

NASKAH PUBLIKASI

PROYEK TUGAS AKHIR

**SISTEM BERBASIS PENGETAHUAN PENDETEKSI
PENYAKIT TANAMAN PADI DENGAN METODE FORWARD
CHAINING DAN CERTAINTY FACTOR**

Program Studi Informatika
Fakultas Teknologi Informasi dan Elektro



Disusun oleh
Apfia Trezia Mangalla'
5150411404

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN ELEKTRO
UNIVERSITAS TEKNOLOGI YOGYAKARTA
2020**

Naskah Publikasi

PROYEK TUGAS AKHIR

**SISTEM BERBASIS PENGETAHUAN PENDETEKSI
PENYAKIT TANAMAN PADI DENGAN METODE FORWARD
CHAINING DAN CERTAINTY FACTOR**

Program Studi Informatika
Fakultas Teknologi Informasi dan Elektro

Disusun oleh :
Apfia Trezia Mangalla'
5150411404

Telah disetujui oleh pembimbing

Pembimbing

Yuli Asriningtias, S.Kom., M.Kom.

Tanggal: 03/03/22

SISTEM BERBASIS PENGETAHUAN PENDETEKSI PENYAKIT TANAMAN PADI DENGAN METODE FORWARD CHAINING DAN CERTAINTY FACTOR

Apfia Trezia Mangalla'

*Program Studi Informatika, Fakultas Teknologi Informasi dan Elektro
Universitas Teknologi Yogyakarta*

Jl. Ringroad Utara Jombor Sleman Yogyakarta

Email: apfiatreziamangalla69@gmail.com

ABSTRAK

Tanaman padi dapat diserang berbagai macam penyakit, penyakit tersebut dapat diketahui dari gejala-gejala yang ditimbulkannya, akan tetapi untuk mengetahui secara tepat jenis penyakit yang menyerang padi tersebut, memerlukan seorang pakar/ahli pertanian. Sedangkan jumlah pakar pertanian terbatas dan tidak dapat mengatasi permasalahan petani dalam waktu yang bersamaan, sehingga diperlukan suatu sistem yang mempunyai kemampuan seperti seorang pakar, yang mana didalam sistem ini berisi pengetahuan keahlian seorang pakar pertanian mengenai penyakit dan gejala tanaman padi. Pada penelitian ini dirancang sistem berbasis pengetahuan yang berbasis web menggunakan basis aturan (rule based reasoning) dengan metode inferensi forward chaining setelah itu dihitung dengan metode certainty factor yang dimaksudkan untuk membantu petani dalam mendiagnosa penyakit tanaman padi. Sistem berbasis pengetahuan diagnosa penyakit tanaman padi berbasis web yang telah dikembangkan mempunyai keunggulan dalam kemudahan akses dan kemudahan pemakaian. Dengan fitur yang berbasis web yang dimiliki, sistem berbasis pengetahuan untuk diagnosa penyakit tanaman padi yang telah dibangun dapat digunakan sebagai alat bantu untuk diagnosa penyakit tanaman padi dan dapat diakses oleh petani dimanapun juga untuk mengatasi persoalan keterbatasan jumlah pakar pertanian dalam membantu petani mendiagnosa penyakit tanaman padi.

Kata kunci: Forward Chaining, Rule Based Reasoning, Certainty Factor, Tanaman Padi

1. PENDAHULUAN

1.1. Latar Belakang

Data Badan Pusat Statistik Kabupaten Kulon Progo (2017), menunjukkan bahwa produksi padi di Kulon Progo pada tahun 2017 adalah 113.359 ton. Namun demikian, produksi padi tersebut mengalami penurunan sebesar 2,66 persen atau sebesar 3.093 ton dibandingkan dari tahun sebelumnya. Beberapa faktor yang menyebabkan penurunan produksi padi, diantaranya: berkurangnya area sawah, banyaknya jenis hama yang menyerang dan penyakit tanaman padi yang belum dapat diatasi dengan baik. Salah satu faktor yang paling merugikan dalam produksi tanaman padi adalah penyakit.

Agar dapat memberikan solusi terhadap suatu permasalahan yang telah diuraikan tersebut maka

penulis mencoba untuk mengembangkan suatu sistem berbasis pengetahuan pendeteksi penyakit tanaman padi dengan metode *Forward Chaining* dan *Certainty Factor* yang diharapkan akan membantu masyarakat luas terutama para petani yang tanaman padinya sedang terserang penyakit agar tahu cara membasmi penyakit pada padi tersebut sehingga tidak mengalami gagal panen.

1.2. Rumusan Masalah

Perumusan masalah memang merupakan salah satu bagian yang sangat penting di dalam sebuah pembuatan sistem, sebab dengan adanya rumusan masalah akan memudahkan untuk pembuatan sistem tersebut. Maka perumusan masalah dalam sistem ini, yaitu sebagai berikut :

- a. Bagaimana mendiagnosis penyakit pada padi secara efektif?
- b. Seberapa akurasi *Forward Chaining* dan *Certainty Factor* dalam mendiagnosis penyakit tanaman padi?

1.3. Batasan Masalah

Di dalam pembuatan sistem ini ada pula batasan masalah yang di buat, yaitu sebagai berikut :

- a. Data yang digunakan adalah data dari Dinas Pertanian dan Pangan Kabupaten Kulon Progo.
- b. Sistem yang akan dibuat hanya dapat mengenali penyakit pada tanaman padi.
- c. Proses diagnosis dilakukan dengan menginputkan gejala yang dialami tanaman padi.
- d. Gejala yang digunakan adalah gejala yang dapat dilihat langsung dengan mata.
- e. Penyakit tanaman padi yang dapat didiagnosis sebanyak 8 penyakit dengan 26 gejala.
- f. Proses pada sistem meliputi proses diagnosis dan proses pengelolaan basis pengetahuan.
- g. Metode penalaran yang digunakan adalah metode *Forward Chaining* dan *Certainty Factor* sebagai metode mengitung nilai kepercayaan.
- h. Output dari sistem ini adalah hasil dari proses diagnosis dan bagaimana cara menanganinya.

1.4. Tujuan Penelitian

Sesuai dengan masalah yang dirumuskan dan yang telah di identifikasi maka tujuan dari pembuatan sistem ini adalah :

- a. Membantu mengetahui penyakit yang sedang di alami padi agar para petani tidak mengalami gagal panen dan tahu bagaimana mengatasinya.
- b. Meningkatkan hasil panen petani sehingga panen para petani akan menjadi melimpah dari mengatasi kerugian pada petani.

1.5. Manfaat Penelitian

Dari perancangan pembuatan sistem ini mempunyai manfaat sebagai berikut:

- a. Mempermudah cara mengetahui penyakit yang sedang dialami padi.
- b. Diharapkan dapat meningkatkan panen petani.
- c. Mempermudah petani mengatasi penyakit yang sedang dialami pada padi.
- d. Sebagai referensi masyarakat untuk ikut berpartisipasi dalam pengembangan potensi pertanian di desa-desa.

2. KAJIAN PUSTAKA DAN LANDASAN TEORI

2.1 Kajian Pustaka

Beberapa hasil penelitian yang pernah dilakukan oleh peneliti sebelumnya yang memiliki bidang dan tema yang sama dengan penelitian yang akan dilakukan.

Adapun penelitian sebelumnya yang berjudul “Sistem Pakar Diagnosa Hama Tanaman Padi Menggunakan Metode *Case Based Reasoning* (CBR)”. Pembangunan sistem pakar ini menggunakan metode *Case Based Reasoning* yaitu suatu proses pendiagnosaan yang berdasarkan kejadian dimasa lampau. Sistem Pakar Diagnosa Hama Tanaman Padi Menggunakan Metode *Case Based Reasoning* ini memiliki akurasi 80% [1].

Penelitian lain dengan judul “Sistem Identifikasi Penyakit Tanaman Padi Dengan Menggunakan Metode *Forward Chaining*”. Pada Penelitian ini merancang sistem pakar berbasis web dengan menggunakan metode *forward chaining* dan *certainty factor*. Data yang diproses dalam sistem ini adalah 16 penyakit dan 26 gejala penyakit. Berdasarkan hasil pengujian tingkat kepuasan pengguna, sistem identifikasi penyakit tanaman padi ini masuk dalam interval sangat baik dengan rata-rata persentase 81% [2].

Juga penelitian dengan judul “Analisis Diagnosa Penyakit Tanaman Padi Dengan Metode *Fuzzy Inference System* Dan *Certainty Factor*”. Pada penelitian *Fuzzy Inference System* yang digunakan dalam sistem ini adalah *Fuzzy Tsukamoto* yang digunakan untuk memperoleh nilai *measure of belief (MB)* dan *measure of disbelief (MD)* gejala terhadap suatu penyakit. Hasil yang didapatkan yaitu analisis ini berhasil memberikan diagnosa penyakit yang padi dari gejala-gejala yang menyerang padi tersebut [3].

