SISTEM PAKAR DIAGNOSIS PENYAKIT TANAMAN MENTIMUN DENGAN MENGGUNAKAN METODE *CERTAINTY FACTOR* BERBASIS WEB

TUGAS AKHIR

Disusun Untuk Memenuhi Syarat Kelulusan Program Strata I pada
Sekolah Tinggi Manajemen Informatika dan Komputer
(STMIK) Palangkaraya

OLEH

IRGI OKTAVIANUS EZINOTU
NIM C1855201048
PROGRAM STUDI TEKNIK INFORMATIKA

SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA
2022

SISTEM PAKAR DIAGNOSIS PENYAKIT TANAMAN MENTIMUN DENGAN MENGGUNAKAN METODE *CERTAINTY FACTOR* BERBASIS WEB

TUGAS AKHIR

Disusun Untuk Memenuhi Syarat Kelulusan Program Strata I pada
Sekolah Tinggi Manajemen Informatika dan Komputer
(STMIK) Palangkaraya

OLEH

IRGI OKTAVIANUS EZINOTU
NIM C1855201048
PROGRAM STUDI TEKNIK INFORMATIKA

SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA
2022

LEMBAR PERNYATAAN

Yang bertanda tangan dibawah ini :

Nama Mahasiswa : IRGI OKTAVIANUS EZINOTU

NIM : C1855201048

Menyatakan bahwa Tugas Akhir dengan judul :

SISTEM PAKAR DIAGNOSIS PENYAKIT TANAMAN MENTIMUN DENGAN MENGGUNAKAN METODE *CERTAINTY FACTOR* BERBASIS WEB

Adalah hasil karya saya dan bukan merupakan duplikasi sebagian atau seluruhnya dari karya orang lain, kecuali bagian yang sumber informasi dicantumkan.

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.

Palangka Raya, 20 Juli 2022

Yang membuat peryataan,

IRGI OKTAVIANUS EZINOTU

PERSETUJUAN

SISTEM PAKAR DIAGNOSIS PENYAKIT TANAMAN MENTIMUN DENGAN MENGGUNAKAN METODE *CERTAINTY FACTOR* BERBASIS WEB

Tugas Akhir ini Telah Disetujui Untuk Diujikan Pada
Tanggal 22 Juli 2022

Pembimbing I,	Pembimbing II,
Sulistyowati, S.Kom., M.Cs.	Christia Putra, S.Kom., M.MSI.
NIK. 198212162007002	NIK. 198512232021102

Mengetahui

Ketua STMIK Palangkaraya,

Suparno, M.Kom
NIK. 196901041995105

PENGESAHAN

SISTEM PAKAR DIAGNOSIS PENYAKIT TANAMAN MENTIMUN DENGAN MENGGUNAKAN METODE *CERTAINTY FACTOR* BERBASIS WEB

Tugas Akhir ini Telah Diujikan, Dinilai, dan Disahkan
Oleh Tim Penguji pada Tanggal 22 Juli 2022

Tim Penguji Tugas Akir :

1. …..
   Ketua

2. …..
   Sekretaris

3. …..
   Anggota

4. …..
   Anggota

5. …..
   Anggota

MOTTO DAN PERSEMBAHAN

(2 Korintus 5 : 7)

--Sebab hidup kami ini adalah hidup

karena percaya, buukan karena

melihat--

Tugas Akhir ini saya persembahkan untuk :

- Ayah dan Ibu tercinta, yang tidak kenal lelah berdo'a mendidik dan membesarkan saya hingga sekarang.
- Keluarga besar yang telah memberikan dukungan dan doa sehingga dapat menyelesaikan tugas akhir ini dengan tepat waktu.
- Pacar dan teman-teman yang selalu mendukung untuk penyelesainya penulisan ini.
- Dosen-dosen STMIK Palangka Raya, yang membimbing dari semester satu hingga saat ini.

INTISARI

Irgi Oktavianus Ezinotu, C1855201048, 2022, *Sistem Pakar Diagnosis Penyakit Tanaman Mentimun Dengan Menggunakan Metode Certainty Factor Berbasis Web*, Pembimbing I Sulistyowati, S.Kom., M.Cs.Pembimbing II Christia Putra, S.Kom., M.MSI.

Sistem pakar adalah sistem yang menyimpan pengetahuan dan penalaran para pakar. Sistem pakar memiliki kemampuan untuk memecahkan masalah layaknya seorang pakar. Sistem pakar diagnosis penyakit tanaman mentimun dengan menggunakan metode *certainty factor* ini dibuat untuk memberikan informasi mengenai gejala dan penyakit tanaman mentimun kepada petani yang tidak memiliki informasi terkait penyakit tanaman mentimun.

Manfaat yang diperoleh dari sistem pakar yang mampu melakukan diagnosis dengan cepat, tepat terhadap gejala penyakit yang ditimbulkan diharapkan mampu membantu para petani dalam mengantisipasi kemungkinan terburuk akibat serangan penyakit pada mentimun.

Metode yang digunakan pada penelitian ini meliputi metode observasi, metode wawancara, metode kepustakaan, metode dokumentasi. Teknik dan alat bantu analisis menggunakan *Tools Unifield Modeling Language* (UML),

Hasil dari penelitian menghasilkan sebuah sistem pakar yang mampu mendiagnosis penyakit tanaman mentimun berdasarkan data yang didapat dari pakar. Hasil penelitian ini siap dipublikasikan untuk memberikan informasi kepada petani yang tidak memiliki informasi tentang gejala dan penyakit tanaman mentimun dengan mudah dan terjangkau.

Kata Kunci : Mentimun, *Certainty Factor*, Penyakit, Sistem Pakar.

ABSTRACT

Irgi Oktavianus Ezinotu, C1855201048, 2022, Expert System for Diagnosis of Cucumber Plant Diseases Using Web-Based Certainty Factor Method, Supervisor I Sulistyowati, S.Kom., M.Cs. Supervisor II Christia Putra, S.Kom., M.MSI.

An expert system is a system that stores the knowledge and reasoning of experts. Expert systems have the ability to solve problems like an expert. This expert system for diagnosis of cucumber plant diseases using the Certainty Factor method was created to provide information about the symptoms and diseases of cucumber plants to farmers who do not have information related to cucumber plant diseases.

The benefits obtained from a system that is able to make a quick, precise diagnosis of the symptoms of the disease caused are expected to be able to help farmers in anticipating the worst possible consequences of disease in cucumbers.

The methods used in this study include observation methods, interview methods, library methods, documentation methods. Techniques and analytical tools using the Tools Unifield Modeling Language (UML),

The results of the study resulted in a system that was able to diagnose cucumber plant diseases based on data obtained from experts. The results of this study are ready to provide easy and affordable information to farmers who do not have information about symptoms and diseases of cucumber plants.

Keywords: Cucumber, Certainty Factor, Disease, Expert System.

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Tuhan yang Maha Esa atas berkat dan penyertaannya sehingga penulis bisa menyelesaikan tugas akhir dengan judul "Sistem Pakar Diagnosis Penyakit Tanaman Mentimun Dengan Menggunakan Metode *Certainty Factor* Berbasis Web*."*

Penulis memahami tanpa bantuan, doa dan bimbingan dari semua orang akan sangat sulit untuk menyelesaikan Tugas Akhir ini. Maka dari itu penulis ingin mengucapkan terima kasih kepada :

1. Krisyetno, S.ST. selaku pakar yang memberikan hasil dari penilitian pada sistem Tugas Akhir ini.
2. Sulistyowati S.Kom., M.Cs selaku dosen pembimbing I
3. Christia Putra, S.Kom., M.MSI selaku dosen pembimbing II

   Penulis mengharapkan kritik dan saran membangun demi penyempurnaan Tugas Akhir ini sehingga bermanfaat bagi semua pihak yang membutuhkan, sekian dan terima kasih.

Palangka Raya, 20 Juli 2022

Penulis

DAFTAR ISI

LEMBAR PERNYATAAN ii
PERSETUJUAN iii
PENGESAHAN iv
MOTTO DAN PERSEMBAHAN v
INTISARI vi
ABSTRACT vii
KATA PENGANTAR viii
DAFTAR ISI ix
DAFTAR TABEL xi
DAFTAR GAMBAR xii
DAFTAR LAMPIRAN xiii
BAB I PENDAHULUAN 1
1.1 Latar Belakang Masalah 1
1.2 Perumusan Masalah 3
1.3 Batasan Masalah 3
1.4 Tujuan dan Manfaat 4
1.5 Sistematika Penulisan 5
BAB II LANDASAN TEORI 6
2.1 Tinjauan Pustaka 6
2.2 Kajian Teori 11
a. Materi 11
b. Penyakit Tanaman Mentimun 19
c. Permodelan Sistem 20
BAB III METODE PENILITIAN 30
3.1 Lokasi Penelitian 30
3.2 Perencanaan Alat dan Bahan 30
3.3 Jenis Penelitian 32
3.4 Teknik Pengumpulan Data 34
3.5 Analisis Kebutuhan 35
a. Analisis Data 35
b. Analisis Proses 41
c. Analisis Kelemahan 42
3.6 Desain Sistem 43

a. Desain Proses 43
b. Desain Perangkat Lunak 58
c. Desain Basis Data 61
BAB IV HASIL DAN PEMBAHASAN 64
4.1 Hasil 64
4.1.1 Implementasi 64
a. Tampilan Halaman Awal 64
b. Tampilan Halaman *Admin* 65
c. Tampilan Halaman *User*/Pengunjung 69
4.1.2 Pengujian Sistem 72
4.1.3 Perhitungan Certainty Factor 73
4.2 Pembahasan 76
4.2.1 Pembahasan Hasil Responden 76
KESIMPULAN DAN SARAN 79
5.1 Kesimpulan 79
5.2 Saran 79
DAFTAR PUSTAKA 80

DAFTAR TABEL

Tabel 1. Penelitian Yang Relevan 7
Tabel 2. Logika Metode *certainty factor* 14
Tabel 3. Simbol dalam *Use Case* Diagram 23
Tabel 4. Simbol dalam *Activity* Diagram 24
Tabel 5. Simbol dalam *Sequence* Diagram 25
Tabel 6. Simbol *Class* Diagram 27
Tabel 7. Spesifikasi Perangkat Keras 30
Tabel 8. Spesifikasi Perangkat Lunak 31
Tabel 9. Data Jenis Penyakit mentimun 35
Tabel 10. Tabel gejala penyakit mentimun 36
Tabel 11. Tabel Pakar Penyakit Mentimun 37
Tabel 12. Tabel Pembobotan Nilai User 38
Tabel 13. Tabel Sampel Nilai Diagnosa Busuk Daun 39
Tabel 14. Kegiatan Aktor dan Interaksinya dengan Sistem 43
Tabel 15. Tabel Spesifikasi Admin 62
Tabel 16. Tabel Data Gejala 62
Tabel 17. Tabel data penyakit 63
Tabel 18. Tabel Basis Pengetahuan 63
Tabel 19. Tabel Uji Sistem 72
Tabel 20. Tabel Nilai CF Gejala pada Penyakit Busuk daun 73
Tabel 21. Hasil Responden Pertanyaan Ke 1 76
Tabel 22. Hasil Responden Pertanyaan Ke 2 77
Tabel 23. Hasil Responden Pertanyaan Ke 3 77
Tabel 24. Hasil Responden Keseluruhan 78

DAFTAR GAMBAR

Gambar 1. Model Prototype 32
Gambar 2. Rancangan Alur Proses Menentukan Diagnosa Mentimun..... 42
Gambar 3. Use Case *System* Yang Dirancang 44
Gambar 4. *Activity Diagram* Diagnosa (Pengunjung) 45
Gambar 5. *Aktivity Diagram* cetak hasil diagnosa (pengunjung) 46
Gambar 6. Activity Diagram Login (Admin) 47
Gambar 7. Activity Diagram Kelola Data Penyakit (Admin)..... 48
Gambar 8. Activity Diagram Kelola Data Gejala (Admin) 49
Gambar 9. Activity Diagram Kelola Data Gejala (Admin) 50
Gambar 10. Activity Diagram Kelola Gejala Penyakit (Admin)..... 51
Gambar 11. Activity Diagram Logout (Admin) 52
Gambar 12. *Sequence Diagram* Diagnosa (Pengujung) 53
Gambar 13. *Sequence Diagram* Cetak Hasil Diagnosa (Pengunjung)..... 54
Gambar 14. *Sequence Diagram* Login (Admin)..... 55
Gambar 15. *Sequence Diagram* Kelola Data Gejala (Admin)..... 55
Gambar 16. *Sequence Diagram* Kelola Data Penyakit (Admin) 56
Gambar 17. *Class Diagram* Sistem Pakar Diagnosis Penyakit Tanaman Mentimun Dengan Menggunakan Metode *Certainty Factor* Berbasis Web..... 57
Gambar 18. Halaman Gejala Penyakit User 58
Gambar 19. Halaman Diagnosa User..... 58
Gambar 20. Halaman *Login* Admin..... 59
Gambar 21. Halaman Menu Admin..... 59
Gambar 22. Halaman Kelola Data Gejala..... 60
Gambar 23. Halaman Kelola Data Kelompok Gejala..... 60
Gambar 24. Halaman Kelola Data Penyakit 61
Gambar 25. Halaman Kelola Data Gejala Penyakit..... 61
Gambar 26. Tampilan Home Gejala Penyakit 64
Gambar 27. Tampilan Halaman Tentang Program 65
Gambar 28. Tampilan Halaman Login (Admin)..... 65
Gambar 29. Tampilan Halaman Menu *(Admin)*..... 66
Gambar 30. Tampilan Halaman Gejala *(Admin)*..... 67
Gambar 31. Tampilan Halaman Kelompok Gejala *(Admin)*..... 67
Gambar 32. Tampilam Halaman Penyakit *(Admin)*..... 68
Gambar 33. Tampilan Halaman Gejala Penyakit *(Admin)*..... 69
Gambar 34. Tampilan Halaman Gejala Penyakit User/Pengunjung..... 69
Gambar 35. Tampilan Halaman Diagnosis Hasil..... 70
Gambar 36. Halaman Cetak Hasil Diagnosis..... 71
Gambar 37. Hasil Analisis Gejala Yang Dipilih..... 74

DAFTAR LAMPIRAN

Lampiran 1. Surat Tugas Pembimbingan Tugas Akhir
Lampiran 2. Lembar Konsultasi Bimbingan Tugas Akhir
Lampiran 3. Surat Keterangan Melakukan Penelitian
Lampiran 4. Surat Keterangan Telah Melakukan Penelitian
Lampiran 5. Lembar Wawancara
Lampiran 6. Lembar Dokumentasi Wawancara
Lampiran 7. Dokumentasi Tempat Penelitian
Lampiran 8. Berita Acara Pengujian Sistem
Lampiran 9. Surat Tugas Pengujian Sidang Tugas Akhir

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Perkembangan pertanian di Indonesia masih menjadi salah satu hal penting sebagai roda penggerak ekonomi negara. Hal ini dikarenakan pertanian dari segi produksi menjadi sektor kedua paling berpengaruh setelah industri pengolahan. Sedangkan bila dibandingkan sektor lainnya pertanian masih berada diposisi teratas selain sektor perdagangan dan sektor konstruksi. Dengan demikian sektor pertanian mampu mengangkat citra Indonesia di mata dunia, terutama sebagai negara agraris yang cukup produktif.

"Sistem pakar merupakan pemikiran dan pengetahuan pakar yang dimasukkan dalam program komputer untuk menyelesaikan suatu masalah. Sistem pakar diterapkan sebagai cara penyimpanan pengetahuan pakar kedalam suatu program sesuai bidang untuk membantu kepentingan masyarakat. Sistem pakar merupakan cabang dari kecerdasan buatan dan bidang ilmu untuk mengadopsi pengetahuan manusia ke dalam komputer yang menghubungkan dasar pengetahuan dengan sistem inferensi dan memberikan hasil yang lebih cepat dari seorang pakar" (Nuryana & Fitriati, 2018).

Mentimun merupakan jenis tanaman semusim yang bersifat menjalar atau memanjat. Peningkatan produksi mentimun dapat dipacu dengan usaha *intensifikasi*, *ekstensifikas*i, dan *diversifikasi* melalui penerapan paket teknologi budidaya secara intensif. Namun terdapat kendala pada budidaya

mentimun ialah adanya penyakit pada tanaman yang menyebabkan gagal panen. Berdasarkan dari observasi, petani sulit mendiagnosa penyakit yang menyerang tanamannya karena sistem yang berjalan masih manual dimana petani yang ingin melakukan konsultasi harus konsultasi ke penyuluh langsung, cara tersebut kurang efisien karena membutuhkan waktu yang lama.

Sistem pakar diagnosis penyakit mentimun dengan menggunakan metode *Certainty Factor* berbasis web ini dibuat untuk memberikan informasi mengenai mentimun kepada petani yang tidak memiliki informasi pada mentimun.

Berdasarkan uraian diatas, penulis berniat membuat sistem pakar yang menerapkan metode inferensi *Certainty Factor*. Sistem pakar ini akan berguna bagi petani mentimun untuk mengenali jenis penyakit yang menyerang tanaman mentimun, sehingga nantinya dapat dilakukan tindakan lanjutan melalui output yang dihasilkan yaitu berupa diagnosa dan solusi penanganannya.

Maka dibangunlah sebuah Sistem Pakar yang penulis beri judul "Sistem Pakar Diagnosis Penyakit Tanaman Mentimun dengan menggunakan Metode *Certainty Factor* Berbasis Web".

1.2 Perumusan Masalah

Berdasarkan latar belakang masalah yang ada maka dapat dirumuskan permasalahan dari penelitian ini "Bagaimana membangun sebuah Web **Sistem Pakar Diagnosis Penyakit Tanaman Mentimun Dengan Menggunakan Metode *Certainty Factor* Berbasis Web?**"

1.3 Batasan Masalah

Agar sistem dan penelitian sesuai dengan tujuan yang ingin dicapakai maka diperlukan batasan masalah agar menjadi sistematis. Batasan-batasan masalah sistem pakar ini adalah sebagai berikut:

a. Dalam sistem ini mengindentifikasi 8 Jenis Penyakit yaitu, Busuk daun, Penyakit embun tepung, Antraknosa, Layu fusarium, Kudis, Buah busuk, Layu bakteri, Virus dan 19 gejala.
b. Aplikasi yang dibuat merupakan aplikasi berbasis web menggunakan bahasa program PHP, Visual studio core, dan DBMS menggunakan MySQL.
c. Metode yang digunakan *Certainty Factor* dengan metode berupa pemodelan MB (tingkat keyakinan) dan MD (tingkat ketidakyakinan)
d. Data diambil dari BPTP Kalimantan Tengah
e. Stakeholder/orang yang berkepentingan untuk mengelola admin web seperti Kepala Balai, penyuluh, untuk kepala balai dan penyuluh. mengakses seperti Login, Halaman gejala penyakit, Kelola data gejala, Kelola kelompok gejala, Kelola data penyakit, Kelola gejala penyakit, Data hasil diagnosa, Logout. Kemudian untuk petani dapat mengakses

pengunjung, seperti Halaman Gejala Penyakit, Diagnosa, Cetak Hasil Diagnosa.

1.4 Tujuan dan Manfaat

a. Tujuan

Tujuan yang ingin dicapai oleh peneliti adalah membangun sistem untuk mendiagnosis penyakit pada tanaman Mentimun.

b. Manfaat

1. Manfaat bagi pengguna

Manfaat yang diharapkan dalam pembuatan sistem ini adalah memberikan solusi dan mempermudah untuk mendeteksi penyakit yang ada pada tanaman Mentimun.

2. Manfaat bagi penulis

Sebagai bentuk pengimplementasian ilmu yang telah didapat selama di bangku perkuliahan dan memberikan informasi terkait penyakit pada tanaman Mentimun dengan menggunakan metode *Certainty Factor* berbasis web.

3. Bagi Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Palangkaraya

Manfaat yang diberikan kepada kampus adalah sebagai penambah literatur pustaka pada perpustakaan STMIK Palangkaraya serta dapat digunakan sebagai referensi dan dokumen akademik dan juga informasi terkait bagaimana penanganan terdiagnosis pada tanaman Mentimun.

1.5 Sistematika Penulisan

Sistematika dari penulisan tugas akhir ini dapat dijelaskan sebagai berikut:

BAB I PENDAHULUAN

Bab ini berisi tentang latar belakang masalah, rumusan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan

BAB II LANDASAN TEORI

Bab ini berisikan tentang tinjauan pustaka berisi penelitian yang relevan, susunan kajian teori disesuaikan dengan tema tugas akhir, isi sesuai dengan penjelasan yang digunakan selama melakukan tugas akhir seperti perangkat yang digunakan (perangkat lunak atau keras).

