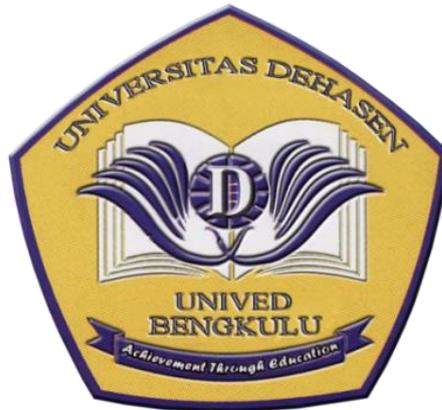


**SISTEM PAKAR PENYAKIT PADA TANAMAN SINGKONG  
MENGUNAKAN METODE DEMPSTER SHAFER  
BERBASIS WEB**

**SKRIPSI**



Oleh :

**WIDYA AYU SAFITRI**  
**NPM : 18010108**

**PROGRAM STUDI INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS DEHASEN  
BENGKULU**

**2022**

**SISTEM PAKAR PENYAKIT PADA TANAMAN SINGKONG  
MENGUNAKAN METODE DEMPSTER SHAFER  
BERBASIS WEB**

**SKRIPSI**

Diajukan Sebagai Salah Satu Syarat untuk Memperoleh Gelar Sarjana  
Komputer

Oleh :

**WIDYA AYU SAFITRI**  
**NPM : 18010108**

**PROGRAM STUDI INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS DEHASEN  
BENGKULU**

**2022**

**SISTEM PAKAR PENYAKIT PADA TANAMAN SINGKONG  
MENGUNAKAN METODE DEMPSTER SHAFER  
BERBASIS WEB**

**SKRIPSI**

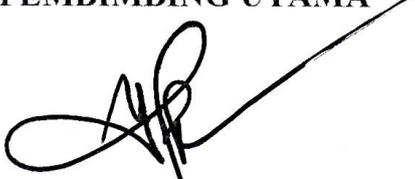
**OLEH :**

**WIDYA AYU SAFITRI**

**NPM : 18010108**

**Disetujui Oleh :**

**PEMBIMBING UTAMA**

  
**Yupianti, S.Kom, M.Kom**  
**NIDN : 02.030486.02**

**PEMBIMBING PENDAMPING**

  
**Rizka Tri Alinse S.Kom M.Kom**  
**NIDN : 02.250992.01**

**Mengetahui**  
**Ketua Program Studi Informatika**

  
**Liza Yulianti, S.Kom, M.Kom**  
**NIDN : 02.160772.01**

**SISTEM PAKAR PENYAKIT PADA TANAMAN SINGKONG  
MENGUNAKAN METODE DEMPSTER SHAFER  
BERBASIS WEB**

**SKRIPSI**

Disusun Oleh :

**WIDYA AYU SAFITRI**

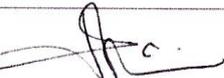
**NPM : 18010108**

Telah dipertahankan didepan Tim Penguji Fakultas Ilmu Komputer Pada:

Hari : Senin

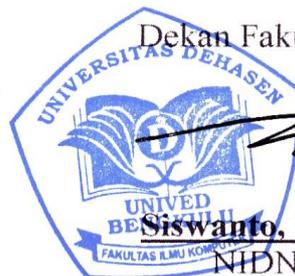
Tanggal : 16 Januari 2023

Skripsi ini telah diperiksa dan disetujui oleh TIM Penguji sbb:

Penguji	Nama	NIDN	Tanda Tangan
Ketua	Yupianti, S.Kom., M.Kom	02.030486.02	
Sekretaris	Rizka Tri Alinse, S.Kom.,M.Kom	02.250992.01	
Anggota	Dra Maryaningsih, M.kom	00.200569.01	
Anggota	Ilayati Beti, S.Kom.,M.Kom	02.240488.03	

**Mengetahui :**

Dekan Fakultas Ilmu Komputer,



  
**Siswanto, SE, S.Kom, M.Kom**

NIDN : 02.240363.01

## RIWAYAT HIDUP



Penulis bernama Widya Ayu Safitri dilahirkan di Lubuk Unen pada tanggal 05 juni 2000, anak pertama dari dua bersaudara. Ayah bernama Sutan dan ibu bernama Nunung. Menyelesaikan pendidikan Tingkat Sekolah Dasar (SD) pada SDN 08 Bengkulu Tengah pada tahun 2006 dan selesai pada tahun 2012, dan pada tahun yang sama penulis melanjutkan Sekolah Tingkat Pertama (SMP) pada SMPN 01 Bengkulu Tengah dan selesai pada tahun 2015, dan pada tahun yang sama penulis melanjutkan Pendidikan Sekolah Menengah Atas (SMA) Pada SMAN 01 Bengkulu Tengah penulis mengambil jurusan IPS dan selesai pada tahun 2018. Pada tahun 2018 penulis melanjutkan ke tingkat Perguruan tinggi yaitu pada Universitas Dehasen (UNIVED) Bengkulu dengan mengambil Prodi Informatika pada Fakultas Ilmu Komputer, untuk jenjang Strata Satu (S-1).

## MOTO DAN PERSEMBAHAN

### Motto :

*“Sesungguhnya bersama kesulitan ada kemudahan. Maka apabila engkau telah selesai (dari suatu urusan), tetaplah bekerja keras untuk (urusan yang lain). Dan hanya kepada Tuhanmulah engkau berharap”*

*(Q.S. Al-Insyirah:6-8)*

*“Beramallah Untuk Kehidupan Duniamu Seakan-akan Engkau Hidup Selamanya, dan Beramallah Untuk Kehidupan Akhiratmu Seakan-akan Engkau Akan Mati hari Ini.”*

### Kupersembahkan Untuk :

*Dengan penuh rasa syukur yang terdalam, kupersembahkan karya kecil ini kepadamu wahai Zat yang Maha Agung, dan Maha Penyayang, atas takdirmu telah kau jadikan aku manusia yang senantiasa berpikir, berilmu, beriman dan bersabar dalam menjalani kehidupan ini. Semoga keberhasilan ini menjadi satu langkah awal bagiku untuk meraih cita-cita besarku.*

*Kupersembahkan sebuah karya kecil ini untuk Ayahanda (Maulana) dan Ibundaku (Latipah) tercinta, yang tiada pernah hentinya selama ini memberiku semangat, doa, dorongan, nasehat dan kasih sayang serta pengorbanan yang tak tergantikan hingga aku selalu kuat menjalani setiap rintangan yang ada didepanku. Untuk ke dua kakak ku, satu adikku dan untuk seluruh keluargaku tercinta.*

*Untuk Pembimbing Ibu Dra. Asnawati, M. Kom, dan Ibu Lena Elfianty, M.Kom yang selalu sabar membimbing dan banyak memberikan masukan yang terbaik,*

*Untuk Para Murabbi yang telah menjaga, membina, memberikan ilmu, motivasi dan nasehatnya*

*Untuk semua hal berharga, teman, sahabat dan setiap orang yang memberikan pengalaman serta kenangan penting yang mendewasakan yang tak akan terulang lagi*

*The last but not least i wanna thank me for being brave as i now and never quitting still life and grow up nana*

## PERNYATAAN KEASLIAN SKRIPSI

Yang bertanda tangan di bawah ini :

Nama : Widya Ayu Safitri

NPM : 18010108

Program Studi : Informatika

Menyatakan dengan sesungguhnya bahwa :

1. Selama melakukan penelitian dan pembuatan skripsi ini saya tidak melakukan pelanggaran etika akademik dalam bentuk apapun atau pelanggaran lain yang bertentangan dengan etika akademik.
2. Skripsi yang saya buat merupakan karya ilmiah saya sebagai penulis, bukan jiplakan atau karya orang lain.
3. Apapun dikemudian hari ditemukan bukti yang menyakinkan bahwa dalam proses pembuatan skripsi ini terdapat pelanggaran etika akademik atau skripsi ini hasil jiplakan atau skripsi hasil karya orang lain, maka saya bersedia menerima sanksi akademik yang ditetapkan oleh Universitas Dehasen Bengkulu.

Demikian Pernyataan ini saya buat dengan sebenarnya untuk dipergunakan bilamana perlu.

Bengkulu, November 2022

Yang Menyatakan,



**Widya Ayu Safitri**

NPM.18010108

## ABSTRAK

### SISTEM PAKAR PENYAKIT PADA TANAMAN SINGKONG MENGUNAKAN METODE DEMPSTER SHAFER BERBASIS WEB

Oleh:

Widya Ayu Safitri<sup>1</sup>

Yupianti, S.Kom., M.Kom<sup>2</sup>

Rizkah Tri Alinse, S.Kom., M.Kom<sup>2</sup>

Penyakit pada tanaman singkong menjadi hambatan bagi para petani. Hal ini juga mengakibatkan Dinas Tanaman Pangan dan Holtikultura Provinsi Bengkulu mengalami kesulitan dalam menangani hama dan penyakit pada tanaman singkong. Salah satu dari penyakit tanaman singkong adalah mati pucuk atau *antraknose* yang dapat disebabkan oleh hama. Beberapa petani singkong tidak mengetahui gejala-gejala penyakit pada tanaman singkong, sehingga para petani tidak memiliki wawasan dan persiapan dalam menghadapi hal tersebut. Penelitian ini bertujuan merancang sistem pakar yang dapat digunakan untuk mengidentifikasi penyakit pada tanaman singkong serta solusi penanganannya secara umum. Dalam metode Dempster Shafer, sistem mengidentifikasi gejala dan nilai *belief* yang telah ditentukan oleh pakar, sehingga menghasilkan nilai *plausibility*. Pada akhirnya menghasilkan nilai persentase pada semua penyakit yang terdapat dalam sistem dan penyakit dengan nilai persentase tertinggi merupakan penyakit yang paling sesuai dengan gejala yang telah dipilih. Kesimpulan dari penelitian ini yaitu penerapan algoritma Dempster Shafer dalam aplikasi ini dapat digunakan untuk menentukan gambaran penyakit tanaman singkong.

**Kata Kunci :** Sistem Pakar, *Dempster Shafer*, Singkong, *Antraknose*,

Keterangan :

1: Peneliti

2: Dosen Pembimbing

**ABSTRACT**

## **AN EXPERT SYSTEM FOR DISEASES IN CASSAVA PLANT USING WEB-BASED DEMPSTER SHAFER METHOD**

**By:**  
**Widya Ayu Safitri<sup>1</sup>**  
**Yupianti<sup>2</sup>**  
**Rizkah Tri Alinse<sup>2</sup>**

*The diseases in cassava plants are an obstacle for farmers. This also resulted in Food Crops and Horticulture Department of Bengkulu Province having difficulties in dealing with pests and diseases in cassava plants. One of the cassava plant diseases is shoot death or anthracnose which can be caused by pests. Some cassava farmers do not know the symptoms of disease in cassava plants, so the farmers do not have the insight and preparation to deal with this. This study aims to design an expert system that can be used to identify diseases in cassava plants and general handling solutions. In the Dempster Shafer method, the system identifies symptoms and belief values that have been determined by experts, resulting in a plausibility value. In the end it produces percentage values for all diseases contained in the system and the disease with the highest percentage value is the disease that best fits the symptoms that have been selected. The conclusion of this study is that the application of the Dempster Shafer algorithm can be used to determine the description of cassava plant diseases.*

*Keywords : Expert System, Dempster Shafer, Cassava, Anthracnose.*

*Information :*

*1: Student*

*2: Supervisors*

## KATA PENGANTAR

Puji Syukur saya panjatkan kehadirat Allah SWT yang telah memberikan rahmat dan karunianya sehingga skripsi yang berjudul “**Sistem Pakar Penyakit Pada Tanaman Singkong Menggunakan Metode Dempster Shafer Berbasis WEB**” dapat diselesaikan dalam waktu yang telah ditetapkan.

Pada kesempatan ini penulis ingin menyampaikan ucapan terima kasih kepada pihak-pihak yang telah banyak membantu penulis selama penyusunan skripsi ini kepada :

1. Bapak Prof. Dr. Husain, SE., M.Si., Ak., CA., CRP Selaku Rektor Universitas Dehasen Bengkulu
2. Bapak Siswanto, SE., S.Kom, M.Kom Selaku Dekan Fakultas Ilmu Komputer Universitas Dehasen Bengkulu
3. Ibu Liza Yulianti, S.Kom, M.Kom. Selaku Ketua Prodi Informatika Fakultas Ilmu Komputer yang telah memberikan pembinaan sehingga dalam penulisan ini bisa berjalan lancar.
4. Ibu Yupianti, S.Kom, M.Kom Selaku pembimbing 1 yang penuh dengan kesabaran, jiwa mendidiknya dalam memberikan masukan.
5. Ibu Rizka Tri Alinse, S.Kom, M.Kom Selaku Pembimbing 2 yang telah memberikan saran dan nasihat dan kesabaran dalam memberikan pembinaan.
6. Pimpinan beserta seluruh pegawai Dinas Tanaman Pangan dan Holtikultura Provinsi Bengkulu yang telah membantu dalam menyelesaikan penelitian ini.
7. Ibu Dan Bapak kedua orang tua penulis yang selalu berjuang, berusaha keras baik dari segi materi tenaga dan fasilitas selama saya belajar dan tak lupa

henti-hentinya memberikan do'a dan dukungan kepada penulis sehingga dapat menyelesaikan skripsi ini tepat waktu.

