

**SISTEM PAKAR TENTANG MENDIAGNOSA PENYAKIT PADA TANAMAN PADI
MENGUNAKAN METODE *FORWARD CHAINING***

SKRIPSI

Diajukan Sebagai Salah Satu Syarat Untuk Menyusun Skripsi

Program Studi Informatika



NURUL AULIA MUSLIMAH

105841103920

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH MAKASSAR**

2024



UNIVERSITAS MUHAMMADIYAH MAKASSAR

FAKULTAS TEKNIK

GEDUNG MENARA IQRA LT. 3

Jl. Sultan Alauddin No. 259 Telp. (0411) 866 972 Fax (0411) 865 588 Makassar 90221

Website: www.unismuh.ac.id, e_mail: unismuh@gmail.com

Website: <http://teknik.unismuh.makassar.ac.id>

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

PENGESAHAN

Skripsi atas nama **Nurul Aulia Muslimah** dengan nomor induk Mahasiswa **105 84 11039 20**, dinyatakan diterima dan disahkan oleh Panitia Ujian Tugas Akhir/Skripsi sesuai dengan Surat Keputusan Dekan Fakultas Teknik Universitas Muhammadiyah Makassar Nomor : 0010/SK-Y/55202/091004/2024, sebagai salah satu syarat guna memperoleh gelar Sarjana Komputer pada Program Studi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar pada hari Jumat tanggal 30 Agustus 2024.

Panitia Ujian :

Makassar,

25 Safar 1446 H

30 Agustus 2024 M

1. Pengawas Umum

a. Rektor Universitas Muhammadiyah Makassar

Dr. Ir. H. Abd. Rakhim Nanda, ST., MT., IPU.

b. Dekan Fakultas Teknik Universitas Hasanuddin

Prof. Dr. Eng. Muhammad Isran Ramli, ST., MT.

2. Penguji

a. Ketua : Dr. Ir. Zahir Zainuddin, M.Sc.

b. Sekretaris : Fahrim Irhamna Rahman S.Kom., MT

3. Anggota

1. Lukman, S.Kom., MT

2. Muhyiddin A. M. Hayat, S.Kom., MT

3. Lukman Anas, S.Kom., MT.

Mengetahui :

Pembimbing I

Rizki Yusliana Bakti ST., MT.

Pembimbing II

Titin Wahyuni, S.Pd., MT



Dr. Ir. H. Nurmayaty, ST., MT., IPM.

NBM : 795 108



UNIVERSITAS MUHAMMADIYAH MAKASSAR
FAKULTAS TEKNIK

GEDUNG MENARA IQRA LT. 3

Jl. Sultan Alauddin No. 259 Telp. (0411) 866 972 Fax (0411) 865 588 Makassar 90221

Website: www.unismuh.ac.id, e_mail: unismuh@gmail.com

Website: <http://teknik.unismuh.makassar.ac.id>

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

HALAMAN PENGESAHAN

Tugas Akhir ini diajukan untuk memenuhi syarat ujian guna memperoleh gelar Sarjana Komputer (S.Kom) Program Studi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar.

Judul Skripsi : **SISTEM PAKAR TENTANG MENDIAGNOSA PENYAKIT PADA TANAMAN PADI MENGGUNAKAN METODE FORWARD CHAINING**

Nama : Nurul Aulia Muslimah

Stambuk : 105 84 11039 20

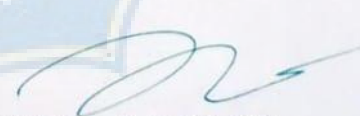
Makassar, 30 Agustus 2024

Telah Diperiksa dan Disetujui
Oleh Dosen Pembimbing;

Pembimbing I

Pembimbing II


Rizki Yusliana Bakti ST.,MT.


Titin Wahyuni, S.Pd.,MT.

Mengetahui,

Ketua Program Studi Informatika


Muhyiddin A.M. Hayat, S.Kom., MT.

NBM 1504 577

ABSTRAK

Penelitian ini berfokus pada perancangan dan implementasi sistem pakar berbasis web dengan menggunakan metode forward chaining untuk diagnosis penyakit pada tanaman padi. Sistem ini dikembangkan untuk membantu petani mengidentifikasi penyakit padi secara cepat dan akurat berdasarkan gejala yang terdeteksi. Metode penelitian yang digunakan mencakup pengumpulan data, desain sistem, implementasi, serta pengujian dengan metode black box untuk memastikan sistem berfungsi sesuai dengan yang diharapkan. Dari hasil perhitungan dan pengujian yang telah dilakukan memperlihatkan gejala dan penyakit yang dihitung mendapatkan hasil yang sangat akurat. Sistem ini juga memberikan saran penanganan yang sesuai berdasarkan hasil diagnosis. Evaluasi terhadap metode forward chaining dalam sistem pakar menunjukkan bahwa metode ini efektif dalam menganalisis gejala yang diinput pengguna untuk mengidentifikasi penyakit. Secara keseluruhan, sistem pakar berbasis web ini dapat menjadi alat yang bermanfaat bagi petani dalam mendeteksi dan menangani penyakit pada tanaman padi secara lebih efisien dan efektif, serta berpotensi meningkatkan produktivitas hasil pertanian.

Kata Kunci: Sistem Pakar, Diagnosis Penyakit Tanaman Padi, *Forward Chaining*, Pengujian *Black Box*, Aplikasi Berbasis Web.

ABSTRACT

This study focuses on the design and implementation of a web-based expert system using the forward chaining method for diagnosing diseases in rice plants. The system was developed to assist farmers in quickly and accurately identifying rice plant diseases based on detected symptoms. The research methodology includes data collection, system design, implementation, and testing using the black box method to ensure the system functions as expected. The results from calculations and testing showed that the symptoms and diseases calculated by the system achieved highly accurate results. The system also provides appropriate treatment recommendations based on the diagnosis. The evaluation of the forward chaining method within the expert system indicates that this method is effective in analyzing user-input symptoms to identify diseases. Overall, the web-based expert system can be a useful tool for farmers in detecting and managing rice plant diseases more efficiently and effectively, with the potential to enhance agricultural productivity.

Keywords: Expert System, Rice Plant Disease Diagnosis, Forward Chaining, Black Box Testing, Web-Based Application

KATA PENGANTAR

Assalamualaikum Warahmatullahi Wabarakatuh

Segala puji bagi Allah SWT yang selalu memberikan kita kenikmatan yang tiada batasnya yaitu nikmat iman, Kesehatan, kesempatan dan masih banyak lagi nikmat lainnya. Sehingga penulis dapat menyelesaikan pengerjaan skripsi ini dengan tepat. Shalawat serta salam tak lupa juga kita haturkan kepada junjungan kita baginda nabi Muhammad SAW yang menjadi suri tauladan seluruh umat yang telah menyebarkan islam dari sembunyi-sembunyi dan terang-terangam. Sehingga sampai detik ini kita bisa merasakan nikmatnya islam.

Dengan rasa terima kasih yang mendalam kepada Allah SWT, karena atas berkat dan Rahmat-Nya lah penulis dapat menghadirkan skripsi ini sebagai hasil perjalanan ilmiah dalam mengeksplorasi bidang sistem pakar dan pertanian dengan judul “ Sistem Pakar Untuk Mendiagnosa Penyakit Pada Tanaman Padi Menggunakan Metode Forward Chaining”

Dalam penulisan skripsi ini penulis ingin mengucapkan rasa terima kasih atas kepada semua pihak yang telah membantu dan memberikan bimbingan, bantuan, saran, serta petunjuk sehingga proposal ini dapat diselesaikan dengan baik dan tepat. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan rasa hormat dan banyak terima kasih kepada :

1. Teristimewa Kedua orang tua penulis, bapak dan almarhumah ibu penulis tercinta, serta ibu sambung, kakak dan adik-adik penulis yang selalu mendoakan dan memberikan dukungan moral serta materi dan semangat dalam setiap langkah perjalanan hidup penulis. Karena tanpa kehadiran mereka, penulis tidak akan mampu menyelesaikan skripsi ini dengan baik
2. Ibu Dr. Ir. Hj. Nurnawaty, ST., MT selaku Dekan Fakultas Teknik Universitas Muhammadiyah Makassar
3. Bapak Muh Syafat S Kuba, S.T., M.T, Selaku Wakil Dekan Fakultas Teknik Universitas Muhammadiyah Makassar.
4. Bapak Muhyiddin A M Hayat, S.Kom., M.T, Selaku Ketua Prodi Informatika Fakultas Teknik Informatika Universitas Muhammadiyah Makassar

5. Ibu Rizki Yusliana Bakti S.T., M.T, Selaku Dosen Pembimbing 1 Skripsi yang telah memberikan bimbingan, arahan, serta masukan yang berharga dalam penyusunan skripsi ini.
6. Ibu Titin Wahyuni S.Pd., M.T, Selaku Dosen Pembimbing 2 Skripsi yang telah memberikan bimbingan, arahan serta masukan yang sangat berharga dalam penyusunan skripsi ini
7. Dosen dan Staff Fakultas Teknik Univeristas Muhammadiyah Makassar
8. Sahabat-sahabat penulis di Kampus Univeristas Muhammadiyah Makassar, terkhusus Putri Nirmanda, Fitra M Natsir, Andri Raodatul Adwiyah, Meisya Putri, Fatimah Azzahra, Jihan Fahira, yang selalu memberikan dukungan, semangat, serta motivasi dalam setiap perjuangan, kebersamaan dan Kerjasama yang terjalin, terima kasih karena sudah membersamai sampai akhir
9. Terakhir untuk diri saya sendiri, Nurul Aulia Muslimah, terima kasih karena sudah kuat dan bertahan sampai sejauh ini

**Syukron Jazakumullah Khairan, *“Billahi fii sabililhaq fastabiqul khairat”*,
Baarakallahufiik.**

Makassar, 23 Februari 2024

Nurul Aulia Muslimah

DAFTAR ISI

ABSTRAK.....	ii
ABSTRACT.....	iii
KATA PENGANTAR.....	iv
DAFTAR ISI.....	vi
DAFTAR GAMBAR.....	vii
DAFTAR TABEL	viii
DAFTAR LAMPIRAN	ix
BAB I PENDAHULUAN	1
A. Latar Belakang masalah.....	1
B. Rumusan Masalah	2
C. Tujuan Penelitian	3
D. Manfaat Penelitian.....	3
E. Ruang Lingkup Penelitian	4
F. Sistematika Penulisan	4
BAB II TINJAUAN PUSTAKA.....	6
A. Landasan Teori.....	6
B. Penelitian Terkait.....	11
C. Kerangka Pikir	14
BAB III METODE PENELITIAN	15
A. Tempat dan Waktu Penelitian.....	15
B. Alat dan Bahan.....	15
C. Perancangan Sistem.....	16
D. Teknik Pengujian Sistem	19
E. Teknik Analisis Data.....	19

BAB IV HASIL DAN PEMBAHASAN.....	20
A. Pengumpulan Data.....	20
B. Data Set.....	20
C. Desain Tabel	28
D. Desain Web.....	31
E. Desain From	36
F. Pengujian Sistem	40
BAB V PENUTUP.....	41
A. Kesimpulan.....	41
B. Saran.....	41
DAFTAR PUSTAKA.....	42
LAMPIRAN	44



DAFTAR GAMBAR

Gambar 1.Struktur sistem pakar	10
Gambar 2.Gambaran kerja <i>forward chaining</i>	11
Gambar 3.Diagram Kerangka pikir	14
Gambar 4.Diagram perancangan sistem	17
Gambar 5. Diagram <i>Flowchart Sistem</i>	18
Gambar 6.Halaman <i>Login</i>	31
Gambar 7.Halaman <i>Home</i>	31
Gambar 8.Halaman Daftar Penyakit.....	32
Gambar 9 Halaman Daftar Penyakit.....	32
Gambar 10.Halaman Daftar Gejala	33
Gambar 11.Halaman Daftar Gejala.....	33
Gambar 12.Halaman Daftar Penanganan.....	34
Gambar 13. Haman Daftar Penanganan.....	34
Gambar 14.Halaman Daiagnosa.....	35
Gambar 15. Halaman Diagnosa.....	35
Gambar 16.From <i>Login</i>	36
Gambar 17. From <i>Home</i>	37
Gambar 18.From Diagnosa	37
Gambar 19.From Daftar Penyakit.....	38
Gambar 20.From Daftar Gejala	39
Gambar 21.From Daftar Penanganan	39

DAFTAR TABEL

Tabel 1. Tabel Gejala dan Nama penyakit Pada Padi	8
Table 2. Tabel Data Set	21
Tabel 3. Tebel User	29
Tabel 4. Tabel Penyakit	30
Tabel 5. Tabel Gejala.....	31
Tabel 6. Tabel Hasil Diagnosa.....	31
Tabel 7. Tabel Fitur Pengujian.....	41



DAFTAR LAMPIRAN

Lampiran 1. Surat Pengantar Penelitian Kepada Ketua LP3M.....	46
Lampiran 2. Surat Pengantar Penelitian Kepada Gubernur Prov Sulsel.....	47
Lampiran 3. Source Code	48
Lampiran 4. Hasil Plagiasi Bab I.....	63
Lampiran 5. Hasil Plagiasi Bab II.....	64
Lampiran 6. Hasil Plagiasi Bab III	65
Lampiran 7. Hasil Plagiasi Bab IV	66
Lampiran 8. Hasil Plagiasi Bab V	67



BAB I

PENDAHULUAN

A. LATAR BELAKANG

Indonesia dikenal sebagai negara agraris dengan sebagian besar penduduknya bekerja sebagai petani. Dengan pertumbuhan penduduk yang tinggi, kebutuhan pangan di Indonesia juga semakin meningkat, sehingga bidang pertanian memerlukan perhatian khusus. Indonesia memiliki berbagai sumber bahan pokok seperti gandum, ubi-ubian, dan padi, namun padi merupakan hasil produksi terbesar yang menjadi makanan utama masyarakat setiap hari. Peran pertanian sangat penting dalam menjaga ketahanan pangan dan kesejahteraan masyarakat Indonesia. Oleh karena itu, peningkatan produktivitas pertanian serta pengembangan teknologi pertanian sangat krusial untuk memastikan ketersediaan pangan yang cukup dan berkelanjutan.

