# RANCANG BANGUN SISTEM PAKAR DETEKSI PENYAKIT PADA TANAMAN SEMANGKA MENGGUNAKAN METODE NAIVE BAYES

## PROPOSAL TUGAS AKHIR

Disusun Untuk Memenuhi Syarat Kelulusan Program Strata 1 pada
Sekolah Tinggi Manajemen Informatika dan Komputer
(STMIK) Palangka Raya


OLEH :
ZAKARIA
NIM C1755201044
PROGRAM STUDI TEKNIK INFORMATIKA



**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER**
**(STMIK) PALANGKA RAYA**
**2021**

# RANCANG BANGUN SISTEM PAKAR DETEKSI PENYAKIT PADA TANAMAN SEMANGKA MENGGUNAKAN METODE NAIVE BAYES

## TUGAS AKHIR

Disusun Untuk Memenuhi Syarat Kelulusan Program Strata 1 pada
Sekolah Tinggi Manajemen Informatika dan Komputer
(STMIK) Palangka Raya

OLEH :
ZAKARIA
NIM C1755201044
PROGRAM STUDI TEKNIK INFORMATIKA

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER**

**(STMIK) PALANGKA RAYA**

**2021**

# LEMBAR PERNYATAAN

Yang bertanda tangan dibawah ini :

Nama Mahasiswa : **ZAKARIA**

NIM : **C1755201075**

Menyatakan bahwa Tugas Akhir dengan judul :

**RANCANG BANGUN SISTEM PAKAR DETEKSI PENYAKIT PADA TANAMAN SEMANGKA MENGGUNAKAN METODE NAIVE BAYES**

Adalah hasil karya saya dan bukan merupakan duplikasi sebagian atau seluruhnya dari karya orang lain.

Pernyataan ini dibuat dengan sebenar-benarnya secara sadar dan bertanggung jawab dan saya bersedia menerima sanksi pembatalan Tugas Akhir apabila terbukti melakukan duplikasi terhadap Tugas Akhir atau karya ilmiah lain yang sudah ada.

Palangkaraya, …… 2021

Yang membuat pernyataan,

**ZAKARIA**

**PERSETUJUAN**

**RANCANG BANGUN SISTEM PAKAR DETEKSI PENYAKIT PADA TANAMAN SEMANGKA MENGGUNAKAN METODE NAIVE BAYES**

Tugas Akhir ini telah disetujui untuk diujikan
pada Tanggal  September 2021

| Dosen Pembimbing I, | Dosen Pembimbing II, |
|---|---|
| Hotmian Sitohang, M.Kom. | Sherly Jayanti, ST., M.Cs. |
| NIK. 198503282008002 | NIK. 198501102012004 |

Mengetahui
Ketua STMIK Palangka Raya,

Suparno, M. Kom
NIK. 196901041995105

**PENGESAHAN**

**RANCANG BANGUN SISTEM PAKAR DETEKSI PENYAKIT PADA TANAMAN SEMANGKA MENGGUNAKAN METODE NAIVE BAYES**

Tugas Akhir ini telah Diuji, Dinilai dan Disahkan Oleh Tim Penguji Pada Tanggal ……. 2021

Tim Penguji :

1. : ……..
2. : ……..
3. Sam'ani,S.T.,M.Kom : ……
   Ketua

2. Hotmian Sitohang, M.Kom : ……
   Sekretaris

3. Sherly Jayanti, S.T., M.Cs. : ……
   Anggota

# MOTTO DAN PERSEMBAHAN

*Takut akan Tuhan adalah*
*Permulaan pengetahuan*
*Tetapi orang bodoh*
*Menghina hikmat dan didikan.*

Tugas Akhir ini saya persembahkan untuk :

- Orang Tua tercinta, yang tidak kenal lelah berdo'a, mendidik, mendukung, dan membesarkan saya hingga sekarang
- Kakak saya Fery Suharta, Novendry, Bayu Saputra, dan Monica yang telah memberikan dukungan kepada saya sehingga saya dapat menyelesaikan tugas akhir ini tepat waktu.
- Ling Ling Wei yang selalu memberikan semangat tiada henti dan memberikan saya dukungan dalam menyelesaikan tugas akhir ini
- Teman-teman Teknik Informatika B Angkatan 2017, yang tidak kenal lelah untuk saling membantu dan memberikan dukungannya.
- Dosen-dosen STMIK Palangka Raya, yang telah memberikan ilmu serta membimbing saya dari awal kuliah sampai pada akhirnya saya lulus dan mendapatkan gelar sarjana komputer (S.Kom)

# INTISARI

**ZAKARIA,** C1755201044, 2021, *Rancang Bangun Sistem Pakar Deteksi Penyakit Pada Tanaman Semangka Menggunakan Metode Naïve bayes,* Pembimbing I Hotmian Sitohang, M.Kom. Pembimbing II Sherly Jayanti, ST., M.Cs.

Permasalahan dalam penelitian ini adalah dimana para Petani sering mengalami kerugian dikarenakan tanaman semangka yang sering terkena penyakit yang mengakibatkan produksi buah seamangka menurun. Hal ini disebabkan karena keterbatasan jumlah pakar atau ahli yang dapat menentukan penyakit dari tanaman buah semangka dan cara penanggulangannya. Oleh karena itu, bantuan untuk pengganti pakar penyakit pada semangka sangat dibutuhkan para petani untuk membantu menangani penyakit pada semangka dengan tepat.

Sistem pakar merupakan suatu sistem yang dirancang untuk dapat menirukan keahlian seorang pakar dalam menjawab pertanyaan dan memecahkan suatu masalah. penelitian ini merancang sebuah perangkat lunak sistem pakar dengan menggunakan metode *Naive Bayes* sehingga dapat menentukan probabilitas penyakit berdasarkan gejala dengan aturan yang didapat dari pakar.

Hasil akhir dari penelitian ini adalah berupa Aplikasi Sistem Pakar Deteksi Penyakit Pada Tanaman Seamangka Menggunakan Metode Naïve Bayes untuk membantu Petani mendeteksi penyakit pada tanaman semangka.

**KATA KUNCI**: Sistem Pakar, *Naive Bayes*,*Diagnosis* Penyakit Tanaman Semangka dan *Web*

## ABSTRACT

**ZAKARIA,** C1755201044, 2021, *Rancang Bangun Sistem Pakar Deteksi Penyakit Pada Tanaman Semangka Menggunakan Metode Naïve bayes*, Pembimbing I Hotmian Sitohang, M.Kom. Pembimbing II Sherly Jayanti, ST., M.Cs.

The problem in this study is that farmers often suffer losses due to watermelon crops that are often affected by diseases that result in decreased fruit production. This is due to the limited number of experts or experts who can determine the disease of the watermelon fruit crop and how to deal with it. Therefore, assistance for the replacement of disease experts in watermelon is needed by farmers to help deal with diseases in watermelon appropriately.

An expert system is a system designed to mimic an expert's expertise in answering questions and solving problems. This study designed an expert system software using the Naive Bayes method so as to determine the probability of disease based on symptoms by the rules obtained from experts.

The final result of this study is the Application of Expert System of Disease Detection in Seamangka Plants Using Naïve Bayes Method to help Farmers detect diseases in watermelon plants.



KEYWORDS: System Expert, Naive Bayes,Watermelon Plant Disease Diagnosis and Web

# KATA PENGANTAR

Segala puji dan syukur bagi Allah atas ridho-Nya, penulis dapat menyelesaikan Proposal Tugas Akhir ini dengan baik. Laporan Tugas Akhir ini ditujukan untuk memenuhi salah satu syarat kelulusan program studi Teknik Informatika di STMIK Palangkaraya.

Penulis memahami tanpa bantuan, doa, dan bimbingan dari semua orang akan sangat sulit untuk menyelesaikan laporan ini. Maka dari itu penulis ingin mengucapkan terima kasih sebesar-besarnya atas dukungan dan kontribusi kepada;

1. Ibu Hotmian Sitohang, M.Kom, selaku dosen pembimbing I yang telah membimbing selama penyusunan laporan ini;
2. Sherly Jayanti, S.T., M.Cs, selaku dosen pembimbing II yang telah membimbing selama penyusunan laporan ini;
3. Ibu Sri Agustini, SP, selaku Pakar tanaman buah semangka di BPTP ;

Semoga Laporan ini dapat bermanfaat bagi penulis sendiri dan pembaca.

Palangkaraya, …….. 2021

Penulis

# DAFTAR ISI

DAFTAR ISI ..... viii

DAFTAR TABEL ..... xi

DAFTAR GAMBAR ..... xii

DAFTAR LAMPIRAN ..... xiv

BAB I ..... 1

PENDAHULUAN ..... 1

1.1 Latar Belakang Masalah ..... 1

1.2 Perumusan Masalah ..... 2

1.3 Batasan Masalah ..... 2

1.4 Tujuan dan Manfaat ..... 2

1.5 Sistematika Penulisan ..... 4

BAB II ..... 6

LANDASAN TEORI ..... 6

2.1 Tinjauan Pustaka ..... 6

2.1.1 Kajian Penelitian Yang Relevan ..... 6

2.2 Kajian Teori ..... 11

2.2.1 Sistem Pakar ..... 11

2.2.2 .Naive Bayes Classifier ..... 13

2.2.3 Tanaman Semangka ..... 16

2.2.4 Website ..... 17

2.2.5 Flowchart ..... 18

2.2.6 Data Flow Diagram (DFD) ..... 21

2.2.7 Entity Relationship Diagram (ERD) ..... 22

2.2.8 Basis Data (Database) ..... 23

2.2.9 MySQL ..... 24

2.2.10 XAMPP ..... 25
2.2.11 PHP (*PHP Hypertext Processor)* ..... 25
2.2.12 Web Browser dan HTML ..... 26
2.2.13 Sublime Text ..... 26
2.2.14 Blackbox Testing ..... 27
BAB III ..... 28
METODE PENELITIAN ..... 28
3.1 Lokasi Penelitian ..... 28
3.2 Teknik Pengumpulan Data ..... 28
3.3 Analisis ..... 29
3.3.1 Metodologi Penelitian ..... 29
3.3.2 Analisis Proses ..... 31
3.3.3 Analisis Kelemahan Sistem ..... 32
3.3.4 Analisis Kebutuhan ..... 34
3.3.5 Analisis Kelayakan Sistem ..... 43
3.4 Desain Sistem ..... 44
3.4.1 Desain Proses ..... 44
3.4.2 Desain Perangkat Lunak ..... 45
3.4.3 Desain Basis Data ..... 50
3.4.4 Desain Tabel ..... 51
BAB IV ..... 53
HASIL DAN PEMBAHASAN ..... 53
4.1 Hasil ..... 53
4.1.1 Impelentasi Program ..... 53
4.1.2 Pengujian Sistem dan Uji Coba ..... 53
4.1.3 Manual Program ..... 55

4.1.4 Manual Instalasi ..... 56
4.2 Pembahasan ..... 57
4.2.1 Pembahasan Listing Program ..... 57
4.2.2 Pembahasan *Interface* / Antarmuka Program ..... 70
BAB V ..... 83
5.1 Kesimpulan ..... 83
5.2 Saran ..... 83
DAFTAR PUSTAKA ..... 84

# DAFTAR TABEL

Tabel 1.Kajian Penelitian Yang Relevan ..... 8
Tabel 2.Simbol *Flow Direction* ..... 19
Tabel 3.Simbol *Processing* ..... 19
Tabel 4. Simbol *Input / Output* ..... 20
Tabel 5.Simbol Data Flow /Diagram (DFD) Menurut Yourdan /De Marco ..... 22
Tabel 6.*Simbol-simbol Entity Relational Database (ERD)* ..... 23
Tabel 7.Perbandingan *Performance*(kinerja) ..... 33
Tabel 8.Perbandingan *Information*(Informasi) ..... 33
Tabel 9.Perbandingan *Economy*(Ekonomi) ..... 33
Tabel10.Perbandingan *Control*(Pengendalian) ..... 33
Tabel 11.Perbandingan *Efficiency*(Efisiensi) ..... 34
Tabel 12.Perbandingan *Service*(Layanan) ..... 34
Tabel 13.Spesifikasi Perangkat Keras ..... 35
Tabel 14.Spesifikasi Perangkat Lunak ..... 36
Tabel 15.Data Penyakit Tanaman Semangka ..... 36
Tabel 16.Daftar Gejala ..... 37
Tabel 17.Basis Pengetahuan (Rule) ..... 38
Tabel 18.Desain Struktur Tabel Akun ..... 51
Tabel 19.Desain Struktur Tabel Gejala ..... 51
Tabel 20.Desain Struktur Tabel Penyakit ..... 52
Tabel 21.Desain Struktur Tabel Relasi ..... 52
Tabel 22.Hasil Pengujian BlackBox Pengunjung ..... 53
Tabel 23.Hasil Pengujian BlackBox Admin ..... 54

# DAFTAR GAMBAR

Gambar 1. *Rapid Application Development* (Sukamto & Shalahudin, 2016:13)..... 29
Gambar 2.Flowchart Sistem Lama..... 32
Gambar 3.Flowchart cara kerja sistem pakar..... 35
Gambar 4.Sistem Baru Admin..... 41
Gambar 5.Sistem Baru User ..... 42
Gambar 6.Diagram Konteks ..... 44
Gambar 7.Halaman Login..... 45
Gambar 8.Halaman Beranda..... 46
Gambar 9.Halaman Kelola Akun..... 46
Gambar 10.Halaman Kelola Penyakit..... 47
Gambar 11.Halaman Kelola Gejala ..... 47
Gambar 12.Halaman Bernada (User)..... 48
Gambar 13.Halaman Konsultasi(User)..... 48
Gambar 14.Halaman Hasil Analisis..... 49
Gambar 15.Halaman Cetak Hasil Konsultasi ..... 49
Gambar 16.Entity Relationship Diagram (ERD) ..... 50
Gambar 17.Beranda Pengunjung ..... 70
Gambar 18.Fitur Konsultasi..... 71
Gambar 19.Hasil Analisis Lanjutan(2) ..... 72
Gambar 20.Hasil Analisis Lanjutan(3) ..... 73
Gambar 21.Fitur Sign In ..... 73
Gambar 22.Sign In Gagal ..... 74
Gambar 23.Sign In Berhasil..... 74
Gambar 24.Fitur Beranda Admin..... 75
Gambar 25.Fitur Kelola Akun ..... 75
Gambar 26.Kelola Data Penyakit..... 76
Gambar 27. Kelola Data Penyakit (Tambah)..... 77
Gambar 28.Kelola Data Penyakit (Ubah) ..... 77
Gambar 29.Kelola Data Penyakit (Hapus)..... 78

Gambar 30.Kelola Data Penyakit (Export) ..... 78
Gambar 31.Kelola Data Gejala ..... 79
Gambar 32. Kelola Data Gejala (Tambah) ..... 79
Gambar 33.Kelola Data Gejala (Ubah) ..... 80
Gambar 34.Kelola Data Gejala (Hapus) ..... 80
Gambar 35.Kelola Data Gejala (Export) ..... 81
Gambar 36.Informasi SignOut ..... 81

## DAFTAR LAMPIRAN

Lampiran 1 Surat Tugas Dosen Pembimbing

Lampiran 2 Surat Izin Penelitian

Lampiran 3 Lembar Konsultasi Bimbingan

Lampiran 4 Lembar Wawancara

Lampiran 5 Surat Tugas Penguji Seminar Tugas Akhir

Lampiran 6 Berita Acara Seminar Proposal

Lampiran 7 Daftar Hadir Peserta Seminar Proposal Tugas Akhir

# BAB I

# PENDAHULUAN

## 1.1 Latar Belakang Masalah

Indonesia adalah salah satu negara penghasil buah tropis tak terkecuali diwilayah Kalimantan Tengah yang memiliki keanekaragaman dan keunggulan cita rasa yang cukup baik, salah satunya adalah buah semangka. Semangka (Citrullus vulgaris, Schard) merupakan salah satu buah yang sangat digemari masyarakat Indonesia karena rasanya yang manis, renyah dan kandungan airnya yang banyak sehingga para petani mulai menyadari bahwa membudidaya tanaman semangka memang sangat menjanjikan dan dapat memberikan keuntungan yang besar. Hal itu tentu saja jika tanaman semangka dibudidayakan dengan cara yang baik dan benar. Namun hingga saat ini, para petani sering mengalami kerugian dikarenakan tanaman semangka yang sering terkena penyakit yang mengakibatkan produksi buah seamangka menurun. Hal ini disebabkan karena keterbatasan jumlah pakar atau ahli yang dapat menentukan penyakit dari tanaman buah semangka dan cara penanggulangannya. Oleh karena itu, bantuan untuk pengganti pakar penyakit pada semangka sangat dibutuhkan para petani untuk membantu menangani penyakit pada semangka dengan tepat.

Sistem pakar merupakan suatu sistem yang dirancang untuk dapat menirukan keahlian seorang pakar dalam menjawab pertanyaan dan memecahkan suatu masalah. Sistem pakar akan memberikan pemecahan suatu masalah yang diperoleh berdasarkan dialog dengan pemakai. Dengan bantuan

sistem pakar seseorang yang bukan pakar/ahli dapat menyelesaikan masalah serta mengambil keputusan yang biasanya dilakukan oleh seorang pakar.

Berdasarkan pada permasalahan yang ada, maka penelitian ini merancang sebuah perangkat lunak sistem pakar dengan menggunakan metode *Naive Bayes* untuk menentukan probabilitas yang ada berdasarkan fakta atau aturan dari sistem pakar tersebut.

## 1.2 Perumusan Masalah

Berdasarkan latar belakang tersebut diatas dapat dirumusan masalah yaitu bagaimana merancang dan membangun sistem pakar untuk mendeteksi penyakit pada tanaman semangka dengan metode *Naive Bayes* menggunakan bahasa pemrograman PHP dan database MySQL

## 1.3 Batasan Masalah

Batasan masalah dalam penelitian ini adalah :

a. Studi kasus penelitian dilakukan di daerah Palangka Raya.
b. Penentuan penyakit tanaman semangka berdasarkan gejala yang diinputkan user.
c. Mekanisme penalaran/penelusuran sistem pakar yang dikembangkan menggunakan metode *Naïve Bayes.*
d. Program dibuat dengan menggunakan bahasa PHP dan databse MySQL.

## 1.4 Tujuan dan Manfaat

### a. Tujuan

Tujuan penelitian ini adalah untuk membangun sebuah sistem pakar yang dapat mendeteksi penyakit pada tanaman semangka berbasis web

serta mampu memberikan informasi mengenai masalah yang ada beserta penanganan awal yang dapat dilakukan.

**b. Manfaat**

Manfaat yang diharapkan dari penelitian ini adalah :

1. Bagi Pengguna yaitu dapat digunakan sebagai alat bantu dalam penanganan penyakit pada tanaman semangka berbasis web.
2. Bagi Akademik yaitu dapat dijadikan sebagai sarana atau acuan penelitian bagi mahasiswa yang ingin menambah wawasan lebih jauh tentang penggunaan inferensi dengan metode penilai ketidakpastian *naïve bayes* dengan berbasis PHP dan MySQL untuk membuat sebuah sistem pakar.
3. Bagi Penulis yaitu mendapatkan pengetahuan dan pemahaman tentang penyakit pada tanaman semangka, memberikan tambahan pengetahuan mengenai sistem pakar untuk penanganan penyakit pada tanaman semangka berbasis web dan sebagai literatur pada perpustakaan kampus dan menerapkan ilmu pengetahuan tentang program PHP dan MySQL yang didapat dari perkuliahan dengan merancang suatu sistem pakar untuk penanganan penyakit pada tanaman semangka berbasis web.

## 1.5 Sistematika Penulisan

Sistematika dalam penulisan Tugas Akhir menggunakan beberapa bab yang tersusun sebagai berikut :

**BAB I PENDAHULUAN**

Pada Bab ini akan diuraikan mengenai latar belakang masalah, perumusan masalah, batasan masalah, tujuan dan manfaat serta sistematika penulisan yang digunakan selama penelitian.

**BAB II LANDASAN TEORI**

Pada bab ini berisikan landasan teori yang nantinya akan menguraikan teori-teori pendukung penelitian dan menjadi dasar bagi pemecahan masalah pada proses penelitian dan perancangan mengenai sistem pakar.

**BAB III METODE PENELITIAN**

Pada bab ini berisi pembahasan mengenai metodologi penelitian yang digunakan untuk mengevaluasi masalah yang berlaku dalam menganalisis, merancang dan membangun sebuah sistem pakar. Pada bab ini juga akan dibahas mengenai Metodoloi Penelitian yang digunakan serta proses desain sistem, contohnya perhitungan metode *Naive Bayes*, pembuatan tampilan sistem dan desain lainnya.

**BAB IV HASIL DAN PEMBAHASAN**

Pada bab ini akan menjelaskan tentang tahapan dari implementasi metode *Naive Bayes* dalam pembuatan sistem pakar berbasis web, tahapan langkah penggunaan serta testing program.

# BAB V SIMPULAN DAN SARAN

Pada bab ini akan menguraikan kesimpulan beserta saran yang didapatkan dari awal pembuatan sampai terselesaikannya program

# BAB II

# LANDASAN TEORI

## 2.1 Tinjauan Pustaka

### 2.1.1 Kajian Penelitian Yang Relevan

Dalam melakukan suatu penelitian, diperlukan data dukung berupa hasil penelitian yang telah ada sebelumnya yang berkaitan dengan penelitian yang akan dilakukan. Berikut adalah hasil penelitian relevan yang telah ada sebelumnya dan akan dibandingkan dengan penelitian yang akan dilakukan oleh penulis.