8. Seluruh rekan-rekan, teman dan sahabat yang telah membantu dalam proses penyelesaian ini.
9. Pihak-pihak lain yang tidak dapat penulis sebutkan namanya satu persatu, yang telah ikut membantu dalam menyelesaikan ini.

Penulis menyadari yang telah penulis susun ini banyak terdapat kekurangan baik dari segi materi maupun penyusunannya untuk itu kritik dan saran sangat diharapkan guna perbaikan dan penyempurnaan ini.

Bengkulu, 2022

Penulis

## DAFTAR ISI

	<b>Halaman</b>
<b>HALAMAN JUDUL</b> .....	<b>i</b>
<b>SAMPUL DALAM</b> .....	<b>ii</b>
<b>PERSETUJUAN</b> .....	<b>iii</b>
<b>PENGESAHAN</b> .....	<b>iv</b>
<b>DAFTAR RIWAYAT HIDUP</b> .....	<b>v</b>
<b>MOTTO DAN PERSEMBAHAN</b> .....	<b>vi</b>
<b>ABSTRAK</b> .....	<b>vii</b>
<b>ABSTRACT</b> .....	<b>viii</b>
<b>RINGKASAN</b> .....	<b>ix</b>
<b>PERNYATAAN KEASLIAN SKRIPSI</b> .....	<b>x</b>
<b>KATA PENGANTAR</b> .....	<b>xi</b>
<b>DAFTAR ISI</b> .....	<b>xii</b>
<b>DAFTAR TABEL</b> .....	<b>xiv</b>
<b>DAFTAR GAMBAR</b> .....	<b>xv</b>
<b>DAFTAR LAMPIRAN</b> .....	<b>xvii</b>
<b>BAB I PENDAHULUAN</b>	
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah .....	3
1.3. Batasan Masalah .....	3
1.4. Tujuan Penelitian.....	4
1.4.1. Tujuan Umum.....	4
1.4.2. Tujuan Khusus.....	4
1.5. Manfaat Penelitian.....	4
<b>BAB II. LANDASAN TEORI</b>	
2.1. Sistem Pakar .....	6
2.1.1 Konsep Dasar Sistem Pakar.....	8
2.1.2 Arsitektur Sistem Pakar .....	9
2.1.3 Mesin Inferensi .....	10
2.2. Pengertian Metode <i>Dempster Shafer</i> .....	12

2.3. Website .....	14
2.4. Bahasa Pemrograman PHP .....	14
2.5. Konsep Rancangan Database .....	16
2.5.1 Pengertian Database .....	16
2.5.2 MySQL ( <i>My Structure Query Language</i> ) .....	17
2.6. UML ( <i>Unified Modeling Language</i> ) .....	17
2.6.1. <i>Activity Diagram</i> .....	18
2.6.2. <i>Use case Diagram</i> .....	19
2.6.3. <i>Class Diagram</i> .....	21
2.7. <i>Flowchart</i> .....	22
<b>BAB III. METODOLOGI PENELITIAN</b>	
3.1. Subjek Penelitian .....	24
3.1.1. Gambaran Umum Dinas Tanaman Pangan .....	24
3.1.2. Struktur Organisasi .....	24
3.1.3. Waktu dan Tempat Penelitian.....	25
3.2. Metode Penelitian .....	25
3.3. <i>Software</i> dan <i>Hardware</i> .....	26
3.3.1. Perangkat Keras ( <i>hardware</i> ).....	26
3.3.2. Perangkat Lunak ( <i>software</i> ).....	27
3.4. Metode Pengumpulan Data .....	27
3.5. Metode Perancangan Sistem.....	28
3.5.1. Analisis Sistem Aktual .....	28
3.5.2. Analisa Sistem Pakar Metode <i>Dempster Shafer</i> .....	28
3.6. Perancangan Pengujian.....	52
<b>BAB IV. HASIL DAN PEMBAHASAN</b>	
4.1. Hasil Program dan Pembahasan .....	53
4.2. Hasil Pengujian.....	63
<b>BAB V. KESIMPULAN DAN SARAN</b>	
5.1. Kesimpulan.....	65
5.2. Saran .....	65
<b>DAFTAR PUSTAKA</b>	
<b>LAMPIRAN</b>	

## DAFTAR TABEL

<b>Tabel</b>	<b>Halaman</b>
2.1 Notasi <i>Activity</i> Diagram .....	19
2.2 Simbol Use Case Diagram .....	20
2.3 Simbol Class Diagram.....	22
2.4 Simbol dan Fungsi Flowchart .....	23
3.1 Data Penyakit .....	29
3.2 Gejala dan Nilai Belief.....	30
3.3 Solusi.....	31
3.4 Gejala .....	32
3.5 Rulebase .....	33
3.6. Range Nilai Belief dan Plausability .....	34
3.7. Nilai Belief dan Plausability .....	35
3.8. Nilai Belief.....	36
3.9. Matriks I M3 .....	36
3.10. Densitas Baru Untuk M3.....	38
3.11. Nilai Densitas .....	38
3.12. Rancangan Tabel Pengguna .....	41
3.13. Rancangan Tabel Gejala .....	41
3.14. Rancangan Tabel Penyakit.....	42
3.15. Rancangan Tabel Rule .....	42
3.16. Rancangan Tabel Solusi.....	42
3.17. Rancangan Tabel Konsultasi.....	43
4.1. Pengujian.....	64

## DAFTAR GAMBAR

<b>Gambar</b>	<b>Halaman</b>
2.1 Konsep Dasar Sistem Pakar .....	8
2.2 Arsitektur Sistem Pasar .....	10
3.1. Tahapan Metode Waterfall.....	25
3.2 Diagram Konteks .....	39
3.3 Diagram Flow Data Level 0 .....	39
3.4 Entity Relationship Diagram.....	40
3.5 Struktur Menu .....	40
3.6 Rancangan Menu Halaman Awal.....	43
3.7 Rancangan Daftar Penyakit.....	45
3.8 Rancangan Login .....	45
3.10 Rancangan Dashboard Admin.....	45
3.11 Rancangan Data Penyakit dan Solusi.....	45
3.12 Rancangan Tambah Data Penyakit .....	46
3.13 Rancangan Data Solusi .....	46
3.14 Rancangan Data Gejala .....	47
3.15 Rancangan Tambah Data Gejala .....	47
3.16 Rancangan Rule Dempster Shafer .....	48
3.17 Rancangan Laporan Data Gejala.....	48
3.18 Rancangan Laporan Data Pengguna .....	49
3.19 Rancangan Cetak Laporan Admin .....	49
3.20 Rancangan Log out .....	50
3.21 Rancangan Registrasi User .....	50
3.22 Rancangan Konsultasi .....	50
3.23 Rancangan Hasil Diagnosa Penyakit .....	51
3.24 Rancangan Cetak Laporan Pengguna.....	51
4.1 Tampilan Halaman Awal .....	53
4.2 Tampilan Halaman Daftar Penyakit.....	54
4.3 Tampilan Halaman Login Admin .....	54

4.4 Tampilan Halaman Dashboard Admin.....	55
4.5 Tampilan Halaman Data Penyakit dan Solusi.....	55
4.6 Tampilan Halaman Tambah Data Penyakit .....	56
4.7 Tampilan Halaman Data Solusi .....	56
4.8. Tampilan Halaman Data Gejala.....	57
4.9. Tampilan Halaman Tambah Data Gejala.....	57
4.10 Tampilan Halaman Rule Dempster Shafer .....	58
4.10 Tampilan Halaman Rule Dempster Shafer .....	58
4.11 Tampilan Halaman Laporan Data Gejala .....	59
4.12 Tampilan Halaman Cetak Laporan pengguna .....	59
4.13 Tampilan Halaman Cetak Laporan Admin .....	60
4.14 Tampilan Halaman Logout .....	60
4.15 Tampilan Halaman Registrasi User .....	61
4.16 Tampilan Halaman Konsultasi.....	61
4.17 Tampilan Halaman Hasil Diagnosa Gejala yang dipilih.....	62
4.18 Tampilan Halaman Hasil Diagnosa Table nilai Gejala.....	62
4.19 Tampilan Halaman Hasil Diagnosa .....	63
4.20 Tampilan Halaman Cetak Laporan Pengguna.....	63

## **DAFTAR LAMPIRAN**

1. Surat Balasan Penelitian
2. Surat Keterangan Melaksanakan Penelitian
3. Surat Demo Program
4. Kartu Bimbingan Skripsi
5. Time Schedule
6. Data Pendukung
7. Kode Program

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Dengan memanfaatkan kemajuan teknologi terutama dibidang teknologi komputer sangat pesat, Berbagai produk perangkat lunak (*software*) maupun perangkat keras (*hardware*) di produksi secara berkala. Implementasinya pun sudah meluas ke berbagai bidang diantaranya pertanian, kesehatan, industri dan di berbagai bidang lainnya. Sistem pakar (*expert system*) merupakan salah satu bidang teknik kecerdasan buatan (*artificial intelligence*) yang cukup diminati karena penerapannya diberbagai bidang baik bidang ilmu pengetahuan maupun bisnis.

Dinas Tanaman Pangan dan Holtikultura Provinsi Bengkulu merupakan instansi pemerintah yang kegiatannya melakukan, memonitoring perkembangan akan tanaman pangan dan holtikultura yang ada di provinsi Bengkulu mengalami kesulitan akan serangan hama dan penyakit pada tanaman singkong, sehingga perlu dilakukan penanganan yang serius. Beberapa diantara petani singkong ini ada yang tidak tahu kapan hama dan penyakit menyerang untuk waktu dan musim tertentu, sehingga mereka tidak bisa melakukan persiapan dan pencegahan lebih dini untuk mengatasi serangan hama dan penyakit tersebut. Organisme Pengganggu Tanaman (OPT) yang menimbulkan kerusakan pada pertanaman singkong, di antaranya adalah mati pucuk (*Antraknose*). Penyakit mati pucuk merupakan salah satu penyebab rusaknya pertanaman.

Dalam Menangani OPT yang terdapat pada tanaman singkong dengan memaparkan gejala yang diderita tanaman singkong tersebut, seorang pakar memberikan berbagai kemungkinan penyakit yang diderita tersebut. Oleh karena itu sistem pakar dengan menerapkan metode Dempster Shafer ini memungkinkan seorang petani dapat melakukan pekerjaan seorang pakar sekaligus dapat mengetahui probabilitas atau presentase dari penyakit yang kemungkinan dideritanya, sehingga produktivitas kerja semakin meningkat dan waktu yang digunakan untuk menyelesaikan pekerjaan semakin singkat. Kurangnya pengetahuan untuk menindaklanjuti permasalahan-permasalahan di atas tidak hanya dirasakan oleh para petani singkong, namun hal itu juga dirasakan oleh pihak penyuluh pertanian. Penyuluh pertanian kadang juga mengalami kesulitan dalam mengidentifikasi hama dan penyakit yang menyerang pada tanaman singkong walaupun terlihat adanya perubahan pada tanaman. Bahkan kadang-kadang penyuluh tidak tahu obat yang digunakan untuk memberantas hama tanaman tersebut. Penyuluh juga kesulitan dalam memberikan penjelasan kepada petani tentang gejala-gejala yang dialami tanaman apabila terkena suatu hama dan penyakit, karena itu program ini akan membantu meringankan dan membantu penyuluh pertanian dan petani yang masih baru dalam mengidentifikasi hama dan penyakit tanaman yang menyerang pada tanaman singkong.

Sistem pakar merupakan salah satu pemecahan yang potensial untuk mengatasi masalah identifikasi penyakit, khususnya penyakit pada tanaman singkong. Teori Dempster-Shafer mengijinkan untuk menentukan *degree of belief* (derajat kepercayaan) dan *plausible reasoable* (pemikiran yang masuk

akal) yang digunakan untuk mengkombinasikan potongan informasi yang terpisah untuk mengkalkulasikan kemungkinan dari suatu peristiwa. Setiap potongan informasi memiliki nilai pengaruh *evidence* terhadap hipotesa yang ada. Dengan menggunakan nilai densitas maka diagnosa dapat dilakukan, hipotesa yang memiliki nilai probabilitas densitas akhir paling tinggi yang merupakan diagnosa yang paling baik.

Berdasarkan latar belakang yang telah diuraikan, maka penulis untuk merancang sistem pakar yang diberi judul “**Sistem Pakar Penyakit Pada Tanaman Singkong Menggunakan Metode Dempster Shafer Berbasis WEB**”.

## 1.2 Rumusan Masalah

Dari permasalahan diatas maka yang menjadi rumusan masalah pada penelitian ini adalah bagaimana merancang sistem pakar penyakit pada tanaman singkong menggunakan metode Dempster Shafer berbasis Web.