Tanaman padi (*Oryza sativa*) merupakan tanaman sejenis rumput-rumputan yang menjadi makanan pokok. Padi dapat tumbuh dengan baik di wilayah yang tropis yang banyak mengandung air atau padi dapat tumbuh dengan iklim yang baik, panas dengan udara yang lembab dengan temperatur, ketinggian sinar matahari, curah hujan dan angin. Tanaman padi tergolong tumbuhan yang berdaun sempit, dengan ciri khas yang berbeda-beda dari segi bentuk, susunan, atau bagian-bagiannya. Ciri khas daun padi secara umum yaitu adanya sisik dan telinga daun. Kumpulan bunga padi yang keluar dari buku paling atas dinamakan malai. Bulir padi terletak pada cabang pertama dan cabang kedua. Panjang malai tergantung kepada jenis varietas padinya. Ukuran malai dapat dibedakan kepada malai pendek (Amardani, 2022)

Namun dalam proses pertumbuhannya, padi kerap kali diserang oleh hama dan beberapa jenis penyakit yang dapat memperlambat proses pertumbuhannya. Jika tidak ditangani dengan tepat, serangan hama dan penyakit pada tanaman padi dapat mengakibatkan pertumbuhan yang tidak optimal atau bahkan kematian tanaman, yang pada akhirnya berdampak pada hasil panen. Diagnosa jenis hama dan penyakit dapat dilakukan dengan mengamati gejala serta karakteristik yang muncul pada tanaman padi. Proses ini harus dilakukan dengan cepat dan juga akurat karena hama dan penyakit

dapat menyebar dengan cepat dilahan pertanian. Diagnosa seringkali sulit dilakukan karena gejala penyakit pada tanaman padi memiliki kemiripan satu sama lain, sehingga memerlukan keahlian, pengetahuan dan pengalaman khusus dari ahli pertanian. (Zuraida, Kusbianto, and Pahlevi 2023)

Penerapan yang tepat dari masalah tersebut adalah sistem pakar dalam bidang pertanian, di mana pengetahuan seorang ahli digunakan oleh sistem untuk menjawab pertanyaan para petani yang dapat menyelesaikan masalah yang biasanya membutuhkan bantuan seorang ahli tanpa harus bertemu langsung dengan ahli tersebut. Prosesnya melibatkan *user* yang memasukkan gejala penyakit ke dalam sistem pakar, kemudian gejala tersebut dikirimkan ke *knowledge-base* dan diproses oleh mesin inferensi. Sistem kemudian memberikan jawaban berdasarkan pengetahuan yang sebelumnya telah dimasukkan ke dalamnya. peneliti mengembangkan sistem pakar yang mampu menganalisis gejala penyakit pada tanaman padi dan menentukan solusi yang tepat untuk mengatasinya menggunakan metode *Forward Chaining*. Proses pelacakan metode *Forward Chaining* dimulai dari data yang tersedia, kemudian mencocokkan data tersebut dengan bagian IF dari aturan IF-THEN. Jika data sesuai dengan bagian IF tersebut, maka aturan akan dieksekusi untuk memperoleh suatu kesimpulan yang akurat (Darussalam and Yulianton 2023).

Berdasarkan uraian latar belakang tersebut maka peneliti mengangkat judul “**SISTEM PAKAR TENTANG MENDIAGNOSA PENYAKIT PADA TANAMAN PADI MENGGUNAKAN METODE FORWARD CHAINING**”

B. RUMUSAN MASALAH

Berdasarkan latar belakang penelitian yang sudah dijelaskan di atas, peneliti dapat menjabarkan masalah sebagai berikut:

1. Bagaimana merancang dan mengimplementasikan web berbasis metode *forward chaining* yang mampu mendiagnosa penyakit pada tanaman padi
2. Bagaimana tingkat akurasi dari metode *forward chaining* dalam mendiagnosa penyakit pada tanaman padi

C. TUJUAN PENELITIAN

Berdasarkan informasi latar belakang dan perumusan masalah penelitian diatas, peneliti dapat menjabarkan tujuan penelitian sebagai berikut:

1. Merancang dan mengimplementasikan sistem pakar yang efektif untuk mendiagnosa penyakit pada tanaman padi menggunakan metode *Forward Chaining*.
2. Mendiagnosa penyakit pada tanaman padi, beserta gejala yang ditimbulkan

D. MANFAAT PENELITIAN

1. Manfaat bagi peneliti:
 - a) Penelitian ini dapat memberikan pengalaman dalam pengembangan sistem pakar, khususnya dalam konteks pertanian. Peneliti akan memperdalam pemahaman tentang metodologi penelitian, pengolahan data dan implementasi teknologi informasi dalam Solusi penelitian
 - b) Hasil penelitian ini dapat dipublikasikan dalam jurnal ilmiah, konferensi atau seminar yang akan meningkatkan reputasi akademis peneliti dan memberikan pengakuan terhadap kontribusi mereka dalam pengembangan ilmu pengetahuan dan teknologi
2. Manfaat bagi dosen:
 - a) Dosen dapat menggunakan hasil penelitian ini untuk memperkaya materi dalam mata kuliah yang mereka ajarkan, khususnya dalam bidang sistem pakar, kecerdasan buatan, dan pertanian berbasis teknologi.
 - b) Dosen dapat memberikan bimbingan dan arahan kepada mahasiswa yang tertarik untuk melakukan penelitian lebih lanjut dalam bidang sistem pakar atau pertanian berbasis teknologi.
3. Manfaat bagi masyarakat:
 - a) Sistem pakar yang dikembangkan dapat membantu petani dalam mendiagnosa penyakit pada tanaman padi dengan lebih cepat dan akurat.

- b) Dengan bantuan sistem pakar, petani dapat mengambil langkah-langkah yang tepat dalam penanganan penyakit tanaman padi, yang pada akhirnya akan meningkatkan kesejahteraan mereka melalui peningkatan produksi dan pendapatan.
- c) Dengan meningkatnya produktivitas pertanian, hasilnya adalah meningkatnya ketersediaan pangan, yang dapat membantu mengatasi masalah kelaparan dan memperkuat ketahanan pangan di tingkat lokal, nasional, dan global.

E. RUANG LINGKUP PENELITIAN

1. Mendiagnosa penyakit tanaman padi, beserta gejala yang ditimbulkan.
2. Dalam penelitian ini peneliti menggunakan sistem pakar dengan metode *forward chaining*
3. Data yang diambil melalui sesi wawancara dengan pakarnya melalui observasi di lapangan

F. SISTEMATIKA PENULISAN

BAB I PENDAHULUAN

Pada bab ini menjelaskan tentang bagaimana pentingnya identifikasi penyakit pada tanaman padi dan bagaimana peran sistem pakar dalam memfasilitasi proses tersebut. Pada bab ini juga dijelaskan masalah apa saja yang akan dipecahkan serta tujuan penelitian yang akan menjadi pedoman bagi peneliti. Dalam bab ini juga menjelaskan apa saja manfaat ataupun kontribusi yang diberikan peneliti kepada Masyarakat .

BAB II TINJAUAN PUSTAKA

pada bab ini menjelaskan tentang penyakit apa saja yang menyerang tanaman padi beserta gejala-gejalanya, konsep dasar dari sistem pakar, dan metode *forward chaining*. serta tinjauan ringkas terhadap penelitian-penelitian terdahulu yang berkaitan dalam bidang sistem pakar mendiagnosa penyakit pada tanaman padi menggunakan metode *forward chaining*.

BAB III METODE PENELITIAN

Pada bab ini menjelaskan waktu penelitian dan alat yang akan digunakan peneliti dalam melakukan penelitian. Serta menjelaskan pendekatan yang akan digunakan dalam mengembangkan sistem pakar mendiagnosa penyakit pada tanaman padi menggunakan metode *forward chaining*. Serta bagaimana proses pengumpulan data mengenai berbagai jenis penyakit dan gejala apa saja yang dialami.

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini menjelaskan data-data apa saja yang telah terkumpul dari penelitian tentang penyakit apa saja yang terjadi pada tumbuhan padi serta gejalanya. Dan menjelaskan bagaimana hasil kinerja dari *forwad chaining* dalam mendiagnosa peyakit pada tanaman padi. Dan Memberikan gambaran yang akurat yang digunakan dalam penelitian.

BAB V KESIMPULAN

Pada bab ini menjelaskan secara singkat kesimpulan dari penelitian yang telah dilakukan serta menjelaskan pula manfaat praktis dari sistem bagi para petani dalam mengurangi dampak serangan penyakit yang terjadi pada tanaman padi. Selain itu, pada bab ini akan menjelaskan juga kontribusi penelitian dlam pengembangan solusi teknologi yang inovatif dan bermanfaat bagi pertanian berkelanjutan

BAB II

TINJAUAN PUSTAKA

A. Landasan Teori

1. Tanaman Padi

Menurut (Jeraman, Faizah, and Koryanto 2023) tanaman padi (*Oryza Sativa*) merupakan salah satu jenis tanaman budi daya atau eksploitasi yang terutama untuk kesinambungan

hayati insan, dimana tanaman ini bisa menghasilkan beras yang merupakan makanan utama bagi sebagian akbar insan tidak terkecuali di Indonesia, sehingga tanaman padi sebagai salah satu jenis tanaman yg dibudidayakan hampir sebagian akbar wilayah di Indonesia. Aktivitas budidaya tanaman padi sebagai mata pencaharian bagi sebagian masyarakat, sehingga taraf kebesaran kuantitas padi yang dipenuaianpun menjadi amat berharga lantaran mereka mengharapkan hidup mereka dari sekian banyaknya padi yang bisa mereka panen. Akan tetapi terdapat banyak hal yang tak jarang terjadi sebagai akibatnya mereka mengalami kerugian, dan kerugian bagi para petani yang ditimbulkan sebab dengan adanya kelainan tanaman padi yang dipengaruhi karena aspek alam, manusia serta mikro organisme pengganggu berupa virus, bakteri, dan jamur yang terlambat untuk ditelaah karena masih sulitnya para petani untuk menentukan penyakit yang menyerang tanaman padi mereka sebagai akibatnya menyebabkan gagal panen. Tampaknya urusan ini merupakan salah satu pemicu terhalangnya produk beras dari hasil kerja para petani. Dalam pemeliharaan tanaman padi terdapat beberapa jenis penyakit yang menyerang pertumbuhannya. Sehingga menyebabkan kerugian. Penyakit pada tanaman padi diakibatkan karena adanya berbagai factor sepeerti factor lingkungan, manusia, serrta hama tanaman khususnya pathogen atau mikro organisme pengganggu (virus, bakteri dan jamur) yang terlambat untuk di diagnosa. Sehingga mengakibatkan gagal panen. Serta masih sulitnya para petani mengetahui penyebab terjadinya penyakit pada tanaman padi.

Adapaun morfologi dari tanaman padi yaitu (Gergely 2024) :

- a. Akar merupakan bagian tanaman yang berfungsi menyerap air dan zat makanan dari dalam tanah, kemudian di salurkan ke bagian atas tanaman. Bagian akar tanaman padi dibagi menjadi 4 yaitu:
- 1). Radikula adalah akar yang tumbuh pada saat perkecambahan. Benih yang sedang berkecambah, timbul calon akar mengalami pertumbuhan ke arah bawah sehingga terbentuk akar tunggang, sedangkan calon batang akan tumbuh ke atas sehingga terbentuk batang dan daun.
 - 2). Akar serabut tumbuh setelah 5-6 hari akar tunggang tumbuh
 - 3). Akar rambut merupakan akar yang keluar dari akar tunggang dan akar serabut.
 - 4). Akar tajuk merupakan akar yang tumbuh dari ruas batang terendah.
- b. Batang padi mempunyai batang yang beruas-ruas, tinggi batang tanaman padi ada pada kisaran 107-115 cm. Ruas-ruas itu merupakan bubung kosong. Pada kedua ujung bubung kosong itu bubungnya di tutup oleh buku. Panjangnya ruas tidak sama. Ruas yang kedua, ruas yang ketiga, dan seterusnya adalah lebih Panjang daripada ruas yang didahulunya. Pada buku bagian bawah dari ruas tumbuh daun pelepah yang membalut ruas sampai buku bagian bawah dari ruas tumbuh.
- c. Daun padi umumnya bersisik, dan terbagi atas helaian daun, pelepah daun, dan lidah daun. Daun padi berwarna hijau posisi daun tegak dan daun benderanya tegak.
- d. Bunga padi adalah bunga telanjang artinya mempunyai perhiasan bunga. Berkelamin dua jenis dengan bakal buah yang diatas. Jumlah benang sari ada 6 buah, tangkal sarinya pendek dan tipis, kepala sari tipis, kepala sari besar serta mempunyai 2 kandung serbuk
- e. Buah padi seringkali di sebut biji, akan tetapi buah pada tanaman padi yang sebenarnya disebut dengan gabah/bulir. Buah padi tertutup ada lemma dan pelepah dan mengalami penyerbukan

Table 1. gejala penyakit pada tanaman dan penyebabnya (Sumber: Kantor Dinas Pertanian)

penyakit	Gejala	Penyebab
Hawar daun (Blast)	Terdapat bercak-bercak berbentuk seperti belah ketupat dengan ujung rancing	Jamur <i>pyricularia</i>
Busuk Pelepah	Lesi berbentuk oval pada pelepah daun dengan tepi coklat dan pusat abu-abu	jamur <i>Rhizoctonia solani</i>
Kerdil Rumput	Tanaman yang terinfeksi berat menunjukkan penampilan seperti rumput dengan banyak tunas pendek atau kaku	Wereng Coklat
Kerdil Hampa	Daun muda yang terinfeksi menguning dan berubah menjadi oranye kemerahan	Wereng Coklat
Tungro	Daun yang lebih muda menunjukkan perubahan warna yang lebih cepat dan lebih intens dibandingkan daun yang lebih tua	Virus Tungro yang ditularkan oleh tungau (<i>Nilaparvata lugens</i>) dan wereng coklat

2. Sistem Pakar

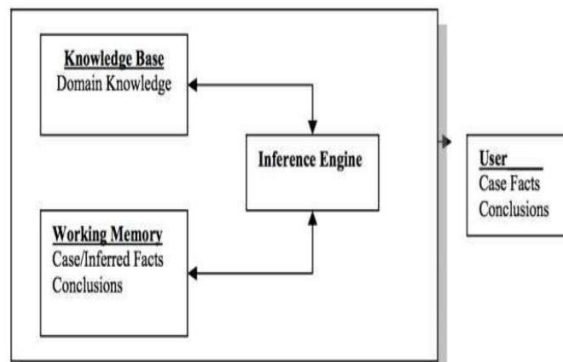
Menurut (Alim, Lestari, and Rusliyawati 2020) Sistem pakar adalah program komputer yang mensimulasi penilaian dan perilaku manusia atau organisasi yang memiliki pengetahuan dan pengalaman ahli dalam bidang tertentu. Biasanya sistem seperti ini berbasis pengetahuan yang berisi akumulasi pengalaman dan satu set aturan untuk menerapkan

pengetahuan dasar untuk setiap situasi tertentu. Sistem pakar yang canggih dapat ditingkatkan dengan menambah basis pengetahuan atau set aturan. Pemrosesan yang dilakukan oleh sistem pakar merupakan pemrosesan pengetahuan bukan pemrosesan data pada sistem pakar komputer konvensional. Pengetahuan (knowledge) adalah pemahaman secara praktis maupun teoritis terhadap suatu obyek atau domain tertentu.

Sistem pakar memiliki kemampuan untuk merekomendasikan serangkaian tindakan spesifik bagi pengguna, sehingga mereka dapat menjalankan sistem koreksi dengan tepat dan akurat. Sistem ini memanfaatkan kemampuan proses penalaran yang canggih untuk mencapai kesimpulan berdasarkan analisis data dan fakta yang ada. Selain itu, struktur sistem pakar meliputi beberapa komponen utama, salah satunya adalah antarmuka pengguna. Antarmuka pengguna ini berfungsi sebagai mekanisme komunikasi yang memungkinkan interaksi langsung antara sistem dan pengguna, sehingga pengguna dapat dengan mudah memberikan masukan dan menerima rekomendasi dari sistem.