Penelitian oleh Sandy Kosasi (2014) tentang sistem pakar yang dirancang untuk mendiagnosa hama dan penyakit pada tanaman semangka menggunakan metode certainty factor dalam proses perhitungan nilai ketidakpastiannya.

Penelitian oleh Mugirahayu Handayani (2016) tentang Sistem Pakar Diagnosa Penyakit Tanaman Semangka Menggunakan Metode *Dempster Shafer* Berbasis Web. Sistem dirancang untuk mendeteksi penyakit tanaman semangka berdasarkan pada tingkat kepercayaan (probabilitas) untuk sebuah pertanyaan tertentu.

Penelitian oleh Tika Wulandari (2018) tentang Perancangan Sistem Pakar Deteksi Pertumbuhan Tanaman Semangka Berbasis Website Dengan *Certainty Factor*. Sistem pakar dirancang untuk mendeteksi pertumbuhan tanaman semangka dengan nilai kriteria yang memenuhi sarat yang telah ditentukan oleh pakar dengan

menggunakan metode Certainty Factor dalam menentukan jenis penyakit yang menghambat pertumbuhan tanaman semangka dengan tingkat akurasi yang baik.

Penelitian oleh Alan Wiedy Mariana (2019) tentang Sistem pakar yang dirancang untuk membantu para petani dalam mendeteksi penyakit tanaman semangka dan solusi penanganannya dengan menggunakan metode Certainty Factor.

Penelitian oleh M. Ibnu Pati (2020) tentang Sistem pakar yang dirancang untuk mendiagnosa gejala-gejala dan memberikan sebuah kesimpulan dan solusi dari penyakit dan hama tanaman semangka dengan menggunakan metode Forward Chaining.

**Tabel 1.Kajian Penelitian Yang Relevan**

| No | Nama / Penulis | Topik Penelitian | Metode | Hasil | Perbedaan |
|---|---|---|---|---|---|
| 1 | Sandy Kosasi /2014 | Sistem Pakar Diagnosa Hama dan Penyakit Tanaman Semangka Menggunakan Metode Certainty Factor | Certainty Factor | Sistem dirancang untuk mendiagnosa hama dan penyakit pada tanaman semangka menggunakan metode certainty factor dalam proses perhitungan nilai ketidakpastiannya. | Perbedaan penelitian tersebut dengan penulis yaitu metode yang digunakan dalam penelitian dimana nilai keyakinan ditentukan berdasarkan dari bukti / fakta yang telah ada. |
| 2 | Mugirahayu Handayani /2016 | Sistem Pakar Diagnosa Penyakit Tanaman Semangka Menggunakan Metode Dempster Shafer Berbasis Web | Metode Dempster-Shafer | Sistem dirancang untuk mendeteksi penyakit tanaman semangka berdasarkan pada tingkat kepercayaan (probabilitas) untuk sebuah pertanyaan tertentu. | Perbedaan penelitian tersebut dengan penulis yaitu metode yang digunakan dalam penelitian ini memberikan tingkat kepercayaan/probabilitasnya, |

| | | | | | |
|---|---|---|---|---|---|
| 3 | Tika Wulandari / 2018 | Perancangan Sistem Pakar Deteksi Pertumbuhan Tanaman Semangka Berbasis Website Dengan Certainty Factor | *System Development Life Cycle* (SDLC) | Sistem pakar dirancang untuk mendeteksi pertumbuhan tanaman semangka dengan nilai kriteria yang memenuhi sarat yang telah ditentukan oleh pakar dengan menggunakan metode Certainty Factor dalam menentukan jenis penyakit yang menghambat pertumbuhan tanaman semangka dengan tingkat akurasi yang baik. | Perbedaan penelitian tersebut dengan penulis yaitu metode yang digunakan dalam penelitian dimana nilai ketidakpastian ditentukan berdasarkan dari bukti / fakta yang ada. |
| 4 | Alan Wiedy Mariana / 2019 | Sistem Pakar Diagnosa Penyakit Tanaman Semangka Menggunakan Metode Certainty Factor | *Certaitny Factor* | Sistem pakar dirancang untuk membantu para petani dalam mendeteksi penyakit tanaman semangka dan solusi penanganannya. | Perbedaan penelitian tersebut dengan penulis yaitu metode yang digunakan dalam penelitian dimana nilai keyakinan ditentukan berdasarkan dari bukti / fakta yang telah ada. |
| 5 | M. Ibnu Pati / 2020 | Sistem Pakar dengan Metode Forward | *Forward Chaining* | Sistem pakar ini dirancang untuk mendiagnosa gejala-gejala dan | Perbedaan penelitian tersebut dengan penulis yaitu metode |

| | | | | | |
|---|---|---|---|---|---|
| | | Chaining untuk Diagnosa Penyakit dan Hama Tanaman Semangka | | memberikan sebuah kesimpulan dan solusi dari penyakit dan hama tanaman semangka. | yang digunakan dalam penelitian ini mencari aturan inteferensi sampai menemukan satu klausa yang benar. |
| | Agus Sifaunajah ,Tholib Hariono, Moch.Chumaidi / 2018 | *Application of Certainty factor on The Design of Expert System of Detection of Plant Destruction Organisms (OPT) Watermelon* | *Certaitny Factor* | Sistem pakar ini dirancang untuk mengidentifikasi,organisme pengganggu tanaman semangka dengan memberikan aturan-aturan gejala serta memberikan nilai pada setiap gejala yang diperoleh dari seorang pakar. Semua nilai yang didapatkan diproses menggunakan algoritma certainty factor. | Perbedaan penelitian tersebut dengan penulis yaitu sistem pakar yang dirancang penulis digunakan untuk mendeteksi penyakit dan hama pada tanaman semangka menggunakan algoritma naïve bayes. |

## 2.2 Kajian Teori

### 2.2.1 Sistem Pakar

Menurut Turban (2001:402), sistem pakar adalah sebuah sistem yang menggunakan pengetahuan manusia dimana pengetahuan tersebut dimasukkan ke dalam sebuah komputer dan kemudian digunakan untuk menyelesaikan masalah-masalah yang biasanya membutuhkan kepakaran atau keahlian manusia.

Dalam Sutojo, dkk (2011:159), sistem pakar merupakan cabang dari *Artificial Intelligent* (AI) yang cukup tua karena sistem ini mulai dikembangkan pada pertengahan tahun 1960. Sistem pakar yang muncul pertama kali adalah *General-Purpose Problem Solver* (GPS) yang dikembangkan oleh Newel dan Simon. Sampai saat ini sudah banyak sistem pakar yang dibuat, seperti MYCIN untuk diagnosis penyakit, DENDRAL untuk mengidentifikasi struktur molekul campuran yang tidak dikenal, XCON & XSEL untuk membantu konfigurasi sistem komputer besar, SOPHIE untuk analisis sirkuit elektronik, Prospector digunakan di bidang geologi untuk membantu dan menemukan deposit, FOLIO digunakan untuk membantu memberikan keputusan bagi seorang manager dalam stok dan investasi, DELTA dipakai untuk pemeliharaan lokomotif listrik diesel, dan sebagainya.

Istilah sistem pakar berasal dari istilah *knowledge-based expert system.* Istilah ini muncul karena untuk memecahkan masalah, sistem

pakar menggunakan pengetahuan seorang pakar yang dimasukkan ke dalam komputer. Seseorang yang bukan pakar menggunakan sistem pakar untuk meningkatkan kemampuan pemecahan masalah, sedangkan seorang pakar menggunakan sistem pakar untuk *knowledge assistant*.

Sistem pakar itu sendiri merupakan aplikasi berbasis komputer yang digunakan untuk menyelesaikan masalah sebagaimana yang dipikirkan oleh pakar. Pakar yang dimaksud adalah orang yang mempunyai 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 kesimpulan terhadap penyakit tersebut. Tidak semua orang dapat mengambil keputusan mengenai diagnosis dan memberikan kesimpulan terhadap suatu penyakit. Contoh lain, montir adalah seorang yang punya keahlian dan pengalaman dalam menyelesaikan kerusakan mesin motor/mobil, psikolog adalah orang yang ahli dalam memahami kepribadian seseorang, dan lain-lain.

Sistem pakar mencoba memecahkan masalah yang biasanya hanya bisa dipecahkan oleh seorang pakar, dan akan dipandang berhasil jika mampu mengambil keputusan seperti yang dilakukan oleh pakar aslinya, baik dari sisi proses pengambilan keputusan maupun hasil keputusan yang diperoleh.

### 2.2.2 .Naive Bayes Classifier

*Naive-Bayes Classifier* adalah algoritma klasifikasi dengan teknik prediksi berbasis probabilistik sederhana yang dikembangkan dari teorema Bayes dengan menggunakan asumsi independensi yang kuat (Prasetyo, 2012). Meskipun merupakan salah satu algoritma klasifikasi tertua, akan tetapi karena sifatnya yang simpel (hanya perkalian dan pembagian) menjadikannya sebagai salah satu algoritma yang paling umum digunakan.

Terdapat dua model *Naive-Bayes* yang umum digunakan, yaitu model *Multivariete Bernoulli* dan model *Multinomial*. Pada model *Multivariete Bernoulli*, frekuensi kemunculan token data tidak berpengaruh pada hasil klasifikasi, yang diperhatikan hanyalah apakah token tersebut muncul dalam dokumen atau tidak. Sebaliknya, pada model *Multinomial* frekuensi kemunculan token data mempengaruhi hasil klasifikasi. Secara empiris model *Multinomial* memiliki performa dan akurasi yang lebih tinggi dalam melakukan klasifikasi teks, terutama jika jumlah kosakata yang digunakan cukup besar (McCallum dan Nigam et al, dikutip dalam Raschka, 2014).

Model *Multinomial* sendiri memiliki dua jenis model, yaitu model Normal *Multinomial* dan model *Boolean Multinomial*. Pada model *Normal Multinomial*, frekuensi kemunculan token data dihitung seperti biasa. Sedangkan pada model *Boolean Multinomial*, frekuensi kemunculan token data di setiap dokumen dihitung secara biner, yaitu

ada (bernilai 1) atau tidak ada (bernilai 0).Berdasarkan penelitian Metsis, Androutsopoulos dan Paliouras (2006), model *Boolean Multinomial* dapat memberikan hasil yang lebih akurat dibandingkan model *Normal Multinomial* walaupun dengan selisih yang relatif kecil.

Naïve Bayes merupakan pengklasifikasi probabilitas sederhana berdasarkan pada teorema Bayes. Keuntungan dari klasifikasi adalah bahwa ia hanya membutuhkan sejumlah kecil data pelatihan untuk memperkirakan parameter (sarana dan varians dari variabel) yang diperlukan untuk klasifikasi. Karena variabel independen diasumsikan, hanya variasi dari variabel untuk masing-masing kelas harus ditentukan, bukan seluruh matriks kovarians. Dalam prosesnya, Naïve Bayes mengasumsikan bahwa ada atau tidaknya suatu fitur pada suatu kelas tidak berhubungan dengan ada atau tidaknya fitur lain di kelas yang sama.

Teorema Bayes Menyatakan :

$$\mathbf{P(B|A)} = \frac{\mathbf{P(A|B)}\boldsymbol{P(B)}}{\boldsymbol{P(A)}} \qquad \mathbf{(1)}$$

Dimana :

P(B|A) = Peluang B jika diketahui keadaan jenis penyakit A

P(B|A) = Peluang evidence A jika diketahui hipotesis B

P(B) = Probabilitas hipotesis B tanpa memandang evidence apapun

P(A) = Peluang evidence penyakit A

Metode *Naive Bayes Classifier* Merupakan model penyederhanaan dari *teorema bayes* yang cocok untuk mengidentifikasi tipe kepribadian siswa. Berikut ini rumus persamaan pada *naïve bayes classifier* (Putra, Aksara, & Ramadhan, 2014):

$$\boldsymbol{V_{MAP} = argmax_{vjeV} \frac{P(a_1, a_2 \ldots . a_n, 1a|v_j).P(v_j)}{P(a_1, a_2 \ldots . a_n)}} \quad \textbf{(2)}$$

$V_{MAP}$ = Probabilitas tertinggi.

$P(v_j)$ = Peluang jenis penyakit ke

$P(a_1\ a_2 \ldots\ a_n\ |v_j)$ = Peluang atribut-atribut (inputan) jika diketahui keadaan $v_j$.

$P(a_1\ a_2 \ldots\ a_n)$ = Peluang atribut-atribut (inputan)

Karena nilai $P(a_1\ a_2 \ldots\ a_n)$ nilainya konstan untuk semua vj sehingga persamaan ini dapat ditulis :

$$\boldsymbol{V_{MAP} = argmax_{vjeV}\ P(v_j|\ a_1, a_2 \ldots . a_n|v_j)P(v_j)} \quad \textbf{(3)}$$

VMAP = Probabilitas tertinggi.

P(vj) = Peluang jenis penyakit ke j

P(a1 a2... an|vj)= Peluang atribut-atribut (inputan) jika diketahui keadaan vj

Untuk menghitung P(a1 a2... an|vj)P(vj) semakin sulit karena jumlah gejala P(a1 a2... an|vj)P(vj) bisa jadi sangat besar. Hal ini disebabkan jumlah gejala tersebut sama dengan jumlah semua kombinasi gejala dikali dengan jumlah kategori yang ada.

Perhitungan Naïve bayes classifier adalah menghitung P(ai|vj) dengan rumus :

$$\boldsymbol{P(a_i|v_j) = \frac{n_c + mp}{n+m}} \quad \textbf{(4)}$$

Dimana :

nc = jumlah record pada data learning yang v = vj dan a = ai

p = 1/ banyaknya jenis class / penyakit

m = jumlah parameter / gejala

n = jumlah record pada data learning yang v = vj / tiap class

### 2.2.3 Tanaman Semangka

Menurut Noradilla,(2015) Semangka merupakan tanaman dari famili Cucurbitaceae (labu-labuan) yang bersifat semusim. Semangka merupakan salah satu jenis buah hortikultura yang berasal dari Afrika. Buah semangka satu kerabat dengan buah melon, mentimun, dan labu-labuan yang memiliki rasa manis dan segar sehingga banyak dikonsumsi oleh berbagai lapisan masyarakat. Buah semangka yang kaya zat gizi dan kaya akan kandungan airnya sangat cocok untuk dikonsumsi di daerah tropis

Buah semangka diketahui mengandung zat-zat tertentu yang cukup efektif dalam membunuh sel-sel kanker, yaitu zat yang mampu menghidupkan aktivitas fungsi sel darah putih yang mampu meningkatkan sistem kekebalan. Hasil percobaan menunjukkan bahwa semangka mengandung zat-zat yang dapat menstimulir phagocyte, yaitu suatu sel darah yang mampu melindungi s istem darah dari infeksi dengan cara menyerap mikroba untuk mematikan sel-sel penyebab penyakit kanker. Kandungan kalori buah semangka sangat rendah sehingga semangka dapat berfungsi sebagai diuretik. Buah semangka

mengandung pigmen karotenoid jenis flavonoid yang memberikan warna daging buah merah atau kuning (Laksono, 2018).

Diketahui bahwa Semangka memiliki bentuk yang cukup besar dengan bentuk bulat atau lonjong dan diameter hingga 20 cm. Secara fisik sebenarnya buah semangka ini memiliki bentuk yang beragam bahkan dari hasil penelitian. Masih ada buah berbentuk kotak ataupun kubus yang populer terutama di jepang.Apabila dilihat kulit buah semangka menyelimuti daging buah berdaging dan tebal permukaan luarnya terkesan licin dan daging kulit yang memiliki warna putih disebut sebagai albedo, cukup untuk melindungi buah yang ada di dalam buah (Sri Agustini).

### 2.2.4 Website

Menurut Yuhefizar (2013:2) "Website adalah keseluruhan halamanhalaman web yang terdapat dalam sebuah domain yang mengandung informasi. Sebuah website biasanya dibangun atas banyak halaman web yang saling berhubungan. Hubugan antara satu halaman web dengan halaman web yang lainnya disebut dengan hyperlink, sedangkan teks yang dijadikan media penghubung disebut hypertext".

Menurut Hikmah, dkk (2015:1) "Website atau situs dapat diartikan sebagai kumpulan halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan/atau gabungan dari semuanya, baik yang bersifat statis

maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman".

Berdasarkan pengertian para ahli diatas dapat disimpulkan bahwa website adalah kumpulan dari kesuluruhan halaman-halaman web yang berisi sebuah data atau informasi baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

### 2.2.5 Flowchart

Menurut Ladjamudin (2013:263), "Flowchart adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. Flowchart merupakan cara penyajian dari suatu algoritma".

a. Tujuan Membuat Flowchart :

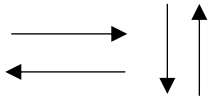
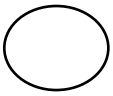
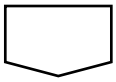
   1. Menggambarkan suatu tahapan penyelesaian masalah.
   2. Secara sederhana, terurai, rapi dan jelas.
   3. Menggunakan simbol-simbol standar.

b. Simbol Flowchart :

   Simbol-simbol yang di pakai dalam flowchart dibagi menjadi 3 kelompok, yaitu :

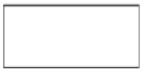
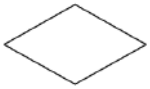
   1. Simbol *Flow Direction :* Di gunakan untuk menghubungkan simbol satu dengan yang lain di sebut juga *connecting line*.

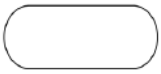
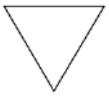
**Tabel 2.Simbol *Flow Direction***

| | |
|---|---|
|  | Simbol arus/*flow*, yaitu menyatakan jalannya arus suatu proses. |
|  | Simbol *communication link*, yaitu menyatakan transmisi data dari satu lokasi ke lokasi lain. |
|  | Simbol *connector*, yaitu menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama. |
|  | Simbol *offline connector*, yaitu menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda. |

2. Simbol *Processing* *:* Menunjukan jenis operasi pengolahan dalam suatu proses / prosedur

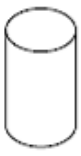
**Tabel 3.Simbol *Processing***

| **Simbol** | **Keterangan** |
|---|---|
|  | Simbol *process*, yaitu menyatakan suatu tindakan (proses) yang dilakukan oleh komputer |
|  | Simbol *manual*, yaitu menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer (manual) |
|  | Simbol *decision*, yaitu menujukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban : ya / tidak |
|  | Simbol predefined process, yaitu menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal. |

| | |
|---|---|
|  | Simbol *terminal*, yaitu menyatakan permulaan atau akhir suatu program |
|  | Simbol *keying operation*, Menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai *keyboard* |
|  | Simbol *offline-storage*, menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu |
|  | Simbol *manual input*, memasukkan data secara manual dengan menggunakan online keyboard |

3. Simbol *Input / Output :* Menunjukkan jenis peralatan yang digunakan sebagai media *input* dan *output*.

**Tabel 4. Simbol *Input / Output***

| Simbol | Keterangan |
|---|---|
|  | Simbol *input*/*output*, menyatakan proses input atau output tanpa tergantung jenis peralatannya |
|  | Simbol *punched card*, menyatakan input berasal dari kartu atau output ditulis ke kartu |
|  | Simbol *magnetic tape*, menyatakan input berasal dari pita magnetis atau output disimpan ke pita magnetis |
|  | Simbol *magnetic disk*, menyatakan input berasal dari disk magnetis atau output disimpan ke disk magnetis |

| | |
|---|---|
|  | Simbol *document*, mencetak keluaran dalam bentuk dokumen (melalui printer) |
|  | Simbol *disk storage*, menyatakan input berasal dari dari disk atau output disimpan ke disk |

(Sumber: Ladjamudin (2013:268)

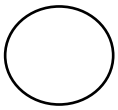
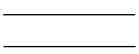
### 2.2.6 Data Flow Diagram (DFD)

Data Flow Diagram adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data tersimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data tersimpan dan proses yang dikenakan pada data tersebut. Data Flow Diagram (adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (input) dan keluaran (output). (A. S., Rosa (2018:70)

Data Flow Diagram (DFD) dapat digunakan untuk merepresentasikan sebuah sistem atau perangkat lunak pada beberapa level abstraksi. Data Flow Diagram (DFD) dapat dibagi menjadi beberapa level yang lebih detail untuk merepresentasikan aliran informasi yang lebih detail. Data Flow Diagram (DFD) menyediakan mekanisme untuk pemodelan fungsional ataupun pemodelan aliran informasi. Oleh karena itu, Data Flow Diagram (DFD) lebih sesuai digunakan untuk memodelkan fungsi-fungsi perangkat lunak yang akan diimplementasikan menggunakan pemrograman terstruktur, karena

pemrograman terstruktur membagi-bagi bagiannya dengan fungsi-fungsi dan prosedur-prosedur.