BAB III METODE PENELITIAN

Pada bab ini menjelaskan tentang subyek penelitian, metode penelitian yang digunakan, Perangkat yang digunakan dan pembangunan sistem yang dibuat.

BAB IV IMPLEMENTASI DAN PEMBAHASAN

Bab ini berisikan implementasi terhadap analisis dan rancangan yang dipaparkan pada bab 3 ke dalam bentuk Bahasa pemrograman. Selain itu bab ini berisi tentang hasil penguji terhadap sistem yang dibangun.

BAB V KESIMPULAN DAN SARAN

Bab ini berisi tentang kesimpulan dan saran yang diperoleh dalam penulisan tugas akhir.

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Untuk membantu dalam proses penelitian ini, penulis melakukan penelitian yang relevan terhadap kajian yang telah dilakukan oleh beberapa pihak sebelumnya, hal ini dimaksudkan agar tidak terjadi duplikasi, dengan penelitian yang sudah di lakukan oleh pihak lain.

Sistem Pakar Diagnosis Penyakit tanaman Mentimun dengan menggunakan Metode *Certainty Factor* berbasis web ini merupakan web yang mengandung pengetahuan, pengalaman yang di masukan satu banyak pakar ke dalam suatu area pengetahuan tertentu sehingga setiap orang dapat menggunakannya untuk memecahkan berbagai masalah yang bersifat spesifik dalam hal ini, yang pada umumnya digunakan untuk sistem pendukung keputusan dan sistem pakar. algoritma ini akan berjalan sangat baik ketika permasalahan bermula dari mengumpulkan ataupun menyatukan informasi lalu kemudian mencari kesimpulan yang dapat diambil dari informasi tersebut.

Berikut hasil-hasil penelitian yang relevan dan perbandingan penelitian yang telah ada sebelumnya yang serupa dengan penelitian yang sedang dilakukan yang disajikan dalam bentuk tabel 1.

Tabel 1. Penelitian Yang Relevan

No	Penulis/Tahun	Topik Penilitian	Metode	Pembahasan	Hasil
1	Amri/2021	Sistem Pakar Mendiagnosis Penyakit Pada Tanaman Mentimun menggunakan Metode Dempster Shafer Berbasis Android	*Dempster Shafer*	Penerapan metode Dempster Shafer memberikan suatu keputusan pengendalian berdasarkan dari hasil perhitungan nominal tertinggi. *Dempster shafer* adalah suatu teori matematika untuk pembuktian berdasarkan *belif functions and plausible reasoning* (fungsi kepercayaan dan pemikiran yang masuk akal), yang digunakan untuk mengkombinasikan potongan informasi yang terpisah (bukti).	*Certainty factor* menggunakan suatu nilai untuk mengasumsikan derajat keyakinan seorang pakar terhadap suatu data. Secara umum Teori *Certainty Factor* ditulis dalam suatu interval: *Certainty factor* menggunakan suatu nilai untuk mengasumsikan derajat keyakinan seorang pakar terhadap suatu data.

No	Penulis/Tahun	Topik Penilitian	Metode	Pembahasan	Hasil
2	Normawardah/2019	Sistem pakar diagnosis penyakit tanaman jagung menggunakan metode certainty factor berbasis web	*Certainty factor*	Pada penelitian ini ada 7 jenis penyakit dari jagung : Penyakit Bulai, Penyakit Bercak Daun, Penyakit Karat Daun, Penyakit Gosong Bengkak, Penyakit Busuk Pelepah, Penyakit Hawar Daun, Penyakit Busuk Batang	Untuk penyakit pada mentimun ada 8 jenis penyakit : Busuk daun, Penyakit embun tepung, Antraknosa, Layu fusarium, Kudis, Buah busuk, Layu bakteri, Virus.
3	Rohmat Indra Borman/2020	Implemetasi *Certainty Factor* dalam mengatasi ketidakpastian pada sistem pakar diagnosa penyakit kuda laut	*Certainty Factor*	Penyakit pada kuda laut ada 4 jenis : penyakit Penyakit Bakteri Vibrio, Penyakit Aeromenas, penyakit Parasit *Nemotoda*, penyakit *Protozoa.*	Untuk penyakit pada mentimun ada 8 jenis penyakit : Busuk daun, Penyakit embun tepung, Antraknosa, Layu fusarium, Kudis, Buah busuk, Layu bakteri, Virus.

No	Penulis/Tahun	Topik Penilitian	Metode	Pembahasan	Hasil
4	Alfie Nur Rahmi/2019	Sistem pakar diagnose penyakit dan hama pada penyakit mentimun menggunakan metode *Forward Chaining*	*Forward Chaining*	1. Penggunaan Sistem Pakar dalam sistem ini digunakan untuk mendiagnosa penyakit pada tanaman mentimun dengan meniru kerja para ahli / dokter, dan dapat menjawab pertanyaan yang menyangkut penyakit tanaman mentimun. 2. Perancangan sistem pakar diagnosa penyakit tanaman mentimun menggunakan metode forward chaining sebagai solusi alternatif dalam penyakit dan solusi. Hasil dari	Penulis menggunakan metode *certainty factor*, desain system menggunakan UML dan bahasa pemrograman PHP sedangkan penelitian yang sebelumnya menggunakan Sistem pakar diagnose penyakit dan hama pada penyakit mentimun menggunakan metode *Forward Chaining*

				sistem pakar menggunakan metode forward chaining dapat diterapkan oleh para pengguna sesuai jenis gejala yang ada, agar para pengguna terutama masyarakat awam agar dapat mengetahui penyakit secara dini.	

2.2 Kajian Teori

a. Materi

1) Sistem Pakar

Dalam bukunya Perancangan Sistem Pakar, beberapa definisi sistem pakar menurut beberapa ahli yaitu sebagai berikut.

a) Menurut Durkin : Sistem pakar adalah suatu program komputer yang dirancang untuk memodelkan kemampuan penyelesaian masalah yang dilakukan seorang pakar.
b) Menurut Ignizo : Sistem pakar adalah suatu model dan prosedur yang berkaitan, dalam suatu domain tertentu, yang mana tingkat keahliannya dapat dibandingkan dengan keahlian seorang pakar.
c) Menurut Giarratano dan Riley : Sistem pakar adalah suatu sistem komputer yang bisa menyamai atau meniru kemampuan seorang pakar (Luther A. Latumakulita, 2012).

Bentuk umum sistem pakar adalah suatu program yang dibuat berdasarkan suatu aturan yang menganalisis informasinya (biasanya diberikan oleh pengguna suatu sistem) mengenai suatu kelas masalah spesifikasi serta analisis matematis dari masalah tersebut. Tergantung dari desainnya, sistem pakar juga mampu merekomendasikan suatu rangkaian tindakan pengguna untuk dapat menerapkan koreksi. Sistem ini memanfaatkan kapabilitas penalaran untuk mencapai suatu simpulan (Hidayat, 2017).

2) Manfaat Sistem Pakar

Sistem pakar menjadi sangat popular karena sangat banyak kemampuan dan manfaat yang diberikannya, diantar1anya adalah :

a) Meningkatkan produktivitas.
b) Membuat orang awam bekerja seperti layaknya seorang pakar.
c) Mampu menangkap pengetahuan dan kepakaran seseorang.
d) Memudahkan akses pengetahuan seorang pakar.
e) Meningkatkan kapabilitas sistem komputer. Integrasi sistem pakar dengan sistem komputer lain membuat sistem lebih efektif dan mencakup lebih banyak aplikasi.
f) Mampu bekerja dengan informasi yang tidak lengkap atau tidak pasti.
g) Bisa digunakan sebagai media pelengkap dalam pelatihan. Pengguna pemula yang bekerja dengan sistem pakar akan menjadi lebih berpengalaman karena adanya fasilitas penjelas yang berfungsi sebagai guru.
h) Meningkatkan kemampuan untuk menyelesaikan masalah karena sistem pakar mengambil sumber pengetahuan dari banyak pakar. dengan buku, basis data laporan penelitian dan pengalaman pemakai (Ulty, 2020).

3) *Certainty Factor*

Certainty factor adalah suatu metode untuk membuktikan apakah suatu fakta itu pasti ataukah tidak pasti yang berbentuk metric yang biasanya digunakan dalam sistem pakar. Metode ini sangat cocok untuk sistem pakar yang mendiagnosis sesuatu yang belum pasti.

Teori *Certainty Factor* adalah faktor kepastian (*certainty factor*) menyatakan kepercayaan dalam sebuah kejadian (fakta atau hipotesa) berdasar bukti atau penilaian pakar. *Certainty factor* menggunakan suatu nilai untuk mengasumsikan derajat keyakinan seorang pakar terhadap suatu data. Secara umum Teori *Certainty Factor* ditulis dalam suatu interval: *Certainty factor* menggunakan suatu nilai untuk mengasumsikan derajat keyakinan seorang pakar terhadap suatu data (Dewi, 2017).

Rumus menentukan nilai MB dan MD

CF[H,E] = MB[H,E] –MD[H,E]

CF[H,E] = Nilai MB – Nilai MD

= 0,8 – 0,4

=0,4*100%

=40%

Keterangan :

CF (H,E) = *certainty factor* hipotesa yang dipengaruhi oleh *evidence* e diketahui dengan pasti.

MB (H,E) = *measure of belief* terhadap hipotesa H, jika diberikan *evidence* E (antara 0 dan 1).

MD(H,E) = *measure of disbelief* terhadap *evidence* H, jika diberikan *evidence* E(antara 0 dan 1) *Certainty factor* untuk kaidah premis tunggal.

Tabel 2. Logika Metode *certainty factor*

No	Keterangan	Nilai
1	Kemungkinan Besar	0,6
2	Hampir Pasti	0,8
3	Pasti	1

a) Kelebihan metode *Certainty Factor* adalah:

1) Metode ini cocok dipakai dalam sistem pakar untuk mengukur sesuatu apakah pasti atau tidak pasti dalam mendiagnosa penyakit sebagai salah satu contohnya.

2) Perhitungan dengan menggunakan metode ini dalam sekali hitung hanya dapat mengolah 2 data saja sehingga keakuratan data dapat terjaga.

b) Kekurangan metode *Certainty Factor* adalah:

1) Ide umum dari pemodelan ketidakpastian manusia dengan menggunakan numer metode *certainty factor* biasanya diperdebatkan. Sebagian orang akan membantah pendapat bahwa formula untuk metode *certainty factor* diatas memiliki sedikit kebenaran.

2) Metode ini hanya dapat mengolah ketidakpastian/kepastian hanya 2 data saja. Perlu dilakukan beberapa kali pengolahan data untuk data yang lebih dari 2 buah.

3) *Xampp*

XAMPP adalah sebuah paket kumpulan Software yang terdiri dari Apache, MySQL, phpMyAdmin PHP,Perl, FileZilla, dan lain-lain XAMPP berfungsi untuk memudahkan instalasi lingkungan di PHP, di mana biasanya lingkungan pengembangan Web memerlukan PHP Apache, MySQL, dan phpMyAdmin serta Software-Software yang terkait dengan pengembangan Web.

4) *MySQL*

Menurut (Nirsal, 2020) MySQL merupakan salah satu jenis database server yang sangat terkenal dan banyak dibangun untuk membangun aplikasi web yang menggunakan database sebagai sumber pengolah datanya.

MySQL adalah salah satu jenis database server yang sangat terkenal dan banyak digunakan untuk membangun aplikasi web yang menggunakan database sebagai sumber dan pengelolaan datanya.

MySQL merupakan database yang pertama kali didukung oleh bahasa pemrograman script untuk internet (PHP dan Perl). MySQL dan PHP dianggap pasangan software pengembangan aplikasi web yang ideal. MySQL lebih sering di gunakan untuk membangun aplikasi berbasis web, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman script PHP.

5) *Netbeans*

Menurut (Nofriadi, 2016:4) ”Netbeans Merupakan Sebuah aplikasi Integrated Development Environment (IDE) yang Menggunakan Bahasa Pemograman Java dari Sun Microsystems yang berjalan diatas swing”.

Netbeans adalah Suatu aplikasi untuk membuat suatu program yang dimana bahasa pemograman yang di pakai adalah java, maupun bahasa pemograman yang lain dengan mudahnya penggunaan aplikasi netbeans ini maka programmer sering menggunakan aplikasi ini sebagai wadah untuk membuat berbagai macam program yang dimana netbeans IDE ini bersifat Open Source.

6) HTML (*Hypertext Markup Language*)

Hyper Text Markup Language (HTML) adalah sebuah bahasa markup yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi didalam sebuah penjelajah web Internet dan *formatting hyper text* sederhana yang ditulis kedalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegerasi. Dengan kata lain, berkas yang dibuat dalam perangkat lunak pengolah kata dan disimpan kedalam format ASCII normal sehingga menjadi *home page* dengan perintah-perintah HTML. Bermula dari sebuah bahasa yang sebelumnya banyak digunakan di dunia penerbitan dan percetakan yang disebut dengan SGML (*Standard Generalized Markup Langu age*), HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman web. HTML saat ini merupakan standar Internet

yang didefinisikan dan dikendalikan penggunaannya oleh *World Wide Web*. Consortium (W3C). HTML dibuat oleh kolaborasi Caillau TIM dengan Berners-lee Robert ketika mereka bekerja di CERN pada tahun 1989 (CERN) adalah lembaga penelitian fisika energi tinggi di Jenewa.

7) *Balsamiq Mockup*

Balsamiq Mockup adalah program aplikasi yang digunakan dalam pembuatan tampilan *user interface* sebuah aplikasi. *Software* ini sudah menyediakan tools yang dapat memudahkan dalam membuat desain *prototyping* aplikasi yang akan dibuat. *Software* ini berfokus pada konten yang ingin digambar dan fungsionalitas yang dibutuhkan oleh pengguna. Menurut website resmi *balsamiq* *http://balsamiq.com/*. *Balsamiq Mockup* adalah alat *wire framing* cepat yang membantu bekerja lebih cepat dan lebih pintar. *Balsamiq Mockup* menciptakan pengalaman sketsa dipapan tulis, tetapi menggunakan computer, membuat mockup menjadi cepat. Pengguna akan mengasilkan lebih banyak ide, sehingga pengguna akan dapat membuang yang buruk dan menemukan solusi terbaik.

8) *Visual Studio Code*

Visual Studio Code adalah *editor source code* yang dikembangkan oleh *Microsoft* untuk *Windows, Linux* dan *MacOS*. Ini termasuk dukungan untuk *debugging,* GIT *Control* yang disematkan, penyorotan sintaks, penyelesaian kode cerdas, cuplikan, dan *kode refactoring*. Hal ini juga dapat disesuaikan, sehingga pengguna dapat mengubah tema editor, *shortcut keyboard*, dan *preferensi*. *Visual Studio Code* gratis dan

opensource, meskipun unduhan resmi berada di bawah *lisensi proprietary*. *Kode Visual Studio* didasarkan pada *Elektron*, kerangka kerja yang digunakan untuk menyebarkan aplikasi *Node* untuk *desktop* yang berjalan pada *Blink layout*. Meskipun menggunakan kerangka *Elektron*, *Visual Studio Code* tidak menggunakan Atom dan menggunakan komponen editor yang sama (diberi kode nama "Monaco") yang digunakan dalam *Visual Studio Team Services* yang sebelumnya disebut *Visual Studio Online* (Lardinois, 2015).

9) *StarUML*

StarUML adalah sebuah proyek *open source* untuk pengembangan secara cepat, *fleksibel*, *extensible*, *featureful*, dan bebas-tersedia. *UML/platform MDA* berjalan pada *platform Win32*. Tujuan dari proyek *StarUML* adalah untuk membangun sebuah alat pemodelan perangkat lunak dan juga platform yang menarik adalah pengganti alat UML komersial seperti *Rational Rose*, *Together* dan sebagainya. *Star UML* mendukung UML (*Unified Modeling Language*). Berdasarkan pada UML version 1.4 dan dilengkapi 11 macam diagram yang berbeda, selanjutnya mendukung notasi UML 2.0 dan juga mendukung pendekatan MDA (*Model Driven Architecture*) dengan dukungan konsep UML. *StarUML* dapat memaksimalkan pruduktivitas dan kualitas dari suatu *software project*. Hal yang paling penting dalam pengembangan perangkat lunak adalah *Usability*. *StarUML* diimplementasikan untuk

memberikan berbagai fitur yang *user-friendly* seperti dialog cepat, manipulasi *keyboard*, *ikhtisar diagram*, dll.

b. Penyakit Tanaman Mentimun

Penyakit adalah kerusakan yang terjadi pada tanaman yang disebabkan oleh jamur, bakteri atau virus yang menyerang atau merusak organ tanaman (rahmi, verawati dan kurniasih 2019).

Pada tanaman mentimun *varietas* roman, penyakit yang sering kali muncul antara lain sebagai berikut :

1) Busuk daun *(Downy mildew),* Penyakit yang disebabkan oleh jamur *Pseudoperonospora* yang merusak daun yang dipancarkan oleh angin sehingga menginfeksi kulit daun mentimun. Infeksi pada tanaman terjadi jika kelembaban udara tinggi.
2) Penyakit embun tepung *(Powdery mildew)*, Penyakit ini disebabkan oleh *Erysiphe cichoracerum*. Penyakit ini dipancarkan oleh angin, kemudian berkembang dengan cepat bila suhu dan keadaan tanahnya kering. Biasanya penyakit ini menyerang pada saat musim kemarau yang menyerang daun dan batang
3) Antraknosa, penyakit ini disebabkan oleh cendawan Colletotrichum lagenarium Pass dan mengakibatkan daun berwarna kuning atau kecoklatan, tangkai batang berwarna kuning dan buah berwarna kuning hingga kehitaman.
4) Layu Fusarium, Tanaman layu mulai dari bagian bawah, daun dan

batang layu kemudian mengering dan akhirnya mati.

5) Kudis, Penyakit ini disebabkan oleh cendawan Cladosporium cucumerinum dipancarkan oleh angin dan sisa-sisa tanaman.
6) Busuk buah, Penyakit yang disebabkan oleh Rhizopus, infeksi dan serangan busuk buah dapat terjadi pada saat dilahan (kebun) maupun saat panen
7) Layu Bakteri, penyakit yang disebabkan bakteri erwinia tracheiphila (E.F.Sm) Holland. Penyakit layu bakteri penyebarannya melalui perantaraan kumbang mentimun (cucumber beetle). Bakteri ini hanya bisa hidup dalam tubuh kumbang mentimun.
8) Virus, Penyakit ini disebabkan oleh cucumber Mosaic virus. Akibat yang ditimbulkan virus ini adalah permukaan daun belang berwarna hijua tua dan hijau muda, daun layu dan mengering serta tanaman kerdil.

c. Permodelan Sistem

1) *Black Box Testing*

Metode pengujian perangkat lunak yang penulis gunakan adalah metode *Black Box Testing*. *Black Box Testing* adalah pengujian aspek fundamental sistem tanpa memperhatikan struktur logika internal perangkat lunak berfungsi dengan benar. Pengujian *black box* merupakan metode perancangan data uji yang didasarkan pada spesifikasi perangkat lunak. Data diuji dibangkitkan, dieksekusi pada perangkat lunak kemudian keluaran dari perangkat lunak dicek apakah telah sesuai

dengan yang diharapkan. Pengujian *black box* berfokus pada pengujian persyaratan fungsional perangkat lunak, untuk mendapatkan serangkaian kondisi input yang sesuai dengan persyaratan fungsional suatu program. Pengujian *black box* berusaha menemukan kesalahan dalam kategori berikut ini:

1) Fungsi-fungsi yang tidak benar atau hilang dalam aplikasi.
2) Kesalahan *interface*.
3) Kesalahan dalam struktur data.
4) Kesalahan kerja inisialisasi dan kesalahan terminasi.

2) UML (*Unified Modeling Language*)

Menurut (Adi, 2020) UML (Unified Modeling Language) adalah "bahasa" pemodelan untuk sistem atau perangkat lunak yang berparadigma 'Berorientasi Objek'. Permodelan (modeling) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami.