## 1.3 Batasan Masalah

Agar tidak memperluas materi penulisan maka batasan-batasan dan ruang lingkup penulisan antara lain adalah sebagai berikut :

1. Data yang diinput merupakan data penyakit, data gejala dan data solusi
2. Hasil keluaran atau output berupa analisa akhir berupa nama penyakit pada tanaman singkong dan nilai densitas berdasarkan gejala, beserta penjelasan solusi untuk menangani penyakit tersebut secara umum.
3. Penyimpanan data-data pada sistem pakar ini menggunakan database MySQL dengan bahasa pemrograman PHP

4. Menggunakan metode *Dempster Shafer* untuk memberikan tingkat kepercayaan terhadap hasil analisa yang diberikan dan dengan metode inferensi yang digunakan dalam penulisan ini adalah *forward chaining*.

## **1.4 Tujuan Penelitian**

### **1.4.1 Tujuan Umum**

Sebagai salah satu syarat untuk melanjutkan penulisan skripsi pada fakultas Ilmu Komputer Program Studi Informatika Universitas Dehasen Bengkulu

### **1.4.2 Tujuan Khusus**

Adapun tujuan khusus dari penelitian ini adalah sebagai berikut :

1. Untuk merancang sistem pakar yang dapat digunakan untuk mengidentifikasi penyakit pada tanaman singkong serta solusi penanganannya secara umum.
2. Untuk membangun sebuah sistem berbasis web yang dapat memberikan informasi tentang penyakit pada tanaman singkong serta tindakan pencegahan yang dapat dengan mudah diakses secara cepat.

## **1.5 Manfaat Penelitian**

Adapun manfaat yang diharapkan dari penulisan skripsi ini adalah :

1. Dapat memudahkan pengguna (user) untuk mengidentifikasi penyakit pada tanaman singkon serta penanganannya secara umum.
2. Meringankan permasalahan masyarakat untuk mengakses informasi dan berkonsultasi dengan sistem secara mudah dan cepat

3. Dapat memudahkan pengguna (*user*) dalam mengetahui informasi tentang penyakit pada tanaman singkong serta tindakan pengendalian dan cara budidaya yang baik
4. Bagi penulis, melalui penulisan skripsi ini dapat dijadikan sebagai sarana pembelajaran dalam mengimplementasikan bahasa pemrograman PHP.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Pengertian Sistem Pakar**

Sistem pakar adalah aplikasi berbasis komputer yang digunakan untuk menyelesaikan masalah sebagaimana yang dipikirkan oleh pakar. Pakar yang dimaksud di sini adalah orang yang memiliki keahlian khusus yang dapat menyelesaikan masalah yang tidak dapat diselesaikan oleh orang awam. Sebagai contoh, dokter adalah seorang pakar yang mampu mendiagnosis penyakit yang diderita pasien serta dapat memberikan penatalaksanaan terhadap penyakit tersebut. Tidak semua orang dapat mengambil keputusan mengenai diagnosis dan memberikan penatalaksanaan suatu penyakit (Hidayat & Gumilang, 2017).

Sistem pakar adalah orang yang memiliki pengetahuan, penilaian, pengalaman, dan metode khusus serta kemampuan untuk menerapkan bakat dan memberikan nasihat serta memecahkan persoalan. Pakar mengetahui fakta mana yang penting dan memahami arti hubungan diantaranya. Sejauh ini tidak ada definisi standar untuk pakar, tetapi performa keputusan dan tingkat pengetahuan orang adalah kriteria umum dalam menentukan apakah seseorang adalah pakar (Alim et al., 2020).

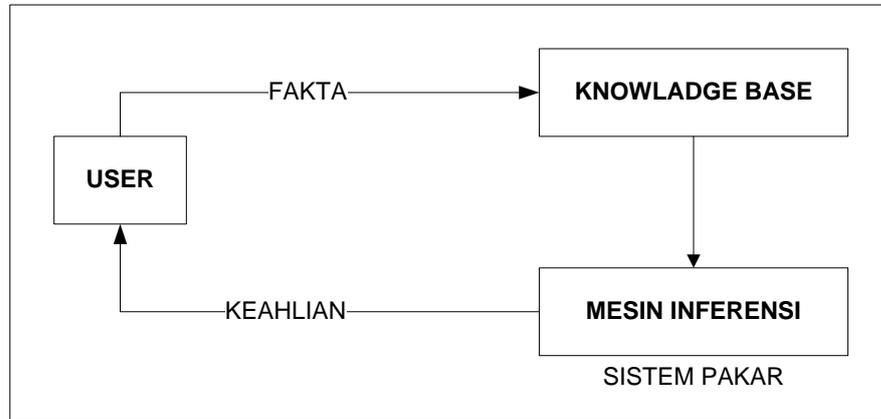
Sistem pakar (*Expert System*) adalah penggabungan pengetahuan dan penelusuran data untuk memecahkan masalah secara yang normal memerlukan keahlian manusia sebagai seorang pakar. Sistem pakar memiliki ciri dan karakteristik yaitu: adanya pakar, pemakai dan sistem

sedangkan pakar adalah orang yang mempunyai basis pengetahuan dalam menangani suatu masalah. Pemakai adalah orang yang ingin berkomunikasi atau berkonsultasi, sedangkan sistem menyediakan berbagai fasilitas untuk menggabungkan pakar dan pemakai. Adapun konsep dasar dari sistem pakar mengandung keahlian, ahli, pengalihan keahlian, inferensi, dan kemampuan untuk menjelaskan.

Sistem pakar adalah program komputer yang mensimulasi penilaian dan perilaku manusia atau organisasi yang memiliki pengetahuan dan pengalaman ahli dalam bidang tertentu. Biasanya sistem seperti ini berbasis pengetahuan yang berisi akumulasi pengalaman dan satu set aturan untuk menerapkan pengetahuan dasar untuk setiap situasi tertentu. Sistem pakar yang canggih dapat ditingkatkan dengan menambah basis pengetahuan atau set aturan. Pemrosesan yang dilakukan oleh sistem pakar merupakan pemrosesan pengetahuan bukan pemrosesan data pada sistem pakar komputer konvensional. Pengetahuan (knowledge) adalah pemahaman secara praktis maupun teoritis terhadap suatu obyek atau domain tertentu (Alim et al., 2020).

Pengetahuan yang digunakan dalam sistem pakar merupakan serangkaian informasi mengenai gejala diagnosa, sebab-akibat, aksi reaksi tentang suatu domain tertentu (misalnya, domain diagnosa medis). Secara umum, definisi tradisional sebuah program komputer biasa: Algoritma + Struktur data = program dalam sistem pakar, definisi berubah menjadi. Mesin inferensi + Pengetahuan = Sistem Pakar. Dengan sistem pakar, masalah yang seharusnya hanya dapat diselesaikan oleh pakar atau ahli,

dapat diselesaikan oleh orang biasa/awam. Sedangkan para ahli, sistem pakar membantu aktifitas mereka sebagai asisten yang seolah-olah sudah mempunyai banyak pengalaman.



**Gambar 2.1 Konsep Dasar Sistem Pakar**

Gambar diatas merupakan gambar konsep dasar sistem pakar, dimana pengguna (*User*) menyampaikan fakta atau informasi kepada sistem pakar, kemudian fakta dan informasi tersebut akan di simpan ke *Knowledge-base* (Basis Pengetahuan), dan di olah dengan mekanisme inferense, sehingga sistem dapat memberikan respon kepada penggunanya berupa keahlian atau jawaban berdasarkan pengetahuan yang dimilikinya.

### 2.1.1 Konsep Dasar Sistem Pakar

Sistem Pakar memiliki beberapa unsur yang terdiri dari enam konsep yaitu sebagai berikut (Aldo & Putra, 2020):

#### a. Keahlian

Keahlian adalah pengetahuan khusus yang didapatkan dari proses belajar dan latihan. Pengetahuan dapat berupa fakta, teori dan aturan untuk memecahkan suatu masalah

b. Ahli

Ahli dapat melakukan kegiatan memformulasikan masalah untuk memecahkan suatu masalah secara cepat dan tepat.

c. Mentransfer keahlian (*Transferring Expertise*)

Mentransfer keahlian adalah suatu kegiatan proses pentransferan keahlian dari seseorang Pakar kedalam komputer agar bisa dipakai oleh orang lain yang bukan Pakar. Pengetahuan tersebut diterapkan kedalam komputer yang dinamakan basis pengetahuan.

d. Menyimpulkan Aturan

Menyimpulkan Aturan adalah kemampuan komputer yang diprogram, penyimpulan ini dilakukan Mesin Inferensi yang meliputi prosedur yang telah ada untuk menyelesaikan suatu masalah.

e. Peraturan (*Rule*)

*Rule* ini diperlukan oleh Sistem Pakar karena menyimpan pengetahuan dalam bentuk peraturan (*Rule*) yang berbentuk *IF THEN ELSE*.

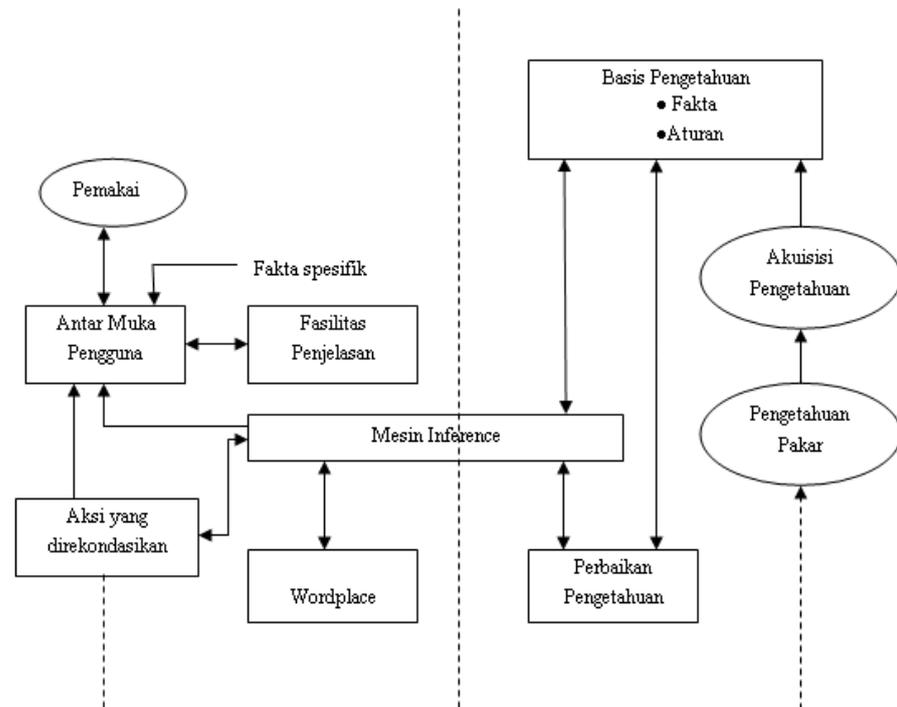
f. Kemampuan menjelaskan (*Explanation Capability*)

Karakteristik Sistem Pakar yang memiliki kemampuan memberi suatu saran mengapa tindakan tersebut harus dikerjakan atau tidak.

### 2.1.2 Arsitektur Sistem Pakar

Arsitektur Sistem pakar disusun dalam dua bagian utama, yaitu lingkungan pengembangan dan lingkungan konsultasi. Lingkungan

pengembangan digunakan untuk memasukan pengetahuan pakar kedalam sistem pakar, sedangkan lingkungan konsultasi digunakan pengguna bukan pakar untuk memperoleh pengetahuan pakar



**Gambar 2.2** Arsitektur Sistem Pakar

### 2.1.3 Mesin Inferensi

Mesin Inferensi merupakan bagian yang mengandung mekanisme fungsi berpikir dan pola-pola penalaran sistem yang digunakan oleh seorang pakar. Mekanisme akan menganalisa suatu masalah tertentu dan selanjutnya akan mencari jawaban, kesimpulan atau keputusan yang terbaik (Aldo & Putra, 2020). Ada dua cara yang dapat dikerjakan dalam melakukan inferensi, yaitu :

#### a. *Forward Chaining* (Alur Maju)

Forward Chaining merupakan strategi yang digunakan dalam Sistem Pakar untuk mendapatkan kesimpulan/keputusan yang

dimulai dengan menelusuri fakta - fakta dan tempat(Aldo & Putra, 2020)

*Forward chaining* merupakan suatu penalaran yang dimulai dari fakta untuk mendapatkan kesimpulan (*conclusion*) dari fakta tersebut. *Forward chaining* bisa dikatakan sebagai strategi inference yang bermula dari sejumlah fakta yang diketahui. Pencarian dilakukan dengan menggunakan rules yang premisnya cocok dengan fakta yang diketahui tersebut untuk memperoleh fakta baru dan melanjutkan proses hingga goal dicapai atau hingga sudah tidak ada rules lagi yang premisnya cocok dengan fakta yang diketahui maupun fakta yang diperoleh (Santi & Septiawan, 2018). *Forward chaining* bisa disebut juga runut maju atau pencarian yang dimotori data (*data driven search*). Jadi, pencarian dimulai dari premis-premis atau informasi masukan (*if*) dahulu kemudian menuju konklusi atau *derived information* (*then*). *Forward Chaining* berarti menggunakan himpunan aturan kondisi-aksi. Dalam metode ini, data digunakan untuk menentukan aturan mana yang akan dijalankan atau dengan menambahkan data ke memori kerja untuk diproses agar ditemukan suatu hasil.

b. *Backward Chaining* (Alur Mundur)

*Backward Chaining* adalah pencocokan fakta atau pernyataan dimulai dari bagian sebelah kanan (THEN dulu). Dengan kata lain, penalaran dimulai dari hipotesis terlebih dahulu, dan untuk

menguji kebenaran hipotesis tersebut harus dicari fakta-fakta yang ada dalam basis pengetahuan (Aldo & Putra, 2020).