**Tabel 5.Simbol Data Flow /Diagram (DFD) Menurut Yourdan /De Marco**

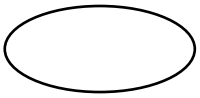
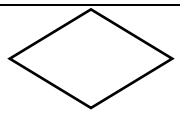
| Simbol | Nama | Keterangan |
|---|---|---|
|  | *External Entity* | Entitas eksternal, dapat berupa individu, organisasi, atau sistem yang berada di luar sistem, akan tetapi berinteraksi dengan sistem. |
|  | *Data Flow* | Aliran data dengan arah khusus dari sumber ke tujuan. |
|  | *Process* | Aktivitas atau fungsi yang dilakukan untuk alasan bisnis yang spesifik, bisa berupa manual atau terkomputerisasi. |
|  | *Data Store* | Tempat penyimpanan data hasil suatu proses. |

Sumber: Rosa, A.S, (2013)

### 2.2.7 Entity Relationship Diagram (ERD)

Menurut Ladjamudin (2013:142), “Entity Relationship Diagram (ERD) adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak”. ERD digunakan oleh professional sistem untuk berkomunikasi dengan pemakai eksekutif tingkat tinggi dalam suatu organisasi. Adapun simbol-simbol dari Entity Relationship Diagram (ERD) adalah sebagai berikut:

**Tabel 6.*Simbol-simbol Entity Relational Database (ERD)***

| No. | Simbol | Keterangan |
|---|---|---|
| 1 |  | Suatu kumpulan objek atau sesuatu yang dapat dibedakan atau dapat didefinisikan secara unik. |
| 2 |  | Karakteristik dari entitas atau Relationship yang menyediakan penjelasan detail entitas atau relation. |
| 3 |  | Hubungan yang teradi antara satu entitas atau lebih. |
| 4 |  | Baris sebagai penghubung antara himpunan, relasi dan himpunan entitas dari atributnya. |

( Sumber: Ladjamudin (2013:149)

### 2.2.8 Basis Data (Database)

Menurut Priyadi (2014:2) “Basis Data adalah sekumpulan fakta berupa representasi tabel yang saling berhubungan dan disimpan dalam media penyimpanan secara digital.” Menurut Fathansyah (2015:3) “Basis Data terdiri dari 2 kata, yaitu Basis dan Data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang/berkumpul. Sedangkan Data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang hewan, peristiwa, konsep, keadaan, dan sebagainya, yang diwujudkan dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya ”.

Berdasarkan pengertian diatas dapat disimpulkan bahwa basis data (database) adalah sebuah kumpulan fakta berupa representasi tabel yang saling berhubungan dan disimpan dalam media penyimpanan secara digital dengan memelihara data yang sudah di olah atau informasi dan membuat informasi tersedia saat dibutuhkan.

### 2.2.9 MySQL

Menurut Sibero (2013:97) “MySQL atau dibaca “My Sekuel” dengan adalah suatu RDBMS (Relational Database Management System) yaitu aplikasi sistem yang mejalankan fungsi pengolahan data”. Menurut Hidayatullah dan Jauhari (2015:180) “MySQL adalah salah satu aplikasi DBMS yang sudah banyak oleh para pemogram aplikasi web. Contoh DBMS lainnya adalah : PostgreSQL(freeware), SQL Server, MS Access dari Microsoft, DB2 dari IBM, Oracle dan Oracle Corp, Dbase, FoxPro, dsb”.

Menurut Buana (2014:2), “MySQL Merupakan database server yang paling sering digunakan dalam pemograman PHP. MySQL digunakan untuk menyimpan data dalam database dan memanipulasi data-data yang diperlukan. Manipulasi data tersebut berupa menambah, mengubah, dan menghapus data yang berada dalam database”.

Berdasarkan penjelasan diatas dapat disimpulkan bahwa MySQL adalah aplikasi DBMS yang menjalankan fungsi pengelolahan data untuk membangun sebuah aplikasi web.

### 2.2.10 XAMPP

Menurut Purbadian (2016:1), berpendapat bahwa “XAMPP merupakan suatu software yang bersifat open source yang merupakan pengembangan dari LAMP (Linux, Apache, MySQL, PHP dan Perl)”. Xampp merupakan tool yang menyediakan paket perangkat lunak ke dalam satu buah paket.

Sedangkan menurut Buana (2014:4), “XAMPP adalah perangkat lunak opensource yang diunggah secara geratis dan bisa dijalankan di semua semua operasi seperti

windows, linux, solaris, dan mac”.Berdasarkan pengertian diatas dapat disimpulkan bahwa Xampp merupakan tool pembantu pengembangan paket perangkat lunak berbasis open source yang menggabungkan Apache web server, MySQL, PHP dan beberpa modul lainnya di dalam satu paket aplikasi.

### 2.2.11 PHP (*PHP Hypertext Processor)*

Menurut Hikmah, dkk (2015:1) ”PHP merupakan kependekan dari Hypertext Preprocessor. PHP tergolong sebagai perangkat lunak open source yang diatur dalam aturan general purpose licences (GPL). Bahasa pemograman PHP sangat cocok dikembangkan dalam lingkungan web, karena PHP bisa diletakkan pada script HTML atau sebaliknya. PHP dikhususkan untuk pengembangan web dinamis”.

Menurut Sibero (2013:49) “PHP adalah pemograman interpreter yaitu proses penerjemahan baris kode sumber menjadi kode mesin yang dimenegerti komputer secara langsung pada saat baris kode dijalankan”.

Dari pendapat di atas dapat disimpulkan bahwa PHP adalah penerjemahan baris kode yang bisa dibaca atau dimengerti oleh komputer karena PHP bisa diletakkan pada script HTML atau sebaliknya. PHP dikhususkan untuk pengembangan web dinamis.

### 2.2.12 Web Browser dan HTML

*Web browser* adalah program untuk menampilkan halaman yang berbentuk kode HTML. Semua halaman web ditulis dengan bahasa HTML ( *Hypertext Mark Up Language)*. Walaupun beberapa file mempunyai ekstensi yang berbeda (contoh .html, .php, .php3), output file-file tersebut tetap HTML. HTML adalah medium yang selalu dikirimkan ke web browser baik halaman itu berupa halaman statis, sebuah *script* (seperti PHP), ataupun yang dibuat oleh program CGI (*Common Gateway Interface*). (Rosario Alexander.F.K , 2013).

### 2.2.13 Sublime Text

Menurut Supono (2016:14) "Sublime Text merupakan perangkat lunak text editor yang digunakan untuk membuat atau meng-edit suatu aplikasi. Sublime Text mempunyai fitur plugin tambahan yang memudahkan programmer. Selain itu, Sublime Text terkesan elegan untuk sebuah syntax editor. Selain ringan, IDE ini memiliki kecepatan proses simpan dan buka file. Tidak heran kalau IDE ini paling banyak digunakan terutama dikalangan programmer berbasis web".

Menurut Eric Haughee (2013) "Sublime Text adalah aplikasi editor untuk kode dan teks yang dapat berjalan di berbagai platform operating system dengan menggunakan teknologi Phyton API. Terciptanya aplikasi ini terinspirasi dari aplikasi Vim. Aplikasi ini sangatlah fleksibel dan powerfull. Fungsionalitas dari aplikasi ini dapat dikembangkan dengan menggunakan sublime-packages. Sublime Text bukanlah aplikasi open source, yang artinya aplikasi ini membutuhkan lisensi (license) yang harus dibeli. Akan tetapi beberapa fitur pengembangan fungsionalitas (packages) dari aplikasi

ini merupakan hasil dari temuan dan mendapat dukungan penuh dari komunitas serta memiliki linsensi (license) aplikasi gratis".

### 2.2.14 Blackbox Testing

*Black box* adalah metode pengujian sistem tanpa melihat *source code* sistem tersebut. Oleh karena itu, bentuk pengujian yang dilakukan bersifat percobaan dengan berbagai macam kemungkinan *input* dan melihat hasil yang diberikan oleh sistem. Berbeda dengan metode *white box testing*, metode pengujian ini dijalankan setelah system selesai dibangun. Dengan demikian dapat diketahui apakah sistem memberikan tanggapan yang benar terhadap *input* atau kesalahan yang mungkin terjadi.

Klasifikasi black box testing mencakup beberapa pengujian menurut Simarmata (2010:316), yaitu:

a. Pengujian Fungsional *(Functional Testing*), Pada pengujian ini, perangkat lunak diuji untuk persyaratan fungsional.
b. Pengujian Penyelidikan *(Exploratory Testing*), Pengujian penyelidikan mirip dengan pengujian khusus dan dilakukan untuk mempelajari/mencari aplikasi.
*c.* Pengujian Usabilitas (*Usability Testing*), Pengujian ini disebut juga sebagai pengujian untuk keakraban pengguna (*testing for user-friendliness).*
d. Pengujian Skenario (*Scenario Testing),* Pengujian skenario adalah pengujian yang realistis, kredibel dan memotivasi stakeholder, tantangan untuk program dan mempermudah penguji untuk melakukan evaluasi.
e. Pengujian Pengguna (*Use Acceptance*), Pada jenis pengujian ini, perangkat lunak akan diserahkan kepada pengguna untuk mengetahui apakah perangkat lunak memenuhi harapan pengguna dan bekerja seperti yang diharapkan.

# BAB III

# METODE PENELITIAN

## 3.1 Lokasi Penelitian

Dalam membuat sistem pakar deteksi penyakit pada tanaman semangka ini, dibutuhkan data pendukung dan sumber informasi yang relefan. Oleh sebab itu, peneliti melakukan penelitian pada Intansi BPTP (Balai Pengkajian Teknologi Pertanian) yang terletak pada jalan G. Obos Km. 5, Menteng, Kecamatan Jekan Raya, Kota Palangka Raya, Kalimantan Tengah.

## 3.2 Teknik Pengumpulan Data

Tenik pengumpulan data yang dilakukan oleh peneliti untuk membuat sistem pakar ini ada dua yaitu studi literatur dan Wawancara. Berikut adalah uraiannya :

**a. Studi Literatur**

Mempelajari literatur mengenai sistem pakar, pemrograman PHP dan MySQL.

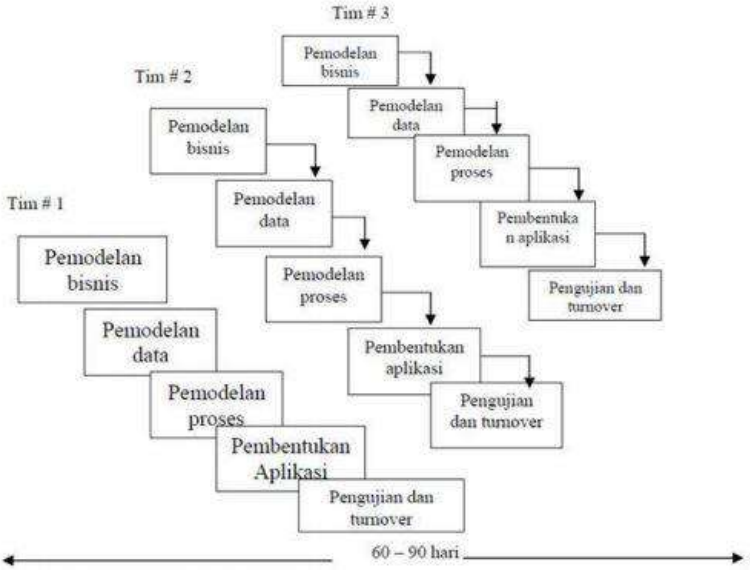
**b. Wawancara**

Pada kesempatan ini Peneliti melakukan pengumpulan data melalui Wawancara deangan Pakar/Ahli di BPTP mengenai penyakit pada tanaman semangka.

## 3.3 Analisis

### 3.3.1 Metodologi Penelitian

Metodologi penelitian yang digunakan dalam membangun Rancang Bangun Sistem Pakar Deteksi Penyakit Pada Tanaman Semangka adalah metodologi RAD (*Rapid Application Development)* . Berikut adalah urutan metodologi RAD :




**Gambar 1. *Rapid Application Development* (Sukamto & Shalahudin, 2016:13)**

**a. Pemodelan Bisnis**

Pada tahapan ini peneliti mengumpulkan bahan-bahan serta melakukan pengamatan terhadap kebutuhan aplikasi berbasis website dibanding dengan aplikasi berbasis desktop. Informasi tersebut dianalisis untuk mendapatkan spesifikasi sistem

**b. Pemodelan Data**

Memodelkan data apa saja yang dibutuhkan berdasarkan pemodelan bisnis dan mendefinisikan atribut-atributnya beserta relasinya dengan data-data yang lain. Tahapan ini peneliti menggunakan ERD untuk pemodelan basis data sehingga dapat diketahui atribut apa saja yang diperlukan dan bagaimana relasi datanya.

**c. Pemodelan Proses**

Perancangan dan permodelan sistem yang digunakan adalah *Data Flow Digram* (DFD) Sedangkan untuk perancangan *database* digambarkan dengan *Entity Relationship Diagram* (ERD). Untuk representasi pengetahuan sistem menggunakan metode naive bayes. Setelah perancangan konsep sistem dan perancangan *database*, selanjutnya akan dibuat perancangan atau pembuatan aplikasi untuk lebih memahami gambaran besar dari apa yang akan dikerjakan.

**d. Pembuatan Aplikasi**

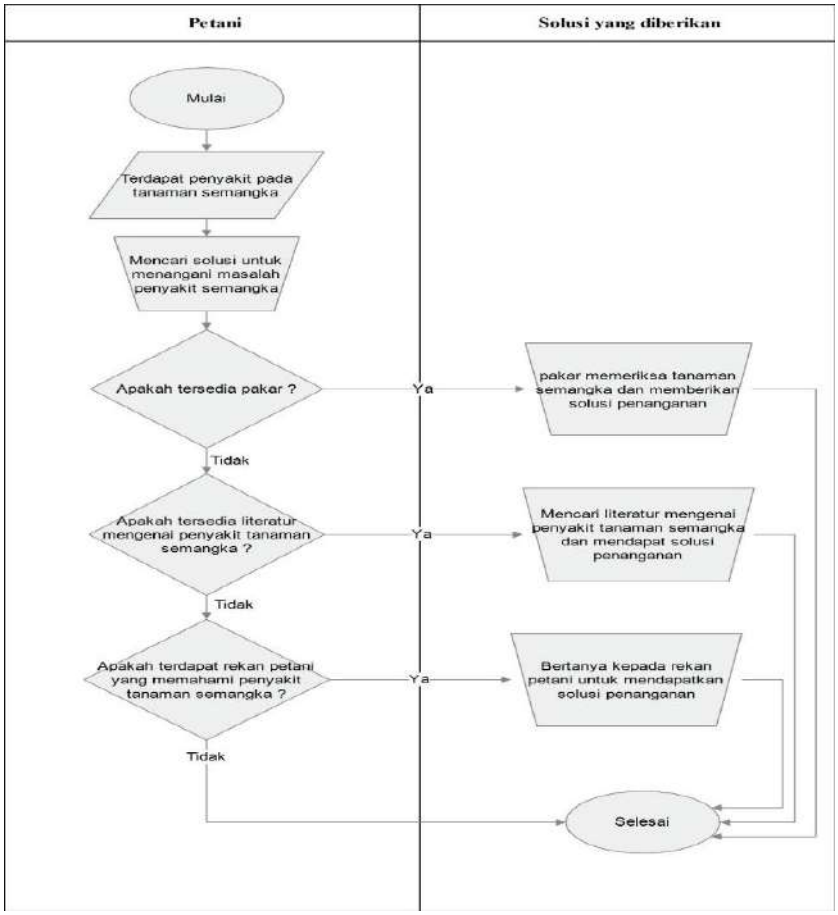
Tahapan pembuatan aplikasi merupakan proses penerjemahan bentuk desain menjadi kode atau bentuk/bahasa yang dapat dibaca oleh mesin. Pada tahap ini rancangan yang telah dibuat dengan bahasa pemrograman PHP dan *database* MySQL,serta hak akses ada dua yaitu Admin (Pakar) dan User (Pengguna).

**e. Pengujian dan Pergantian**

Tahapan Pengujian dan pergantian merupakan tahapan pengujian atau testing terhadap sistem dan juga kode yang sudah dibuat. Tujuannya untuk menemukan kesalahan yang mungkin terjadi untuk nantinya diperbaiki.Biasanya dilakukan testing untuk memastikan apakah hasilnya seperti yang diinginkan. Dalam tahapan testing ini digunakan 1 buah testing, yaitu Blackbox testing jika terjadi kesalahan maka dilakukan pergantian.

### 3.3.2 Analisis Proses

Pada umumnya pemeriksaan atau deteksi untuk mengetahui penyakit pada tanaman semangka pada saat ini masih manual yaitu para petani mencari beberapa sumber solusi yang ada, jika terdapat seorang pakar maka petani akan menghubungi pakar tersebut, apabila tidak ada maka petani akan mencari solusi dari beberapa literatur tentang penyakit tanaman semangka, apabila hal itu juga tidak berhasil petani akan mencari solusi kepada sesama petani semangka yang mungkin mengetahui tentang penyakit tersebut. Berikut adalah bisnis proses lama untuk mendeteksi penyakit pada tanaman semangka yang disajikan dalam bentuk flowchart :




**Gambar 2.Flowchart Sistem Lama**

### 3.3.3 Analisis Kelemahan Sistem

Pada tahap analisis kelemahan sistem, penulis menggunakan metode analisis *PIECES* untuk mengidentifikasi masalah pada sistem lama yang masih menggunakan metode manual melalui pengamatan ke sistem baru berbasis *web* analisis meliputi *performance, information, economy, control, efficiency* dan *service.* Penjelasan kinerja sistem lama ke sistem baru akan dijelaskan pada table dibawah ini.

**Tabel 7.Perbandingan *Performance*(kinerja)**

| Sistem Lama | Sistem Baru |
|---|---|
| 1) Kinerja sistem lama tergolong lama dan memakan waktu yang lama, dikarenakan petani perlu mencari ahli atau pakar dibidang penyakit pada tanaman semangka sehingga memperlama waktu dan kurang efisien | 1) Kinerja sistem baru lebih cepat dikarenakan seluruh proses dilakukan dengan otomatis dari web, sehingga petani hanya perlu mengisi atau menjawab pertanyaan pada web sistem pakar tanpa mencari ahli atau pakar. |

**Tabel 8.Perbandingan *Information*(Informasi)**

| Sistem Lama | Sistem Baru |
|---|---|
| 1) Sistem terlalu lama ketika Petani sedang ingin melakukan konsultasi masalah penyakit karena informasi data penyakit yang diberikan membutuhkan waktu yang cukup lama | 1) Sistem baru akan memberikan kemudahan kepada petani dalam mengecek dan mengakses informasi serta konsultasi dengan web sitem pakar tanpa menunggu lama |

**Tabel 9.Perbandingan *Economy*(Ekonomi)**

| Sistem Lama | Sistem Baru |
|---|---|
| 1) Dari segi ekonomi petani seringkali mengeluarkan biaya untuk konsultasi masalah penyakit karena factor jarak | 1) Sistem baru memberikan perubahan yang besar dalam proses konsultasi sehingga menghemat petani dari segi biaya karena melakukan konsultasi penyakit hanya melalui web |

**Tabel10.Perbandingan *Control*(Pengendalian)**

| Sistem Lama | Sistem Baru |
|---|---|
| 1) Kontrol sistem terlalu rumit karena seluruh proses konsultasi manual dan hanya dikerjakan oleh ahli atau pakar. | 1) Kontrol sistem menjadi lebih mudah dikarenakan seluruh proses dikerjakan dari web. Pakar hanya memberikan pengawasan dan mengimput data untuk data penyakit |

**Tabel 11.Perbandingan *Efficiency*(Efisiensi)**

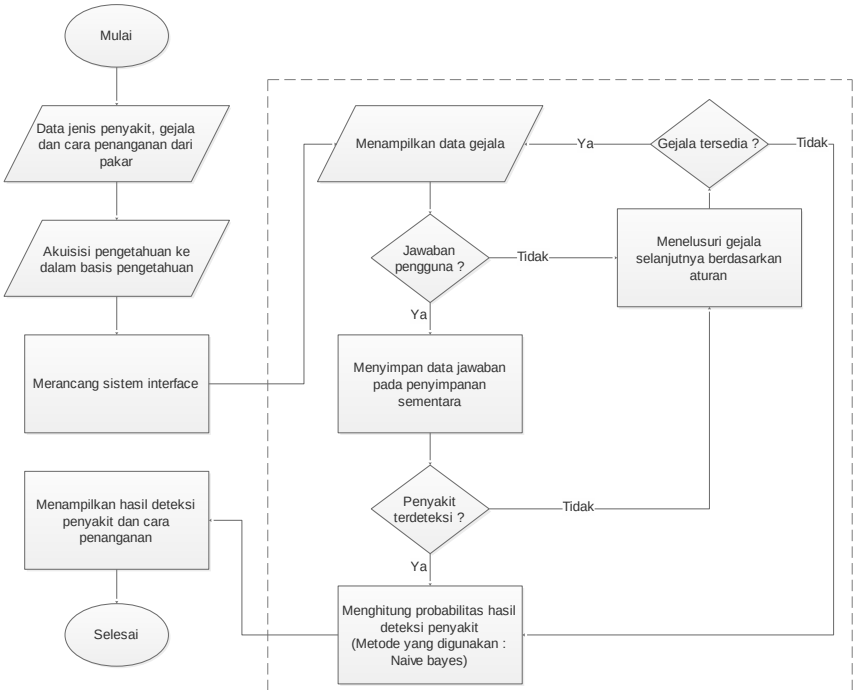
| Sistem Lama | Sistem Baru |
|---|---|
| 1) Ahli/pakar dapat melakukan konsultasi penyakit kepada petani secara bersamaan jika memiliki gejala yang sama kecuali jika petani memiliki gejala yang berbeda maka harus bergantian dan menimbulkan waktu yang lama | 1) Proses konsultasi bisa dilakukan secara sekaligus dengan gejala yang berbeda-beda tanpa harus menunggu lama. |

**Tabel 12.Perbandingan *Service*(Layanan)**

| Sistem Lama | Sistem Baru |
|---|---|
| 1) Pelayanan menjadi terlalu lama dan memakan waktu yang lama dikarenakan ahli/pakar harus menetukan penyakit/hama dengan gejala yang berbeda-beda secara manual dan itu berakibat melambatnya penganan yang dilakukan petani | 1) Pada sistem yang baru petani menjadi lebih cepat dalam menetukan penyakit tanpa harus menunggu lama hanya dengan menggunakan hp atau leptop petani dapat mengakses web sistem pakar dengan mudah |

### 3.3.4 Analisis Kebutuhan

Berdasarkan analisis sistem lama tersebut maka dibuatlah sistem baru berupa sistem pakar yang dapat membantu mendeteksi penyakit pada tanaman semangka. Flowchart dari cara kerja sistem pakar yang akan dibentuk adalah sebagai berikut :




**Gambar 3.Flowchart cara kerja sistem pakar**

**a. Kebutuhan Perangkat Keras**

Kebutuhan perangkat keras (*Hardware)* yang peneliti gunakann dalam pembuatan sistem pakar adalah sebagai berikut :

**Tabel 13.Spesifikasi Perangkat Keras**

| No | Perangkat Keras | Spesifikasi |
|---|---|---|
| 1 | *Type* | AcerOne |
| 2 | *Processor* | Intel Core i3 |
| 3 | *Memmory* | 1000 GB |
| 4 | RAM | RAM 4 GB |
| 5 | *Keyboard* | Standar |

**b. Kebutuhan Perangkat Lunak**

Kebutuhan perangkat lunak (software) yang peneliti gunakan dalam pembuatan sistem pakar adalah sebagai berikut :

**Tabel 14.Spesifikasi Perangkat Lunak**

| No | Perangkat Lunak | Fungsi |
|---|---|---|
| 1 | CorelDRAW | Untuk mendesain sistem |
| 2 | Sublime Text | Untuk melakukan editing kode program |
| 3 | My_SQL | Untuk membuat basis data |
| 4 | XAMPP | Untuk menghubungkan basis data dengan sistem pakar |
| 5 | Chrome | Untuk membuka situs website |

**c. Kebutuhan Informasi**

Dalam membentuk sistem pakar ini, peneliti membutuhkan informasi mengenai penyakit tanaman semangka dan cara menangani penyakit tersebut. Informasi ini didapat peneliti dari Kantor BPTP, buku-buku tentang tanaman semangka dan beberapa jurnal yang dapat dijadikan referensi untuk sistem ini.