UML (Unified Modeling Language) merupakan metode pemodelan secara visual sebagai sarana untuk merancang atau membuat software berorientasi objek. UML adalah salah satu tool/model untuk merancang pengembangan *software* yang berbasis object oriented. UML sendiri juga memberikan standar penulisan sebuah sistem blue print, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen komponen yang diperlukan

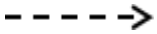
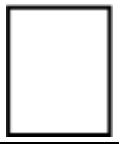
dalam sistem *software*. UML adalah sebuah bahasa standar untuk pengembangan sebuah *software* yang dapat menyampaikan bagaimana membuat dan membentuk model-model, tetapi tidak menyampaikan apa dan kapan model yang seharusnya dibuat yang merupakan salah satu proses implementasi pengembangan *software*. UML mendefinisikan diagram-diagram sebagai berikut :

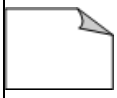
a) *Use Case* Diagram

Use Case Diagram menggambarkan apa saja aktifitas yang dilakukan oleh suatu sistem dari sudut pandang pengamatan luar. yang menjadi persoalan itu apa yang dilakukan bukan bagaimana melakukannya. *Use Case* Diagram dekat kaitannya dengan kejadian-kejadian. Kejadian atau skenario merupakan contoh apa yang terjadi ketika seseorang berinteraksi dengan system. *Use Case* Diagram berguna dalam tiga hal :

a) Menjelaskan fasilitas yang ada
b) Memudahkan komunikasi dengan klien
c) Membuat tes dari kasus-kasus yang muncul

Tabel 3. Simbol dalam *Use Case* Diagram

Notasi	**Nama**	**Keterangan**
1	**2**	**3**
	*Actor*	peran yang pengguna mainkan ketika berinteraksi dengan *Use Case*.
	*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (*independent*) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mendiri (*independent*).
	*Generalization*	Hubungan dimana objek anak (*descendent*) berbagi prilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
	*Include*	Menspesifikasikan bahwa *Use Case* sumber secara *eksplisit*
	Extend	Menspesifikasikan bahwa *use case* target memperluas prilaku dari *use case* sumber pada suatu titik yang diberikan.
	Association	Apa yang menghubungkan antara objek satu dengan objek lainnya.
	*System*	Menspesifikasikan paket yang menampilkan sistem secara terbatas

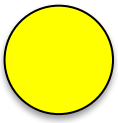
Notasi	Nama	Keterangan
1	2	3
	*Use Case*	Deskripsi dari urutan aksi-aksi yang ditampikan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor.
	*Collaboration*	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya.
	*Note*	Elemen fisik yang ada saat aplikasi dijalankan

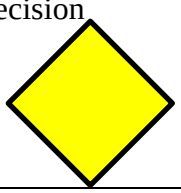
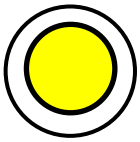
Sumber. Julianto Simatupang, S.S(2019)

b) *Activity* Diagram

Activity diagram berfokus pada aktifitas-aktifitas yang terjadi yang terkait dalam suatu proses tunggal. Jadi dengan kata lain, diagram ini menunjukkan bagaimana aktifitas-aktifitas tersebut bergantung satu sama lain. *Activity* diagram dapat dibagi menjadi beberapa jalur kelompok yang menunjukkan obyek yang mana yang bertanggung jawab untuk suatu aktifitas.

Tabel 4. Simbol dalam *Activity* Diagram

Simbol	Deskripsi
1	2
Status Awal 	Status awal aktivitas sistem, sebuah activity diagram memiliki sebuah status awal.

Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan/Decision 	Asosiasi percabangan jika ada pilihan aktivitas lebih dari satu.
Status Akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

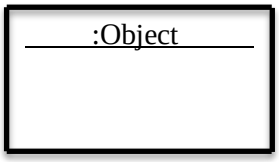
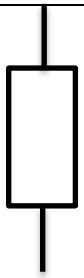
Sumber. Julianto Simatupang, S.S(2019)

3. *Sequence Diagram*

Diagram sequence merupakan diagram yang digunakan untuk menggambarkan perilaku objek pada *use case* dengan mendeskripsikan proses objek dengan pesan yang dikirimkan. Simbol- simbol yang ada pada S*equence Diagram* dapat dilihat pada tabel 5.

Tabel 5. Simbol dalam *Sequence* Diagram

Simbol	Diskripsi
1	2
Aktor 	*Actor* juga dapat berkomunikasi dengan *Object*, maka *Actor* juga dapat diurutkan sebagai kolom.

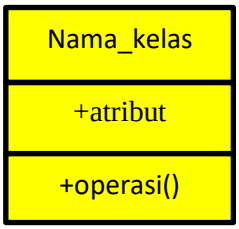
Garis HIdup/life line	*Lifeline* mengindikasikan keberadaan sebuah *object* dalam basis waktu. Notasi untuk *Lifeline* adalah garis putus-putus Vertikal yang ditarik dari sebuah *Object*
Object 	*Object* merupakan instance dari sebuah *class* dan dituliskan tersusun secara horizontal. Digambarkan sebagai sebuah *class* (kotak) dengan nama *object* didalamnya yang diawali dengan sebuah titik koma.
Activation 	*Activation* dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah Lifeline. mengindikasikan sebuah objek yang akan melakukan sebuah aksi.
Message 	*Message, digambarkan dengan anak panah horizontal antara Activation.* *Message mengindikasikan komunikasi antara Object-Object.*

Sumber. Julianto Simatupang, S.S(2019)

4) *Class Diagram*

Class Diagram digunakan untuk menggambarkan struktur dalam objek sistem. Diagram ini menunjukkan *Class Object* yang menyusun sistem dan juga hubungan antara *Class Object*. Simbol-simbol yang ada pada *Class Diagram* dapat dilihat pada tabel 6.

Tabel 6. Simbol *Class* Diagram

Simbol	Deskripsi
1	2
Kelas 	Kelas pada struktur sistem, tiap kelas memiliki nama, attribute, dan operation atau method.
Antar muka/*Interface*  nama_interface	Sama dengan konsep *interface* dalam pemprograman berorientasi objek.
Asosiasi/Association	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity.
Asosiasi berarah/*DirectedAssociation* 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicy.*
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisi (umum khusus).
Kebergantngan/Dependency 	Relasi antar kelas dengan makna kebergantungan antar kelas.
Agregasi/Aggregation 	Relasi antar kelas dengan makna semua-bagian (*whole-part*)

Sumber. Julianto Simatupang, S.S(2019)

3) Konsep Basis Data

Konsep basis data yaitu kumpulan dari catatan , atau potongan dari pengetahuan. Sebuah basis data memiliki penjelasan terstruktur dari jenis fakta yang tersimpan didalamnya, penjelasan ini disebut skema. Skema menggambarkan objek yang diwakili suatu basisdata, dan hubungan

diantara objek tersebut. Ada banyak cara untuk mengorganisasikan skema, atau memodelkan struktur basis data, dikenal sebagai model basis data atau model data. Ada beberapa komponen dasar data base yang digunakan. Sumber : Jayanti, N. K. D. A., & Sumiari, N. K. (2018).

a) *Field*

   Field merupakan implementasi dari suatu atribut data. *Field* merupakan unit terkecil dari data yang disimpan dalam suatu fileatau basis data. *Field- field* tersebut di organisasikan dalam *record -record.*

b) *Record*

   Record merupakan koleksi dari *field– field* yang disusun dalam *format* yang telah ditentukan. Selama desain sistem *record* kandiklasifikasikan sebagai *fixed-length record* atau *variable-length record*. *Fixed-length record* adalah tipe *instancerecord* yang punya *field*, jumalah *field* dan ukuran logik yang sama *variable-lengthrecord*. Adalah mengijinkan *record-record* yang berbeda dalam file yang sama memiliki panjang yang berbeda.

c) *File* dan *Table*

   Record-record yang serupa diorganisasikan dalam group-group yang disebut file. Jadi file merupkan kumpulan semua kejadian dari struktur *record* yang diberikan. Table merupakan ekuivalen basis data relasional dari sebuah file.

d) Kunci (*key*)

   Kunci merupakan elemen *record* yang dipakai untuk menemukan *record* tersebut pada waktu akses atau bisa digunakan untuk

identifikasi tiap *record* kesebuah file. Ada pun jenis kuncinya adalah sebagai berikut :

1) *Super Key*

   Superkey dengan jumlah atribut minimal. *Candidate key* ini tidak boleh berisi atribut dari table yang lain.

2) *Candidate Key*

   Superkey dengan jumlah atribut minimal. *Candidate key* ini tidak boleh berisi atribut dari table yang lain.

3) *Primary Key*

   Kumpulan atribut dari suatu tabelyang dapat digunakan untuk mengidentifikasi *entity* atau *record* dari tabel tersebut secara unik.

4) *Alternate Key*

   Setiap atribut dari *Candidatekey* yang tidak terpilih sebagai *Primary Key* akan dinamakan *Alternate Key*.

5) *Foreign Key*

   Merupakan sembarang atribut yang menunjuk kepada *Primary Key* pada tabel lain. Akan terjadi suatu relasi yang memiliki kardinalitas *onetomany* atau *manytomany*

BAB III

METODE PENILITIAN

3.1 Lokasi Penelitian

Penulis membuat Tugas Akhir penelitian mengenai Sistem Pakar Diagnosis Penyakit Tanaman Mentimun dengan Menggunakan Metode Certainty Factor Berbasis Web ini berfungsi sebagai media diagnosis penyakit dan cara pencegahan penyakit pada tanaman Mentimun. Lokasi penelitian sebagai bahan penulis mengetahui penyakit dan pencegahan yang sering terjadi pada tanaman Mentimun dilakukan di Balai Pengkajian Teknologi Pertanian Kalimantan Tengah.

3.2 Perencanaan Alat dan Bahan

a. Perangkat Keras

Dalam kebutuhan perangkat keras (*Hardware*) yang penulis gunakan dalam pembuatan aplikasi seperti pada tabel 7.

Tabel 7. Spesifikasi Perangkat Keras

No	Perangkat Keras	Spesifikasi
1	Type	Acer ES1-432
2	Processor	Intel(R) Celeron(R) CPU N3350 @ 1.10GHz 1.10 GHz
3	Grafik	Intel(R) HD Graphics
4	Memory	500 GB
5	RAM	2 GB

b. Perangkat Lunak

Perangkat lunak (*Software*) dalam aplikasi ini digunakan sebagai berikut :

Tabel 8. Spesifikasi Perangkat Lunak

Perangkat Keras	Spesifikasi
Xampp	Sebagai web server
Goole Crome	Sebagai web browser.
CodeIgniter	*framework* PHP yang digunakan.
MySQL	Sebagai web server
Balsamiq *Wireframes*	Sebagai perancangan desain interface sistem (Prototype).
HTML	Sebagai bahasa markup untuk membuat sebuah halaman web.
Bootstrap	Untuk membuat website yang bersifat responsive dan lebih menarik.
StarUML	Utuk membuat diagram UML.

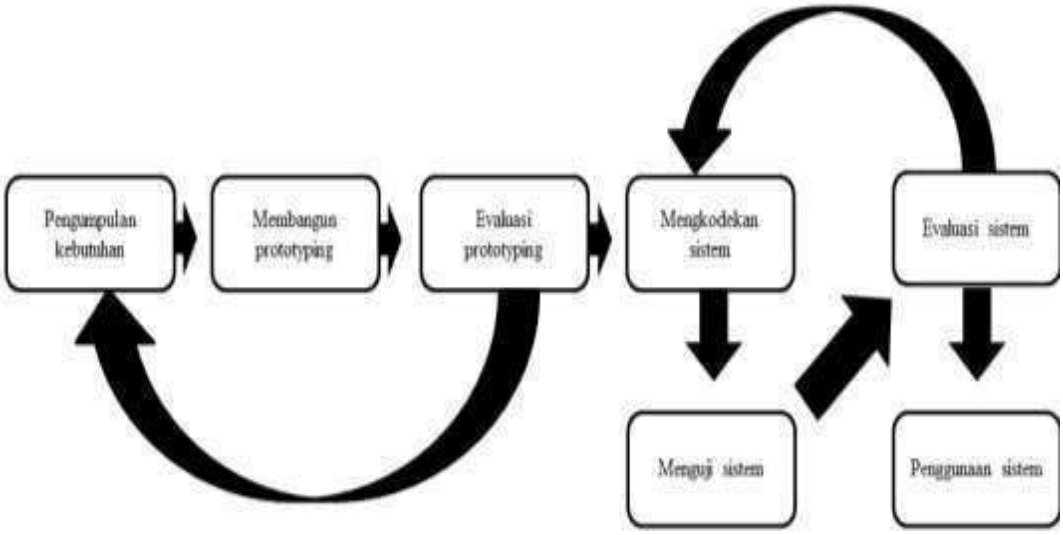
c. Informasi dan Data

Infomasi dan data yang penulis butuhkan di dapat dari beberapa sumber seperti jurnal, dan ilmu yang didapatkan secara langsung melalui wawancara dengan narasumber.

3.3 Jenis Penelitian

a. Model *Prototype*

Prototype adalah tahapan yang ditujukan untuk mentransformasi sifat-sifat abstrak dari sebuah ide menjadi lebih berwujud. Tahapan ini tidak hanya berupa proses visualisasi ide tetapi juga proses pembangunan ide. Tujuannya adalah mengembangkan model menjadisistem final. Artinya sistem akan dikembangkan lebih cepat dari padametode tradisional dan biayanya menjadi lebih rendah. Ciri dari metode ini adalah pengembang dan pelanggan dapat melihat dan melakukan pengerjaan dengan bagian dari sistem komputer.



Gambar 1. Model Prototype

Sumber. YRM, (2020)

Fase–fase dalam *prototype* adalah sebagai berikut :

1) Pengumpulan kebutuhan

   Ditahap ini pengembang melakukan identifikasi *software* dan semua kebutuhan sistem yang akan dibuat.

2) Membangun *prototype*

   Membangun *prototype* dengan membuat peracangan sementara yang berfokus pada penyajian kepada pelanggan (misalnya dengan membuat *input* dan format *output*).

3) Evaluasi *prototype*

   Evaluasi ini dilakukan oleh pelanggan apakah *prototype* yang sudah dibangun sudah sesuai dengan keinginan pelanggan. Jika sudah sesuai maka langkah keempat akan diambil. Jika tidak, maka *prototype* diperbaiki dengan mengulang langkah 1, 2 dan 3.

4) Mengkodekan *System*

   Pada tahap ini *prototype* yang sudah disetujui akan diubah kedalam bahasa pemrograman.

5) Menguji *System*

   Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, maka sistem akan melalui tahap pengujian. Pengujian inidapat dilakukan dengan menggunakan *White Box*, *Black Box, Basic Path*, Pengujian arsitektur dan lain-lain.

6) Evaluasi *System*

Pelanggan mengevaluasi apakah *system* yang sudah jadi sudah sesuai dengan yang diharapkan. Jika sudah, maka langkah selanjutnya dilakukan, jika belum maka mengulangi langkah d dan e.

7) Menggunakan *system* Perangkat lunak yang sudah diuji dan disetujui oleh pelanggan siap digunakan

3.4 Teknik Pengumpulan Data

Penulis mengunakan beberapa metode dalam melukan penelitian ini, yaitu :

a. Studi Pustaka

Pada tahap ini penulis mengumpulkan beberapa penelitian yang relevan seperti jurnal dari berbagai sumber dengan tema yang hampir menyerupai tema penelitian penulis yang nantinya akan dijadikan sebagai sumber referensi untuk proposal penelitian yang sedang dilakukan dilakukan pada Balai Pengkajian Teknologi Pertanian Kalimantan Tengah.

b. Metode Dokumentasi

Pada tahap ini penulis melakukan dokumentasi terhadap beberapa bukti dalam penelitian berupa foto dengan narasumber di BPTP Kalteng, dokumentasi ini diperlukan untuk mengumpulkan bukti hasil dari penelitian penulis.

c. Metode Wawancara

Pada tahap ini penulis melakukan wawancara dengan Krisyetno, S.ST. Selaku pakar penyakit mentimun dengan menanyakan terkait jenis-jenis penyakit pada tanaman Mentimun.

3.5 Analisis Kebutuhan

a. Analisis Data

Teknik anaisis data yang digunakan dalam penelitian ini adalah analisis data induktif. Analisis data induktif adalah penarikan kesimpulan yang berangkat dari fakta-fakta khusus, untuk kemudian ditarik kesimpulan secara umum. Adapun langkah-langkah untuk menganalisis data dalam penelitian ini adalah sebagai berikut:

1) Data Jenis Penyakit

Daftar nama penyakit Mentimun ditunjukan pada Tabel 9. Jumlah nama penyakit Mentimun adalah 8 penyakit dan masing-masing nama penyakit diberi kode.

Tabel 9. Data Jenis Penyakit mentimun

Kode	Penyakit
P01	Busuk daun
P02	Penyakit embun tepung
P03	Antraknosa
P04	Layu fusarium
P05	Kudis
P06	Buah busuk
P07	Layu bakteri
P08	Virus

2) Data Gejala

Tabel 10 menunjukan gejala penyakit mentimun didefinisikan dan diberi kode.Terdapat 19 gejala penyakit.

Tabel 10. Tabel gejala penyakit mentimun

Kode	Gejala
G01	Daun berwarna kuning
G02	Daun berjamur
G03	Daun mudah membusuk
G04	Daun berubah berwarna kuning dan mengering
G05	Pada buah berwarna putih atau berjamur
G06	Bercak kecil pada daun berwarna coklat dan mati
G07	Tangkai batang berwarna kuning atau coklat
G08	Pada buah berwarna kuning hingga kehitaman
G09	Pada tulang daun menguning
G010	Bercak kuning menjalar ke tangkai daun
G011	Daun layu
G012	Buah mudah busuk dan berjatuhan
G013	Buah mengeluarkan cairan bening sampai coklat
G014	Buah busuk dan berjatuhan
G015	Bercak kehitaman dan coklat pada buah
G016	Helai daun layu menyeluruh
G017	Batang tanaman layu mengeluarkan lender berwarna putih
G018	Tanaman berukuran kerdil
G019	Belang hijau pucat pada permukaan daun berlekuk – lekuk

3) Tabel Kombinasi

Tabel di bawah ini berisikan tentang kombinasi antara tabel data Penyakit pada tabel 9 dan tabel Gejala pada tabel 10.

Tabel 11. Tabel Pakar Penyakit Mentimun

Kode	P01	P02	P03	P04	P05	P06	P07	P08
G01	X							
G02	X							
G03	X							
G04		X						
G05		X						
G06			X					
G07			X					
G08			X					
G09				X				
G010				X				
G011				X				
G012					X			
G013					X			
G014						X		
G015						X		
G016							X	
G017							X	
G018								X
G019								X

Keterangan kode penyakit :

P01 : Busuk daun
P02 : Penyakit embun tepung
P03 : Antraknosa
P04 : Layu fusarium
P05 : Kudis
P06 : Buah busuk
P07 : Layu bakteri
P08 : Virus

Keterangan kode gejala :

G01 : Daun berwarna kuning
G02 : Daun berjamur
G03 : Daun mudah membusuk
G04 : Daun berubah berwarna kuning dan mengering
G05 : Pada buah berwarna putih atau berjamur
G06 : Bercak kecil pada daun berwarna coklat dan mati
G07 : Tangkai batang berwarna kuning atau coklat
G08 : Pada buah berwarna kuning hingga kehitaman
G09 : Pada tulang daun menguning
G010 : Bercak kuning menjalar ke tangkai daun
G011 : Daun layu
G012 : Buah mudah busuk dan berjatuhan
G013 : Buah mengeluarkan cairan bening sampai coklat
G014 : Buah busuk dan berjatuhan
G015 : Bercak kehitaman dan coklat pada buah
G016 : Helai daun layu menyeluruh
G017 : Batang tanaman layu mengeluarkan lender berwarna putih
G018 : Tanaman berukuran kerdil
G019 : Belang hijau pucat pada permukaan daun berlekuk – lekuk

4) Data Pembobotan Nilan User

Pengguna konsultasi diberi pilihan jawaban yang masing-masing bobotnya sebaga berikut.