## 2.2 Metode Dempster Shafer

Metode *Dempster Shafer* dikenalkan oleh Dempster yang melakukan percobaan model ketidakpastian dengan *range probabilities* dari pada sebuah probabilitas tunggal. Pada tahun 1976, Shafer mempublikasikan teori yang dikenalkan oleh Dempster pada sebuah buku yang berjudul *Mathematical Theory Of Evident*, dimana pada teori tersebut dapat membedakan ketidakpastian dan ketidaktahuan (Nas, 2019)

Teori Dempster Shafer merupakan representasi, kombinasi dan propogasi ketidakpastian, dimana teori ini memiliki beberapa karakteristik yang secara institutif sesuai dengan cara berfikir seorang pakar, namun dengan dasar matematika yang kuat (Aldo & Putra, 2020). Metode Dempster Shafer diterapkan untuk menentukan dan mendefinisikan tingkat keyakinan dan fungsi yang masuk akal untuk mengevaluasi suatu kemungkinan.

Secara umum teori Dempster-Shafer ditulis dalam suatu interval: *[Belief,Plausibility]* *Belief* (Bel) adalah ukuran kekuatan *evidence* dalam mendukung suatu himpunan proposisi. Jika bernilai 0 maka mengindikasikan tidak ada *evidence*, dan jika bernilai 1 menunjukkan adanya kepastian. *Plausibility* (Pls) akan mengurangi tingkat kepastian dari *evidence*. *Plausibility* bernilai 0 sampai 1. Jika yakin akan X', maka dapat dikatakan bahwa  $Bel(X') = 1$ , sehingga rumus di atas nilai dari  $Pls(X) = 0$ .

Adapun langkah-langkah proses perhitungan metode Dempster Shafer sebagai berikut (Nahumury et al., 2020) :

$$Bel(X) = \sum_{Y \subseteq X} m(Y) \dots \dots \dots (1)$$

Keterangan :

Bel (X) : *Belief* (X)

m (Y) : *mass function* dari (Y)

sedangkan *Plausibility* (Pls) diformulasikan sebagai berikut :

$$Pls(X) = 1 - Bel(X') = 1 - \sum_{Y \subseteq X} m(X') \dots \dots \dots (2)$$

Keterangan :

Bel (X) : *Belief* (X)

Pls (X) : *Plausibility* (X)

m (X') : *mass function* dari (X)

m (Y) : *mass function* dari (Y)

Pada teori Dempster Shafer semesta pembicaraan dari sekumpulan hipotesis sering disebut environment, dinotasikan dengan  $\Theta$  seperti persamaan 3

$$(\Theta) = \{\theta_1, \theta_2 \dots \theta_N\} \dots \dots \dots (3)$$

Keterangan .:

$\Theta$  : FOD atau *environment*

$\theta_1, \theta_2 \dots \theta_N$  = elemen atau unsur bagian dari environment

Mass function(m) dalam teori Dempster Shafer adalah tingkat kepercayaan dari suatu evidence. Mass function (m) diformulasikan pada persamaan 4 (Nas, 2019)

$$m_3(Z) = \frac{\sum X \cap Y = Z m_1(X). m_2(Y)}{1 - k} \dots \dots \dots (4)$$

Keterangan :

$m_1, m_2, m_3$  = densitas gejala

$X, Y, Z$  = himpunan penyakit

$k$  = jumlah *conflict evidence*

### 2.3 Website

Website adalah halaman informasi yang disediakan melalui jalur internal sehingga bisa diakses di seluruh dunia, selama terkoneksi dengan jaringan internet (Handayani et al., 2018)

Website awalnya merupakan suatu layanan sajian informasi yang menggunakan konsep hiperlink yang memudahkan surfer (sebutan bagi pemakai komputer yang melakukan penyelusuran informasi di internet) untuk mendapatkan informasi dengan cukup mengklik suatu link berupa teks atau gambar maka informasi dari teks atau gambar akan ditampilkan secara lebih terperinci (Nurmalasari et al., 2019)

Sedangkan *web browser* menggambarkan bahwa Web browser Digunakan untuk menampilkan hasil website yang telah dibuat. Web browser yang paling sering digunakan, di antaranya Mozilla Firefox, Google Chrome, Internet Explorer, Opera, dan Safari (Handayani et al., 2018).

### 2.4 Bahasa Pemrograman PHP

PHP singkatan dari *Hypertext Preprocessor* yang merupakan *server-side programming*, yaitu bahasa pemrograman yang diproses di sisi server (Handayani et al., 2018)

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

Dari pendapat di atas dapat disimpulkan bahwa web merupakan layanan yang dapat oleh pemakai komputer terhubung ke internet, baik berupa teks, gambar, suara maupun video yang interaktif dan mempunyai kelebihan untuk menghubungkan (link) satu dokumen dengan dokumen lainnya (*hypertext*) yang dapat diakses melalui sebuah *browser*.

Kelebihan bahasa pemrograman PHP yaitu :

- a. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
- b. *Web server* yang mendukung PHP dapat ditemukan dimana-mana dari mulai apache, IIS, Lightpd, nginx, hingga Xitami dengan konfigurasi lebih mudah.
- c. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan developer yang siap membantu pengembangan.
- d. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak
- e. PHP adalah bahasa open source yang dapat digunakan di beberapa mesin (*Linux, Unix, Macintosh, Windows*) dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah system

## 2.5 Konsep Perancangan Database

### 2.5.1 Pengertian Database

*Database* (Basis Data) adalah suatu kumpulan data terhubung (*interrelated data*) yang disimpan secara bersama-sama pada suatu media, tanpa mengatap satu sama lain atau tidak perlu suatu kerangkapan data (*controlled redundancy*) (Novendri, Saputra, & Firman, 2019).

Basis data (*Database*) yaitu kumpulan koleksi data-data yang saling berhubungan secara logika yang isinya didesain untuk memenuhi kebutuhan informasi dari suatu perusahaan (Sahi, 2020) Ada beberapa istilah umum yang sering dipakai pada database, yaitu sebagai berikut :

- a. *Field*, yaitu sekumpulan kecil dari kata atau sebuah deretan angka-angka
- b. *Record*, yaitu kumpulan dari field yang berelasi secara logis
- c. *File*, yaitu kumpulan dari record yang berelasi secara logis
- d. *Entity*, yaitu orang, tempat, benda, atau kejadian yang berkaitan dengan informasi yang disimpan
- e. *Attribute*, yaitu setiap karakteristik yang menjelaskan suatu entity
- f. *Primary key*, yaitu sebuah *field* yang nilainya unik yang tidak sama antara satu record dengan *record* yang lain.
- g. *Foreign key*, yaitu sebuah *field* yang nilainya berguna untuk menghubungkan *primary key* yang berada pada *table* yang berbeda

### 2.5.2 MySQL (*My Structure Query Language*)

MySQL merupakan *database server* yang bersifat *multiuser* dan *multi-threaded*. SQL adalah bahasa database standar yang memudahkan penyimpanan, pengubahan dan akses informasi. Pada MySQL dikenal istilah database dan tabel. Tabel adalah sebuah struktur data dua dimensi yang terdiri dari baris-baris record dan kolom (Nurmalasari et al., 2019).

MySQL adalah produk *DataBase Management System* (DBMS) *open source* yang berjalan pada UNIX, Linux, dan Windows. Sumber dan kode biner MySQL dapat didownload dari situs Web MySQL (<http://www.mysql.com>). Keterbatasan MySQL tidak mendukung *View*, prosedur tersimpan, maupun trigger. Akan tetapi, semua hal tersebut ada pada *to-do-list* MySQL, sehingga periksa dokumentasi terakhir untuk menentukan apakah beberapa fitur-fitur tersebut telah ditambahkan ke produk tersebut pada realease-realease yang terbaru. (Sahi, 2020)

### 2.6 UML (*Unified Modeling Language*)

UML adalah salah satu *tool* atau model untuk merancang pengembangan software yang berbasis *object-oriented*. UML sendiri juga memberikan standar penulisan sebuah sistem *blueprint*, yang meliputi konsep proses bisnis, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen yang diperlukan dalam sistem *software* (Sonata & Sari, 2019)

UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung”. Beberapa pemodelan yang termasuk kedalam pemodelan UML seperti *use case diagram*, *class diagram*, *activity diagram*, dan *sequence diagram* (Syarif & Nugraha, 2020)

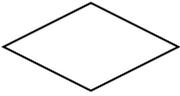
Adapun tujuan dari UML adalah:

1. Merancang perangkat lunak.
2. Sarana komunikasi antara perangkat lunak dengan proses bisnis.
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasi sistem yang ada, proses-proses dan organisasinya

### **2.6.1 Activity Diagram**

*Activity* diagram menggambarkan berbagai aliran aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, keputusan yang mungkin terjadi, dan bagaimana mereka berakhir (Andrianto dan Softwan, 2018). Pada dasarnya, *activity* diagram merupakan variasi dari statechart diagram. *Activity* diagram mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah *activity* diagram bisa mendukung perilaku paralel sedangkan *flowchart* tidak bisa. Berikut adalah notasi *activity* diagram.

Tabel 2.1 Notasi *Activity Diagram*

Gambar	Nama Simbol	Keterangan
	Status Awal	Sebuah diagram aktivitas memiliki sebuah status awal
	Status Akhir	Status akhir yang dilakukan sistem sebuah diagram aktivitas memiliki sebuah status akhir
	Aktivitas	Aktivitas yang dilakukan sistem biasanya diawali dengan kata kerja
	Decision / Percabangan	Percabangan dimana ada pilihan aktivitas yang lebih dari satu
	<i>Fork</i>	Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu

### 2.6.2 Use Case Diagram

*Use case* diagram digunakan untuk memodelkan bisnis proses berdasarkan perspektif pengguna (Andrianto dan Softwan, 2018).

*Use case* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-*create* sebuah daftar belanja, dan sebagainya.

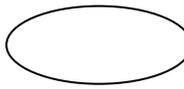
Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. *Use case diagram* dapat sangat membantu bila kita sedang menyusun *requirement* sebuah sistem, mengkomunikasikan

rancangan dengan klien, dan merancang *test case* untuk semua *feature* yang ada pada sistem. Sebuah *use case* dapat meng *include* fungsionalitas *use case* lain sebagai bagian dari proses dalam dirinya.

Secara umum diasumsikan bahwa *use case* yang di *include* akan dipanggil setiap kali *use case* yang meng *include* dieksekusi secara normal. Sebuah *use case* dapat di *include* oleh lebih dari satu *use case* lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang *common*. Sebuah *use case* juga dapat meng *extend use case* lain dengan *behavior* nya sendiri. Sementara hubungan generalisasi antar *use case* menunjukkan bahwa *use case* yang satu merupakan spesialisasi dari yang lain.

**Tabel 2.2 Simbol Use Case Diagram**

Gambar	Nama Simbol	Keterangan
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>Use Case</i> .
	<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>Independent</i> )
	<i>Generalization</i>	Hubungan dimana objek anak( <i>Descended</i> ) berbagi perilaku dan struktur data dari objek yang di atasnya objek induk.

Gambar	Nama Simbol	Keterangan
	<i>Include</i>	Menspesifikasikan bahwa use case sumber secara explicit.
	<i>Extend</i>	Menspesifikasikan bahwa use case target memperluas perilaku pada use case sumber pada sebuah titik diberikan.
	<i>Assosiation</i>	Apa yang menghubungkan objek satu dengan objek yang lainnya.
	<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur dari sebuah <i>actor</i> .
	<i>Colaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya.
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

### 2.6.3 Class Diagram

*Class* diagram menggambarkan struktur statis dari kelas dalam sistem anda dan menggambarkan atribut, operasi dan hubungan antara kelas (Haviluddin, 2017). *Class* diagram membantu dalam memvisualisasikan struktur kelas -kelas dari suatu sistem dan

merupakan tipe diagram yang paling banyak dipakai. Selama tahap desain, *class* diagram berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat.