Sebuah sistem pakar terdiri dari beberapa komponen, yaitu sebagai berikut:

- a. Basis pengetahuan (knowledge Base) adalah basis pengetahuan inti program dari sistem pakar karena basis pengetahuan ini merupakan representasi dari seorang pakar
- b. Basis Data adalah komponen dari sistem pakar yang berisi data-data pengetahuan yang diperlukan
- c. Mesin inferensi adalah bagian yang berisi mekanisme fungsi berfikir dan pola penalaran yang digunakan oleh sistem pakar, mirip dengan cara seorang pakar berpikir.
- d. Antarmuka pengguna adalah bagian yang menghubungkan antara pengguna dan program sistem pakar, memfasilitasi interaksi dan pemahaman antara keduanya.



Gambar 1. Struktur Sistem Pakar (Normah et al. 2022)

Sistem pakar memiliki keunggulan antara lain sebagai berikut (Cantika, Kanedi, and Jumadi 2023) :

- a. Menghimpun data dalam jumlah yang sangat besar
- b. Menyimpan data tersebut untuk jangka waktu yang panjang dalam suatu bentuk tertentu
- c. Mengajarkan perhitungan secara cepat dan tepat tanpa jemu mencari kembali data yang tersimpan dengan kecepatan tinggi

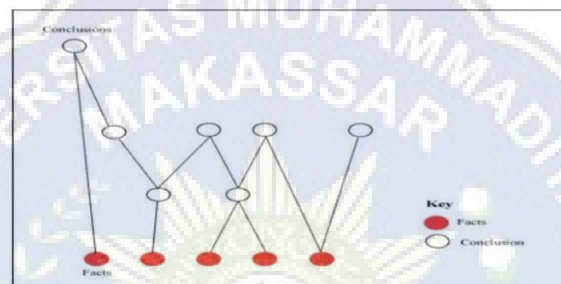
3. *Forward Chaining*

Forward chaining merupakan suatu strategi pengambilan keputusan yang dimulai dari sebelah kiri (IF dahulu). Dengan kata lain, penalaran dimulai dari fakta terlebih dahulu untuk menguji kebenaran hipotesis. *forward chaining* bisa dikatakan sebagai strategi inference yang bermula dari sejumlah fakta yang diketahui. Pencarian dilakukan dengan menggunakan aturan (rule) yang premisnya cocok dengan fakta yang diketahui tersebut untuk memperoleh fakta baru dan melanjutkan proses hingga goal tercapai atau hingga sudah tidak ada rule lagi yang premisnya cocok dengan fakta yang diketahui maupun fakta yang diperoleh. Inferensi *forward chaining* cocok untuk memecahkan masalah control dan prediksi (Nasution 2022)

Adapun kelebihan yang dimiliki oleh *forward chaining* yaitu:

- a. Penyelesaian Langsung dan Efisien: Cepat menyelesaikan masalah dengan fakta awal dan cocok untuk pemrosesan data berurutan.
- b. Sederhana dan Intuitif: Mudah dimengerti, diimplementasikan, dan transparan.
- c. Penanganan Fakta Dinamis: Efektif untuk fakta yang terus berubah dan fleksibel dalam berbagai bidang aplikasi .

Forward chaining beroperasi dengan memulai dari sekumpulan fakta awal yang sudah diketahui, kemudian memeriksa setiap aturan dalam basis pengetahuan untuk menentukan apakah syarat-syarat aturan tersebut sesuai dengan fakta yang ada. Jika syarat terpenuhi, aturan tersebut diterapkan dan hasilnya ditambahkan ke basis fakta. Proses ini diulangi, dengan memperbarui basis fakta dan memeriksa aturan, hingga mencapai tujuan atau tidak ada lagi aturan yang dapat diterapkan. Jika tujuan tercapai, sistem memberikan kesimpulan: jika tidak, sistem menyatakan bahwa Solusi tidak ditemukan berdasarkan fakta yang ada.



Gambar 2. Gambaran kerja *forward chaining* (Normah et al. 2022)

B. Penelitian Terkait

1. (Darussalam and Yulianton 2023) "Sistem Pakar Diagnosa Penyakit Tanaman Padi Menggunakan Metode Forward Chaining"

Penelitian ini bertujuan untuk menciptakan sistem pakar yang bisa melakukan diagnosa dari gejala yang input oleh user sendiri menjadi sebuah informasi berupa nama penyakit padi dan penjelasan tentang penyakit tersebut baik berupa gejala yang menjadi penyebab dan juga solusi mengenai penyakit tersebut yang dirasa sesuai untuk menggunakan metode Forward Chaining berbasis Android. Berdasarkan confusion matrix maka dapat diambil kesimpulan metode forward chaining dapat digunakan untuk Sistem pakar diagnosa penyakit tanaman padi dengan nilai akurasi cukup baik yaitu sebesar 84% dengan nilai presisi 84% dan recall sebesar 100%.

2. (Zuraida, Kusbianto, and Pahlevi 2023) “ Sistem Pakar Diagnosis Penyakit dan Hama pada Tanaman Padi dengan Metode Forward Chaining”

Penelitian ini bertujuan untuk mengembangkan sistem pakar berbasis website untuk melakukan diagnosis hama dan penyakit tanaman padi serta memberikan rekomendasi solusi berdasarkan informasi gejala yang muncul. Sistem pakar dibangun menggunakan metode forward chaining. Sistem pakar ini diharapkan dapat membantu petani melakukan diagnosis secara mandiri sehingga serangan hama maupun penyakit dapat ditanggulangi dengan lebih cepat dan meminimalisir kemungkinan pertumbuhan tanaman padi yang tidak optimal atau bahkan kegagalan panen. Hasil penelitian menunjukkan bahwa sistem pakar dengan metode Forward Chaining dapat digunakan untuk melakukan diagnosis hama dan penyakit tanaman padi dengan tingkat akurasi 87,4%. Hal ini menunjukkan bahwa sistem pakar yang dikembangkan cukup baik untuk membantu petani melakukan diagnosa hama dan penyakit secara mandiri sehingga penanganan yang tepat dapat segera dilakukan. Pengembangan sistem pakar lebih lanjut dapat berupa implementasi metode pelengkap yang mempertimbangkan keraguan pengguna jika gejala yang dihadapi di lapangan kurang sesuai dengan gejala-gejala yang disediakan pakar.

3. (Nasution 2022) ” Sistem Pakar dalam Mendiagnosis Hama Blas dan Kresek pada Tanaman Padi Menggunakan Metode Forward Chaining”

tujuan dari penelitian yaitu membuat berupa sistem untuk membantu para Petani Padi di Kabupaten Mandailing Natal untuk mendiagnosis penyakit Blas dan Kresek pada tanaman Padi, pada tahapan analisa sistem memiliki tujuan untuk mendapatkan pengetahuan mengenai diagnosis penyakit Blas dan Kresek menggunakan metode Forward Chaining berdasarkan data penyakit Blas dan Kresek yang ditemukan penulis di lokasi persawahan di Kabupaten Mandailing Natal. hasil penelitian ini Aplikasi Sistem Pakar yang dihasilkan berupa aplikasi web base dengan menggunakan bahasa pemograman PHP MYSQL untuk diagnosis hama pada tanaman padi dengan menggunakan metode Forward Chaining.

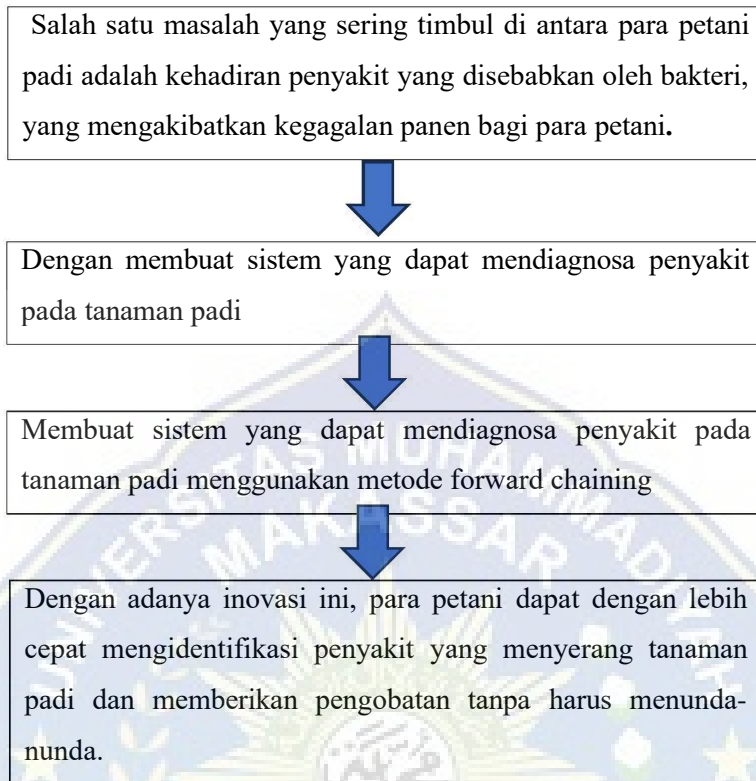
4. (Kuswanto 2020) “ Sistem Pakar menggunakan Metode Forward Chaining untuk Diagnosa Penyakit Tanaman Padi berbasis Android “

Tujuan utama penelitian ini untuk membangun aplikasi sistem pakar untuk membantu

petani menyelesaikan masalah petani dalam mendiagnosa penyakit tanaman padi berbasis android. Data yang digunakan pada penelitian ini terdiri dari 7 penyakit pada tanaman padi yaitu tungro, hawar daun bakteri(HDB), busuk batang, busuk pelepah, blast, kerdil rumput dan Hawar pelepah daun. Metode yang digunakan pada penelitian ini adalah Metode Forward Chaining. Metode yang digunakan untuk membangun aplikasi sistem pakar, proses pelacakan metode ini yaitu dari data, kemudian mencocokkan data tersebut dengan bagian IF dari rules IF-THEN. Jika sesuai dengan rules yang ada, selanjutnya rules akan di eksekusi untuk mendapatkan suatu kesimpulan. Hasil penelitian ini aplikasi sistem pakar dengan menggunakan Forward chaining dapat digunakan untuk diagnosa penyakit tanaman padi dengan beberapa langkah yaitu memilih gejala, melakukan konsultasi, dan menampilkan hasil konsultasi.

5. (L Tobing et al. 2019) “ Sistem Pakar Mendeteksi Penyakit Pada Tanaman Padi Menggunakan Metode Forward Chaining ”
penelitian ini mencoba memberikan sebuah solusi yang baru kepada para petani dengan menggunakan sebuah sistem yang berbasis komputerisasi agar petani dapat segera mencegah penyebaran penyakit atau hama tanaman. Dengan hadirnya sebuah sistem yang dapat dijadikan sebagai alat konsultasi para petani tidak perlu menunggu penyuluhan dari pemerintah tentang cara dalam mengatasi penyakit padi tersebut. Dari perancangan sistem pakar peneliti menarik beberapa kesimpulan, diantaranya adalah Sistem pakar ini sudah dirancang dan dapat digunakan untuk mendiagnosa penyakit pada tanaman padi, Sistem pakar dirancang untuk membantu petani dan penyuluh pertanian dalam mendeteksi penyakit pada tanaman padi. Dari hasil pengujian yang telah dilakukan menggunakan sistem pakar, 15 kasus yang berbeda dilapangan selanjutnya di cross check dengan hasil Analisa pakar dan memiliki kesesuaian sebanyak 93%

C. Kerangka Pikir



Gambar 3. Diagram kerangka pikir

BAB III

METODE PENELITIAN

A. Tempat dan Waktu Penelitian

Menentukan lokasi penelitian memiliki peran yang sangat krusial dalam menilai kevalidan dan ketepatan hasil penelitian. Hal ini menjadi faktor utama yang harus dipertimbangkan secara cermat untuk memastikan keberhasilan dan relevansi dari penelitian yang dilakukan. Adapun lokasi penelitian yang akan dilakukan adalah pada Dinas Perikanan dan pertanian kota makassar Jln Urip Sumaharjo No 08, Maccini, Kec. Makassar, Kota Makassar

B. Alat dan Bahan

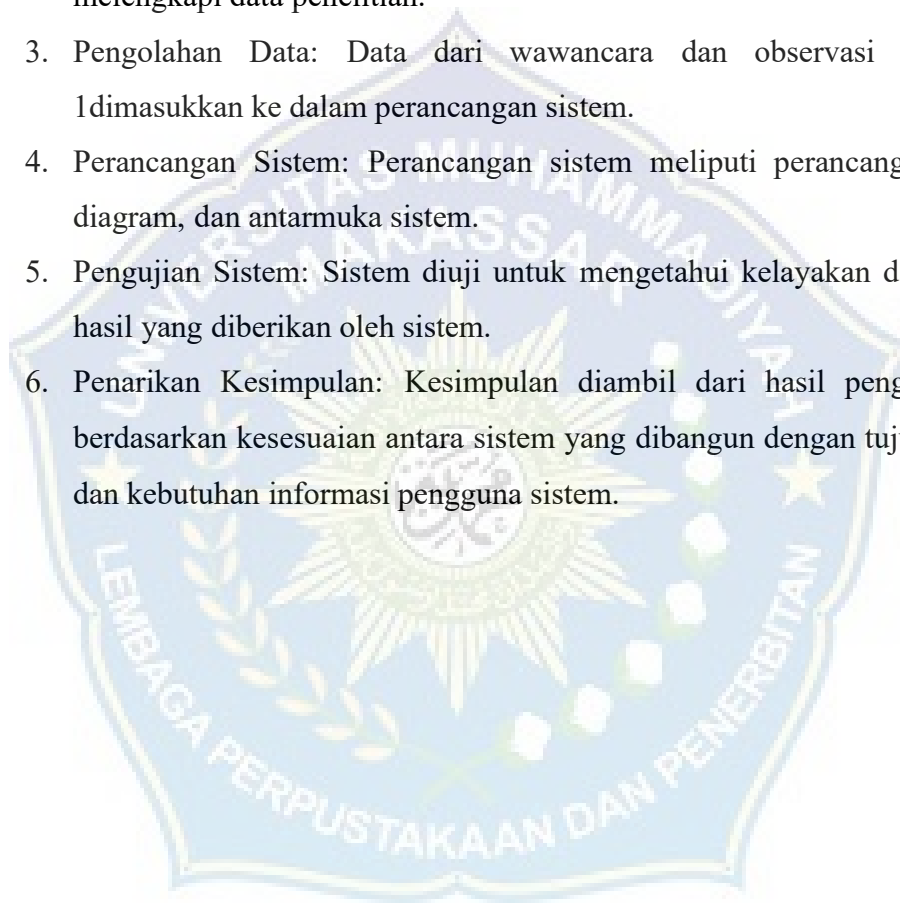
Adapun alat dan bahan yang akan digunakan dalam penelitian ini yaitu sebagai berikut:

