

**SISTEM PAKAR DETEKSI PENYAKIT MULUT DAN KUKU PADA HEWAN
TERNAK SAPI DENGAN CASE BASED REASONING
DAN CERTAINTY FACTOR**

SKRIPSI

MIFTAH NUR HABIBI

191401064



**PROGRAM STUDI S1-ILMU KOMPUTER FAKULTAS ILMU
KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2024**

**SISTEM PAKAR DETEKSI PENYAKIT MULUT DAN KUKU PADA HEWAN
TERNAK SAPI DENGAN CASE BASED REASONING
DAN CERTAINTY FACTOR**

SKRIPSI

Diajukan untuk melengkapi tugas dan memenuhi syarat memperoleh ijazah Sarjana Ilmu
Komputer

MIFTAH NUR HABIBI

191401064



**PROGRAM STUDI S1-ILMU KOMPUTER FAKULTAS ILMU
KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2024**

PERSETUJUAN

Judul : SISTEM PAKAR DETEKSI PENYAKIT MULUT
DAN KUKU PADA HEWAN TERNAK SAPI
DENGAN CASE BASED REASONING DAN
CERTAINTY FACTOR

Kategori : SKRIPSI

Nama : MIFTAH NUR HABIBI

Nomor Induk Mahasiswa : 191401064

Program Studi : SARJANA (S-1) ILMU KOMPUTER

Departemen : ILMU KOMPUTER

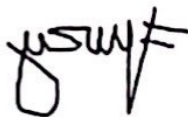
Fakultas : ILMU KOMPUTER DAN TEKNOLOGI
INFORMASI

Medan, 5 Januari 2024

Komisi Pembimbing:


Pembimbing 2

Pembimbing 1



Fuzy Yustika Manik S.Kom., M.Kom.

NIP. 198710152019032010



Herriyance S.T., M.Kom.

NIP. 198010242010121002

Diketahui/disetujui oleh

Program Studi S-1 Ilmu Komputer



Dr. Amalia, S.T., M.T.

NIP. 197812212014042001

PERNYATAAN**SISTEM PAKAR DETEKSI PENYAKIT MULUT DAN KUKU PADA HEWAN
TERNAK SAPI DENGAN CASE BASED REASONING
DAN CERTAINTY FACTOR****SKRIPSI**

Saya mengakui bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing telah disebutkan namanya.

Medan, 05 Januari 2024



Miftah Nur Habibi

191401064

PENGHARGAAN

Puji Syukur penulis panjatkan kepada Tuhan yang maha Esa yang telah memberikan rahmat dan hidayahnya sehingga penulis dapat menyelesaikan skripsi ini sebagai salah satu syarat kelulusan di program studi Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.

Selama dalam penyelesaian tugas akhir ini, banyak sekali bantuan dan dukungan serta doa dari berbagai pihak sehingga penulis ingin menyampaikan rasa syukur, beruntung serta rasa terima kasih sedalam-dalamnya dari penulis untuk semua pihak yang telah membantu penulis dalam penyelesaian skripsi ini. Penulis mengucapkan terima kasih kepada :

1. Bapak Prof. Dr. Murianto Amin, S.Sos., M.Si. selaku rektor Universitas Sumatera Utara.
2. Ibu Dr. Maya Silvi Lydia B.Sc., M.sc., selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
3. Ibu Dr. Amalia ST., M.T., selaku Ketua Program Studi S1 Ilmu Komputer Universitas Sumatera Utara.
4. Bapak Herriyance S.T., M.Kom., selaku dosen pembimbing I penulis.
5. Ibu Fuzy Yustika Manik S.Kom., M.Kom., selaku dosen pembimbing II penulis.
6. Bapak drh. Anwar, M.Pt, selaku Pakar yang memberikan informasi terkait penelitian ini.
7. Kepada kedua orang tua penulis yang telah memberikan doa dan dukungan baik moral maupun materi yang tiada hentinya selalu bersama penulis.
8. Kepada kakak yang telah memberikan dukungan dan doa yang tiada henti kepada penulis.

9. Teman – teman yang di dalam dan di luar dunia perkuliahan yang selalu ada untuk memberikan dukungan dan memberi warna dalam kehidupan penulis, yang tidak dapat disebutkan satu persatu.
10. *Last but not least*, terima kasih untuk diri sendiri, yang telah bertahan hingga sampai saat ini.

Medan, 05 Januari 2024

Penulis,

A handwritten signature in black ink, appearing to read 'miftah' followed by a stylized flourish.

Miftah Nur Habibi

ABSTRAK

Penyakit mulut dan kuku pada hewan ternak sapi merupakan masalah kesehatan ternak yang memiliki dampak serius terhadap produksi peternakan dan kesejahteraan hewan serta kerugian besar bagi peternak. Diagnosa dini dan tepat merupakan kunci untuk mengurangi dampak penyakit ini. Dalam rangka meningkatkan diagnosis penyakit mulut dan kuku pada sapi, skripsi ini bertujuan untuk mengembangkan sebuah Sistem Pakar yang dapat membantu dokter hewan dan peternak dalam proses diagnosis. Untuk memudahkan para peternak mendiagnosis penyakit mulut dan kuku, para ahli membuat sebuah program aplikasi untuk memberikan solusi melalui proses deteksi dini dan pemberian rekomendasi pengobatan yang tepat. Dengan menggunakan algoritma *Certainty Factor* dan *Case Based Reasoning*. *Case Based Reasoning* untuk mencari nilai kemiripan kasus baru dengan kasus- kasus yang telah ada, dan *Certainty Factor* untuk mencari nilai kepastian dari kerusakan yang dialami dengan menghitung nilai bobot. Hasil perhitungan nilai akurasi terhadap pengujian sistem pakar ini, didapatkan nilai akurasi sebesar 86% dan membuktikan bahwa sistem pakar diagnosis Penyakit mulut dan kuku dapat berjalan dengan baik.

Kata Kunci : Penyakit Mulut dan Kuku, Sapi, Sistem Pakar, *Certainty Factor*, *Case Based Reasoning*

ABSTRACT

The aim of this thesis is to develop an Expert System that can assist veterinarians and farmers in diagnosing foot and mouth disease on cattle, which is one of the health problems of cattle and has a negative impact on livestock production and animal welfare, and causes many losses for farmers. It will be very important to reduce the effects of this disease by diagnosing it quickly and precisely. This program provides solutions through an early detection process and appropriate treatment advice for foot and mouth disease (FMDV), an infectious and acute disease. To determine the value of similarity between the new case and the previous case, the certainty factor and case-based reasoning algorithms are used. By using the Certainty Factor and Case-Based Reasoning algorithms. Case Based Reasoning to find the similarity value of new cases with existing cases, and Certainty Factor to find the certainty value of the damage experienced by calculating the weight value. The results of calculating the accuracy value for testing this expert system, obtained an accuracy value of 86% and prove that the expert system for diagnosing foot and mouth disease can work well.