**Tabel 15.Data Penyakit Tanaman Semangka**

| Kode | Penyakit |
|---|---|
| P1 | Layu fusarium *(Fusarium oxysporum f. sp. Lagenariae Matsouo et Yamamoto)* |
| P2 | Rebah batang *(Pythium ultimum Trow)* |
| P3 | *Antraknosa (Colletotrichum lagenarium (pass) Ell. Et. Halst)* |
| P4 | Layu bakteri *(Erwinia tracheiphila E. F. Sm.)* |
| P5 | Embun tepung/Powdery mildew *(Spaerotheca fuliginea Schlech)* |
| P6 | Penyakit virus *(MWMV (Water Melon Virus)* |
| P7 | Ulat Grayak *(Spodoptera litura F.)* |
| P8 | Gangsir *(Brachytripes portentosus Licht)* |
| P9 | *Helicoverpa armigera* |
| P10 | Thrips |
| P11 | Kutu Kebul *(Bemisia tabaci Genn.)* |
| P12 | *Liriomyza sp* |
| P13 | *Henosepilachna spp* |
| P14 | Kumbang daun *(Aulacophora femoralis Motschulsky)* |
| P15 | Aphids *(Aphis gossypii Glover).* |
| P16 | Lalat buah *(Dacus spp)* |

**Tabel 16.Daftar Gejala**

| Kode | Nama Gejala |
|---|---|
| G1 | Tanaman tampak layu seperti kekurangan air |
| G2 | Pada pagi dan sore hari tanaman tampak segar |
| G3 | Daun-daun tua menguning dan rontok |
| G4 | Dalam waktu 2-3 hari tanaman mati kering dan berwarna coklat |
| G5 | Batang mengerut. |
| G6 | Batang bibit berwarna coklat |
| G7 | Batang rebah kemudian mati |
| G8 | Daun terlihat bercak-bercak coklat yang akhirnya berubah warna kemerahan dan akhirnya daun mati |
| G9 | Buah tampak bulatan berwarna merah jambu yang lama kelamaan semakin meluas |
| G10 | Tanaman mengalami layu permanen |
| G11 | Jika tanaman dipotong melintang tampak pembuluh *xylem* menghitam |
| G12 | Daun atau batang muda dilapisi semacam tepung berwama putih |
| G13 | Daun menjadi cokelat dan mengeriput |
| G14 | Pertumbuhan tanaman terhambat |
| G15 | Tanaman menjadi lemah |
| G16 | Daun melepuh |
| G17 | Daun belang-belang |
| G18 | Daun cenderung berubah bentuk |
| G19 | Tanaman kerdil dan timbul rekahan membujur pada batang. |
| G20 | Ulat menyerang daun sehingga menjadi bolong-bolong dan rusak |
| G21 | Ulat menyerang dan melubangi buah. |
| G22 | Pada malam hari pangkal batang tanaman muda sampai terpotong |
| G23 | Buah digerek menyebabkan busuk |
| G24 | Daun berubah warna seperti perunggu terutama pada bagian permukaan daun bawah |
| G25 | Daun menjadi keriting dan mengering |
| G26 | Tanaman kadang kala mati. |
| G27 | Bercak *nekrotik* pada daun |
| G28 | Rusaknya sel-sel dan jaringan daun |
| G29 | Jalur korokan berbentuk guratan-guratan bewarna perak pada daun |
| G30 | Larva dan imago hidup pada permukaan daun dan memakan jaringan daun |
| G31 | Daun meninggalkan tulang daun sehingga daun berlubang-lubang |
| G32 | Serangga aktif pada malam hari dan terbang dari satu tanaman ke tanaman lain dengan cepat |
| G33 | Daun menjadi bolong-bolong. |
| G34 | Daun-daun mengerut sampai keriting, terutama pada bagian daun-daun muda (pucuk). |
| G35 | Busuknya buah semangka |
| G36 | Buah di dalamnya terdapat *larva* lalat |
| G37 | Rusaknya jaringan daging buah. |

**Tabel 17.Basis Pengetahuan (Rule)**

| Kode | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 | P11 | P12 | P13 | P14 | P15 | P16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| G1 | * | | | | | | | | | | | | | | | |
| G2 | * | | | | | | | | | | | | | | | |
| G3 | * | | | | | | | | | | | | | | | |
| G4 | * | | | | | | | | | | | | | | | |
| G5 | * | | | | | | | | | | | | | | | |
| G6 | | * | | | | | | | | | | | | | | |
| G7 | | * | | | | | | | | | | | | | | |
| G8 | | | * | | | | | | | | | | | | | |
| G9 | | | * | | | | | | | | | | | | | |
| G10 | | | | * | | | | | | | | | | | | |
| G11 | | | | * | | | | | | | | | | | | |
| G12 | | | | | * | | | | | | | | | | | |
| G13 | | | | | * | | | | | | | | | | | |
| G14 | | | | | * | | | | | | | | | | | |
| G15 | | | | | * | | | | | | | | | | | |
| G16 | | | | | | * | | | | | | | | | | |
| G17 | | | | | | * | | | | | | | | | | |
| G18 | | | | | | * | | | | | | | | | | |
| G19 | | | | | | * | | | | | | | | | | |
| G20 | | | | | | | * | | | | | | | | | |
| G21 | | | | | | | * | | | | | | | | | |
| G22 | | | | | | | | * | | | | | | | | |
| G23 | | | | | | | | | * | | | | | | | |
| G24 | | | | | | | | | | * | | | | | | |
| G25 | | | | | | | | | | * | | | | | | |
| G26 | | | | | | | | | | * | | | | | | |
| G27 | | | | | | | | | | | * | | | | | |
| G28 | | | | | | | | | | | * | | | | | |
| G29 | | | | | | | | | | | | * | | | | |
| G30 | | | | | | | | | | | | | * | | | |
| G31 | | | | | | | | | | | | | * | | | |
| G32 | | | | | | | | | | | | | | * | | |
| G33 | | | | | | | | | | | | | | * | | |
| G34 | | | | | | | | | | | | | | | * | |
| G35 | | | | | | | | | | | | | | | | * |
| G36 | | | | | | | | | | | | | | | | * |
| G37 | | | | | | | | | | | | | | | | * |

Contoh perhitungan dengan menggunakan klasifikasi Naïve bayes dapat diterapkan pada tanaman yang mengalami gejala nomor 1, 4, 18, 35.

Keterangan gejala :
1. Batang bibit berwarna coklat
2. Batang rebah kemudian mati
Langkah-langkah perhitungannya

Penyakit ke - 1 : **Layu fusarium**

n = 1 1.nc = 1
p = 1/16 = 0.0625 2. nc = 1
m = 37

**Menghitung nilai P ($a_i|v_j$) dan menghitung nilai P($v_j$)**

$P(1|PB) = \frac{1+37 \times 0.0625}{1+37} = 0.0871$

$P(2|PB) = \frac{1+37 \times 0.0625}{1+37} = 0.0871$

P(PB) = 1/16 = 0.0625

**Menghitung P ($a_i$) x P ($v_j$) untuk tiap v**

P(PB) = 1/16 = 0.0625
= P(PB) x [ P(1|PB) x P(2|PB)]
= 0.0625 x [0.0871 x 0.0871]
= 0.00047415062

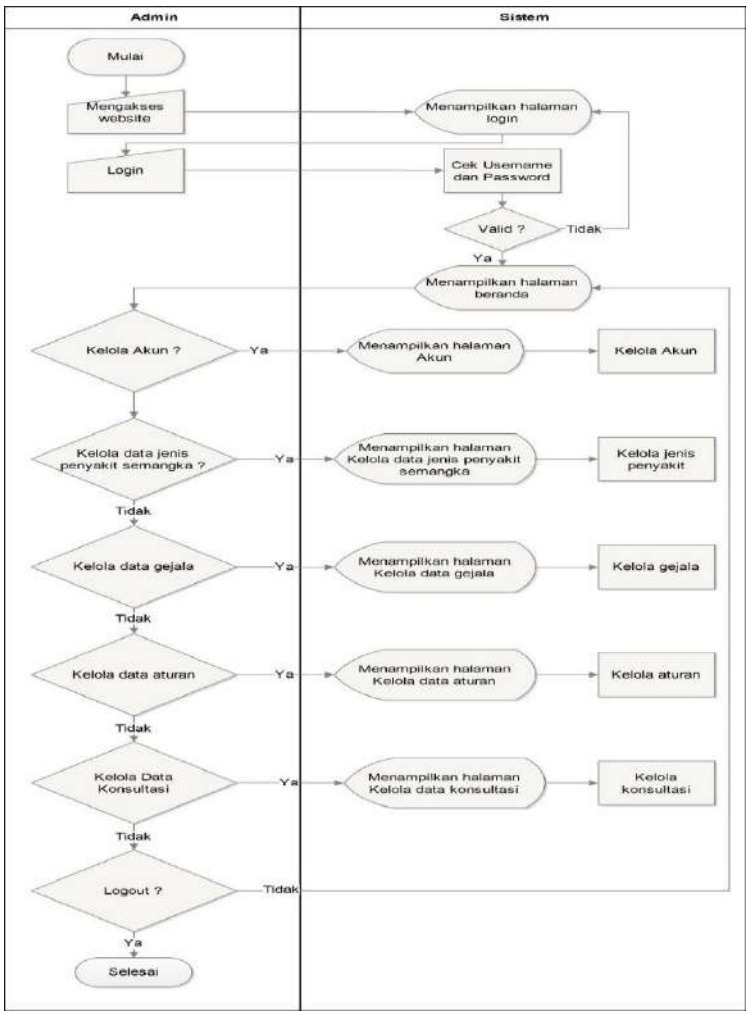
Pada contoh kasus yang diambil pada penelitian ini adalah penyakit Layu fusarium. Pertama melihat dari gejala-gejala yang dialami oleh tanaman yang mengalami penyakit. Setelah itu langkah selanjutnya adalah menentukan nilai nc (jumlah record pada data ) untuk class penyakit Layu fusarium. Setelah kita mengetahui

nilainya langkah selanjutnya adalah menghitung nilai P(ai|vj) dan nilai P(vj) untuk masing-masing gejala pada penyakit Layu fusarium. Setelah hasilnya didapatkan selanjutnya menghitung nilai P(ai|vj) x P(vj) untuk tiap v sesuai dengan persamaan 3 diatas. Hasil yang didapatkan untuk penyakit Layu fusarium dengan gejala-gejala nya mendapatkan hasil 9,04141370609527E-07. Hal ini seharusnya dibandingkan dengan gejala-gejala lainnya untuk mengetahui jenis penyakit tanaman.

**d. Kebutuhan Pengguna (User)**

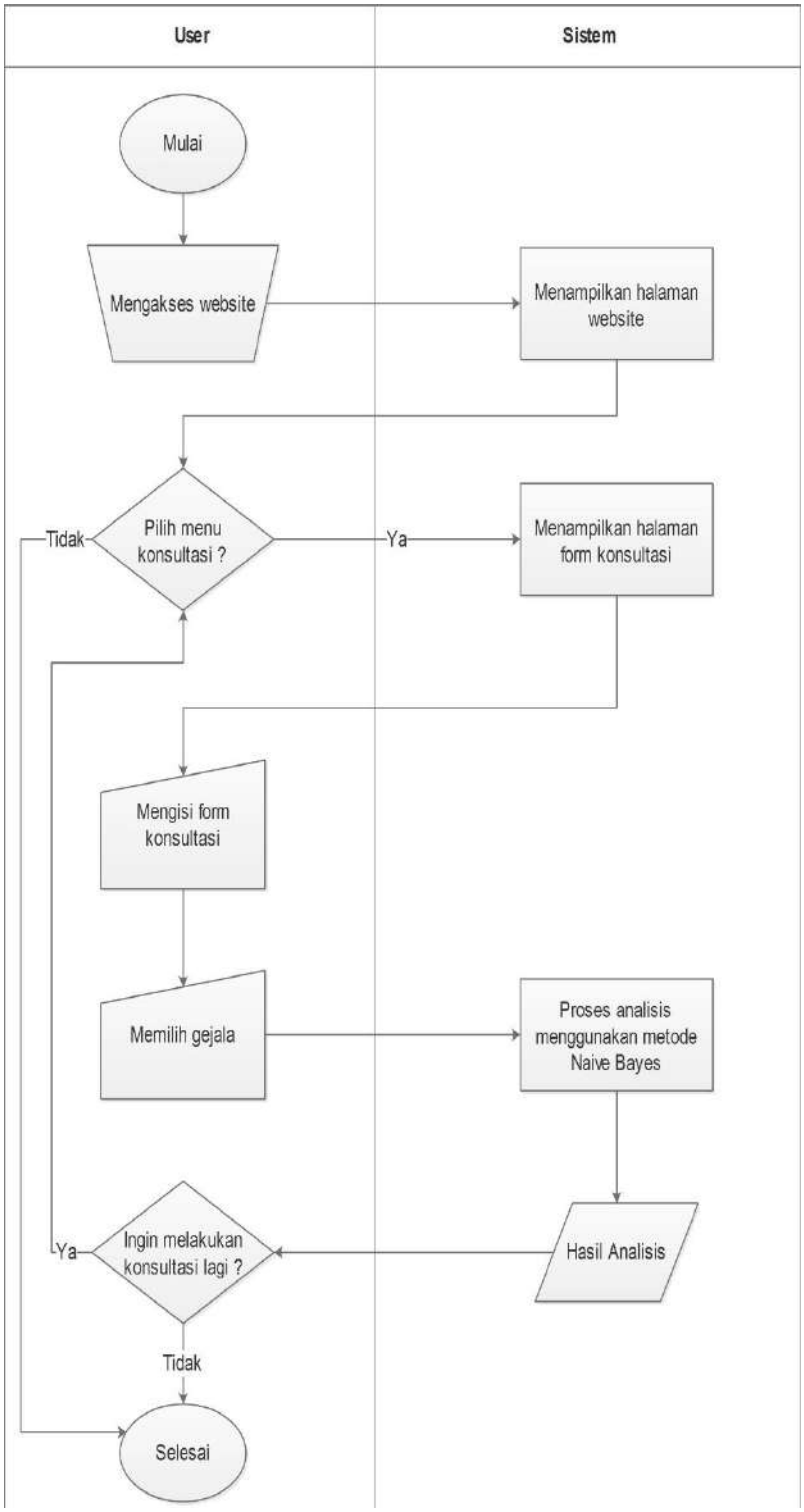
Dari hasil analisis kebutuhan yang sudah diuraikan sebelumya, maka didapat 2 kebutuhan pengguna yaitu admin dan user. Bisnis proses sistem baru yang akan dibentuk adalah sebagai berikut :

1. **Admin** (memiliki hak askes penuh terhadap semua fitur di dalam sistem)




**Gambar 4.Sistem Baru Admin**

**2.** **User** (memiliki hak akses terbatas yaitu hanya melakukan konsultasi. )




**Gambar 5.Sistem Baru User**

### 3.3.5 Analisis Kelayakan Sistem

Ada beberapa kriteria untuk analisis kelayakan sistem dalam penelitian ini yang diuraikan sebagai berikut :

**a. Kelayakan Teknologi**

Untuk kelayakan teknologi yang diberikan sistem ini adalah sebagai perangkat lunak yang dapat memberikan bantuan kepada para petani untuk mendeteksi penyakit pada tanaman semangka dan cara mengatasinya melalui website sistem pakar ini.

**b. Kelayakan Hukum**

Untuk kelayakan hukum sistem ini semua menggunakan perangkat lunak gratis atau *open source* sehingga tidak ada masalah dengan pelanggaran hukum atau *software* bajakan. Perangkat lunak yang dimaksud dalam pembuatan aplikasi ini Sublime tekt, MySQL dan Xampp.

**c. Kelayakan Operasional**

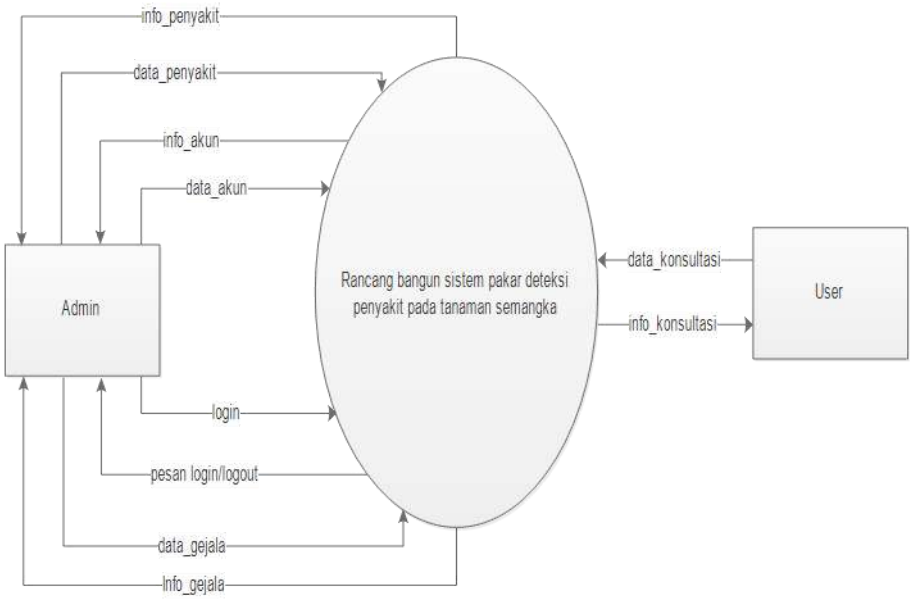
Untuk kelayakan operasional sistem yang dibangun, peneliti merancang tampilan yang sederhana agar pengguna sistem ini dapat lebih mudah memahami saat menggunakan sistem pakar tersebut.

## 3.4 Desain Sistem

Desain sistem yang digunakan untuk membuat sistem pakar ini menggunakan *Data Flow Diagram* (DFD) dan ERD (*Entity Relationship Diagram*). Berikut adalah uraiannya :

### 3.4.1 Desain Proses

**a. Diagram Konteks (DFD Level 0)**




**Gambar 6.Diagram Konteks**

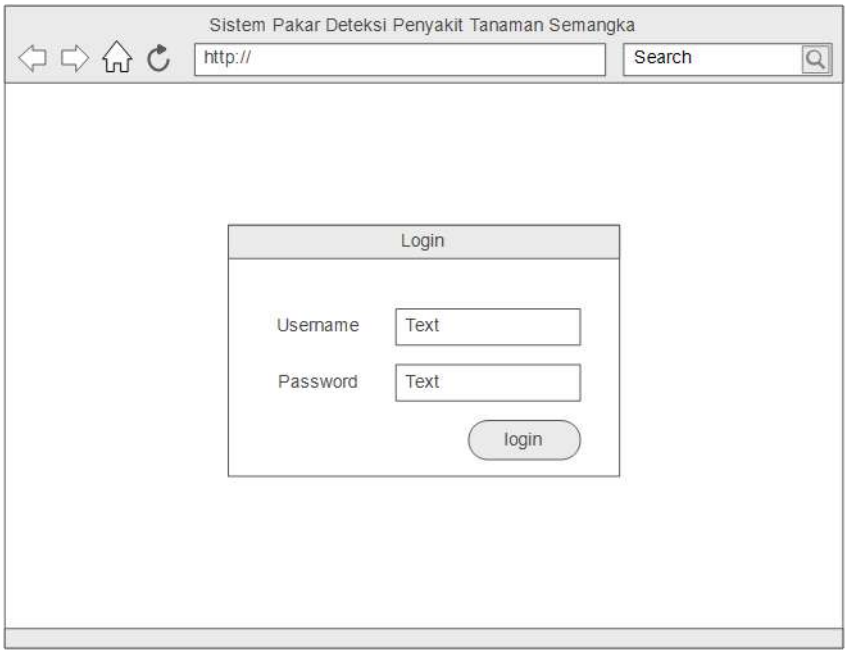
Keterangan :

Diagram konteks adalah gambaran umum untuk sistem yang akan dibangun dimana ditampilkan semua data yang masuk dan data yang keluar dari sistem. Entitas di dalam sistem ini ada dua yaitu admin dan user. Pada entitas admin, data yang masuk dan keluar dari sistem adalah data akun, data gejala, data penyakit, data aturan dan data konsultasi. Sedangkan pada entitas user, data yang masuk dan keluar dari sistem hanya data konsultasi.