Penelitian lain yang berjudul “*Expert System For Analysis Of Disease Rice Plants Based On Symptoms Using The Forward Chaining Method Web-Based*”. Dari hasil penelitian, sistem pakar penyakit padi ini mampu menganalisis penyakit tanaman padi berdasarkan gejalanya sebagai alat pengganti ahli. Tetapi untuk hasilnya diagnosis tidak dapat memberikan solusi atau saran yang benar-benar tepat dalam menanganinya [4].

Penelitian lain dengan judul “Penerapan Sistem Pakar Menggunakan Metode *Fuzzy Sugeno* Identifikasi Hama Tanaman Padi”. Penelitian ini menggunakan metode *sugeno*, dimana tahapan metode *fuzzy sugeno* adalah Pembentukan himpunan *Fuzzy* mengaplikasikan fungsi implikasi (aturan). Komposisi aturan, didapat dari kumpulan data hubungan antar aturan. Penegasan (*Defuzzifikasi*), input dari *defuzzifikasi* adalah konstanta atau persamaan linier. Penelitian ini dapat menghasilkan perangkat lunak yang dapat membantu para petani memecahkan permasalahan

dalam mendiagnosis hama tanaman padi, yang didasarkan pada gejala-gejala yang ditemukan di lapangan. serta dapat memberikan solusi yang baik berupa hasil jenis hama yang ditimbulkan, cara pengendaliannya, pengobatan, penanggulangannya.

2.2 Dasar Teori

2.2.1 Kecerdasan Buatan

Kecerdasan buatan (AI) adalah cabang dari bidang komputer dan ilmu informasi. Ini berfokus pada pengembangan sistem perangkat keras dan perangkat lunak yang memecahkan masalah dan menyelesaikan tugas-tugas yang "jika dilakukan oleh manusia" akan dianggap sebagai tampilan kecerdasan. Bidang AI termasuk mempelajari dan mengembangkan mesin seperti robot, pilot otomatis untuk pesawat terbang dan kapal ruang angkasa, dan senjata militer "pintar" [6].

2.2.2 Sistem Berbasis Pengetahuan

Sistem berbasis pengetahuan adalah salah satu anggota keluarga utama kelompok AI. Dengan ketersediaan fasilitas komputasi canggih dan sumber daya lainnya, perhatian kini beralih ke tugas yang semakin menuntut, yang mungkin membutuhkan kecerdasan. Masyarakat dan industri menjadi berorientasi pengetahuan dan bergantung pada kemampuan pengambilan keputusan para ahli yang berbeda. Sistem berbasis pengetahuan dapat bertindak sebagai ahli sesuai permintaan tanpa membuang waktu, kapan saja dan di mana saja. Sistem berbasis pengetahuan dapat menghemat uang dengan memanfaatkan ahli, memungkinkan pengguna untuk berfungsi di tingkat yang lebih tinggi dan mempromosikan konsistensi. Bahkan, sistem berbasis pengetahuan adalah sistem berbasis komputer, yang menggunakan dan menghasilkan pengetahuan dari data, informasi, dan pengetahuan. Sistem ini mampu memahami informasi dalam proses dan dapat mengambil keputusan berdasarkan informasi / pengetahuan yang berada dalam sistem sedangkan sistem komputer tradisional tidak mengetahui atau memahami data / informasi yang mereka proses[7].

2.2.3 Forward Chaining

Algoritma *forward-chaining* adalah satu dari dua metode utama *reasoning* (pemikiran) ketika menggunakan *inference engine* (mesin pengambil keputusan) dan bisa secara logis dideskripsikan sebagai aplikasi pengulangan dari *modus ponens* (satu set aturan inferensi dan argumen yang valid). Lawan dari *forward-chaining* adalah *backward-chaining*.

Forward-chaining mulai bekerja dengan data yang tersedia dan menggunakan aturan-aturan inferensi untuk mendapatkan data yang lain sampai sasaran atau kesimpulan didapatkan. Mesin inferensi yang menggunakan *forward-chaining*

mencari aturan-aturan inferensi sampai menemukan satu dari *antecedent* (dalil hipotesa atau klausa IF - THEN) yang benar. Ketika aturan tersebut ditemukan maka mesin pengambil keputusan dapat membuat kesimpulan, atau konsekuensi (klausa THEN), yang menghasilkan informasi tambahan yang baru dari data yang disediakan. Mesin akan mengulang melalui proses ini sampai sasaran ditemukan.

Forward-chaining adalah contoh konsep umum dari pemikiran yang dikendalikan oleh data (*data-driven*) yaitu, pemikiran yang mana focus perhatiannya dimulai dari data yang diketahui. *Forward-chaining* bisa digunakan didalam agen untuk menghasilkan kesimpulan dari persepsi-persepsi yang datang, seringkali tanpa query yang spesifik .

2.2.4 Certainty Factor

Dalam pembuatan sistem pakar diagnosa penyakit tanaman padi, metode pengambilan kesimpulan yang digunakan adalah *Certainty Factor*. *Certainty Factor* merupakan bagian dari *Certainty Theory*, yang pertama kali diperkenalkan oleh Shorliffe Buchanan dalam pembuatan MYCIN mencatat bahwa dokter sering kali menganalisa informasi yang ada dengan ungkapan seperti misalnya: mungkin, kemungkinan besar, hampir pasti. Untuk mengakomodasi hal ini tim MYCIN menggunakan *certainty factor (CF)* guna menggambarkan tingkat kepercayaan pakar terhadap masalah yang sedang dihadapi. Dalam CF dikenalkan konsep *Measures of Belief (MB)* atau ukuran kepercayaan dan *Measures of Disbelief (MD)* atau ukuran ketidakpercayaan..

Berikut tabel 2.1 aturan nilai-nilai kepercayaan dan tabel 2.2 nilai interpretasi untuk MB dan MD yang diberikan oleh MYCIN.

Tabel 2.1 Aturan Nilai-Nilai Kepercayaan

Kepercayaan	CF
Tidak Pasti	-1,0 sampai -0,79
Hampir Tidak Pasti	-0,8 sampai -0,59
Kemungkinan Tidak	-0,6 sampai -0,39
Mungkin Tidak	-0,4 sampai -0,19
Tidak Tahu	-0,2 sampai 0,2
Mungkin	0,4 sampai 0,59
Kemungkinan Besar	0,6 sampai 0,79
Hampir Pasti	0,8 sampai 0,89
Pasti	0,9 sampai 1,0

Tabel 2.2 Nilai Interpretasi Untuk MB dan MD

Kepercayaan	MB / MD
Tidak tahu	0 – 0,29
Mungkin	0,3 – 0,49
Kemungkinan Besar	0,5 – 0,69
Hampir Pasti	0,7 – 0,89
Pasti	0,9 – 1,0

Certainty Factor menggunakan ukuran kepercayaan (MB) dan ukuran ketidakpercayaan (MD) dalam mengkombinasikan beberapa *evidence* untuk menentukan nilai CF suatu hipotesis. Konsep ini kemudian di formulasikan ke dalam rumus 1 : *Certainty Factor* didefinisikan sebagai persamaan berikut :

$$CF(H, E) = MB(H, E) - MD(H, E) \quad (1)$$

CF (H, E) : *Certainty Factor* dari hipotesis H yang dipengaruhi oleh gejala (*evidence*) E. Besarnya CF berkisar antara -1 sampai 1. Nilai -1 menunjukkan ketidakpercayaan mutlak sedangkan nilai 1 menunjukkan kepercayaan mutlak.

MB (H, E) : ukuran kenaikan kepercayaan (*measure of increased belief*) terhadap hipotesis H yang dipengaruhi oleh gejala E.

MD (H, E) : ukuran kenaikan ketidakpercayaan (*measure of increased disbelief*) terhadap hipotesis H yang dipengaruhi oleh gejala E.

Bentuk dasar rumus *certainty factor*, adalah sebuah aturan JIKA E MAKA H seperti ditunjukkan oleh persamaan 2 berikut:

$$CF(H, e) = CF(E, e) * CF(H, E) \quad (2)$$

Dimana :

CF (H, e) : *certainty factor* hipotesis yang dipengaruhi oleh *evidence* e.

CF (E, e) : *certainty factor* *evidence* E yang dipengaruhi oleh *evidence* e.

CF (H, E) : *certainty factor* hipotesis dengan asumsi *evidence* diketahui dengan pasti, yaitu ketika $CF(E, e) = 1$

Jika semua *evidence* pada *antecedent* diketahui dengan pasti maka persamaannya akan menjadi:

$$CF(E, e) = CF(H, E) \quad (3)$$

Dalam aplikasinya, CF(H,E) merupakan nilai kepastian yang diberikan oleh pakar terhadap suatu aturan, sedangkan CF(E,e) merupakan nilai kepercayaan yang diberikan oleh pengguna terhadap gejala yang dialaminya.

