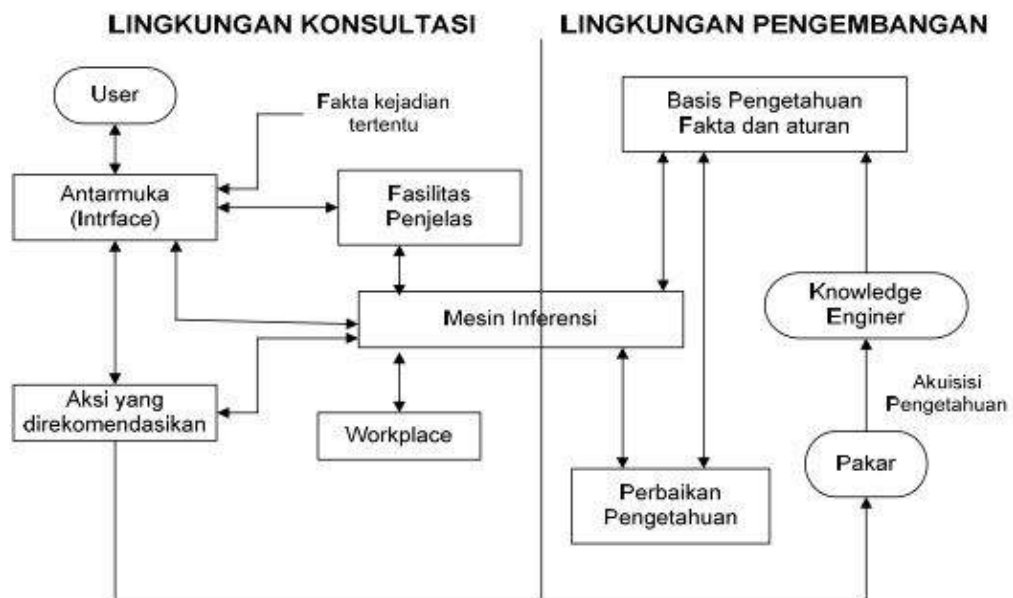
- a. Rencana : bagaimana menghadapi masalah.
- b. Agenda : aksi – aksi yang potensial yang sedang menunggu untuk dieksekusi.

### 2.2.3 Struktur Sistem Pakar

1 bagian utama sistem pakar :

- Lingkungan pengembangan (*development environment*) Digunakan untuk memasukkan pengetahuan pakar ke dalam lingkungan sistem pakar.
- Lingkungan konsultasi (*consultation environment*) Digunakan oleh pengguna yang bukan pakar untuk memperoleh pengetahuan pakar.

Arsitektur Sistem Pakar



Gambar 2.1 Arsitektur Sistem Pakar

Komponen-komponen yang terdapat dalam arsitektur/shuktur sistem pakar

1. Antarmuka Pengguna (*User Interface*)

Merupakan mekanisme yang digunakan oleh pengguna dan sistem pakar untuk berkomunikasi. Antarmuka menerima informasi dari pemakai dan mengubahnya ke dalam bentuk yang dapat diterima oleh sistem. Selama itu antarmuka menerima dari sistem dan menyajikannya ke dalam bentuk yang dapat dimengerti oleh pemakai.

2. Basis Pengetahuan

Basis pengetahuan mengandung pengetahuan untuk pemahaman, formulasi dan penyelesaian masalah. Komponen sistem pakar ini disusun atas 2 elemen dasar, yaitu

📖 Fakta : informasi tentang obyek dalam area permasalahan tertentu

📖 Aturan : informasi tentang cara bagaimana memperoleh fakta baru dari fakta yang telah diketahui.

3. Akuisitasi Pengetahuan (*Knowledge Acquisition*)

Akuisisi pengetahuan adalah akumulasi, transfer, dan transformasi keahlian dalam menyelesaikan masalah dan sumber pengetahuan ke dalam program komputer. Dalam tahap ini knowledge engineer berusaha menyerap pengetahuan untuk selanjutnya ditransfer ke dalam basis pengetahuan. Pengetahuan diperoleh dan pakar, dilengkapi dengan buku, basis data, laporan penelitian dan pengalaman pemakain

## Metode akuisasi pengetahuan

- Wawancara

Metode yang paling banyak digunakan, yang melibatkan pembicaraan dengan pakar secara langsung dalam suatu wawancara.

- Analisis protocol

Dalam metode ini pakar diminta untuk melakukan suatu pekerjaan dan mengungkapkan proses pemikirannya dengan menggunakan kata-kata. Pekerjaan tersebut direkam, dituliskan, dan dianalisis.

- Observasi pada pekerjaan pakar

Pekerjaan dalam bidang tertentu yang dilakukan pakar direkam dan diobservasi.

- Induksi aturan dan contoh

Induksi adalah suatu proses penalaran dari khusus ke umum. Suatu sistem induksi aturan diberi contoh-contoh dari suatu masalah yang hasilnya telah diketahui. Setelah diberikan beberapa contoh, sistem induksi aturan tersebut dapat membuat aturan yang benar untuk kasus-kasus contoh. Selanjutnya aturan dapat digunakan untuk menilai kasus lain yang hasilnya tidak diketahui.

## 4. Mesin/Motor Inferensi (*inference engine*)

Komponen ini mengandung mekanisme pola pikir dan penalaran yang digunakan oleh pakar dalam menyelesaikan suatu masalah. Mesin *inferensi* adalah program komputer yang membenkan metodologi untuk penalaran tentang informasi yang ada dalam basis pengetahuan dan dalam workplace, dan untuk memformulasikan kesimpulan.

### 5. *Workplace / Blackboard Workplace*

Merupakan area dan sekumpulan memori kerja (*working memory*), digunakan untuk merekam kejadian yang sedang berlangsung termasuk keputusan sementara. Ada 3 keputusan yang dapat direkam :

- Rencana : bagaimana menghadapi masalah
- Agenda : aksi-aksi yang potensial yang sedang menunggu untuk dieksekusi
- Solusi : calon aksi yang akan dibangkitkan

### 6. Fasilitas Penjelasan

Adalah komponen tambahan yang akan meningkatkan kemampuan sistem pakar. Digunakan untuk melacak respon dan memberikan penjelasan tentang kelakuan sistem pakar secara interaktif melalui pertanyaan:

- Mengapa suatu pertanyaan ditanyakan oleh sistem pakar ?
- Bagaimana konklusi dicapai ?
- Mengapa ada alternatif yang dibatalkan ?
- Rencana apa yang digunakan untuk mendapatkan solusi ?

### 7. Perbaikan Pengetahuan

Pakar memiliki kemampuan untuk menganalisis dan meningkatkan kinerjanya serta kemampuan untuk belajar dari kinerjanya. Kemampuan tersebut adalah penting dalam pembelajaran terkomputerisasi, sehingga program akan mampu menganalisis penyebab kesuksesan dan kegagalan yang dialaminya dan juga mengevaluasi apakah pengetahuan-pengetahuan yang ada masih cocok untuk digunakan di masa mendatang.

### 2.2.3 Cabai

Cabai atau Cabai (*Capsium annum L.*) adalah komoditas sayuran yang banyak mendapatkan perhatian karena memiliki nilai ekonomis yang cukup tinggi. Kebutuhan cabai semakin lama semakin meningkat setiap tahunnya sejalan dengan berkembangnya sebuah industri yang membutuhkan bahan baku cabai, tanaman ini menjadi salah satu jenis tanaman yang sangat dibutuhkan oleh semua orang di Indonesia karena mayoritas penduduk Indonesia menyukai makanan pedas.

#### Ciri-ciri Penyakit Pada Tanaman Cabai

##### 1. a. Penyakit Antraknosa

##### b. Gejala :

- ☐ Mati pucuk.
- ☐ Daun, ranting, dan cabang busuk kering berwarna coklat kehitam-hitaman.
- ☐ Buah timbul bercak lunak berwarna hitam dan busuk lunak.
- ☐ Pada batang *acervuli* cendawan terlihat berupa benjolan.

##### 2. a. Penyakit Bercak Daun *Serkospora*

##### b. Gejala :

- ☐ Bercak berbentuk bulat berwarna abu-abu tua dan warna coklat dipinggirannya.
- ☐ Daun menjadi tua (menguning) sebelum waktunya.
- ☐ Bercak daun berukuran sekitar 0,25 cm.
- ☐ Sering terjadi sobekan dipusat daun

##### 3. a. Penyakit Busuk daun *Fitoftora*.

##### b. Gejala :

- ☐ Busuk batang menjadi kering dan mengeras
- ☐ Seluruh daun menjadi layu
- ☐ Pada daun timbul bercak putih seperti tersiram air
- ☐ -Serangan pada akar mengakibatkan tanaman layu, mengering, dan mati.



Di Mowewe mulai banyak petani yang membudidayakan cabai merah/ Cabai. Banyak petani Indonesia yang membudidayakan tanaman Cabai untuk rotasi dalam bercocok tanam atau menjadi mata pencaharian dari menanam Cabai. Tetapi masih banyak dari petani yang belum dapat menanam tanaman Cabai sesuai dengan standart yang ada, yang bisa mengakibatkan terserangnya tanaman oleh hama dan penyakit, kebanyakan tidak mengetahui dengan pasti jenis penyakit dan hama yang sedang mengenai Cabai. Kebanyakan petani akan menduga duga penyakit apa yang sedang menyerang tanaman Cabai dan mencoba alternatif pengobatan dan bisa akan mengakibatkan para petani merugi, karena tanamannya mati, harga jual yang rendah dan hasil panen yang sedikit. Sehingga petani membutuhkan sebuah ilmu pengetahuan dari pakar untuk meningkatkan kualitas dalam menanam dan juga bisa meningkatkan harga jual tanaman Cabai. Pengetahuan yang akan direpresentasikan ke dalam sistem pakar dengan menggunakan data yang dipindahkan ke seluruh jaringan kemudian direpresentasikan ke dalam sistem pakar dengan unsur ketidakpastian. Cara untuk mengatasi permasalahan ketidakpastian dengan menggunakan metode *Forward Chaining dan Certainty Factor*.

#### **2.2.4 Diagnosis Penyakit**

Dalam tahap ini setelah memasukkan gejala pada sistem akan memberikan hasil diagnosis jenis penyakit apa yang menyerang tanaman Cabai pada petani dan akan memberikan solusi pada petani untuk menanggulangi penyebaran penyakit pada tanaman Cabai. Dalam beberapa tahun terakhir ini sudah banyak petani di Mowewe yang mulai menanam Cabai sebagai salah satu alternatif dalam bercocok tanam. Cara perawatan yang salah atau tidak sesuai dengan standart yang ada, dapat mengakibatkan tanaman Cabai rentan terkena hama penyakit dan dapat mengakibatkan hasil panen tidak bisa maksimal, hasil penjualan yang rendah dan bahkan bisa mengakibatkan gagal panen. Hama utama pada tanaman Cabai antara lain ulat hama, ulat grayak, kutu daun, thrips, tungau, lalat buah hama tersebut yang bisa mengakibatkan tanaman Cabai terkena penyakit. Penyakit utama pada tanaman cabai antara lain adalah busuk buah bercak

daun, layu fusarium, penyakit virus. Dalam budidaya tanaman Cabai atau cabai, salah satu masalah yang dihadapi adalah perubahan iklim yang cukup ekstrim. Hal ini mengakibatkan intensitas serangan hama dan penyakit semakin meningkat. Disamping itu, penggunaan insektisida yang berlebihan merupakan faktor yang mempengaruhi tanaman Cabai. Pengendalian hama dan penyakit tanaman cabai adalah salah satu hal yang menentukan produktivitas tanaman Cabai itu sendiri. Oleh sebab itu, petani Cabai perlu memiliki pengetahuan tentang penanggulangan masalah hama dan penyakit pada tanaman Cabai.

### 2.2.5 Metode *Certainty Factor*

Dalam aplikasi sistem pakar terdapat suatu metode untuk menyelesaikan masalah ketidakpastian data, salah satu metode yang digunakan adalah faktor kepastian (*certainty factor*). Ada dua macam faktor kepastian yang digunakan, yaitu faktor kepastian yang diisikan oleh pakar bersama dengan aturan dan faktor kepastian yang diberikan pengguna. *Certainty factor* (CF) menunjukkan ukuran kepastian suatu fakta atau aturan (Kusumadewi, 2003). Rumus umum menentukan *Certainty Factor* adalah sebagai berikut.  $CF[h,e] = MB[h,e] - MD[H,E]$

Dimana :

$CF[h,e]$  adalah faktor kepastian,

$MB[h,e]$  adalah ukuran kepercayaan terhadap hipotesis  $h$ , jika diberikan evidence  $e$  (antara 0 dan 1), dan

$MD[h,e]$  adalah ukuran ketidakpercayaan terhadap evidence  $h$ , jika diberikan evidence  $e$  (antara 0 dan 1).

Untuk menentukan nilai CF akhir pada suatu diagnosis maka menggunakan rumus CF paralel sebagai berikut.

$$CF[h,e_1 \wedge e_2] = CF[h,e_1] + CF[h,e_2] \cdot (1 - CF[h,e_1])$$

Dengan  $CF[h,e_1 \wedge e_2]$  adalah faktor kepastian paralel,  $CF[h,e_1]$  adalah ukuran kepercayaan terhadap hipotesis  $h$ , jika diberikan evidence  $e$  pertama (antara 0 dan 1),  $CF[h,e_2]$  adalah ukuran kepercayaan terhadap hipotesis  $h$ , jika diberikan evidence  $e$  kedua (antara 0 dan 1). Di dalam MYCIN terdapat

aturan untuk menggabungkan *evidence anteseden* yang terdapat dalam sebuah kaidah (Hartati & Iswanti, 2013). Hal ini dapat dilihat pada Tabel 2.2

Tabel 2.2 Kombinasi *evidence anteseden*

Evidence	Nilai Ketidakpastian
E1 and E2	$\text{Min}[\text{CF}(\text{H}, \text{E1}), \text{CF}(\text{H}, \text{E2})]$
E1 or E2	$\text{Max}[\text{CF}(\text{H}, \text{E1}), \text{CF}(\text{H}, \text{E2})]$
Not E	$-\text{CF}(\text{H}, \text{E})$

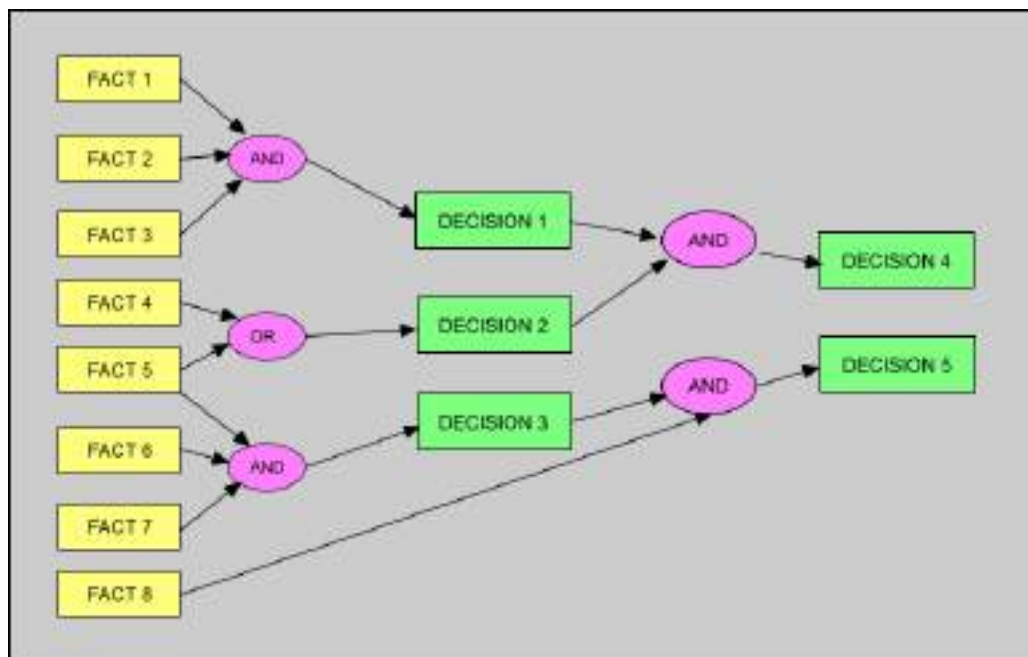
Tabel 2.3. Tabel Nilai Ketidakpastian

Nilai CF	Teorema bayes
0 – 0,2	Tidak Mungkin
0,3 – 0,4	Mungkin
0,5 – 0,6	Kemungkinan besar
0,7 – 0,8	Hampir Pasti
0,9 – 1	Pasti

Pada tabel 2.3 terdapat *range* nilai ketidakpastian, mmulai dari 0-0,2 dengan keterangan tidak mungkin, 0,3 – 0,4 dengan keterangan mungkin, 0,5 – 0,6 dengan keterangan kemungkin besar, 0,7 – 0,8 dengan keterangan hamper pasti dan 0,9 – 1 dengan keterangan pasti.