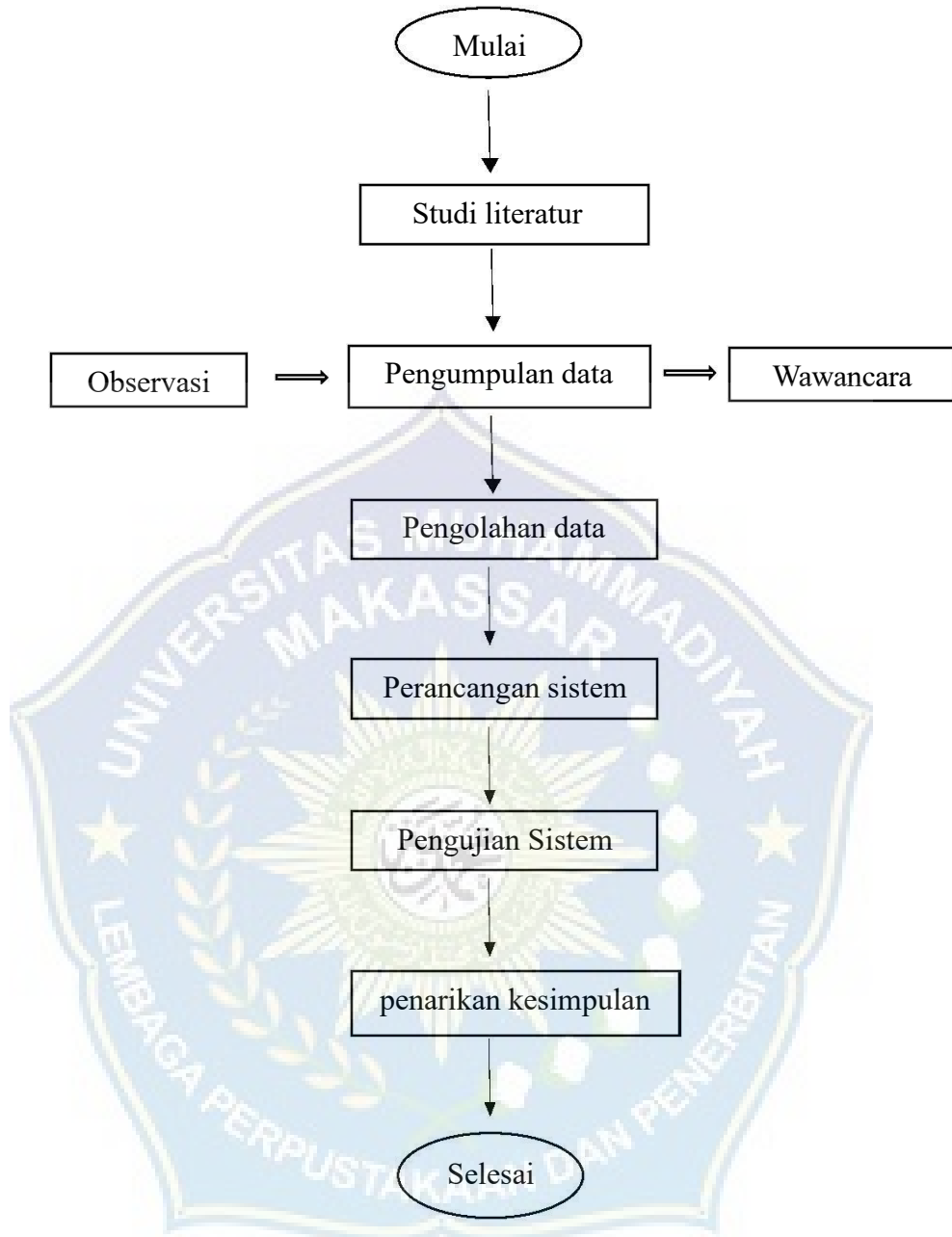
1. Kebutuhan Hardwere
 - a. Leptop Asus Vivobook X14MA-A416MA
 - b. RAM 4.00 GB
2. Kebutuhan Penunjang
 - a. Printer
 - b. Kertas HVS
 - c. Tinta printer
 - d. Buku
 - e. Pulpen

C. Perancangan Sistem

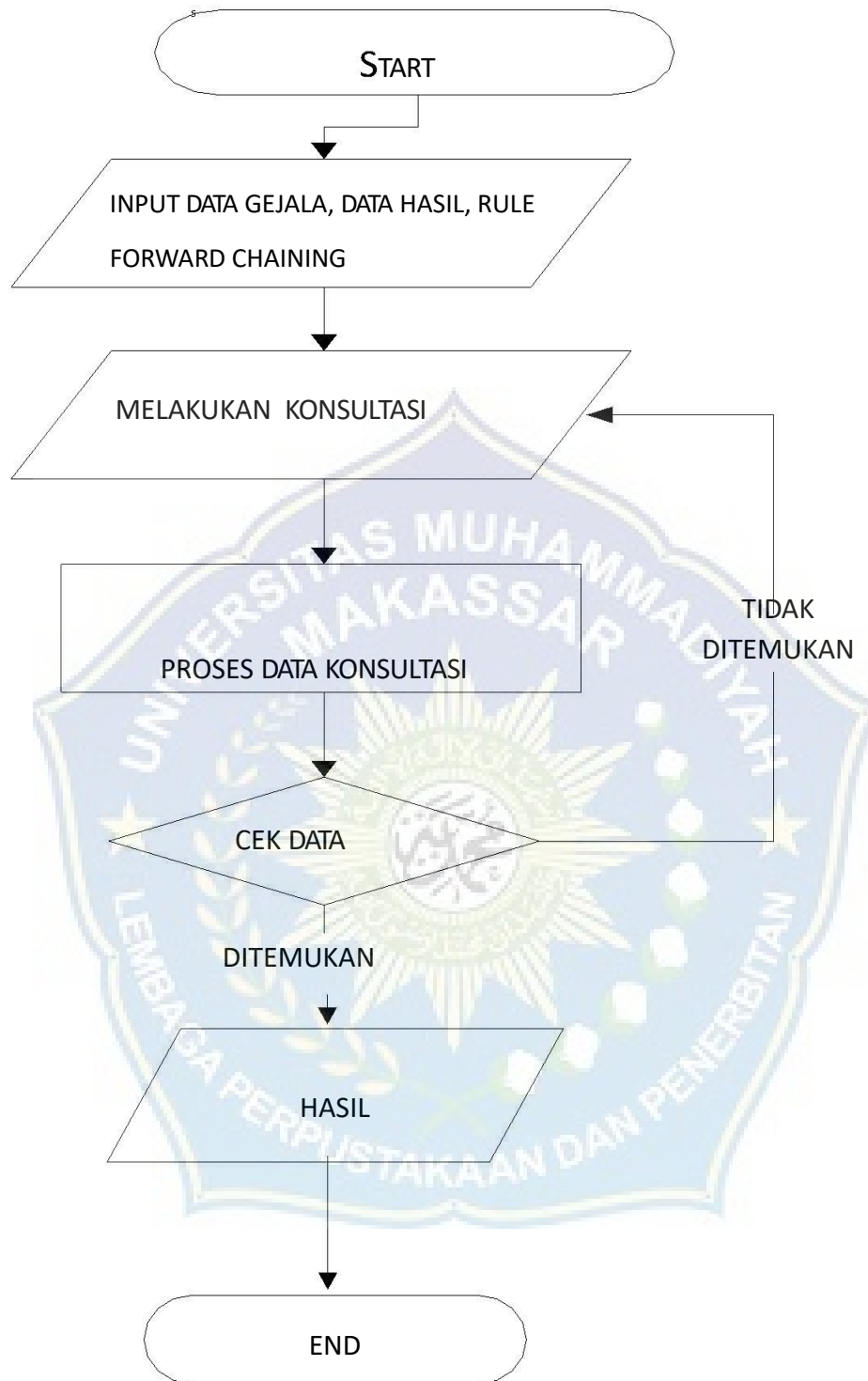
Perancangan sistem sangatlah penting dalam pengembangan sebuah sistem karena menguraikan bagaimana suatu sistem itu dibangun, ditahap perencanaan hingga tahap pembuatan fungsi-fungsi yang diperlukan untuk pengoperasian sistem. Perancangan sistem juga merupakan Langkah kerja dari pembuatan benda uji. Tujuan dari perancangan sistem yaitu untuk menentukan apakah sistem yang dikembangkan akan menghasilkan hasil yang diinginkan.

1. Studi Literatur: Penelitian ini mencakup studi literatur terkait diagnosa penyakit pada tanaman nilam dan penerapan metode Forward Chaining pada sistem pakar. Studi literatur dilakukan melalui buku, jurnal ilmiah, skripsi, paper, dan sumber lainnya.
2. Pengumpulan Data: Data dikumpulkan melalui wawancara dengan seorang pakar. Wawancara dilakukan sebelum dan selama proses pembangunan sistem, untuk memahami penyakit yang sering menyerang tanaman nilam dan melengkapi data penelitian.
3. Pengolahan Data: Data dari wawancara dan observasi diolah untuk dimasukkan ke dalam perancangan sistem.
4. Perancangan Sistem: Perancangan sistem meliputi perancangan arsitektur, diagram, dan antarmuka sistem.
5. Pengujian Sistem: Sistem diuji untuk mengetahui kelayakan dan kesesuaian hasil yang diberikan oleh sistem.
6. Penarikan Kesimpulan: Kesimpulan diambil dari hasil pengujian sistem, berdasarkan kesesuaian antara sistem yang dibangun dengan tujuan penelitian dan kebutuhan informasi pengguna sistem.





Gambar 4. Diagram perancangan sistem



Gambar 5. Diagram flowchart sistem

D. Teknik Pengujian Sistem

Teknik pengujian sistem untuk penelitian sistem pakar mendiagnosa penyakit pada tanaman padi menggunakan metode *forward chaining* yaitu dimulai dengan memasukkan daftar-daftar gejala kondisi yang sedang dialami oleh tanaman padi saat konsultasi. Kemudian diolah untuk menentukan Solusi, yaitu metode pengendalian hama dan penyakit berdasarkan jenis perusak dan gejala yang telah dipilih. Data digunakan dalam sistem ini berasal dari ahli atau pakar mengenai jenis hama dan penyakit. Serta cara pengendaliannya. Data dari ahli diorganisir dan dikodekan untuk memudahkan proses diagnosis selama konsultasi. Kode diberikan pada data gejala, dan kemudian pada data perusak, yang mencakup informasi tentang gejala dan penyakit. Setelah kode telah diberikan pada data gejala dan perusak, Langkah selanjutnya yaitu pembuatan aturan. Aturan ini digunakan dalam sistem selama konsultasi dengan metode *forward chaining*.

E. Teknik dan Analisis Data

Dalam konteks mendiagnosa penyakit pada tanaman padi menggunakan metode *forward chaining*, Teknik analisis data mencakup beberapa Langkah. Pertama, pengumpulan dan kategorisasi data dilakukan melalui observasi langsung dan konsultasi dengan ahlinya untuk mengelompokkan penyakit tanaman padi berdasarkan gejala dan penyebabnya. Kedua, pengetahuan yang diperoleh menjadi basis pengetahuan dengan aturan yang jelas, yang divalidasikan melalui literatur dan pakar. Ketiga, pengembangan aturan menggunakan logika *forward chaining* memungkinkan sistem untuk mendiagnosa berdasarkan gejala yang telah diinputkan. Selanjutnya yaitu analisis heuristik dan manajemen ketidakpastian digunakan untuk mengatasi data yang tidak lengkap atau ambigu. Sistem ini juga diuji dan divalidasi melalui penggunaan data nyata atau memastikan akurasi dan kehandalan. Terakhir yaitu penarikan kesimpulan, untuk menguji sistem.

BAB IV
HASIL DAN PEMBAHASAN

A. Pengumpulan Data

Dalam proses pengumpulan data untuk sistem pakar mendiagnosa penyakit pada tanaman padi menggunakan metode forward chaining, data yang digunakan adalah data yang diambil dari Dinas Perikanan dan Pertanian Kota Makassar. Data ini mencakup informasi mengenai berbagai jenis penyakit yang kerap menyerang tanaman padi, beserta gejala yang ditimbulkan, serta metode pengendalian yang bisa dilakukan. Pengumpulan data yang dilakukan melalui wawancara dengan ahli pertanian, observasi langsung dilapangan, dan analisis penyakit yang terdokumentasi. Data yang terkumpul kemudian dianalisis dan diolah menjadi aturan-aturan dalam sistem *forward chaining*, yang digunakan untuk mengidentifikasi penyakit pada tanaman padi berdasarkan gejala yang teramati, sehingga menghasilkan diagnosa yang akurat dan rekomendasi penanganan yang sesuai.

B. Data Set

Tabel 2. Data Set

No	Nama Penyakit	Gejala	Pengendalian
1	Blass (<i>Pirycularia oryzae</i>)	1. Terdapat bercak-bercak berbentuk seperti belah ketupat dengan ujung rancing 2. Pusat bercak berwarna kelabu atau keputih-putihan mulai yang busuk dan mempunyai tepi coklat atau kemerahan 3. busuknya ujung tangkai malai yang disebut busuk leher	1. Pola tanaman atau Pergiliran tanaman dengan tanaman bukan padi 2. Penggunaan Varietas Tahan penyakit 3. Aplikasi AH corynebacterium sebagai seed treatmen dan umur 2, 4, 6 mst

4. tangkai malai yang busuk mudah patah dan menyebabkan gabah hampa
5. pada gabah yang sakit terdapat bercak-bercak kecil yang bulat

4. Penggunaan Fungisida seperti Delsen, Rabcide, Puanmur, Nordox dll.
5. Menanam dengan jarak yang cukup untuk memastikan sirkulasi udara yang baik dan mengurangi kelembaban
6. Aplikasi fungisida secara preventif pada tahap awal pertumbuhan tanaman yang dapat membantu mencegah infeksi

2 Wereng batang coklat

1. Tanaman padi tampak menguning dan kering
2. Terjadinya kerusakan yang seperti terbakar (hopperburn)
3. Pengisian bulir pada padi tidak maksimal
4. Tanaman terlihat kerdil dan tidak berkembang dengan baik terdapat penyakit kerdil rumput (grassy stunt) dan virus kerdil hampa (ragged stunt)

1. pola tanam
2. Tanam serentak sekurang-kurangnya meliputi satu wilayah kelompok
3. Pergiliran varietas tahan dengan sumber ketahanan (tertua) yang berbeda
4. penanaman variates unggul tahan wereng (VUTW) sesuai dengan biotipe setempat

3 Penggerek batang

1. matinya pucuk ditengah atau biasa disebut sundep
2. terjadinya serangan pada stadia generatif yang menyebabkan
- 3.tanaman muda yang diserang akan mati sebelum atau sesudah terjadinya anakan
4. terdapat lubang-lubang kecil pada batang yang menjadi tanda keluar