Keywords : Mouth and nails disease, cattle, Expert System, Certainty Factor, Case Based Reasoning

DAFTAR ISI

| | |
|--|------------|
| PERSETUJUAN | i |
| PERNYATAAN..... | ii |
| PENGHARGAAN..... | iii |
| ABSTRAK | v |
| ABSTRACT | vi |
| DAFTAR ISI..... | vii |
| DAFTAR GAMBAR..... | ix |
| DAFTAR TABEL | x |
| DAFTAR LAMPIRAN | xi |
| BAB 1 PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Rumusan Masalah | 3 |
| 1.3 Batasan Masalah | 4 |
| 1.4 Tujuan Penelitian | 4 |
| 1.5 Manfaat Penelitian | 4 |
| 1.6 Penelitian yang Relevan..... | 5 |
| 1.7 Metodologi Penelitian | 6 |
| 1.8 Sistematika Penulisan | 6 |
| BAB 2 LANDASAN TEORI..... | 8 |
| 2.1 Sistem Pakar..... | 8 |
| 2.1.1 Kelebihan Sistem Pakar | 9 |
| 2.1.2 Kekurangan Sistem Pakar | 9 |
| 2.1.3 Ciri-Ciri Sistem Pakar | 9 |
| 2.1.4 Struktur Sistem Pakar..... | 9 |
| 2.3 Case Based Reasoning | 11 |
| 2.3.1 Kelebihan Case Based Reasoning..... | 12 |
| 2.3.2 Kekurangan Case Based Reasoning | 12 |
| 2.4 Certainty Factor..... | 13 |
| 2.4.1 Kelebihan Certainty Factor | 14 |
| 2.4.2 Kekurangan Certainty Factor | 14 |
| BAB 3 ANALISIS DAN PERANCANGAN SISTEM | 15 |
| 3.1 Analisis Sistem..... | 15 |
| 3.1.1 Analisis Masalah | 15 |
| 3.1.2 Analisis Kebutuhan | 16 |
| 3.1.3 Analisis Proses | 16 |
| 3.2 Arsitektur Umum | 21 |

| | | |
|-----------------------------|--|-----------|
| | viii | |
| 3.3 | Pemodelan Sistem | 22 |
| 3.4 | Flowchart | 27 |
| 3.4.1 | Diagram Case Based Reasoning | 27 |
| 3.4.2 | Flowchart Certainty Factor | 30 |
| 3.4.3 | Flowchart Sistem | 31 |
| 3.5 | Perancangan <i>Interface</i> | 32 |
| BAB 4 | IMPLEMENTASI DAN PENGUJIAN SISTEM | 36 |
| 4.1 | Implementasi Sistem | 36 |
| 4.1.1 | Halaman Beranda..... | 36 |
| 4.1.2 | Halaman Info..... | 36 |
| 4.1.3 | Halaman Admin | 37 |
| 4.1.4 | Halaman Cek Sekarang..... | 38 |
| 4.1.5 | Halaman Hasil..... | 38 |
| 4.2 | Hasil Pengujian | 39 |
| BAB 5 | PENUTUP | 41 |
| 5.1 | Kesimpulan | 41 |
| 5.2 | Saran | 41 |
| DAFTAR PUSTAKA | | 42 |
| LAMPIRAN..... | | 43 |

DAFTAR GAMBAR

| | |
|---|----|
| Gambar 2.1 Struktur Sistem Pakar | 10 |
| Gambar 3.1 Diagram Ishikawa | 15 |
| Gambar 3.2 Arsitektur Umum Sistem | 22 |
| Gambar 3.3 Use Case Diagram | 26 |
| Gambar 3.4 Activity Diagram | 27 |
| Gambar 3.5 Sequence Diagram | 28 |
| Gambar 3.6 diagram Case Based Reasoning | 29 |
| Gambar 3.7 Flowchart Certainty Factor | 30 |
| Gambar 3.8 Flowchart Sistem | 31 |
| Gambar 3.9 Rancangan Halaman Beranda..... | 32 |
| Gambar 3.10 Rancangan Halaman Info..... | 33 |
| Gambar 3.11 Rancangan Halaman Admin | 33 |
| Gambar 3.12 Halaman Cek | 34 |
| Gambar 3.13 Halaman Hasil Diagnosis..... | 35 |
| Gambar 4.1 Halaman Beranda..... | 36 |
| Gambar 4.2 Halaman Info | 37 |
| Gambar 4.3 Halaman Admin..... | 37 |
| Gambar 4.4 Halaman Cek Sekarang..... | 38 |
| Gambar 4.5 Halaman Hasil | 39 |

DAFTAR TABEL

| | |
|---|----|
| Tabel 3.1 Bobot Kondisi Penyakit Mulut dan Kuku | 17 |
| Tabel 3.2 Gejala-gejala Penyakit mulut dan kuku | 18 |
| Tabel 3.3 Solusi dan Penanganan Penyakit mulut dan kuku | 18 |
| Tabel 3.4 Dataset Penyakit mulut dan kuku..... | 19 |
| Tabel 3.5 Gejala Penyakit mulut dan kuku | 19 |
| Tabel 3.6 Contoh Inputan User..... | 19 |
| Tabel 3.7 Perhitungan Nilai Kepastian..... | 20 |
| Tabel 4.1 Hasil Pengujian Sistem kepada Pakar | 39 |

DAFTAR LAMPIRAN

| | |
|-----------------------------------|----|
| Pseudo code..... | 44 |
| Code system | 45 |
| Foto bersama pakar..... | 50 |
| Bukti hasil wawancara pakar | 51 |
| Foto bersama peternak..... | 52 |

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Penyakit kuku dan mulut merupakan penyakit infeksius dan akut yang disebabkan oleh Foot and Mouth Disease Virus(FMDV)dan dapat menyerang hewan berkuku genap atau belah seperti sapi, kuda, kambing dan babi (Rohma, M. R., & Zamzami, A., 2022).

Pada tahun 2022 muncul penyakit ternak yaitu penyakit mulut dan kuku, wabah virus yang meluas pada ternak. Wabah virus ini sangat menular dan mempengaruhi semua jenis hewan dengan kuku datar atau terbelah. Penyebaran wabah tersebut sangat mengkhawatirkan peternak, salah satunya peternak sapi. Akibat wabah ini, banyak peternak yang mengalami kerugian besar, baik secara finansial maupun waktu yang mereka habiskan untuk beternak(Bani, A. U., & Asruddin, A., 2022).

Dampak dari PMK berpengaruh negatif secara signifikan terhadap produksi dan produktifitas ternak mulai dari penurunan produksi susu, bobot badan, keguguran hingga kematian. Sifat penyakit mulut dan kuku (PMK) yang sangat menular merupakan cerminan dari serangkaian faktor, antara lain sebagai berikut:

- berbagai macam hewan berkuku belah yang rentan
- sejumlah besar virus yang dilepaskan oleh hewan yang terinfeksi
- kisaran ekskresi dan sekresi yang dapat menular
- banyaknya rute infeksi
- dosis kecil virus yang dapat memulai infeksi
- Stabilitas moderat virus di lingkungan. (Donaldson, 2001)

Dengan Teknologi zaman sekarang yang semakin canggih, memudahkan masyarakat untuk mengetahui gejala-gejala atau tanda-tanda dari penyakit mulut dan kuku, hal ini tentu dapat memudahkan serta dapat membantu mengurangi efek yang lebih parah dari penyakit ini. Salah satu caranya adalah dengan menggunakan Expert System.

Expert System atau Sistem Pakar adalah salah satu cabang dari AI (Artificial Intelligence) yang membuat penggunaan secara luas mengetahui sebuah pengetahuan atau knowledge yang khusus untuk penyelesaian sebuah masalah. Sistem Pakar dapat diterapkan dalam berbagai bidang seperti Pendidikan, Kesehatan, industri dan lainnya.

Sistem pakar ini akan menampilkan pilihan gejala-gejala serta tindakan atau pengobatan yang harus dilakukan. Sistem pakar ini diharapkan dapat membantu para peternak dalam upaya mendeteksi cepat Penyakit mulut dan kuku ini, sehingga dapat melakukan pengendalian atau pengobatan yang tepat. Pada penelitian ini, metode system pakar yang digunakan adalah Case Based-Reasoning dan Certainty Factor.