#### 2.2.6. *Forward Chaining*

*Forward Chaining* (Runut maju) merupakan proses perunutan yang dimulai dengan menampilkan kumpulan data atau fakta yang meyakinkan menuju konklusi akhir (Hartati dan Iswanti, 2008). Dalam melakukan perunutan untuk mencocokkan fakta yang tersimpan dalam basis pengetahuan menggunakan beberpa aturan inferensi untuk sampai pada kesimpulan. Menurut Akil (2017) Mesin inferensi yang menggunakan forward chaining mencari aturanaturan inferensi sampai menemukan satu dari anticedent (klausa IF-THEN) yang benar, ketika aturan tersebut ditemukan maka mesin pengambil keputusan dapat membuat kesimpulan serta akan terus mengulang melalui proses ini sampai sasaran ditemukan. Berikut ini adalah contoh algoritma *forward chaining*.



**Gambar 2.2 Algoritma Forward Chaning**

### 2.2.7. Perancangan Sistem

Sistem yang akan dibangun adalah berbasis web, dimana website ini dapat menampilkan data Jenis Cabai dan jenis penyakitnya, serta nilai probabilitas kepastian hama dan penyakit pada tanaman Cabai, dan kemudian hipotesanya dipilih dengan nilai terbesar. Dengan adanya aplikasi ini Untuk mengatasi masalah ketidakpastian maka dapat digunakan penalaran statistik. *Bayesian* digunakan untuk memperbaharui tingkat kepercayaan diri dari suatu informasi dengan diagnosis secara statistik yang berhubungan dengan probabilistik serta kemungkinan dari penyakit dan gejala gejala yang berkaitan.

Nurfauzi 2010, dalam merancang basis data kita dapat melakukan dengan 2 cara yaitu :

- a. Menerapkan normalisasi pada struktur tabel yang telah di ketahui;

Nurfauzi 2010, dalam normalisasi merupakan suatu teknik untuk menghasilkan sekumpulan hubungan dengan properti yang diinginkan, yang memberikan kebutuhan data terhadap suatu perusahaan. Pada proses normalisasi selalu diuji dengan berbagai kondisi seperti menambah (*insert*), menghapus (*delete*), mengubah (*update*), melihat (*review*). Pada sebuah database dan bila terdapat kesulitan pada pengujian tersebut, maka relasi yang

harus dipecahkan lagi menjadi beberapa table sehingga bentuknya menjadi database normal.

b. Menerapkan Model ERD (*Entity Relationship Diagram*);

ERD adalah penggambaran dari sebuah kebutuhan penyimpanan data dengan cara kerja dari suatu perusahaan atau organisasi tersebut yang bebas dari ambiguitas. ERD digunakan untuk mengidentifikasi data yang akan disimpan, diolah dan diubah untuk mendukung aktifitas bisnis suatu organisasi. Malik, Nurfauzi 2010.

Basis data adalah kumpulan data yang saling berhubungan yang disimpan secara bersama – sama sedemikian rupa dan tanpa pengulangan ( redundansi ) yang tidak perlu untuk memenuhi berbagai kebutuhan. Prinsip utama database adalah pengaturan data. Tujuan utama basis data adalah kemudahan dan kecepatan dalam pengambilan kembali data. Basis data terdiri dari dua kata, yaitu basis dan data. Basis dapat diartikan sebagai tempat, sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia. Basis data sendiri dapat didefinisikan sebagai berikut.

- a. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa sehingga dapat dimanfaatkan kembali dengan cepat dan mudah.
- b. Kumpulan file atau tabel atau arsip yang saling berhubungan yang disimpan dalam media penyimpanan *elektronik*. (Fathansyah, 2004)

Dalam merancang basis data kita dapat melakukan dengan :

a. Menerapkan normalisasi pada struktur table yang telah di ketahui.

Normalisasi adalah suatu tehnik untuk mengorganisasi data kedalam tabel untuk memenuhi kebutuhan pemakai, proses normalisasi model data yaitu menemukan entitas-entitas utama model data, menemukan hubungan antara setiap entitas , menemukan atribut yang di miliki masing masing entitas.

b. Menerapkan model ERD

*Entity Relationship diagram (ERD)* merupakan suatu model jaringan yang menggunakan data yang disimpan pada sistem secara abstrak. *ERD* juga

menggambarkan hubungan antara entitas yang memiliki jumlah atribut dengan entitas yang lain dalam suatu sistem yang terintegrasi. *EDR* juga merupakan model konseptual yang dapat mendeskripsikan hubungan antara file yang akan digunakan untuk memodelkan struktur data serta hubungan antara data (Yakub, 2012).

Terdapat 4 komponen pokok sistem basis data, yaitu :

- a. Data yaitu suatu data dalam basis data yang merupakan kumpulan dari berbagai file dari aplikasi berbeda yang disusun dengan cara menghilangkan bagian-bagian yang rangkap dan masing-masing bagian dari basis data dapat diakses oleh pemakai dalam waktu yang bersamaan untuk aplikasi yang berbeda.
- b. *Hardware* yaitu Terdiri dari semua peralatan computer yang digunakan untuk mengelola sistem basis data.
- c. *Software* yaitu perantara atau interface antara pemakai dengan data
- d. *User*/pemakai.

### 2.2.7 Entity Relationship Diagram (ERD)

*ERD* merupakan salah satu bentuk pendekatan model database atau struktur data selain pendekatan hirarki, dan jaringan ( *network* ). Istilah relational menunjukkan adanya hubungan antara elemen-elemen data baik antara suatu file atau dalam satu file. *Entiti* diagram juga merupakan model data yang berkaitan erat dengan model relational. Pada model ini mempunyai 3 komponen dasar yaitu *entitas*, Atribut, dan Relasi.

*ERD* menggunakan tiga elemen ini dan memungkinkan sistem analis untuk mengambil atribut dari entiti dan mendefinisikan hubungan antara entiti, sehingga menghasilkan model database yang sederhana dan mudah untuk dipahami atau skema sistem. *ERD* berfungsi untuk menggambarkan relasi dari dua file atau dua tabel. *ERD (Entity Relationship Diagram)*, atau diagram Hubungan *Entitas* didefinisikan sebagai suatu model jaringan yang memakai susunan data yang disimpan dalam sistem secara abstrak. *ERD* terdiri dari 3 utama yaitu (Fathansyah, Ir.)

a. Entitas (*Entity*)

Entitas merupakan individu yang mewakili sesuatu yang nyata (*eksistensinya*) dan dapat dibedakan dari sesuatu yang lain. Adalah suatu hal (objek, orang, peristiwa, proses atau konsep) yang telah didefinisikan data dan entiti merupakan elemen data. Tiap entiti harus mempunyai atribut. Jenis entitas (*entity type*), dapat berupa :

1. Suatu elemen lingkungan,
2. Sumber daya,
3. Transaksi, yang berupa pentingnya bagi perusahaan sehingga didokumentasikan dengan data.

b. Atribut (*Attribute*)

adalah sifat dan karakteristik suatu entitas yang menyediakan penjelasan detail tentang entitas tersebut.

c. Relasi (*relationship*) adalah Hubungan suatu asosiasi atau kaitan antara dua entitas. Hubungan digambarkan dengan bentuk belah ketupat. Tiap belah ketupat diberi label kata kerja.

Ada 3 jenis keterkaitan, yaitu : satu – ke-satu, satu-ke-banyak, banyak-ke-banyak. Ada 3 kemungkinan hubungan yang ada yaitu :

1. *One to one* (1: 1)

Tingkat hubungan yang dinyatakan satu pada satu kejadian entitas yang pertama, hanya mempunyai satu hubungan dengan satu kejadian pada entitas yang kedua dan sebaliknya.

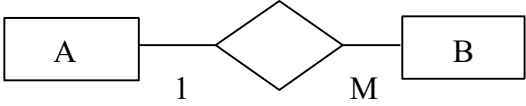
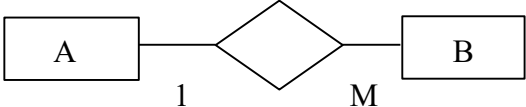
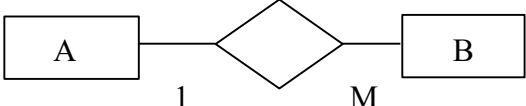
2. *One to many* (1: M)

Tingkat hubungan satu dan banyak sama dengan banyak pada satu tergantung dari arah mana hubungan tersebut dilihat.

3. *Many to many* (M:M)

Tingkat hubungan yang dinyatakan satu pada kejadian *entitas* yang pertama, mempunyai banyak hubungan dengan satu kejadian pada *entitas* yang kedua dan sebaliknya.


Tabel 2.3 Contoh ERD

Gambar	Keterangan
	Banyak ke banyak ( <i>many to many</i> )
	Banyak ke banyak ( <i>many to many</i> )
	Banyak ke banyak ( <i>many to many</i> )


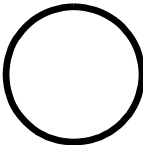

### 2.2.8 Data Flow Diagram (DFD)

Data flow diagram (DFD) merupakan alat untuk membuat diagram yang serbaguna. Data flow diagram terdiri dari notasi penyimpanan data (*data store*), proses (*process*), aliran data (*flow data*), dan sumber masukan (*entity*). DFD berfungsi untuk menggambarkan subsistem dan aliran data dalam sistem. DFD adalah suatu representasi grafik dari suatu sistem yang menggambarkan komponen dari suatu tujuan dan penyimpanan data (Yakub, 2012:155).

Tabel 2.4 Simbol-Simbol DFD

Jenis	Deskripsi	Gambar
Kesatuan Luar ( <i>Entity</i> )	Merupakan kesatuan diluar sistem yang akan memberikan input atau memberikan <i>output</i> sistem , dapat berupa orang, <i>organisasi</i> , sumber	







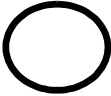


Aliran Data ( <i>Data Flow</i> )	Disimbolkan dengan anak panah, dimana aliran data mengalir diantara proses, penyimpanan data, dan kesatuan luar.	
Proses ( <i>Process</i> )	Merupakan kegiatan yang dilakukan orang atau mesin computer, dimana aliran data masuk, ditransformasikan ke aliran data keluar. Biasanya selalu menunjukkan suatu perubahan	
Penyimpanan Data ( <i>Data Store</i> )	Dapat berupa suatu file atau sistem database dari suatu computer dan diberi nama sesuai dengan data yang disimpan didalamnya.	

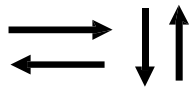
### 2.2.9 Flowchart

*Flowchart* merupakan diagram alir yang menggambarkan urutan logika dari satu prosedur pemecahan masalah. Tujuan utama dari penggunaan *flowchart* adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai rapi dan jelas dengan menggunakan simbol-simbol yang standar. *Flowchart* merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program. Bagan alir program dibuat dari verifikasi bagan alir.

a. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa sehingga dapat dimanfaatkan kembali dengan cepat dan mudah.

Tabel 2.5 Simbol-Simbol dalam *Flowchart*






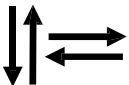
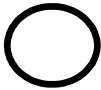


NO.	SIMBOL	KETERANGAN
1		Simbol <i>Start</i> atau <i>End</i> yang mendefinisikan awal atau akhir dari sebuah <i>flowchart</i> .
2.		Simbol pemrosesan yang terjadi pada sebuah alur kerja
3.		Simbol yang menyatakan bagian dari program
5.		Simbol <i>Input/Output</i> yang mendefinisikan masukan dan keluaran proses
6.		Simbol konektor untuk menyambung proses pada lembar kerja yang sama.
7.		Simbol untuk memutuskan proses lanjutan dari kondisi tertentu.
8.		Simbol yang menyatakan piranti keluaran, seperti layar monitor, <i>printer</i> , dll.

9.		Simbol untuk menghubungkan antar proses atau antar symbol
----	---	---

### 2.2.10 Flowmap

Menurut Jogiyanto (2005) *flowmap* merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Bagan ini menjelaskan urutan urutan dari prosedur prosedur yang ada di dalam sistem. Bagan alir sistem menunjukkan apa yang di kerjakan di sistem.

Tabel 2.6 Simbol-Simbol *Flowmap*

Simbol	Keterangan
	Simbol dokumen Menunjukkan dokumen <i>input</i> dan <i>output</i>
	Simbol kegiatan manual Menunjukkan kegiatan atau pekerjaan manual
	Simbol proses Menunjukkan kegiatan proses dari operasi program computer
	Simbol <i>keyboard</i> Menunjukkan input yang menggunakan <i>on-line keyboard</i>
	Simbol harddisk Menunjukkan input ataupun output menggunakan <i>harddisk</i>
	Simbol garis alir Menunjukkan arus dari setiap proses
	Simbol penghubung Menunjukkan penghubung ke halaman yang masih sama atau ke halaman lain
	Simbol arsip Menunjukkan pengarsipan file tanpa menggunakan computer
	Simbol keputusan Digunakan untuk suatu penyeleksian kondisi di dalam program

## 2.2.11 Pengertian *Web, MySQL, PHP*

### 2.2.11.1 *Web*

Menurut Rohi Abdulloh (2015), mendefinisikan pengertian *web* pada dasarnya adalah salah satu layanan *internet* yang paling banyak digunakan dibanding dengan layanan lain seperti *fip*, *gopher*, *news* atau bahkan *email*. Perkembangan dunia *web* sangat begitu cepat dan menggembirakan. Berbagai teknologi *web* bermunculan dan ini semua memberi manfaat dan kemudahan yang sangat besar bagi semua orang, terutama sekali bagi mereka yang ingin menguasai *web* tetapi tidak mempunyai latar belakang khusus dalam bidang komputer.