Tabel 12. Tabel Pembobotan Nilai User

Keterangan	**Nilai User**
Tidak	0.0
Tidak Tahu	0.2
Sedikit Yakin	0.4
CukupYakin	0.6
Yakin	0.8
Sangat Yakin	1.0

Contoh perhitungan CF untuk kasus penyakit busuk daun :

Tabel 13. Tabel Sampel Nilai Diagnosa Busuk Daun

Penyakit	Gejala	MB	MD	Nilai User
Busuk Daun	Daun berwarna kuning	0,6	0.2	Yakin (0.8)
Busuk Daun	Daun berjamur	0.8	0.1	Sedikit Yakin (0.4)
Busuk Daun	Daun mudah membusuk	0.9	0.1	Tidak Tahu (0.2)

Hasil Perhitungan :

Rumus Awal:

CF[H,E] = MB[H,E] – MD[H,E]

CF[H,e] = CF [E,e] * CF [H,E]

CFcombine CF[H,E]old,gejala = CF[H,E]old + CF[H,E]gejala * (1-CF[H,E]old)

Maka dengan perhitungan manual :

1. Menghitung CF pakar masing-masing gejala

a. Gejala G001

CF(H,E) = 0.6 – 0.2

= 0.4

b. Gejala G002

CF(H,E) = 0.8 – 0.1

= 0.7

c. Gejala G003

CF(H,E) = 0.9 – 0.1

= 0.8

2. Menghitung CF masing-masing gejala dengan mengkalikan CF pakar dengan CF user

a. Gejala G001

CF(H,e) = 0.4 * 0.8

= 0.32

b. Gejala G002

CF(H,e) = 0.7 * 0.4

= 0.28

c. Gejala G003

CF(H,e) = 0.8 * 0.2

= 0.16

3. Mengkombinasikan nilai CF dari masing-masing gejala

$CF_{combine1}$ = 0.32 + 0.28 * (1 – 0.32)

$CF(H,E)_{1,2}$ = 0.5104

$CF_{combine2}$ = 0.5104 + 0.16 * (1 – 0.5104)

$CF(H,E)_{old,3}$ = 0.5887

$CF_{combine3}$ = 0.5887 + 0.24 * (1 - 0.5887)

$CF(H,E)_{old,3}$ = 0.6874

Bila dijadikan dalam bentuk angka presentase, maka

= 0.6874 * 100

= 68.74%

Tiga gejala yang menjadi sampel diatas adalah gejala dari penyakit Busuk Daun pada tanaman Mentimun maka kesimpulannya: Berdasarkan perhitungan diagnosa dialami adalah Busuk Daun dengan kepercayaan 68.74%

b. Analisis Proses

Tipe sistem yang dapat dicari dengan Certaity Factor, terdapat 8 aturan yang tersimpan dalam basis pengetahuan yaitu :

RULE 1 : IF G01 AND G02 AND G03 THEN P01

RULE 2 : IF G04 AND G05 THEN P02

RULE 3 : IF G06 AND G07 AND G08 THEN P03

RULE 4 : IF G09 AND G010 AND G011 THEN P04

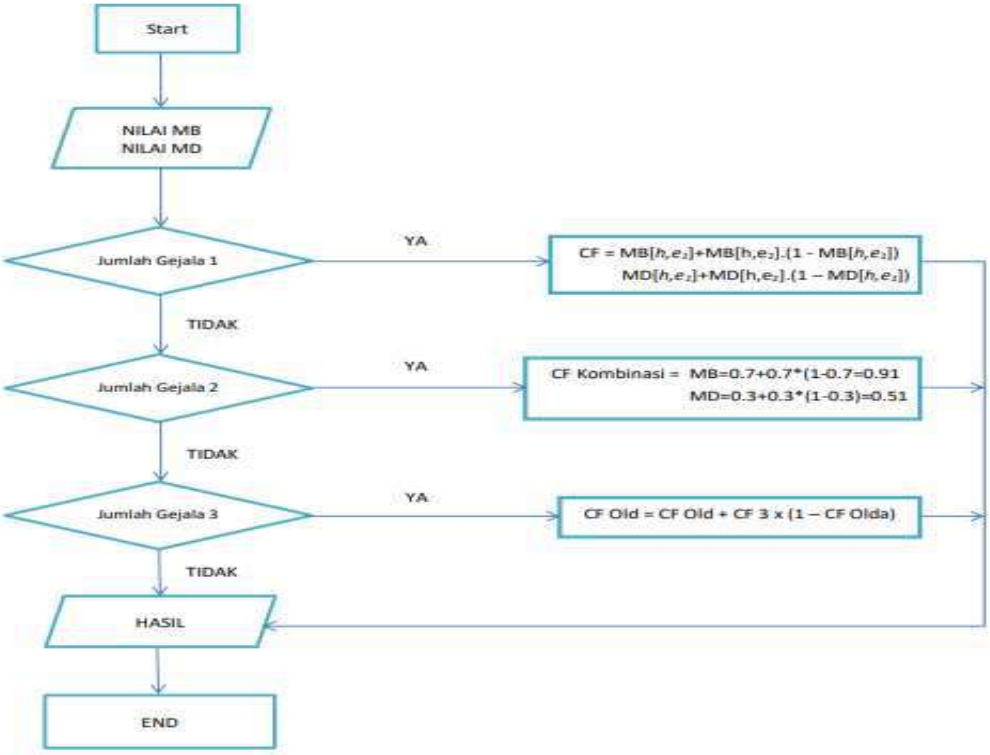
RULE 5 : IF G012 AND G013 THEN P05

RULE 6 : IF G014 AND G015 THEN P06

RULE 7 : IF G016 AND G017 THEN P07

RULE 8 : IF G018 AND G019 THEN P08

Pada tahap perancangan, data yang diggunakan sebagai data masukan sistem yaitu 19 gejala yang terlihat pada mentimun. Gejala ini didapat dari pakar di BPTP Kalimantan Tengah. Gejala akan dipilih oleh pengunjung/*user* dengan melakukan pada interface gejala. Rancangan alur proses diagnosa dapat dilihat pada gambar 2.



Gambar 2. Rancangan Alur Proses Menentukan Diagnosa Mentimun

Sumber. Normawardah (2019)

c. Analisis Kelemahan

Pada Balai Pengkajian Teknologi Pertanian Kalimantan Tengah itu sendiri masih belum memiliki suatu *system* berupa sebuah perangkat aplikasi (*software*) yang dapat menentukan suatu jenis penyakit mentimun dengan mengidentifikasi melalui gejala dan penyakit dari mentimun itu sendiri.

3.6 Desain Sistem

a. Desain Proses

Dalam perancangan sistem, disini penulis menggunakan diagram UML (*Unified Modeling language*). Adapun diagram yang digunakan adalah *Use case diagram, Activity Diagram, Sequence Diagram, Class diagram.*

1) *Use Case Diagram*

a) Identifikasi *actor* dan interaksinya dengan sistem

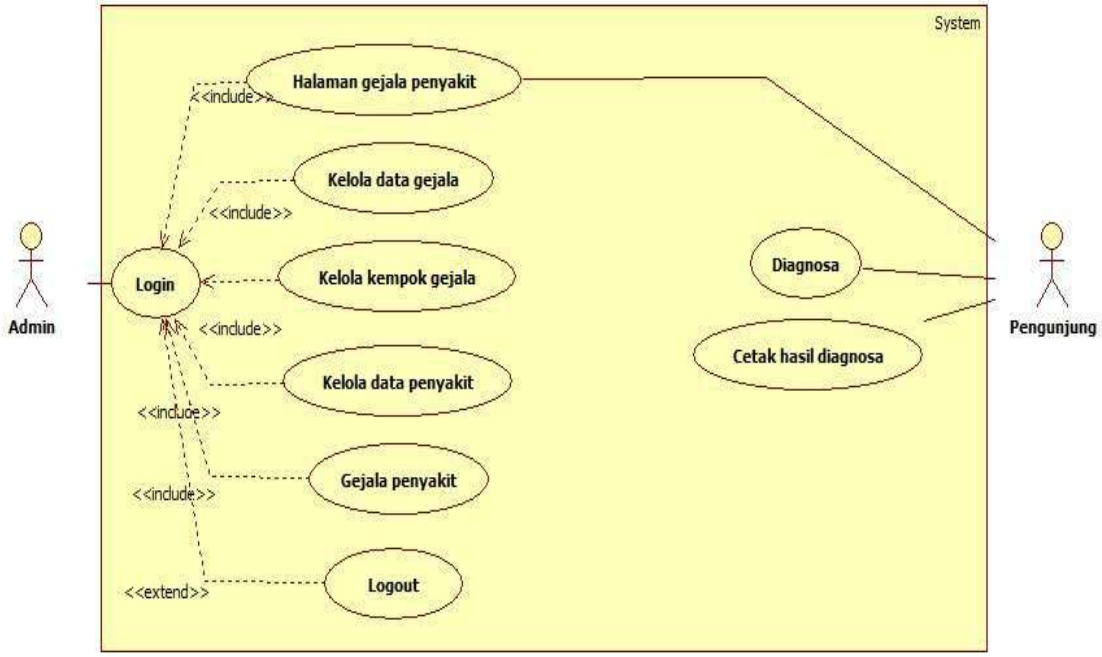
Sistem yang dirancang ini digunakan oleh admin pengujung sistem. Interaksi aktor dengan sistem akan dijelaskan pada tabel 14 berikut ini.

Tabel 14. Kegiatan Aktor dan Interaksinya dengan Sistem

Admin	Pengguna
1. Login 2. Halaman gejala penyakit 3. Kelola data gejala 4. Kelola kelompok gejala 5. Kelola data penyakit 6. Gejala penyakit 7. Logout	1. Halaman gejala penyakit 2. Diagnosa 3. Cetak hasil Diagnosa

2) Pembuatan *Use Case Diagram*

Berdasarkan tabel 14 di dapatkan spesifikasi kebutuhan sehingga *use case diagram* dari sistem dapat dilihat pada gambar 3 berikut ini :



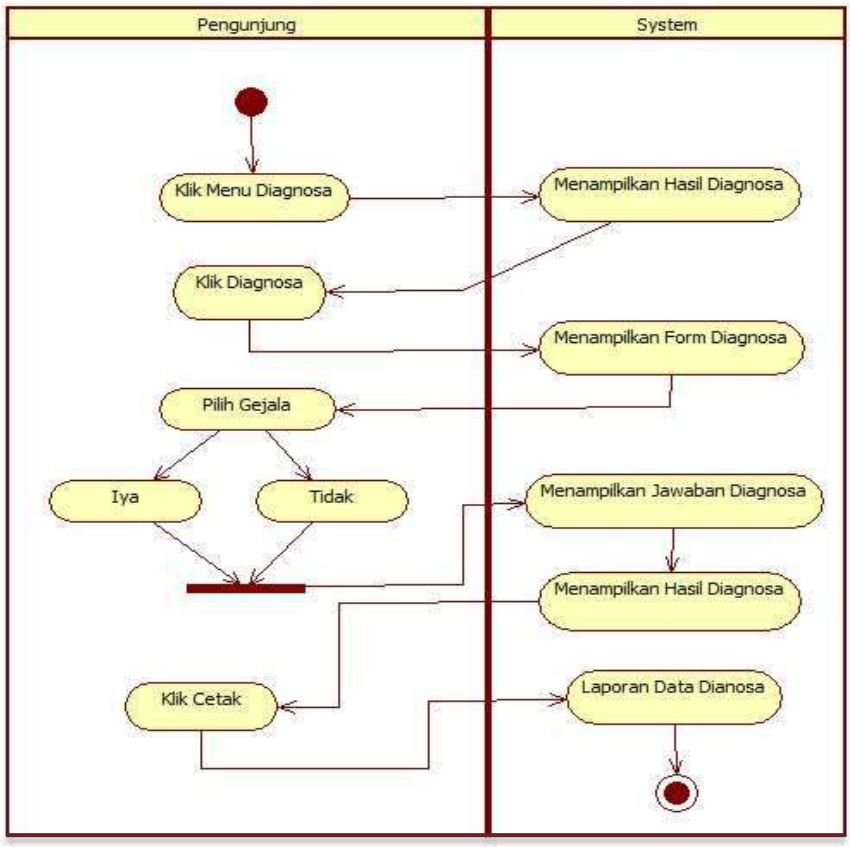
Gambar 3. Use Case *System* Yang Dirancang

Dilihat dari gambar 3, *use case* ini terdiri dari satu sub-sistem yaitu sistem yang dirancang. Dalam sub-sistem ini aktor admin diharuskan login terlebih dahulu untuk dapat kemenu Halaman Gejala penyakit, kelola data gejala, kelola kelompok gejala, kelola data penyakit, gejala penyakit, pada admin dapat melakukan logout setelah melakukan *login*. Untuk aktor Pengunjung sistem web masuk kemenu Gejala penyakit, diagnosa, cetak hasil diagnosa.

3) *Activity Diagram*

Pada pemodelan UML, *Activity Diagram* dapat digunakan untuk menjelaskan bisnis dan alur kerja operasional secara *step-by-step* dari komponen suatu sistem. *Activity Diagram* menunjukkan keseluruhan dari aliran kontrol. Berikut adalah *Activity Diagram* dari sistem yang dirancang.

a) *Activity Diagram* Diagnosa (Pengunjung)

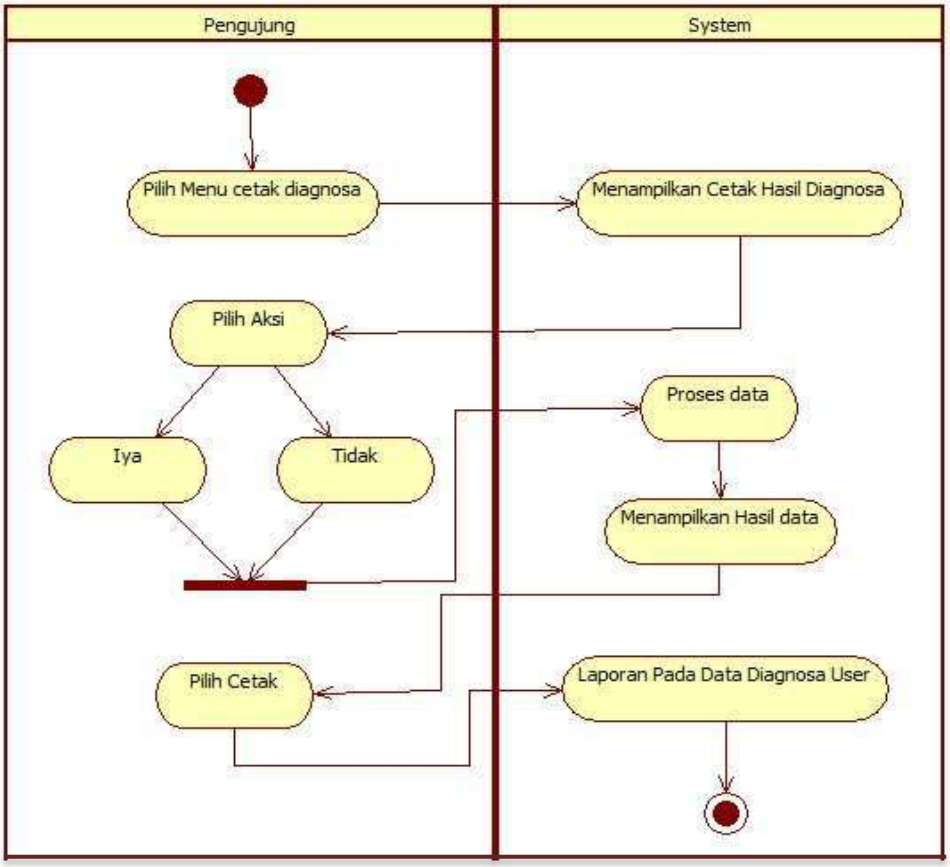


Gambar 4. *Activity Diagram* Diagnosa (Pengunjung)

Gambar diatas adalah aktivitas pengguna dalam melakukan diagnosa penyakit mentimun, yang dimana pengguna akan mengklik menu

diagnosa, menampilkan beberapa gejala, pilih gejala, dan hasil yang akan di print atau di *save* dalam bentuk pdf.

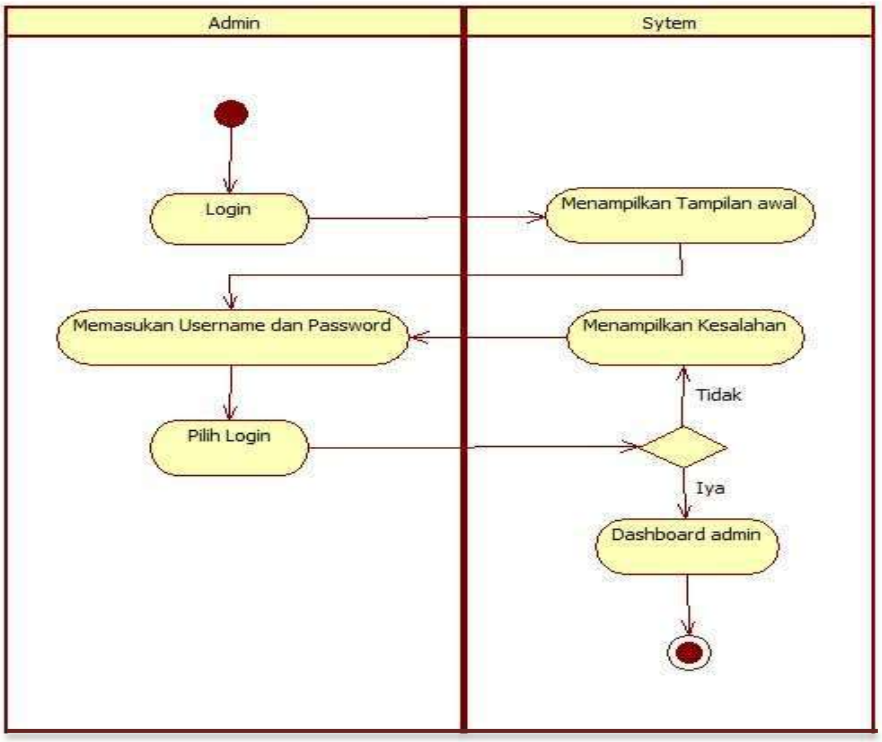
b) *Activity Diagram* Cetak Hasil Diagnosa (Pengunjung)



Gambar 5. *Aktivity Diagram* cetak hasil diagnosa (pengunjung)

Gambar diatas mengambarkan aktivitas pengguna pada saat ingin mencetak hasil dari diagnosa pada mentimun

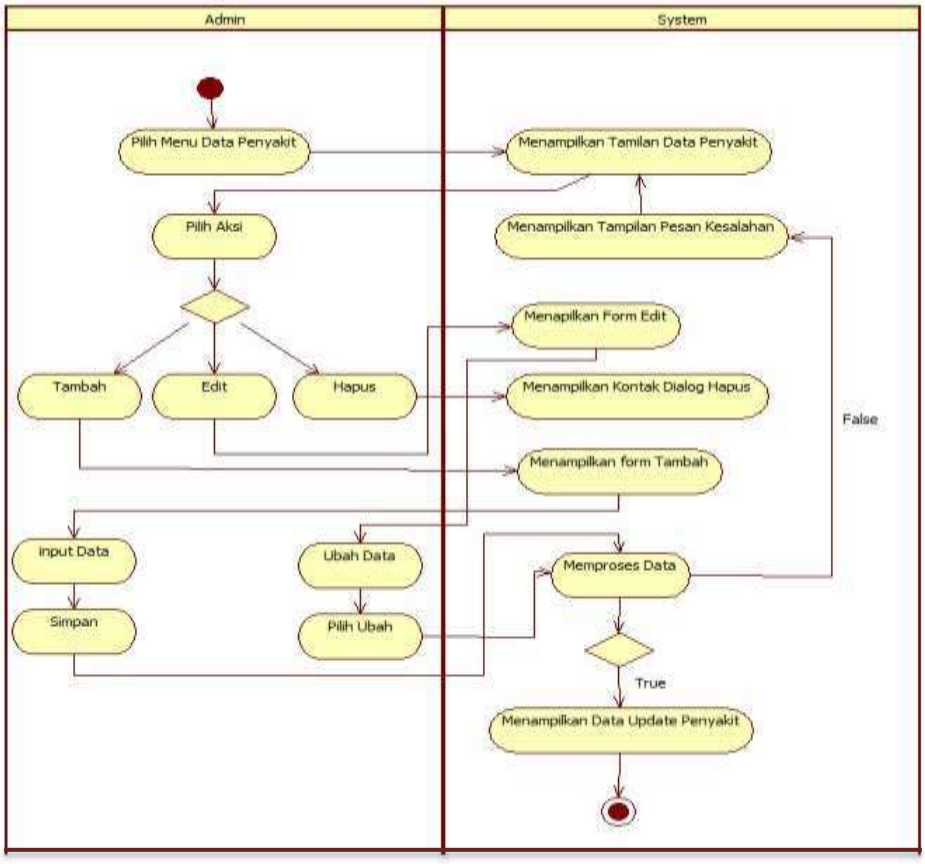
c) *Activity Diagram Login* (Admin)



Gambar 6. *Activity Diagram* Login (Admin**)**

Gambar diatas merupakan aktivitas untuk login admin, aktivitas ini dimulai dengan admin mengklik tombol login kemudian sistem merespon dengan menampilkan halaman login admin, selanjutnya admin diminta memasukkan *username* dan *password* kemudian mengklik tombol *login*, apabila data masukkan bernilai *false* maka sistem akan menampilkan pesan kesalahan dan admin diminta untuk memasukkan *username* dan *password* kembali, tetapi apabila data bernilai *true* maka sistem akan merespon untuk menampilkan halaman *dashboard* admin.