Case Based Reasoning (CBR) menyelesaikan masalah dengan cara memanfaatkan pengalaman dari kasus-kasus serupa yang pernah terjadi di masa lalu. Metode ini bertujuan untuk menemukan solusi untuk masalah baru dengan caramencocokkan karakteristik masalah baru dengan kasus-kasus yang telah ada sebelumnya, lalu mengekstrak solusi dari kasus terbaik yang paling relevan dengan masalah baru. CBR digunakan untuk mendukung diagnosa penyakit atau memilih strategi pengobatan yang sesuai dengan kasus yang mirip. Keuntungan utama CBR adalah kemampuannya dalam menangani masalah yang kompleks dan tidak terstruktur, serta mampu mengekstrak pengetahuan dari kasus-kasus spesifik yang ada.

Certainty Factor (CF) merupakan suatu metode untuk membuktikan apakah suatu fakta itu pasti atau tidak pasti. Metode ini sangat cocok untuk sistem pakar yang mendiagnosis sesuatu yang belum pasti. CF menggunakan suatu nilai untuk mengasumsikan derajat keyakinan pemikiran (inexact reasoning) seseorang pakar terhadap suatu nilai. Metode ini menggunakan perhitungan berdasarkan kemiripan yang dibagi dengan bobot yang telah ditentukan.

Kombinasi antara CBR dan CF dapat memperkaya kemampuan sistem pakar dalam memecahkan masalah, dengan memanfaatkan pengalaman dari kasus-kasus sebelumnya (CBR) dan memberikan penilaian kepercayaan yang lebih terukur (CF) terhadap solusi yang dihasilkan.

Metode CF menunjukkan suatu ukuran kepastian terhadap suatu fakta atau aturan. Cara kerja metode CF ini adalah dengan menunjukkan ukuran kepastian terhadap suatu fakta atau aturan. Metode CF melakukan penalaran layaknya seorang pakar, dan untuk mendapatkan nilai kepercayaan (Santi, 2019).

Algoritma CBR dan CF tentunya digunakan karena CBR merupakan penalaran berbasis kasus, yang menitik beratkan pemecahan masalah baru dengan didasarkan pada kasus-kasus sebelumnya yaitu mendiagnosa dengan mencari nilai kemiripan kasus baru tersebut dengan kasus-kasus yang sudah ada. Sedangkan metode CF disini digunakan untuk mencari nilai kepastian dengan menghitung nilai bobot yang telah didapat. Dengan menggunakan metode CF diharapkan mempermudah pengguna untuk mendapatkan sebuah informasi kepastian.(Ayu dkk,2020).

Pada penelitian ini, akan ada tiga tingkatan penyakit mulut dan kuku pada sapi. Dimana dideteksi oleh sistem pakar dengan bantuan dari seorang pakar, dokter hewan. Tingkatan ini didapat dari dokter hewan, buku serta jurnal terdahulu yang membahas tentang penyakit mulut dan kuku serta berdasarkan umumnya kasus penyakit mulut dan kuku pada sapi.

1.2 Rumusan Masalah

Penyakit Mulut dan Kuku merupakan suatu penyakit yang disebabkan oleh suatu virus yang menyerang hewan ternak dengan kuku terbelah salah satunya hewan ternak sapi yang menyebabkan nafsu makan sapi menurun bahkan kematian yang membuat para peternak mengalami kerugian baik secara finansial maupun waktu pemeliharaan. Penyakit mulut dan kuku dapat semakin memburuk jika tidak ditangan dengan cepat. Oleh karena itu, diperlukan sebuah aplikasi berbasis Web dari sistem pakar dengan mengimplementasikan metode *Case Based Reasoning* dan *Certainty Factor* yang dapat mendeteksi penyakit mulut dan kuku sebagai edukasi dini agar peternak dapat melakukan penanganan yang sesuai sesegera mungkin.

1.3 Batasan Masalah

Untuk menghindari penelitian yang tidak terarah, maka peneliti membatasi ruang lingkup masalah yang akan diteliti dengan batasan masalah sebagai berikut :

1. Metode yang dipakai pada penelitian ini adalah *Case Based Reasoning* dan *Certainty Factor*.
2. Referensi data yang digunakan didapat dari pakar dalam hal ini dokter hewan.
3. Penyusunan program hanya ditujukan untuk hewan ternak sapi yang memiliki kemungkinan gejala penyakit mulut dan kuku tanpa adanya komplikasi penyakit lain.
4. Aplikasi berbasis *Website* yang dirancang menggunakan bahasa pemrograman PHP.

1.4 Tujuan Penelitian

Tujuan dilakukannya penelitian ini adalah sebagai berikut :

1. Melakukan deteksi penyakit mulut dan kuku pada hewan ternak sapi dengan metode *Case Based Reasoning* dan *Certainty Factor*.
2. Mengetahui kinerja metode *Case Based Reasoning* dan *Certainty Factor* pada pendeteksian penyakit mulut dan kuku pada hewan ternak sapi.

1.5 Manfaat Penelitian

Manfaat yang diperoleh dari penelitian ini antara lain :

1. Menghasilkan *Website* yang dapat mendeteksi penyakit mulut dan kuku pada hewan ternak sapi dengan metode *Case Based Reasoning* dan *Certainty Factor*.
2. Dapat dijadikan sebagai rujukan untuk penelitian selanjutnya yang berkaitan dengan sistem pakar menggunakan metode *Case Based Reasoning* dan *Certainty Factor*.

1.6 Penelitian yang Relevan

Berikut beberapa penelitian terdahulu yang berkaitan dengan penelitian yang akan dilakukan yaitu :

1. Berdasarkan penelitian (Elisawati, dkk, 2023) yang berjudul “Sistem Pakar Diagnosa Penyakit Sapi dengan Metode Case Based Reasoning Dan Similarity Srgenfrei.” menyatakan bahwa diagnosa penyakit pada hewan ternak sapi dapat dilakukan dengan mudah dengan tingkat kepercayaan hasil diagnosa berupa perhitungan nilai kesamaan/kemiripan dengan data sebelumnya yaitu data kasus yang sudah terdiagnosa dan data yang sudah ditambahkan kedalam database dengan dilihat dari persentasi tertinggi diatas 50% dari hasil perhitungan sebagai hasil kesimpulan diagnosa.
2. Berdasarkan penelitian (Budiman Saiful Nur, Lestanti Sri, Ubaidillah M. M, 2022) yang berjudul “Sistem Pakar Diagnosis Tanaman Anggrek Bulan Dengan Metode Case Based Reasoning Dan Certainty Factor” menyatakan bahwa hasil pengujian sistem sangat membantu dalam mendiagnosis penyakit secara dini, akan tetapi masih perlu dikaji ulang dari data dan algoritma yang dipakai karena hasil yang kurang baik. Dikarenakan memiliki tingkat akurasi bernilai 45% dari total data uji.
3. Berdasarkan penelitian (Aziz Mustafa, dkk, 2021) yang berjudul “Sistem Pakar Penyakit Bawang Merah Menggunakan Case Based Reasoning Dan Certainty Factor” menyatakan bahwa hasil pengujian terhadap responden 10 petani ponorogo sebesar 78,5 % dengan tingkat kepuasan dari semua responden adalah “Setuju”.
4. Berdasarkan penelitian (Ridwan Mujib, dkk, 2021) yang berjudul “Penentuan Ekstrakurikuler Siswa Sesuai Minat Bakat dengan Case-Based Reasoning dan Certainty Factor” menyatakan bahwa testing aktivitas ekstrakurikuler di MAN 1 Lamongan memiliki nilai akurasi yang tinggi, 95.76%.
5. Berdasarkan penelitian (Asruddin, dkk 2022) yang berjudul “Pendektesian Penyakit Mulut dan Kuku Pada Sapi Menggunakan Naïve Bayes” menyatakan bahwa Naïve Bayes dapat digunakan dalam mendeteksi penyakit mulut dan kuku dengan memprediksi peluang yang akan datang berdasarkan pengalaman di masa lalu, dimana hasil penerapan metode Naïve Bayes mendapatkan hasil berdasarkan input gejala user mendapatkan tingkat akurasi hasil penyakit mulut dan kuku pada sapi dengan gejala umum sebesar 86% dan gejala serius sebesar 75%