- 5.Eradikasi dan sanitasi tanaman terserang, ringan sampai berat serta eradikasi pada tanaman yang puso
6. Penggunaan Agens Hayati seperti Beauveria, Metarhizium
7. Aplikasi pestisida nabati seperti Nimba, sirsak, daun tembakau dll.
8. Aplikasi insektisida buprofezin (Applaud 10 WP) apabila ditemukan wereng coklat rata-rata 1 ekor pertunas dengan stadia dominan nimfa pada semua umur tanaman
- 9.Aplikasi insektisida golongan karbamat (BPMC,MIPC) apabila ditemukan wareng coklat rata-rata 1 ekor pertunas pada semua tanaman
- 1.dilakukan penanaman swrentak
- 2.Pergiliran tanaman bukan padi
- 3.Pengaturan waktu tanam yaitu pada awal musim hujan berdasarkan penerbangan ngegat atau populasi larva ditunggal padi
4. Cara fisik yaitu dengan cara penyabitan tanaman serendah

masuknya penggerek batang

mungkin

5. Cara mekanik dapat dilakukan dengan cara mengumpulkan kelompok telur penggerek batang padi di persemaian

6. Pemanfaatan musuh alami baik parasitoid. Konversasi musuh alami dengan cara menghindari aplikasi isektisida secara semprotan

7. Penggunaan seks feromon

8. Penggunaan insektisida butiran mengandung karbofuran dengan nama dagang Furadan 3G, Dharmafur 3G, Curaterr 3G, Indofuran 3G, Tomafur 3G, Taburan 3G, Petrofur 3G, Hidrofur 3G

4 Busuk Pelepah

1. Lesi berbentuk oval pada pelepah daun dengan tepi coklat dan pusat abu-abu

2. Pelepah yang terinfeksi bisa membusuk dan patah

3. Lesi sering memiliki tepi yang lebih gelap, memberikan penampilan berbatasan yang jelas

4. Lesi pada daun mirip dengan yang ada pada pelepah tetapi biasanya

1. Menggunakan varietas tahan penyakit

2. Mengatur jarak tanam yang optimal untuk sirkulasi udara yang baik

3. Menghindari penanaman terlalu rapat

4. Aplikasi fungisida secara tepat waktu, gunakan fungisida

lebih kecil

5. Lesi dapat meluas dengan cepat dan menyebabkan pembusukan pada Sebagian besar pelepah daun

berbahan aktif seperti benomyl, propikonazol, atau azoksistrobin.

5. Mengatur sistem irigasi untuk mengurangi kelembaban berlebih

6. . Mengeringkan sawah secara periodik untuk mengurangi kondisi lembab yang mendukung

perkembangan jamur

7. Rotasi tanaman dengan tanaman non padi untuk memutas siklus hidup jamur

8. Sanitasi lapangan dengan membersihkan sisa-sisa tanaman yang terinfeksi untuk mengurangi sumber inokulum

5 Hawar daun bakteri (Bacterial Leaf Blight)

1. Munculnya garis-garis coklat panjang pada daun

2. Daun menjadi kering dan menguning

3. Malai bisa menjadi kering dan tidak menghasilkan bulir

4. jika infeksi terjadi pada fase generatif, malai bisa terinfeksi dan menyebabkan bulir menjadi hampa atau berisi butir padi yang kecil dan berkualitas rendah

5. Lesi dapat memanjang hingga

1. Menggunakan varieties tahan penyakit

2. Menghindari kepadatan tanam yang berlebih

3. Mengatur jarak tanam yang baik untuk memastikan sirkulasi udara yang cukup

4. Menggunakan pupuk dengan tepat dan seimbang

5. Menghindari penggunaan

mencapai panjang beberapa sentimeter dan sering kali bergabung satu sama lain

6.. Garis-garis atau lesi memanjang sejajar dengan tulang daun utama

7. Muncul bercak atau garis-garis berwarna hijau kelabu hingga hijau muda

nitrogen yang berlebihan

6. Aplikasi bahan kimia seperti antibiotik seperti streptomisin atau bakteri sida yang dapat membantu mengendalikan penyakit, meskipun penggunaannya terbatas dan harus sesuai dengan rekomendasi

7. Mengatur irigasi untuk mengurangi kelembaban berlebih disekitar tanaman

8. Menghindari genangan air yang berkepanjangan di sawah

9. Sanitasi lapangan dengan membersihkan sisa-sisa tanaman yang terinfeksi untuk mengurangi sumber inoculum

10. Rotasi tanaman dengan tanaman non padi untuk memutus siklus hidup bakteri

6 Tungro

1. Daun berubah menjadi kuning oranye yang dimulai dari ujung dan tepi daun

2. Biji yang dihasilkan sedikit atau tidak ada

3. Tanaman menjadi kerdil dan pertumbuhan terhambat

1. Menggunakan varietas tahan penyakit

2. Penggunaan insektisida untuk mengendalikan populasi wereng hijau

3. Menghilangkan gulma dan sisa-sisa tanaman yang dapat

		menjadi tempat persembunyian wereng hijau
	4. Daun yang lebih muda menunjukkan perubahan warna yang lebih cepat dan lebih intens dibandingkan daun yang lebih tua	4. Mengatur rotasi tanaman dengan tanaman non padi untuk memutus siklus virus
	5. Internoda batang menjadi lebih pendek, yang menyebabkan tanaman tampak lebih pendek dari biasanya	5. Menanam dengan jarak yang cukup untuk memastikan sirkulasi udara yang baik
	6. Malai yang kurang terisi atau hanya setengah terisi	6. Mengatur waktu tanam untuk menghindari periode populasi wereng hijau yang tinggi
		7. Menggunakan musuh alami wereng hijau, seperti predator dan parasitoid untuk mengendalikan populasi wereng hijau
7	Kerdil Rumput	1. Penggunaan instektisida berbahan aktif seperti imidakloprid, fipronil atau buprofezin
	1. Daun yang terinfeksi menjadi pendek, kaku, berwarna hijau gelap	2. Menggunakan varietas tahan penyakit
	2. Tanaman menunjukkan pertumbuhan yang terhambat, tampak kerdil dan tidak tumbuh setinggi tanaman sehat	3. Pemantauan populasi wereng secara rutin dan aplikasi insektisida secara tepat waktu
	3. Tunas lateral atau anakan berkembang lebih banyak, tetapi tidak tumbuh dengan baik	4. Menanam dengan jarak yang
	4. Tanaman yang terinfeksi berat	

	tidak menghasilkan malai	cukup untuk memastikan sirkulasi udara baik dan mengurangi kelembaban yang mendukung populasi wereng
	5. Tanaman yang terinfeksi berat menunjukkan penampakan seperti rumput dengan banyak tunas pendek atau kaku	5. Pengaturan menanam pada waktu yang tepat untuk menghindari periode populasi wereng yang tinggi
	6. Daun menjadi mengecil dan menggulung	6. Melakukan rotasi dengan tanaman non padi untuk memutus siklus hidup virus
		7. Mengatur sistem irigasi untuk menghindari genangan air yang berkepanjangan
8	Kerdil Hampa	1. Menggunakan variates tahan penyakit
	1. Daun muda yang terinfeksi menguning dan berubah menjadi oranye kemerahan	2. Menggunakan musuh alami wereng coklat seperti predator dan parasitoid untuk mengendalikan populasi wereng secara alami
	2. Tanaman tumbuh kerdil dan batang menjadi memendek	3. Penggunaan insektisida berbahan aktif seperti imidakloprid, fipronil atau buprofezin
	3. Pada infeksi berat, malai tidak muncul dan tanaman tidak menghasilkan biji	4. Melakukan pemantauan populasi wereng secara rutin
	4. Daun yang terinfeksi menggulung dan berkerut	5. Menanam dengan jarak yang cukup untuk memastikan sirkulasi udara yang baik dan
	5. Ujung daun dapat tampak bergerigi atau terpotong secara tidak teratur	

6. Daun menjadi lebih sempit dan pendek

mengurangi kelembaban
6. Mengatur sistem irigasi untuk menghindari genangan air yang berkepanjangan yang dapat mendukung populasi wereng

7. Internoda batang menjadi lebih pendek, menyebabkan tanaman tampak lebih pendek dari biasanya

8. Bulir pada malai yang terinfeksi sering kali tidak terisi atau hanya setengah terisi

9. Munculnya bulu halus pada bagian bawah daun atau batang

Pada tabel diatas dijelaskan mengenai 8 penyakit yang kerap menyerang proses pertumbuhan tanaman padi beserta gejala yang ditimbulkan serta apa saja pengendalian yang dapat dilakukan untuk mengatasi berbagai macam penyakit tersebut.

C. Desain Tabel

1. Tabel User

Tabel 3. Tabel User

Table	Type	Collaction	Null	Default	Extra	Action
Id	Int (11)		No	None	AUTO_INCREMENT	Change
Email	Varcher (191)	Utf8mb4_unicode_ci	No	None		Change
Password	Varcher (191)	Utf8mb4_unicode_ci	No	None		Change
Name	Varcher (191)	Utf8mb4_unicode_ci	Yes	Null		Change
Refresh token	Varcher (500)	Utf8mb4_unicode_ci	Yes	Null		Change

Pada tabel ini terdiri dari beberapa kolom yang memiliki fungsi spesifik dalam konteks sistem. Kolom **'id'** berfungsi sebagai kunci utama yang memberikan identitas unik pada setiap elemen atau entri dalam tabel, memastikan tidak adanya duplikasi. Kolom **'Email'** dan **'Password'** digunakan untuk autentikasi pengguna, Dimana 'Email' berfungsi sebagai mengidentifikasi login, dan **'Password'** memastikan keamanan akses ke sistem. Selain itu, kolom 'nama' berfungsi sebagai penyimpanan untuk informasi yang berkaitan dengan penyakit dan gejala yang terkait dengan pengguna, yang memungkinkan pencatatan dan analisis data medis. Terakhir, kolom **'Refresh token'** berfungsi dalam proses autentikasi berkelanjutan, Dimana token ini dapat membantu memperpanjang sesi pengguna tanpa adanya login ulang, memastikan bahwa identitas pengguna dapat diverifikasi secara aman dan efisien dalam jangka waktu tertentu.

2. Tabel Penyakit

Table 4. Tabel Penyakit

Table	Type	Collaction	Null	Default	Extra	Action
Id	Int (11)		No	None	AUTO_INCREMENT	Change
Kode_penyakit	Varcher (191)	Utf8mb4_unicode_ci	No	None		Change
Nama	Varcher (191)	Utf8mb4_unicode_ci	No	None		Change

Pada tabel ini mencakup beberapa kolom penting yang masing-masing terdapat fungsi. Kolom **'id'** berfungsi sebagai kunci utama, memberikan identitas pada setiap entri, sehingga setiap elemen dapat dibedakan dengan jelas. Kolom **'kode penyakit'** berfungsi menyimpan klarifikasi yang sesuai dengan standar yang telah disepakati, serta Tindakan yang disarankan berdasarkan kriteria penyakit tersebut. Kolom **'nama'** berfungsi untuk mencatat nama penyakit yang sudah ditetapkan, yang memudahkan dalam mengidentifikasi dan akses terhadap penyakit tertentu dalam proses diagnosis dan perawatan.

3. Tabel Gejala

Tabel 5. Tabel Gejala

Table	Type	Collaction	Null	Default	Extra	Action
Id	Int (11)		No	None	AUTO_INCREMENT	Change
Kode_gejala	Varcher (191)	Utf8mb4_unicode_ci	No	None		Change
Deksripsi	Varcher (191)	Utf8mb4_unicode_ci	No	None		Change
penyakitid	Int (11)		No	None		Change

Pada tabel ini mencakup bagian-bagian yang memiliki peran dan fungsinya masing-masing. Bagian **'id'** berfungsi sebagai kunci utama, memberikan identitas pada setiap elemen didalam bentuk tabel guna memastikan bahwa setai entri dapat dibedakan dengan jelas. Bagian **'kode gejala'** berfungsi untuk menyimpan kode klariifikasi yang sesuai dengan standar gejala yang telah ditetapkan, serta Tindakan yang disarankan berdasarkan kriteria gejala tersebut. Bagian **'deksripsi'** berfungsi memberikan rincian tentang gejala, yang dapat membantu pemgguna untuk lebih memahami karakteristik gejala. Bagian **'penyakit id'** berfungsi menghubungkan gejala dengan penyakit tertentu yang telah ditetapkan oleh admin, memungkinkan sistem untuk mengaitkan gejala dengan penyakit yang relevan.

4. Tabel Hasil Diagnosa

Tabel 6. Tabel Hasil Diagnosa

Table	Type	Collaction	Null	Default	Extra	Action
Id	Int (11)		No	None	AUTO_INCREMENT	Change
Tanggal diagnosa	Datetime (3)		No	Current_timesta mp(3)		Change
User	Varcher (191)	Utf8mb4_unicode_ci	No	None		Change
Persentase	Double		Yes	Null		Change

Tabel ini mencakup beberapa bagian yang memiliki fungsinya masing-masing yang spesifik. Bagian **'id'** bertindak sebagai kunci utama, memberikan identitas unik untuk setiap elemen dalam tabel, sehingga setiap entri dapat dikenali dengan jelas. Bagian **'tanggal diagnosa'** berfungsi mencatat kapan diagnosa dilakukan, memungkinkan pelacakan waktu yang berkaitan dengan proses tersebut. Bagian **'user'** berfungsi sebagai antarmuka, di mana pengguna dapat memilih gejala yang mereka alami atau ingin diperiksa. Setelah gejala dipilih, sistem akan menghitung dan menampilkan persentase kemungkinan penyakit yang mungkin diderita oleh pengguna, berdasarkan metode perhitungan yang telah diterapkan dalam sistem. Bagian ini

membantu memberikan estimasi atau prediksi yang lebih informatif terkait penyakit yang mungkin dialami, mendukung proses diagnosa yang berbasis data.