### 3.4.2 Desain Perangkat Lunak

Desain perangkat lunak merupakan sebuah gambaran awal atau rancangan tampilan dari sistem yang akan dibangun. Berikut adalah desain perangkat lunak dari sistem pakar penyakit pada tanaman semangka :

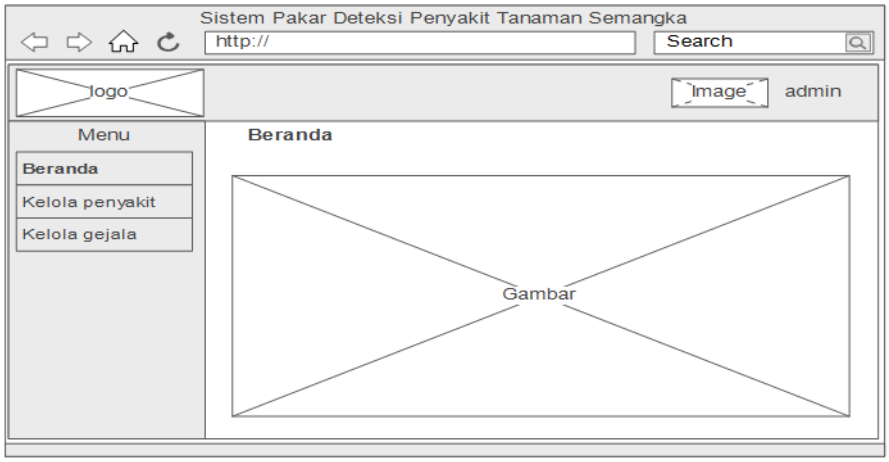
**1. *Login***




**Gambar 7.Halaman Login**

Tampilan diatas adalah rancangan untuk halaman *login* dimana admin harus memasukkan username dan password untuk masuk ke dalam sistem pakar.

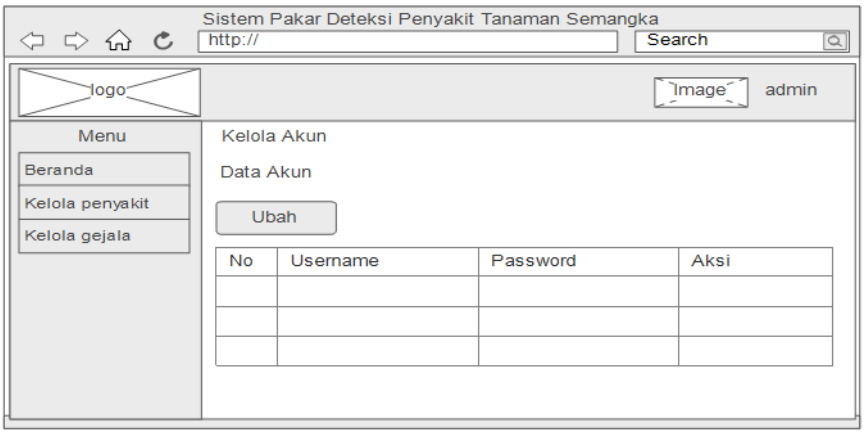
## 2. Beranda




**Gambar 8.Halaman Beranda**

Tampilan diatas adalah rancangan untuk halaman beranda. Tampilan ini adalah halaman pertama yang akan tampil ketika admin berhasil *login* atau masuk ke dalam sistem pakar. Pada halaman beranda ini terdapat fitur yang bisa diakses oleh admin untuk mengelola data sistem pakar tersebut.

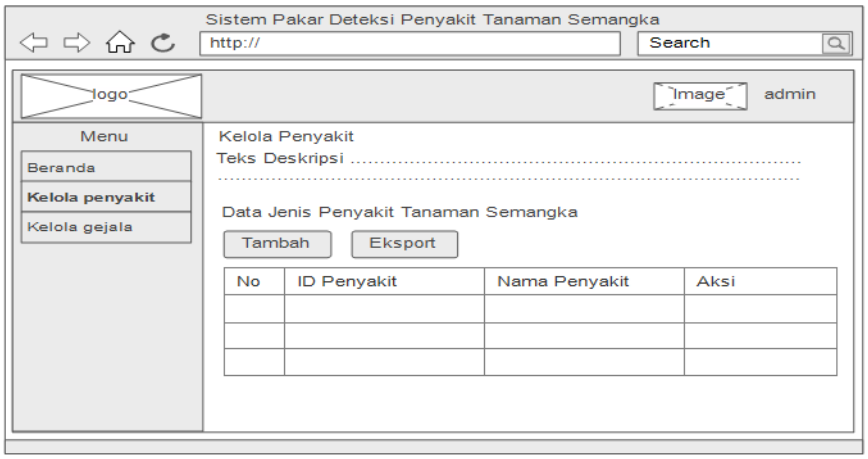
## 3. Kelola Akun




**Gambar 9.Halaman Kelola Akun**

Tampilan diatas adalah rancangan untuk halaman kelola akun. Pada halaman ini admin dapat mengubah *username* atau *password*. Tetapi tidak dapat menambah atau menghapus akun yang ada.

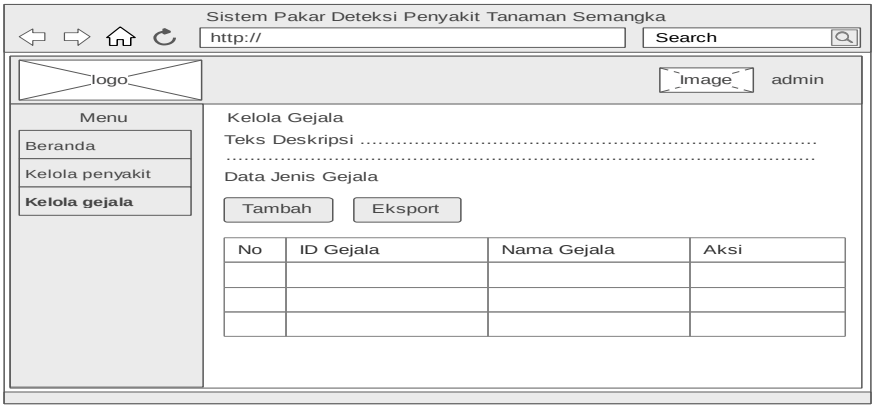
**4. Kelola Penyakit**




**Gambar 10.Halaman Kelola Penyakit**

Tampilan diatas adalah rancangan untuk halaman kelola penyakit. Pada halaman ini admin dapat menambah data penyakit atau mengeksport data penyakit yang ada di dalam sistem pakar ke dalam bentuk file.

**5. Kelola Gejala**

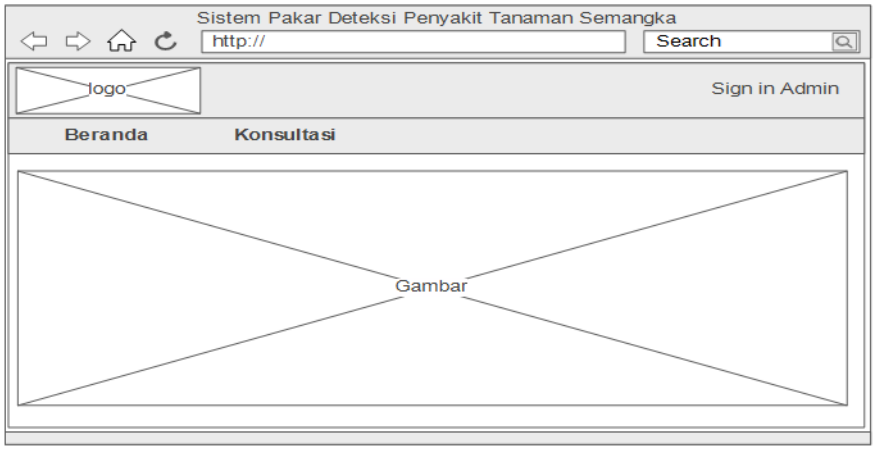



**Gambar 11.Halaman Kelola Gejala**

Tampilan diatas adalah rancangan untuk halaman kelola gejala. Pada halaman ini admin dapat menambah data gejala baru

atau mengeksport data gejala yang ada di dalam sistem pakar ke dalam bentuk file.

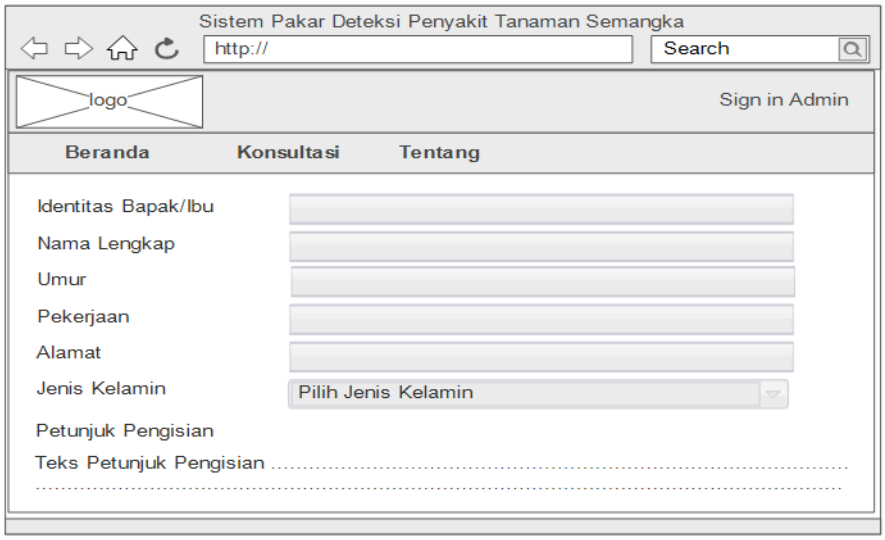
**6. Beranda (User)**




**Gambar 12.Halaman Bernada (User)**

Tampilan diatas adalah halaman beranda ketika user masuk ke dalam website sistem pakar deteksi penyakit tanaman semangka.

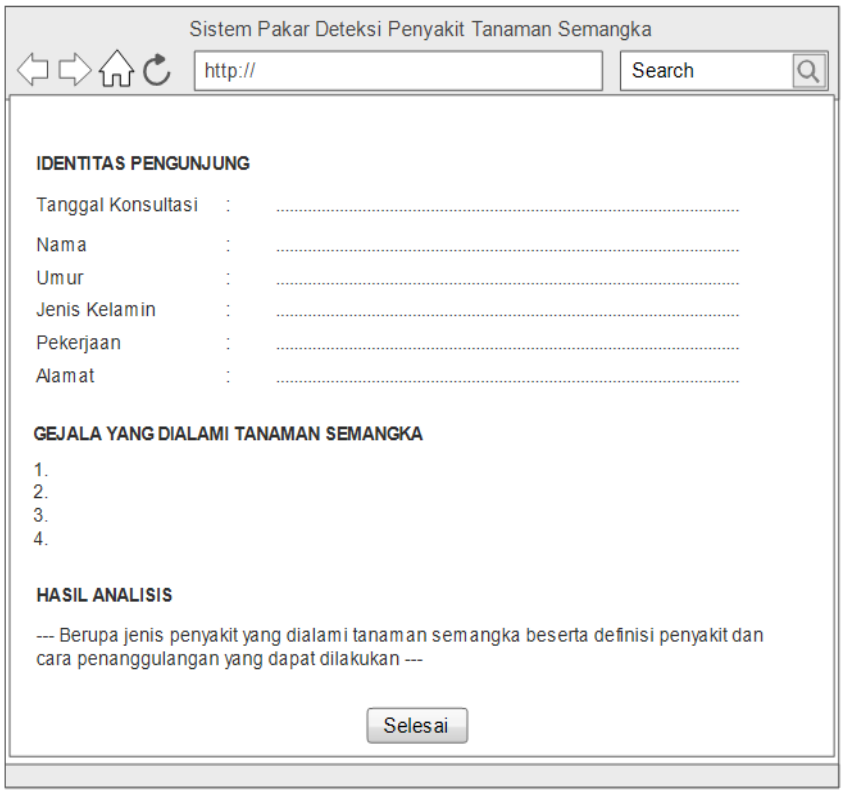
**7. Konsultasi (User)**




**Gambar 13.Halaman Konsultasi(User)**

Tampilan diatas adalah halaman konsultasi ketika user memiih menu konsultasi mengenai penyakit tanaman semangka. Dimana user memasukan identitas terlebih dahulu sebelum diarahkan untuk menjawab pertanyaan.

**8. Hasil Analisis**

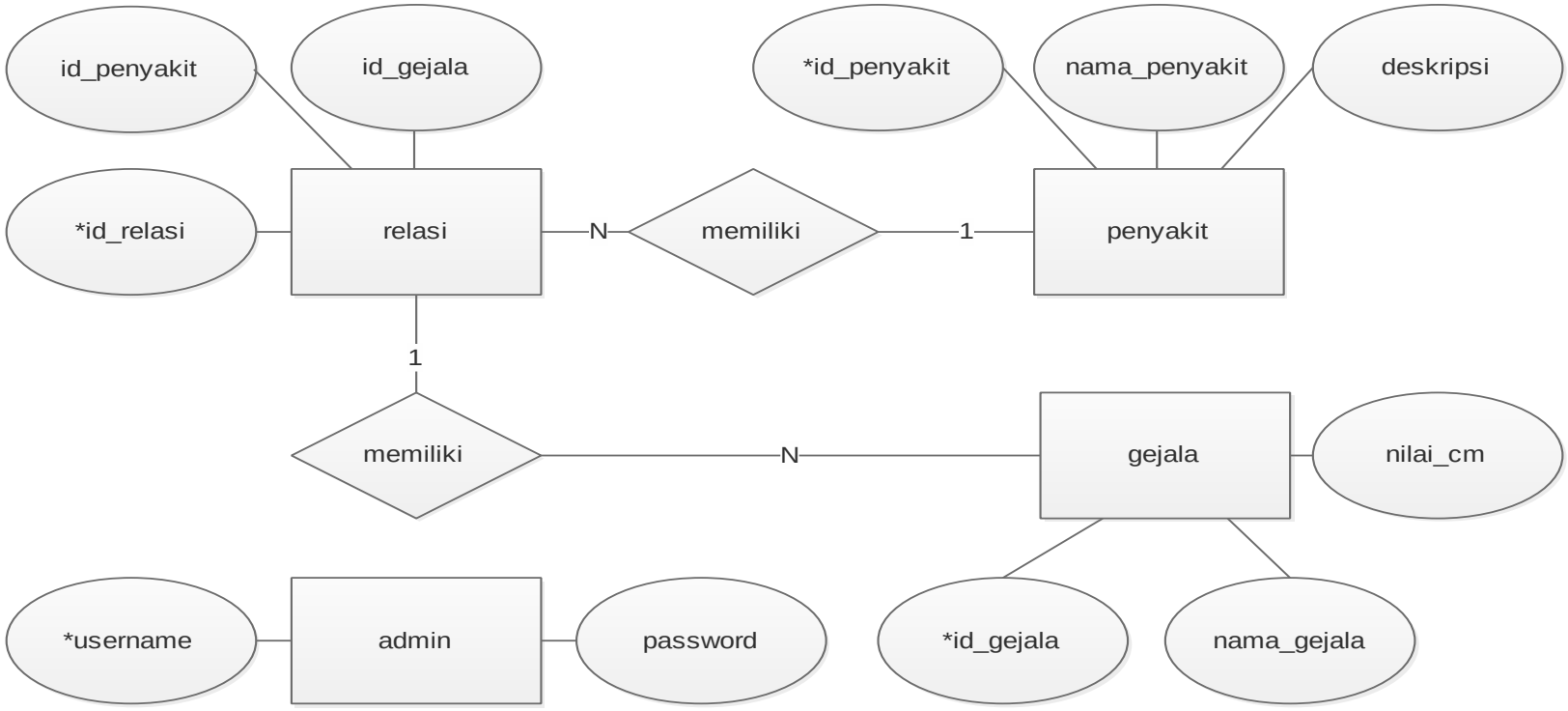



**Gambar 14.Halaman Hasil Analisis**

Tampilan diatas adalah halaman cetak hasil konsultasi yang akan tersedia setelah user selesai menjawab seluruh pertanyaan yang ada. Pada halaman ini akan diberikan hasil dari analisis penyakit tanaman semangka.

### 3.4.3 Desain Basis Data

Desain database sistem pakar ini menggunak an ERD (*Entity Relationship Diagram*). Berikut adalah desain database yang akan dibentuk :




**Gambar 16.Entity Relationship Diagram (ERD)**

### 3.4.4 Desain Tabel

Berikut adalah desain tabel dari database yang akan dibangun :

a. Tabel Akun

Tabel akun adalah tabel yang berisi data akun. Adapun field-field yang terdapat di dalam tabel meliputi :

**Tabel 18.Desain Struktur Tabel Akun**

| No | Nama Field | Tipe Data | Keterangan |
|---|---|---|---|
| 1. | username | Varchar (20) | *Primary Key* |
| 2. | password | Varchar (20) | |

b. Tabel Gejala

Tabel gejala adalah tabel yang berisi data gejala. Adapun field-field yang terdapat di dalam tabel meliputi :

**Tabel 19.Desain Struktur Tabel Gejala**

| No | Nama Field | Tipe Data | Keterangan |
|---|---|---|---|
| 1. | id_gejala | Varchar (20) | *Primary Key* |
| 2. | nama_gejala | Varchar (100) | |
| 3. | nilai_cm | Float | |

c. Tabel Penyakit

Tabel penyakit adalah tabel yang berisi data penyakit. Adapun field-field yang terdapat di dalam tabel meliputi :

**Tabel 20.Desain Struktur Tabel Penyakit**

| No | Nama Field | Tipe Data | Keterangan |
|---|---|---|---|
| 1. | id_penyakit | Varchar (20) | *Primary Key* |
| 2. | nama_penyakit | Varchar (100) | |
| 3. | deskripsi | Text | |

d. Tabel Relasi

Tabel relasi adalah tabel yang berisi data relasi dari tabel gejala dan tabel penyakit. Adapun field-field yang terdapat di dalam tabel meliputi :

**Tabel 21.Desain Struktur Tabel Relasi**

| No | Nama Field | Tipe Data | Keterangan |
|---|---|---|---|
| 1. | id_relasi | Int (20) | *Primary Key* |
| 2. | id_penyakit | Varchar (20) | *Foreign Key* |
| 3. | id_gejala | Varchar (20) | *Foreign Key* |

# BAB IV
# HASIL DAN PEMBAHASAN

## 4.1 Hasil

### 4.1.1 Impelentasi Program

Setelah analisis dan desain sistem terpenuhi, maka selanjutnya adalah pengimplentasian program. Program dibuat dengan metode *Naive Bayes* menggunakan bahasa pemrograman PHP dan database MySQL yang dibuat berbasis website sehingga bisa diakses oleh siapa saja menggunakan browser.

### 4.1.2 Pengujian Sistem dan Uji Coba

Pada bagian pengujian program ini menggunakan blackbox testing untuk mengetahui apakah program bisa berjalan dengan baik dan untuk menghindari kesalahan dari program yang dibuat sehingga fungsi dari sistem berjalan sesuai harapan. Berikut adalah pengujian sistem secara Blacboxt :

**Tabel 22.Hasil Pengujian BlackBox Pengunjung**

| No | Interface | Pengujian | Hasil |
|---|---|---|---|
| 1. | Beranda | Muncul tampilan beranda | Sesuai |
| 2. | Konsultasi | Data konsultasi bisa dimasukkan dan gejala penyakit bisa dipilih | Sesuai |
| 3. | Hasil Analisis | Data hasil analisis ditampilkan | Sesuai |

**Tabel 23.Hasil Pengujian BlackBox Admin**

| No | Interface | Pengujian | Hasil |
|---|---|---|---|
| 1. | SignIn | Admin berhasil SignIn ke dalam sistem | Sesuai |
| 2. | Beranda | Halaman Beranda ditampilkan | Sesuai |
| 3. | Kelola Akun | Password dan Username dapat diubah | Sesuai |
| 4. | Kelola Data Penyakit | Fitur Tambah dapat menambahkan data penyakit baru | Sesuai |
| | | Fitur Ubah dapat mengubah data penyakit yang sudah ada | Sesuai |
| | | Fitur Hapus dapat menghapus data penyakit yang ada | Sesuai |
| | | Fitur Export dapat mengekport data penyakit | Sesuai |
| 5. | Kelola Data Gejala | Fitur Tambah dapat menambahkan data gejala baru | Sesuai |
| | | Fitur Ubah dapat mengubah data gejala yang sudah ada | Sesuai |
| | | Fitur Hapus dapat menghapus data gejala yang ada | Sesuai |
| | | Fitur Export dapat mengekport data gejala | Sesuai |
| 6. | SignOut | Admin dapat keluar dari sistem | Sesuai |

### 4.1.3 Manual Program

Pada bagian ini akan diuraikan cara penggunaan program secara manual untuk setiap fitur yang ada dalam program sehingga pengguna dapat mengoperasikan program tersebut.