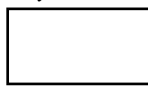
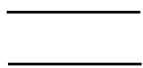
Metode *certainty factor* ini hanya bisa mengolah 2 bobot dalam sekali perhitungan. Untuk bobot yang lebih dari 2 banyaknya, untuk melakukan perhitungan tidak terjadi masalah apabila bobot yang dihitung teracak, artinya tidak ada aturan untuk mengkombinasikan bobotnya, karena untuk kombinasi seperti apapun hasilnya akan tetap sama.

2.2.5 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) adalah suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem, yang penggunaannya sangat membantu untuk memahami

sistem secara logika, tersruktur dan jelas. DFD merupakan alat bantu dalam menggambarkan atau menjelaskan DFD ini sering disebut juga dengan nama *Bubble chart*, *Bubble diagram*, model proses, diagram alur kerja, atau model fungsi.

Tabel 2.1 Simbol DFD

No	Gambar	Keterangan
1		Menggambarkan entitas atau pengguna dari sistem atau aplikasi.
2		Aliran data yang masuk dan keluar dari system
3		Menggambarkan suatu proses atau sistem yang akan dibangun
4		Menggambarkan suatu tabel untuk menyimpan data, dimana nantinya data store ini akan menjadi salah satu tabel dalam perancangan basis data.

2.2.6 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah diagram yang menyediakan cara untuk mendeskripsikan rancangan basis data pada tingkatan logis. ERD didasarkan pada persepsi terhadap dunia nyata yang tersusun atas kumpulan objek-objek dasar yang disebut entitas dan hubungan antar objek. ERD merupakan komplemen dari DFD, perbedaannya adalah DFD berfokus pada proses aliran data (*data flow*) sedangkan ERD berfokus pada data dan hubungan antara data itu sendiri. ERD dapat membantu mengorganisasi data yang digunakan oleh sistem, selain itu ERD juga membantu untuk memastikan kelengkapan (*completeness*), kemampuan adaptasi (*adaptability*) dan stabilitas (*stability*) data [8].

ERD mempunyai tiga komponen utama yaitu entitas, atribut dan relasi. Berikut penjelasan komponen-komponennya, yaitu:

- Entitas (*Entity*)
- Atribut (*Attributes*)
- Relasi (*Relationship*)

3. METODE PENELITIAN

3.1 Metode Penelitian

Metode *waterfall* adalah model klasik yang bersifat sistematis, berurutan dalam membangun software. Nama model ini sebenarnya adalah "*Linear Sequential Model*". Model ini sering disebut juga dengan "*classic life cycle*" atau metode *waterfall*. Model ini termasuk ke dalam model

generic pada rekayasa perangkat lunak dan pertama kali diperkenalkan oleh Winston Royce sekitar tahun 1970 sehingga sering dianggap kuno, tetapi merupakan model yang paling banyak dipakai dalam *Software Engineering* (SE). Model ini melakukan pendekatan secara sistematis dan berurutan. Disebut dengan waterfall karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan. Fase-fase dalam Waterfall Model menurut referensi Pressman [9] :

a. *Communication (Project Initiation & Requirements Gathering)*

Langkah pertama diawali dengan komunikasi dengan customer demi memahami dan mencapai tujuan yang ingin dicapai. Hasil dari komunikasi tersebut adalah untuk inisialisasi proyek, seperti menganalisis permasalahan yang dihadapi dan mengumpulkan data-data yang diperlukan, serta membantu mendefinisikan fitur dan fungsi software. Pengumpulan data-data tambahan bisa juga diambil dari jurnal, artikel, dan internet.

b. *Planning (Estimating, Scheduling, Tracking)*

Tahap berikutnya adalah penulis menetapkan rencana untuk pembuatan sistem yang meliputi mengidentifikasi estimasi tugas-tugas teknis yang akan dilakukan, resiko-resiko yang dapat terjadi, sumber daya yang diperlukan dalam membuat sistem, produk kerja yang ingin dihasilkan, penjadwalan kerja yang akan dilaksanakan, dan tracking proses pengerjaan sistem.

c. *Modeling (Analysis & Design)*

Tahapan ini adalah tahap dimana penulis melakukan perancangan dan permodelan arsitektur sistem yang berfokus pada perancangan struktur data, arsitektur *software*, tampilan *interface*, dan algoritma program. Tujuannya untuk lebih memahami gambaran besar dari apa yang akan dikerjakan.

d. *Construction (Code & Test)*

Tahapan *Construction* ini merupakan proses penulis menerjemahkan bentuk desain menjadi kode atau bentuk/bahasa yang dapat dibaca oleh mesin. Setelah pengkodean selesai, dilakukan pengujian terhadap sistem dan juga kode yang sudah dibuat. Tujuannya adalah untuk menemukan kesalahan yang mungkin terjadi untuk nantinya diperbaiki.

e. *Deployment (Delivery, Support, Feedback)*

Tahapan *Deployment* merupakan tahapan implementasi *software* ke customer, pemeliharaan *software* secara berkala, perbaikan *software*, evaluasi *software*, dan pengembangan *software* berdasarkan umpan balik yang diberikan agar sistem dapat tetap berjalan dan berkembang sesuai dengan fungsinya. .

3.1.1. Pengumpulan Data

a. Wawancara

Pengumpulan data yang dilakukan penulis adalah dengan melakukan wawancara kepada salah satu Dinas Pertanian dan Ketahanan Pangan Provinsi D.I.Yogyakarta guna mendapatkan data-data penyakit tanaman padi, gejala, dan nilai CF yang nantinya diperlukan dalam penelitian ini. Berikut adalah identitas narasumber :

Nama : Sapta Dalyana

Tgl : Bantul 19 Maret 1964

Pekerjaan : PNS

NIP : 196403191987031096

Jabatan : Pengendali Organisme

Pengganggu Tumbuhan(POPT)

Pangkat : Penata tk.I./IIId

Alamat : RT 22/RW 08 Kaligintung, Temon, Kulon Progo

b. Studi Literatur

Studi pustaka yang dilakukan penulis adalah dengan metode pengumpulan data yang bersumber dari buku referensi, jurnal, paper, website dan bacaan – bacaan yang ada kaitannya dengan judul penelitian yang dapat menunjang pemecahan permasalahan yang didapatkan dalam penelitian.

3.1.2 Analisis Data

Keberhasilan suatu aplikasi terletak pada pengetahuan dan bagaimana mengolah pengetahuan tersebut agar dapat ditarik suatu kesimpulan. Pengetahuan yang diperoleh dari hasil kuesioner dan analisa lewat buku dikonversi kedalam sebuah tabel penyakit dan gejala guna mempermudah proses pencarian solusi dan mempermudah penerapannya dalam sistem. Tabel penyakit dan gejala tersebut nantinya digunakan sebagai pola pencocokan informasi yang dimasukan oleh pengguna dan sebagai basis pengetahuan pada sistem yang akan dibangun.

3.1.3 Analisis Kebutuhan Sistem

Kebutuhan fungsional merupakan jenis kebutuhan yang berisi proses apa saja yang nantinya dapat dilakukan oleh sistem, serta berisi informasi apa saja yang harus ada dan dihasilkan oleh sistem.

- Menampilkan menu konsultasi/penelusuran pengguna ;
- Menampilkan menu pilihan gejala;
- Menampilkan hasil dianogsa dan saran;
- Menampilkan menu basis pengetahuan;
- Menampilkan menu basis aturan
- Menampilkan menu bantuanatau program manual;
- Menampilkan menu informasi;

3.1.4 Analisis Perancangan Sistem

Merupakan tahap yang menspesifikasikan bagaimana sistem dapat memenuhi kebutuhan informasi. Untuk dapat memenuhi kebutuhan

pengguna, sistem ini memerlukan beberapa tahap yaitu :

a. Desain Sistem

Tahapan ini adalah tahap dimana penulis melakukan perancangan dan permodelan arsitektur sistem yang berfokus pada Perancangan sistem yang meliputi perancangan diagram konteks, diagram jenjang, DFD level 0, DFD level 1, DFD level 2.

b. Desain Basis Data

Tahapan ini adalah tahap dimana penulis melakukan perancangan dan permodelan arsitektur sistem yang berfokus pada Perancangan basisdata yaitu perancangan Struktur basisdata yang meliputi tabel admin, diagnosa, gejala, relasi, konsultasi, dan tabel informasi.

c. Desain Interface

Desain *interface* merupakan tahap perancangan antar muka antara sistem dengan *user*. Model tampilan yang diterapkan yaitu berbasis website, bagaimana tampilan aplikasi mudah di pahami dan memiliki kesesuaian fungsi.

1. Desain Input

Data yang akan diinputkan dan menjadi data master pada sistem ini meliputi, data diagnosa, data gejala, data relasi, dan data informasi.