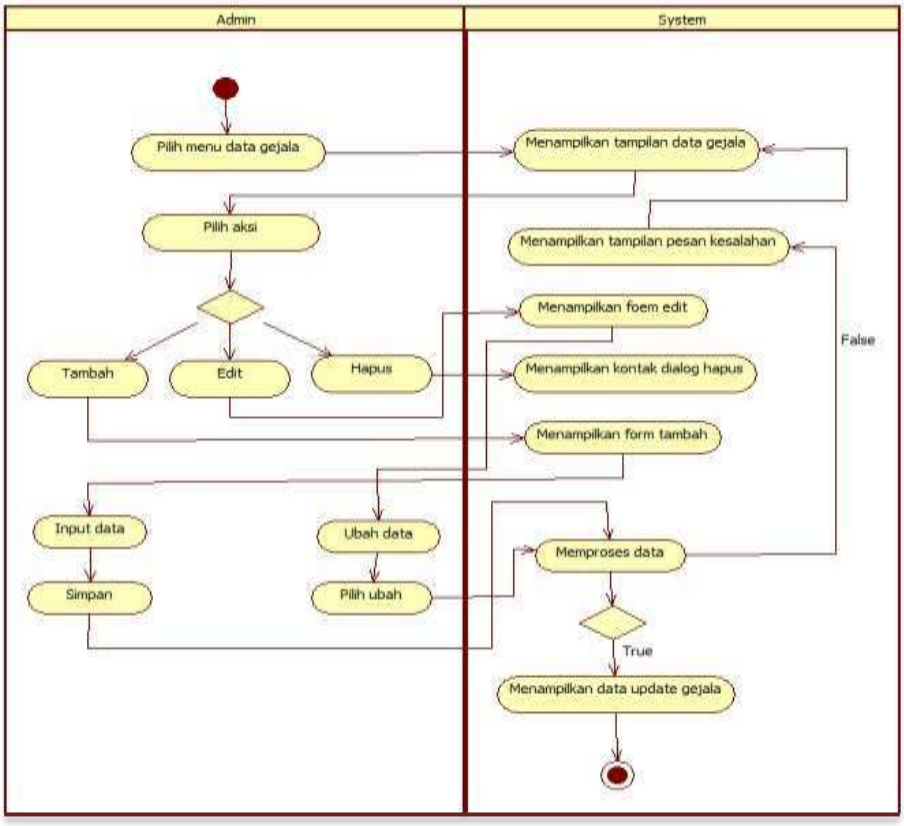
d) *Activity Diagram* Kelola Data Penyakit (Admin)



Gambar 7. *Activity Diagram* Kelola Data Penyakit (Admin)

Gambar diatas adalah aktivitas admin. Dimana dapat di jelaskan aktivitas admin bisa melakukan penambahan, mengedit dan menghapus data penyakit sesuai dengan keinginan admin dan memproses data hingga mendapatkan hasil yang diingikan atau sesuai dengan data yang ingin ditambahkan.

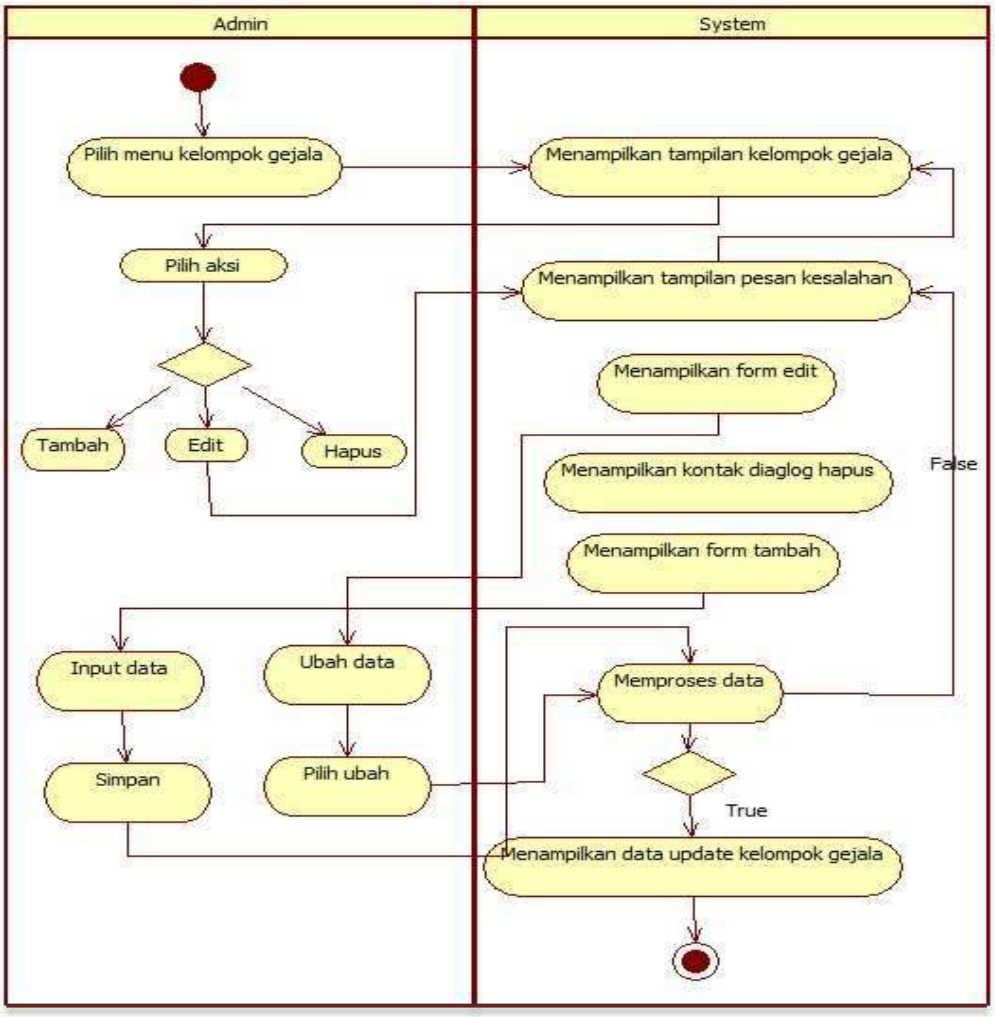
e) *Activity Diagram* Kelola Data Gejala (Admin)



Gambar 8. *Activity Diagram* Kelola Data Gejala (Admin)

Gambar diatas adalah aktivitas admin. Dimana dapat di jelaskan aktivitas admin bisa melakukan penambahan dan menghapus data gejala sesuai dengan keinginan admin dan memproses data hingga mendapatkan hasil yang diingikan atau sesuai dengan data yang ingin ditambahkan.

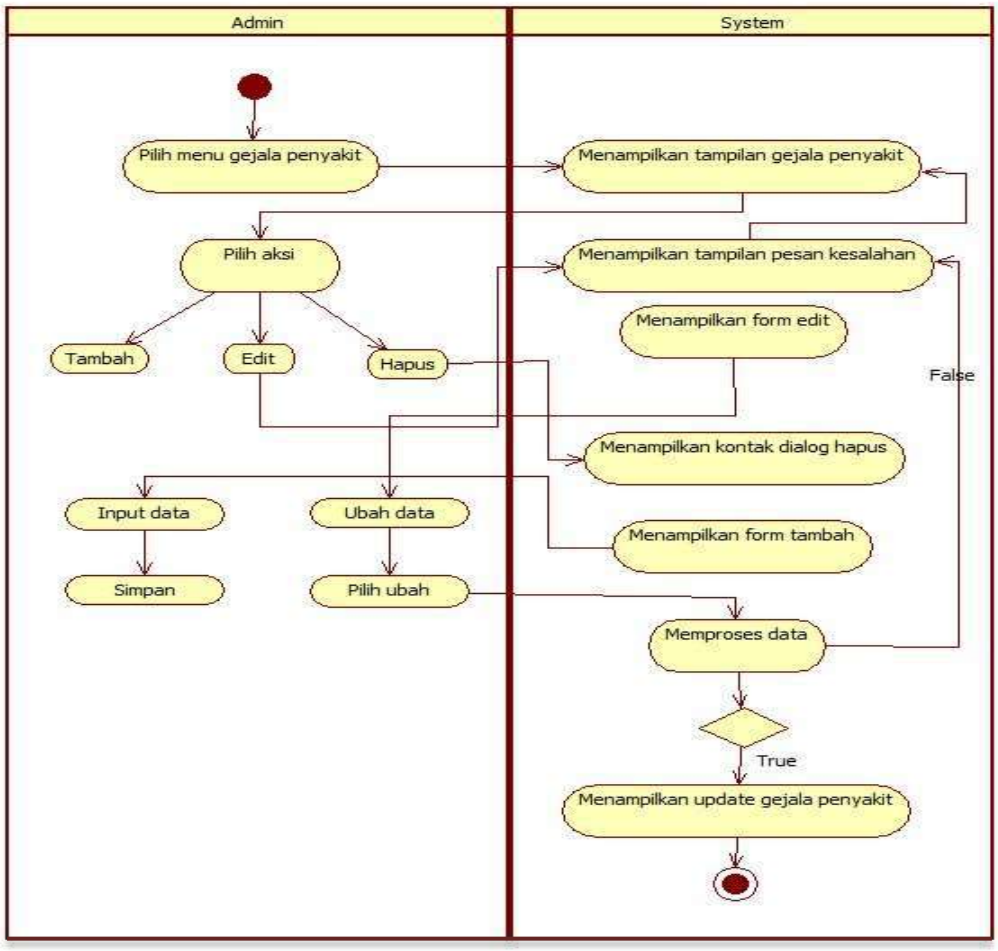
f) *Activity Diagram* Kelola Kelompok Gejala (Admin)



Gambar 9. *Activity Diagram* Kelola Data Gejala (Admin)

Gambar diatas adalah aktivitas admin. Dimana dapat di jelaskan aktivitas admin bisa melakukan penambahan dan menghapus data kelompok gejala sesuai dengan keinginan admin dan memproses data hingga mendapatkan hasil yang diingikan atau sesuai dengan data yang ingin ditambahkan.

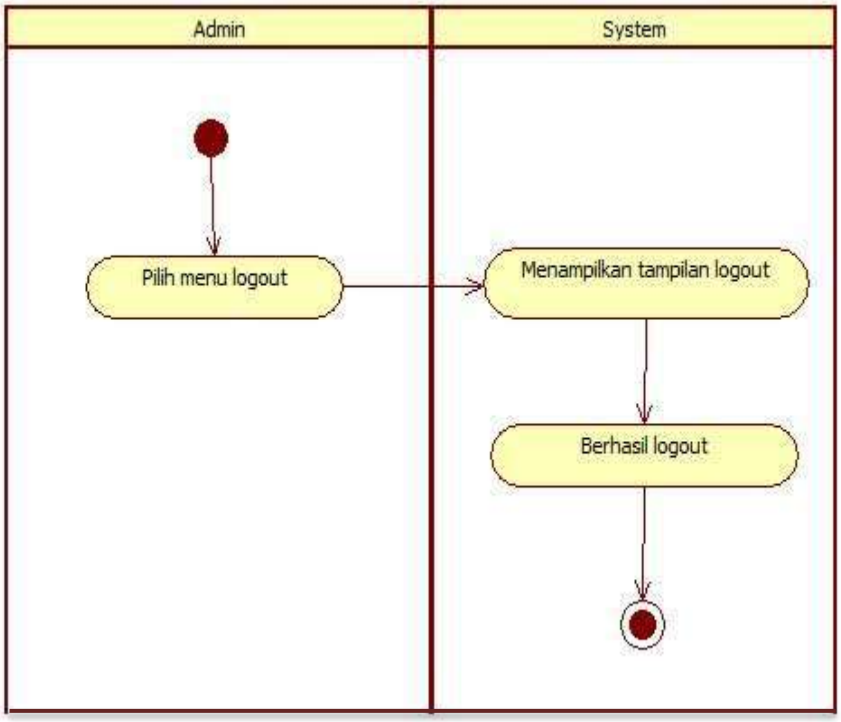
g) *Activity Diagram* Gejala Penyakit (Admin)



Gambar 10. *Activity Diagram* Kelola Gejala Penyakit (Admin)

Gambar diatas adalah aktivitas admin. Dimana dapat di jelaskan aktivitas admin bisa melakukan penambahan dan menghapus data kelola gejala penyakit sesuai dengan keinginan admin dan memproses data hingga mendapatkan hasil yang diingikan atau sesuai dengan data yang ingin ditambahkan.

h) *Activity Diagram Logout* (Admin)



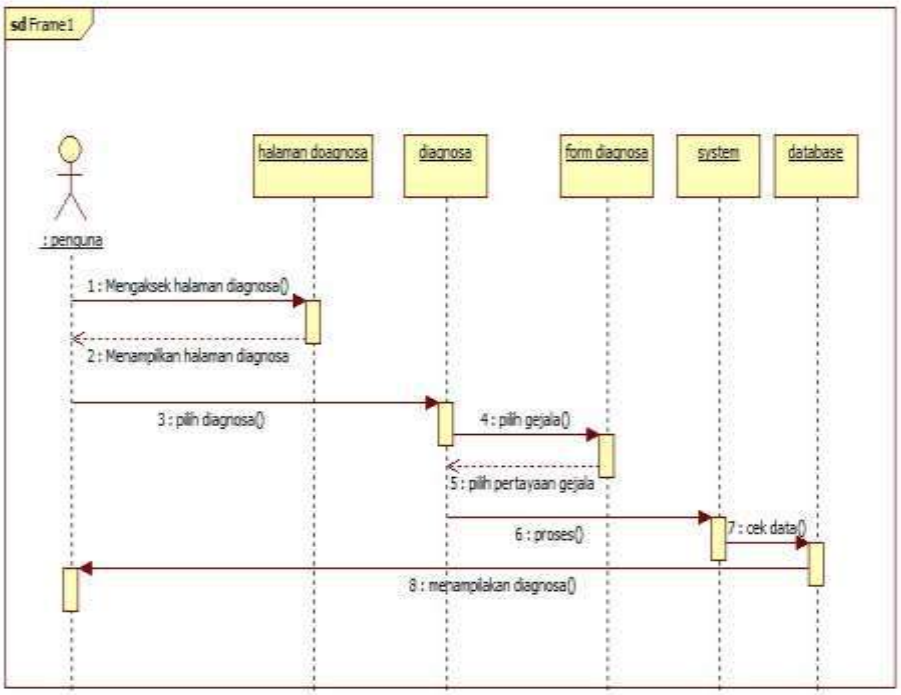
Gambar 11. *Activity Diagram* Logout (Admin)

Gambar diatas merupakan aktivitas *logout* untuk admin yang telah melakukan *login*. Klik logout apabila ingin melakukan *logout*, klik batal apabila ingin membatalkan proses *logout*.

4) *Sequence Diagram*

Berikut ini merupakan *Sequence Diagram* dari rancangan system pada mentimun.

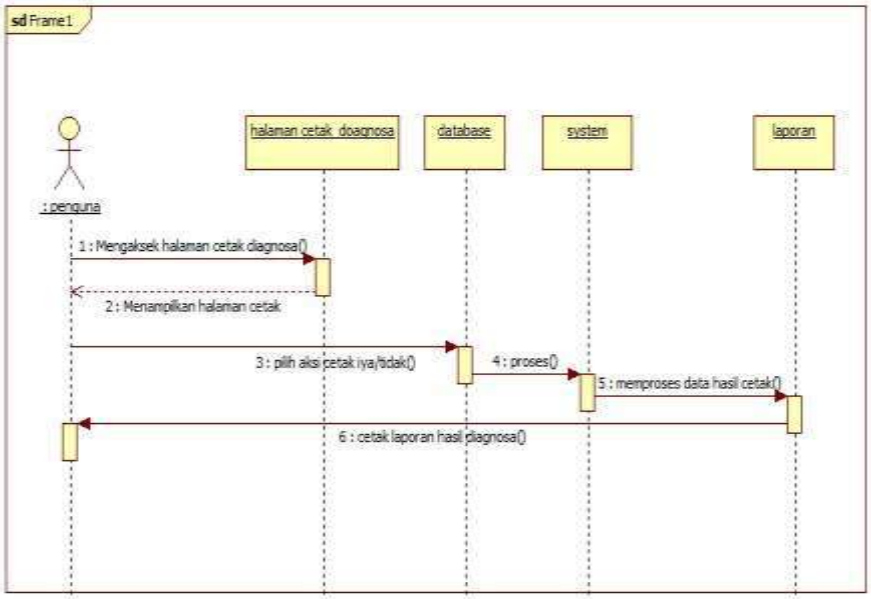
a) *Sequence Diagram* Diagnosa (pengguna)



Gambar 12. *Sequence Diagram* Diagnosa (Pengujung)

1. Untuk *user* mengakses halaman diagnosa, sistem akan merespon akan menampilkan halaman diagnosa, pilih gejala yang dialami, lalu pilih pertanyaan gejala.
2. Kemudian sistem akan memproses data jawaban tentang gejala oleh *user* dan sistem bejalan akan memproses diagnosa dan terkahir sistem akan menampilkan hasil diagnosa berdasarkan tanggapan *user*.

b) *Sequence Diagram* Cetak Hasil Diagnosa (Pengunjung)

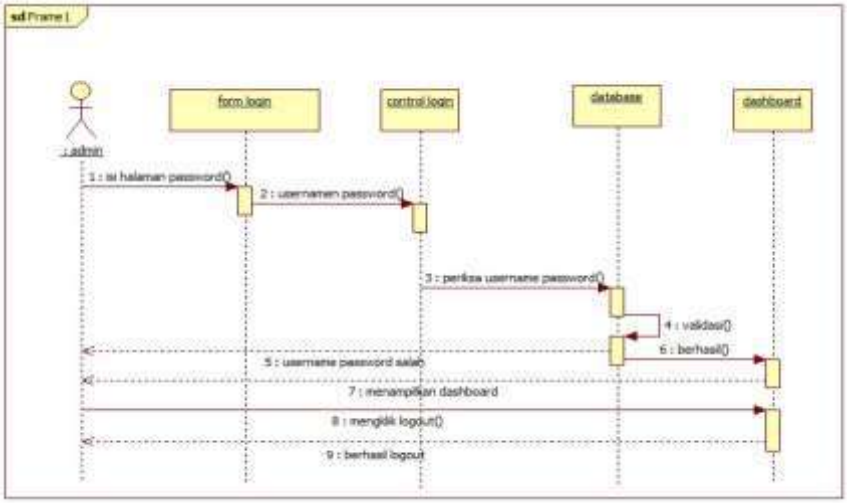


Gambar 13. *Sequence Diagram* Cetak Hasil Diagnosa (Pengunjung)

1. *User* mengakses halaman cetak hasil, kemudian sistem akan menampilkan halaman cetak, lalu user akan melakukan printah iya atau tidak untuk cetak data hasil diagnosa.

2. Kemudian sistem akan memproses printah yang dipilih oleh *user*, lalu sistem akan memproses data yang ingin di cetak dan langkah akhir sistem akan mengajak *user* mencetak hasil diagnosa.

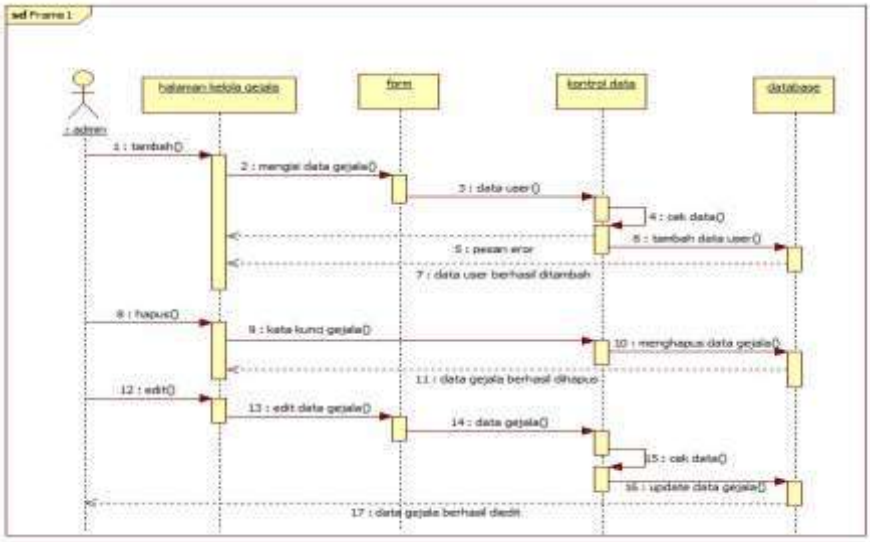
c) *Sequence Diagra*m Login (Admin)



Gambar 14. *Sequence Diagram* Login (Admin)

Admin memasukkan *user name password*, lalu data akan diperiksa apakah ada di database atau tidak. Apabila data ada maka akan berhasil masuk ke dashboard admin.