*Class* diagram memiliki tiga area pokok:

1. Nama (dan *stereotype*)
2. Atribut
3. Metoda

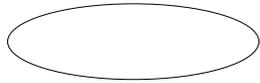
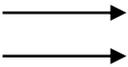
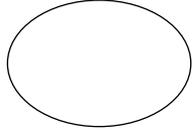
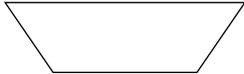
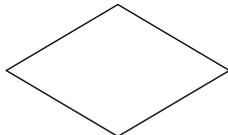
**Tabel 2.3 Simbol Class Diagram**

Gambar	Nama Simbol	Keterangan
	<i>Class</i>	Himpunan dari objek- objek yang berbagi atribut serta operasi yang sama.
	<i>Associatiom</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

## 2.7 Flowchart

*Flowchart* adalah representasi secara simbolik dari suatu algoritma atau prosedur untuk menyelesaikan suatu masalah, dengan menggunakan *flowchart* akan memudahkan pengguna melakukan pengecekan bagian-bagian yang terlupakan dalam analisis masalah, disamping itu *flowchart* juga berguna sebagai fasilitas untuk berkomunikasi antara pemrogram yang bekerja dalam tim suatu proyek. *Flowchart* membantu memahami urutan-urutan logika yang rumit dan panjang. *Flowchart* membantu mengkomunikasikan jalannya program ke orang lain (bukan pemrogram) akan lebih mudah (Santoso & Nurmalina, 2017).

Tabel 2.4 Simbol dan Fungsi *Flowchart*

Gambar	Nama Simbol	Keterangan
	<i>Start / Mulai</i> <i>End / Selesai</i>	Simbol yang digunakan untuk memulai / selesai
	<i>Flow</i>	Simbol arus/ <i>flow</i> yang menyatakan jalannya proses
	<i>Connector</i>	Simbol <i>connector</i> , (menyatakan sambungan dari proses ke proses lainnya dalam hal yang sama)
	<i>Process</i>	Simbol proses yaitu menyatakan suatu tindakan
	<i>Manual Operation</i>	Simbol manual, menyatakan suatu tindakan
	<i>Decision</i>	Simbol <i>decision</i> , menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan
	<i>Keying Operation</i>	Simbol <i>keying operation</i> menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard
	<i>Input/Output</i>	Simbol <i>input/output</i> menyatakan proses input/output
	<i>Document</i>	Simbol dokumen mencetak keluaran dalam bentuk dokumen

## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1 Subjek Penelitian**

##### **3.1.1 Gambaran Umum Dinas Tanaman Pangan dan Holtikultura Provinsi Bengkulu**

Dinas Tanaman Pangan dan Holtikultura Provinsi Bengkulu Terbentuk dengan berlakunya Peraturan Daerah Nomor 57 Tahun 2016 dan Peraturan Gubernur Bengkulu Nomor 63 Tahun 2016 pada Tanggal 21 Desember 2016 Tentang Kedudukan, Susunan Organisasi Tugas dan Fungsi serta Tata Kerja Dinas Tanaman Pangan dan Holtikultura Provinsi Bengkulu, yang sebelumnya menjadi Badan Ketahanan Pangan adalah SKPD yang berdiri sendiri (terpisah dari Dinas Pertanian), yang saat ini sudah memiliki Gedung Sendiri serta memiliki satu Gedung UPT OKKP-D yang beralamat di Jl. Basuki Rahmat No.13, Padang Jati, Kec. Ratu Samban, Kota Bengkulu, Bengkulu 38222

##### **3.1.2 Struktur Organisasi**

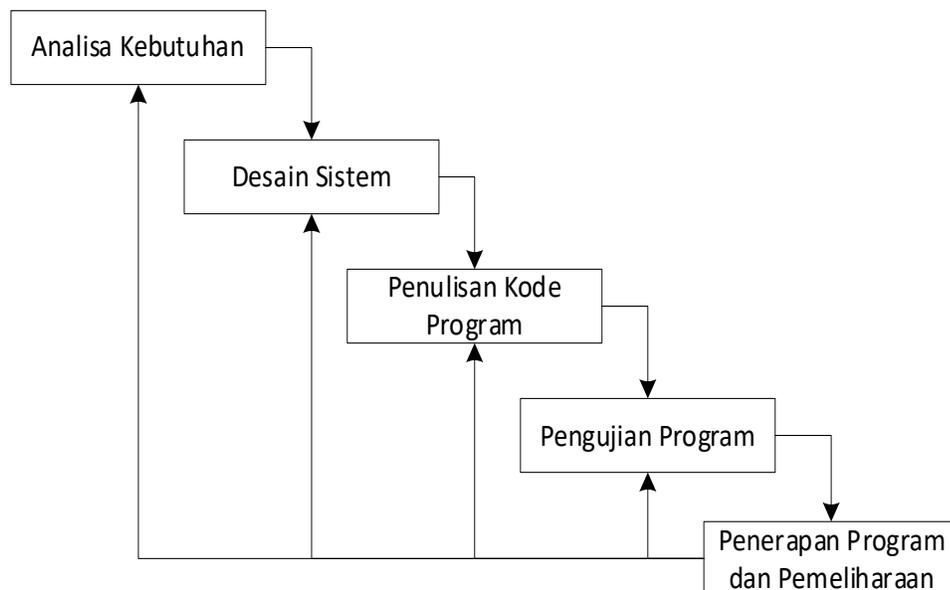
Struktur organisasi merupakan salah satu sarana untuk mencapai tujuan organisasi atau perusahaan melalui fungsi-fungsi manajemen yang dilakukan oleh seorang pemimpin. Adapun struktur organisasi dari Balai Proteksi Tanaman Pangan dan Holtikultura terdapat pada lampiran 1 (terlampir).

##### **3.1.3 Waktus dan Tempat Penelitian**

Penulis melakukan penelitian pada Dinas Tanaman Pangan dan Holtikultura Propinsi Bengkulu yang beralamat di Jl. Basuki Rahmat No.13, Padang Jati, Kec. Ratu Samban, Kota Bengkulu, Bengkulu 38222. Pra penelitian ini dilaksanakan pada bulan Mei 2022 sampai dengan bulan Oktober 2022.

### 3.2 Metode Penelitian

Adapun metode penelitian yang digunakan penulis adalah metode pengembangan sistem. Metode pengembangan sistem yang digunakan adalah *Waterfall*. Adapun tahapan-tahapan metode *waterfall*, antara lain :



**Gambar 3.1. Tahapan Metode *Waterfall***

#### 1. Analisa Kebutuhan

Analisa kebutuhan dilakukan untuk mengetahui permasalahan serta memberikan solusi terhadap permasalahan tersebut, sehingga dapat diketahui sistem seperti apa yang dibutuhkan.

#### 2. Desain Sistem

Desain sistem dilakukan untuk merancang sistem yang diinginkan sesuai dengan hasil analisa kebutuhan sistem. Desain sistem diperlukan sebelum membuat penulisan kode program.

### 3. Penulisan Kode Program

Penulisan kode program merupakan penerjemahan design dalam bahasa yang bisa dikenali oleh komputer.

### 4. Pengujian Program

Program akan dilakukan pengujian untuk mengecek apakah program tersebut sudah berjalan sesuai dengan semestinya atau belum. Jika belum maka akan dilakukan perbaikan terlebih dahulu sebelum program diterapkan ke tempat penelitian.

### 5. Penerapan Program dan Pemeliharaan

Tahapan ini bisa dikatakan final dalam pembuatan sebuah sistem. Setelah melakukan analisa, design dan pengkodean maka sistem yang sudah jadi akan digunakan oleh *user*.

## 3.3 Perangkat Keras dan Perangkat Lunak

### 3.3.1 Perangkat Keras (*Hardware*)

Adapun perangkat lunak yang digunakan dalam penelitian ini adalah sebagai berikut :

- a. Laptop Hp Asus.
- b. RAM 4 Gb
- c. Printer IP 2770
- d. Flashdisk Kingstone 8Gb
- e. Harddisk 500 Gb

f. Mouse Optic Logitec

### 3.3.2 Perangkat Lunak (*Software*)

Adapun perangkat lunak yang digunakan dalam penelitian ini adalah sebagai berikut :

- a. *Microsoft Visual Studio* (VB.Net) 2010.
- b. *Microsoft Office* 2010
- c. *Crystal Report for Visual Studio* versi 13

### 3.4 Metode Pengumpulan Data

Adapun metode pengumpulan data yang penulis gunakan pada penelitian ini adalah sebagai berikut :

#### 1. Observasi

Metode pengumpulan data yang dilakukan dengan cara mengadakan pengamatan atau peninjauan secara langsung ke Dinas Tanaman Pangan dan Holtikultura Provinsi Bengkulu dengan tujuan mendapatkan informasi atau data yang diperlukan untuk penelitian ini.

#### 2. Wawancara

Penulis melakukan tanya jawab langsung kepada Ibu Ir.Idawati selaku Kasubag TU UPTD Perlindungan Tanaman Pangan, dan Holtikultura Provinsi Bengkulu terkait dengan objek penelitian.

#### 3. Studi Pustaka

Metode dimana penulis mempelajari dan mencari data yang berasal dari buku, jurnal dan referensi yang diperoleh dari toko buku dan perpustakaan Universitas Dehasen yang berhubungan dengan masalah yang ditulis.

## 3.5 Metode Perancangan Sistem

### 3.5.1 Analisa Sistem Aktual

Analisa sistem aktual dilakukan agar dapat menemukan masalah-masalah dari pengolahan sistem pakar dan solusi yang akan diberikan dalam menangani penyakit tanaman singkong. Minimnya informasi mengenai penyakit singkong membuat masyarakat kesulitan untuk melakukan pencegahan dan pengobatan terhadap tanaman singkong. Banyak gejala yang timbul ketika tanaman terserang penyakit tersebut tetapi tidak dipahami oleh masyarakat sehingga menyulitkan masyarakat, belum lagi minimnya waktu untuk berkonsultasi dengan pakar (ahli). Oleh karena itu untuk membantu mengatasi permasalahan tersebut dibutuhkan suatu sistem yang dapat mempermudah masyarakat khususnya petani singkong dalam menangani penyakit pada tanaman singkong dengan menerapkan metode *Dempster Shafer*.

### 3.5.2 Analisa Sistem Pakar dengan Metode *Dempster Shafer*

Teori *Dempster-Shafer* merupakan teori yang mampu menangani berbagai kemungkinan yang mengkombinasikan satu kemungkinan dengan fakta yang ada. Dalam teori ini ada berbagai konflik yang dipersatukan untuk mengkombinasikan dari berbagai informasi yang ada. Kumpulan informasi yang bersifat berbeda dan menyeluruh dalam teori ini dikenal dengan *frame discernment* yang dinotasikan dengan  $q$  (*theta*). Teori ini juga melakukan pembuktian berdasarkan *belief function* and *plausible reasoning* (fungsi kepercayaan dan pemikiran yang masuk akal) yang digunakan untuk mengkombinasikan potongan informasi yang terpisah untuk mengkalkulasikan kemungkinan dari suatu peristiwa.

Berikut studi kasus yang akan diimplementasikan dengan metode *Dempster Shafer*.

### A. Data Penyakit

Pada tanaman singkong terdapat beberapa penyakit yang dapat dilihat pada tabel 3.1

**Tabel 3.1 Data Penyakit**

Kode Penyakit	Nama Penyakit
P01	Busuk pangkal batang
P02	Selaput Lendir Putih
P03	Bakteri Hawar Daun
P04	Bercak pada Daun
P05	Antraknose (busuk umbi)

### B. Data Gejala

Untuk mengetahui hasil diagnosa penyebab penyakit singkong, maka dilakukan proses pengujian sistem berupa gejala-gejala yang dapat dilihat oleh mata secara langsung pada tanaman yang mengalami masing-masing gejala, dimana

diambil berdasarkan hasil pengamatan dan penelusuran. Gejala penyakit singkong dapat dilihat pada tabel 3.2

**Tabel 3.2 Gejala dan Nilai Belief**

Kode Gejala	Nama Gejala	Nilai Belief
G01	Batang Berlobang dan kekuningan	0.3
G02	Tampak benjolan seperti bisul bagian	0.5

	batang	
G03	Bercakputih/coklat pada bagian atas daun	0.7
G04	Bercak tepi dibatasi lingkaran ungu	0.5
G05	Bercak coklat	0.4
G06	Daun berkerut	0.6
G07	Daun rontok / gugur	0.6
G08	Daun berlubang	0.7
G09	Daun menguning	0.3
G10	Daun kering	0.3
G11	Ada jamur di bagian bawah daun	0.4
G12	Bercak berukuran besar	0.5
G13	Bercak sering pada ujung daun	0.4
G14	Bercak berbentuk V terbalik	0.6
G15	Daun bagian atas berwarna coklat merata	0.4
G16	Daun bagian bawah berwarna abu-abu	0.4
G17	Bagian tengah bercak berwarna abu-abu penghasil jamur	0.8
G18	Menyerang daun muda	0.5
G19	Menyerang daun dan batang	0.5
G20	Gejala awal berupa lesion berwarna abu-abu	0.2
G21	Lesion dibatasi tulang daun dan membentuk sudut	0.6
G22	Lesio meluas menjadi bercak nekrotik	0.2
G23	Perlendiran masa bakteri yang terjadi pada tangkai, helai daun dan batang	0.3