2. Desain Proses

Desain proses merupakan tahapan untuk mengolah proses konsultasi atau diagnosis.

3. Desain Output

Output yang akan dihasilkan dari hasil proses konsultasi atau diagnosis.

3.2.4. Pembangunan Sistem

Sistem informasi ini akan dibuat dengan menggunakan bahasa pemrograman berbasis website yaitu :PHP dan untuk database menggunakan MySQL Community. Kemudian software yang digunakan dalam pembangunan.

3.2.5. Implementasi dan Pengujian

Pengujian untuk perancangan sistem informasi ini dilakuakn dengan menggunakan metode blackbox testing, yaitu sebagai berikut :

a. *Blackbox Testing*

Pengujian sistem yang digunakan pada penelitian ini adalah metode pengujian langsung yaitu dengan menggunakan pengujian *BlackBox*. Digunakan untuk menguji fungsi-fungsi khusus dari perangkat lunak yang dirancang. Kebenaran perangkat lunak yang diuji hanya dilihat berdasarkan keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan untuk fungsi yang ada tanpa melihat bagaimana proses untuk

mendapatkan keluaran tersebut. Dari keluaran yang dihasilkan, kemampuan program dalam memenuhi kebutuhan pemakai dapat diukur sekaligus dapat diketahui kesalahan-kesalahannya.*Blacbox*.

4. ANALISIS DAN PERANCANGAN SISTEM

4.1 Analisis

4.1.1 Analisis Kebutuhan Sistem

Dari data yang diperoleh melalui hasil wawancara dan studi literatur, menghasilkan sebuah gagasan penyelesaian masalah dalam penelitian ini. Gagasan tersebut selanjutnya akan diuraikan dan dianalisis untuk mendapatkan beberapa kebutuhan proses dalam pemngembangan sistem ini. Kebutuhan proses adalah kebutuhan yang akan memberi gambaran bagaimana urutan suatu kejadian pada sistem berjalan. Komponen dalam kebutuhan ini akan dijelaskan pada Tabel 4.1.

Tabel 4.2 Tabel Kebutuhan Proses

INPUT	PROSES	OUTPUT
<i>Input login admin, (username & password)</i>	Verifikasi username & password	Masuk ke halaman dashboard admin.
<i>Input data konsultasi</i>	Verifikasi data dengan aturan/relasi	Hasil Konsultasi
Manajemen admin, diagnosa, gejala, relasi, dll (lihat, tambah, ubah, hapus data)	Update data (tampil, tambah, edit, hapus)	Data baru tersimpan Data berubah Data terhapus

4.1.2 Analisis Kebutuhan Perangkat Lunak

Pemilihan perangkat lunak yang digunakan, dipilih berdasarkan keputusan hasil akhir dari *output* sistem yang sudah ditentukan pada tahap planning. Hasil akhir dari pembuatan sistem ini adalah berupa *web* aplikasi. Untuk itu dibutuhkan perangkat lunak yang dapat mencakup kebutuhan sistem seperti pembuatan *DFD*, *ERD*, pembuatan *interface*, *framework*, *web server*, *script editor*, *database server* dan *web preview*. Untuk mengetahui detail perangkat lunak apa saja yang digunakan dalam penelitian ini, dapat dilihat seperti berikut:

Kebutuhan Perangkat Lunak Yang Digunakan Dalam Penelitian

- a. Perancangan *Interface* : Balsamiq Mockups 3
- b. *Script Editor (PHP, SQL, etc)* : Sublime Text 3
- c. *Database Server*: MySQL
- d. *Web Preview* : Firefox, Google Chrome
- e. Bootstrap

4.1.3 Analisis Kebutuhan Perangkat Keras

Agar perangkat lunak yang digunakan dalam pembuatan sistem dapat bekerja dengan optimal,

maka dibutuhkan komponen perangkat keras komputer yang mendukung. Spesifikasi minimal yang digunakan peneliti dalam membuat sistem ini dapat dilihat seperti berikut :

Kebutuhan Perangkat Keras Yang Digunakan Dalam Penelitian :

- Laptop HP Pavilion 15-an010TX
- Sistem Operasi Windows 10
- RAM 8 GB DDR3
- HardDisk 1 TB

4.2 Desain Sistem

Desain sistem adalah tahapan setelah proses analisis data sistem selesai. Tahapan selanjutnya adalah perancangan desain dari sistem dengan memikirkan bagaimana membentuk sistem yang dibuat. Tujuan dibuat desain sistem agar pendefinisian dari kebutuhan-kebutuhan fungsional dapat dipahami secara sistematis. Selain itu juga dapat memberikan sebuah informasi gambaran-gambaran yang jelas dari sistem. Didalam penelitian ini, peneliti melibatkan *user* dari sistem yang dibuat. Agar *user* dapat mengkaji ulang komponen-komponen sistem informasi yang didesain hingga memenuhi standar kebutuhan pemakai. Perancangan desain sistem dilakukan melalui beberapa tahapan.

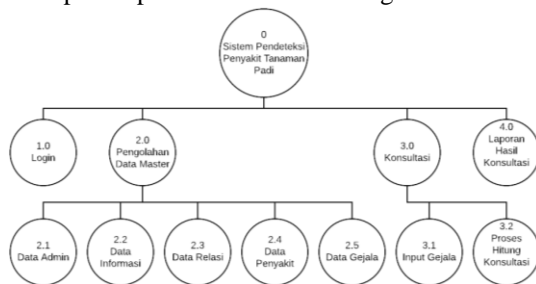
Tahapan-tahapan tersebut meliputi :

- Perancangan *Data Flow Diagram (DFD)*
- Perancangan *Entity Relationship Diagram (ERD)*
- Perancangan Antarmuka (*Interface*)

4.2.1 Perancangan Data Flow Diagram (DFD)

4.2.1.1 Diagram Jenjang

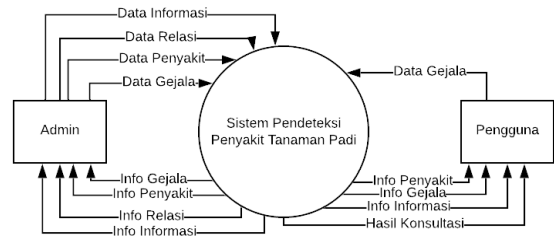
Berikut adalah desain diagram jenjang pembangunan Sistem Berbasis Pengetahuan Pendeteksi Penyakit Tanaman Padi Dengan Metode Forward Chaining Dan Certainty Factor yang akan ditampilkan pada Gambar 4.1. sebagai berikut :



Gambar 4.1 Perancangan Diagram Jenjang

4.2.1.2 Diagram Konteks

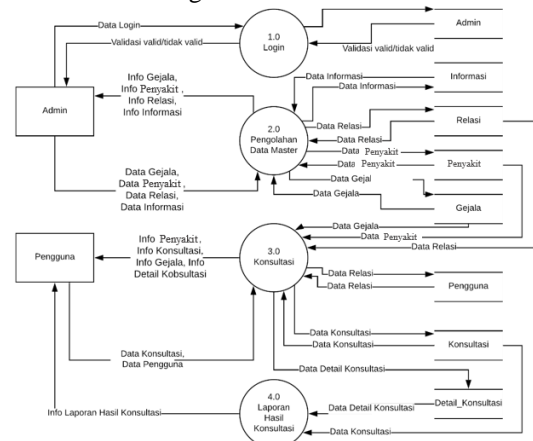
Berikut adalah desain diagram konteks pembangunan Sistem Berbasis Pengetahuan Pendeteksi Penyakit Tanaman Padi Dengan Metode Forward Chaining Dan Certainty Factor yang akan ditampilkan pada Gambar 4.2. sebagai berikut :



Gambar 4.2 Perancangan Diagram Konteks

4.2.1.3 Diagram Level 1

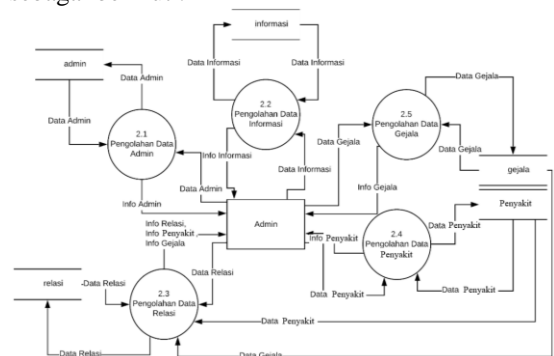
Berikut adalah diagram level 1 pembangunan Sistem Berbasis Pengetahuan Pendeteksi Penyakit Tanaman Padi Dengan Metode Forward Chaining Dan Certainty Factor yang akan ditampilkan pada Gambar 4.3. sebagai berikut :