Aplikasi Berbasis *Web* (*web based applications*) adalah aplikasi yang di jalankan melalui *web browser*. Aplikasi berbasis *web* cukup sekali diinstall pada web server dan dapat diakses dari manapun, selama terdapat koneksi ke *server* (menggunakan intranet atau *internet*). Adapun keunggulan menggunakan Aplikasi Berbasis *Web* antara lain:

- a. *Platform Independent*: artinya aplikasi ini dapat dijalankan dari sistem operasi *windows*, *linux*, *BSD*, *Mac*.
- b. Untuk dijalankan di banyak komputer, anda tidak perlu install di aplikasi disetiap komputer, cukup kopi aja *script* programnya ke *server* atau salah satu komputer. Untuk komputer lain yang ingin menjalankan program ini cukup buka browsernya dan membuka alamat *host server* dimana program ini disimpan.
- c. Aplikasi ini dapat dijalankan dari jarak jauh dengan menggunakan *internet*.
- d. Aplikasi dapat dijalankan menggunakan *PDA/Smartphone* yang telah menggunakan browser canggih.

Aplikasi berbasis web dibuat dengan menggunakan program *server side*, contohnya *ASP*, *JSP*, *PHP*, dan bahasa lainnya.

*Website* (*Situs Web*) merupakan kumpulan dari halaman-halaman web yang berhubungan dengan file-file lain yang terkait. Dalam sebuah *website*

terdapat suatu halaman yang dikenal dengan sebutan *home page*. *Home page* adalah sebuah halaman yang pertama kali dilihat ketika seseorang mengunjungi *website*. Dari *home page*, pengunjung dapat mengklik *hyperlink* untuk pindah ke halaman lain yang terdapat dalam *website* tersebut (Jhonsen 2004)

#### 2.2.11.2 MySQL

MySQL adalah program *database* yang mampu mengirim dan menerima data dengan sangat cepat dan *multiuser*. MySQL memiliki dua bentuk lisensi, yaitu *Free software* dan *shareware*. MySQL yang *free software* bebas digunakan untuk keperluan pribadi atau usaha tanpa harus membeli atau membayar lisensi, yang berada di bawah lisensi GNU/GPL (*general public lisen*ce). MySQL *database server* adalah RDBMS (*Relational Database Management System*) yang dapat menangani data yang bervolume besar. Meskipun begitu, tidak menuntut *resource* yang besar. MySQL adalah sebuah manajemen sistem *database server* yang mampu menangani beberapa user, yaitu mampu menangani beberapa instruksi sekaligus dari beberapa user dalam satu waktu. dan, MySQL merekam semua data *user* di dalam sistemnya dalam tabel *user*. (Wahana Komputer, 2010)

MySQL (*My Structured Query Language*) menurut Hirin dan Virgi (2011), MySQL adalah salah satu perangkat lunak sistem manajemen basis data (*database*) SQL atau sering disebut dengan DBMS (*Database Mnajemen System*). Berbeda dengan basis data konvensional seperti. *Dat*, *.dbf*, *Mdb*, MySQL memiliki kelebihan yaitu bersifat *multithread*, dan *multi-userd* serta mendukung sistem jaringan. MySQL didistribusikan secara gratis di bawah lisensi GNU *General Public Lisen*ce (GPL), namun ada juga versi komersial bagi kalangan tertentu yang menginginkannya. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam *database* sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoprasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoprasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu sistem *database* (DBMS) dapat diketahui dari cara kerja optimizer-nya Dalam melakukan proses

perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya.

Menurut Raharjo (2011), “MySQL merupakan RDBMS (atau *server database*) yang mengelola *database* dengan cepat manampung dalam jumlah sangat besar dan dapat di akses oleh banyak user”.

Menurut Kadir (2008), “MySQL adalah sebuah *software open source* yang digunakan untuk membuat sebuah *database*”. Berdasarkan pendapat yang dikemukakan di atas dapat ditarik kesimpulan bahwa MySQL adalah suatu *software* atau program yang digunakan untuk membuat sebuah *database* yang bersifat *open source*.

### 2.2.11.3 PHP

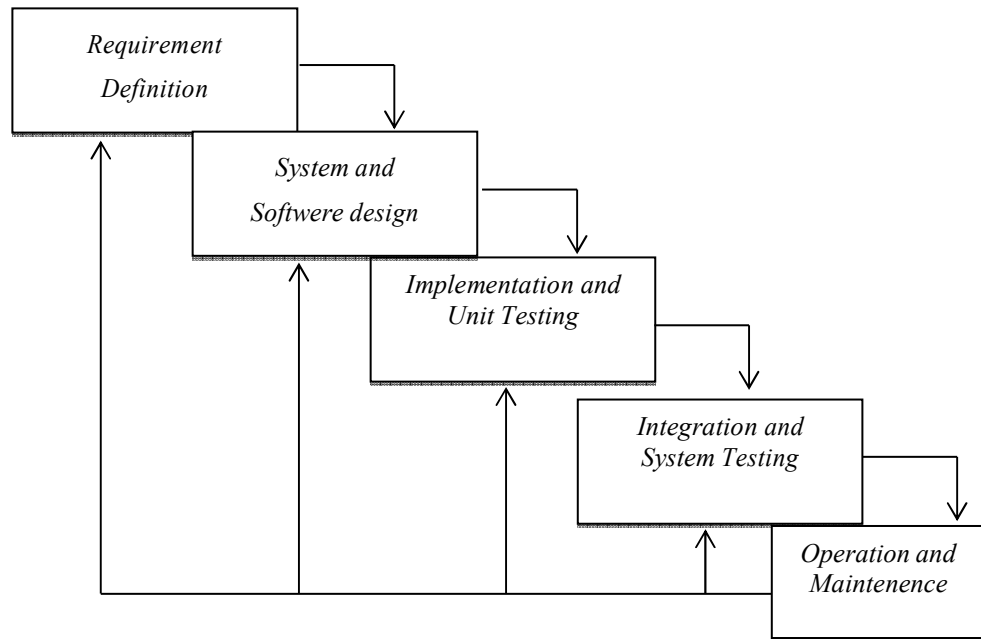
PHP atau kependekan dari *Hipertext Preprocessor* adalah salah satu bahasa pemrograman *Open Source* yang sangat cocok atau dikusukan untuk pengembangan *Web* dan dapat ditanamkan pada skrip HTML (Hirin dan Virgi, 2011).

Menurut kutipan buku Bimo Sunarfrihantono , PHP adalah bahasa *scripting* yang menyatu dengan HTML dan dijalankan pada *server side*. PHP berbeda dengan HTML karena dengan PHP kita tidak langsung menuliskan yang ingin dimunculkan di *browser*, tetapi kita menuliskan kode yang memerintahkan *server* untuk memproses dan menghasilkan halaman *web* yang diinginkan. Artinya semua sintak yang kita berikan akan sepenuhnya dijalankan pada *server* sedangkan yang dikirimkan ke *browser* hanya hasilnya saja. Aplikasi PHP juga memanfaatkan HTTP untuk komunikasi antara *server* dan *client*. HTTP menjelaskan tata cara suatu *client* informasi dari *server* dan mengambil informasi yang diminta termasuk aplikasi tersebut didapat atau tidak.

### 2.2.12 Metode Pengembangan Sistem

Sommerville (2011) mengatakan model *waterfall* mengambil proses dasar spesifikasi, pengembangan, validasi dan evolusi dan mewakili kegiatan

tersebut sebagai tahap proses terpisah seperti spesifikasi persyaratan, perancangan perangkat lunak, implementasi, pengujian dan seterusnya. Berikut adalah gambar model *waterfall*, untuk lebih jelas dapat di lihat pada gambar dengan tahapan.



Gambar 2.3 Tahapan model *Waterfall* (Sommerville, 2011)

### 1. *Requirement Definition*

Seluruh kebutuhan *software* harus bisa didapatkan dalam fase ini, termasuk didalamnya kegunaan *software* yang diharapkan pengguna dan batasan *software*. Informasi ini biasanya dapat diperoleh melalui wawancara, *survey* atau diskusi. Informasi tersebut dianalisis untuk mendapatkan dokumentasi kebutuhan pengguna untuk digunakan pada tahap selanjutnya.

### 2. *System and Software Design*

Tahap ini dilakukan sebelum melakukan *coding*. Tahap ini bertujuan untuk memberikan gambaran apa yang seharusnya dikerjakan dan bagaimana tampilannya.

### 3. *Implementation and Unit Testing*

Dalam tahap ini dilakukan pemrograman. Pembuatan software dipecah menjadi modul-modul kecil yang nantinya akan digabungkan dalam tahap berikutnya.

### 4. *Integration System Testing*

Di tahap ini dilakukan penggabungan modul- modul yang sudah dibuat dan dilakukan pengujian, ini dilakukan untuk mengetahui apakah software yang dibuat telah sesuai dengan desainnya dan masih terdapat kesalahan atau tidak.

### 5. *Operation and Maintenance*

Ini merupakan tahap terakhir dalam model *waterfall*. Yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya. Perbaikan *implementasi* unit sistem dan peningkatan jasa sistem sebagai kebutuhan baru.

## 2.2.13 Teknik pengujian sistem

Teknik pengujian System menggunakan *blackbox (blackbox testing)* adalah salah satu metode pengujian perangkat lunak yang berfokus pada sisi fungsionalitas, khususnya pada input dan output aplikasi (apakah sudah sesuai dengan apa yang diharapkan atau belum). Tahap pengujian atau testing merupakan salah satu tahap yang harus ada dalam sebuah siklus pengembangan perangkat lunak (selain tahap perancangan atau desain). Pengujian ini fokus kepada persyaratan fungsional perangkat lunak. Pengujian ini memungkinkan mahasiswa yang melakukan penelitian mendapatkan serangkaian kondisi input yang memenuhi persyaratan fungsional suatu program. Kasus uji ini bertujuan untuk menunjukkan fungsi perangkat lunak tentang cara beroperasinya. Teknik pengujian ini berfokus pada domain informasi dari perangkat lunak, yaitu melakukan kasus uji dengan mempartisi domain *input* dan *output* program. Metode *black-box* memungkinkan perancang perangkat lunak mendapatkan serangkaian kondisi *input* yang sepenuhnya menggunakan semua persyaratan



fungsional untuk suatu program. Pengujian ini berusaha menemukan kesalahan dalam kategori fungsi-fungsi yang tidak benar atau hilang, kesalahan *interface*, kesalahan dalam struktur data atau akses basis data eksternal, kesalahan kinerja, dan inisialisasi dan kesalahan terminal (Pressman, 2001).

## BAB III

### METODE PENELITIAN

#### 3.1 Jadwal Penelitian

Tempat yang dijadikan penelitian adalah Di Desa Mowewe. Sedangkan waktu penelitian ini dilaksanakan pada bulan april juni 2021. Untuk lebih jelas penelitian ini, berikut jadwal penelitian :

Tabel 3.1 Jadwal Penelitian

No	Keterangan	Bulan											
		April				Mei				Juni			
		1	2	3	4	1	2	3	4	1	2	3	4
1	<i>Requirement Analysis and Definition</i>												
2	<i>System and Software Design</i>												
3	<i>Implementation and Unit Testing</i>												
4	<i>Integration and System Testing</i>												
5	<i>Operation and Maintenance</i>												

#### 3.2 Teknik Pengumpulan Data

Metode pengumpulan data yang digunakan dalam penyusunan tugas akhir ini adalah sebagai berikut :

a. Metode Penelitian Langsung (*Observation*)

Tahap obeservasi ini, peneliti mengumpulkan data dengan mengamati dan mencatat peristiwa dan keluhan pada objek penelitian secara langsung di desa Mowewe

b. Metode Wawancara (*interview*)

Pada tahap ini, peneliti melakukan *interview* Tanya jawab langsung antara peneliti dengan petani cabai desa Mowewe.

c. Metode Pustaka (*Library*)

Setelah melakukan wawancara dan observasi langsung, peneliti mengumpulkan sumber-sumber bacaan yang dapat menunjang penelitian yang diambil, seperti penelitian terdahulu yang memiliki kesamaan dalam penelitian tentang sistem pakar.

### 3.3 Metode Pengembangan Sistem

Tahap ini penulis menggunakan Metode *waterfall* dengan tahapan :

#### a. *Requirement Analysis and Definition*

Pada tahapan *Requirement Analysis and Definition*, sistem pendukung keputusan, kendala, dan tujuan ditetapkan melalui konsultasi peneliti dengan petani cabai desa Mowewe. Kemudian didefinisikan secara rinci dan berfungsi sebagai spesifikasi sistem. Proses menganalisis dan pengumpulan data sistem yang sesuai dengan domain informasi tingkah laku, unjuk kerja, dan antar muka (*interface*) yang diperlukan.

#### b. *System and Software Design*

Dalam tahap *System and Software Design* ini difokuskan pada empat atribut, struktur data, arsitektur perangkat lunak, representasi antarmuka, detail (*algoritma*) *procedural*. Yang dimaksud struktur data adalah representasi dari hubungan logis antara elemen-elemen data individual. Dalam tahap ini penulis akan merancang desain dan model aplikasi yang dikembangkan berdasarkan hasil analisa pada tahap *Requirement Analysis and Definition*. Tahap perencanaan sistem atau desain yang digunakan untuk membuat spesifikasi secara rinci mengenai rancangan dari kebutuhan untuk pengembangan sistem sistem pakar yang akan dibuat

#### c. *Implementation and Unit Testing*

Dalam tahap *Implementation and Unit Testing*, desain perangkat lunak direalisasikan sebagai seperangkat program atau unit program. Unit pengujian melibatkan memverifikasi bahwa setiap unit memenuhi spesifikasinya. Pada tahap ini juga disebut tahap pengkodean (*coding* PHP) yang merupakan proses menerjemahkan desain ke dalam suatu bahasa yang dimengerti oleh komputer, pengembangan ini merupakan proses menghasilkan sistem pendukung keputusan yang sesuai dengan yang diinginkan.

#### d. *Integration and Black Box Testing*

Dalam tahap *Integration and System Testing*, unit program individu atau program diintegrasikan dan diuji sebagai sistem yang lengkap untuk memastikan bahwa perangkat lunak persyaratan telah dipenuhi. Proses

pengujian berfokus pada logika internal *software*, memastikan bahwa semua pernyataan sudah diuji, dan pada eksternal fungsional, yaitu mengarahkan pengujian untuk menemukan kesalahan-kesalahan dan memastikan bahwa input yang dibatasi akan memberikan hasil yang sesuai dengan yang dibutuhkan. Sistem pendukung keputusan yang dihasilkan akan memasuki proses validasi oleh petani cabai Desa Mowewe untuk mengetahui kelayakan sistem yang dikembangkan. Selain itu dapat memperoleh saran dan rekomendasi pengembangan.

e. *Operation and Maintenance*

Tahap *Operation and Maintenance* adalah tahap terpanjang pada fase siklus *waterfall* (siklus hidup). Sistem ini dipasang dan dimasukkan kedalam penggunaan yang sederhana. Pemeliharaan melibatkan mengoreksi kesalahan yang tidak ditemukan di awal tahap siklus *waterfall* tersebut, meningkatkan pelaksanaan sistem unit dan meningkatkan pelayanan sistem sebagai persyaratan baru yang ditemukan.

### 3.4 Analisis kebutuhan

a. Perangkat Keras

Sistem ini dapat dibangun pada sistem yang peneliti pakai dengan spesifikasi sebagai berikut :

Laptop Acer Aspire V5-471G, Processor Intel(R) Core(TM) i5-3337U @ 1.80 GHz, Memori RAM 4 GB

b. Perangkat Lunak

Dalam Pembuatan sistem ini adalah perangkat yang berkaitan dengan pengembangan sistem pendukung keputusan. Adapun perangkat lunak dan sstem operasi yang digunakan adalah sebagai berikut :

1. *Sublime Text 3*

3. Sistem Operasi Windows 7 Ultimate 64 bit

4. *Xampp*

## **BAB IV**

### **HASIL DAN PEMBAHASAN**

#### **4.1 Analisis Sistem**

Pada penelitian ini dilakukan sebuah analisa sistem untuk menerapkan Sistem Pakar Diagnosis penyakit Pada Tanaman Cabai Dengan Menggunakan Metode *Forward Chaining* dan *Certainty Factor* Berbasis WEB. Analisa perangkat lunak merupakan langkah pemahaman persoalan sebelum mengambil tindakan atau keputusan penyelesaian hasil utama.