1. Install Xampp pada komputer
2. Pindahkan file Sistem Pakar pada folder Htdocs di dalam folder xampp.
3. Jalankan Xampp lalu buka browser
4. Import database pada PHPMyAdmin
5. Buka localhost/sistem_pakar untuk mengakses website
6. Pengunjung akan diarahkan ke halaman beranda dan bisa langsung melakukan konsultasi di website.
7. Fitur konsultasi digunakan untuk melakukan konsultasi dengan memasukkan data pengunjung terlebih dahulu
8. Jika data pengunjung sudah terisi, maka pengunjung bisa memilih gejala yang dialami oleh tanaman semangka.
9. Setelah memilih gejala, pengunjung bisa mengklik tombol hasil analisis, maka pengunjung akan diarahkan ke halaman yang berisi hasil dari analisis penyakit pada tanaman semangka tersebut.
10. Pengunjung dapat mengklik tombol selesai untuk kembali ke halaman beranda. Jika ingin melakukan konsultasi kembali, pengunjung dapat mengklik menu konsultasi dan melakukan konsultasi lagi.s

11. Untuk admin, bisa melakukan Signin terlebih dahulu, selanjutnya jika login berhasil maka akan diarahkan ke halaman beranda admin.
12. Didalam halaman admin, terdapat 5 fitur utama yang bisa diakses oleh admin yaitu :
    a. Fitur beranda, yaitu untuk mengakses halaman beranda
    b. Fitur Kelola Akun, yaitu untuk mengelola akun dimana admin bisa mengubah username dan password.
    c. Fitur Kelola Data Penyakit, yaitu untuk mengelola data penyakit yang ada di dalam program. Disini admin bisa menambahkan data penyakit baru, mengubah dan menghapus data penyakit yang sudah ada.
    d. Fitur Kelola Data Gejala, yaitu untuk mengelola data gejala penyakit yang ada di dalam program. Disini admin bisa menambahkan data gejala baru, mengubah dan menghapus data penyakit yang sudah ada.
    e. Fitur Logout, yaitu fitur agar admin bisa keluar dari program.

### 4.1.4 Manual Instalasi

Pada bagian ini akan diuraikan tahapan untuk cara instalasi program yang dibuat. Berikut adalah tahapan instalasi :

1. Installasi Xampp : Siapkan file xampp, pilih install lalu program akan mulai menginstall. Ketika memilih direktori penyimpanan, pastikan direktori yang dipilih adalah direktori untuk sistem (biasanya localdisk(C) ). Tunggu hingga proses installasi selesai.

2. Pindahkan folder yang berisi file program sistem pakar ke dalam localdisk(c) – xampp – htdocs.
3. Buka browser yang biasa anda gunakan.
4. Masukkan alamat file http://localhost/phpmyadmin/index.php di kolom tab browser
5. Klik menu baru untuk membuat basis data baru, beri nama sikar_semangka lalu klik tombol buat, maka basis data baru dengan nama sikar_semangka akan dibuat.
6. Masih dalam basis data sikar_semangka, klik menu import, lalu pilih file sikar_semangka.sql yang ada pada folder db di dalam folder sikar_semangka. Maka file database akan diimport.
7. Buka tab browser baru, masukkan alamat http://localhost/sikar_semangka/, maka akan diarahkan ke halaman beranda website sistem pakar.
8. Pengunjung bisa mengakses fitur yang ada dihalaman beranda serta dapat melakukan konsultasi mengenai penyakit tanaman semangaka dan untuk admin bisa melalukan SignIn dan mengelola data gejala dan data penyakit.

## 4.2 Pembahasan

### 4.2.1 Pembahasan Listing Program

Dalam pembahasan listing program akan dijelaskan mengenai kode program yang ada beserta penjelasannya. Akan tetapi dalam pembahasan ini penulis tidak membahas *source code* secara menyeluruh,

melainkan hanya membahas *source code* yang dirasa penting dan bisa menjadi bahan referensi dalam membangun sebuah website Sistem Pakar deteksi penyakit pada tanaman buah semangka.

1. *Source code* Konsultasi

   *Source code* ini berfungsi menampilkan halaman konsultasi, dimana pengunjung memasukkan identitas dan memilih gejala yang dialami oleh tanaman semangka. Jika data sudah terisi, maka pengunjung dapat mengklik tombol hasil analisis untuk mendapatkan hasil dari analisisnya. Dapat dilihat pada gambar.18

Gambar 18. *Source code* Konsultasi

```
<!DOCTYPE html>
<html lang="en">
<?php include 'head.php';?>

        <div class="pcoded-wrapper">
          <div class="pcoded-content">
            <div class="pcoded-inner-content">

              <!-- Main-body start -->
              <div class="main-body" >

                <div class="page-wrapper">
                  <!-- Page-header start -->
                  <div class="page-header m-t-50">

                  </div>
                  <!-- Page-header end -->
                  <!-- Page body start -->
                  <div class="page-body">

                    <div class="row">
                      <div class="col-lg-12">
                        <!-- Default card start -->
                        <div class="card">
                          <div class="card-header">
                          </div>
                          <div class="card-block" >
```

```
          <div class="row">
             <div class="col-sm-12">
               <!-- sini -->
               <div class="product-edit">

                  <!-- Tab panes -->
               <h2>Form Konsultasi</h2>
               <form action="proses.php" class="md-float-material card-
block" method="POST">
                  <div class="tab-content">
                     <div class="tab-pane active" id="bio" role="tabpanel">
                     <br>
                        <h4>Identitas Bapak/Ibu</h4>
                        <br>
                           <div class="row">
                           <div class="col-sm-12">
                                 <div class="input-group">
                                 <label class="form-label col-lg-2">Nama Lengkap
</label>
                                 <input value="" class="form-control col-lg-8"
type="text" id="nama_pasien" name="nama_pasien">
                              </div>
                           </div>
                           <div class="col-sm-12">
                                 <div class="input-group">
                                 <label class="form-label col-lg-2">Umur </label>
                                 <input value="" class="form-control col-lg-8"
type="number" id="umur" name="umur">
                              </div>
                           </div>
                           <div class="col-sm-12">
                              <div class="input-group">
                                 <label class="form-label col-lg-2">Jenis Kelamin
</label>
                                 <select name="jenis_kelamin" id="jenis_kelamin"
class="form-control form-control-primary col-lg-8">
                                 <option value="Laki-laki">Laki-laki</option>
                                 <option
value="Perempuan">Perempuan</option>
                                 </select>
                              </div>
```

```
                        </div>
                        <div class="col-sm-12">
                            <div class="input-group">
                            <label class="form-label col-lg-2">Pekerjaan
</label>
                            <input value="" class="form-control col-lg-8"
type="text" id="pekerjaan" name="pekerjaan">
                        </div>
                        </div>
                        <div class="col-sm-12">
                            <div class="input-group">
                            <label class="form-label col-lg-2">Alamat </label>
                            <input value="" class="form-control col-lg-8"
type="text" id="alamat" name="alamat">
                        </div>
                        </div>
                        <div class="col-sm-12">
                        <div class="input-group">
                            <label class="form-label col-lg-2">Gejala Pada
Tanaman</label>
                                <select class="js-example-basic-multiple form-
control col-lg-8" name="gejala[]" multiple="multiple">
                                    <?php $i=1;
                                    $result = query("SELECT * FROM gejala order
by id_gejala");
                                    while ($data = mysqli_fetch_assoc($result)) {
                                        $id= $data['id_gejala']; ?>
                                    <option value="<?php echo $id ?>"><?php
echo $data['nama_gejala']; ?></option>
                                    <?php $i++; } ?>
                                </select>
                        </div>
                        </div>
                        </div>
                        <hr>
                        <div class="row" role="tablist">
                        <div class="col-sm-12">
                            <div class="text-center m-t-20">
                                <button type="submit" class="btn btn-primary
waves-effect waves-light m-r-10">Hasil Analisis</button>
                            </div>
                        </div>
                        </div>
```

```
                    </div>
                      </form>
                    </div>
                    <!-- sini -->
                  </div>
                </div>
              </div>
                                  </div>
                                </div>
                                <!-- Default card end -->
                              </div>
                            </div>
                          </div>
                          <!-- Page body end -->
                        </div>
                      </div>
                      <!-- Main-body end -->
                    </div>
                  </div>
                </div>
              </div>
            </div>
          </div>
 <?php include 'footer.php'; ?>
 <script type="text/javascript">
   $(document).ready(function() {
   $('.js-example-basic-multiple').select2({
       placeholder: "Pilih Gejala Pada Tanaman Semangka",
       language :"en",
   });
});
</script>

</body>
<!-- Mirrored from colorlib.com//polygon/adminty/default/menu-horizontal-fixed.html by HTTrack Website Copier/3.x [XR&CO'2014], Thu, 13 Jun 2019 11:57:12 GMT -->
</html>
```

2. *Source Code* Proses perhitungan

*Source Code* ini berfungsi menampilkan dimana pengunjung memasukkan identitas dan memilih gejala yang dialami oleh tanaman semangka. Jika data sudah terisi, maka pengunjung dapat mengklik tombol hasil analisis untuk mendapatkan hasil dari analisisnya.

```
<!DOCTYPE html>
<html lang="en">
<?php include 'head.php';
if(!isset($_POST['nama_pasien'])||!isset($_POST['umur'])||!isset($_POST['jenis_kelamin'])||!isset($_POST['pekerjaan'])||!isset($_POST['alamat'])){
  echo"<script> alert ('Data Identitas Belum Lengkap. Silahkan diulang kembali'); history.back();</script>";
}else if(!isset($_POST['gejala'])){
  echo"<script> alert ('Data Gejala Belum Diinput. Silahkan diulang kembali'); history.back();</script>";
}
$nama = $_POST['nama_pasien'];
$umur = $_POST['umur'];
$jenis_kelamin = $_POST['jenis_kelamin'];
$pekerjaan = $_POST['pekerjaan'];
$alamat = $_POST['alamat'];
$gejala = $_POST['gejala'];
$tanggal = tgl_indo(date('Y-m-d'));
echo "<br><br><br>";
// foreach ($gejala as $key => $value) {
//   echo $key." - ".$value."<br>";
// }
$arr_penyakit  = array();
```

```
$arr_hasil  = array();
 $q = query("SELECT * FROM penyakit");
 while ($data_gm = mysqli_fetch_assoc($q)) {
  $relasi = array();
  // $arr_penyakit[]= $data_gm['id_penyakit'];
  $id_p = $data_gm['id_penyakit'];
    $q1 = query("SELECT * FROM relasi where id_penyakit='$id_p'");
    while ($data_relasi = mysqli_fetch_assoc($q1)) {
     $relasi[] = $data_relasi['id_gejala'];
    }
     $arr_penyakit[$data_gm['id_penyakit']]= $relasi;
     $arr_penyakit1[$data_gm['id_penyakit']]=
array($data_gm['nama_penyakit'],$data_gm['deskripsi']);
  }
ksort($arr_penyakit,SORT_NATURAL);
ksort($arr_penyakit1,SORT_NATURAL);
 $q = query("SELECT * FROM gejala");
 while ($data_gm = mysqli_fetch_assoc($q)) {
  if (in_array($data_gm['id_gejala'], $gejala)) {
    $arr_gejala[$data_gm['id_gejala']] = 1;
  }else{
    $arr_gejala[$data_gm['id_gejala']] = 0;
  }
    $arr_gejala1[$data_gm['id_gejala']]=$data_gm['nama_gejala'];
 }
ksort($arr_gejala,SORT_NATURAL);
ksort($arr_gejala1,SORT_NATURAL);
// p= 1/ banyaknya jenis class / penyakit
$p= 1/count($arr_penyakit);
```

```
// m= jumlah parameter / gejala
$m = count($arr_gejala);
// n= jumlah recordpada data learning yang v = vj / tiap class
$n = 1;
// nc= jumlah recordpada data learning yang v = vj dan a = ai
// var_dump($arr_penyakit);
foreach ($arr_penyakit as $kp => $vp) {
  $hasil = 1;
  foreach ($vp as $kg => $vg) {
     //(nc+(m.p))/(n+m)
     $arr_hitung[$kp][$vg] = ($arr_gejala[$vg]+($m*$p))/($n+$m);
     // echo "$hasil *=($arr_gejala[$vg]+($m*$p))/($n+$m)<br>";
     $hasil *=($arr_gejala[$vg]+($m*$p))/($n+$m);
  }
  $arr_hasil[$kp] = $p * $hasil;
}
ksort($arr_hasil,SORT_NATURAL);
?>
<div class="pcoded-wrapper">
   <div class="pcoded-content">
      <div class="pcoded-inner-content">
         <!-- Main-body start -->
         <div class="main-body" >
            <div class="page-wrapper">
               <!-- Page-header start -->
               <div class="page-header m-t-50">
```

```
</div>
<!-- Page-header end -->
<!-- Page body start -->
<div class="page-body">
   <div class="row">
      <div class="col-lg-12">
         <!-- Default card start -->
         <div class="card">
            <div class="card-header">
               <h2><center>Hasil Analisis</center></h2>
            </div>
 <div class="card-block" >
  <div class="row" align="center">
     <div class="col-sm-12">
       <br>
       <h5>Perhitungan (<em>Naive Bayes</em>) :</h5>
       <br>
       <p>Data Jawaban Gejala</p>
        <table class="col-sm-8" border="1">
          <tr>
            <td>Kode Gejala</td>
            <td>Gejala</td>
            <td>Nilai</td>
          </tr>
         <tbody>
          <?php
          foreach ($arr_gejala1 as $key => $value) { ?>
```

```
                        <tr>
                          <td><?php echo $key ?></td>
                          <td><?php echo $value ?></td>
                          <td align="center"><?php echo
$arr_gejala[$key]?></td>
                        </tr>
                       <?php } ?>
                      </tbody>
                     </table>
                     <p>Data Hasil Perhitungan</p>
                      <table class="col-sm-8" border="1">
                       <tr>
                        <td>Kode Penyakit</td>
                        <td>Nama Penyakit</td>
                        <td>Total Nilai</td>
                       </tr>

                      <tbody>
                       <?php
                       foreach ($arr_penyakit1 as $key => $value) { ?>
                        <tr>
                          <td><?php echo $key ?></td>
                          <td><?php echo $value[0] ?></td>
                          <td align="center"><?php echo
$arr_hasil[$key]?></td>
                        </tr>
                       <?php } ?>
                     </tbody>
```

```
                    </table>
                </div>
            </div>
            <div class="card-block" >
                <div class="row">
                    <div class="col-sm-2"></div>
                    <div class="col-sm-8">
                        <!-- sini -->
                        <!-- <br> -->
                        <h5>Identitas Bapak/Ibu :</h5>
                        <br>
                        <table class="col-sm-8">
                        <tr><td>Tanggal Konsultasi</td><td>:  <?php echo $tanggal; ?> </td></tr>
                        <tr><td>Nama</td><td>:  <?php echo $nama; ?> </td></tr>
                        <tr><td>Umur</td><td>:  <?php echo $umur; ?> </td></tr>
                        <tr><td>Jenis Kelamin</td><td>:  <?php echo $jenis_kelamin; ?> </td></tr>
                        <tr><td>Pekerjaan</td><td>:  <?php echo $pekerjaan; ?> </td></tr>
                        <tr><td>Alamat</td><td>:  <?php echo $alamat; ?> </td></tr>
                        </table>
                        <br>
                        <br>
                        <h5>Riwayat Pertanyaan dan Jawaban :</h5>
                        <br>
```

```
<br>
                            <table>
                          <?php
                          // $tinggi = max($arr_hasil)."<hr>";
                          // $kode_tinggi = array_search($tinggi, $arr_hasil);
                          $tinggi = array_keys($arr_hasil,max($arr_hasil));
                          $nom = 0;
                          foreach ($gejala as $k => $v) {
                           if($v!=""){ ?>
                           <tr><td><?php echo ++$nom ?>.  </td><td>
<?php echo $arr_gejala1[$v]; ?> </td></tr>
                          <?php }
                         } ?>
                        </table>
                          <br>
                          <br>
                           <h5>Kesimpulan :</h5>
                           <br>
                            <h6> <strong><?php echo
$arr_penyakit1[$tinggi[0]][0];?></strong> dengan Nilai : <?php echo
$arr_hasil[$tinggi[0]];?></h6>
                            <p style="text-align: justify;"><?php echo
$arr_penyakit1[$tinggi[0]][1];?> </p>
                       <!-- sini -->
                          </div>
                          <div class="col-sm-2"></div>
                        </div>
                        <div class="col-sm-12">
                         <div class="text-center m-t-20">
```

```
                        <a href="index.php">
                         <button class="btn btn-primary waves-effect waves-light m-
r-10">Selesai</button>
                        </a>
                       </div>
                     </div>
                    </div>
                                  </div>
                                 </div>
                                 <!-- Default card end -->
                               </div>
                              </div>
                             </div>
                             <!-- Page body end -->
                            </div>
                           </div>
                           <!-- Main-body end -->
                          </div>
                         </div>
                        </div>
                       </div>
                     </div>
<?php include 'footer.php'; ?>
<script type="text/javascript">
  $(document).ready(function() {
   $('#ben').DataTable({
      responsive: tdrue,
   });
});
```

```
</script>

</body>

<!-- Mirrored from colorlib.com//polygon/adminty/default/menu-horizontal-fixed.html by HTTrack Website Copier/3.x [XR&CO'2014], Thu, 13 Jun 2019 11:57:12 GMT -->

</html>
```

### 4.2.2 Pembahasan *Interface* / Antarmuka Program

Pada tahapan ini akan dijelaskan mengenai Interface / Antarmuka Program sehingga pengguna dapat lebih memahami penggunaan program serta menampilkan bentuk tampilan dari website sistem pakar yang telah dibangun. Berikut adalah interface dari sistem pakar deteksi penyakit pada tanaman semangka :

1. **Beranda Pengunjung**

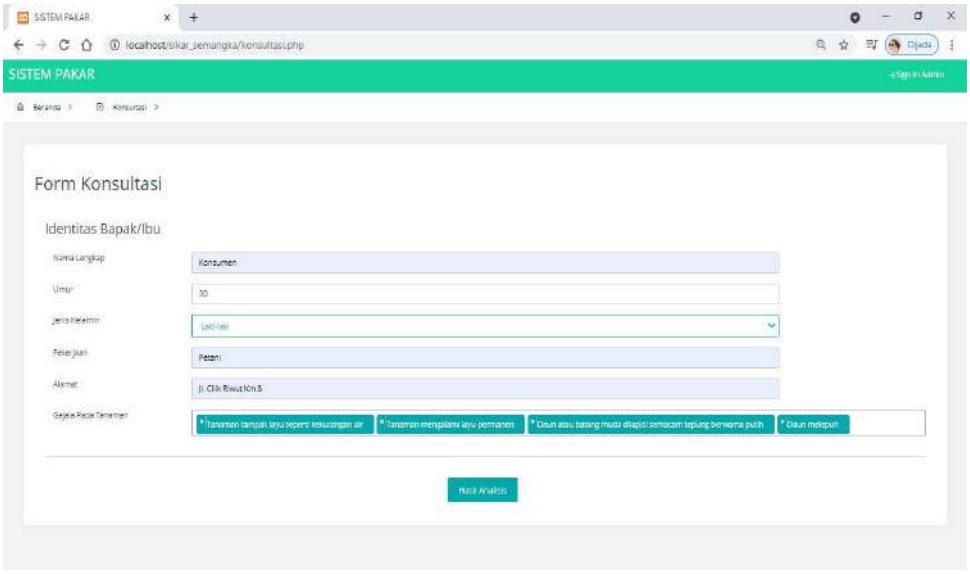



**Gambar 17.Beranda Pengunjung**

Tampilan diatas adalah halaman beranda yang akan muncul ketika pengunjung pertama kali masuk ke dalam website. Pada halaman beranda terdapat fitur konsultasi yang dapat digunakan

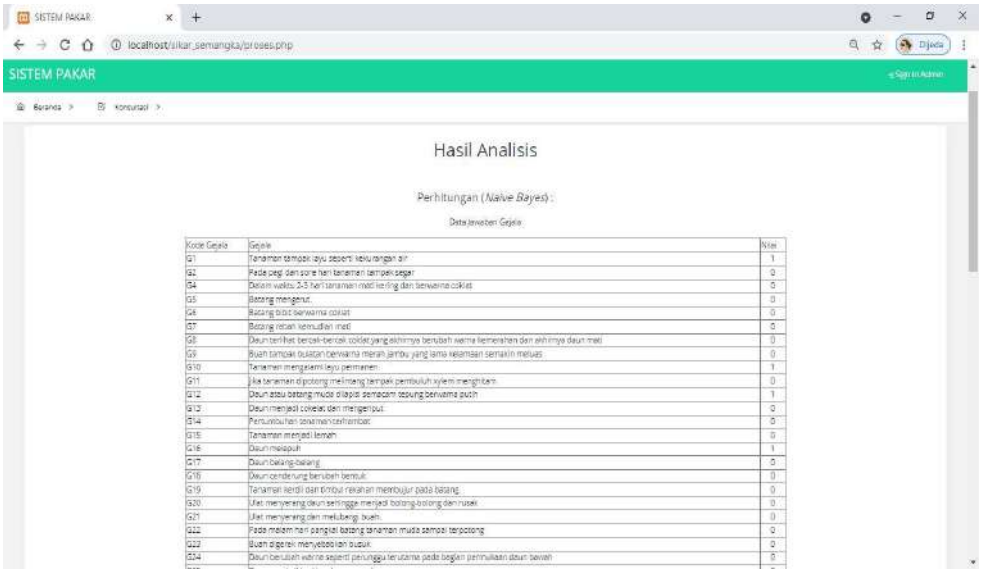
pengunjung untuk melakukan konsultasi mengenai penyakit pada tanaman semangka.