Gambar 4.3 Data Flow Diagram (DFD) level 1

4.2.1.4 Diagram Level 2 Proses 2 Pengolahan Data Master

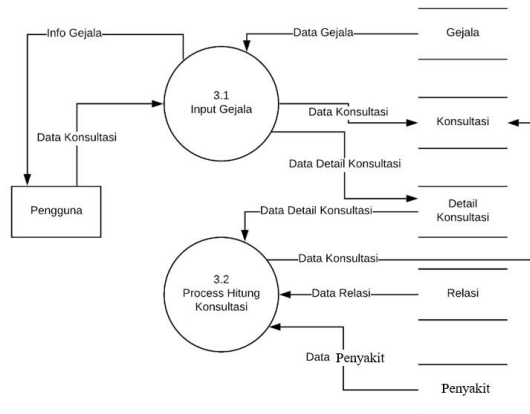
Berikut adalah diagram level 2 proses 1 pengolahan Data Master Sistem Berbasis Pengetahuan Pendeteksi Penyakit Tanaman Padi Dengan Metode Forward Chaining Dan Certainty Factor yang akan ditampilkan pada Gambar 4.4 sebagai berikut :



Gambar 4.4 Diagram Level 2 Proses 2 Pengolahan Data Master

4.2.1.5 Diagram Level 2 Proses 3 Konsultasi

Berikut adalah diagram level 2 proses 3 Konsultasi Sistem Berbasis Pengetahuan Pendeteksi Penyakit Tanaman Padi Dengan Metode Forward Chaining Dan Certainty Factor yang akan ditampilkan pada Gambar 4.5 sebagai berikut :



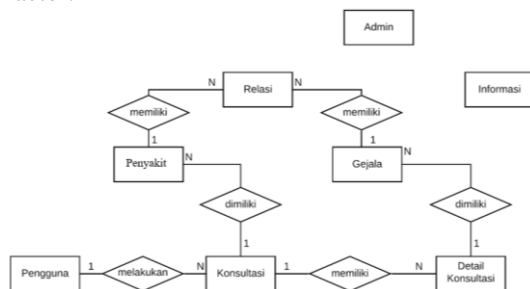
Gambar 4.5 Diagram Level 2 Proses 3 Konsultasi

4.2.1 Perancangan Struktur Basis Data

Pada Sistem Berbasis Pengetahuan Pendeteksi Penyakit Tanaman Padi Dengan Metode Forward Chaining dan Certainty Factor ini, dalam perancangan basis data menggunakan Php My Admin. Php My Admin sendiri termasuk dalam satu paket aplikasi yang disediakan didalam aplikasi Xampp yang digunakan untuk membuat basis data/database. Kebutuhan database utama yang diperlukan dalam pembuatan sistem informasi ini akan dijelaskan pada bagian dibawah ini:

4.2.2.1 Entity Relationship Diagram

ERD merupakan salah satu cara untuk memodelkan suatu kumpulan data yang digambarkan dalam bentuk entitas. Gambar 4.6 di bawah ini merupakan ER-Diagram Sistem Berbasis Pengetahuan Pendeteksi Penyakit Tanaman Padi Dengan Metode Forward Chaining Dan Certainty Factor.



Gambar 4.6 Skema ERD Sistem

4.2.2.2 Struktur Tabel

a. Tabel Admin

Tabel ini berfungsi untuk menyimpan seluruh data admin. Struktur tabel dapat dilihat pada Tabel 4.2.

Tabel 4.3 Tabel Admin

No	Nama Kolom	Tipe Data	Constrain
1	Id	Int(11)	Primary key
2	Username	Varchar(16)	
3	Password	Varchar(16)	

b. Tabel Pengguna

Tabel ini mempunyai fungsi untuk menyimpan seluruh data pengguna. Struktur table pengguna dapat dilihat pada Tabel 4.3

Tabel 4.4 Tabel Pengguna

No	Nama Kolom	Tipe Data	Constrain
1	Id_pengguna	Varchar(16)	Primary key
2	Username	Varchar(16)	
3	Password	Varchar(16)	
4	Nama	Varchar(50)	
5	Alamat	Varchar(50)	

c. Tabel Diagnosa

Tabel ini mempunyai fungsi untuk menyimpan seluruh data diagnosa. Struktur tabel diagnosa dapat dilihat pada Tabel 4.4

Tabel 4.5 Tabel Penyakit

No	Nama Kolom	Tipe Data	Constrain
1	Kode_penyakit	Varchar(16)	Primary key
2	Nama_penyakit	Varchar(255)	
3	Penyebab	Text	
4	Solusi	Text	

d. Tabel Gejala

Tabel ini mempunyai fungsi untuk menyimpan seluruh data gejala. Struktur tabel gejala dapat dilihat pada Tabel 4.5

Tabel 4.6 Tabel Gejala

No	Nama Kolom	Tipe Data	Constrain
1	kode_gejala	Varchar(16)	Primary key
2	Nama_gejala	Varchar(255)	
3	Keterangan	Text	
4	Gambar_ya	Varchar(255)	
5	Gambar_tidak	Varchar(255)	

e. Tabel Relasi

Tabel ini berfungsi untuk menyimpan seluruh data relasi. Struktur tabel relasi dapat dilihat pada Tabel 4.6

Tabel 4.7 Tabel Relasi

No	Nama Kolom	Tipe Data	Constrain
1	Id_relasi	Int(11)	Primary key
2	Kode_diagnosa	Varchar(16)	Foreign key
3	Kode_gejala	Varchar(16)	Foreign key
4	Cf	Double	

f. Tabel Konsultasi

Tabel ini mempunyai fungsi untuk menyimpan seluruh data konsultasi. Struktur table konsultasi dapat dilihat pada Tabel 4.7

Tabel 4.8 Tabel Konsultasi

No	Nama Kolom	Tipe Data	Constrain
1	id_konsul	Int(11)	Primary key
2	Tanggal	Date	
3	Id_pengguna	Varchar(16)	Foreign key
4	Kode_diagnosa	Varchar(16)	Foreign key
5	Nilai	Double	

g. Tabel Detail Konsultasi

Tabel ini mempunyai fungsi untuk menyimpan seluruh data detail konsultasi secara sementara. Struktur table detail konsultasi dapat dilihat pada Tabel 4.8

Tabel 4.9 Tabel Detail Konsultasi

No	Nama Kolom	Tipe Data	Constrain
1	Id_detail	Int(11)	Primary key
2	Id_konsul	Int(11)	Foreign key
3	Kode_gejala	Varchar(16)	Foreign key
4	Jawaban	Varchar(6)	

h. Tabel Detail

Tabel ini mempunyai fungsi untuk menyimpan seluruh data detail konsultasi. Struktur table detail konsultasi dapat dilihat pada Tabel 4.9

Tabel 4.9 Tabel Detail

No	Nama Kolom	Tipe Data	Constrain
1	Id	Int(11)	Primary key
2	Id_konsul	Int(11)	Foreign key
3	Kode_gejala	Varchar(16)	Foreign key
4	Jawaban	Varchar(6)	

i. Tabel Informasi

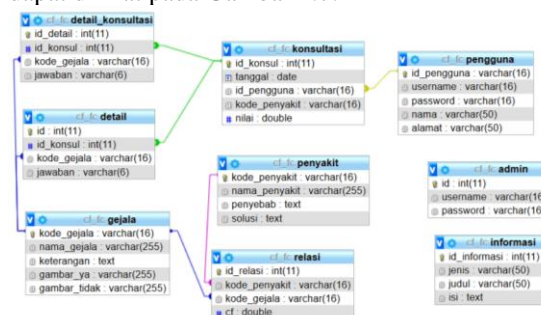
Tabel ini mempunyai fungsi untuk menyimpan seluruh data halaman. Struktur tabel halaman dapat dilihat pada Tabel 4.10

Tabel 4.10 Tabel Informasi

No	Nama Kolom	Tipe Data	Constrain
1	Id_informasi	Int(11)	Primary key
2	Jenis	Varchar(50)	
3	Judul	Varchar(50)	
4	Isi	Text	

4.2.2.3 Relasi Tabel

Tabel penyimpanan yang sudah dirancang akan saling berhubungan karena didalamnya terdapat sebuah proses pengambilan informasi dari masing-masing tabel. Struktur dari relasi tabel dapat dilihat pada Gambar 4.7:



Gambar 4.7 Relasi Tabel

4.3 Representasi Pengetahuan

Metode penelusuran diperlukan untuk menarik simpulan dari data-data yang telah di isikan oleh user. Metode yang digunakan adalah

forward chaining. Metode *forward chaining* adalah metode dimana penelusuran di mulai dari mengambil fakta-fakta terlebih dahulu baru kemudian digunakan untuk menarik simpulan. Dalam hal ini gejala digunakan sebagai fakta, setelah semua data gejala terpenuhi dapat digunakan untuk menarik simpulan mengenai suatu penyakit. Adapun basis pengetahuan yang digunakan adalah penalaran berbasis aturan (*Rule-Based Reasoning*) sebagaimana ditunjukkan pada Tabel 4.11