D. Desain Web

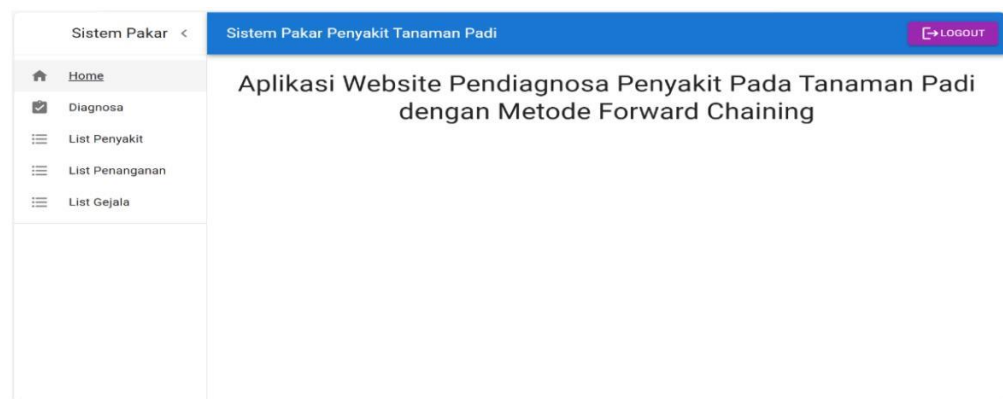
1. Proses Admin
 - a. Halaman login



Gambar 6. Halaman Login

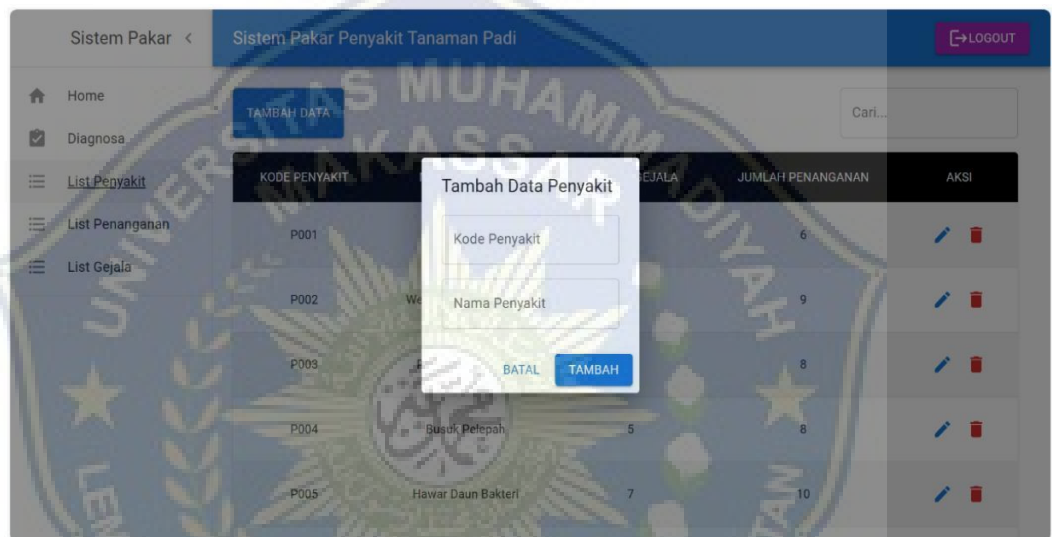
Halaman login merupakan antarmuka pertama yang muncul pada saat aplikasi dibuka, yang memiliki fungsi sebagai pintu utama yang mengontrol akses pengguna ke fitur dan konten di dalam sebuah aplikasi. Pada halaman ini, admin atau pengguna diminta untuk memasukkan kredensial mereka, yaitu Alamat email dan kata sandi password, ke dalam kolom input yang tersedia.

- b. Halaman Home



Halaman Home akan muncul apabila admin berhasil melakukan login, yang menandakan bahwa autentikasi telah dilakukan dengan benar dan tepat. Pada halaman ini, admin memiliki akses penuh ke berbagai fitur yang telah tersedia di dalam aplikasi web seperti fitur list penyakit, list penanganan, dan juga list gejala. Halaman home ini dirancang dengan antarmuka yang mudah digunakan, menampilkan informasi secara jelas dan terorganisir, yang memudahkan menjalankan tugas administratif.

c. Halaman Daftar Penyakit

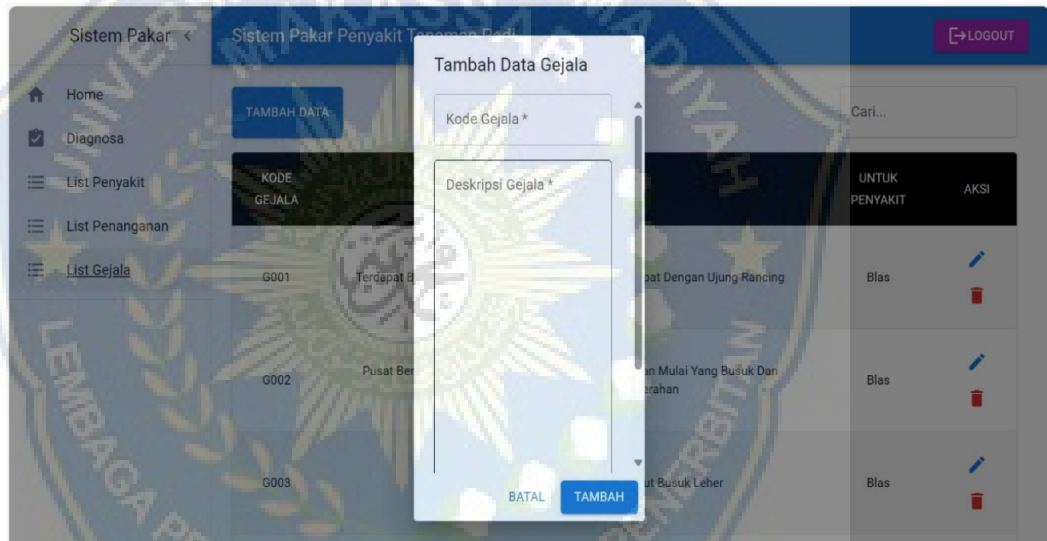


Gambar 8. Halaman Daftar penyakit









Pada halaman ini, admin dapat mengakses daftar penyakit dan menambahkan kode serta nama penyakit yang diinginkan. Di halaman ini juga admin dapat melihat tabel yang menampilkan informasi penting seperti kode penyakit, nama penyakit, jumlah gejala, jumlah penanganan serta terdapat fitur mengedit atau memperbarui data tersebut. Tabel tersebut memberikan kemudahan admin dalam melihat dan mengelola data penyakit dalam sistem, memastikan setiap penyakit memiliki kode unik dan gejala yang sesuai. Fitur edit memungkinkan admin untuk melakukan perubahan ataupun pembaruan data kapan saja, sehingga data tetap akurat.

d. Halaman Daftar Gejala



Gambar 10. Halaman Daftar Gejala

KODE GEJALA	DESKRIPSI GEJALA	UNTUK PENYAKIT	AKSI
G001	Terdapat bercak-bercak berbentuk seperti belah ketupat dengan ujung rancing	Bias	 
G002	Pusat bercak berwarna kelabu atau keputih-putihan mulai yang busuk dan mempunyai tepi coklat atau kemerahan	Bias	 
G003	Busuknya ujung tangkai malai yang disebut busuk leher	Bias	 

Pada halaman ini, admin akan membuka daftar gejala dan menambahkan informasi seperti kode gejala, deksripsi gejala dari penyakit yang terkait. Di halaman ini, terdapat tabel yang menampilkan kode gejala, deksripsi gejala, serta penyakit yang dihubungkan dengan gejala tersebut. Dalam juga fitur untuk mengedit data, yang memungkinkan admin dapat memperbarui atau mengubah informasi sesuai kebutuhan. Tabel ini dapat membantu admin dalam memantau dan mengelola data gejala, memastikan setiap gejala diidentifikasi dengan benar dan dikaitkan dengan penyakit yang tepat. Fitur edit juga memberikan fleksibilitas untuk menjaga agar data tetap akurat dan juga relevan.

e. Halaman Daftar Penanganan

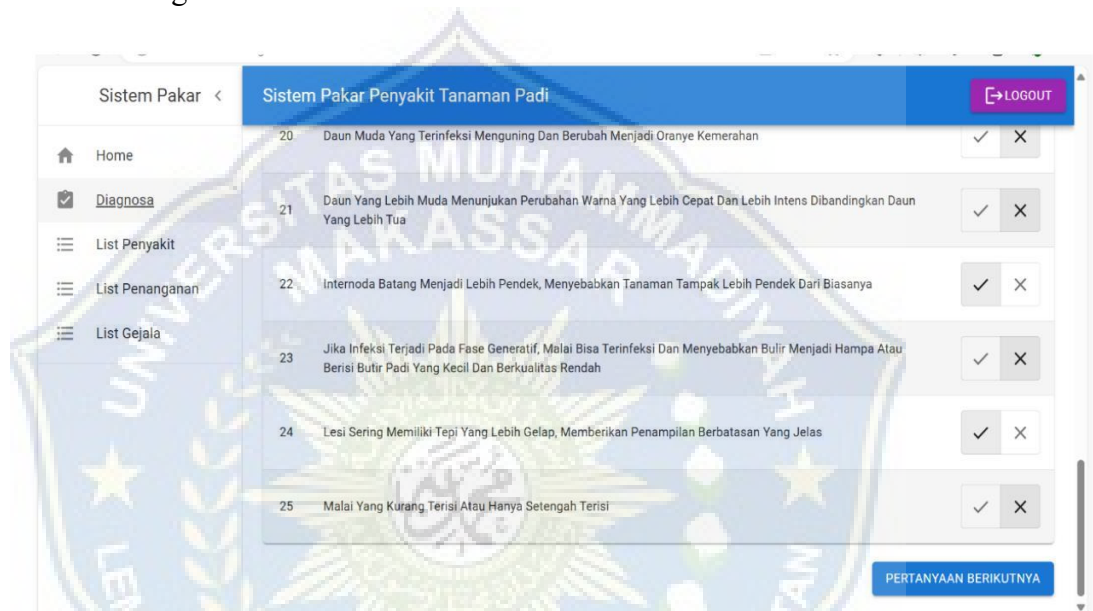


Gambar 12. Halaman Daftar Penanganan

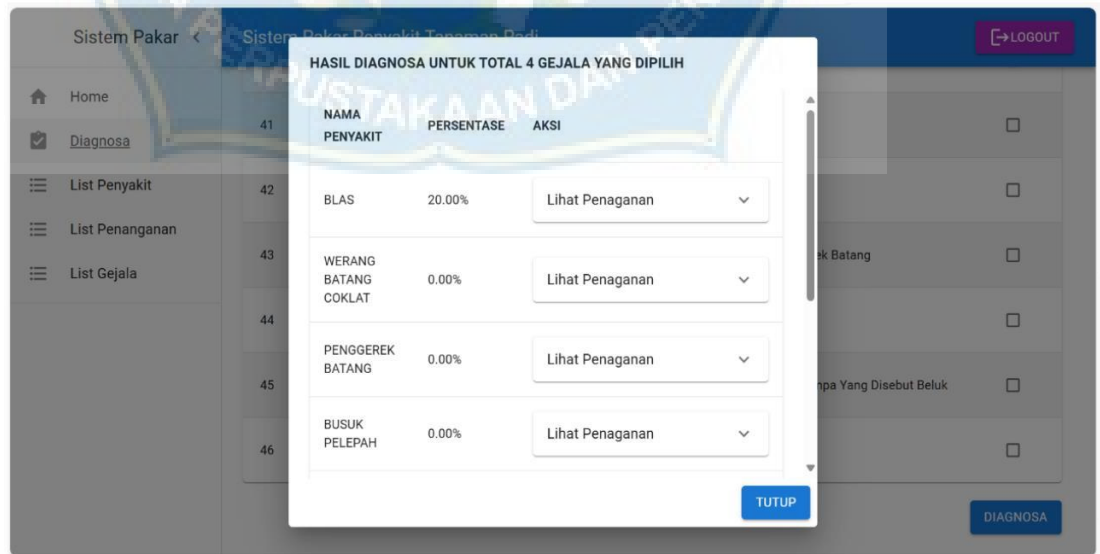
KODE PENGENDALIAN	DESKRIPSI PENGENDALIAN	UNTUK PENYAKIT	AKSI
S001	.Pola Tanaman Atau Pergiliran Tanaman Dengan Tanaman Bukan Padi	Bias	 
S002	Penggunaan Varietas Tahan Penyakit	Bias	 
S003	Penggunaan Varietas Tahan Penyakit Mst	Bias	 

Pada halaman ini, admin dapat mengakses penanganan-penanganan yang dapat dilakukan untuk mengatasi penyakit-penyakit pada tanaman padi. Pada halaman ini juga menampilkan kode pengendalian, deskripsi pengendalian dan juga nama penyakit yang berkaitan dengan pengendalian tersebut.

2. Proses User
 a. Halaman Diagnosa



Gambar 14. Halaman Diagnosa



Pada halaman ini, admin terlebih dahulu akan memilih gejala-gejala yang telah tersedia dengan cara menjawab pertanyaan yang sesuai dengan gejala yang dialami dengan cara mencentang ya atau tidak. Setelah semua gejala yang relevan telah dipilih, admin akan menekan tombol 'Diagnosa'. Sistem akan memproses data yang telah diinput dan menggunakan algoritma diagnosa yang telah ditetapkan untuk menghasilkan diagnosa. Hasil tersebut akan tampil dilayar, menunjukkan penyakit yang paling mungkin berdasarkan gejala yang telah dipilih, Bersama dengan Tingkat keakuratan serta Langkah penanganan yang harus dilakukan.

E. Desain From

1. Form Login



Gambar 16. Form Login

Halaman login adalah antarmuka pertama yang ditampilkan saat aplikasi dibuka, berfungsi sebagai pintu masuk utama. Pada halaman ini, admin diminta untuk memasukkan email dan password ke dalam kolom yang telah disediakan. Jika data telah dimasukkan, sistem akan memverifikasi kombinasi email dan password tersebut. Jika data email dan password yang dimasukkan tidak terverifikasi maka sistem akan menampilkan pesan kesalahan.

2. Form Home



Gambar 17. Form Home

Form Home adalah tampilan awal yang muncul setelah admin berhasil membuka website pendiagnosa penyakit. Halaman ini berfungsi sebagai pusat navigasi utama, yang memungkinkan admin untuk mengakses berbagai fitur dan data yang telah tersedia. Adapun fitur-fitur penting seperti menu untuk memilih daftar penyakit, gejala penyakit, penanganan serta diagnosa yang telah dilakukan.

3. Form Diagnosa



Gambar 18. Form Diagnosa

Form Diagnosa adalah halaman khusus yang memungkinkan pengguna mendiagnosis penyakit berdasarkan gejala yang telah dialami. Di halaman ini, terdapat 46 gejala yang telah ditampilkan secara lengkap. Admin terlebih dahulu harus memilih gejala yang telah disediakan dengan cara mencentang checkbox yang sesuai dengan kondisi yang dialami. Setelah semua gejala yang relevan telah dipilih, admin akan menekan tombol ‘Diagnosa’, sistem akan memproses gejala yang dipilih dan menjalankan algoritma diagnostic untuk menghasilkan hasil diagnosa. Hasil diagnosa ini akan ditampilkan dilayar, menunjukkan penyakit apa yang paling mungkin berdasarkan gejala yang dipilih, serta informasi tambahan seperti Tingkat keakuratan dan penanganan dari penyakit tersebut.

4. Form Daftar Penyakit

KODE PENYAKIT	NAMA PENYAKIT	JUMLAH GEJALA	JUMLAH PENANGANAN	AKSI
P001	Bilas	5	6	 
P002	Werang Batang Coklat	4	9	 
P003	Penggerek Batang	4	8	 
P004	Busuk Pelepah	5	8	 
P005	Hawar Daun Bakteri	7	10	 

Gambar19. Form Daftar Penyakit

Form daftar penyakit mencakup beberapa elemen yang penting yang memungkinkan admin dapat mengelola data penyakit dengan efisien. Di dalam form ini, terdapat opsi ‘Tambah data’, ‘kode penyakit’, ‘jumlah gejala’, ‘jumlah penanganan’, dan ‘aksi’ yang di dalam aksi tersebut terdapat fitur edit dan hapus.