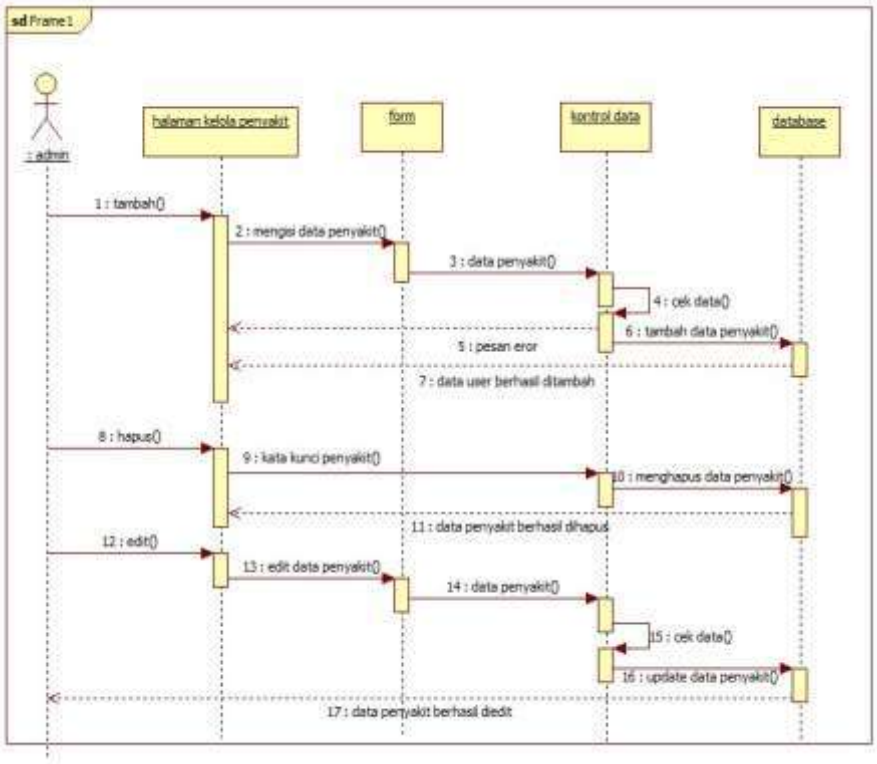
d) *Sequence Diagram* Kelola Data Gejala *(*Admin)



Gambar 15. *Sequence Diagram* Kelola Data Gejala (Admin)

Admin dapat melakukan proses pengelolaan data gejala yaitu tambah, edit dan hapus data.

e) *Sequence Diagram* Kelola Data Penyakit (Admin)

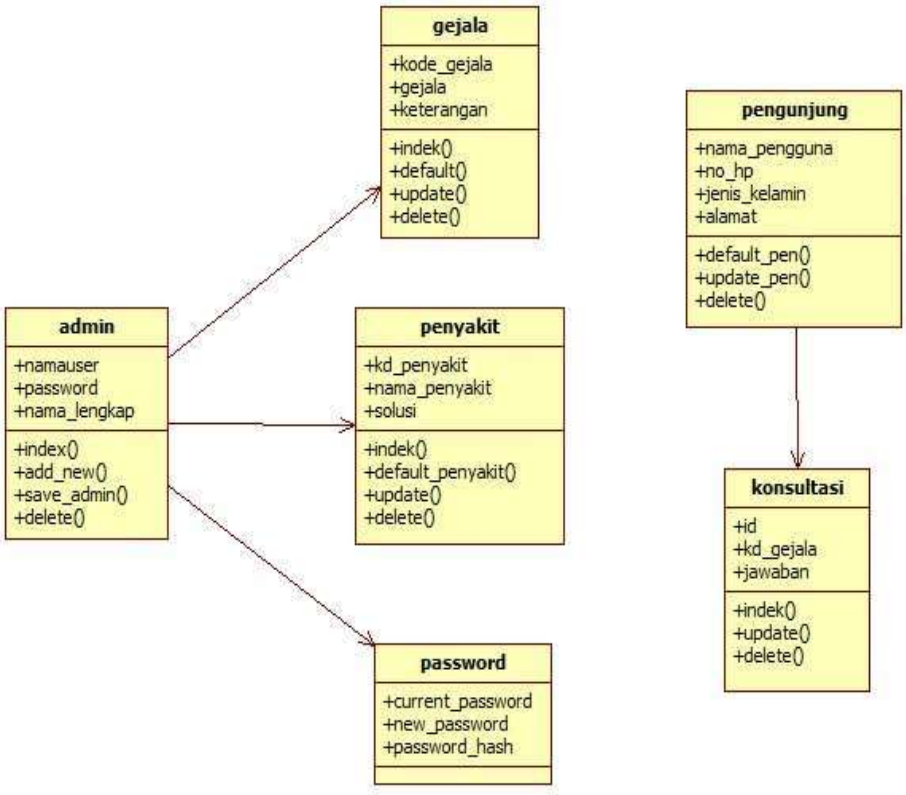


Gambar 16. *Sequence Diagram* Kelola Data Penyakit (Admin)

Admin dapat melakukan proses pengelolaan data penyakit yaitu tambah, dan hapus data.

5. *Class Diagram*

Class diagram adalah diagram yang menunjukkan kelas-kelas yang ada dari sebuah sistem yang hubungannya secara logika. Karena itu class diagram merupakan tulang punggung atau kekuatan dasar dari hampir setiap metode berorientasi objek termasuk UML. Class diagram bersifat statis yang digambarkan dengan kotak yang terbagi atas tiga bagian yaitu, nama kelas, atribut, dan operasi. Gambar 20 menggambarkan diagram kelas dari sistem yang akan dibangun.

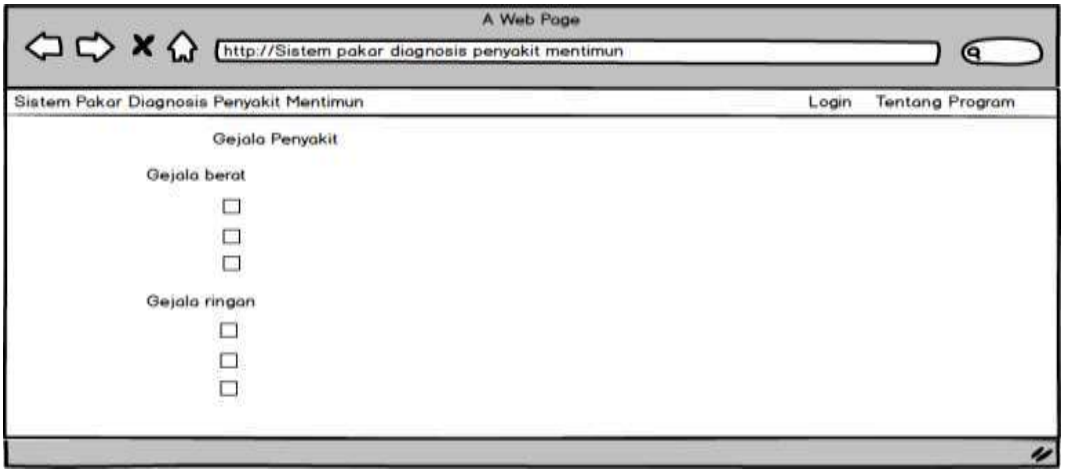


Gambar 17. *Class Diagram* Sistem Pakar Diagnosis Penyakit Tanaman Mentimun Dengan Menggunakan Metode *Certainty Factor* Berbasis Web

b. Desain Perangkat Lunak

1) Halaman Gejala Penyakit *User*

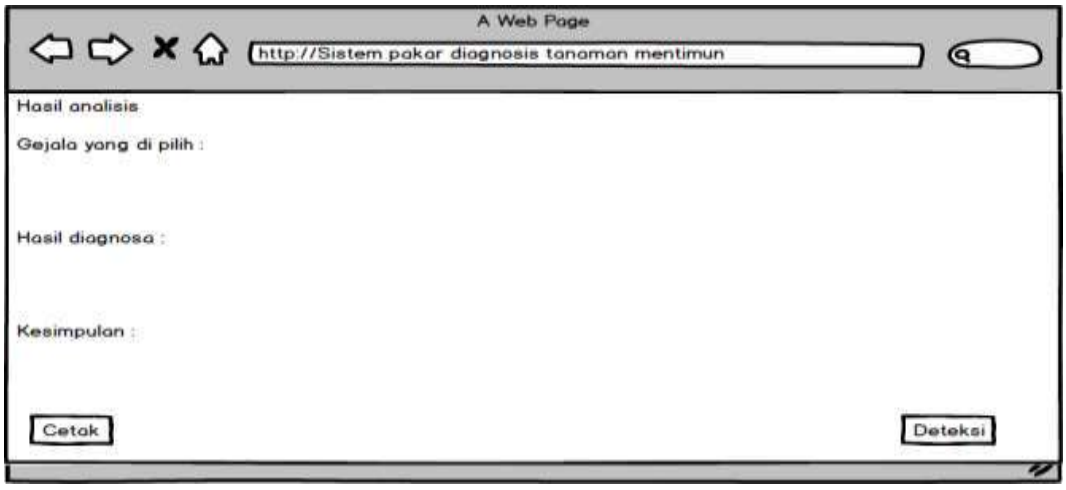
Halaman Gejala penyakit adalah menu yang di tampilkan pada saat *user* telah masuk dan dapat memilih gejala yang ingin di diagnosis



Gambar 18. Halaman Gejala Penyakit User

2) Halaman Diagnosa *User*

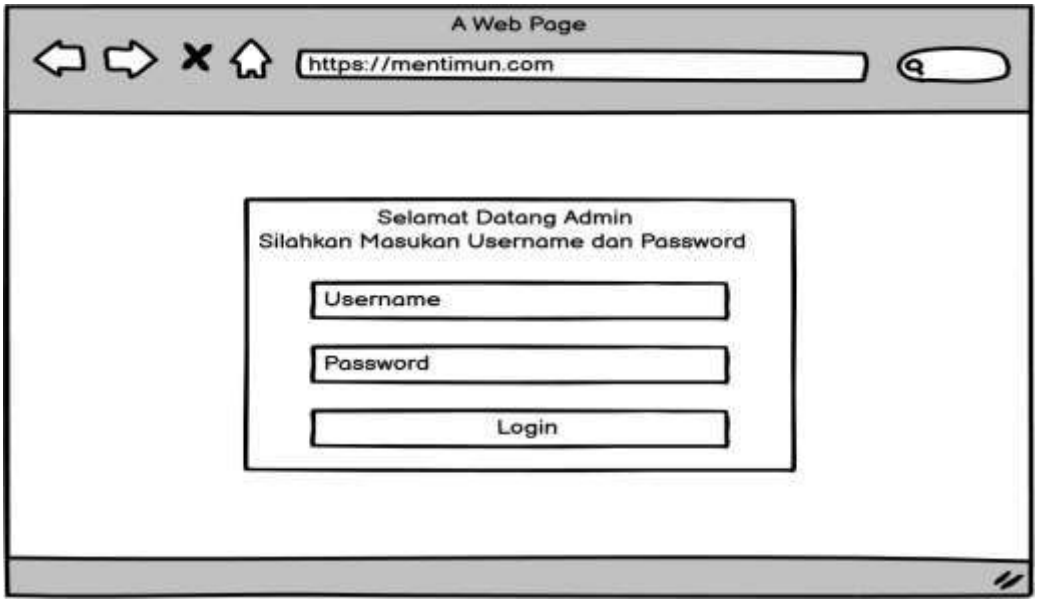
Halaman diagnosa user adalah menu yang di tampilkan pada saat pengunjung telah mendiagnosa dan memilih gejala.



Gambar 19. Halaman Diagnosa User

3) Halaman *Login* Admin

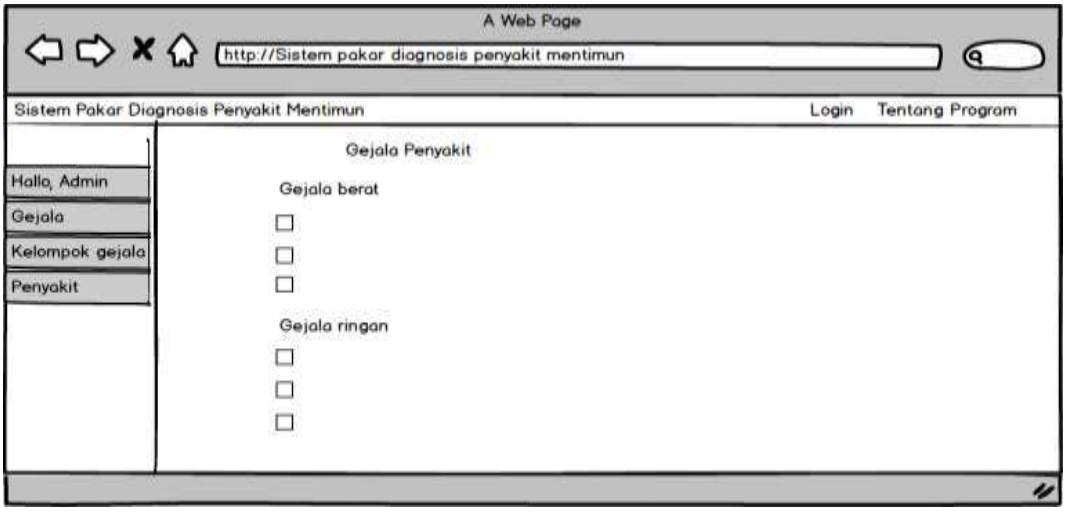
Halaman yang ditampilkan sistem adalah wadah admin akan login untuk masuk dan mengolah data.



Gambar 20. Halaman *Login* Admin

4) Halaman Menu Admin

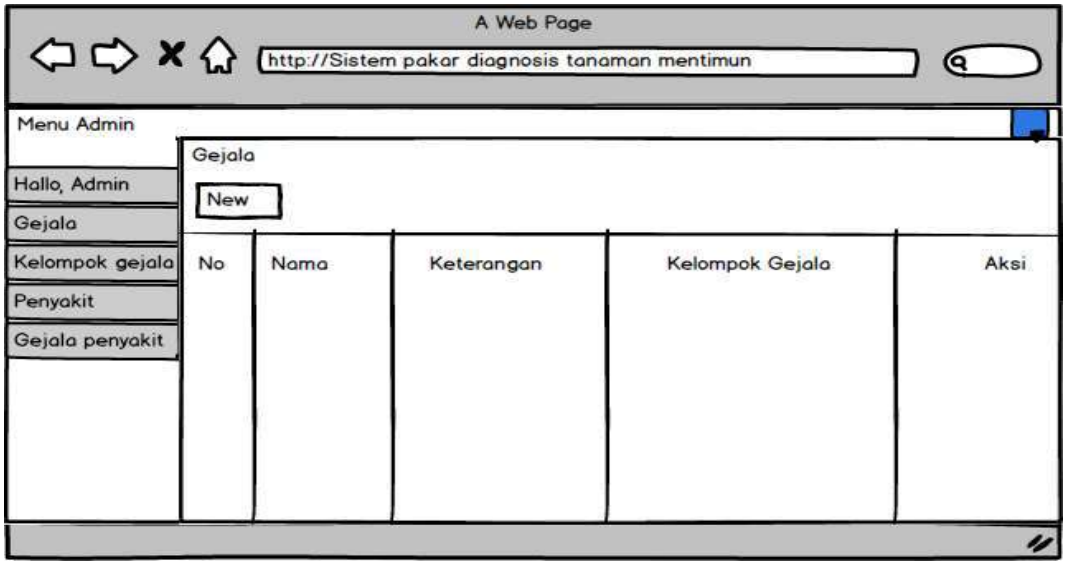
Halaman menu admin akan ditampilkan sitem ketika admin berhasil masuk dengan cara *login*.



Gambar 21. Halaman Menu Admin

5) Halaman Kelola Data Gejala

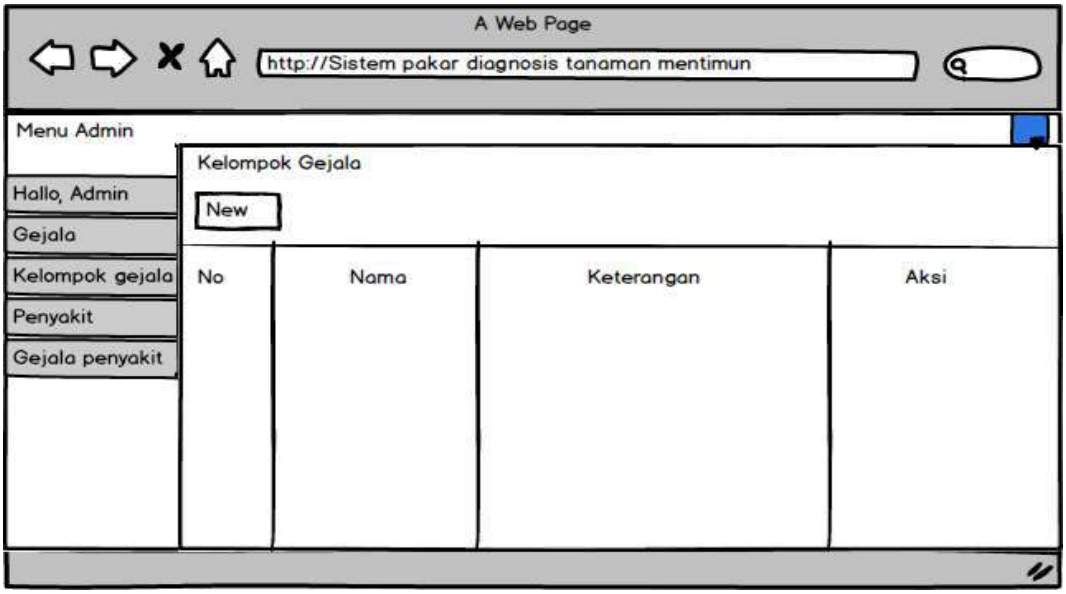
Halaman kelola data gejala akan ditampilkan sistem ketika admin akan menambah dan menghapus data penyakit.



Gambar 22. Halaman Kelola Data Gejala

6) Halaman Kelola Data Kelompok Gejala

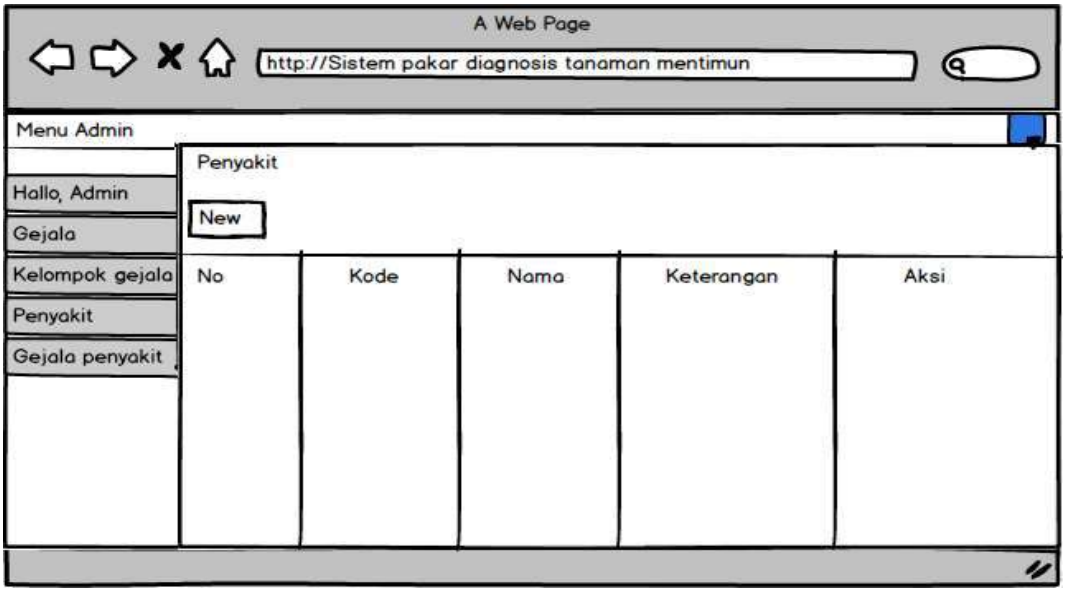
Halaman kelola data kelompok gejala akan ditampilkan sistem ketika admin akan menambah dan menghapus data kelompok gejala.