Analisa sistem pada penelitian ini bertujuan untuk merumuskan suatu permasalahan yang terjadi dan mengidentifikasi kebutuhan sistem yang diharapkan, sehingga dapat diusulkan perbaikan dan meyakinkan bahwa analisa sistem telah berjalan pada jalur yang benar.

Sistem Pakar adalah sistem informasi berbasis komputer yang menggunakan pengetahuan pakar untuk mencapai *performa* keputusan tingkat tinggi dalam domain persoalan sempit. Bagian dalam sistem pakar terdiri dari 2 komponen utama yakni berisi *knowledge base* yang berisi basis pengetahuan dan mesin *inferensi* yang menggambarkan pengetahuan.

#### **4.2 Metode *Forward Chaining***

Mesin inferensi yang menggunakan *forward chaining* mencari aturanaturan inferensi sampai menemukan satu dari *antecedent* (klausa IF-THEN) yang benar, ketika aturan tersebut ditemukan maka mesin pengambil keputusan dapat membuat kesimpulan serta akan terus mengulang melalui proses ini sampai sasaran ditemukan. Dalam metode ini segala macam permasalahan dideteksi semenjak awal komputer dirakit dan biasanya teknik ini hanya digunakan oleh orang-orang dealer komputer yang sering melakukan perakitan komputer. Pada teknik ini hanya dilakukan pendeteksian masalah secara sederhana dan dilakukan sebelum komputer dinyalakan (dialiri listrik). Metode *forward chaining* akan digunakan untuk pembuatan aturan/*rule* basis pengetahuan.

Adapun jenis-jenis penyakit tanaman cabai adalah sebagai berikut :

Tabel 4.1 Tabel Penyakit

Kode Penyakit	Penyakit
<b>P01</b>	Antraknosa
<b>P02</b>	Bercak Daun Serkospora
<b>P03</b>	Busuk daun Fitofthora

Tabel 4.1 merupakan tipe tipe penyakit tanaman cabai yang peneliti teliti berupa Antraknosa, Bercak Daun Serkospora dan Busuk daun Fitofthora.

Adapun gejala gejala penyakit saluran pernapasana adalah sebagai berikut :

Tabel 4.2 Tabel Gejala

Kode	Gejala
<b>G1</b>	Mati pucuk.
<b>G2</b>	Daun, ranting, dan cabang busuk kering berwarna coklat kehitam-hitaman.
<b>G3</b>	Buah timbul bercak lunak berwarna hitam dan busuk lunak.
<b>G4</b>	Pada batang <i>acervuli</i> cendawan terlihat berupa benjolan.
<b>G5</b>	Bercak berbentuk bulat berwarna abu-abu tua dan warna coklat di pinggirannya.
<b>G6</b>	Daun menjadi tua (menguning) sebelum waktunya.
<b>G7</b>	Bercak daun berukuran sekitar 0,25 cm.
<b>G8</b>	Sering terjadi sobekan dipusat daun.
<b>G9</b>	Busuk batang menjadi kering dan mengeras.
<b>G10</b>	Seluruh daun menjadi layu.
<b>G11</b>	Pada daun timbul bercak putih seperti tersiram air.
<b>G12</b>	Serangan pada akar mengakibatkan tanaman layu, mengering, dan mati.

Tabel 4.2 merupakan tipe tipe gejala tanaman cabai yang peneliti teliti berupa mati pucuk, daun, ranting, dan cabang busuk kering berwarna coklat kehitam-hitaman, buah timbul bercak lunak berwarna hitam dan busuk lunak, pada batang *acervuli* cendawan terlihat berupa benjolan, bercak berbentuk bulat berwarna abu-abu tua dan warna coklat di pinggirannya, daun menjadi tua (menguning) sebelum waktunya, bercak daun berukuran sekitar 0,25 cm, sering terjadi sobekan dipusat daun, busuk batang menjadi kering dan mengeras, seluruh daun menjadi layu, pada

daun timbul bercak putih seperti tersiram air dan serangan pada akar mengakibatkan tanaman layu, mengering, dan mati.

Tabel 4.3 Basis Pengetahuan

No	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	Penyakit	CF
1	*	*	*	*									P01	0.80
2	*		*	*	*								P01	0.85
3	*		*		*	*							P01	0.90
4							*	*	*	*			P02	0.80
5							*		*	*	*		P02	0.75
6	*						*	*		*			P02	0.85
7									*	*	*	*	P03	0.90
8					*				*	*		*	P03	0.70
9				*				*	*			*	P03	0.80

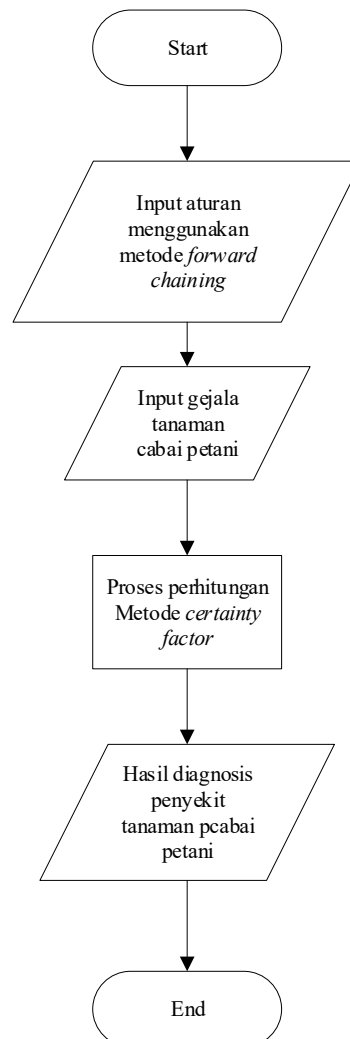
Pada tabel 4.13 basis pengetahuan dijelaskan aturan-aturan yang akan digunakan adalah sebagai berikut :

1. Aturan Ke-1 IF G1,G2,G3,G4 THEN P01 dengan nilai CF 0.80
2. Aturan Ke-2 IF G1,G3,G4,G5 THEN P01 dengan nilai CF 0.85
3. Aturan Ke-3 IF G1,G3,G5,G6 THEN P01 dengan nilai CF 0.90
4. Aturan Ke-4 IF G7,G8,G9,G10 THEN P02 dengan nilai CF 0.80
5. Aturan Ke-5 IF G7,G9,G10,G11 THEN P02 dengan nilai CF 0.75
6. Aturan Ke-6 IF G1,G7,G8,G10 THEN P02 dengan nilai CF 0.85
7. Aturan Ke-7 IF G9,G10,G11,G12 THEN P03 dengan nilai CF 0.90
8. Aturan Ke-8 IF G1,G9,G10,G12 THEN P03 dengan nilai CF 0.70
9. Aturan Ke-9 IF G4,G8,G9,G12 THEN P03 dengan nilai CF 0.80

#### 4.3 Metode *Certainty Factor*

*Certainty Factor* atau faktor kepastian diperkenalkan pertama kali pada tahun 1975 oleh shortliffe buchanan. *Certainty factor* adalah suatu metode untuk membuktikan apakah suatu fakta itu pasti ataukah tidak pasti. Metode ini sangat cocok untuk sistem pakar yang mendiagnosis sesuatu yang belum pasti. *Certainty factor* (CF) menunjukkan ukuran kepastian suatu fakta atau aturan. Rumus umum menentukan *Certainty Factor* adalah sebagai berikut.  $CF[h,e] = MB[h,e] - MD[H,E]$ .

Berikut adalah *flowchart* sistem yang peneliti rancang.



Gambar 4.1 *Flowchart* Sistem

Pada *flowchat* sistem dimulai dengan menginput aturan aturan yang telah dibuat menggunakan meote *forward chaining*, setelah itu menginput gejala gejala penyakit tanaman cabai petani dan sistem akan melakukan proses perhitungan metode *certainty factor*, sehingga menghasilkan hasil diagnosis penyakit tanaman cabai petani.

Contoh Kasus :

Seseorang berkonsultasi pada pakar untuk mengetahui penyakit jenis apa yang dialami oleh tanaman cabai yang ia tanam. Diketahui tanaman cabai yang ia tanam mengalami gejala Mati pucuk (g1) dengan nilai cf 0.40, Bercak berbentuk bulat berwarna abu-abu tua dan warna coklat di pinggirannya (g5) dengan nilai cf 0.6,



Bercak daun berukuran sekitar 0,25 cm (g7) dengan nilai cf 0.5, Seluruh daun menjadi layu (g10) dengan nilai cf 0.8 dan Serangan pada akar mengakibatkan tanaman layu, mengering, dan mati(g12) dengan nilai cf 0.7.

Penyelesaian:

Melihat pada tabel basis pengetahuan yang ada pada Tabel 4.1. di ambil aturan yang mencakup dua gejala yang dialami pemilik tanaman diantaranya adalah : aturan 2, aturan 3, aturan 4, aturan 5, aturan 6, aturan 7, aturan 8 dan aturan 9

1. Untuk aturan 2 dan 3 gejala yang ada yaitu gejala 1 dan gejala 5 dan mengambil nilai cf terendah dan mengalikan dengan nilai cf aturan :

Aturan 2

$$\begin{aligned}\text{Min (g1,g5)} &= \text{Min}(0.4, 0.6) \\ &= 0,4\end{aligned}$$

$$\begin{aligned}\text{Cf Penyakit} &= 0,4 * 0,85 \\ &= 0,34\end{aligned}$$

Aturan 3

$$\begin{aligned}\text{Min (g1,g5)} &= \text{Min}(0.4, 0.6) \\ &= 0,4\end{aligned}$$

$$\begin{aligned}\text{Cf Penyakit} &= 0,4 * 0,9 \\ &= 0,36\end{aligned}$$

$$\begin{aligned}\text{CF Maks} &= (\text{Cf Penyakit1} + \text{Cf Penyakit2}) - (\text{Cf Penyakit1} * \text{Cf Penyakit2}) \\ &= (0.34 + 0.36) - (0.34 * 0.36) \\ &= (0,7) - (0,1224) \\ &= 0,5776\end{aligned}$$

Jadi Nilai CF Penyakit untuk Antraknosa adalah 0,5776 atau sama dengan 57.76%

2. Untuk aturan 4, 5 dan 6 gejala yang ada yaitu gejala 1 dan gejala 6 dan gejala 10 mengambil nilai cf terendah dan mengalikan dengan nilai cf aturan :

Aturan 4

$$\begin{aligned}\text{Min (g7,g10)} &= \text{Min}(0.5, 0.8) \\ &= 0,5\end{aligned}$$

$$\begin{aligned}\text{Cf Penyakit1} &= 0,5 * 0,8 \\ &= 0,4\end{aligned}$$

Aturan 5

$$\begin{aligned}\text{Min (g7,g10)} &= \text{Min}(0.5, 0.8) \\ &= 0,5\end{aligned}$$

$$\begin{aligned}\text{Cf Penyakit2} &= 0,5 * 0.75 \\ &= 0,385\end{aligned}$$

$$\begin{aligned}\text{CF Maks1} &= (\text{Cf Penyakit1} + \text{Cf Penyakit2}) - (\text{Cf Penyakit1} * \text{Cf Penyakit2}) \\ &= (0,4 + 0,385) - (0,4 * 0,385) \\ &= (0,785) - (0,154) \\ &= 0,631\end{aligned}$$

Aturan 6

$$\begin{aligned}\text{Min (g1,g7,g10)} &= \text{Min}(0.4, 0.5, 0.8) \\ &= 0,4\end{aligned}$$

$$\begin{aligned}\text{Cf Penyakit2} &= 0,4 * 0.85 \\ &= 0,34\end{aligned}$$

$$\begin{aligned}\text{CF Maks2} &= (\text{Cf Penyakit1} + \text{CF Maks1}) - (\text{Cf Penyakit1} * \text{CF Maks1}) \\ &= (0,34 + 0,631) - (0,34 * 0,631) \\ &= (0,971) - (0,21454) \\ &= 0,75646\end{aligned}$$

Jadi Nilai CF Penyakit untuk Bercak Daun Serkospora adalah 0,6858 atau sama dengan 68,58%

3. Untuk aturan 7 dan 8 gejala yang ada yaitu gejala 5, gejala 10 dan gejala 12 mengambil nilai cf terendah dan mengalikan dengan nilai cf aturan :

Aturan 7

$$\begin{aligned}\text{Min (g10,g12)} &= \text{Min}(0.8, 0.7) \\ &= 0,7\end{aligned}$$

$$\begin{aligned}\text{Cf Penyakit1} &= 0,7 * 0,9 \\ &= 0,63\end{aligned}$$

Aturan 8

$$\begin{aligned}\text{Min (g5,g10,g12)} &= \text{Min}(0.5,0.8, 0.7) \\ &= 0,5\end{aligned}$$

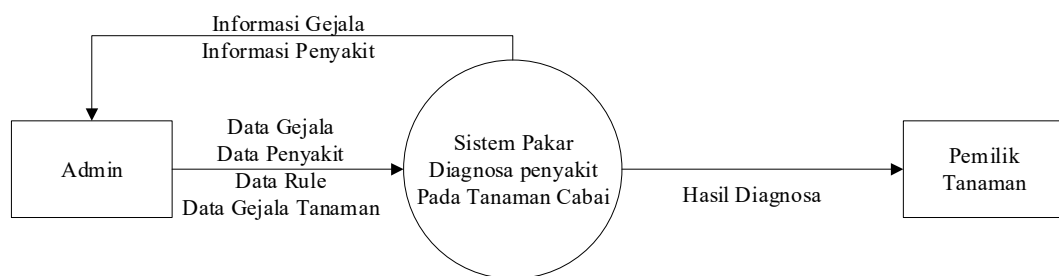
$$\begin{aligned}\text{Cf Penyakit2} &= 0,5 * 0.7 \\ &= 0,35\end{aligned}$$

$$\begin{aligned}\text{CF Maks1} &= (\text{Cf Penyakit1} + \text{Cf Penyakit2}) - (\text{Cf Penyakit1} * \text{Cf} \\ &\quad \text{Penyakit2}) \\ &= (0,63 + 0,35) - (0,63 * 0,35) \\ &= (0,98) - (0,2205) \\ &= 0,5392\end{aligned}$$

Jadi Nilai CF Penyakit untuk Busuk daun Fitofthora adalah 0.5392 atau sama dengan 53,92%

#### 4.4 Perancangan Sistem

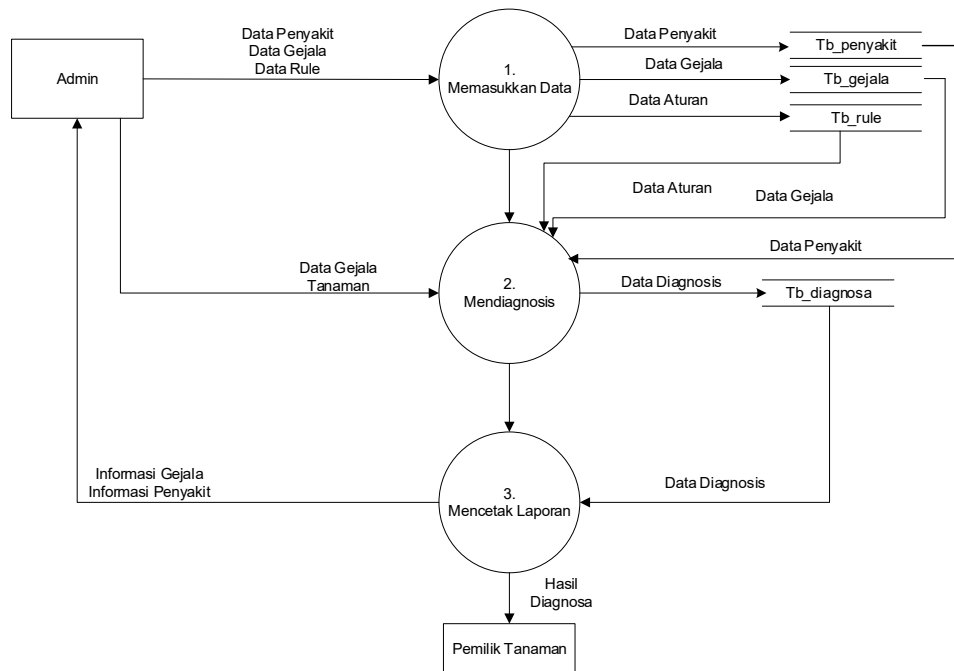
##### a. Diagram Konteks



Gambar 4.2 Diagram Konteks

Pada gambar diagram konteks, admin dapat menginput data gejala, penyakit, aturan dan data gejala tanaman ke dalam sistem. Sehingga sistem menghasilkan laporan hasil diagnosis yang dapat diberikan kepada pemilik tanaman.