5. Form Daftar Gejala

KODE GEJALA	DESKRIPSI GEJALA	UNTUK PENYAKIT	AKSI
G001	Terdapat Bercak-Bercak Berbentuk Seperti Belah Ketupat Dengan Ujung Rancing	Bias	[Edit] [Hapus]
G002	Pusat Bercak Berwarna Kelabu Atau Keputih-Putih Han Mulai Yang Busuk Dan Mempunyai Tepi Coklat Atau Kemerahan	Bias	[Edit] [Hapus]
G003	Busuknya Ujung Tangkai Malai Yang Disebut Busuk Leher	Bias	[Edit] [Hapus]

Gambar 20. Form Daftar Gejala

From daftar gejala mencakup beberapa elemen yang penting yang memungkinkan admin dapat mengelola data gejala dengan efisien. Di dalam form ini, terdapat opsi ‘Tambah data’, ‘kode gejala’, ‘deksripsi gejala’, ‘”untuk penyakit’, dan ‘aksi’ yang di dalam aksi tersebut terdapat fitur edit dan hapus.

6. Form Daftar penanganan

KODE PENGENDALIAN	DESKRIPSI PENGENDALIAN	UNTUK PENYAKIT	AKSI
S001	.Pola Tanaman Atau Pergiliran Tanaman Dengan Tanaman Bukan Padi	Bias	[Edit] [Hapus]
S002	Penggunaan Varietas Tahan Penyakit	Bias	[Edit] [Hapus]
S003	Penggunaan Varietas Tahan Penyakit Mst	Bias	[Edit] [Hapus]

Gambar 21. Form Daftar Penanganan

From daftar penanganan mencakup beberapa elemen yang penting yang memungkinkan admin dapat mengelola data penanganan dengan efisien. Di dalam form ini, terdapat opsi ‘Tambah data’, ‘kode penanganan’, ‘deksripsi penamnganan’, ‘untuk penyakit’, dan ‘aksi’ yang di dalam aksi tersebut terdapat fitur edit dan hapus.

F. Pengujian Sistem

Pengujian sistem dari metode *forward chaining* dirancang untuk mendiagnosa penyakit pada tanaman padi. Pengujian sistem ini menerapkan pengujian *Black box*, yang secara spesifik berfokus pada aspek fungsional perangkat lunak dan evaluasi Keputusan yang dihasilkan oleh perangkat lunak. Tujuan utama dari pengujian ini adalah memastikan bahwa perangkat lunak dapat secara akurat mendiagnosa penyakit pada tanaman padi berdasarkan gejala-gejala yang diberikan, serta dapat menghasilkan Keputusan yang sesuai dengan aturan didalam basis pengetahuan yang telah ditetapkan. Selain itu, pengujian ini juga mencakup evaluasi antarmuka pengguna untuk mengidentifikasi dan mengekspos kesalahan dalam desain antarmuka, seperti kemudahan pengguna dalam kejelasan informasi yang disajikan. Kasus pengujian melibatkan berbagai scenario gejala untuk menilai apakah sistem mampu memberikan diagnosis yang benar dan dapat diandalkan. Hasil dari pengujian ini digunakan untuk menentukan apakah aplikasi berbasis web tersebut siap digunakan secara luas oleh pengguna seperti para petani ataupun ahli pertanian, dan apakah sistem dapat memberikan hasil yang akurat serta mudah dipahami.

Tabel 7. Fitur hasil pengujian

No	Uraian Pengujian	Proses Pengujian	Hasil Pengujian
1	Login, input <i>username</i> dan <i>Password</i>	Dapat login dan masuk ke halaman home	Diterima
2	Input data penyakit	Data berhasil di simpan di database	Diterima
3	Input data gejala	Data berhasil di simpan di database	Diterima
4	Input data penanganan	Data berhasil di simpan di database	Diterima
7	Klik tombol kode penyakit atau gejala	Data berhasil di simpan di database	Diterima
8	Klik tombol hapus	Data yang sudah disimpan akan terhapus	Diterima
9	Klik tombol edit	Akan mengedit data yang telah disimpan	Diterima

BAB V

KESIMPULAN

A. Kesimpulan

1. Perancangan dan pengembangan aplikasi web yang menggunakan metode forward chaining untuk mendiagnosis penyakit pada tanaman padi melibatkan beberapa tahap, penetapan aturan-aturan diagnosis, serta pembuatan antarmuka pengguna. Aplikasi ini harus mampu menerima input gejala dari pengguna dan secara otomatis melakukan proses *forward chaining* untuk menghasilkan diagnosis. Dengan ini, petani atau pengguna dapat mudah melakukan diagnosis penyakit padi sendiri melalui web, yang memudahkan akses dan menghemat waktu dalam indentifikasi dan penanganan penyakit pada tanaman padi.
2. Berdasarkan hasil pengujian, sistem tersebut menunjukkan tingkat akurasi yang tinggi dalam memberikan diagnosis penyakit, yang menunjukkan bahwa metode forward chaining berhasil menelusuri fakta-fakta yang ada dan mencapai kesimpulan yang akurat. Tingginya tingkat akurasi ini menunjukkan bahwa sistem pakar tersebut memiliki potensi yang besar untuk digunakan oleh petani atau penyuluh pertanian sebagai alat bantu dalam mengidentifikasi penyakit pada tanaman padi dengan cepat dan efisien. Dengan demikian, sistem ini dapat memberikan kontribusi yang signifikan dalam meningkatkan produktivitas pertanian dan pengendalian penyakit pada tanaman padi.

B. Saran

Dengan menambahkan fitur kompleks seperti modul pembelajaran berbasis AI, dan kemampuan analitik yang lebih mendalam, perbaikan ini akan meningkatkan akurasi diagnosis, personalisasi rekomendasi, dan kemampuan sistem untuk beradaptasi dengan berbagai kondisi dan variasi tanaman padi, sehingga sistem akan lebih efektif dan bermanfaat bagi penggunanya.

DAFTAR PUSTAKA

- Alim, Syahirul, Peni Puji Lestari, and Rusliyawati Rusliyawati. 2020. "2. Sistem Pakar Diagnosa Penyakit Tanaman Kakao Menggunakan Metode Certainty Factor Pada Kelompok Tani Pt Olam Indonesia (Cocoa) Cabang Lampung." *Jurnal Data Mining dan Sistem Informasi* 1(1): 26. doi:10.33365/jdmsi.v1i1.798.
- Cantika, Shintia, Indra Kanedi, and Juju Jumadi. 2023. "Sistem Pakar Untuk Mendiagnosa Penyakit Bronkho Pneumonia Menggunakan Metode Certainty Factor Pada Klinik Assifa Bengkulu." *Jurnal Media Infotama* 19(2): 358–66. doi:10.37676/jmi.v19i2.4239.
- Darussalam, Fajar Ghulam, and Heribertus Yulianton. 2023. "15. Sistem Pakar Diagnosa Penyakit Tanaman Padi Menggunakan Metode Forward Chaining." *Joutica* 8(2): 39–44. doi:10.30736/informatika.v8i2.1096.
- Gergely, Stefan. 2024. "IMPLEMENTASI SISTEM PAKAR MENGGUNAKAN METODE CERTAINTY FACTOR DALAM MENDIAGNOSA PENYAKIT JAMURAN PADA TUMBUHAN PADI SKRIPSI." (February): 4–6.
- Jeraman, Firgilius, NM Faizah, and Lucky Koryanto. 2023. "13 Sistem Pakar Diagnosa Penyakit Pada Tanaman Padi Kecamatan Satarmese Kabupaten Manggarai Provinsi Nusa Tenggara Timur Berbasis Web Dengan Metode Forward Chaining." *Computer Journal* 1(1): 73–81. doi:10.58477/cj.v1i1.66.
- Kecamatan, Pongsamelung, and Lamasi Kabupaten. 2022. "1. Hubungan Faktor Sosial Ekonomi Petani Padi Dengan Pelaksanaan Pengelolaan Tanaman Terpadu (PTT) Di Desa Pendahuluan." 2(1): 1–10.
- Kuswanto, Joko. 2020. "12. Sistem Pakar Untuk Perlindungan Tanaman Padi Menggunakan Metode Forward Chaining." *Eduatic - Scientific Journal of Informatics Education* 7(1): 31–39. doi:10.21107/edutic.v7i1.8805.
- L Tobing, Demas Matias, Elvis Pawan, Friden E Neno, and Kusrini Kusrini. 2019. "5. Sistem Pakar Mendeteksi Penyakit Pada Tanaman Padi Menggunakan Metode Forward Chaining." *Sisfotenika* 9(2): 126. doi:10.30700/jst.v9i2.440.
- Nasution, Guslila Sari. 2022. "15. Sistem Pakar Dalam Mendiagnosis Hama Blas Dan Kresek

Pada Tanaman Padi Menggunakan Metode Forward Chaining.” *Jurnal Sistim Informasi dan Teknologi* 4: 6–9. doi:10.37034/jsisfotek.v4i4.144.

Normah, Bakhtiar Rifai, Satrio Vambudi, and Rifki Maulana. 2022. “7. Analisa Sentimen Perkembangan Vtuber Dengan Metode Support Vector Machine Berbasis SMOTE.” *Jurnal Teknik Komputer AMIK BSI* 8(2): 174–80. doi:10.31294/jtk.v4i2.

Studi, Program, Teknik Informatika, Fakultas Teknik, and Universitas Mataram. 2019. “3. SISTEM PAKAR DIAGNOSIS PENYAKIT TANAMAN PADI MENGGUNAKAN FORWARD CHAINING DAN DEMPSTER SHAFER.”

Zuraida, Vit, Deddy Kusbianto, and Mohammad Reza Pahlevi. 2023. “4. Sistem Pakar Diagnosis Penyakit Dan Hama Pada Tanaman Padi Dengan Metode Forward Chaining.” *Jurnal Minfo Polgan* 12(1): 378–84. doi:10.33395/jmp.v12i1.12437.



L

A

M

P

I

R

A

N



Lampiran 1. Pengantar Penelitian Kepada Ketua LP3M Unismuh Makassar

**UNIVERSITAS MUHAMMADIYAH MAKASSAR**
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Nomor : 379/05/C.4-VI/VI/45/2024
Lamp. :-
Hal : Pengantar Penelitian

Makassar, 04 Dzulhijjah 1445 H
11 Juni 2024 M

Kepada yang Terhormat,
Ketua LP3M Unismuh Makassar
Di -
Tempat

Assalamu 'Alaikum Warahmatullahi Wabarakatuh

Dengan Rahmat Allah SWT, Semoga aktivitas kita bernilai ibadah di Sisi - Nya. Dalam rangka penyelesaian Tugas Sarjana / Tugas Akhir Mahasiswa pada Program Studi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar dengan judul: **"Sistem Pakar Mendiagnosa Penyakit Pada Tanaman Padi Menggunakan Metode Forward Chaining"**, Sehubungan hal tersebut, maka kami meminta kesediaan Bapak/Ibu agar kiranya berkenan membantu perihal surat tersebut. Bersama ini kami sampaikan mahasiswa(i):

No.	Stambuk	Nama
1.	105 84 11039 20	Nurul Aulia Muslimah

Demikian surat kami atas perhatian dan kerja samanya kami haturkan banyak terima kasih.
Jazakumullah Khaeran Katsiran
Wassalamu 'Alaikum warahmatullah Wabarakatuh

Ketua Program Studi
Informatika

M. M. Hayat, S.Kom., MT.
NPM: 1004577

Tembusan: Kepada Yang Terhormat,
1 Dekan Fakultas Teknik
2 Arsip



Gedung Menara Iqra Lantai 3
Jl. Sultan Alauddin No. 259 Telp. (0411) 866 972 Fax (0411) 865 588 Makassar 90221
Web: <https://teknik.unismuh.ac.id/>, e-mail: teknik@unismuh.ac.id



Lampiran 2. Pengantar Penelitian Bapak Gubernur Prov Sul.Sel

 MAJELIS PENDIDIKAN TINGGI PIMPINAN PUSAT MUHAMMADIYAH
UNIVERSITAS MUHAMMADIYAH MAKASSAR
LEMBAGA PENELITIAN PENGEMBANGAN DAN PENGABDIAN KEPADA MASYARAKAT
D. Sultan Alauddin No. 239 Telp.8449772 Fax (0411)2602000 Makassar 90221 e-mail :ip3m@uimuh.ac.id

Nomor : 4492/05/C.4-VIII/VI/1445/2024
Lamp : 1 (satu) Rangkap Proposal
Hal : Permohonan Izin Penelitian

21 June 2024 M
15 Dzulhijjah 1445

Kepada Yth,
Bapak Gubernur Prov. Sul-Sel
Cq. Kepala Dinas Penanaman Modal & PTSP Provinsi Sulawesi Selatan
di -
Makassar

Berdasarkan surat Dekan Teknik Universitas Muhammadiyah Makassar, nomor: 379/05/C.4-VI/VI/45/2024 tanggal 11 Juni 2024, menerangkan bahwa mahasiswa tersebut di bawah ini :

Nama : NURUL KULIA MUSLIMAH
No. Stambuk : 10504 1103620
Fakultas : Teknik
Jurusan : Informatika
Pekerjaan : Mahasiswa

Bermaksud melakukan penelitian/pengumpulan data dalam rangka penulisan Skripsi dengan judul:

"SISTEM PAKAR MENDIAGNOSA PENYAKIT PADA TANAMAN PADI MENGGUNAKAN METODE FORWARD CHAINING"

Yang akan dilaksanakan dari tanggal 25 Juni 2024 s/d 25 Agustus 2024.

Sehubungan dengan maksud di atas, kiranya mahasiswa tersebut diberikan izin untuk melakukan penelitian sesuai ketentuan yang berlaku.