Gambar 23. Halaman Kelola Data Kelompok Gejala

7) Halaman Kelola Data Penyakit

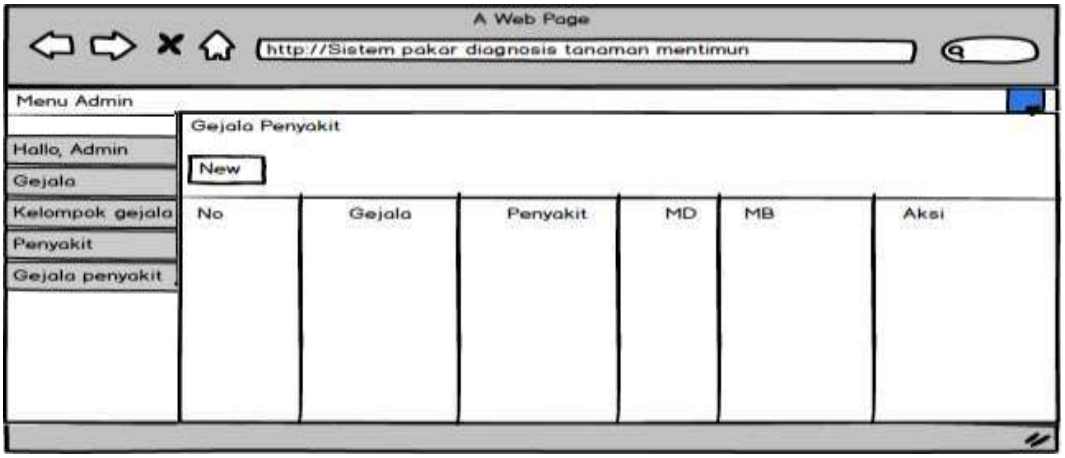
Halaman kelola data penyakit akan ditampilkan sistem ketika admin akan menambah dan menghapus data penyakit.



Gambar 24. Halaman Kelola Data Penyakit

8) Halaman Kelola Data Gejala Penyakit

Halaman kelola data gejala penyakit akan ditampilkan sistem ketika admin akan menambah dan menghapus data gejala penyakit.



Gambar 25. Halaman Kelola Data Gejala Penyakit

c. Desain Basis Data

1. Desain Tabel Basis Data

Database tb ta.sql adalah *database* yang dibuat untuk menyimpan data-data yang akan digunakan dalam proses pengolahan data admin, data gejala, data penyakit, dan data basis_pengetahuan. Berikut adalah desain dari tabel-tabel yang dibuat di dalam *database tb_ta.sql* ini.

1) Desain Tabel Admin

Admin merupakan tabel yang menyimpan data admin yang digunakan untuk login admin pada sistem. Berikut spesifikasi dari admin :

Tabel 15. Tabel Spesifikasi Admin

Nama	Jenis	Kosong
Username	varchar(20)	Tidak
password	varchar(32)	Tidak
nama_lengkap	varchar(30)	Tidak

2) Desain Tabel Gejala

Tabel gejala adalah tabel yang berisi gejala pada mentimun dengan segaligus spesifikasinya.

Tabel 16. Tabel Data Gejala

Nama	Jenis	Kosong
kd_gejala	varchar(5)	Tidak
Gejala	varchar(100)	Tidak
keterangan	text	Tidak

3) Tabel diagnosa

Tabel ini berisikan data tentang penyakit mentimun. Berikut spesifikasinya.

Tabel 17. Tabel data penyakit

Nama	Jenis	Kosong
kode_penyakit	int(11)	Tidak
nama_penyakit	varchar(50)	Tidak
Solusi	Text	Tidak

4) Desain Tabel Basis_Pengetahuan

Tabel menentukan ini berisi tentang basis pengetahuan dan perhitungan cf dengan no mb dan md.

Tabel 18. Tabel Basis Pengetahuan

Nama	Jenis	Kosong
kode_pengetahuan	int(11)	Tidak
kode_penyakit	int(11)	Tidak
kode_gejala	int(11)	Tidak
Mb	double(11,1)	Tidak
Md	double(11,1)	Tidak

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil

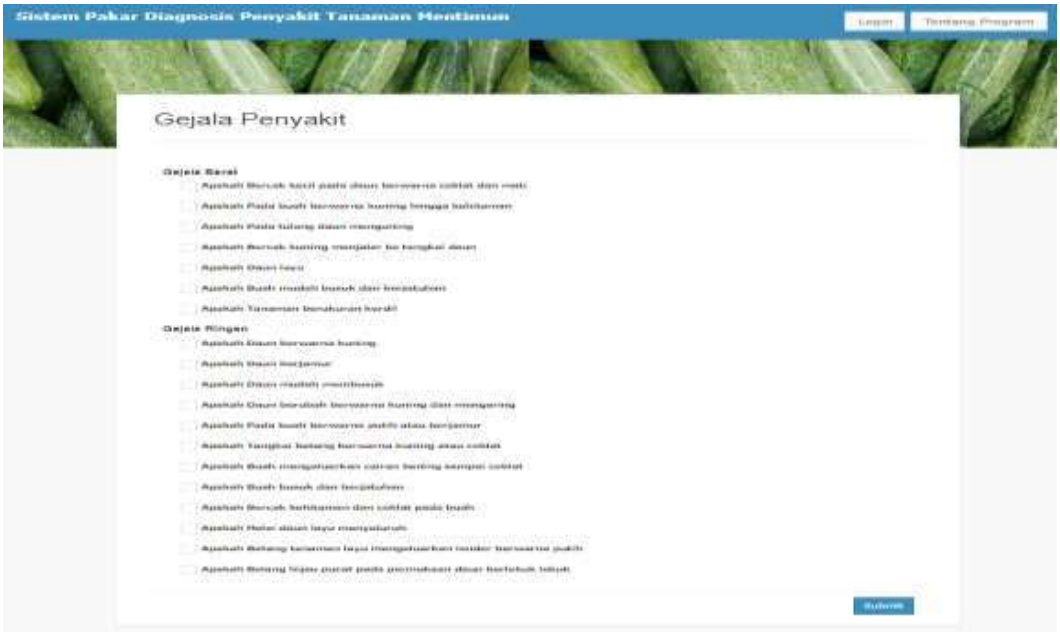
4.1.1 Implementasi

Sistem Pakar Diagnosis Tanaman Mentimun Dengan Menggunakan Metode *Certainty Factor* Berbasis Web**,** telah di implementasikan ke dalam program berbasis Web dengan metode *Certainty Factor*, dengan hasil yang dapat dilihat sebagai berikut:

a. Tampilan Halaman Awal

1) Tampilan Gejala Penyakit

Tampilan Halaman *Home* Gejala Penyakit merupakan tampilan awal dalam web ini, dalam tampilan home ini ada terdiri



Gambar 26. Tampilan Home Gejala Penyakit

2) Tampilan Halaman Tentang Program

Tampilan halaman tentang merupakan penjelas terkait pembuat Aplikasi, nama aplikasi dan Metode yang digunakan.

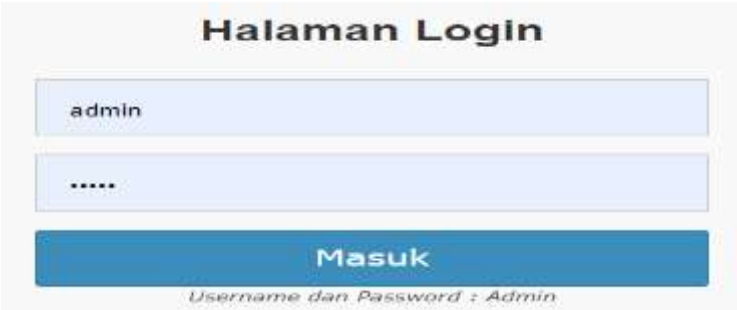


Gambar 27. Tampilan Halaman Tentang Program

b. Tampilan Halaman *Admin*

1) Tampilan Halaman *Login (Admin)*

Sebelum masuk halaman admin, mengisi form admin terlebih dahulu dengan mengisi *username* dan *password*.



Gambar 28. Tampilan Halaman Login (Admin)

2) Tampilan Halaman Menu *(Admin)*

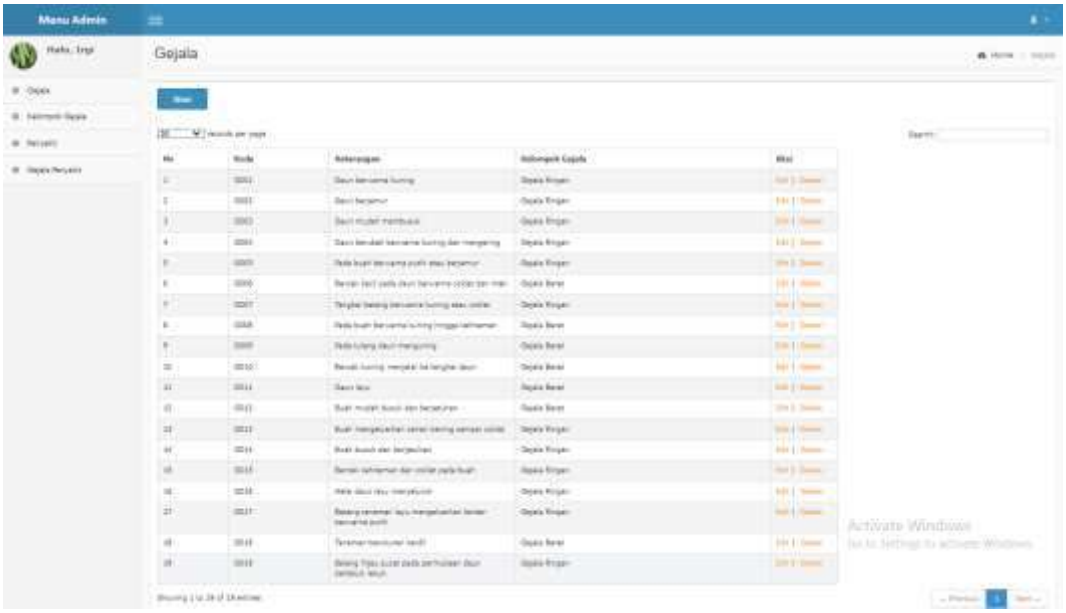
Pada halaman menu admin akan tampil seperti halaman di bawah, dengan opsi pilihan Gejala, Kelompok gejala, Penyakit, Gejala penyakit, kemudian untuk *logout* ada pada *button* sebelah kanan



Gambar 29. Tampilan Halaman Menu *(Admin)*

3) Tampilan Halaman Gejala *(Admin)*

Di dalam halaman Gejala akan tampil halaman seperti di bawah. Dengan beberapa opsi pilihan menu dari *new*, edit dan *delete*.



Gambar 30. Tampilan Halaman Gejala *(Admin)*

4) Tampilan Halaman Kelompok Gejala *(Admin)*

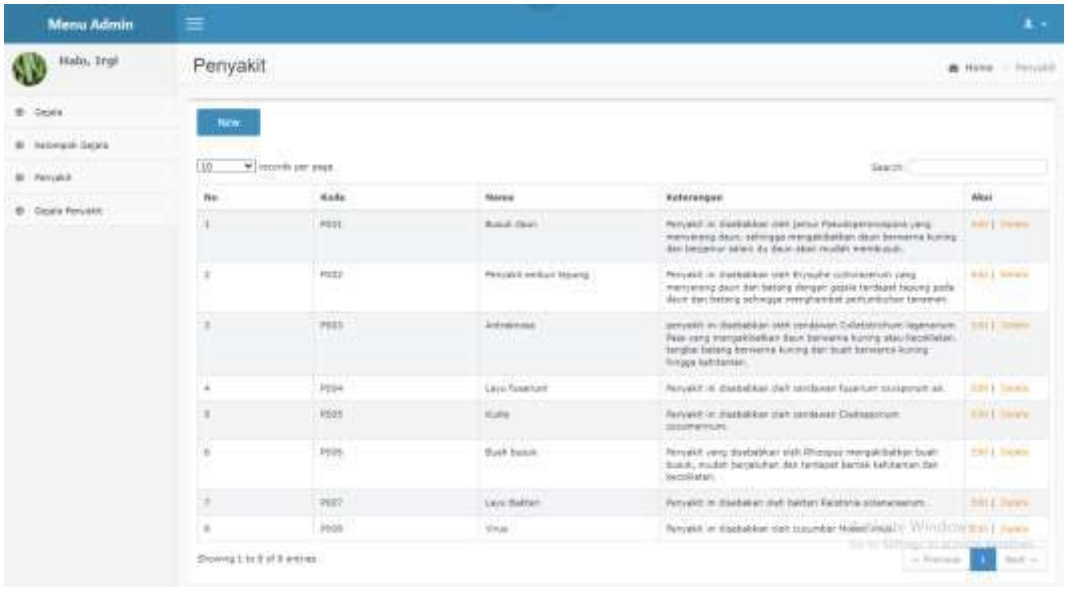
Pada halaman kelompok gejala akan tampil halaman seperti di bawah. dengan opsi pilihan menu *new*, edit dan *delete.*



Gambar 31. Tampilan Halaman Kelompok Gejala *(Admin)*

5) Tampilan Halaman Penyakit *(Admin)*

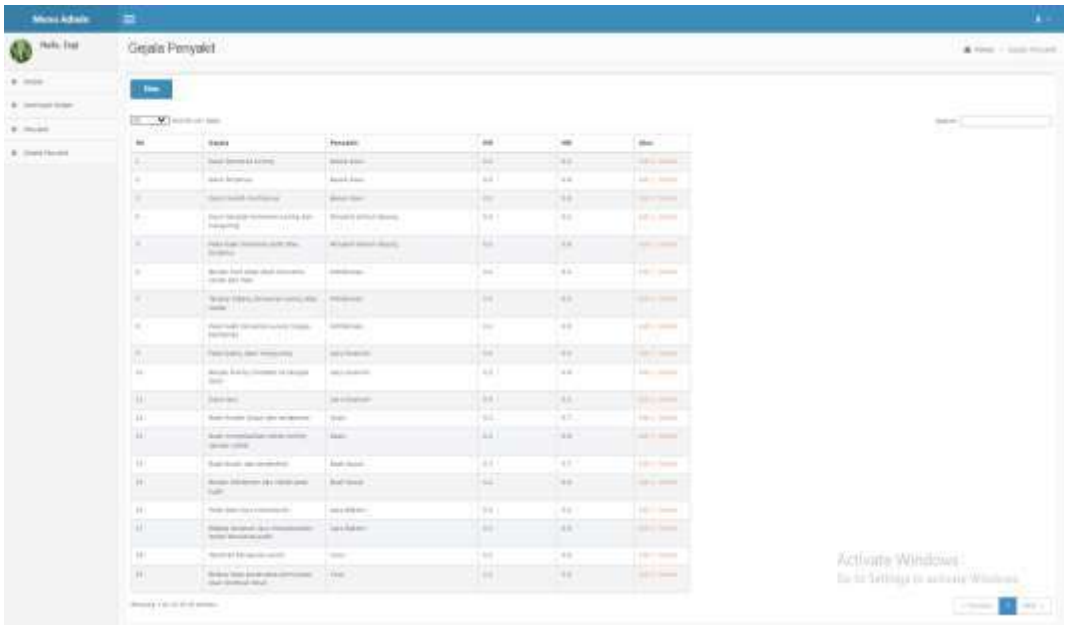
Pada halaman penyakit akan tampil halaman seperti di bawah.Dengan opsi pilihan menu *new*, edit dan *delete.*



Gambar 32. Tampilam Halaman Penyakit *(Admin)*

6) Tampilan Halaman Gejala Penyakit *(Admin)*

Pada halaman Gejala penyakit akan tampil halaman seperti di bawah.Dengan opsi pilihan menu *new*, edit dan *delete.*

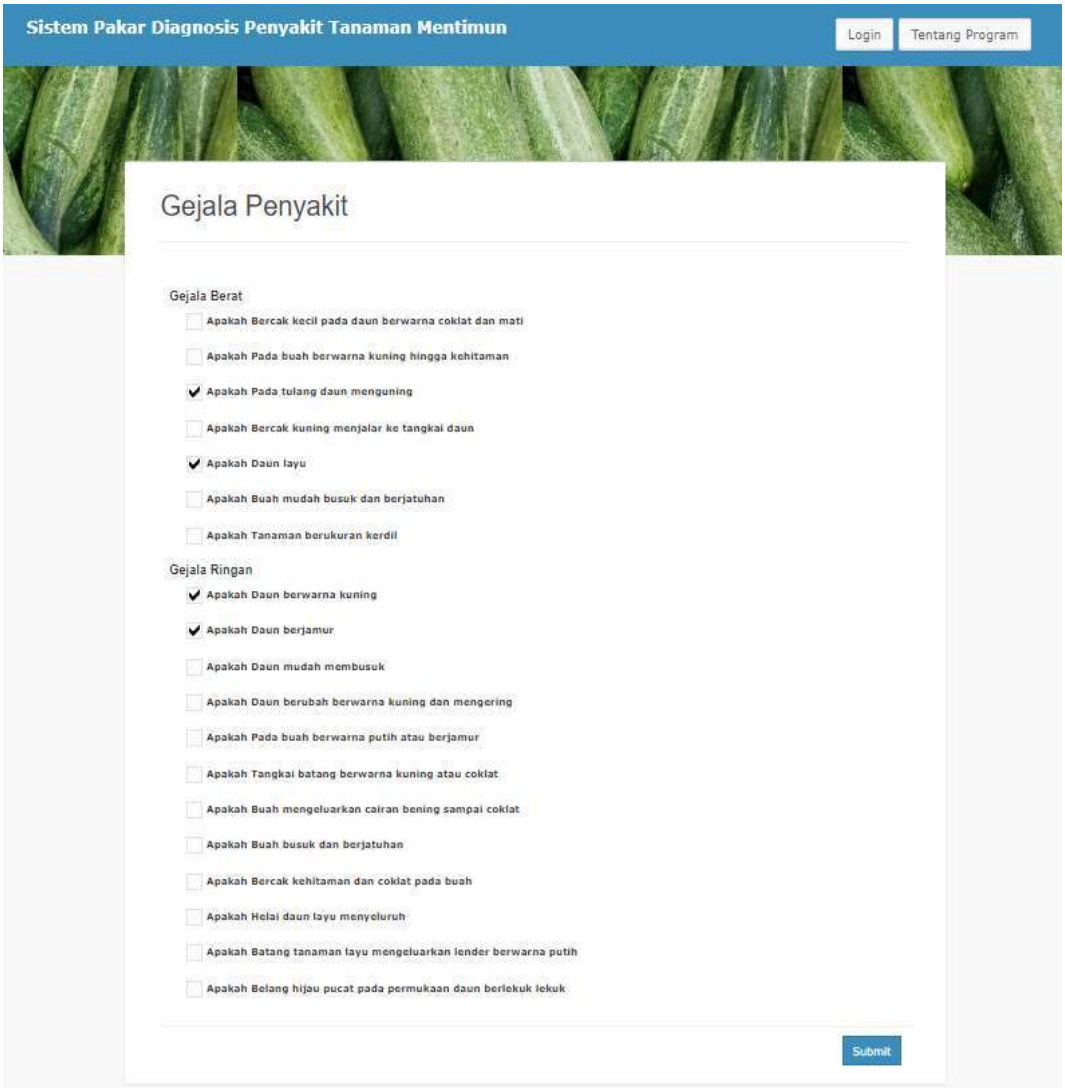


Gambar 33. Tampilan Halaman Gejala Penyakit *(Admin)*

c. Tampilan Halaman *User*/Pengunjung

1) Tampilan Halaman Gejala Penyakit *User*/Pengunjung

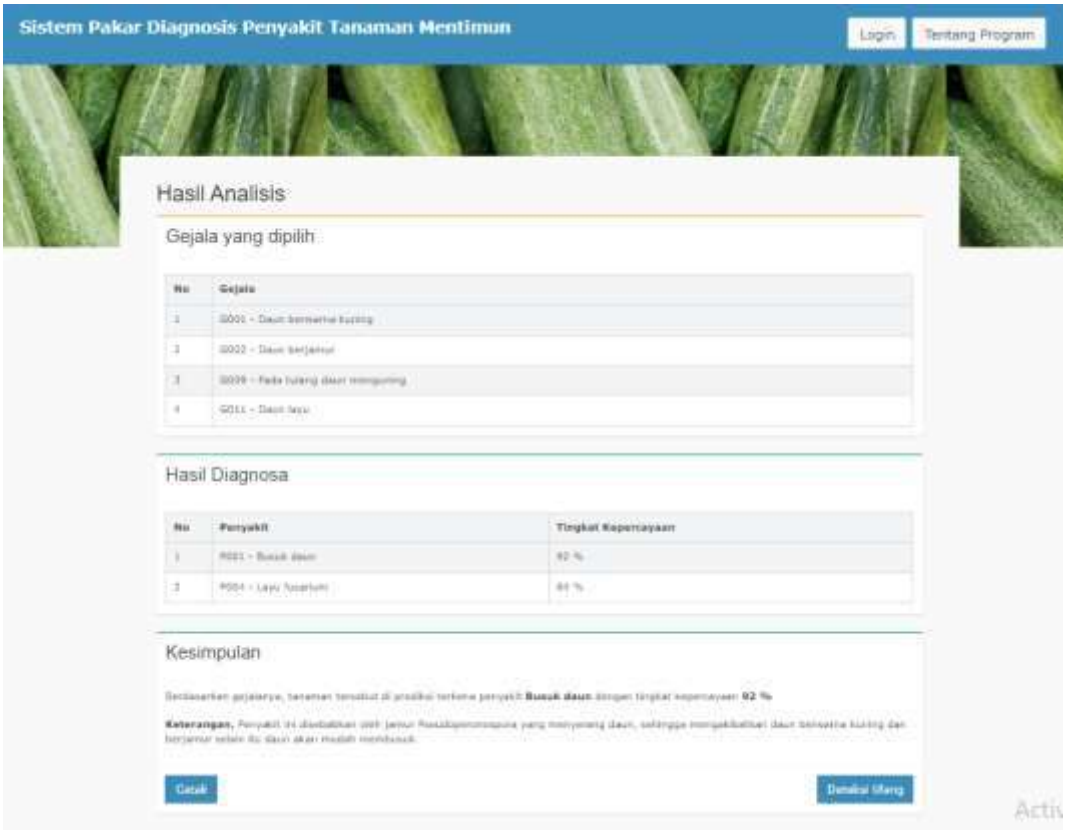
Pada halaman ini pengunjung langsung dapat memilih gejala penyakit yaitu, gejala berat 7 dan gejala ringan 12. Pilih 2 gejala berat dan 2 untuk gejala ringan.