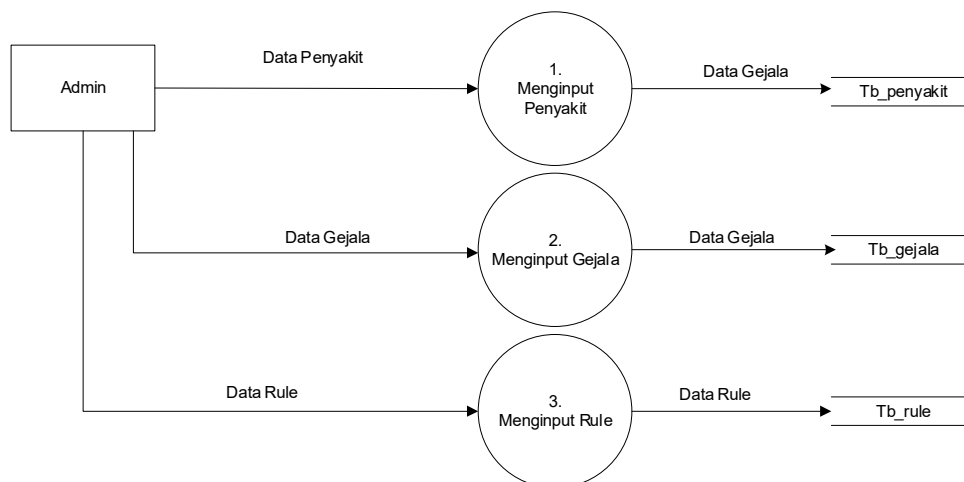
### b. Diagram Level 0



Gambar 4.3 Diagram Level 0

Pada diagram level 0, admin dapat *menginput* data gejala, penyakit dan rule yang masing masing akan tersimpan ke dalam tabel tb\_penyakit, tb\_gejala dan tb\_rule. Admin juga dapat melakukan proses diagnosis dengan memasukkan data gejala tanaman dan data tersebut akan tersimpan ke dalam tabel tb\_diagnosis. Kemudian sistem dapat menghasilkan hasil diagnosis yang dapat diberikan kepada pemilik tanaman.

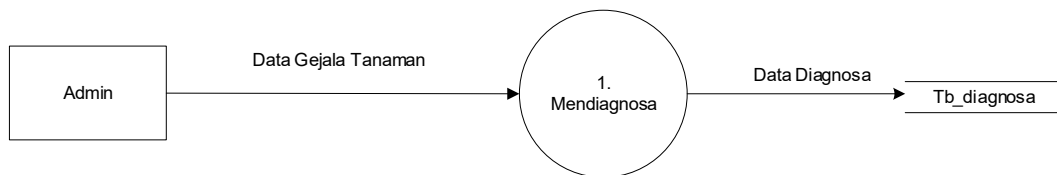
### c. Diagram Level 1 Proses 1



Gambar 4.4 Diagram Level 1 Proses 1

Pada gambar 4.4 diagram level 1 proses 1, admin dapat *menginput* data gejala, penyakit dan rule yang masing masing akan tersimpan ke dalam tabel tb\_penyakit, tb\_gejala dan tb\_rule.

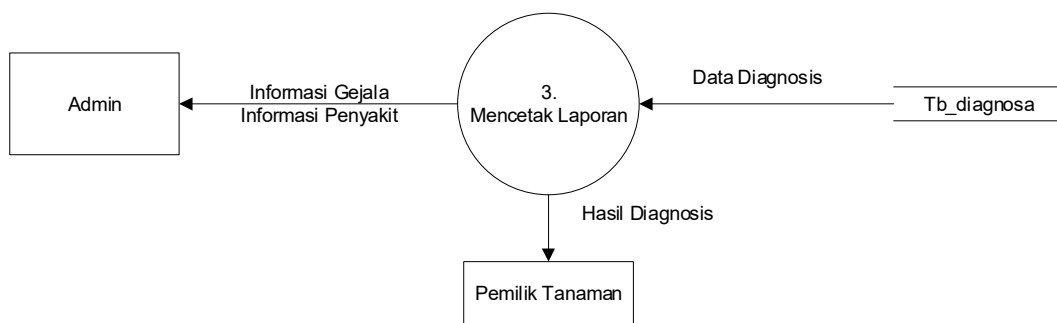
**d. Diagram Level 1 Proses 2**



Gambar 4.5 Diagram Level 1 Proses 2

Pada gambar 4.5 diagram level 1 proses 2, pemilik tanaman dapat *menginput* data gejala tanaman yang ia tanam sehingga sistem akan menghaikan hasil dianosa yang akan tersimpan pada tabel tb\_diagnosis.

**e. Diagram Level 1 Proses 3**



Gambar 4.6 Diagram Level 1 Proses 3

Pada gambar 4.6 diagram level 1 proses 3, sistem akan menghasilkan hasil diagnosis yang dapat dilihat oleh pemilik tanaman dan admin dapat melihat informasi gejala dan penyakit.

## 4.5 Perancangan Basis Data

### 4.5.1 Struktur Tabel

Adapun tabel basis data yang terdapat pada Sistem Pakar Diagnosis penyakit Pada Tanaman Cabai Dengan Menggunakan Metode *Forward Chaining* dan *Certainty Factor* Berbasis WEB adalah sebagai berikut:

a. Tabel Penyimpanan Data Penyakit

Nama Tabel : tb\_penyakit

Primary Key : id\_penyakit

Keterangan : Berisikan data data penyakit

Tabel penyimpanan data penyakit merupakan sebuah tabel yang berfungsi sebagai tempat penyimpanan data data penyakit. penyakit anggota terdiri dari id\_penyakit dan nama\_penyakit. Primary key dari tabel penyakit adalah id\_pegakit.

Tabel 4.4 Tabel Penyimpanan Data Penyakit

Nama Field	Type	Size	Keterangan
id_penyakit	Int	11	Id penyakit
Nama_ penyakit	Varchar	100	Nama penyakit

b. Tabel Penyimpanan Data Gejala

Nama Tabel : tb\_gejala

Primary Key : id\_gejala

Keterangan : Berisikan data data gejala

Tabel penyimpanan data gejala merupakan sebuah tabel yang berfungsi sebagai tempat penyimpanan data data gejala. Tabel gejala terdiri dari id\_gejala dan nama\_gejala. Primary key dari tabel gejala adalah id\_gejala.

Tabel 4.5 Tabel Penyimpanan Data Gejala

Nama Field	Type	Size	Keterangan
id_gejala	Int	11	Id gejala
Nama_gejala	Varchar	100	Nama gejala

c. Tabel Penyimpanan Data Aturan

Nama Tabel : tb\_rule

Primary Key : id\_rule

Keterangan : Berisikan data data aturan

Tabel penyimpanan data aturan merupakan sebuah tabel yang berfungsi sebagai tempat penyimpanan data data aturan. Tabel aturan terdiri dari id\_rule,

id\_gejala, id\_penyakit dan probabilitas. Primary key dari tabel aturan adalah id\_rule.

Tabel 4.6 Tabel Penyimpanan Data Aturan

Nama Field	Type	Size	Keterangan
Id rule	Int	11	Id rule
Id_gejala	Int	11	Id gejala
Id_penyakit	Int	11	Id penyakit
cf	Decimal	4,2	Nilai cf

d. Tabel Penyimpan Data Diagnosis

Nama Tabel : tb\_diagnosis

Primary Key : id\_diagnosis

Keterangan : Berisikan data data diagnosis

Tabel penyimpanan data diagnosis merupakan sebuah tabel yang berfungsi sebagai tempat penyimpanan data data diagnosis. Tabel diagnosis terdiri dari id\_diagnosis, nama\_pemilik, tgl\_diagnosis, id\_penyakit dan nilai. Primary key dari tabel diagnosis adalah id\_diagnosis.

Tabel 4.7. Tabel Penyimpanan Data Diagnosis

Nama Field	Type	Size	Keterangan
Id_diagnosis	Int	11	Id profil
Nama_pemilik	Varchar	100	Nama pemilik
Tgl_diagnosis	Date		Tanggal diagnosis
Id_penyakit	Int	11	Id penyakit
nilai	Decimal	11,5	Nilai probabilitas

e. Tabel Penyimpanan Data *User*

Nama Tabel : tb\_user

Primary Key : id\_user

Keterangan : Berisikan data data admin

Tabel penyimpanan data *user* merupakan sebuah tabel yang berfungsi sebagai tempat penyimpanan data data admin. Tabel *user* terdiri dari id\_user, *user* dan pass. Primary key dari tabel *user* adalah id\_user.

Tabel 4.8 Tabel Penyimpanan Data *User*

Nama Field	Type	Size	Keterangan
<i>Id_user</i>	Int	11	<i>Id user</i>
<i>User</i>	Varchar	20	<i>Username</i>
<i>Pass</i>	Varchar	20	<i>Password</i>

#### 4.5.2 Relasi Antar Tabel



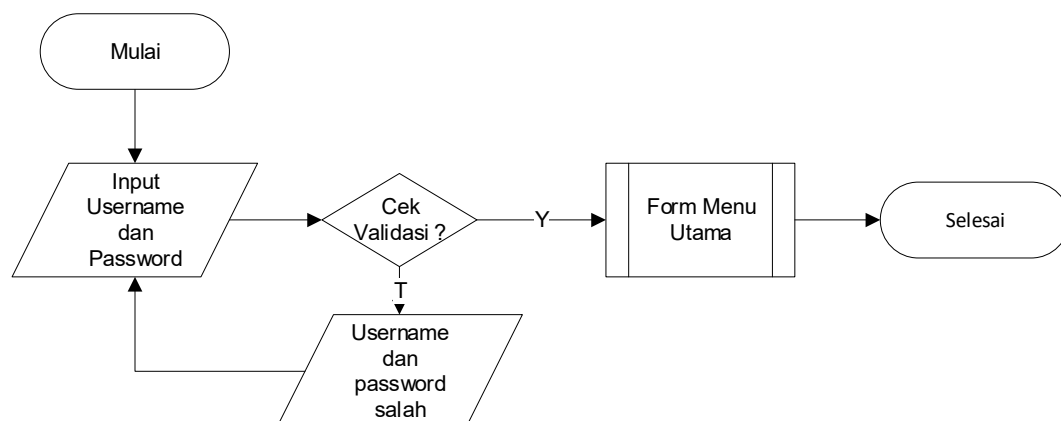
Gambar 4.7 Relasi Antar Tabel

Pada gambar relasi antar tabel terdapat empat tabel yang saling berelasi yaitu *tb\_rule* yang berelasi dengan *tb\_gejala* dan *tb\_penyakit* dan yang terakhir adalah *tb\_penyakit* yang saling berelasi dengan tabel *tb\_diagnosis*

#### 4.6 Flowchart

*Flowchart* adalah suatu skema yang menggambarkan urutan kegiatan suatu program dari awal sampai akhir. Beberapa *flowchart* yang digunakan adalah sebagai berikut:

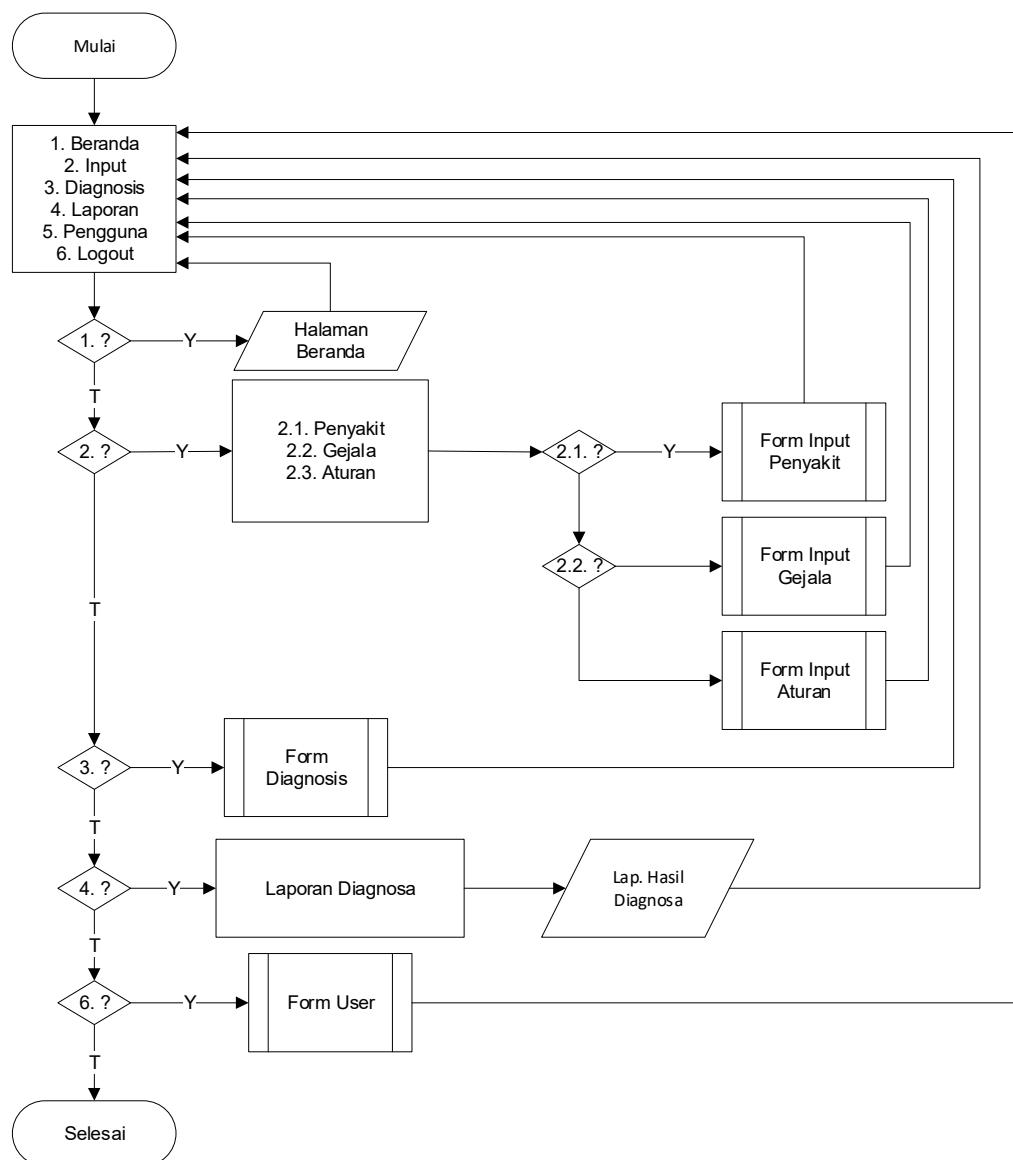
##### 4.6.1 Flowchart Login

Gambar 4.8 *Flowchart* Login Admin



Pada *flowchart* login, admin menginput *username* dan *password*, jika admin menekan tombol login maka sisten akan memvalidasi *username* dan *password* jika benar maka sistem menuju halaman menu utama dan jika salah maka sistem akan menampilkan pesan “maaf login gagal”.