Tabel 4.11 Basis Pengetahuan

No	Aturan
1	IF Daun menguning AND Daun coklat gelap pada anakan yang rusak AND Pelepah daun membusuk dan berubah menjadi coklat AND akar membusuk THEN Busuk Akar
2	IF Jumlah anakan berkurang AND Pertumbuhan tanaman kerdil AND Helaian daun dan pelepah daun memendek AND Warna daun berubah menjadi kuning kemerah-merahan AND Daun tanaman padi menjadi kasar/bergerigi dan tidak teratur THEN Tungro
3	IF Daun benderanya pendek AND Bagian tanaman yang kasar menguning AND Tanaman yang terinfeksi tumbuh sangat kerdil THEN Kerdil Hampa
4	IF Anakannya banyak sehingga menyerupai rumput AND Daunnya sempit, pendek, kaku dan hijau pucat AND Tanaman mempunyai bercak seperti karat AND Terdapat bercak berbentuk bulat kecil berwarna coklat gelap THEN Kerdil Rumpit
5	IF Daun dan pelepah terdapat bercak berbentuk belah ketupat AND Daun berwarna kelabu dengan pinggir berwarna coklat THEN Blas
6	IF Daun berwarna kelabu dengan pinggir berwarna coklat AND Malai dan buah padi dipenuhi spora AND Dalam satu malai hanya beberapa butir saja yang terserang THEN Noda Api Palsu
7	IF Daun benderanya pendek AND terdapat luka berupa garis bercak kebasahan pada daun AND bercak meluas berwarna hijau keabu-abuan AND seluruh daun menjadi keriput AND layu seperti tersiram air panas AND terdapat bercak berupa cairan berwarna kuning menempel pada permukaan daun THEN Hawar Daun
8	IF daun mudah jatuh AND malai dan buah muda berubah menjadi kecoklatan AND akar membusuk THEN Penyakit Fusarium

5. IMPLEMENTASI SISTEM

5.1 Implementasi

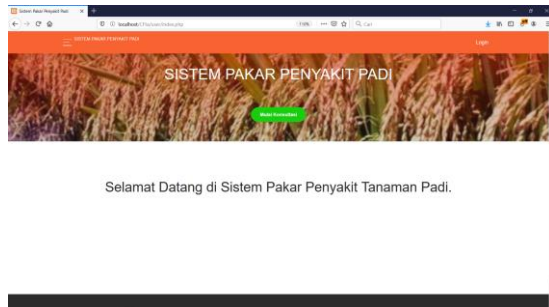
5.1.1 Implementasi Sistem

Implementasi sistem ini adalah tahapan untuk menempatkan hasil perancangan kedalam bentuk operasi. Hal pertama yang dilakukan adalah melakukan pembuatan sistem dengan bantuan perangkat-perangkat lunak dan keras yang sudah disiapkan pada tahapan awal penelitian. Tahap implementasi mengacu pada tampilan sistem (*front-end*) dan fungsi dari sistem (*back-end*).

Adapun beberapa halaman *front-end* yang ditampilkan yang akan disebutkan sebagai berikut:

5.1.1.1 Implementasi Halaman Utama

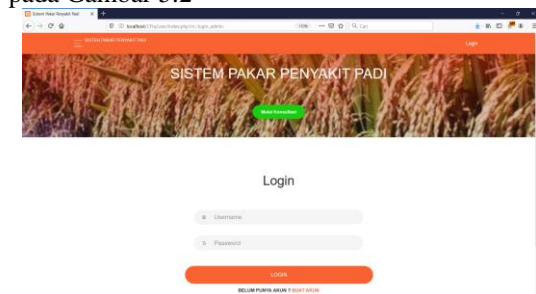
Merupakan halaman pertama yang ditampilkan ketika *User* mengunjungi *website* Sistem Pakar ini. Pada halaman ini sebelum user melakukan login maka user tidak bisa melakukan konsultasi, yang bisa dilakukan user adalah melihat informasi.



Gambar 5.1 Halaman Utama

5.1.1.2 Implementasi Halaman Login

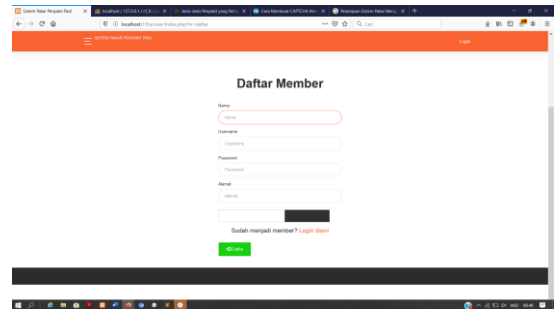
Halaman ini yang digunakan oleh *User* untuk melakukan proses *login* ke sistem untuk dapat mengakses berbagai menu yang terdapat didalam sistem. Pada bagian bawah tombol Login terdapat tautan untuk mendaftar akun atau user baru. Implementasi pada halaman *login* akan ditunjukkan pada Gambar 5.2



Gambar 5.2 Halaman Login

5.1.1.3 Implementasi Halaman Daftar

Halaman ini adalah halaman yang digunakan untuk *input* data *User* ketika melakukan proses daftar. Akan ditampilkan ketika *User* klik tautan daftar pada halaman utama. Pada halaman ini *User* baru akan otomatis memiliki hak akses sebagai pasien. Halaman daftar akan ditunjukkan pada Gambar 5.3



Gambar 5.3 Halaman Daftar

5.1.1.4 Implementasi Halaman Informasi

Halaman ini adalah halaman yang ditampilkan ketika *User* mengakses informasi yang ada dalam sistem ini. Implementasi dari halaman informasi akan ditunjukkan pada Gambar 5.4.



Gambar 5.4 Halaman Informasi

5.1.1.5 Implementasi Halaman Konsultasi

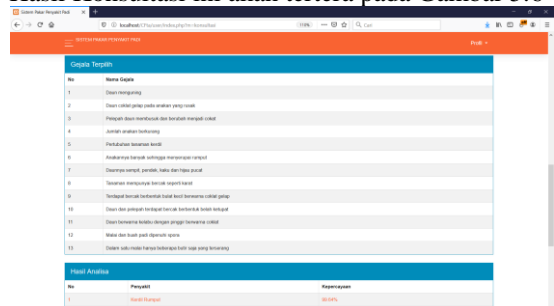
Tampilan dari halaman ketika *User* memilih menu konsultasi pada tombol yang telah tersedia pada halaman utama. Pada halaman ini *User* diminta menjawab pertanyaan pada form konsultasi yang telah tersedia. Implementasi pada halaman ini akan ditunjukkan pada Gambar 5.5



Gambar 5.5 Halaman Konsultasi

5.1.1.6 Implementasi Hasil Konsultasi

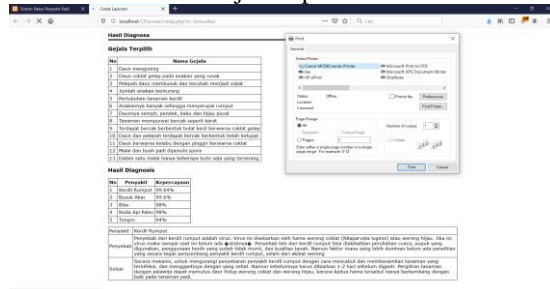
Apabila proses konsultasi selesai, selanjutnya sistem akan memproses jawaban yang dimasukan untuk mendapatkan hasil diagnosis. Implementasi Hasil Konsultasi ini akan tertera pada Gambar 5.6



Gambar 5.6 Halaman Hasil Konsultasi

5.1.1.7 Implementasi Halaman Cetak Hasil Konsultasi

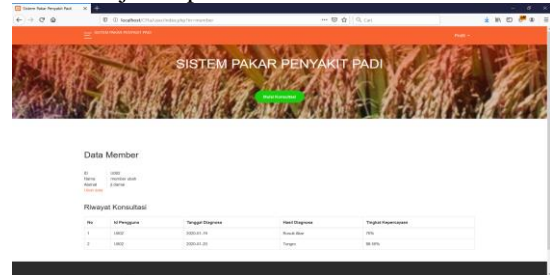
Pada halaman ini user dapat mencetak hasil konsultasinya. Implementasi dari halaman cetak hasil konsultasi ditunjukkan pada Gambar 5.7



Gambar 5.7 Halaman Cetak Konsultasi

5.1.1.8 Implementasi Halaman Member

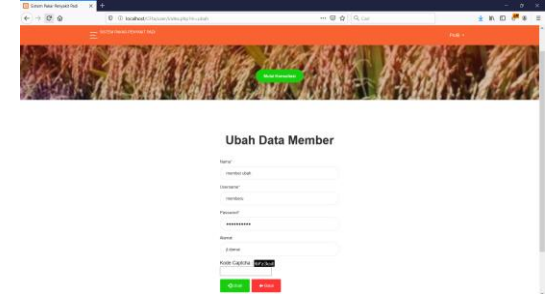
Pada halaman ini member dapat melihat info data member dan riwayat konsultasi yang pernah dilakukan. Implementasi dari halaman member akan ditunjukkan pada Gambar 5.8.