1.7 Metodologi Penelitian

Metodologi penelitian yang dilakukan dalam penelitian ini adalah:

1. Studi Pustaka

Pada tahap ini, penelitian dimulai dengan mengumpulkan referensi dan tinjauan pustaka dari berbagai sumber berupa buku, jurnal, makalah, artikel, dan penelitian terdahulu yang relevan dan telah dipublikasikan dalam bentuk jurnal yang berkaitan dengan penyakit mulut dan kuku pada hewan ternak sapi, *Case based-Reasoning*, dan *Certainty Factor*.

2. Analisis dan Perancangan

Pada tahap ini, dilakukan pengumpulan informasi langsung terhadap pakar yang bertujuan untuk guna kebutuhan penelitian, dalam kasus ini dilakukan proses wawancara terhadap dokter hewan. Berdasarkan ruang lingkup penelitian, penulis melakukan analisa terhadap apa saja yang perlu diketahui tentang penyakit mulut dan kuku.

3. Implementasi

Pada tahap ini, membuat sebuah sistem dengan menggunakan bahasa pemrograman PHP yang sesuai dengan diagram alir (flowchart) yang telah dirancang berbasis *website*.

4. Pengujian

Pada tahap ini, melakukan proses pengujian dan percobaan terhadap sistem sesuai dengan kebutuhan yang ditentukan sebelumnya serta memastikan program yang dibuat berjalan seperti yang diharapkan.

5. Dokumentasi

Pada tahap ini, penelitian dilakukan dokumentasi dan penulisan laporan mengenai program tersebut yang bertujuan untuk menunjukkan hasil penelitian ini dalam format skripsi.

1.8 Sistematika Penulisan

Sistematika penulisan dari skripsi yang digunakan dalam penelitian ini adalah:

BAB 1 PENDAHULUAN

Pada bab ini adalah pendahuluan yang berisi latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, dan sistematika penulisan dalam skripsi ini.

BAB 2 LANDASAN TEORI

Pada bab ini yaitu membahas beberapa teori-teori penunjang yang akan mendukung pembahasan pada bab-bab selanjutnya.

BAB 3 ANALISIS DAN PERANCANGAN

Pada bab ini akan membahas mengenai analisis permasalahan dalam pembuatan Sistem serta menjelaskan tentang rancangan struktur program serta merancang *interface* dari sistem yang akan dibangun.

BAB 4 IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab ini membahas implementasi dari Sistem yang telah dibangun. Yaitu berisikan gambaran antarmuka dari Sistem selain itu juga dilakukan pengujian untuk melihat perangkat lunak yang dibuat berhasil dijalankan atau tidak serta untuk menemukan kesalahan (*error*) dan *debug program*.

BAB 5 KESIMPULAN DAN SARAN

Pada bab ini berisikan kesimpulan dari penelitian yang telah dilakukan beserta saran dari peneliti yang bisa digunakan sebagai acuan untuk penelitian selanjutnya.

BAB 2

LANDASAN TEORI

2.1 Sistem Pakar

Sistem pakar merupakan bagian dari bidang ilmu kecerdasan buatan. Sistem pakar memanfaatkan pengalaman dan pengetahuan yang dimiliki oleh pakar sebagai dasar atau landasan sebuah sistem untuk menyimpulkan suatu permasalahan. Informasi yang di input biasanya berupa gejala-gejala awal yang dialami oleh *user*. Sistem pakar dapat diterapkan di berbagai bidang seperti kesehatan, pendidikan, otomotif, dan industri. Pada dasarnya sistem pakar digunakan untuk menyelesaikan suatu permasalahan dengan memanfaatkan sebuah perangkat komputer atau *smartphone* yang diimplementasikan ke dalam bentuk program. Program yang akan dibuat nanti tidak hanya semata mata berdasarkan ilmu teknologi saja, tetapi juga dilandaskan oleh ilmu pakar dari masing-masing bidang yang akan dibuatkan programnya. Sistem pakar juga tidak hanya bisa mendapatkan *output* berupa benar dan salah, tetapi bisa juga berupa memberikan penjelasan atau bahkan solusi dari permasalahan tersebut. Sistem pakar juga menjadi salah satu bidang ilmu kecerdasan buatan manusia yang sangat berkembang pesat dan di gandrungi. Sistem pakar bisa dibuat dalam bentuk *website* maupun *mobile*. Namun belakangan ini sistem pakar dalam bentuk *website* lebih diminati karena tidak perlu men-download aplikasi yang ingin digunakan dan menghemat memori ponsel pengguna.

Menurut E. Fraim Turban (1992) Menjelaskan bahwa sistem pakar adalah sebuah program yang mengkomputerisasikan laporan yang mencoba untuk menirukan proses pemikiran dan pengetahuan dari pakar-pakar dalam menyelesaikan masalah.

Buatan manusia ini juga tidak luput dari kelebihan dan kekurangan. Berikut kelebihan dan kekurangan dari Sistem Pakar

2.1.1 Kelebihan Sistem Pakar

Beberapa kelebihan sistem pakar yaitu :

1. Dapat membantu dalam pengambilan keputusan dan memprediksi penyebab serta akibat dari sebuah situasi tertentu.
2. Sebagai edukasi bagi masyarakat awam untuk mendapatkan pengetahuan seorang pakar.
3. Efisiensi waktu dalam penyelesaian masalah
4. Meningkatkan kualitas dan produktivitas kerja.
5. Dalam penggunaannya, tidak diperlukan biaya.
6. Sistemnya dapat dikembangkan dan diperbanyak sesuai dengan kebutuhan.