Gambar 34. Tampilan Halaman Gejala Penyakit User/Pengunjung

2) Tampilan Halaman Diagnosis Hasil

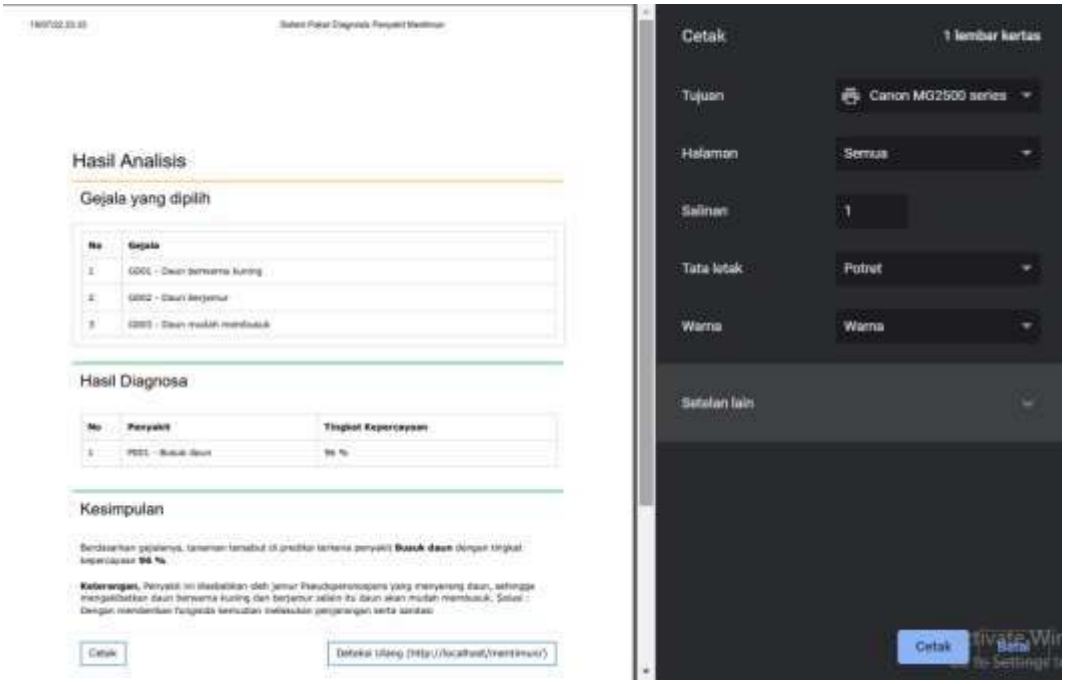
Pada halaman ini dapat di lihat hasil diagnosis dari 2 gejala berat dan 2 gejala ringan, dan apabila ingin cetak dan deteksi ulang tinggal klik.



Gambar 35. Tampilan Halaman Diagnosis Hasil

3) Tampilan Halaman Cetak Hasil Diagnosis

Pada halaman ini dapat dilihat Cetak hasil diagnosis



Gambar 36. Halaman Cetak Hasil Diagnosis

4.1.2 Pengujian Sistem

Pada umunnya terdapat dua macam teknik pengujian *black box testing*. Pengujian *black box testing* tidak terlalu memperhatikan struktur internal (*source code*) program, tetapi lebih menitik beratkan kepada kesalahan program. Pada Sistem Pakar Diagnosis Penyakit Tanaman Mentimun Dengan Menggunakan *Metode Certainty Factor* Berbasis Web digunakan pengujian *black box testing*.

Pengujian pertama dilakukan menggunakan *black box testing*. Tujuan dari metode black box testing ini adalah untuk menentukan kesalahan fungsi pada program. Pengujian black box dapat dilihat pada Tabel 19.

Tabel 19. Tabel Uji Sistem

No	Rancangan Proses	Hasil yang Diharapkan	Keterangan
1	Data login salah isi (memasukkan data username dan password tidak benar)	Gagal masuk dan muncul pemberitahuan "Login Gagal"	**SESUAI**
2	Pengujian Login Benar	Muncul tampilan halaman beranda admin	**SESUAI**
3	Pengujian Masuk Tentang Aplikasi	Halaman tetang aplikasi muncul	**SESUAI**
4	Pengujian Tambah data Gejala	Data Penyakit berhasil disimpan (muncul laporan "Data Gejala Berhasil Di Input")	**SESUAI**
5	Pengujian Tambah data Penyakit	Data Penyakit berhasil disimpan (muncul laporan "Penyakit Berhasil Di Input")	**SESUAI**
6	Pengujian Menu masuk diagnosa	Masuk ke halaman diagnosa Penyakit	**SESUAI**
7	Pengujian Diagnosa	Hasil Diagnosa Tampil	**SESUAI**

Dari hasil pengujian yang dilakukan, dapat disimpulkan bahwa sistem sudah berjalan sesuai dengan yang diharapkan. Maka sistem ini sudah dapat menghasilkan keluaran yang di harapkan.

4.1.3 Perhitungan Certainty Factor

Rumus Certainty factor adalah sebagai berikut :

$CFc(CF_1,CF_2)$ =

$CF_1+CF_2\ (1-CF_1)$ if both > 0

$CF_1+CF_2\ (1+ CF_1)$if both < 0

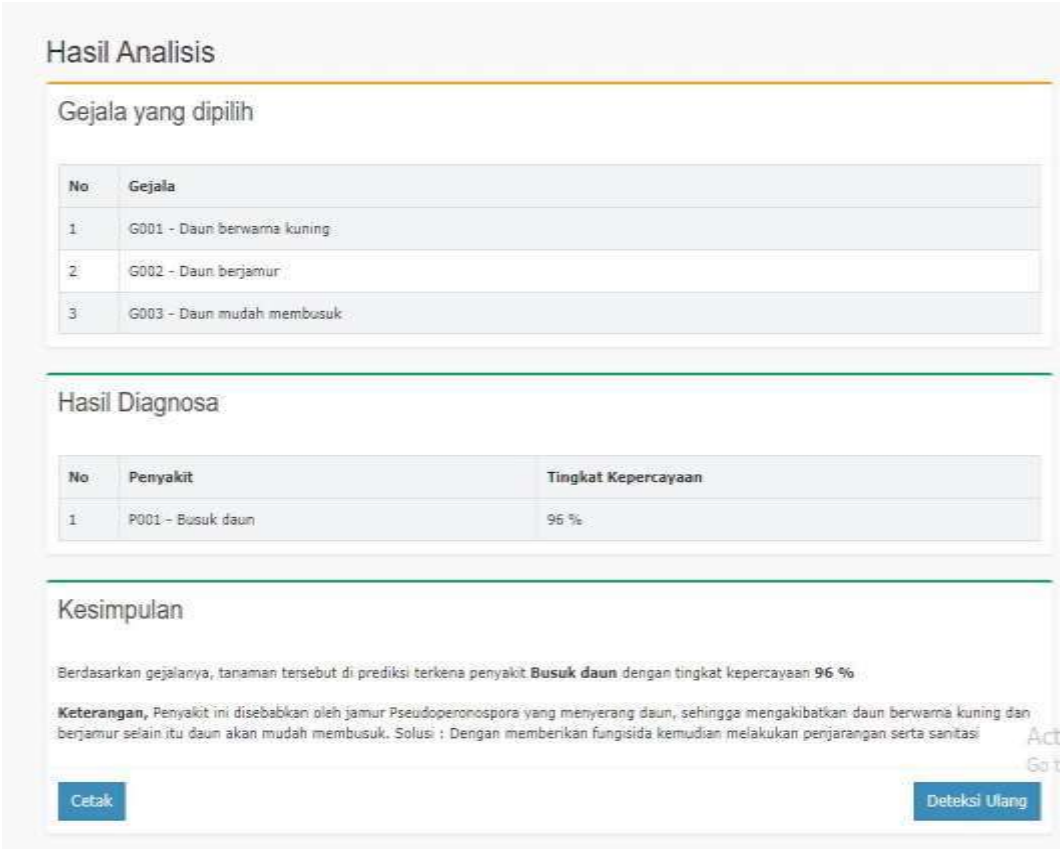
$CF_1,CF_2/(1-((CF_1),(CF_2))$ if one < 0

Pada program yang dibuat penulis adalah menggunakan rumus pertama. Yaitu jika nilai CF bernilai > 0 atau positif, maka rumusnya sebagai berikut :

$CF_1+CF2(1-CF_1)$ Berikut adalah contoh perhitungan manual program (menggunakan Rule 1) :

Tabel 20. Tabel Nilai CF Gejala pada Penyakit Busuk daun

No	Nama Gejala	CF
1	Daun berwarna kuning	0.6
2	Daun berjamur	0.8
3	Daun mudah membusuk	0.8



Gambar 37. Hasil Analisis Gejala Yang Dipilih

Ketika dicek pada pemrograman, maka hasilnya sesuai dengan hasil perhitungan manual. Diagnosa penyakit Busuk Daun (P001). Rule : if G01 and G02 and G03 then P01

Maka nilai CF Penyakit Busuk Daun diperoleh dengan cara :

CF(A) = CF(1) + [CF(2) * (1 – CF(1)] = 0.6 + [0.8 * (1 – 0.6)] = 0.92

0.6 + 0.8 × 0.6 = 0.6 + 0.8 (1 – 0.6)

0.6 + 0.48 = 0.6 + 0.8 × 0.4

1.08 = 0.6 + 0.32

1.08 = 0.92

CF(B) = CF(3) + [CF(A) * (1 – CF(3)] = 0.5 + [0.92 * (1 – 0.5)] = 0.96

0.5 + 0.6 × 0.5 = 0.5 + 0.92 (1 – 0.5)

0.5 + 0.3 = 0.5 + 0.92 × 0.5

0.8 = 0.5 + 0.46

0.8 = 0.96

Sehingga CF penyakit Busuk Daun adalah 0.96 (dibulatkan menjadi 1 atau 100 %)

4.2 Pembahasan

4.2.1 Pembahasan Hasil Responden

Berikut ini data hasil responden dari 3 pernyataan koesioner adalah sebagai berikut.

1. Pernyataan Pertama

   Apakah website Sistem Pakar Diagnosis Penyakit Tanaman Mentimun berjalan dengan baik ?

   Tabel pengolahan untuk pernyataan pertama dapat dilihat pada tabel 21.

Tabel 21. Hasil Responden Pertanyaan Ke 1

Jawaban	Skor	Jumlah Responden	(Skor*Jumlah Responden	Nilai Persentase
Sangat Tidak Baik	1	0	0	74%
Tidak Baik	2	0	0	
Cukup Baik	3	1	3	
Baik	4	9	36	
Sangat Baik	5	7	35	
Total Skor		**17**	**74**	

2. Pernyataan Kedua

   Apakah hasil diagnosa memuaskan ?

   Tabel pengolahan untuk pernyataan ketiga dapat dilihat pada tabel 22.

Tabel 22. Hasil Responden Pertanyaan Ke 2

Jawaban	Skor	Jumlah Responden	(Skor*Jumlah Responden	Nilai Persentase
Sangat Tidak Baik	1	0	0	70%
Tidak Baik	2	1	2	
Cukup Baik	3	1	3	
Baik	4	10	40	
Sangat Baik	5	5	25	
Total Skor		**17**	**70**	

3. Pernyataan Ketiga

Apakah ada kesulitan ketika melakukan diagnosa?

Tabel pengolahan untuk pernyataan ketiga dapat dilihat pada tabel 23.

Tabel 23. Hasil Responden Pertanyaan Ke 3

Jawaban	Skor	Jumlah Responden	(Skor*Jumlah Responden	Nilai Persentase
Sangat Tidak Baik	1	0	0	53%
Tidak Baik	2	0	0	
Cukup Baik	3	1	3	
Baik	4	10	40	
Sangat Baik	5	6	30	
Total Skor		**17**	**53**	

Berdasarkan hasil responden dari ketiga pernyataan tersebut dapat disimpulkan bahwa aplikasi sistem pakar diagnosis penyakit tanaman mentimun berada pada kategori yang baik karena rata-rata dari persentase dari 3 pernyataan dan 17 orang yang memberikan pernyataan tersebut adalah 47%. Adapaun perhitungan secara keseluruhan dapat dilihat pada tabel 24.

Tabel 24. Hasil Responden Keseluruhan

No Pernyataan	**Nilai Persentase**	**Keterangan**
1	74%	BAIK
2	70%	BAIK
3	53%	BAIK
Total Persentase	**197**	BAIK
Rata - Rata	**197%/3=65%**	BAIK

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan uraian yang telah di bahas pada bab-bab sebelumnya, maka dapat disampaikan beberapa hal diantaranya adalah sebagai berikut :

a. Sistem telah dapat menghasilkan informasi bagaimana cara menangani penyakit yang sedang dialami tanaman mentimun.
b. Pengguna dapat mencetak hasil diagnosa.
c. Berdasarkan perhitungan manual program dengan sistem, metode *Certainty Factor* ini mampu memberikan hasil berdasarkan bobot gejala yang telah dipilih pengguna pada sistem dan bisa memberikan jawaban pada kasus yang tidak pasti kebenarannya seperti masalah pada penelitian ini yaitu diagnosa suatu penyakit.

5.2 Saran

Dari hasil penulis dan kesimpulan dapat diambil intisari saran dalam penelitian ini, penulis ingin memberikan beberapa saran yang diharapkan berguna untuk pengembangan lebih lanjut terhadap aplikasi ini yaitu :

a. Sistem pakar ini dapat dibuat menggunakan metode yang lain.
b. Jenis Penyakit bisa ditambah lebih banyak lagi untuk pengembangan sistem pakar ini.
c. Untuk penelitian selanjutnya dapat dikembangkan berbasis android.

DAFTAR PUSTAKA

Sistem Pakar Diagnosa Penyakit Tanaman Kakao Menggunakan Metode Certainty Factor Pada Kelompok Tani Pt Olam Indonesia (Cocoa) Cabang Lampung. (2020). *JDMSI, Vol. 1, No. 4, 2020, 1*, 1-6.

Alfie Nur Rahmi, I. V. (2019). Sistem Pakar Diagnosa Penyakit Dan Hama Pada Tanaman Mentimun Menggunakan Metode Forward Chaining. *Intechno Journal (Information Technology Journal) Volume 1 No. 3 Mei 2019, 1*, 18-22.

Aryasa, K. (2018). Sistem Pakar Diagnosa Penyakit Tanaman Agribisnis Menggunakan Metode Certainty Factor. *Jurnal Sistem Informasi Dan Teknologi Informasi, Vol. 7, No. 1, April 2018, 7*, 54-67.

Dina Maulina, A. M. (2020). Jurnal Of Information System Management Vol. 1, No. 2 (2020). *Metode Certainty Factor Dalam Penerapan Sistem Pakar Diagnosa Penyakit Anak, 1*, 23-32.

Joan Angelina Widians, F. N. (2020). Identifikasi Hama Kelapa Sawit menggunakan Metode Certainty Factor. *ILKOM Jurnal Ilmiah, Vol. 12 No. 1, April 2020, 12*, 58-63.

Julianto Simatupang, S. S. (2019). Perancangan Sistem Informasi Pemesanan Tiket Bus Pada Po. Handoyo Berbasis Online. *Jurnal Intra-Tech Volume 3, No. 2 Oktober 2019, 3*, 11-25.

Latumakulita, L. A. (2012). Sistem Pakar Pendiagnosa Penyakit Anak Menggunakan Certainty Factor (CF). *Jurnal Ilmiah Sains Vol. 12 No. 2, Oktober 2012, 12*, 120-126.

Normawardah. (2019). Sistem Pakar Diagnosis Penyakit Tanaman Jagung Menggunakan Metode Certainty Factor Berbasis Web. *JATI (Jurnal Mahasiswa Teknik Informatika) Vol. 3 No. 1, Maret 2019, 3*, 219-226.

Puji Karuniawan, I. N. (2021). Implementasi Metode Certainty Factor Untuk Mengidentifikasi Penyakit Tanaman Kedelai Dan Padi. *Jurnal NOE, Vol 4, No 01 April 2021, 4*, 1-9.

Rizal Rachman, A. M. (2018). Jurnal Ilmu Komputer dan Informatika Vol. 4 No. 2 Desember 2018. *Penerapan Metode Certainty Factor pada Sistem Pakar Penentuan Minat dan Bakat Siswa SD, 4*, 90-97.

Rohmat Indra Borman, R. N. (2020). Implementasi Certainty Factor Dalam Mengatasi Ketidakpastian Pada Sistem Pakar Diagnosa Penyakit Kuda Laut. *JURTEKSI (Jurnal Teknologi dan Sistem Informasi) Vol. 7 No. 1, Desember 2020, 7*, 1-8.

Rudi Hariyanto, K. S. (2018). Sistem Pakar Diagnosis Penyakit dan Hama Pada Tanaman Tebu Menggunakan Metode Certainty Factor. *(JOINTECS) Journal of Information Technology and Computer Science Vol. 3, No. 1, Januari 2018, 3*, 29-32.

Sihotang, H. T. (2018). Sistem Pakar Untuk Mendiagnosa Penyakit Pada Tanaman Jagung Dengan Metode Bayes. *Journal Of Informatic Pelita Nusantara Volume 3 No 1 Maret 2018, 3*, 17-22.

Sinaga, M. D. (2014). Sistem Pakar Mendeteksi Penyakit Tanaman Terong Belanda dengan Menggunakan Metode Forward Chaining. *Jatisi, Vol. 1 No. 1 September 2014, 1*, 101-110.

Syaiful Amri, R. F. (2021). Sistem Pakar Mendiagnosis Penyakit Pada Tanaman Mentimun menggunakan Metode Dempster Shafer Berbasis Android. *Jurnal Nasional Komputasi dan Teknologi Informasi Vol. 4 No. 2, April 2021, 4*, 178-184.

LAMPIRAN