#### 4.6.2 Flowchart Menu Utama



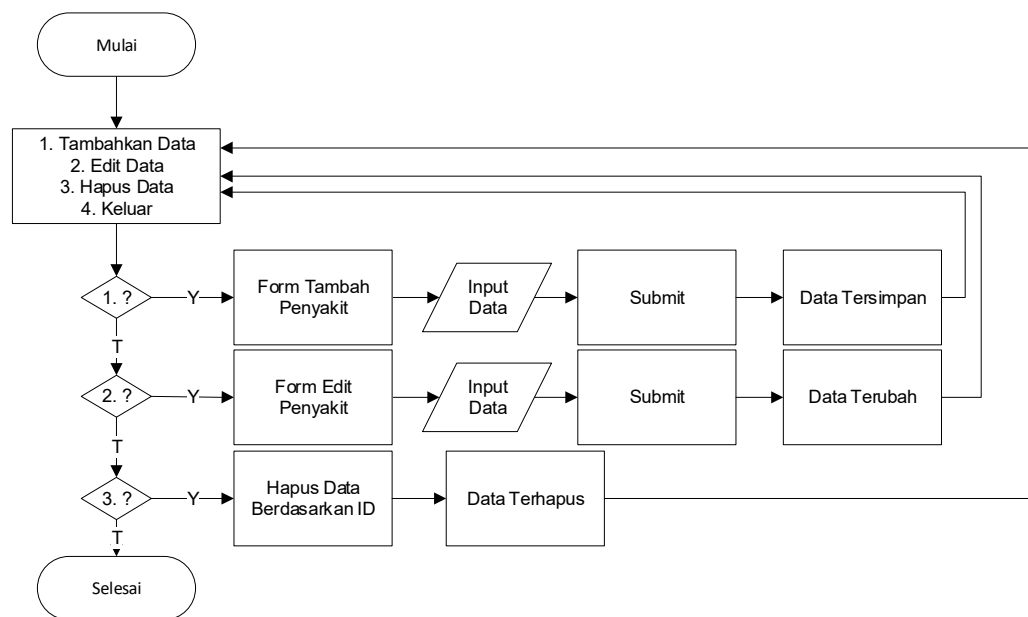
Gambar 4.9 Flowchart Menu Utama

Pada *flowchart* menu utam admin, terdapat enam menu utama yaitu beranda yang jika diklik akan menampilkan halaman beranda, menu *input* yang jika diklik akan menampilkan submenu penyakit, gejala dan rule, menu diagnosis yang jika

diklik akan menampilkan halaman diagnosis, menu *output* yang jika diklik akan menampilkan submenu laporan diagnosis, *user* yang jika diklik akan menampilkan halaman *user* dan menu logout yang jika diklik akan kembali ke halaman login..

#### 4.6.3 Flowchart Form Input Penyakit

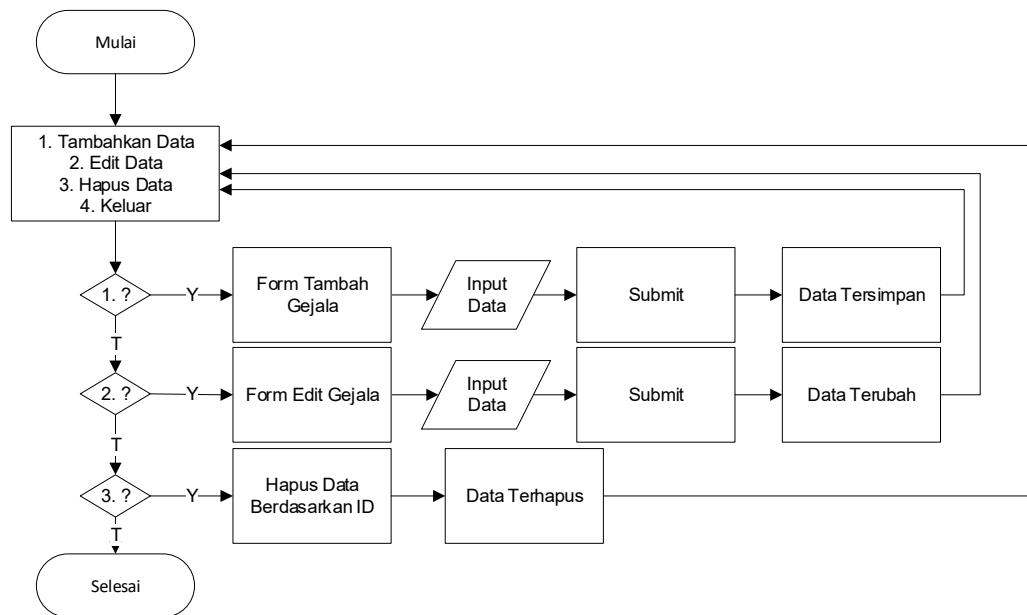
Pada *flowchart form input* penyakit, jika admin menekan tambah data maka sistem akan menampilkan *form* tambah data penyakit dan admin *menginput* data dengan menekan tombol submit data dan sistem akan menyimpan data, jika admin menekan *edit* data maka sistem akan menampilkan *form edit* data penyakit dan admin *menginput* data dengan menekan tombol *update* data dan data akan *terupdate* dan jika menekan hapus maka sistem akan menghapus data berdasarkan id yang dihapus.



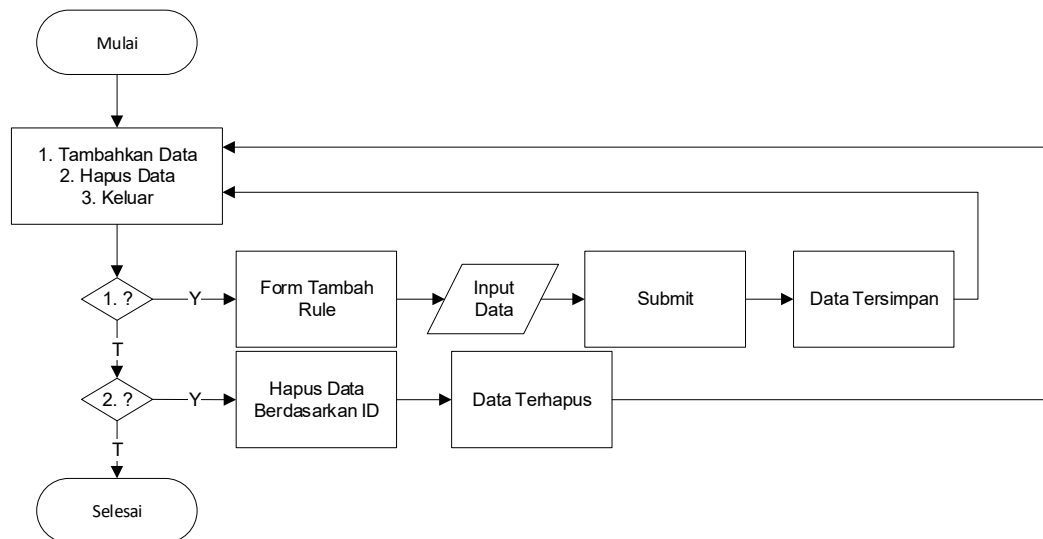
Gambar 4.10 Flowchart Form Input Penyakit

#### 4.6.4 Flowchart Form Input Gejala

Pada *flowchart form input* gejala, jika admin menekan tambah data maka sistem akan menampilkan *form* tambah data gejala dan admin *menginput* data dengan menekan tombol submit data dan sistem akan menyimpan data, jika admin menekan *edit* data maka sistem akan menampilkan *form edit* data gejala dan admin *menginput* data dengan menekan tombol *update* data dan data akan *terupdate* dan jika menekan hapus maka sistem akan menghapus data berdasarkan id yang dihapus.

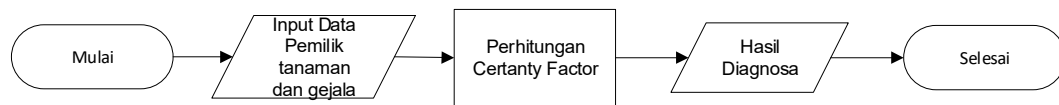
Gambar 4.11 *Flowchart Form Input Gejala*

#### 4.6.5. Flowchart Input Rule

Gambar 4.12 *Flowchart Form Input Rule*

Pada *flowchart form input rule*, jika admin menekan tambah data maka sistem akan menampilkan *form* tambah data rule dan admin menginput data dengan menekan tombol submit data dan sistem akan menyimpan data, dan jika menekan hapus maka sistem akan akan menghapus data berdasarkan id yang dihapus.

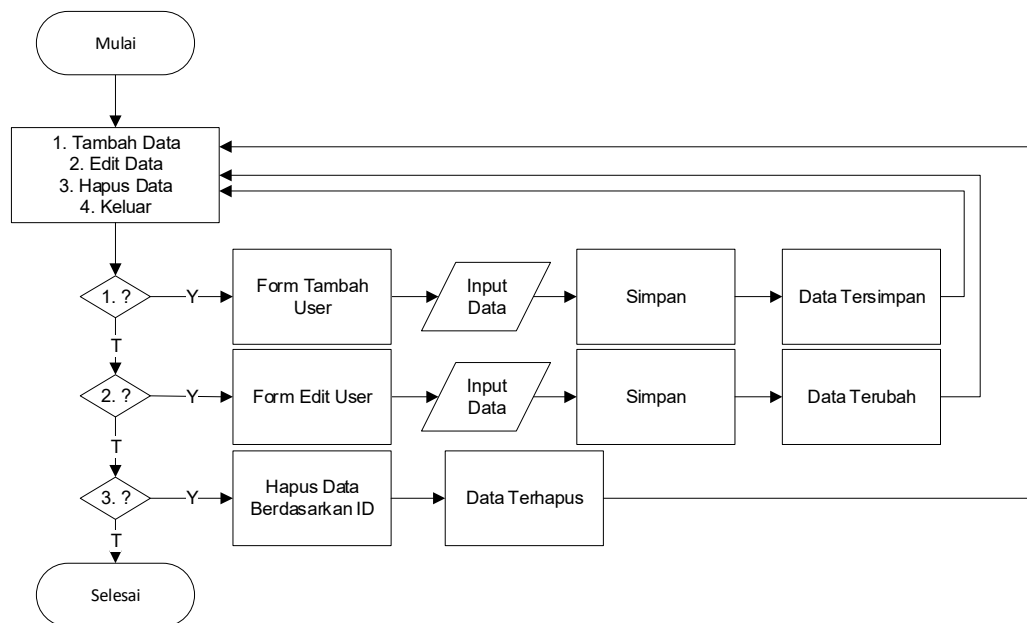
#### 4.6.6. Flowchart Diagnosis



Gambar 4.13 Flowchart Diagnosis

Pada *flowchart* diagnosis admin terlebih dahulu menginput data pemilik tanama dan gejala gejala yang yang dialami tanaman lombok yang ia tanam, kemudian sistem akan melakukan proses perhitungan certanty factor dan menghasilkan diagnosis penyakit tanaman lombok yang dialami pemilik tanaman.

#### 4.6.7. Flowchart User



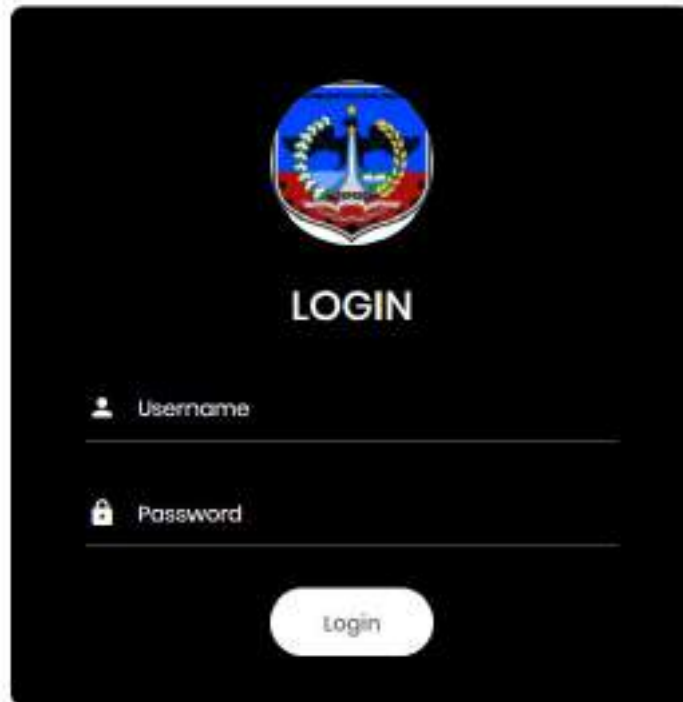
Gambar 4.14 Flowchart Form User

Pada *flowchart form user*, jika admin menekan tambah data maka sistem akan menampilkan *form* tambah data *user* dan admin menginput data dengan menekan tombol simpan data dan sistem akan menyimpan data, jika admin menekan *edit* data maka sistem akan menampilkan *form edit user* dan admin menginput data dengan menekan tombol simpan dan sistem akan menyimpan data dan jika menekan hapus maka sistem akan akan menghapus data berdasarkan id yang dihapus.