2.1.2 Kekurangan Sistem Pakar

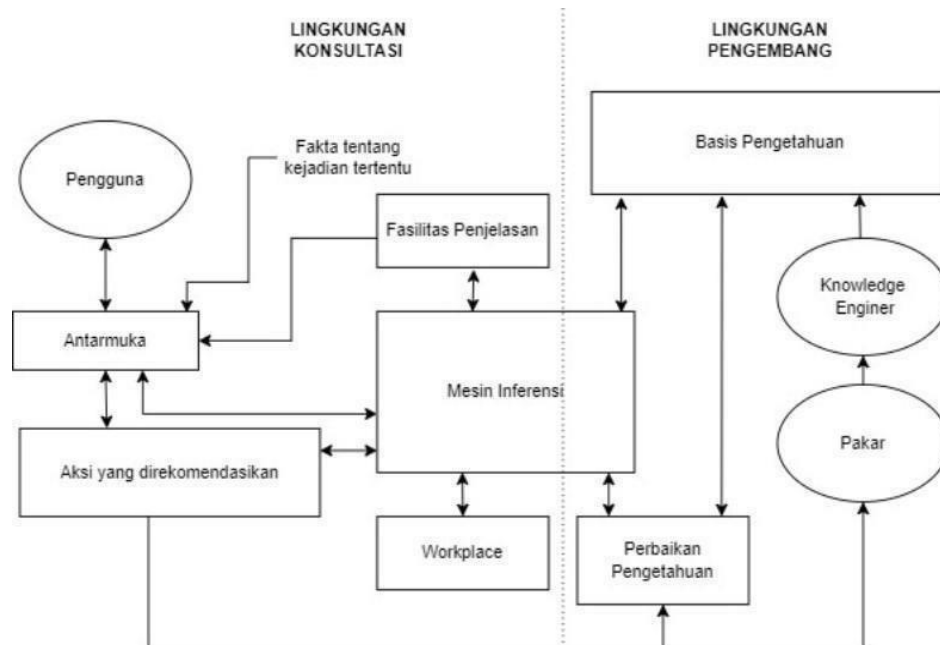
1. Hasil dari sistem perlu diuji ulang secara teliti sebelum digunakan.
2. Dalam pengembangan sistemnya, diperlukan biaya yang cukup besar.
3. Pendekatan setiap pakar dapat berbeda-beda di dalam suatu situasi yang sama.

2.1.3 Ciri-Ciri Sistem Pakar

1. Dapat memberikan penalaran untuk data yang tidak pasti.
2. Dapat mengemukakan rangkaian alasan-alasan yang diberikan dengan cara yang bisa dipahami.
3. Berdasarkan aturan atau rule tertentu.
4. Dirancang untuk dapat dikembangkan secara bertahap agar bisa menghasilkan informasi yang lebih baik dan akurat.
5. Pengetahuan dan mekanisme penalaran jelas terpisah.
6. Keluarnya bersifat anjuran.

2.1.4 Struktur Sistem Pakar

Untuk membangun sistem pakar, komponen yang harus dimiliki adalah antarmuka pengguna (*user interface*), basis pengetahuan (*knowledge base*), mesin inferensi untuk melakukan penalaran untuk menghasilkan penalaran, dan papan tulis (*workplace*) untuk menyimpan fakta-fakta yang diperoleh bersama seorang pakar (Turban, 1995). Komponen-komponen ini disampaikan pula pada gambar 2.1 di bawah.



Gambar 2.1 Struktur Sistem Pakar (Turban, 1995)

1. Basis Pengetahuan (*Knowledge Base*)

Basis pengetahuan merupakan data-data dan aturan-aturan yang diperoleh dari seorang pakar yang nantinya digunakan oleh sistem untuk dapat menyimpulkan suatu permasalahan.

2. Mesin Inferensi (*Inference Engine*)

Mesin inferensi digunakan untuk mengolah fakta-fakta yang diberikan pengguna dan mencari keterkaitan antara fakta-faktor tersebut dengan basis pengetahuan untuk dapat menyimpulkan atau menyelesaikan suatu permasalahan.

3. Fasilitas Penjelasan (*Explanation Facility*)

Fasilitas penjelasan merupakan kemampuan sistem untuk menjelaskan solusi dari suatu permasalahan kepada pengguna.

4. Antar Muka Pengguna (*User Interface*)

User interface merupakan tampilan dalam bentuk visual sebagai interaksi antara pengguna dengan sistem.

5. Memori Kerja (*Working Memory*)

Memori kerja berisi basis data dan fakta yang digunakan oleh aturan.

6. Perbaikan pengetahuan (*Knowledge Refining*)

Perbaikan pengetahuan digunakan untuk meningkatkan kemampuan dengan mengevaluasi kinerja sistem pakar. Kemampuan seorang pakar dalam meningkatkan kinerja pembelajaran dapat diterapkan dalam sistem pakar.

2.2 Penyakit mulut dan kuku

Penyakit mulut dan kuku (PMK) merupakan penyakit yang sangat menular penyakit virus yang disebabkan oleh keluarga Aphthovirus Picornaviridae. (Beyan,dkk, 2020)

Sifat penyakit mulut dan kuku (PMK) yang sangat menular merupakan cerminan dari serangkaian faktor, antara lain sebagai berikut:

- berbagai macam hewan berkuku belah yang rentan
- sejumlah besar virus yang dilepaskan oleh hewan yang terinfeksi
- kisaran ekskresi dan sekresi yang dapat menular
- banyaknya rute infeksi
- dosis kecil virus yang dapat memulai infeksi
- Stabilitas moderat virus di lingkungan. (Donaldson, 2001)

2.3 Case Based Reasoning

Case Based Reasoning (CBR) menyelesaikan masalah dengan cara memanfaatkan pengalaman dari kasus-kasus serupa yang pernah terjadi di masa lalu. Metode ini bertujuan untuk menemukan solusi untuk masalah baru dengan cara mencocokkan karakteristik masalah baru dengan kasus-kasus yang telah ada sebelumnya, lalu mengekstrak solusi dari kasus terbaik yang paling relevan dengan masalah baru.

Berikut kelemahan serta kelebihan CBR :

2.3.1 Kelebihan Case Based Reasoning :

1. Memecahkan masalah dengan mudah
2. Semakin banyak pengalaman, sistem akan menjadi semakin pintar sehingga dapat memecahkan masalah dengan mudah

2.3.2 Kekurangan Case Based Reasoning :

1. Tidak menjamin solusi terbaik atau optimum karena penalaran ini berdasarkan kasus lampau, jika solusi lampau salah maka tahapan revise sangat diperlukan untuk mengurangi tingkat kesalahannya.
2. Semakin banyak pengalaman, proses pencarian semakin lama karena harus membandingkan dengan kasus yang paling mirip.

Case Based Reasoning terdiri dari empat tahapan yaitu:

1. *Retrieve*, merupakan proses penelusuran kasus yang paling mirip dengan kasus baru yang akan dievaluasi.
2. *Reuse*, merupakan proses menggunakan kembali informasi atau pengetahuan yang telah tersimpan pada basis kasus untuk memecahkan masalah baru.
3. *Revise*, merupakan proses meninjau kembali informasi yang akan diproses dan diperbaiki kembali untuk mengatasi kesalahan-kesalahan yang terjadi pada permasalahan baru.
4. *Retain*, merupakan proses menyimpan pengetahuan yang nantinya akan digunakan untuk memecahkan masalah ke dalam basis kasus yang ada.

Similarity atau kemiripan adalah langkah yang digunakan untuk mengenali kesamaan atau kemiripan antara kasus-kasus yang tersimpan dalam basis kasus dengan kasus yang baru. Kasus dengan nilai kemiripan paling besar dianggap sebagai kasus yang paling mirip, nilai 0 artinya tidak mirip dan nilai 1 dianggap kasus mutlak mirip. Berikut merupakan rumus untuk mencari nilai kemiripan :

$$Similarity = \frac{S_1 \times W_1 + S_2 \times W_2 + \dots + S_n \times W_n}{W_1 + W_2 + \dots + W_n}$$

Keterangan

S = Similarity atau nilai kemiripan yaitu 1 (mirip) dan 0 (tidak mirip)

W = Weight atau bobot kemiripan (Parwan,dkk, 2022)

2.4 Certainty Factor

Menurut T.Sutojo dalam Aldino Moto, 2010 certainty factor merupakan suatu metode untuk membuktikan ketidakpastian pemikiran seorang pakar, dimana untuk mengakomodasi hal tersebut seseorang biasanya menggunakan certainty factor untuk menggambarkan tingkat keyakinan pakar terhadap masalah yang sedang dihadapi. *Certainty Factor* (CF) merupakan suatu metode untuk membuktikan apakah suatu fakta itu pasti atau tidak pasti. Metode ini sangat cocok untuk sistem pakar yang mendiagnosis sesuatu yang belum pasti. CF menggunakan suatu nilai untuk mengasumsikan derajat keyakinan pemikiran (*inexact reasoning*) seorang pakar terhadap suatu nilai. Metode ini menggunakan perhitungan berdasarkan kemiripan yang dibagi dengan bobot yang telah ditentukan. Metode CF menunjukkan suatu ukurankepastian terhadap suatu fakta atau aturan. Cara kerja metode *Certainty Factor* ini adalah dengan menunjukkan ukuran kepastian terhadap suatu fakta atau aturan. Metode CF melakukan penalaran layaknya seorang pakar, dan untuk mendapatkan nilai kepercayaan (Santi, 2019).