Gambar 5.8 Halaman Member

5.1.1.9 Implementasi Halaman Ubah Data Member

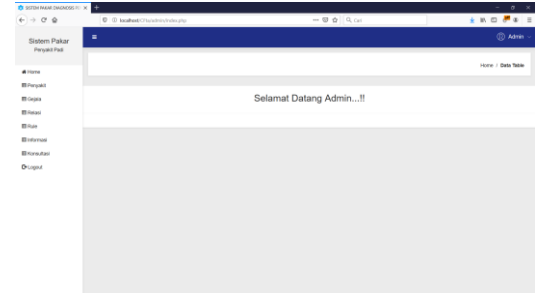
Pada halaman ini member dapat merubah data member. Implementasi dari halaman member akan ditunjukkan pada Gambar 5.9.



Gambar 5.9 Halaman Ubah Data Member

5.1.1.10 Implementasi Halaman Dashboard Admin

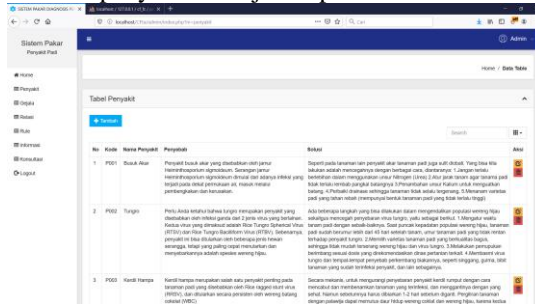
Ketika *User* dengan *level Admin* berhasil melakukan proses *login*, sistem akan mengarahkan ke tampilan *dashboard Admin*. Admin dapat melakukan pengelolaan data-data yang ada pada sistem. Implementasi dari halaman dashboard admin ditunjukkan pada Gambar 5.10



Gambar 5.10 Halaman Dashboard Admin

5.1.1.11 Implementasi Halaman Tabel Penyakit

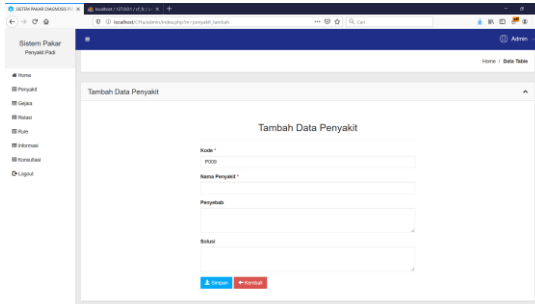
User dengan hak akses admin dapat melakukan pengelolaan data-data pada sistem. Fungsi-fungsi tersebut berjalan melalui tombol fungsi yang tersedia pada kolom aksi pada masing-masing halaman. Berikut contoh implementasi dari halaman penyakit ditunjukkan pada Gambar 5.11



Gambar 5.11 Halaman Tabel Penyakit

5.1.1.12 Implementasi Halaman Tambah Penyakit

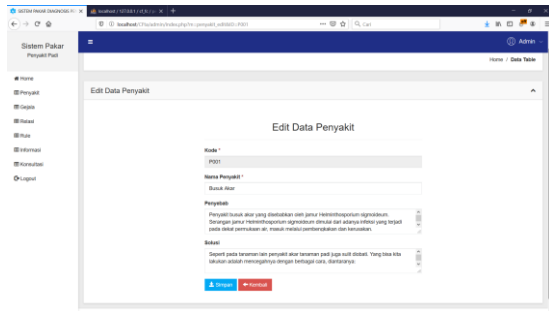
Admin mengisi data penyakit kedalam kolom masukan yang telah disediakan. Implementasi halaman tambah penyakit ditunjukkan pada Gambar 5.12



Gambar 5.12 Halaman Tambah Penyakit

5.1.1.13 Implementasi Halaman Edit Penyakit

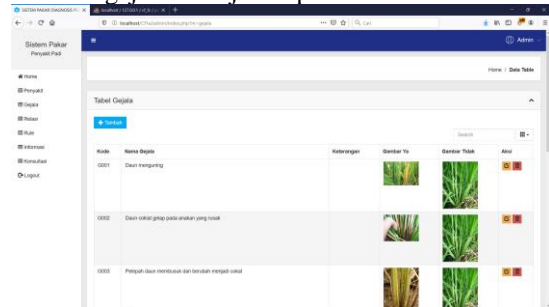
Admin dapat melakukan perubahan data penyakit dengan menekan tombol fungsi *edit* pada kolom aksi. Implementasi halaman *edit* penyakit ditunjukkan pada Gambar 5.13



Gambar 5.13 Halaman Edit Penyakit

5.1.1.14 Impelementasi Halaman Tabel Gejala

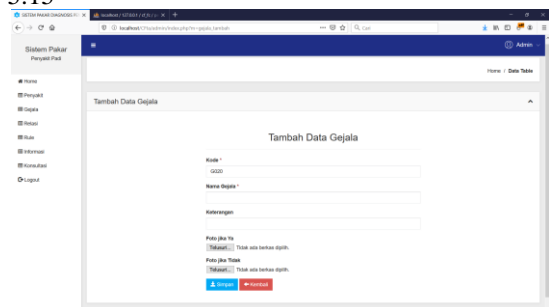
User admin dapat melakukan pengelolaan data-data pada sistem. Pengelolaan yang dapat dilakukan yaitu melihat data, menambah, merubah data, dan menghapus dari database. Fungsi-fungsi tersebut berjalan melalui tombol fungsi yang tersedia pada kolom aksi pada masing-masing halaman. Berikut contoh implementasi dari halaman gejala ditunjukkan pada Gambar 5.14



Gambar 5.14 Halaman Tabel Gejala

5.1.1.15 Impelementasi Halaman Tambah Gejala

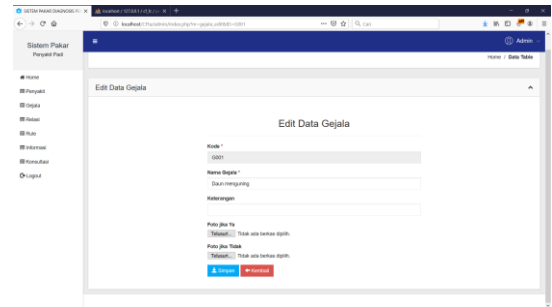
Admin mengisi data gejala kedalam kolom masukan yang telah disediakan. Implementasi halaman tambah gejala ditunjukkan pada Gambar 5.15



Gambar 5.15 Halaman Tambah Gejala

5.1.1.16 Impelementasi Halaman Edit Gejala

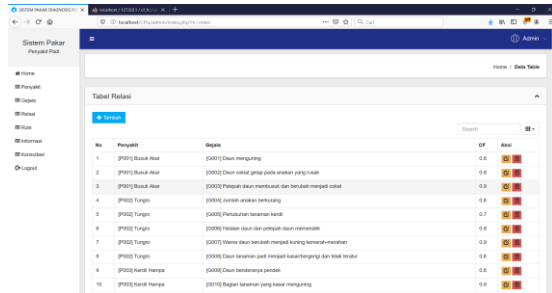
Admin dapat melakukan perubahan data gejala dengan menekan tombol fungsi *edit* pada kolom aksi. Implementasi halaman *edit* gejala ditunjukkan pada Gambar 5.16



Gambar 5.16 Halaman Edit Gejala

5.1.1.17 Impelementasi Halaman Tabel Relasi

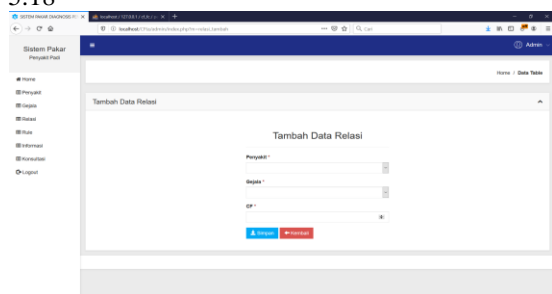
User dengan hak akses admin dapat melakukan pengelolaan data-data pada sistem. Pengelolaan yang dapat dilakukan yaitu melihat data, menambah, merubah data, dan menghapus dari database. Fungsi-fungsi tersebut berjalan melalui tombol fungsi yang tersedia pada kolom aksi pada masing-masing halaman. Berikut contoh implementasi dari halaman relasi ditunjukkan pada Gambar 5.17