## 4.7 Implementasi Sistem

### 4.7.1. Implementasi Sistem Untuk Admin

#### 1. Halaman Login



Gambar 4.15 Halaman Login

Pada halaman login admin perlu menginput *username* dan *password*, setelah itu menekean tombol login dan sitem akan mengecek *username* dan *password* tersebut di dalam *database*. Jika *username* dan *password* tersebut ada dalam *database* maka sistem akan menampilkan halaman utama dan jika tidak ada dalam *database* maka sistem akan menampilkan pesan *username* dan *password* salah

```
$sql = mysql_query("SELECT * from tb_user where user = '$user' and pass = '$pass'") or
die(mysql_error());
$data = mysql_fetch_array($sql);
$cek = mysql_num_rows($sql);
if($cek > 0){
    if ($data['level'] == 'admin') {
        $_SESSION['admin'] = $data['id_user'];
        $_SESSION['time'] = time()+(200*60);
        header("location: index.php");
    } else {
        $_SESSION['kepala'] = $data['id_user'];
        $_SESSION['time'] = time()+(200*60);
        header("location: ../kepala/index.php");
    }
}
```

```

    }else{
        ?><script type="text/javascript">
            alert('Username Atau Password Salah');
            window.history.go(-1);
        </script><?php
    }

```

## 2. Halaman Menu Utama Admin



Gambar 4.16 Halaman Menu Utama Admin

Pada halaman menu utama terdapat enam menu utama yaitu beranda yang jika diklik akan menampilkan halaman beranda, menu *input* yang jika diklik akan menampilkan submenu penyakit, gejala dan rule, menu diagnosis yang jika diklik akan menampilkan halaman diagnosis, menu *output* yang jika diklik akan menampilkan submenu laporan diagnosis, *user* yang jika diklik akan menampilkan halaman *user* dan menu logout yang jika diklik akan akan kembali ke halaman login.

```

if($page == "penyakit"){
    if($aksi == ""){
        include 'penyakit/index.php';
    }else if($aksi == 'edit'){
        include 'penyakit/edit.php';
    }else if($aksi == 'hapus'){
        include 'penyakit/hapus.php';
    }else if($aksi == 'tambah'){
        include 'penyakit/tambah.php';
    }
}
} else if($page == "gejala"){
    if($aksi == ""){
        include 'gejala/index.php';
    }else if($aksi == 'edit'){

```

```

        include 'gejala/edit.php';
    }else if($aksi == 'hapus'){
        include 'gejala/hapus.php';
    }else if($aksi == 'tambah'){
        include 'gejala/tambah.php';
    }
}}else if($page == "rule"){
    if($aksi == ""){
        include 'rule/index.php';
    }else if($aksi == 'edit'){
        include 'rule/edit.php';
    }else if($aksi == 'hapus'){
        include 'rule/hapus.php';
    }else if($aksi == 'tambah'){
        include 'rule/tambah.php';
    }
}

```

### 3. Halaman *Input* Penyakit



Gambar 4.17 Halaman *Input* Penyakit

Pada halaman *input* penyakit, jika admin menekan tambah data maka sistem akan menampilkan *form* tambah data penyakit dan admin *menginput* data dengan menekan tombol submit data dan sistem akan menyimpan data, jika admin menekan *edit* data maka sistem akan menampilkan *form edit* data penyakit dan admin *menginput* data dengan menekan tombol *update* data dan data akan *terupdate* dan jika menekan hapus maka sistem akan akan menghapus data berdasarkan id yang dihapus

```

<?php
$no = 1;
$sql = mysql_query("SELECT * from tb_penyakit");
while ($stampil = mysql_fetch_array($sql)){
?>
<tr>
<td><?php echo $no++ ?></td>
<td><?php echo $stampil['nama_penyakit']; ?></td>
<td align="center">

```

```

<a href="?page=penyakit&aksi=edit&id=?php echo $stampil['id_penyakit']; ?>"
class="btn btn-warning btn-xs">Edit</a>
<a href="?page=penyakit&aksi=hapus&id=?php echo $stampil['id_penyakit']; ?>"
class="btn btn-danger btn-xs">Hapus</a>
</td><?php } ?>

```

#### 4. Halaman *Input* Gejala

No.	Nama Gejala	Aksi
1	Mata pucat	Edit Hapus
2	Dari en, mering, dan kadang banyak kering berwarna coklat seperti indung	Edit Hapus
3	Buah timbul bercaR rusak berwarna hitam dan buah busuk	Edit Hapus
4	Pada batang acuan dan daun terlihat banyak penyakit	Edit Hapus
5	Daun berbentuk busuk berwarna abu-abu tua dan warna coklat di pangkalnya	Edit Hapus
6	Daun menjadi busuk (menguning) berwarna asyutnya	Edit Hapus
7	Daun daun berukunya sekitar 0,25 cm	Edit Hapus
8	Daun menjadi busuk dan busuk di bagian	Edit Hapus
9	Daun busuk menjadi kering dan mengeras	Edit Hapus

Gambar 4.18 Halaman *Input* Gejala

Pada halaman *input* gejala, jika admin menekan tambah data maka sistem akan menampilkan *form* tambah data gejala dan admin *input* data dengan menekan tombol submit data dan sistem akan menyimpan data, jika admin menekan *edit* data maka sistem akan menampilkan *form edit* data gejala dan admin *input* data dengan menekan tombol *update* data dan data akan *terupdate* dan jika menekan hapus maka sistem akan menghapus data berdasarkan id yang dihapus

```

<?php
$no = 1;
$sql = mysql_query("SELECT * from tb_gejala");
while ($stampil = mysql_fetch_array($sql)){
?>
<tr>
<td><?php echo $no++ ?></td>
<td><?php echo $stampil['nama_gejala']; ?></td>
<td align="center">
<a href="?page=gejala&aksi=edit&id=?php echo $stampil['id_gejala']; ?>" class="btn
btn-warning btn-xs">Edit</a>

```



```
<a href="?page=penyakit&aksi=hapus&id=<?php echo $stampil['id_gejala']; ?>"
class="btn btn-danger btn-xs">Hapus</a>
</td><?php } ?>
```

## 5. Halaman *Input Rule*

No.	Nama Penyakit	Gejala	Nilai CF	Aksi
1	Ambeien	01. Maki pasak 02. Sore, nanjing, dan keluar darah dari anus 03. Boleh timbul bercak darah berwarna hitam dan bau busuk 04. Pada bagian anus jika ada darah keluar dari anus	0.80	Hapus
2	Ambeien	01. Maki pasak 02. Boleh timbul bercak darah berwarna hitam dan bau busuk 03. Pada bagian anus jika ada darah keluar dari anus 04. Boleh timbul bercak darah berwarna hitam dan bau busuk	0.80	Hapus
3	Ambeien	01. Maki pasak 02. Boleh timbul bercak darah berwarna hitam dan bau busuk 03. Boleh timbul bercak darah berwarna hitam dan bau busuk 04. Pada bagian anus jika ada darah keluar dari anus	0.80	Hapus
4	Bercak Daun Sirihpohon	01. Bercak daun berwarna hitam 02. Bercak daun berwarna hitam 03. Bercak daun berwarna hitam 04. Bercak daun berwarna hitam	0.80	Hapus

Gambar 4.19 Halaman *Input Rule*

Pada halaman *input rule*, jika admin menekan tambah data maka sistem akan menampilkan *form* tambah data rule dan admin *menginput* data dengan menekan tombol submit data dan sistem akan menyimpan data, jika admin menekan *edit* data maka sistem akan menampilkan *form edit* data rule dan admin *menginput* data dengan menekan tombol *update* data dan data akan *terupdate* dan jika menekan hapus maka sistem akan akan menghapus data berdasarkan id yang dihapus.

```
<?php
$no = 1;
$sql = mysql_query("SELECT * from tb_rule inner join tb_penyakit on tb_rule.id_penyakit
= tb_penyakit.id_penyakit inner join tb_gejala on tb_rule.id_gejala = tb_gejala.id_gejala
order by tb_penyakit.id_penyakit, tb_gejala.id_gejala ");
<td><?php echo $no++ ?></td>
<td><?php echo $stampil['nama_penyakit']; ?></td>
<td><?php echo "G".$stampil['id_gejala']; ?> <?php echo $stampil['nama_gejala']; ?></td>
<td><?php echo $stampil['probabilitas']; ?></td>
<td align="center">
<a href="?page=rule&aksi=edit&id=<?php echo $stampil['id_rule']; ?>" class="btn btn-
warning btn-xs">Edit</a>
```

```
<a href="?page=rule&aksi=hapus&id=?php echo $stampil['id_rule']; ?>" class="btn btn-
danger btn-xs">Hapus</a>
</td>
```

## 6. Halaman Diagnosis

Detail Diagnosa		
<b>Gejala Yang Dipilih</b>		
1. Mati pucuk		
2. Bercak berbentuk bulat berwarna abu-abu tua dan warna coklat di pinggirannya		
3. Bercak daun berukuran sekitar 0,25 cm		
4. Seluruh daun menjadi layu		
5. Serangan pada akar mengakibatkan tanaman layu, mengering, dan mati		
<b>Nilai Tiap Penyakit</b>		
No.	Nama Penyakit	Nilai
1	Antraknosa	0.5776
2	Bercak Daun Serkospora	0.68584
3	Busuk daun Fitofthora	0.5392

Jadi Penyakit Tanaman Cabai Yang ditanam Oleh Bau Tarring Terdiagnosa Penyakit **Bercak Daun Serkospora** Dengan Nilai **0.68584** dan Persentase **68.58 %**

Gambar 4.20 Halaman Diagnosis

Pada halaman diagnosis, admin terlebih dahulu menginput data pemilik tanama dan gejala gejala yang yang dialami tanaman lombok yang ia tanam, kemudian sistem akan melakukan proses perhitungan certainty factor dan menghasilkan diagnosis penyakit tanaman lombok yang dialami pemilik tanaman.

```
<?php
$checkbox    =$_POST['checkbox'];
$jml_dipilih =count($checkbox);
if ($jml_dipilih > 7) {
?>
<script type="text/javascript">
alert('Hanya Tujuh Gejala Yang Dapat Dipilih');
window.history.go(-1);
</script>
<?php
} else {
for($x=0;$x<$jml_dipilih;$x++){
$xx = $x+1;
$ga = mysql_query("SELECT * FROM tb_gejala where id_gejala = '$checkbox[$x]' ");
$ge = mysql_fetch_array($ga);
if ($xx == 1) {
$p1 = $ge['id_gejala'];
} else if ($xx == 2) {
$p2 = $ge['id_gejala'];
} else if ($xx == 3) {
$p3 = $ge['id_gejala'];
} else if ($xx == 4) {
```

```

$P4 = $ge['id_gejala'];
} else if ($xx == 5) {
$P5 = $ge['id_gejala'];
} else if ($xx == 6) {
$P6 = $ge['id_gejala'];
} else if ($xx == 7) {
$P7 = $ge['id_gejala'];
}
echo $xx.' '.$ge['nama_gejala'];echo "<br>";
}
?>

```

## 7. Halaman User



Gambar 4.21 Halaman User

Pada halaman *user*, jika admin menekan tambah data maka sistem akan menampilkan *form* tambah data *user* dan admin menginput data dengan menekan tombol simpan data dan sistem akan menyimpan data, jika admin menekan *edit* data maka sistem akan menampilkan *form edit user* dan admin menginput data dengan menekan tombol simpan dan sistem akan menyimpan data dan jika menekan hapus maka sistem akan akan menghapus data berdasarkan id yang dihapus.

```

<?php
$no = 1;
$sql = mysql_query("select * from tb_user");
while ($stampil = mysql_fetch_array($sql)){
?>
<tr>
<td><?php echo $no++; ?></td>
<td><?php echo $stampil['user']; ?></td>
<td><?php echo md5($stampil['pass']); ?></td>
<td><?php echo $stampil['level']; ?></td>
<td align="center">
<a href="?page=user&aksi=edit&id=<?php echo $stampil['id_user']; ?>" class="btn btn-
warning">Edit</a>

```

```

<a href="?page=user&aksi=hapus&id=?php echo $stampil['id_user']; ?>" class="btn btn-
danger">Hapus</a>
</td>
</tr>
<?php }?>

```

## 8. Laporan

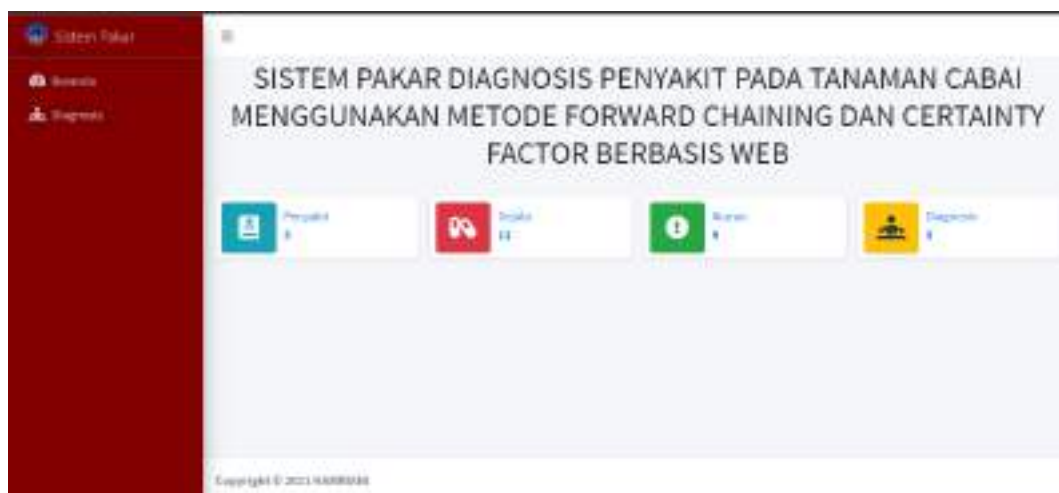
No.	Nama Pasien	Diagnosa	Nilai	Tanggal Diagnosa
1	Wahid Supriyanto	embun	98,75 %	04-05-2022
2	Irni, Nedy Prasana	Kemud Dasi Endogamia	93,32 %	04-05-2022
3	Kadriyati Dewi	Bulat Daun Pithium	95,37 %	04-05-2022
4	Bay Taring	Bulat Daun Endogamia	95,38 %	04-05-2022

Gambar 4.22 Laporan Hasil Diagnosis

Gambar 4.22 laporan hasil diagnosis merupakan sebuah *output* dari Sistem Pakar Diagnosis penyakit Pada Tanaman Cabai Dengan Menggunakan Metode *Forward Chaining* dan *Certainty Factor* Berbasis WEB yang berisikan data data diagnosis yang telah dilakukan oleh admin.

### 4.7.2. Implementasi Sistem Untuk Pemilik Tanaman

#### 1. Halaman Utama



Gambar 4.23 Halaman Menu Utama Pemilik Tanaman

Pada halaman menu utama terdapat dua menu utama yaitu beranda yang jika diklik akan menampilkan halaman beranda dan menu diagnosis yang jika diklik akan menampilkan halaman diagnosis.

```
<?php
    include '../koneksi.php';
    $page = @$_GET['page'];
    $aksi = @$_GET['aksi'];
    if($page == "diagnosa"){
        if($aksi == ""){
            include 'diagnosa/index.php';
        } else if($aksi == 'detail'){
            include 'diagnosa/detail.php';
        }
    } else {
        include 'dashboard.php';
    }
?>
```

## 2. Halaman Diagnosis

Detail Diagnosa

---

**Gejala Yang Dipilih**

1. Mati pucuk
2. Bercak berbentuk bulat berwarna abu-abu tua dan warna coklat di pinggirannya
3. Bercak daun berukuran sekitar 0,25 cm
4. Seluruh daun menjadi layu
5. Serangan pada akar mengakibatkan tanaman layu, mengering, dan mati

**Nilai Tiap Penyakit**

No.	Nama Penyakit	Nilai
1	Antraknosa	0.5776
2	Bercak Daun Serkospora	0.68584
3	Busuk daun Fitotora	0.5392

Jadi Penyakit Tanaman Cabai Yang ditanam Oleh Bau Tarring Terdiagnosa Penyakit **Bercak Daun Serkospora** Dengan Nilai **0.68584** dan Persentase **68.58 %**

Gambar 4.24 Halaman Diagnosis

Pada halaman diagnosis, admin terlebih dahulu menginput data pemilik tanama dan gejala gejala yang yang dialami tanaman lombok yang ia tanam, kemudian sistem akan melakukan proses perhitungan certainty factor dan menghasilkan diagnosis penyakit tanaman lombok yang dialami pemilik tanaman.

```
<?php
$checkbox    =$_POST['checkbox'];
$jml_dipilih =count($checkbox);
if ($jml_dipilih > 7) {
?>
<script type="text/javascript">
alert('Hanya Tujuh Gejala Yang Dapat Dipilih');
```

```

window.history.go(-1);
</script>
<?php
} else {
for($x=0;$x<$jml_dipilih;$x++){
$xx = $x+1;
$ga = mysql_query("SELECT * FROM tb_gejala where id_gejala = '$checkbox[$x]' ");
$ge = mysql_fetch_array($ga);
if ($xx == 1) {
$p1 = $ge['id_gejala'];
} else if ($xx == 2) {
$p2 = $ge['id_gejala'];
} else if ($xx == 3) {
$p3 = $ge['id_gejala'];
} else if ($xx == 4) {
$p4 = $ge['id_gejala'];
} else if ($xx == 5) {
$p5 = $ge['id_gejala'];
} else if ($xx == 6) {
$p6 = $ge['id_gejala'];
} else if ($xx == 7) {
$p7 = $ge['id_gejala'];
}
echo $xx.'. '.$ge['nama_gejala'];echo "<br>";
}
?>

```

## 4.8 Pengujian Sistem

Berikut ini adalah kasus untuk menguji perangkat lunak yang dibangun menggunakan metode *Black Box* berdasarkan pada Tabel Rencana Pengujian.