Certainty Factor berkisar antara -1 hingga +1, dengan beberapa nilai antara -1 dan 1 yang memiliki makna khusus:

1. Jika Certainty Factor = 1, maka pernyataan atau rekomendasi itu pasti benar berdasarkan pengetahuan yang ada.
2. Jika Certainty Factor = 0, maka pernyataan atau rekomendasi itu tidak memiliki pengaruh atau bukti yang cukup untuk mendukung atau menentangnya. Ini bisa diartikan sebagai ketidakpastian.
3. Jika Certainty Factor = -1, maka pernyataan atau rekomendasi itu pasti salah berdasarkan pengetahuan yang ada.

Adapun kelemahan dan kelebihan Certainty Factor sebagai berikut:

2.4.1 Kelebihan Certainty Factor

1. Metode ini cocok dipakai dalam sistem pakar untuk mengukur sesuatu apakah pasti atau tidak pasti dalam mendiagnosis penyakit sebagai salah satu contohnya.
2. Perhitungan dengan menggunakan metode ini dalam sekali hitung hanya dapat mengolah 2 data saja sehingga keakuratan data dapat terjaga.

2.4.2 Kekurangan Certainty Factor

1. Ide umum dari pemodelan ketidakpastian manusia dengan menggunakan metode CF biasanya diperdebatkan. Sebagian orang akan membantah pendapat bahwa formula untuk metode CF diatas memiliki sedikit kebenaran.
2. Metode ini hanya dapat mengolah ketidakpastian/kepastian hanya 2 data saja. Perlu dilakukan beberapa kali pengolahan data untuk data yang lebih dari 2 buah.

Dalam *Certainty Factor* dikenal beberapa premis atau rule yang digunakan untuk mendiagnosa diantaranya sebagai berikut :

- a. CF untuk kaidah dengan premis atau gejala tunggal (*Single premis rles*) :

$$CF[H,E] = CF[user] \times CF[pakar]$$

Keterangan :

$CF[H,E]$ = Faktor Kepastian

$CF[E]$ = Faktor Kepastian *evidence*/fakta/bukti

$CF[rule]$ = Faktor Kepastian hipotesis/dugaan

- b. Apabila terdapat kaidah dengan kesimpulan yang serupa (*similarity concluderules*) atau lebih dari satu gejala, maka CF selanjutnya dihitung dengan persamaan:

$$CF_{combine}[CF1,CF2] = CF1 + CF2 \times (1 - CF1)$$

- c. Faktor kepastian hasil akhir perbandingan:

$$\text{Rasio Kepercayaan} = CF_{combine} \times 100\%$$

BAB 3

ANALISIS DAN PERANCANGAN SISTEM

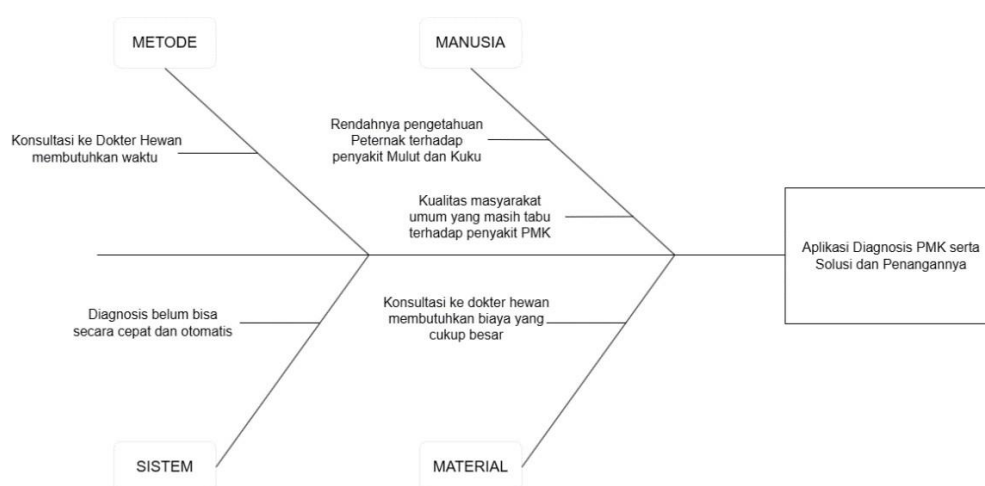
3.1 Analisis Sistem

Analisis sistem merupakan tahapan penelitian terhadap sistem berjalan yang bertujuan untuk mengetahui segala permasalahan yang terjadi serta memudahkan dalam mengidentifikasi dan menjalankan tahap selanjutnya yaitu tahap perancangan sistem. Pada tahap ini, analisis masalah dilakukan untuk mengidentifikasi sebab dan akibat dari suatu masalah yang merupakan latar belakang pembuatan sistem.

3.1.1 Analisis Masalah

Sistem pakar dibangun untuk mempresentasikan kemampuan atau keahlian seorang pakar pada bidang tertentu untuk membantu *user* atau orang yang awam terhadap bidang ilmu tersebut dalam memecahkan suatu permasalahan.

Agar lebih memudahkan dalam menganalisis permasalahan dalam penelitian ini, penulis menggunakan diagram ishikawa. Pada Gambar 3.1 terdapat Diagram Ishikawa (Fishbone Diagram) yang akan menjelaskan akar dari suatu permasalahan dalam 4 faktor yaitu, metode, manusia, sistem, dan material yang digambarkan sebagai berikut.



Gambar 3.1 Diagram Ishikawa

3.1.2 Analisis Kebutuhan

Analisis kebutuhan dilakukan untuk mengidentifikasi hal-hal yang dibutuhkan selama merancang sistem. Terdapat 2 jenis kebutuhan yang dapat diidentifikasi dalam perancangan sistem yaitu, kebutuhan fungsional dan kebutuhan non-fungsional.