### C. Ketentuan Data Penyakit dan Gejala

Sebelum mengetahui hasil diagnosa penyebab penyakit tanaman singkong, maka dilakukan proses pengujian sistem berupa gejala-gejala yang dapat dilihat oleh mata secara langsung pada tanaman yang mengalami masing-masing gejala, dimana berdasarkan hasil pengamatan dan penelusuran. Ada 5 jenis penyakit pada tanaman Singkong yang terbagi berdasarkan gejala-gajala dan solusi sebagai berikut :

#### Tabel 3.3 Solusi

Kode Penyakit	Jenis Penyakit	Kode Solusi	Solusi
P1	Busuk Pangkal batang	S01	Dengan pemilihan tanah dan lahan
		S02	Pengolahan tanaman dan bibit yang baik (stek), bahan tanam yang sehat dan bebas infeksi merupakan langkah strategis untuk mengendalikan penyakit tersebut.
		S03	Rendam bibit kedalam air hangat selama 10 menit, untuk menghindari jamur pada bibit
P2	Selaput Lendir Putih	S04	Sebelum penanaman singkong yang menggunakan pupuk organik hendaknya dicampur dengan pupuk Trichokompos dengan jumlah yang cukup, dan harus menggunakan pupuk organik yang sudah matang sempurna.
P3	Bakteri Hawar Daun	S05	Menggunakan varitas yang tahan terhadap penyakit
		S06	Proses stek yang digunakan diambil dari tanaman yang sehat dan tidak terinfeksi bakteri
		S07	Melakukan rotasi pergiliran tanaman
		S08	Perawatan dengan pemangkasan tanaman yang terserang
P4	Bercak Pada Daun	S09	Pengendalian penyakit menggunakan varitas yang tahan terhadap penyakit
		S10	Melakukan pelebaran jarak tanam
		S11	Memangkas daun yang sakit serta melakukan sanitasi kebun
P5	Atraknose	S12	Dengan menanam varietas dan memberikan fungisida

			pada bibit tanaman, dan tidak menanam singkong saat intensitas hujan yang masih tinggi
--	--	--	----------------------------------------------------------------------------------------

**Tabel 3.4 Gejala**

Kode Penyakit	Nama Gejala	Nilai Belief
P01	G01 : Batang Berlobang dan kekuningan	0.3
	G02 : Tampak benjolan seperti bisul bagian batang	0.5
	G03 : Bercak putih/coklat pada bagian atas daun	0.7
	G04 : Bercak tepi dibatasi lingkaran ungu	0.5
	G05 : Bercak coklat	0.4
	G06 : Daun berkerut	0.6
	G07 : Daun rontok / gugur	0.6
P02	G08 : Daun berlubang	0.7
	G09 : Daun menguning	0.3
	G10 : Daun kering	0.3
	G11 : Ada jamur di bagian bawah daun	0.4
	G12 : Bercak berukuran besar	0.5
P03	G13 : Bercak sering pada ujung daun	0.4
	G14 : Bercak berbentuk V terbalik	0.6
	G15 : Daun bagian atas berwarna coklat merata	0.4
	G16 : Daun bagian bawah berwarna abu-abu	0.4
	G17 : Bagian tengah bercak berwarna abu-abu penghasil jamur	0.8
P04	G18 : Menyerang daun muda	0.5
	G19 : Menyerang daun dan batang	0.5
P05	G20 : Gejala awal berupa lesion berwarna abu-abu	0.2
	G21 : Lesion dibatasi tulang daun dan membentuk sudut	0.6
	G22 : Lesio meluas menjadi bercak nekrotik	0.2
	G23 : Perlendiran masa bakteri yang terjadi pada tangkai, helai daun dan batang	0.3

#### D. Tabel *Rulebase*

*Rulebase* bertujuan untuk mengatur keterkaitan antara penyakit dan gejala yang terdapat pada tanaman singkong, *Rulebase* ini akan digunakan

untuk menentukan proses pencarian atau menentukan kesimpulan yang didapat.

**Tabel 3.5 Rulebase**

<b>Rule</b>	<b>Penyakit</b>	<b>Gejala</b>
<i>Rule 1</i>	Busuk pangkal batang	<p><b>IF</b></p> <p>Penyakit menyerang daun tua  <b>AND</b> Bercak daun pada bagian bawah  <b>AND</b> Bercakputih/coklat pada bagian atas daun  <b>AND</b> Bercak tepi dibatasi lingkaran ungu  <b>AND</b> Bercak coklat  <b>AND</b> Daun berkerut  <b>AND</b> Daunr ontok / gugur</p> <p><b>THEN</b></p> <p>Busuk pangkal batang atau akar</p>
<i>Rule 2</i>	Selaput Lendir Putih	<p><b>IF</b></p> <p>Daun berlubang  <b>AND</b> Daun menguning  <b>AND</b> Daun kering  <b>AND</b> Ada jamur di bagian bawah daun  <b>AND</b> Bercak berukuran besar  <b>AND</b></p> <p><b>THEN</b></p> <p>Selaput Lendir Putih</p>
<i>Rule 3</i>	Bakteri Hawar Daun	<p><b>IF</b></p> <p>Bercak sering pada ujung daun  <b>AND</b> Bercak berbentuk V terbalik  <b>AND</b> Daun bagian atas berwarna coklat merata  <b>AND</b> Daun bagian bawah berwarna abu-abu  <b>AND</b> Bagian tengah bercak berwarna abu-abu penghasil jamur</p> <p><b>THEN</b></p> <p>Bakteri Hawar Daun</p>
<i>Rule 4</i>	Bercak pada Daun	<p><b>IF</b></p> <p>Menyerang daun muda  <b>AND</b> Menyerang daun dan batang</p> <p><b>THEN</b></p> <p>Bercak pada Daun</p>

<i>Rule 5</i>	Antrak Nose (busuk umbi)	<b>IF</b> Gejala awal berupa lesi berwarna abu-abu <b>AND</b> Lesi dibatasi tulang daun dan membentuk sudut <b>AND</b> Lesi meluas menjadi bercak nekrotik <b>AND</b> Perlendiran masa bakteri yang terjadi pada tangkai, helai daun dan batang  <b>THEN</b> Antrak Nose (busuk umbi)
---------------	-----------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### E. Penerapan Metode *Dempster Shafer*

Dalam penerapan metode *dempster shafer* dibutuhkan nilai *belief* dan nilai *plausability* yang digunakan untuk menentukan nilai kepercayaan dan nilai ketidakpercayaan gejala terhadap penyakit yang berada dalam satu rule yang sama. Parameter dalam menentukan *range* nilai *belief* dan *plausability* tersebut terlihat pada Tabel 3.6.

**Tabel 3.6. Range Nilai *Belief* dan *Plausability***

Kemungkinan	Keterangan
[1, 1] (Bel = 1 dan Pls = 1)	Semua benar
[0, 0] (Bel = 0 dan Pls = 0)	Semua salah
[0, 1] (Bel = 0 dan Pls = 1)	Ketidakpastian
[1, 0] (Bel = 1 dan Pls = 0)	Kepastian
[Bel, 1] <i>where</i> 0 < Bel < 1	Cenderung mendukung
[0, Pls] <i>where</i> 0 < Pls < 1	Cenderung menolak
[Bel, Pls] <i>where</i> 0 < Bel ≤ Pls < 1	Cenderung mendukung dan menolak

Dimana :

$$Bel(X) = \sum_{Y \in X} m(Y)$$

$$Pls(X) = 1 - Bel(X) = 1 - \sum_{Y \in X} m(X)$$

**Tabel 3.7. Nilai *Belief* dan *Plausability***

Kode Gejala	Nilai Belief	Nilai Plausibility
G01	0,3	0,7
G02	0,5	0,5
G03	0,7	0,3
G04	0,5	0,5
G05	0,4	0,6
G06	0,6	0,4
G07	0,6	0,4
G08	0,7	0,3
G09	0,3	0,7
G10	0,3	0,7
G11	0,4	0,6
G12	0,5	0,5
G13	0,4	0,6
G14	0,6	0,4
G15	0,4	0,6
G16	0,4	0,6
G17	0,8	0,2
G18	0,5	0,5
G19	0,5	0,5
G20	0,2	0,8
G21	0,6	0,4
G22	0,2	0,8
G23	0,3	0,7

**Contoh kasus** : Misalkan gejala yang dipilih yaitu G01, G08, pada saat konsultasi .

Penyelesaian :

Berdasarkan 2 gejala yang dipilih tersebut teridentifikasi ke 2 penyakit yaitu di P1 dan P2 tersebut dengan nilai belief seperti Table 3.7.

**Tabel 3.8. Nilai Belief**

Kode Gejala	Belief	Plausibility
-------------	--------	--------------

G01	0,3	0,7
G08	0,7	0,3

1) Nilai Densitas P1

$$m_1(G01) = 0,3$$

$$m_1(\Theta) = 1 - m_1(G01)$$

$$= 1 - 0,3$$

$$= 0,7$$

$$m_2(G08) = 0,7$$

$$m_2(\Theta) = 1 - m_2(G08)$$

$$= 1 - 0,7$$

$$= 0,3$$

**Tabel 3.9. Matriks I M<sub>2</sub>**

	<b>m<sub>2</sub>(G08) 0,7</b>	<b>m<sub>2</sub>(Θ) 0,3</b>
<b>m<sub>1</sub>(G01) 0,3</b>	0,21	0,09
<b>m<sub>1</sub>(Θ) 0,7</b>	0,49	0,21

Dari nilai Densitas diatas maka untuk mencari nilai

Densitas m<sub>1</sub> :

$$m_1(G01) 0,3 \times m_2(G08) 0,7 = 0,21$$

$$m_1(\Theta) 0,7 \times m_2(G08) 0,7 = 0,49$$

$$m_1(G01) 0,3 \times m_2(\Theta) 0,3 = 0,09$$

$$m_1(\Theta) 0,7 \times m_2(\Theta) 0,3 = 0,21$$

Selanjutnya menghitung tingkat keyakinan (m) *combine* dengan rumus, maka :

$$m_3(Z) = \frac{\sum_{X \cap Y = Z} m_1(X) \cdot m_2(Y)}{1 - \sum_{X \cap Y = \emptyset} m_1(X) \cdot m_2(Y)}$$

$$m_3(Z) = \frac{m_1(G01) \times m_2(\Theta)}{1 - (m_1(\Theta) \times m_2(\Theta))}$$

$$m_3(Z) = \frac{0,09}{1 - (0,7 \times 0,3)}$$

$$m_3(Z) = \frac{0,09}{1 - 0,21}$$

$$m_3(Z) = \frac{0,09}{0,79}$$

$$m_3(Z) = 0,1139$$

Dikarenakan Penyakit P1 hanya terindikasi 1 gejala sehingga diperoleh nilai densitas sebesar 0,1139

## 2) Nilai Densitas P2

$$m_1(G01) = 0,3$$

$$m_1(\Theta) = 1 - m_1(G02)$$

$$= 1 - 0,3$$

$$= 0,7$$

$$m_2(G08) = 0,7$$

$$m_2(\Theta) = 1 - m_2(G08)$$

$$= 1 - 0,7$$

$$= 0,3$$

**Tabel 3.10. Densitas Baru Untuk M<sub>3</sub>**

	<b>m<sub>2</sub>(G08) 0,7</b>	<b>m<sub>2</sub>(Θ) 0,3</b>
<b>m<sub>1</sub>(G01) 0,7</b>	0,49	0,21
<b>m<sub>1</sub>(Θ) 0,3</b>	0,21	0,09

Selanjutnya menghitung tingkat keyakinan (m) *combine* dengan rumus, maka

:

$$m_3(Z) = \frac{\sum_{X \cap Y = Z} m_1(X) \cdot m_2(Y)}{1 - \sum_{X \cap Y = \emptyset} m_1(X) \cdot m_2(Y)}$$

$$m_3(Z) = \frac{m_1(G01) \times m_2(G08)}{1 - m_1 \times m_2(\Theta)}$$

$$m_3(Z) = \frac{0,49}{1 - (0,7 \times 0,3)}$$

$$m_3(Z) = \frac{0,49}{1 - 0,21}$$

$$m_3(Z) = \frac{0,49}{0,79}$$

$$m_3(Z) = 0,6203$$

Sehingga diperoleh nilai densitas dari P1 dan P2 seperti Tabel 3.11.