Demikian, atas perhatian dan kerjasamanya diucapkan Jazakumullahu khaeran


Ketua LP3M,

Dr. Muh. Ariet Muhsin, M.Pd.
NBM 1127761

06-24

3. Lampiran source code

```
import jwt from "jsonwebtoken"
import { PrismaClient } from "@prisma/client"
import bcrypt from "bcrypt"
import dotenv from "dotenv"
dotenv.config()
const prisma = new PrismaClient()

function generateToken(user) {
  return jwt.sign({ userId: user.id },
    process.env.JWT_SECRET_KEY, { expiresIn:
    process.env.JWT_SECRET_EXPIRES_IN })
}

function generateRefreshToken(user) {
  return jwt.sign({ userId: user.id },
    process.env.JWT_REFRESH_KEY, { expiresIn:
    process.env.JWT_REFRESH_EXPIRES_IN })
}

export const Login = async (req, res) => {
  try {
    const {
      email,
      password
    } = req.body

    const user = await prisma.user.findUnique({ where: {
    email } })
```

```
    if (!user) {
      return res.status(404).json({ msg: "Email tidak
ditemukan" })
    }

    const passwordMatch = await bcrypt.compare(password,
user.password)

    if (!passwordMatch) {
      return res.status(401).json({ msg: "Password salah"
})
    }

    // const accessToken = generateToken(user)
    const accessToken = jwt.sign({ id: user.id },
process.env.JWT_SECRET_KEY, { expiresIn:
process.env.JWT_SECRET_EXPIRES_IN })

    const refreshToken = jwt.sign({ id: user.id },
process.env.JWT_REFRESH_KEY, { expiresIn:
process.env.JWT_REFRESH_EXPIRES_IN })

    // const refreshToken = generateRefreshToken(user)

    await prisma.user.update({
      where: {
        id: user.id
      },
```

```

        data: {
            refreshToken: refreshToken
        }
    })

    res
        .cookie("_XYZabc123", refreshToken, {
            httpOnly: true,
            maxAge: 24 * 60 * 60 * 1000,
        })
        .status(200).json({ accessToken, refreshToken, id:
user.id, email: user.email, username: user.username, roleId:
user.roleId })

    } catch (error) {
        console.error(error)
        res.status(500).json({ msg: "Terjadi kesalahan saat
proses login" })
    }
}

export const Logout = async (req, res) => {
    try {
        const refreshToken = req.cookies._XYZabc123
        if (!refreshToken) return res.sendStatus(401);
        const user = await prisma.user.findUnique({
            where: {
                refreshToken: refreshToken
            }
        })
    }
}

```

```

    })
    if (!user) return res.sendStatus(204)
    await prisma.user.update({
      where: {
        id: user.id
      },
      data: {
        refreshToken: null
      }
    })
    res.clearCookie('refreshToken')
    res.sendStatus(200)
  } catch (error) {
    console.log(error)
  }
}

export const createUser = async (req, res) => {
  try {
    const { name, email, password, id } = req.body
    const hashedPassword = await bcrypt.hash(password, 10)
    const newUser = await prisma.user.create({
      data: {
        name,
        email,
        password: hashedPassword,
      }
    })
  }
}

```

```

    const accessToken = jwt.sign({ id: id },
process.env.JWT_SECRET_KEY, { expiresIn:
process.env.JWT_SECRET_EXPIRES_IN })

    res.cookie('accessToken', accessToken, {
        httpOnly: true,
        maxAge: 24 * 60 * 60 * 1000,
    })

    res.json({newUser, accessToken})
} catch (error) {
    console.error(error)
    res.status(500).json({ error: 'Error creating user.' })
}
}

```

```

import { PrismaClient } from "@prisma/client";
const prisma = new PrismaClient();

export const Diagnosa = async (req, res) => {
    try {
        const { gejalaIds } = req.body;

        // Ambil semua penyakit beserta gejalanya
        const penyakitWithGejala = await
prisma.penakit.findMany({
            include: {
                gejala: true,
                solusi: true
            },
        });
    }
};

```

```

// Proses forward chaining
const result = penyakitWithGejala.map((penyakit) => {
  // Ambil ID gejala yang terkait dengan penyakit
  const gejalaPenyakitIds = penyakit.gejala.map(g =>
g.id);

  // Cocokkan gejala yang diberikan pengguna dengan
gejala penyakit

  const matchingGejala = gejalaPenyakitIds.filter(id
=> gejalaIds.includes(id));

  // Cek apakah gejalaPenyakitIds memiliki elemen
untuk menghindari pembagian dengan nol
  let persentaseKecocokan = 0;
  if (gejalaPenyakitIds.length > 0) {
    // rumus persentase forward chaining
    persentaseKecocokan = (matchingGejala.length /
gejalaPenyakitIds.length) * 100;
  }

  return {
    penyakitName: penyakit.nama,
    persentaseKecocokan:
persentaseKecocokan.toFixed(2) + '%',
    diagnosis: matchingGejala.length ===
gejalaPenyakitIds.length ? 'Ditemukan' : 'Sebagian Cocok',
    solusi: penyakit.solusi
  };
}).filter(Boolean); // Filter out null results

```



```
    if (result.length > 0) {
      res.json(result);
    } else {
      res.status(404).json({ message: 'Tidak ada penyakit
yang cocok dengan gejala tersebut.' });
    }
  } catch (error) {
    console.error(error);
    res.status(500).json({ error: 'Internal Server Error'
});
  }
}
```

```
import { PrismaClient } from "@prisma/client";
const prisma = new PrismaClient();

export const getAllPenyakit = async (req, res) => {
  try {
    const penyakit = await prisma.penakit.findMany({
      include: {
        gejala: true,
        solusi: true,
      }
    })
    res.status(200).json(penyakit);
  } catch (error) {
    console.error(error)
    res.status(500).json({ error: 'Error getting penyakit.'
})
}
```

```
    }  
  }  
  
export const getPenyakitById = async (req, res) => {  
  const { id } = req.params;  
  try {  
    const penyakit = await prisma.penakit.findUnique({  
      where: { id: Number(id) },  
      include: {  
        gejala: true,  
        solusi: true,  
      }  
    })  
    res.status(200).json(penyakit);  
  } catch (error) {  
    console.error(error)  
    res.status(500).json({ error: 'Error getting penyakit.' })  
  }  
}
```

```
export const addPenyakit = async (req, res) => {  
  const { nama, kode penyakit, solusi } = req.body;  
  try {  
    const penyakit = await prisma.penakit.create({  
      data: {  
        kode_penyakit: kode_penyakit,  
        nama: nama,  

```

```

        },
    })
    res.status(200).json(penyakit);
} catch (error) {
    console.error(error)
    res.status(500).json({ error: 'Error adding penyakit.'
})
}
}

export const editPenyakit = async (req, res) => {
    const { id } = req.params;
    const { nama, kode_penyakit, solusi } = req.body;
    try {
        const updatedPenyakit = await prisma.penakit.update({
            where: { id: Number(id) },
            data: {
                kode_penyakit: kode_penyakit,
                nama: nama,
            },
        })
        res.status(200).json(updatedPenyakit);
    } catch (error) {
        console.error(error)
        res.status(500).json({ error: 'Error updating penyakit.'
})
    }
}
}

```

```
export const deletePenyakit = async (req, res) => {
  const { id } = req.params;
  try {
    const penyakit = await prisma.penakit.findUnique({
      where: { id: Number(id) },
      include: {
        gejala: true,
        solusi: true,
      },
    });

    await prisma.gejala.deleteMany({
      where: {
        id: {
          in: penyakit.gejala.map((gejala) =>
            gejala.id),
        },
      },
    });

    await prisma.solusi.deleteMany({
      where: {
        id: {
          in: penyakit.solusi.map((solusi) =>
            solusi.id),
        },
      },
    });
  }
};
```



```
const deletedPenyakit = await prisma.penyakit.delete({
  where: { id: Number(id) },
});

res.status(200).json(deletedPenyakit);
} catch (error) {
  console.error(error);
  res.status(500).json({ error: 'Error deleting penyakit.'
});
}
};
```

```
export const getAllGejala = async (req, res) => {
  try {
    const gejala = await prisma.gejala.findMany()
    res.status(200).json(gejala);
  } catch (error) {
    console.error(error)
    res.status(500).json({ error: 'Error getting gejala.' })
  }
}
```

```
export const getGejalaById = async (req, res) => {
  const { id } = req.params;
  try {
    const gejala = await prisma.gejala.findUnique({
      where: { id: Number(id) },
    })
```

```
        res.status(200).json(gejala);
    } catch (error) {
        console.error(error)
        res.status(500).json({ error: 'Error getting gejala.' })
    }
}
```

```
export const addGejala = async (req, res) => {
    const { kode_gejala, deskripsi, penyakitId } = req.body;
    try {
        const gejala = await prisma.gejala.create({
            data: {
                kode_gejala,
                deskripsi,
                penyakitId: parseInt(penyakitId),
            },
        });
        res.status(200).json(gejala);
    } catch (error) {
        console.error(error)
        res.status(500).json({ error: 'Error adding gejala.' })
    }
}
```

```
export const editGejala = async (req, res) => {
    const { id } = req.params;
    const { deskripsi, kode_gejala, penyakitId } = req.body;
    try {
```

```

const updatedGejala = await prisma.gejala.update({
  where: { id: Number(id) },
  data: {
    deskripsi: deskripsi,
    kode_gejala: kode_gejala,
    penyakitId: parseInt(penyakitId),
  },
})
res.status(200).json(updatedGejala);
} catch (error) {
  console.error(error)
  res.status(500).json({ error: 'Error updating gejala.'
})
}
}

export const deleteGejala = async (req, res) => {
  const { id } = req.params;
  try {
    const deletedGejala = await prisma.gejala.delete({
      where: { id: Number(id) },
    })
    res.status(200).json(deletedGejala);
  } catch (error) {
    console.error(error)
    res.status(500).json({ error: 'Error deleting gejala.'
})
  }
}
}

```

```

export const addSolusi = async (req, res) => {
  const { penyakitId, deskripsi, kode_solusi } = req.body;
  try {
    const solusi = await prisma.solusi.create({
      data: {
        kode_solusi,
        deskripsi,
        penyakitId: parseInt(penyakitId),
      },
    });
    res.status(200).json(solusi);
  } catch (error) {
    console.error(error)
    res.status(500).json({ error: 'Error adding solusi.' })
  }
}

export const editSolusi = async (req, res) => {
  const { id } = req.params;
  const { kode_solusi, deskripsi, penyakitId } = req.body;
  try {
    const penyakit = await prisma.solusi.update({
      where: { id: Number(id) },
      data: {
        kode_solusi: kode_solusi,
        deskripsi: deskripsi,
        penyakitId: parseInt(penyakitId),
      },
    });
  }
}

```



```

        },
    });
    res.status(200).json(penyakit);
} catch (error) {
    console.error(error)
    res.status(500).json({ error: 'Error updating solusi.'
})
}
}

export const getAllSolusi = async (req, res) => {
    try {
        const solusi = await prisma.solusi.findMany()
        res.status(200).json(solusi);
    } catch (error) {
        console.error(error)
        res.status(500).json({ error: 'Error getting solusi.' })
    }
}

export const getSolusiById = async (req, res) => {
    const { id } = req.params;
    try {
        const solusi = await prisma.solusi.findUnique({
            where: { id: Number(id) },
        });
        res.status(200).json(solusi);
    } catch (error) {
        console.error(error)
    }
}

```

```
        res.status(500).json({ error: 'Error getting solusi.' })
    }
}

export const deleteSolusi = async (req, res) => {
    const { id } = req.params;
    try {
        const deletedSolusi = await prisma.solusi.delete({
            where: { id: Number(id) },
        });
        res.status(200).json(deletedSolusi);
    } catch (error) {
        console.error(error)
        res.status(500).json({ error: 'Error deleting solusi.'
    })
    }
}
```



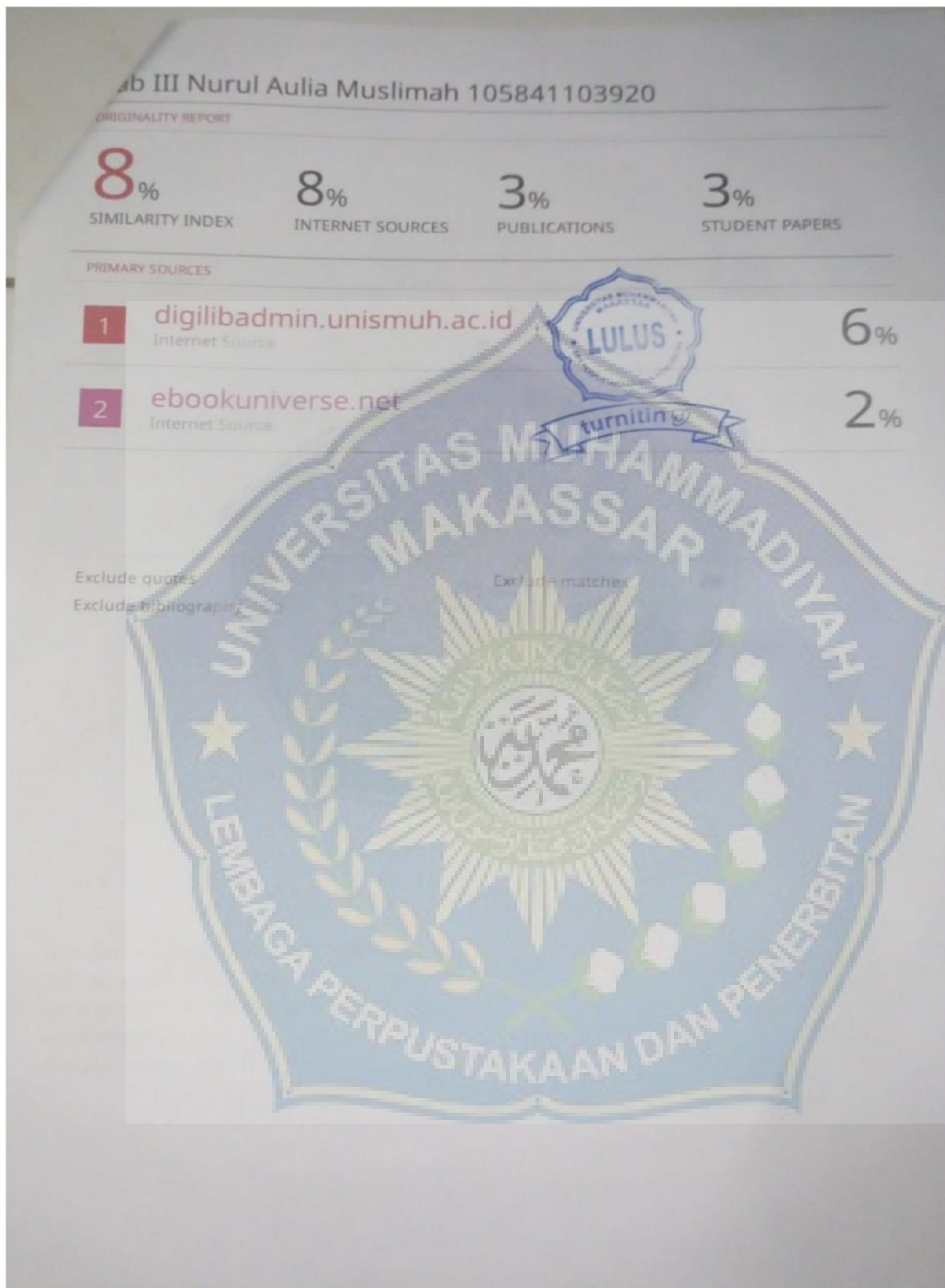
Lampiran 4. Hasil Plagiasi Bab 1



Lampiran 5. Hasil Plagiasi Bab II



Lampiran 6. Hasil Plagiasi Bab III



Lampiran 7. Hasil Plagiasi Bab IV



Lampiran Hasil Plagiasi Bab V

Bab V Nurul Aulia Muslimah 105841103920

ORIGINALITY REPORT

0% SIMILARITY INDEX	0% INTERNET SOURCES	0% PUBLICATIONS	0% STUDENT PAPERS
------------------------	------------------------	--------------------	----------------------

PRIMARY SOURCES

Exclude quotes OFF

Exclude bibliography OFF

Exclude matches OFF

LULUS

turnitin