Gambar 5.17 Halaman Tabel Relasi

5.1.1.18 Impelementasi Halaman Tambah Relasi

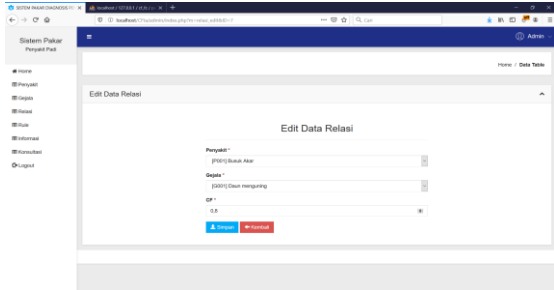
Admin mengisi data relasi kedalam kolom masukan yang telah disediakan. Implementasi halaman tambah relasi ditunjukkan pada Gambar 5.18



Gambar 5.18 Halaman Tambah Relasi

5.1.1.19 Impelementasi Halaman Edit Relasi

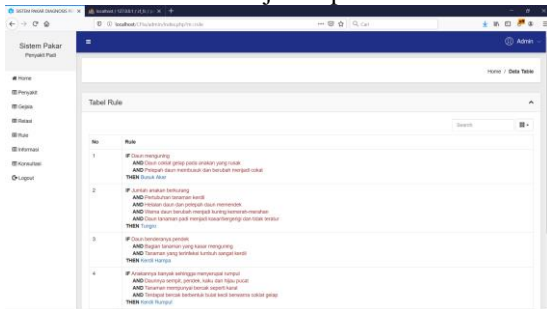
Admin dapat melakukan perubahan data relasi dengan menekan tombol fungsi *edit* pada kolom aksi. Implementasi halaman *edit* relasi ditunjukkan pada Gambar 5.19



Gambar 5.19 Halaman Edit Relasi

5.1.1.20 Implementasi Halaman Table Rule

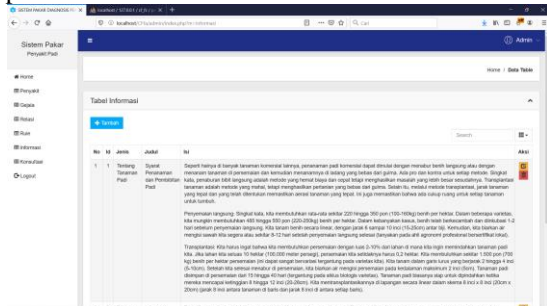
Pada halaman ini ditampilkan rule atau aturan yang dibuat secara otomatis berdasarkan data relasi yang tersimpan di database, jadi tidak bisa ditambah, diubah atau dihapus. Implementasi halaman tabel rule ditunjukkan pada Gambar 5.20



Gambar 5. 20 Halaman Tabel Rule

5.1.1.21 Implementasi Halaman Tabel Informasi

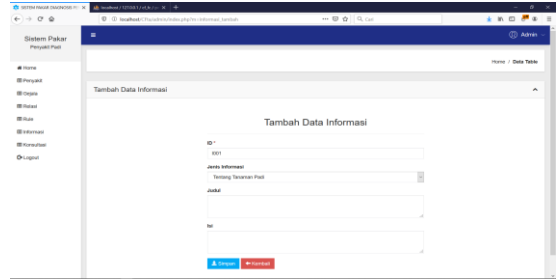
User dengan hak akses admin dapat melakukan pengelolaan data-data pada sistem. Pengelolaan yang dapat dilakukan yaitu melihat data, menambah, merubah data, dan menghapus dari database. Fungsi-fungsi tersebut berjalan melalui tombol fungsi yang tersedia pada kolom aksi pada masing-masing halaman. Berikut contoh implementasi dari halaman informasi ditunjukkan pada Gambar 5.21



Gambar 5. 21 Halaman Tabel Informasi

5.1.1.22 Implementasi Halaman Tambah Informasi

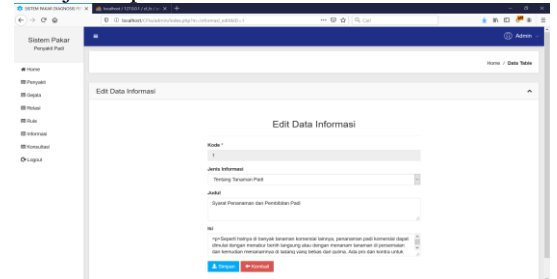
Admin mengisi data informasi kedalam kolom masukan yang telah disediakan. Implementasi halaman tambah informasi ditunjukkan pada Gambar 5.22



Gambar 5. 22 Halaman Tambah Informasi

5.1.1.23 Implementasi Halaman Edit Informasi

Admin dapat melakukan perubahan data informasi dengan menekan tombol fungsi *edit* pada kolom aksi. Implementasi halaman *edit* informasi ditunjukkan pada Gambar 5.23



Gambar 5. 23 Halaman Edit Informasi

6. PENUTUP

6.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan untuk mengembangkan aplikasi Sistem Berbasis Pengetahuan Pendeteksi Penyakit Tanaman Padi dengan Metode Forward Chaining dan Certainty Factor, maka dapat diambil beberapa simpulan sebagai berikut:

- Aplikasi Sistem Berbasis Pengetahuan Pendeteksi Penyakit Tanaman Padi dengan Metode Forward Chaining dan Certainty Factor telah berhasil dibuat. Aplikasi ini berhasil mengimplementasikan metode *Forward Chaining* dan *Certainty Factor* yang digunakan pada proses diagnosis penyakit tanaman padi berdasarkan gejala yang diberikan untuk memberikan tingkat kepercayaan gejala pada suatu penyakit.
- Aplikasi Sistem Berbasis Pengetahuan Pendeteksi Penyakit Tanaman Padi dengan Metode Forward Chaining dan Certainty Factor telah diuji dengan *Blackbox testing*. Hasil dari pengujian *Blackbox* menunjukkan aplikasi ini berjalan sesuai secara fungsionalitas.
- Dari hasil pengujian sistem yang telah dibandingkan dengan hasil keyakinan seorang pakar diperoleh akurasi sistem sebesar 73% dengan 11 kasus berhasil teridentifikasi dengan benar sesuai keyakinan seorang pakar dari total 15 kasus yang diujikan.

6.2 Saran

Mengingat berbagai keterbatasan yang dialami terutama masalah pemikiran dan waktu, berikut adalah saran penulis untuk pengembangan penelitian dimasa yang akan datang :

Saran yang dapat penulis berikan untuk pengembangan aplikasi diagnosis penyakit tanaman padi dengan metode *Forward Chaining* dan *Certainty Factor* adalah dengan memberikan penambahan jumlah data penyakit yang dapat dialami tanaman padi, serta dalam prosesnya perlu ada pembeda bagian yang diserang apakah batang, daun atau akar serta bulir sehingga gejala yang ditampilkan lebih spesifik.

DAFTAR PUSTAKA

- [1] Ahmad, Ardyansyah. (2016). *Sistem Pakar Diagnosa Hama Tanaman Padi Menggunakan Metode Case Based Reasoning (CBR)*. Program Studi Teknik Informatika Fakultas Teknik Universitas Muhammadiyah Jember.
- [2] Arthalia, Ika. (2016). *Sistem Identifikasi Penyakit Tanaman Padi Dengan Menggunakan Metode Forward Chaining*. Jurusan Ilmu Komputer FMIPA Unila.
- [3] Aqmarina, Retma. (2018). *Analisis Diagnosa Penyakit Tanaman Padi Dengan Metode Fuzzy Inference System Dan Certainty Factor*. Prodi Ilmu Komputer FMIPA UNLAM Jl. A. Yani Km 36 Banjarbaru, Kalimantan Selatan.
- [4] Nurdiawan, Odi. (2018). *Penerapan Sistem Pakar Menggunakan Metode Fuzzy Sugeno Identifikasi Hama Tanaman Padi*. Teknologi Informasi STIMIK IKMI Cirebon.
- [5] Yunita, Hilda Dwi. (2019). *Expert System For Analysis Of Disease Rice Plants Based On Symptoms Using The Forward Chaining Method Web-Based*. Sistem Informasi Universitas Mitra Indonesia Lampung
- [6] Moursund, D.G. (2006). *Brief introduction to educational implications of Artificial Intelligence*. Diakses di <http://darkwing.uoregon.edu/~moursund/Books/AIBook/AI.pcdf>
- [7] Sajja, Priti Srinivas.(2010) . *Advanced Knowledge Based Systems : Model, Applications & Research*. diakses di <http://www.tmrfindia.org/eseries/ebookVI.html>.
- [8] Simarmata, Janner. (2006). *Rekayasa Perangkat Lunak*, Andi : Yogyakarta.
- [9] Pressman, R.S.(2015). *Rekayasa Perangkat Lunak: Pendekatan Praktisi Buku I*. Yogyakarta : Andi