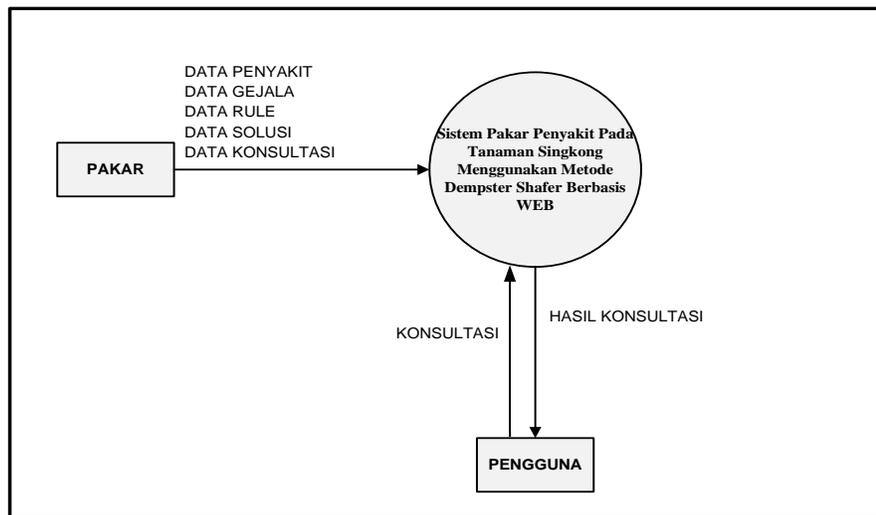
**Tabel 3.11. Nilai Densitas**

<b>Kode Penyakit</b>	<b>Nilai Densitas</b>	<b>Persentase</b>
P1	0,1139	11,39 %
P2	0,6203	62,03%

Berdasarkan perhitungan di atas, nilai densitas Penyakit P2 lebih besar, sehingga dapat disimpulkan gejala pada G01 dan G08 dominan mengarah pada Penyakit P2 (Selaput Lendir Putih) dengan persentase 62,03%, adapun solusi

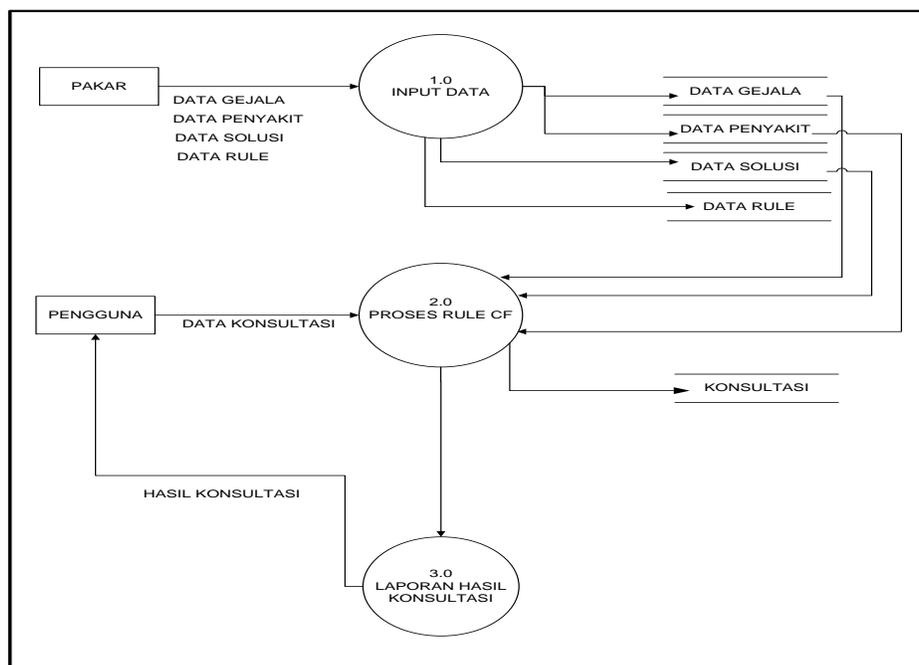
penangannya yaitu sebelum penanaman singkong yang menggunakan pupuk organik hendaknya dicampur dengan pupuk Trickompos dengan jumlah yang cukup, dan harus menggunakan pupuk organik yang sudah matang sempurna.

## F. Diagram Konteks



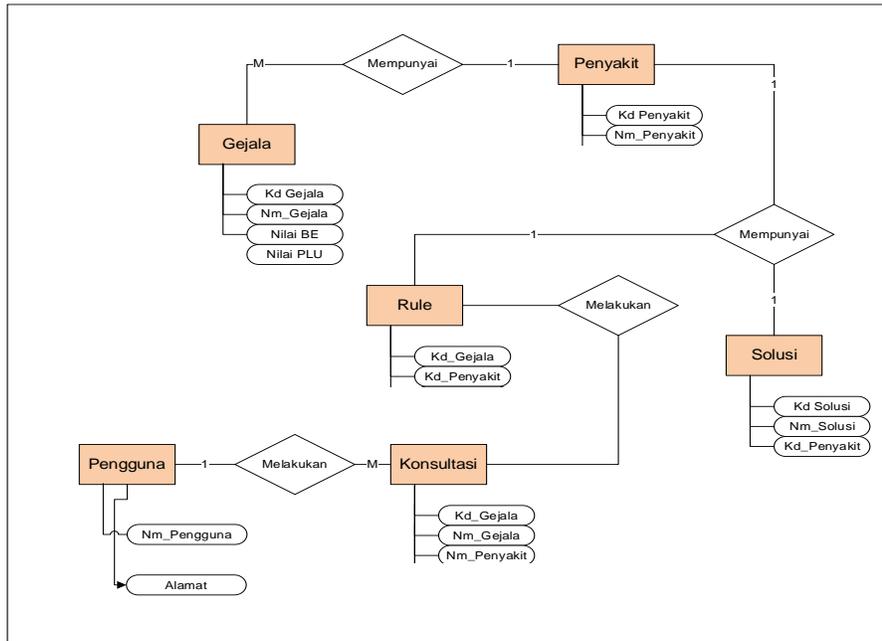
Gambar 3.2 Diagram Konteks

## G. Diagram Flow Data Level 0



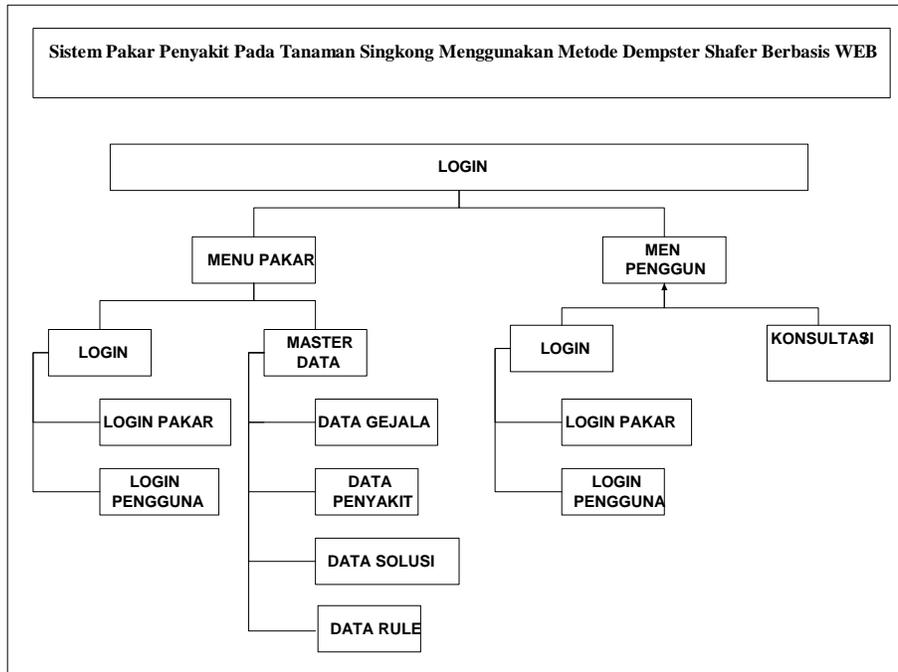
Gambar 3.3 Diagram Flow Data Level 0

**H. Entity Relationship Diagram (ERD)**



**Gambar 3.4 Entity Relationship Diagram**

**I. Sturuktur Menu**



**Gambar 3.5 Struktur Menu**

## J. Rancangan Tabel

Rancangan tabel merupakan rancangan untuk membentuk atau membuat database. Pada rancangan database terdapat 6 (enam) tabel yang penulis buat seperti tabel dibawah ini :

### 1. Tabel Pengguna

Primary Key : Nm\_Pengguna

Secondary Key :

Tabel 3.12 Rancangan Tabel Pengguna

No	Name Field	Type Data	Width	Description
1	Nm_Pengguna	Varchar	25	Nama
2	Alamat	Varchar	100	Alamat

### 2. Tabel Gejala

Primary Key : kd\_gejala

Secondary Key : -

Tabel 3.13 Rancangan Tabel Gejala

No	Name Field	Type Data	Width	Description
1	Kd_gejala	Varchar	5	Kode gejala
2	Nm_gejala	Varchar	35	Nama gejala
3	Nilai_BE	Int	5	Nilai Belief (kepastian)
4	Nilai_PLU	Int	5	Nilai Plausibility (tidak Pasti)

## 3. Tabel Penyakit

Primary Key : Kd\_Penyakit

Secondary Key : -

Tabel 3.14 Rancangan Tabel Penyakit

No	Name Field	Type Data	Width	Description
1	Kd_penyakit	Varchar	5	Kode Penyakit
2	Nm_penyakit	Varchar	35	Nama Penyakit

## 4. Tabel Rule

Primary Key : Kd\_Gejala

Secondary Key :

Tabel 3.15 Rancangan Tabel Rule

No	Name Field	Type Data	Width	Description
1	Kd_Gejala	Varchar	5	Kode Gejala
2	Kd_Penyakit	Varchar	5	Kode Penyakit

## 5. Tabel Solusi

Primary Key : Kd\_solusi

Secondary Key :

Tabel 3.16 Rancangan Tabel Solusi

No	Name Field	Type Data	Width	Description
1	Kd_Solusi	Varchar	5	Kode Solusi
2	Nm_Solusi	Varchar	200	Nama Solusi

3	Kd_Penyakit	Varchar	5	Kode Penyakit
---	-------------	---------	---	---------------

#### 6. Tabel Konsultasi

Primary Key : kd\_Gejala

Secondary Key :

Tabel 3.11 Rancangan Tabel Konsultasi

No	Name Field	Type Data	Width	Description
1	Kd_Gejala	Varchar	5	Kode Solusi
2	Nm_Gejala	Varchar	200	Nama Solusi
3	NM_Penyakit	Varchar	25	Nama Penyakit

### K. Rancangan *Interface* (Antarmuka) Aplikasi

#### 1. Rancangan Halaman Awal

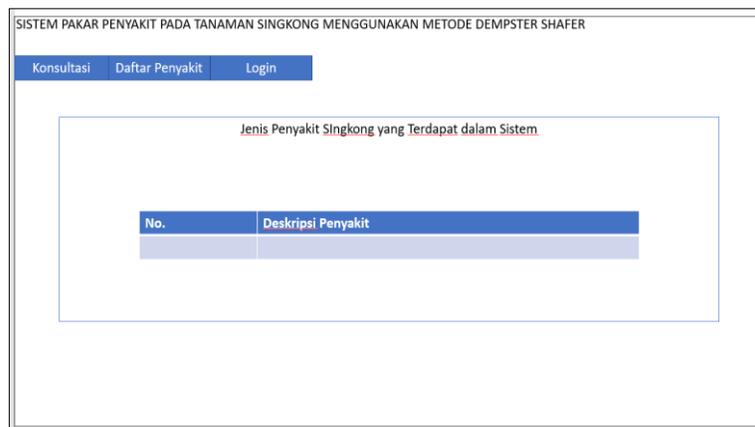
Rancangan halaman awal adalah halaman yang menampilkan tampilan pertama ketika situs diakses.



Gambar 3.6 Rancangan Halaman Awal

## 2. Rancangan Daftar Penyakit

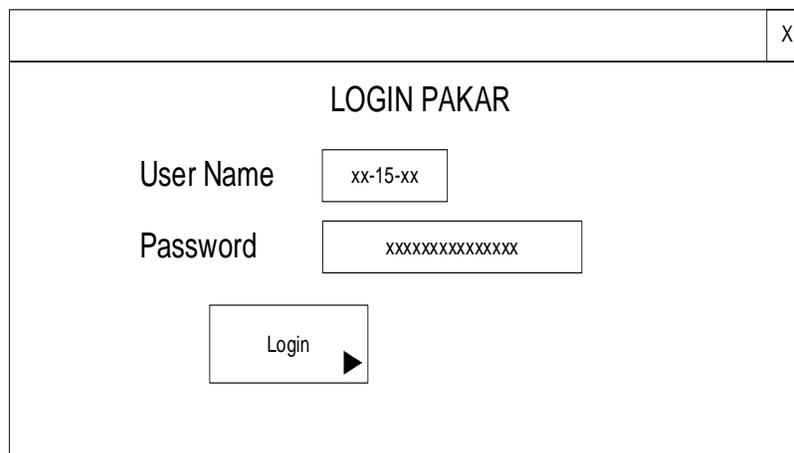
Rancangan ini berguna untuk menampilkan wawasan data penyakit yang terdapat dalam sistem.