1. Kebutuhan Fungsional

Kebutuhan Fungsional merupakan kebutuhan yang harus dilakukan dan berfungsi pada sebuah sistem aplikasi untuk mencapai tujuan dari aplikasi tersebut. Dalam aplikasi yang dibangun tersebut, kebutuhan fungsional yang harus terpenuhi adalah sebagai berikut:

- a. Sistem yang dibangun dapat menampilkan info singkat tentang Penyakit mulut dan kuku
- b. Sistem yang dibangun dapat menampilkan daftar gejala.
- c. Sistem menampilkan jawaban dimana terdapat empat pilihan dan dapat dipilih sesuai dengan yang dirasakan oleh *user*

2. Kebutuhan Non Fungsional

Kebutuhan non fungsional merupakan kebutuhan yang tidak berhubungan langsung dengan sistem dan bertujuan sebagai pelengkap atau tambahan yang dapat mendukung kinerja dari sistem yang akan dibangun. Berikut beberapa kebutuhan non-fungsional:

- a. Tampilan Antarmuka (Interface)
Memiliki tampilan yang sederhana dan *easy to use* sehingga akan mempermudah pengguna untuk menggunakannya.
- b. Performa
Menggunakan Bahasa pemrograman PHP dan database MySQL.
- c. Efisiensi
Tidak membutuhkan akun saat menggunakan *website*.

d. Kualitas

Sistem menerapkan uji persentase dengan Algoritma Certainty Factor sehinggadata yang dimasukkan atau dipilih oleh *user* akan diberi solusi dan penanganan

e. Kontrol

Kontrol sistem sangat mudah dipahami karena hanya berupa dua opsi dari pertanyaan yang ada.

3.1.3 Analisis Proses

Analisis proses adalah mengurai kembali atau menjelaskan kembali proses yang akan dibuat atau dibangun pada sistem yang sedang dibangun. Mulai dari mencari data atau informasi, cara kerja sistem, cara peng-implementasiannya dan juga metode yang digunakan. Pada penelitian ini, sistem akan dibangun dengan mengimplementasikan *Case Based Reasoning* dan *Certainty Factor*. Dimana setelah proses pada CBR dilakukan *similarity* dengan CF antara kasus terdahulu dengan kasus yang sekarang. Hasil dapatdijadikan sebagai acuan dalam mendiagnosis Penyakit mulut dan kuku yang sesuai dengan pilihan yang dipilih oleh *user*. Adapun proses kerja sistem dalam penelitian adalah sebagai berikut.

1. Pengumpulan Data

Tahap pengumpulan data pada penelitian ini digunakan untuk mengambil informasi mengenai bobot kondisi pada Tabel 3.1 dan gejala pada Tabel 3.2 yang akan digunakan sebagai acuan bobot untuk mendiagnosis Penyakit mulut dan kuku yang di dapat dari pakar . Informasi yang didapatkan adalah sebagai berikut :

Tabel 3.1 Bobot Kondisi Penyakit Mulut dan Kuku

| Kode kondisi | Nama kondisi | Bobot kondisi |
|--------------|----------------|---------------|
| K1 | Parah/sering | 1 |
| K2 | Sedang/sedikit | 0.5 |
| K3 | Tidak | 0 |

Tabel 3.2 Gejala-gejala Penyakit mulut dan kuku

| Kode gejala | Nama gejala | CF Pakar |
|-------------|------------------------------------|----------|
| G01 | Suhu tubuh tinggi >39,2 Celcius | 0,4 |
| G02 | Sapi mengalami lesu | 0,5 |
| G03 | Nafsu makan berkurang | 0,5 |
| G04 | Kemampuan berjalan berkurang | 0,6 |
| G05 | Sapi tidak lagi dapat berdiri | 0,6 |
| G06 | Terdapat lepuh pada kaki dan mulut | 0,8 |
| G07 | Lepuh pada kaki dan mulut pecah | 0,8 |
| G08 | Produksi susu menurun | 0,5 |
| G09 | Air liur melimpah | 0,6 |

Tabel 3.3 Solusi dan Penanganan Penyakit mulut dan kuku

| Kode Penyakit | Kode Solusi | Solusi dan Penanganan |
|---------------|-------------|---|
| P01 | S01 | Berikan pakan yang mudah dikonsumsi dan pastikan sapi terinfeksi tetap terhidrasi dengan memberikan air yang bersih dan segar. Pemberian obat pereda demam, vitamin, dan dilakukan vaksinasi. |
| P02 | S02 | Cuci dan obati luka-luka pada mulut dan kaki sapi dengan larutan antiseptik, Berikan obat pereda demam, antibiotik, obat anti inflamasi. |
| P03 | S03 | Berikan obat pereda demam, antibiotik, obat anti inflamasi, serta obat pencegah infeksi sekunder atau mencegah datangnya lalat. |

2. Menentukan Dataset

Berdasarkan informasi gejala dan bobot kondisi, maka didapatkan 12 total atribut untuk penelitian ini. Selanjutnya, pada tahap ini akan dilakukan pengelompokan *dataset* yang terdiri dari 6 gejala kasus, dan 3 bobot kondisi penyakit terdapat pada Tabel 3.4 dalam penentuan tingkatan Penyakit mulut dan kuku dengan rincian berikut.

Tabel 3.4 Dataset Penyakit mulut dan kuku

| Pasien | G1 | G2 | G3 | G4 | G5 | G6 | G7 | G8 | G9 | Hasil |
|--------|--------|--------|--------|-------|-------|-------|-------|-------|-------|-------|
| 1 | sering | sering | sering | tidak | tidak | tidak | tidak | tidak | tidak | P1 |
| 2 | sedang | sedang | sedang | tidak | tidak | tidak | tidak | tidak | tidak | P1 |
| 3 | parah | parah | parah | parah | parah | parah | tidak | parah | parah | P2 |
| 4 | parah | parah | parah | parah | parah | tidak | parah | parah | parah | P3 |
| 5 | parah | parah | parah | parah | parah | parah | parah | parah | parah | P3 |

Contoh perhitungan manual CBR dan CF :

Berikut adalah contoh perhitungan manual dimana terdapat data gejala Penyakit mulut dan kuku pada Tabel 3.5 dan dan Inputan user pada Tabel 3.6 di setiap gejala yangselanjutnya akan menjadi dasar perhitungan CBR dan CF.

Tabel 3.5 Gejala Penyakit mulut dan kuku

| Kode gejala | Nama gejala | CF Pakar |
|-------------|------------------------------------|----------|
| G01 | Suhu tubuh tinggi >39,2 Celcius | 0,4 |
| G02 | Sapi mengalami lesu | 0,5 |
| G03 | Nafsu makan berkurang | 0,5 |
| G04 | Kemampuan berjalan berkurang | 0,6 |
| G05 | Sapi tidak lagi dapat berdiri | 0,6 |
| G06 | Terdapat lepuh pada kaki dan mulut | 0,8 |
| G07 | Lepuh pada kaki dan mulut pecah | 0,8 |
| G08 | Produksi susu menurun | 0,5 |
| G09 | Air liur melimpah | 0,6 |

Tabel 3.6 Contoh Inputan User

| Gejala yang di input | Nama gejala | Bobot User |
|----------------------|------------------------------------|------------|
| G01 | Suhu tubuh tinggi >39,2 Celcius | 1 |
| G02 | Sapi mengalami lesu | 1 |
| G03 | Nafsu makan berkurang | 1 |
| G04 | Kemampuan berjalan berkurang | 1 |
| G05 | Sapi tidak lagi dapat berdiri | 1 |
| G06 | Terdapat lepuh pada kaki dan mulut | 1 |
| G07 | Lepuh pada kaki dan mulut pecah | 1 |
| G08 | Produksi susu menurun | 1 |
| G09 | Air liur melimpah | 1 |

Langkah Penyelesaian :

1. Proses *Retrieve* merupakan tahapan awal untuk mengenali masalah kasus baru dengan kasus yang sudah ada atau kasus lama.
2. Setelah selesai proses *retrieve*, kemudian masuk ke tahap kedua, *reuse*. Untuk mencari nilai kemiripan diberikan keyakinan yang masing-masing memiliki nilai. Jika memiliki kecocokan diberikan nilai 1, dan jika tidak memiliki kecocokan maka diberikan nilai 0.
3. Kemudian menghitung nilai kemiripan kasus lama dengan kasus baru dengan rumus *similarity*.

$$\frac{1 \times 1 + 1 \times 1 + 1 \times 1 + 1 \times 1 + 1 \times 1 + 1 \times 1 + 1 \times 1 + 1 \times 1 + 1 \times 1}{1 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 1}$$

$$= \frac{9}{9}$$

$$= 1$$

Dari perhitungan diatas, didapat hasilnya bernilai 1, dimana ini menunjukkan hasil dari tingkat kondisinya adalah parah.