**2. Fitur Konsultasi**

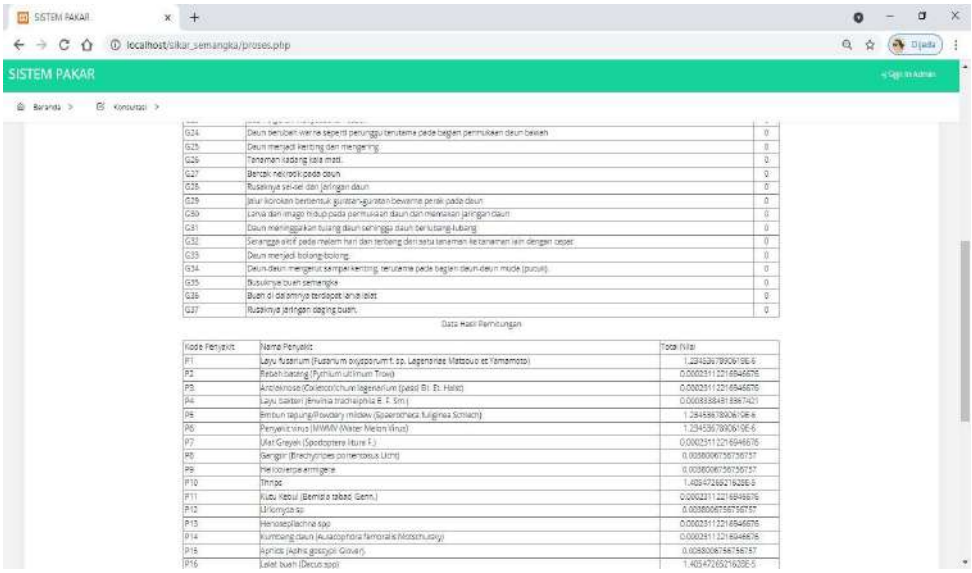



**Gambar 18.Fitur Konsultasi**

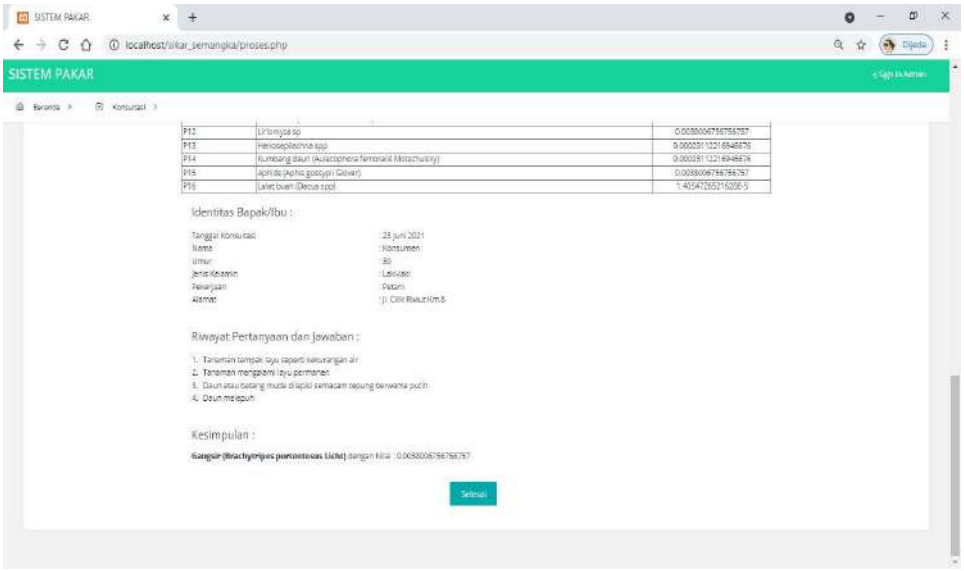
Tampilan diatas adalah halaman konsultasi, dimana pengunjung memasukkan identitas dan memilih gejala yang dialami oleh tanaman semangka. Jika data sudah terisi, maka pengunjung dapat mengklik tombol hasil analisis untuk mendapatkan hasil dari analisisnya. Berikut adalah hasil dari analisis yang diberikan sistem berdasarkan perhitungan menggunakan metode naive bayes.




**Gambar 4.3 Hasil Analisis**



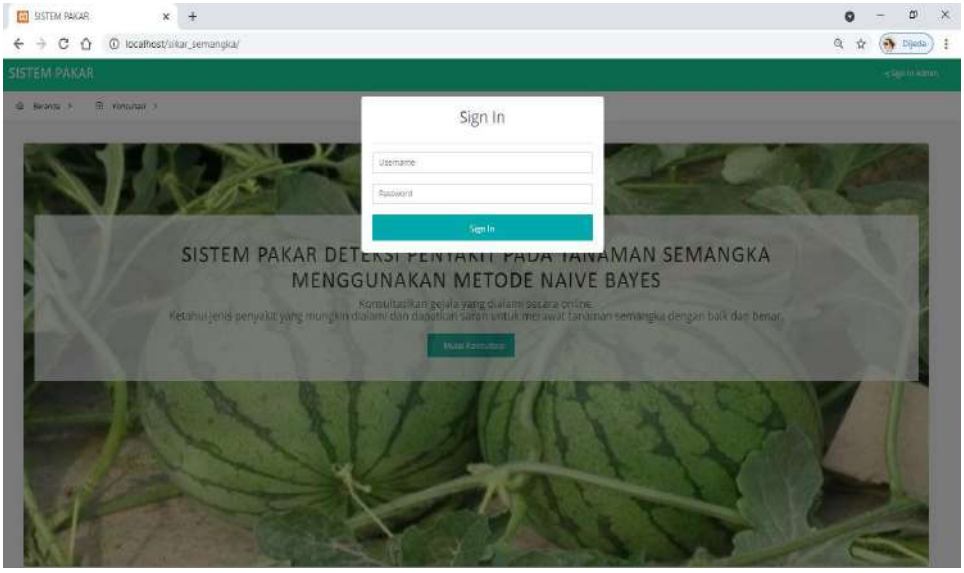

**Gambar 19.Hasil Analisis Lanjutan(2)**




**Gambar 20.Hasil Analisis Lanjutan(3)**

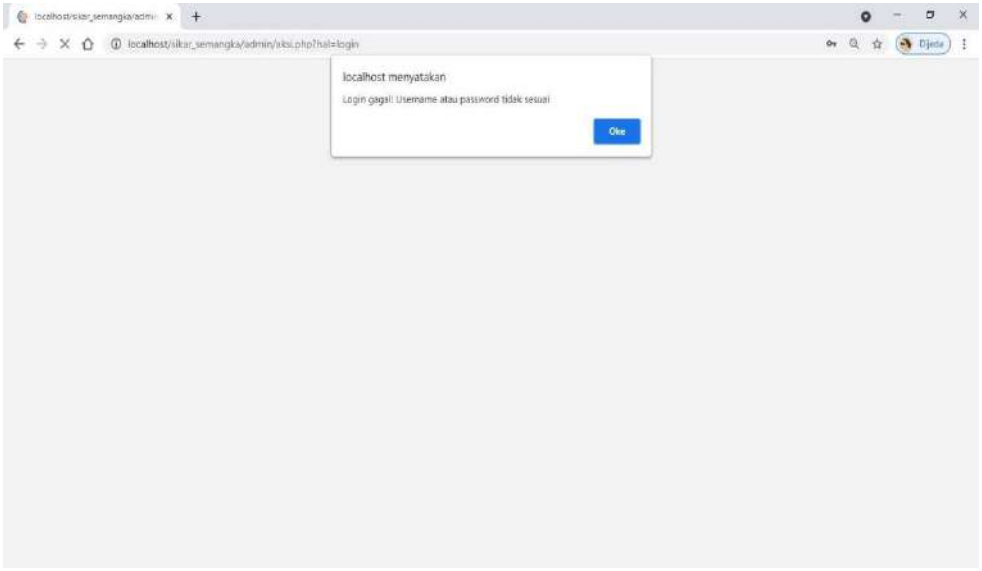
Dapat dilihat hasil dari analisis yang diberikan oleh sistem berdasarkan data gejala yang dipilih oleh pengunjung. Pengunjung dapat mengklik tombol selesai untuk dapat kembali ke halaman beranda.

3. **Fitur Sign In**

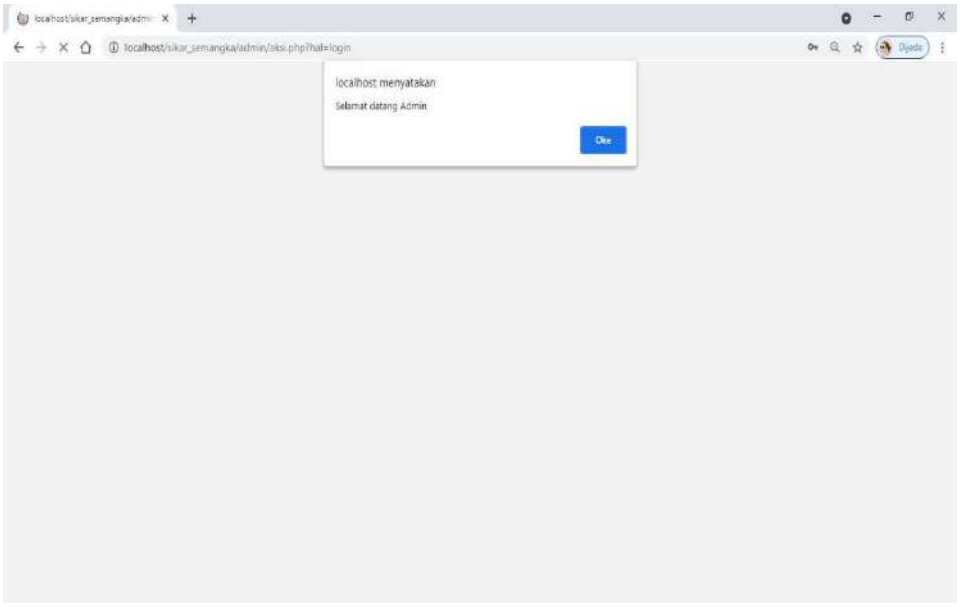



**Gambar 21.Fitur Sign In**

Tampilan diatas adalah halaman Sign In untuk admin dimana admin harus memasukkan username dan password yang benar untuk dapat masuk ke dalam sistem. Berikut adalah pemberitahuan yang muncul ketika admin gagal atau berhasil Sign In.

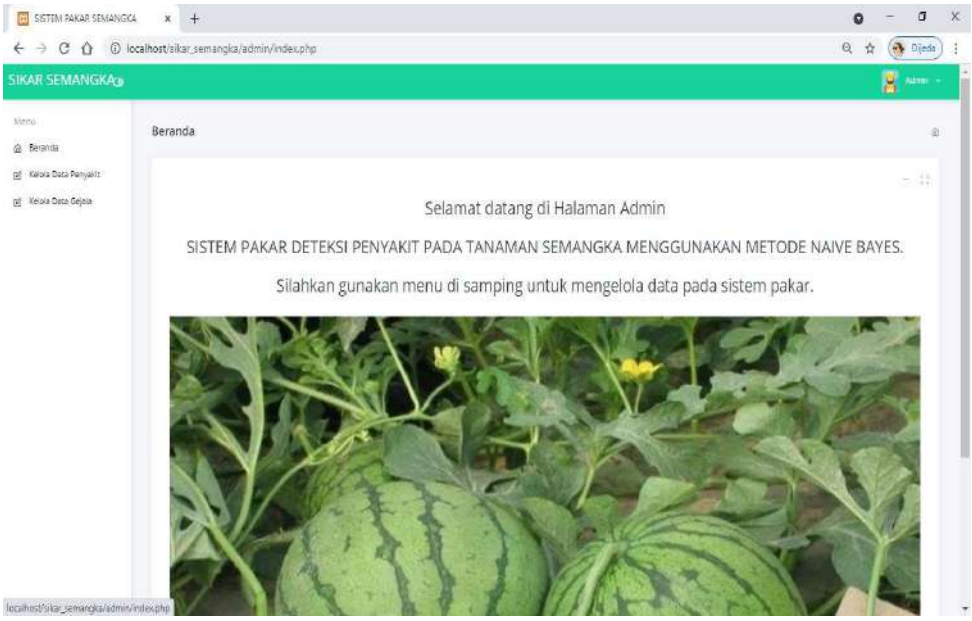



**Gambar 22.Sign In Gagal**




**Gambar 23.Sign In Berhasil**

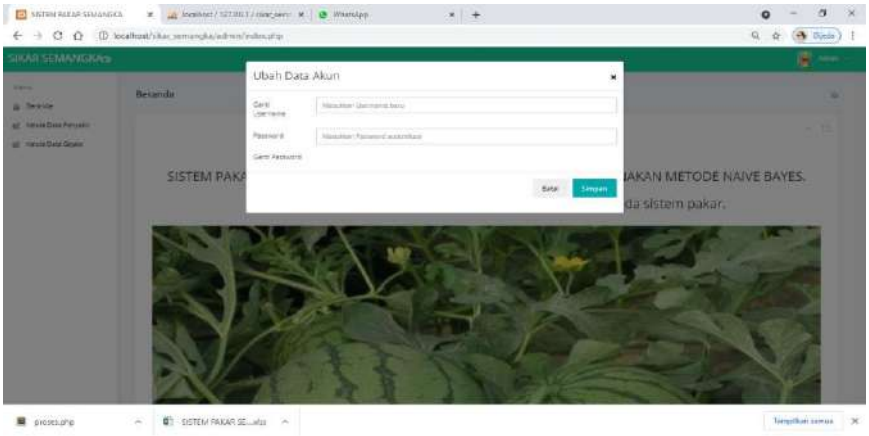
**4. Beranda Admin**




**Gambar 24.Fitur Beranda Admin**

Tampilan diatas adalah halaman beranda yang muncul pertama kali ketika admin berhasil masuk ke dalam sistem. Pada halaman beranda ini terdapat 4 fitur utama yang bisa dipilih admin yaitu beranda, kelola akun, kelola data penyakit dan kelol data gejala.

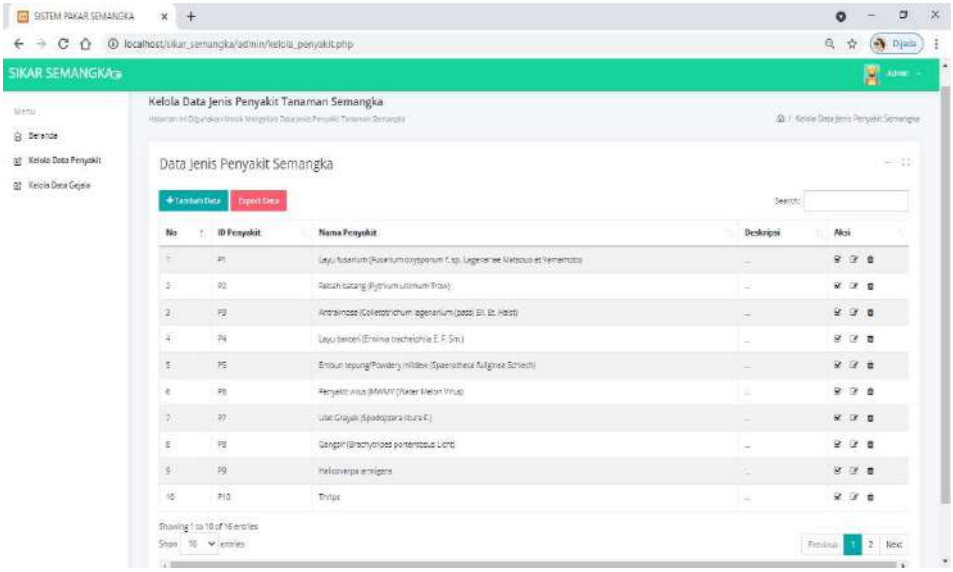
**5. Fitur Kelola Akun**




**Gambar 25.Fitur Kelola Akun**

Tampilan diatas adalah halaman kelola akun dimana admin dapat mengganti username dan password yang ada.

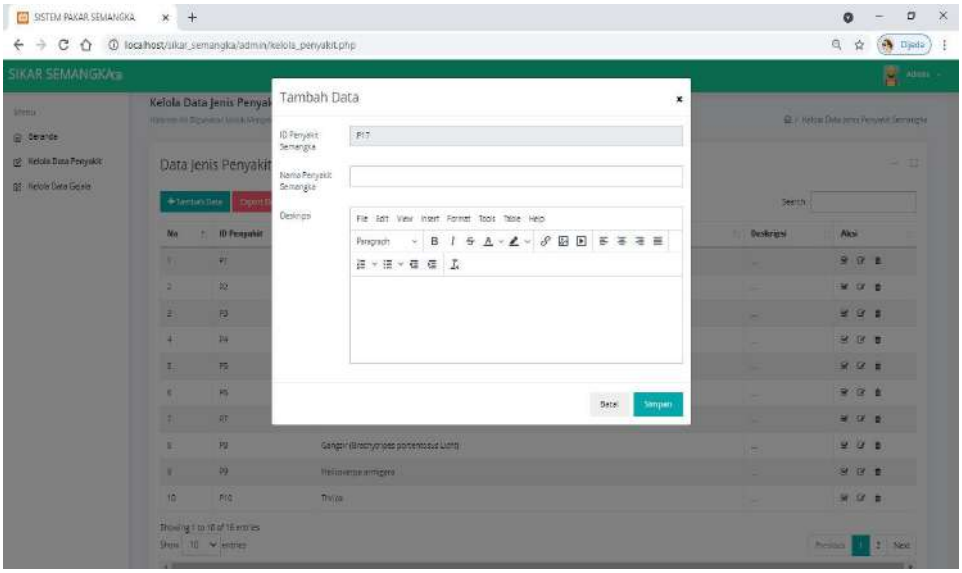
## 6. Fitur Kelola Data Penyakit



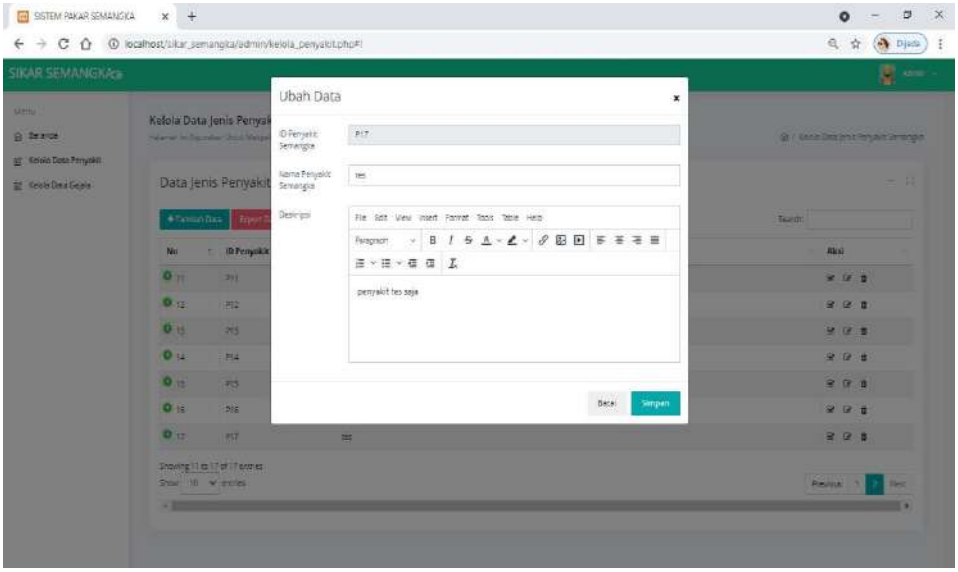

**Gambar 26.Kelola Data Penyakit**

Tampilan diatas adalah halaman kelola data penyakit dimana admin dapat menambah, mengubah, menghapus atau mengeksport data penyakit yang ada.

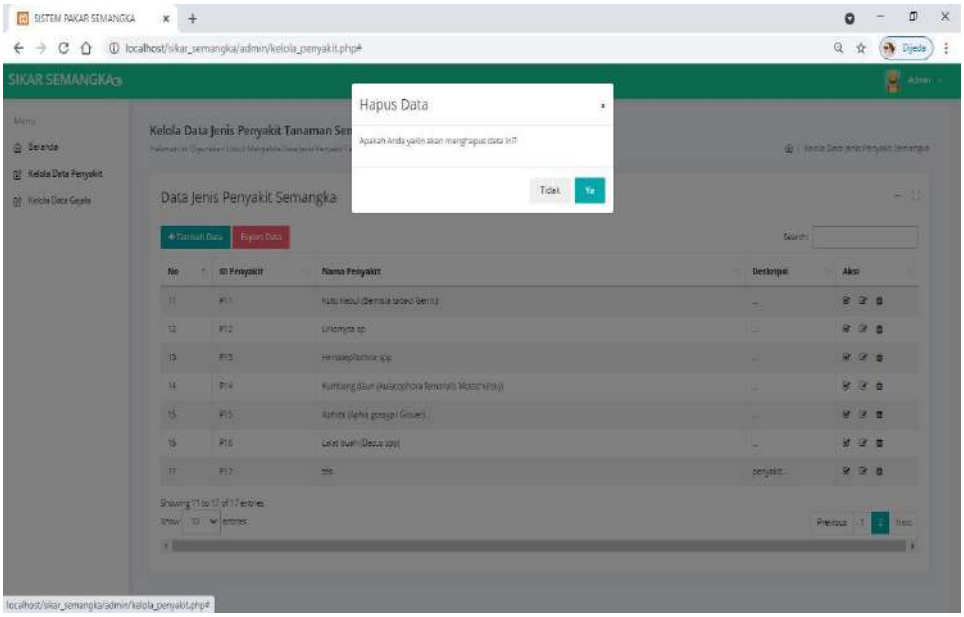
Berikut adalah tampilan dari fitur tambah, ubah, hapus dan export yang ada pada menu kelola data penyakit.