No.	Deskripsi Penyakit
-----	--------------------

**Gambar 3.7 Rancangan Daftar Penyakit**

## 3. Rancangan Login



LOGIN PAKAR

User Name

Password

Login

**Gambar 3.8 Rancangan Login**

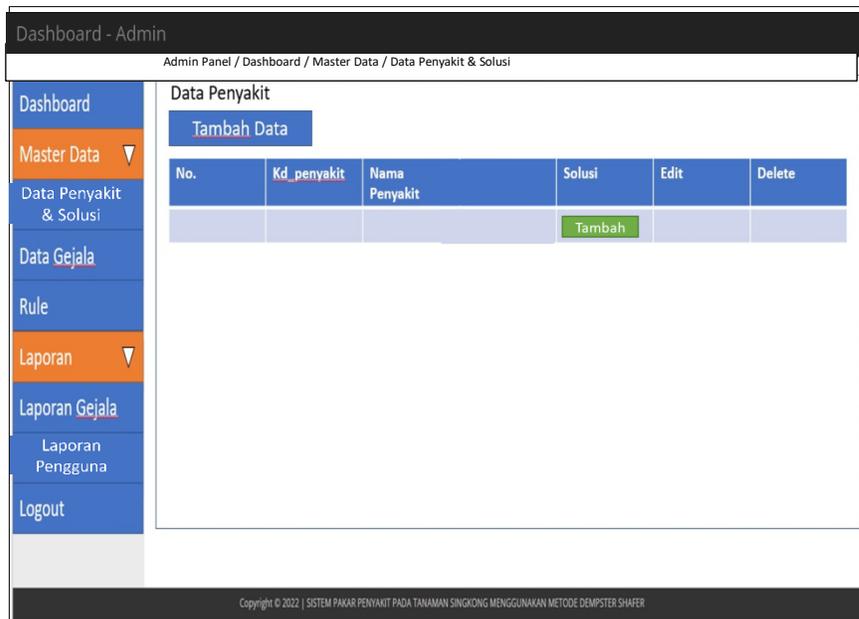
## 4. Rancangan Dashboard Admin

Rancangan dashboard admin adalah halaman yang pertama kali muncul setelah melakukan login.



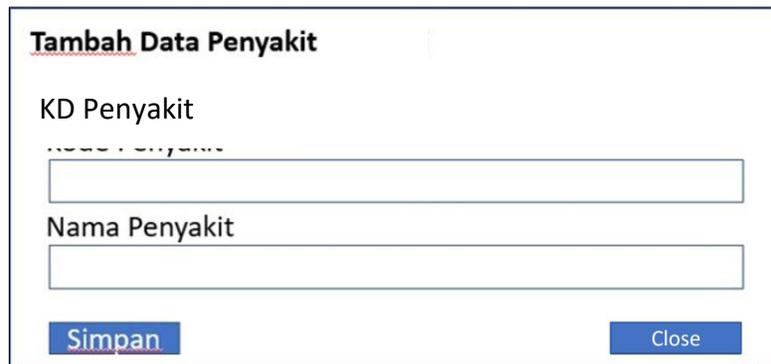
**Gambar 3.9 Rancangan Dashboard Admin**

## 5. Rancangan Data Penyakit dan Solusi



**Gambar 3.10 Rancangan Data Penyakit dan Solusi**

## 6. Rancangan Tambah Data Penyakit



**Tambah Data Penyakit**

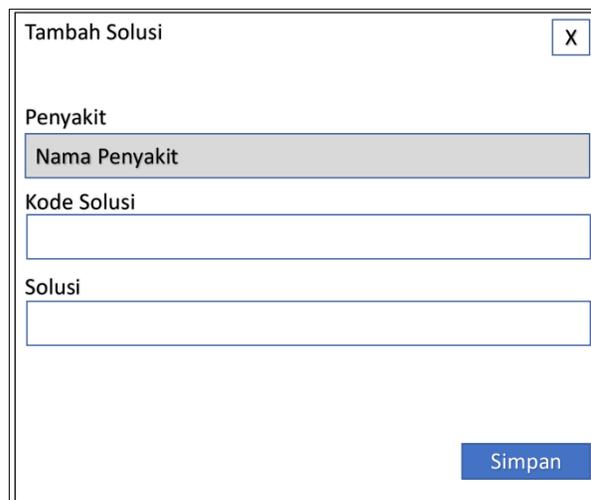
KD Penyakit

Nama Penyakit

Simpan Close

**Gambar 3.11 Rancangan Tambah Data Penyakit**

## 7. Rancangan Tambah Data Solusi



Tambah Solusi X

Penyakit

Nama Penyakit

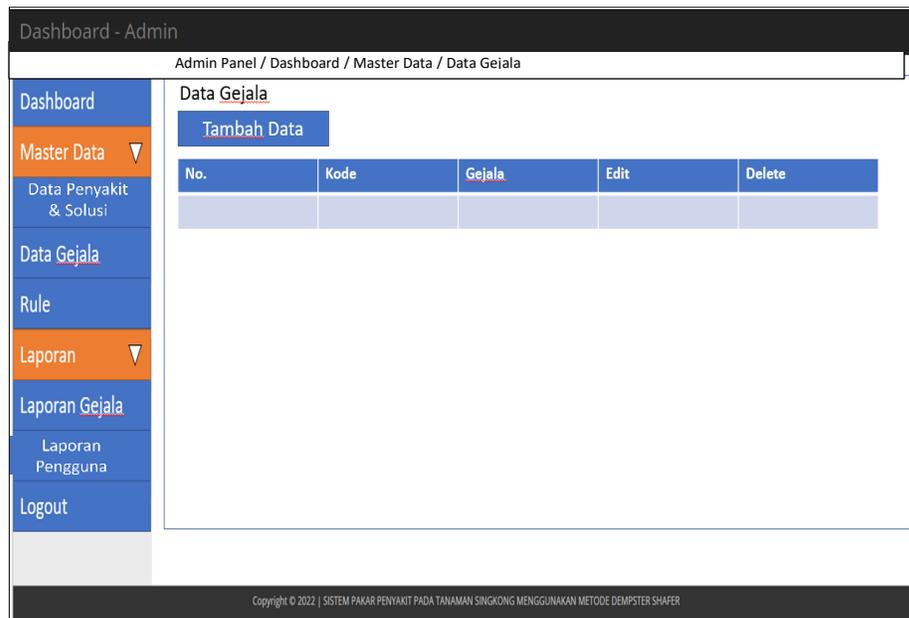
Kode Solusi

Solusi

Simpan

**Gambar 3.12 Rancangan Tambah Data Solusi**

## 8. Rancangan Data Gejala



Dashboard - Admin

Admin Panel / Dashboard / Master Data / Data Gejala

Data Gejala

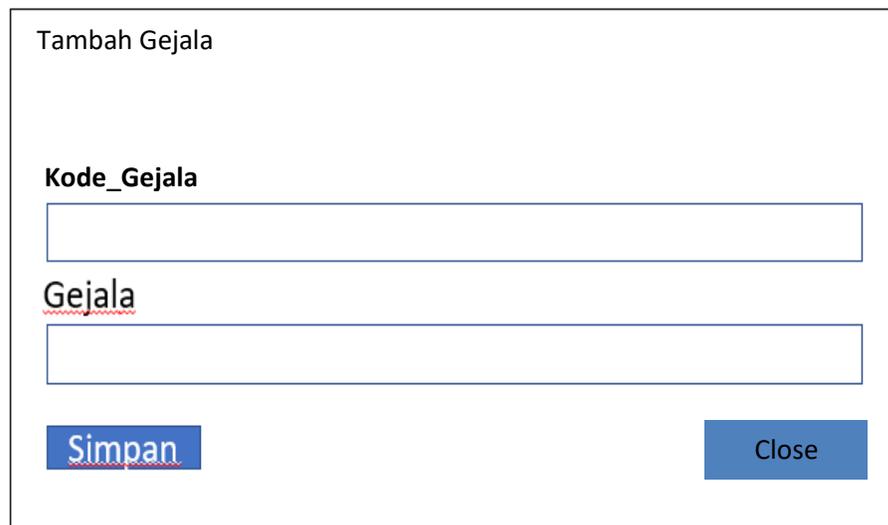
Tambah Data

No.	Kode	Gejala	Edit	Delete

Copyright © 2022 | SISTEM PAKAR PENYAKIT PADA TANAMAN SINGKONG MENGGUNAKAN METODE DEMPSTER SHAFER

**Gambar 3.13 Rancangan Data Gejala**

## 9. Rancangan Tambah Data Gejala



Tambah Gejala

**Kode\_Gejala**

**Gejala**

**Simpan** **Close**

**Gambar 3.14 Rancangan Tambah Gejala**

10. Rancangan Rule Dempster Shafer

Dashboard - Admin

Admin Panel / Rule Dempster Shafer

Dashboard

Master Data ▾

Data Penyakit & Solusi

Data Gejala

Rule

Laporan ▾

Laporan Gejala

Laporan Pengguna

Logout

Data Rule / Basis Pengetahuan

If

Gejala 1

Gejala 2

Gejala 3

Gejala 4

Then

Pilih Penyakit ▾ Nilai Belief

Reset Set Rule

No.	Kode & Nama Gejala	Penyakit 1 Edit Rule	Penyakit 2 Edit Rule	Penyakit 3 Edit Rule	Penyakit 4 Edit Rule	Penyakit 5 Edit Rule
1.	G1 , Penyakit menyerang daun tua	Cf=0.3				

Copyright © 2022 | SISTEM PANAR PENYAKIT PADA TANAMAN SINGKONG MENGGUNAKAN METODE DEMPSTER SHAFER

**Gambar 3.15 Rancangan Rule Dempster Shafer**

11. Rancangan Laporan Data Gejala

Dashboard - Admin

Admin Panel / Dashboard / Laporan / Laporan Gejala

Dashboard

Master Data ▾

Data Penyakit & Solusi

Data Gejala

Rule

Laporan ▾

Laporan Gejala

Laporan Pengguna

Logout

Laporan data Gejala Pilih Penyakit ▾

No.	Kode Gejala	Gejala	Aksi
1.	G1 ,	Penyakit menyerang daun tua	Delete

Copyright © 2022 | SISTEM PAKAR PENYAKIT PADA TANAMAN SINGKONG MENGGUNAKAN METODE DEMPSTER SHAFER

**Gambar 3.16 Rancangan Laporan Data Gejala**

## 12. Laporan Data Pengguna

Dashboard - Admin

Admin Panel / Dashboard / Laporan / Laporan Pengguna

Dashboard

Master Data ▾

Data Penyakit & Solusi

Data Gejala

Rule

Laporan ▾

Laporan Gejala

Laporan Pengguna

Logout

Laporan Pengguna

No.	Nama	Alamat	Penyakit yang dialami	Tanggal Dlagnoza	Delete	Cetak
1.	Xxx	Xxx	P1 = 30 % P2 = 20%	2022-12-01 10:00	Delete	Cetak

Copyright © 2022 | SISTEM PAKAR PENYAKIT PADA TANAMAN SINGKONG MENGGUNAKAN METODE DEMPSTER SHAFER

**Gambar 3.17 Rancangan Laporan Data Pengguna**

## 13. Rancangan Cetak Laporan Admin

<b>Data Diri Pengguna</b>	
<b>Nama :</b>	<b>alamat :</b>
<b>gejala yang terpilih :</b>	P1
P2	
<b>Hasil diagnosa :</b>	<b>1.</b>
xxxxx	<b>kesimpulan :</b>
<b>1. xxxxx</b>	<b>Solu</b> Bengkulu, xxx
<b>1 xxxxx</b>	kepala UPTD perlindungan tanaman
	pangan hortikultura dan
	perkebunan
	xxxxxx

**Gambar 3.18 Rancangan Cetak Laporan Admin**

## 14. Rancangan Logout

Are you sure want to sign out ?

**Gambar 3.19 Rancangan Logout**

## 15. Rancangan Registrasi User

Silahkan Masukkan Data Diri Anda		X
Nama Pengguna	<input type="text"/>	
Alamat	<input type="text"/>	
<input type="button" value="Lanjut"/>		<input type="button" value="Reset"/>

**Gambar 3.20 Rancangan Registrasi User**

16. Rancangan Konsultasi

PROSES IDENTIFIKASI PENYAKIT PADA TANAMAN SINGKONG	
<a href="#">Petunjuk Pengisian</a>	
<b>Gejala 1</b>	
<input type="button" value="Ya"/>	<input type="button" value="Tidak"/>
<input type="button" value="Proses Diaenosa Penvakit"/>	
<input type="button" value="Konsultasi Ulang"/>	
<input type="button" value="Cetak"/>	
<input type="button" value="Dashboard"/>	

**Gambar 3.21 Rancangan Konsultasi**

17. Rancangan Hasil Diagnosa Penyakit



### 3.6 Rancangan Pengujian

Pengujian adalah proses eksekusi suatu program dengan maksud menemukan kesalahan. Rancangan pengujian yang dilakukan menggunakan metode *Black Box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak. Pengujian ini memungkinkan analisis sistem memperoleh kumpulan kondisi input yang akan mengerjakan seluruh keperluan fungsional program. Tujuan metode ini mencari kesalahan pada:

1. Fungsi yang salah atau hilang.
2. Kesalahan pada *interface*.
3. Kesalahan pada struktur data atau akses database.
4. Kesalahan performansi dan kesalahan inisialisasi dan tujuan akhir