### 4.8.1 Pengujian Sistem Pada Halaman Login Admin

Tabel 4.9 Pengujian Sistem Pada Halaman Login

No	Skenario Pengujian	Hasil Yang diharapkan	Pengamatan	Keterangan
1	Mengosongkan salah satu textbox dan tekan tombol login	Sistem akan menampilkan pesan “please fill out this fields”	Sistem menampilkan pesan “ <i>please fill out this fields</i> ”	Sesuai
2	Menginput <i>username</i> dan <i>password</i> yang tidak ada dalam <i>database</i>	Sistem akan menampilkan pesan “Maaf informasi login tidak dikenali <i>username</i> dan <i>password</i> salah”	Sistem menampilkan pesan “Maaf informasi login tidak dikenali <i>username</i> dan <i>password</i> salah”	Sesuai

3	Menginput <i>username</i> dan <i>password</i> yang sesuai dalam <i>database</i>	Sistema akan menampilkan halaman utama admin	Sistema menampilkan halaman utama admin	Sesuai
---	---	--	---	--------

Pada tabel pengujian sistem pada halaman login admin, peneliti melakukan skenario pengujian dengan mengosongkan salah satu textbox dan tekan tombol login, dan dari pemangamatan peneliti sistem telah berjalan sesuai (*valid*) dengan yang diharapkan. Peneliti juga melakukan skenario pengujian dengan menginput *username* dan *password* yang sesuai dalam *database*, dan dari pemangamatan peneliti sistem telah berjalan sesuai (*valid*) dengan yang diharapkan.

#### 4.8.2 Pengujian Sistem Pada Halaman Utama Admin

Tabel 4.10 Pengujian Sistem Pada Halaman Utama Admin

No	Skenario Pengujian	Hasil Yang diharapkan	Pengamatan	Keterangan
1	Klik menu dashboard	Sistem akan menampilkan halaman dashboard	Sistem menampilkan halaman dashboard	Sesuai
2	Klik menu <i>Input</i>	Sitem akan menampilkan submenu <i>Input</i>	Sitem menampilkan submenu <i>Input</i>	Sesuai
3	Klik submenu penyakit	Sistem akan menampilkan <i>form</i> penyakit	Sistem menampilkan <i>form</i> penyakit	Sesuai
4	Klik submenu gejala	Sistem akan menampilkan <i>form</i> gejala	Sistem menampilkan <i>form</i> gejala	Sesuai
5	Klik submenu rule	Sistem akan menampilkan <i>form</i> rule	Sistem menampilkan <i>form</i> rule	Sesuai
6	Klik menu diagnosis	Sitem akan menampilkan halaman diagnosis	Sitem menampilkan halaman diagnosis	Sesuai
7	Klik menu Laporan	Sitem akan menampilkan submenu laporan	Sitem menampilkan submenu laporan	Sesuai
8	Klik menu <i>user</i>	Sistem akan menampilkan <i>form user</i>	Sistem menampilkan <i>form</i> rekap <i>user</i>	Sesuai

9	Klik Menu <i>logout</i>	Sistem akan kembali ke menu <i>Login</i>	Sistem akan kembali ke menu <i>Login</i>	Sesuai
---	-------------------------	--	--	--------

Pada tabel pengujian sistem pada halaman menu utama admin, peneliti melakukan skenario pengujian dengan klik menu dashboard, dan dari pengamatan peneliti sistem telah berjalan sesuai (*valid*) dengan yang diharapkan. Peneliti juga melakukan skenario pengujian dengan klik menu logout, dan dari pengamatan peneliti sistem telah berjalan sesuai (*valid*) dengan yang diharapkan.

#### 4.8.3 Pengujian Sistem Pada Halaman Data Penyakit

Tabel 4.11 Pengujian Sistem Pada Halaman Data Penyakit

No	Skenario Pengujian	Hasil Yang diharapkan	Pengamatan	Keterangan
1	Klik tombol tambah	Sistem akan menampilkan halaman tambah data penyakit	Sistem menampilkan halaman tambah data penyakit	Sesuai
2	Klik tombol <i>edit</i>	Sistem akan menampilkan halaman <i>edit</i> data penyakit	Sistem menampilkan halaman <i>edit</i> data penyakit	Sesuai
3	Klik tombol hapus	Sistem akan menghapus data penyakit sesuai dengan ID	Sistem menghapus data penyakit sesuai dengan ID	Sesuai

Pada tabel pengujian sistem pada halaman data penyakit, peneliti melakukan skenario pengujian dengan klik tombol tambah, dan dari pengamatan peneliti sistem telah berjalan sesuai (*valid*) dengan yang diharapkan. Peneliti juga melakukan skenario pengujian dengan klik tombol hapus, dan dari pengamatan peneliti sistem telah berjalan sesuai (*valid*) dengan yang diharapkan.

#### 4.8.4 Pengujian Sistem Pada Halaman Data Gejala

Tabel 4.12 Pengujian Sistem Pada Halaman Data Gejala

No	Skenario Pengujian	Hasil Yang diharapkan	Pengamatan	Keterangan
1	Klik tombol tambah	Sistem akan menampilkan halaman tambah data gejala	Sistem menampilkan halaman tambah data gejala	Sesuai



2	Klik tombol <i>edit</i>	Sistem akan menampilkan halaman <i>edit</i> data gejala	Sistem menampilkan halaman <i>edit</i> data gejala	Sesuai
3	Klik tombol hapus	Sistem akan menghapus data gejala sesuai dengan ID	Sistem menghapus data gejala sesuai dengan ID	Sesuai

Pada tabel pengujian sistem pada halaman data gejala, peneliti melakukan skenario pengujian dengan klik tombol tambah, dan dari pengamatan peneliti sistem telah berjalan sesuai (*valid*) dengan yang diharapkan. Peneliti juga melakukan skenario pengujian dengan klik tombol hapus, dan dari pengamatan peneliti sistem telah berjalan sesuai (*valid*) dengan yang diharapkan.

#### 4.8.5 Pengujian Sistem Pada Halaman Data Rule

Tabel 4.13 Pengujian Sistem Pada Halaman Data Rule

No	Skenario Pengujian	Hasil Yang diharapkan	Pengamatan	Keterangan
1	Klik tombol tambah	Sistem akan menampilkan halaman tambah data rule	Sistem menampilkan halaman tambah data rule	Sesuai
2	Klik tombol hapus	Sistem akan menghapus data gejala sesuai dengan ID	Sistem menghapus data gejala sesuai dengan ID	Sesuai

Pada tabel pengujian sistem pada halaman data rule, peneliti melakukan skenario pengujian dengan klik tombol tambah, dan dari pengamatan peneliti sistem telah berjalan sesuai (*valid*) dengan yang diharapkan. Peneliti juga melakukan skenario pengujian dengan klik tombol hapus, dan dari pengamatan peneliti sistem telah berjalan sesuai (*valid*) dengan yang diharapkan.

#### 4.9 Pengujian Akurasi Sistem

Pengujian akurasi sistem dilakukan untuk mengetahui keakuratan sebuah sistem. Pengujian akuratan sistem ini dilakukan menggunakan *matriks counfusion*. Berdasarkan 10 data dari hasil diagnosis menggunakan metode *forward chaining* dan *certainty factor* dengan hasil diagnosis yang dilakukan oleh pakar, terdapat 8

data yang sesuai dan 2 data hasil menggunakan metode *forward chaining* dan *certainty factor* yang tidak sesuai dengan hasil pakar.

$$x = \frac{\text{jumlah data yang sesuai}}{\text{jumlah data}} \times 100\%$$

$$x = \frac{8}{10} \times 100\% = 80 \%$$

Berikut ini adalah hasil perhitungan ketidaksesuaian (*error*)

$$x = \frac{\text{jumlah data yang error}}{\text{jumlah data}} \times 100\%$$

$$x = \frac{2}{10} \times 100\% = 20 \%$$

4.14 Tabel Hasil Data Pengujian Sistem Yang Sesuai dan Tidak Sesuai

Nama Pemilik	Hasil Perhitungan Sistem	Hasil Pakar	Keterangan
Ines Aulia	Antraknosa	Antraknosa	Sesuai
Wulandari	Antraknosa	Antraknosa	Sesuai
Noviyanti	Antraknosa	Antraknosa	Sesuai
Risdayani	Antraknosa	Bercak Daun Serkospora	Tidak Sesuai
Ayu Arnianti	Bercak Daun Serkospora	Bercak Daun Serkospora	Sesuai
Muh. Ismail	Bercak Daun Serkospora	Bercak Daun Serkospora	Sesuai
Muh. Rizky	Bercak Daun Serkospora	Bercak Daun Serkospora	Sesuai
Fachrul	Busuk daun Fitofthora	Busuk daun Fitofthora	Sesuai
Rahmad Rivaldi	Busuk daun Fitofthora	Busuk daun Fitofthora	Sesuai
Wahyu Agustam	Busuk daun Fitofthora	Busuk daun Fitofthora	Tidak Sesuai

Pada tabel 4.14 pengujian sistem yang sesuai dan tidak sesuai, peneliti melakukan pengujian untuk mengetahui keakuratan sistem pakar metode *forward chaining* dan *certainty factor* dengan hasil diagnosis yang dilakukan oleh pakar, dan dari pengujian yang dilakukan oleh peneliti bahwa 80% data telah sesuai dengan hasil diagnosis pakar.

## BAB V

### PENUTUP

#### 5.1. Kesimpulan

Berdasarkan pembahasan pada bab-bab sebelumnya, maka peneliti dapat menyimpulkan beberapa hal dengan menggunakan Sistem Pakar Diagnosis penyakit Pada Tanaman Cabai Dengan Menggunakan Metode *Forward Chaining* dan *Certainty Factor* Berbasis WEB sebagai berikut :

1. Dengan menggunakan sistem pakar dengan metode *Forward Chaining* dan *Certainty Factor*, dapat mendiagnosis penyakit saluran pernapasan dengan tingkat akurasi sebesar 80 %.
2. Berdasarkan hasil pengujian *black box* dapat disimpulkan bahwa Sistem Pakar Diagnosis penyakit Pada Tanaman Cabai Dengan Menggunakan Metode *Forward Chaining* dan *Certainty Factor* Berbasis WEB terbebas dari kesalahan program..

#### 5.2. Saran

Berdasarkan hasil penelitian dan kesimpulan Sistem Pakar Diagnosis penyakit Pada Tanaman Cabai Dengan Menggunakan Metode *Forward Chaining* dan *Certainty Factor* Berbasis WEB ini, maka terdapat beberapa saran diantaranya:

1. Untuk penelitian lebih lanjut bagi mahasiswa yang akan melakukan penelitian diharapkan kedepannya agar bisa dikembangkan lagi menjadi berbasis *android*.
2. Diharap sistem ini dapat dikembangkan lagi agar tidak terpaku hanya pada satu daerah saja.

## DAFTAR PUSTAKA

- Frederick,L Jones 2008. Sistem informasi akuntansi buku I, Jakarta :salemba empat
- Hartati,S dkk,2008.sistem pakar dan pengembanganya,graham ilmu,Yogyakarta
- Hirin,2011. Cepat mahir pemrograman *web* dengan *php* dan *Mysql* (level dasar sampai mahir)
- Jogiyanto 2005.*Analisis & design*.Yogyakarta : Andi Offset
- Jogiyanto HM. 2005.PengenalanKomputer. Yogyakarta : Andi Offset
- Johnsond.d.w. 2004.*cooperation in the classroom* bandung; nalfabeta.
- Kadir Abdul. 2003. Pengenalan Sistem Informasi. Yogyakarta :Penerbit Andi. Agung.
- Kadir,2008.Tuntunan praktis belajar *database* menggunakan *mysql*,c.v.andi offset.yogyakarta.
- Kusrini,2008.Aplikasi sistem pakar menentukan faktor kepastian pengguna dengan metode *kuantifikasi*, i,andi :Yogyakarta
- Kusumadewi,s.2003.*Arificial teknik* dan aplikasinya, Yogyakarta: graha ilmu
- Mahmudi, Rokhman, Prasetyo 2016. Rancang bangun sistem pakar untuk mendiagnosis tanaman cabai menggunakan metode *bayes*. (Jurnal Rekayasa Dan Manajemen Sistem Informasi Vol.2, No 2, Agustus 2016 e-ISSN 2502-8995 ISSN 2460- 8181 85)
- Natalius,2011. Metode *Forward Chaining* dan *Certainty Factor Classifier* Dan Penggunaanya Pada Klasifikasi Dokumen, Makala II 2092 Probabilitas Dan Statistika.Pp.1
- Oktaviana,arifin, dan surya 2012, Sistem pakar diagnose penyakit ginjal menggunakan metode *hill climbing*.jurnal teknik *informatika*,vol.1
- Pressman 2010.*Rekayasa perangkat lunak edisin 7*.yogyakarta: andi offset
- Raharjo.budi.heriyanto.rk.2010.Modul Pemrograman *Web*,Modula Bandung.
- Rahayu,s.2013 Sistem pakar untuk mendiagnosa penyakit gagal ginjal dengan menggunakan metode *bayes*.pelita *informatika* budi darma
- Sihotang 2018.Sistem pakar untuk mendiagnosa penyakit pada tanaman jagung dengan metode bayes.(*Journal Of Informatic Pelita Nusantara* Volume 3 No

1 Maret 2018 e-ISSN 2541-3724 17 *Journal Of Informatic Pelita Nusantara*)

Sumpala, A. T. dan Sutoyo 2018 Sistem Pakar Untuk Mendiagnosa Hama dan Penyakit Tanaman Kakao Menggunakan Metode Forward Chaining dan Certainty Factor Seminar Nasional Teknologi Informasi, Komunikasi dan Industri (SNTIKI-10) Fakultas Sains dan Teknologi, UIN Sultan Syarif Kasim Riau.

Sutabri, Tata. 2012. *Konsep Sistem Informasi*. Edisi Pertama. Yogyakarta : Andi. Sutojo, T, et al, 2011. *Kecerdasan buatan*, andi : Yogyakarta  
Sommerville, I., 2011, *Software Engineering*, Boston, Massachusetts: Pearson Education

Wahana computer. 2010. Panduan belajar *MYSQL databaseserver*. media Jakarta selatan.

Wafal, Rahayu 2014. Sistem pakar untuk mendiagnosis hama dan penyakit padatanaman padi dengan metode *Bayesian*.

Wijaya1, Hidayat, Suprpto, 2018. Diagnosis Penyakit Cabai Dengan Menggunakan Metode *Forward Chaining – Dempster-Shafer*( Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer *e-ISSN: 2548- 964X Vol. 2, No. 12*, Desember 2018, *hlm. 7202-7208* <http://j-ptiik.ub.ac.id>)

Yakub 2012 Pengantar sistem informasi .yogyakarta: graham ilmu.

Yakub, 2008. Sistem *basisdata* tutorial konseptual yogyakarta: penerbit graha ilmu