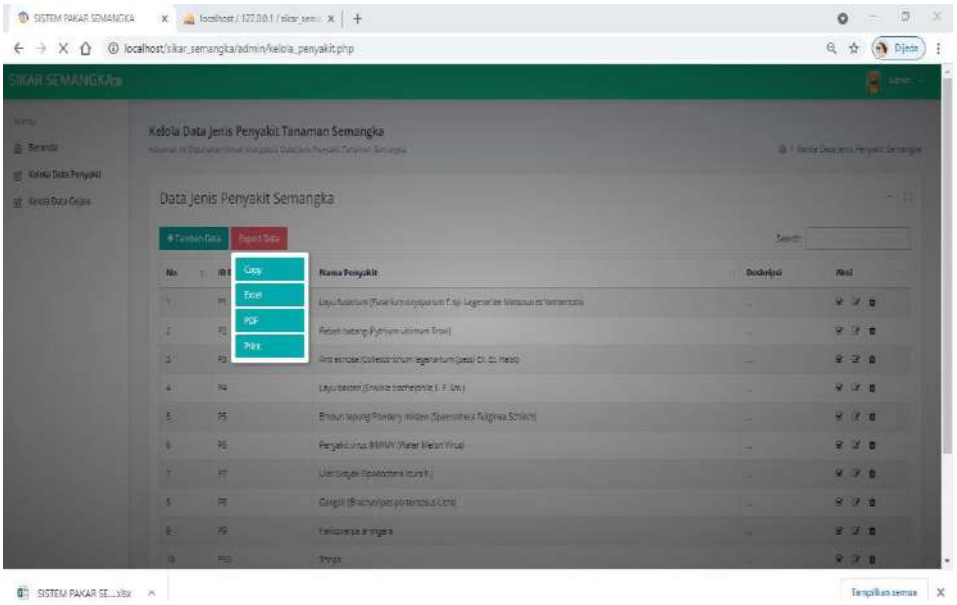

**Gambar 27. Kelola Data Penyakit (Tambah)**




**Gambar 28.Kelola Data Penyakit (Ubah)**

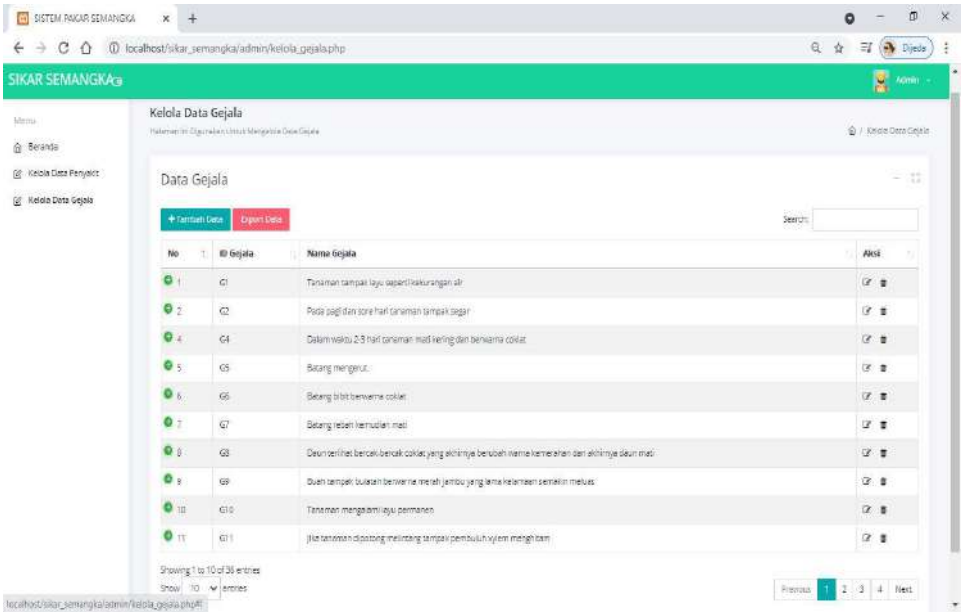



**Gambar 29.Kelola Data Penyakit (Hapus)**



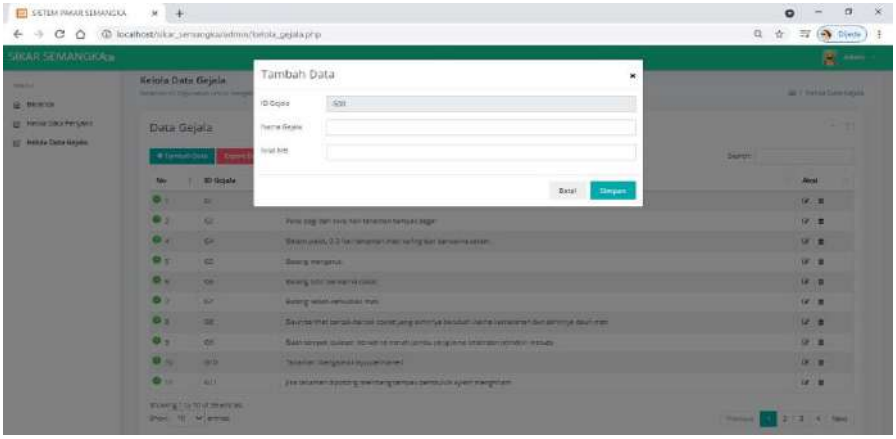

**Gambar 30.Kelola Data Penyakit (Export)**

## 7. Fitur Kelola Data Gejala

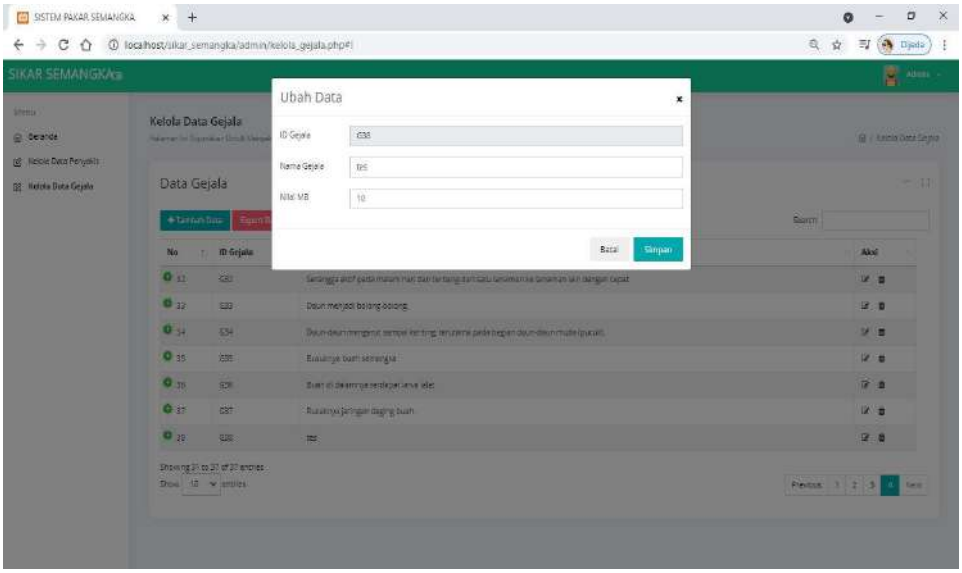



**Gambar 31.Kelola Data Gejala**

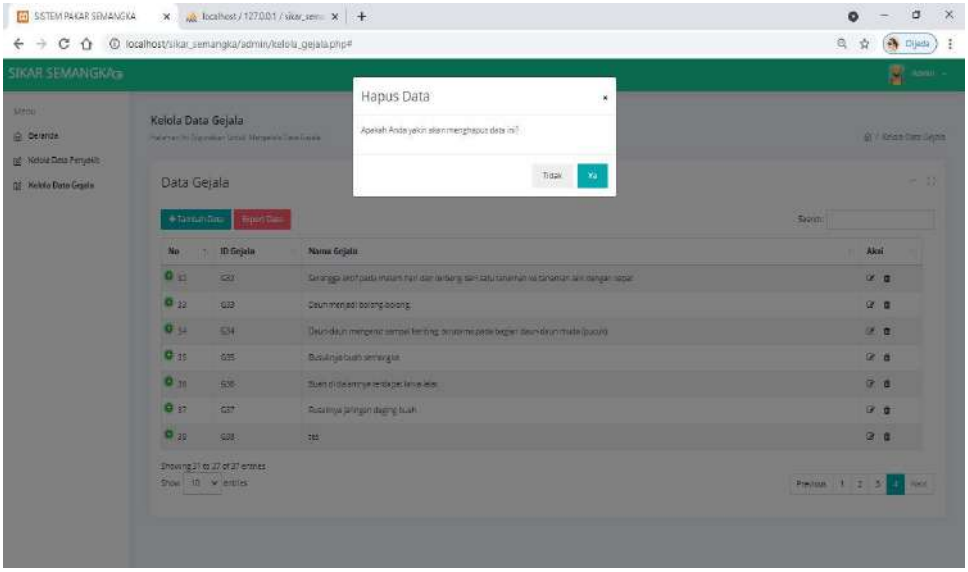
Tampilan diatas adalah halaman kelola data gejala dimana admin dapat menambah, mengubah, menghapus atau mengeksport data gejala yang ada. Berikut adalah tampilan dari fitur tambah, ubah, hapus dan export yang ada pada menu kelola data gejala.



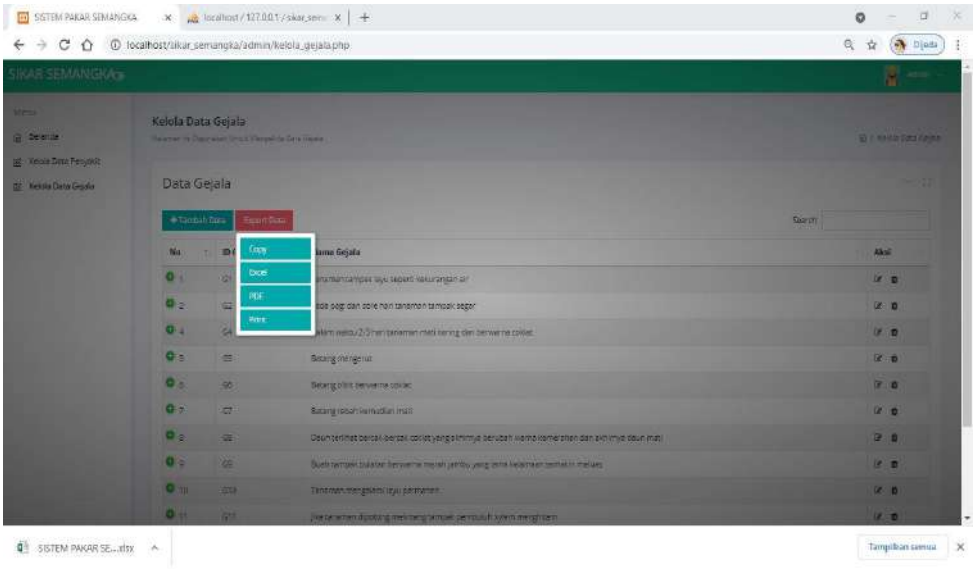

**Gambar 32. Kelola Data Gejala (Tambah)**




**Gambar 33.Kelola Data Gejala (Ubah)**

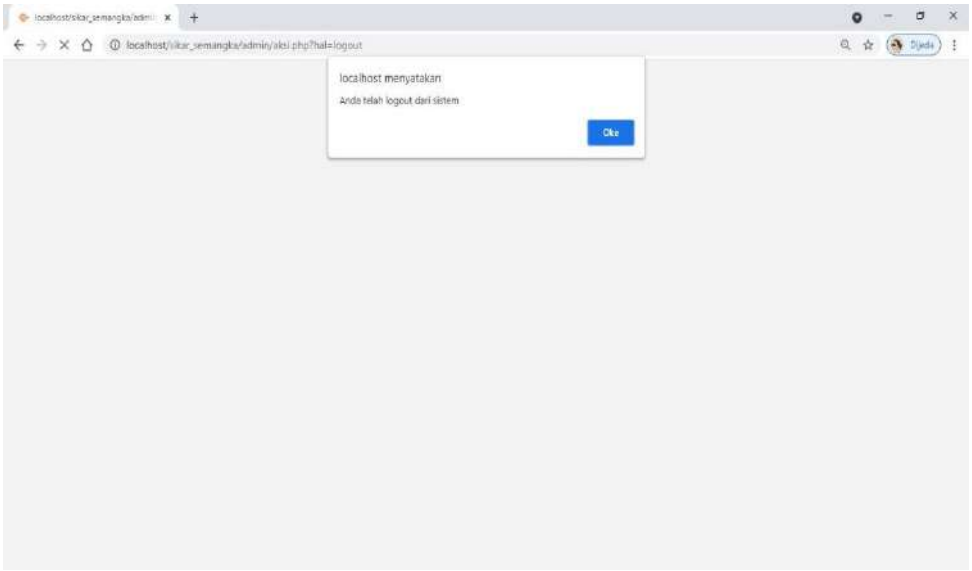



**Gambar 34.Kelola Data Gejala (Hapus)**




**Gambar 35.Kelola Data Gejala (Export)**

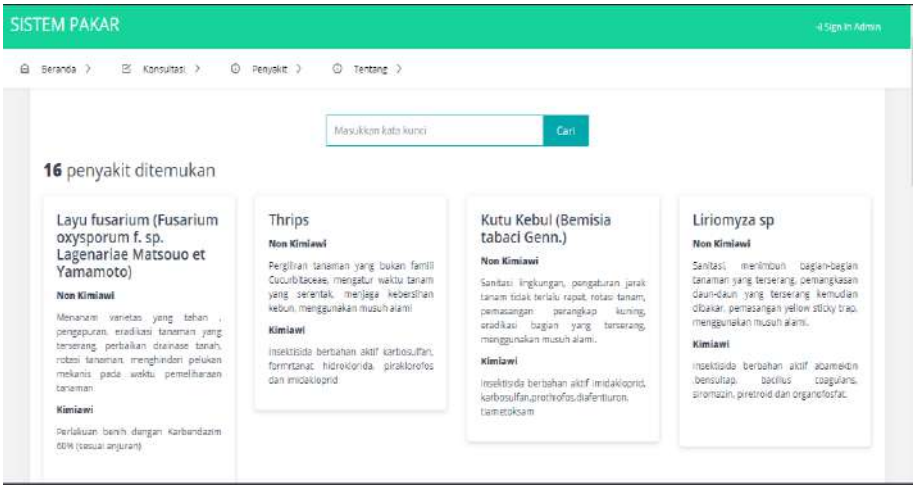
## 8. Fitur SignOut




**Gambar 36.Informasi SignOut**

Tampilan diatas adalah informasi yang diberikan oleh sistem jika admin berhasil Signout. Setelah admin Sign Out, sistem akan kembali menuju ke tampilan beranda.

## 9. Informasi Penyakit




**Gambar 37.Informasi Penyakit**

Tampilan diatas adalah informasi nama-nama penyakit dan solusi pada tanama semangka

# BAB V

# KESIMPULAN DAN SARAN

## 5.1 Kesimpulan

Berdasarkan hasil analisis dan implementasi dari sistem pakar yang dibuat penulis, didapat kesimpulan sebagai berikut :

1. Sistem Pakar dibangun menggunakan bahasa pemrograman PHP, database MySQL dan Xampp sebagai server.
2. Data penyakit didapatkan dari Pakar yaitu Instansi BPTP.
3. Analisis sistem yang digunakan yaitu DFD untuk menggambarkan alur sistem dan ERD sebagai gambaran desain database.
4. Dalam penelitian ini dihasilkan sebuah Sistem Pakar berbasis website yang dapat mendeteksi penyakit pada tanaman semangka.
5. Penggunaan metode naive bayes untuk menentukan probabilitas penyakit pada tanaman semangka berdasarkan gejala yang dipilih dapat diterapkan di dalam sistem.

## 5.2 Saran

Dari hasil pembahasan dan kesimpulan, penulis berharap agar sistem pakar deteksi penyakit pada tanaman semangka ini dapat lebih baik. Berikut saran dari penulis :

1. Ditambahkan algoritma lainnya sebagai perbandingan perhitungan sehingga hasil yang didapat lebih akurat.
2. Sistem dapat dikembangkan ke dalam versi android
3. Sistem dapat dikembangkan lebih dinamis sehingga perhitungan algoritma untuk tiap gejala bisa disesuaikan.

## DAFTAR PUSTAKA


Agustini, S. (2020, November Senin). Tanaman Buah Semangka.

Al-Bahra Bin, L. (2013). *Analisis dan Desain Sistem Informasi.* Yogyakarta: Graha Ilmu.

Buana. (2014). *Pemrograman Database MySQL.* Yogyakarta: MediaKom.

Fathansyah. (2015). *Basis Data.* Bandung: Informatika.

Handayani, M., Taufiq, & Soegiarto. (2016). Sistem Pakar Diagnosa Penyakit Tanaman Semangka Menggunakan Metode Dempster Shafer Berbasis Web. *Program Studi Sistem Informasi, STMIK Banjarbaru, 12, No. 1*, 1243-1250.

Haughee, E. (2013). *Sublime Text Starter.* Birmingham: PACKT Publishing.

Hikmah, & dkk. (2015). *Cara Cepat Membangun Website Dari Nol Studi Kasus :Web Dealer Motor.* Yogyakarta : CV. Andi Offset.

Irnawati, O., & Listianto, G. B. (2018). Metode Rapid Application Development (RAD) pada Perancangan Website Inventory PT. SARANA ABADI MAKMUR BERSAMA (S.A.M.B) JAKARTA. *Manajemen Informatika, AMIK BSI Bekasi, 6 No.2* , 12-18.

Jauhari, & Hidayatullah. (2015). *Pemrograman Web.* Bandung: Informatika.

Komputer, W. (2012). *Panduan Aplikatif & Solusi (PAS) Mudah Membuat Portal Berita Online dengan PHP dan MYSQL.* Yogyakarta: C.V Andi OFFSET.

Kosasi, S. (2014). Sistem Pakar Diagnosa Hama dan Penyakit Tanaman Semangka Menggunakan Metode Certainty Factor. *Jurusan Sistem Informasi, STMIK Pontianak, 108-1012*, 7.

Kustiyahningsih, Y. (2011). *Pemrograman Basis Data Berbasis Web Menggunakan PHP dan MySQL.* Yogyakarta: Graha Ilmu.

Laksono, R. A. (2018). Pengujian Efektivitas Tipe Pemangkasan Terhadap Produksi Tiga Varietas Semangka Pada Hidroponik Sistem Fertigasi (Drip Irrigation). *Jurnal Ilmiah Pertanian PASPALUM, 6 No. 2*, 103-113.

Mariana, A. w. (2019). Sistem Pakar Diagnosis Penyakit Tanaman Semangka Menggunakan Metode Certainty Factor. *JATI(Jurnal Mahasiswa Teknik Informatika), 3*, 7.

Nugroho, B. (2010). *Panduan Membuat Program Toko dengan PHP, MySQL dan Dreamweaver*. Palembang: Maxikom.

Oktavia, N. D. (2015). Penggunaan Pestisida dan Kandungan Residu Pada Tanah Dan Buah Semangka (Citrullus Vulgaris,Schard) .

Pati, M., Defit, S., & Nurcahyo, G. W. (2020). Sistem Pakar Dengan Metode Forward Chaining Untuk Diagnosis Penyakit Dan Hama Tanaman Semangka. *Jurnal Sistem Informasi Dan Theknologi*, *2*, 102-107.

Prasetyo. (2012). *Data Mining Konsep dan Aplikasi Menggunakan Matlab.* Yogyakarta: Andi Offset.

Pressman, R. S. (2010). *Software Engineerring : A Practitioner's Approach.* New York: McGraw-Hill.

Purbadian. (2016). *Trik Cepat Membangun Aplikasi Berbasis Web dengan Framwork CodeIgniter.* Yogyakarta : CV Andi Offset.

Raschka, S. (2014). Naive Bayes and Text Classification I. *Introduction and Theory*, 1-20.

Rosario, A. (2013). *Web Programming Power Pack.* Yogyakarta: Mediakom.

Sibero, A. F. (2013). *Web Programming Power Pack.* Yogyakarta: Mediakom.

Sifaunajah, A., Hariono, T., & Chumaidi, M. (2018). Penerapan Certainty factor pada Rancang Bangun Sistem Pakar Deteksi Organisme Pengganggu Tanaman (OPT) Semangka. *Jurnal Ilmu Komputer dan Desain Komunikasi Visual*.

Simarmata, J. (2010). *Rekayasa WEB.* Yogyakarta: CV Andi Offset.

Sukamto, R. A. (2018). *Rekayasa Perangkat Lunak* (Revisi ed.). Bandung: Informatika.

Sukamto, R. A., & Shalahuddin, M. (2013). *Rekayasa Perangkat Lunak.* Bandung: Informatika.

Sukamto, R. A., & Shalahuddin, M. (2013). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi.* Bandung: Informatika.

Supono. (2016). *Pemrograman Web Dengan Menggunakan PHP dan Framework Codeigniter.* Yogyakarta: CV Budi Utama.

T.Sutojo, E. m. (2011). Yogyakarta: Andi Offset.

Turban. (2001). *Decision Support System and Intelligent System ( Sistem Pendukung Keputusan dan Sistem Cerdas).* Yogyakarta: Andi.

Wulandari, T., Pahu, G. Y., Sari, D. N., & Isnandar, S. (2018). Perancangan Sistem Pakar Deteksi Pertumbuhan Tanaman Semangka Berbasis Website Dengan Certainty Factor. *Jurnal TAM (Technology Acceptance Model), 9, No.2*, 134-141.

Yuhefizar. (2013). *Cara Mudah & Murah Membangun & Mengelola Website.* Yogyakarta : Graha Ilmu.