4. Pada proses *reuse*, solusi yang dapat diberikan adalah dengan bobot kemiripan paling tinggi, yaitu P03 dengan bobot kondisi parah
5. Proses perhitungan dari gejala kasus baru untuk mendapatkan nilai kepastian terlihat pada Tabel 3.7 dengan menggunakan formulasi rumus berikut:

$$CF_{[H,E]} = CF[user] \times CF[pakar]$$

Tabel 3.7 Perhitungan Nilai Kepastian

| | | | | | |
|----------------------|-----|---|---|---|-----|
| CF[H,E] ₁ | 0,4 | x | 1 | = | 0,4 |
| CF[H,E] ₂ | 0,5 | x | 1 | = | 0,5 |
| CF[H,E] ₃ | 0,5 | x | 1 | = | 0,5 |
| CF[H,E] ₄ | 0,6 | x | 1 | = | 0,6 |
| CF[H,E] ₅ | 0,6 | x | 1 | = | 0,6 |
| CF[H,E] ₆ | 0,8 | x | 1 | = | 0,8 |
| CF[H,E] ₇ | 0,8 | x | 1 | = | 0,8 |
| CF[H,E] ₈ | 0,5 | x | 1 | = | 0,5 |
| CF[H,E] ₉ | 0,6 | x | 1 | = | 0,6 |

6. Selanjutnya proses mengkombinasikan nilai CF dengan rumus berikut:

$$CF \text{ Combine } [CF1, CF2] = CF1 + CF2 \times (1 - CF1)$$

$$CF[H,E]_1 = 0,4 + 0,5 \times (1 - 0,9) = 0,45$$

$$CF[H,E]_{old1,3} = 0,5 + 0,45 \times (1 - 0,5) = 0,725$$

$$CF[H,E]_{old2,4} = 0,6 + 0,725 \times (1 - 0,6) = 0,89$$

$$CF[H,E]_{old3,5} = 0,6 + 0,89 \times (1 - 0,6) = 0,956$$

$$CF[H,E]_{old4,6} = 0,8 + 0,956 \times (1 - 0,8) = 0,9912$$

$$CF[H,E]_{old5,7} = 0,8 + 0,9912 \times (1 - 0,8) = 0,99825$$

$$CF[H,E]_{old6,8} = 0,5 + 0,99825 \times (1 - 0,5) = 0,99912$$

$$CF[H,E]_{old7,9} = 0,6 + 0,99912 \times (1 - 0,6) = 0,999648$$

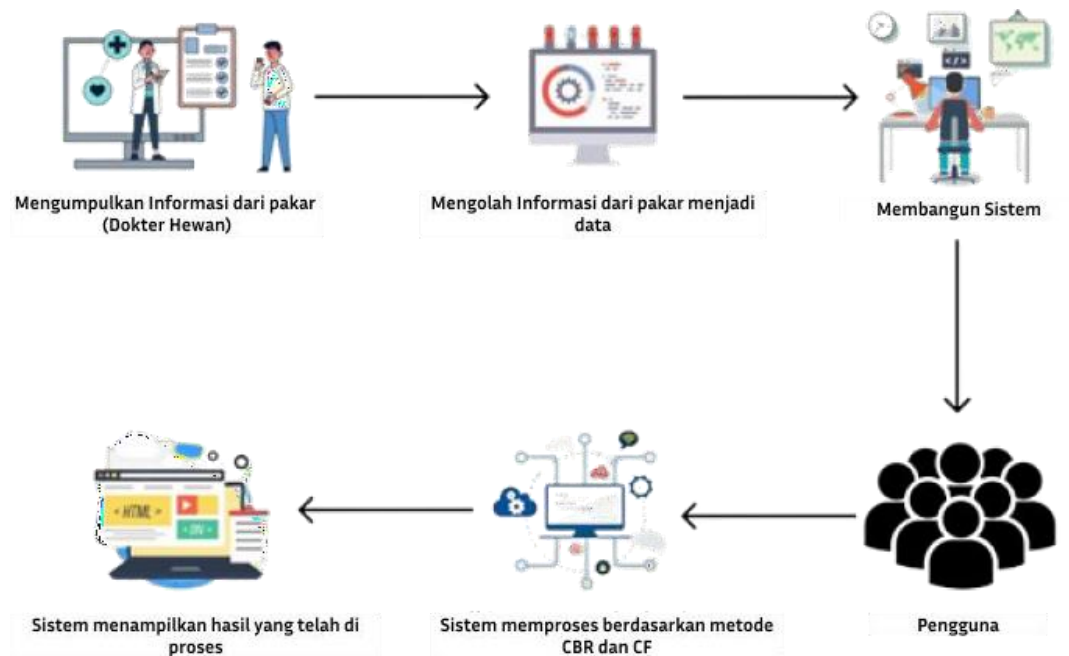
7. Kemudian perhitungan nilai rasio kepercayaan dengan rumus berikut:

$$\begin{aligned} \text{Rasio Kepercayaan} &= CF \text{ Combine} \times 100\% \\ CF \text{ Combine} \times 100\% &= 0,99 \times 100\% \text{ (Parah)} \end{aligned}$$

Hasil dari konsultasi studi kasus diatas menghasilkan **Fase Ulseratif** dengan nilai kemiripan sebesar 100% dan nilai kepastian sebesar 99% (Sangat Yakin). Hasil nilai kemiripan di dapatkan dari kemiripan antara kasus lama dengan kasus baru, sedangkan nilai kepastian di dapat dari perhitungan antara nilai CF yang diberikan pakar dengan nilai CF yang diberikan *user*.

3.2 Arsitektur Umum

Urutan Prosedur dan proses secara umum dalam membuat dan menjalankan sistem pakar diagnosis *Penyakit mulut dan kukud* ditunjukkan pada gambar 3.2 di bawah.



Gambar 3.2 Arsitektur Umum Sistem

Dapat terlihat pada gambar 3.2 diatas, langkah-langkah yang akan dilakukan oleh sistem adalah :

- Mengumpulkan data dan informasi yang berkaitan dengan Penyakit mulut dan kuku dari Seorang pakar, dokter hewan dan sumber terpercaya lainnya.
- Mengolah data dan informasi yang telah didapatkan dari seorang pakar dan sumberlain menjadi dataset
- Merancang dan membangun sistem pakar berbasis *Website* dengan menggunakan *Certainty factor* dan *Case Based-Reasoning*
- Setelah sistem pakar selesai dibangun, *user* dengan kriteria yang sesuai sudah dapat menggunakannya
- Sistem akan memproses inputan *user* berupa gejala-gejala yang telah dipilih dengan metode *Certainty factor* dan *Case Based-Reasoning* untuk mengetahui hasil diagnosis.
- Sistem menampilkan hasil diagnosis yang sesuai dan akurat serta menampilkan solusi dan penanganan yang tepat.

3.3 Pemodelan Sistem

Pemodelan sistem adalah suatu proses yang kompleks dan penting dalam pengembangan sistem, yang melibatkan serangkaian langkah dan metodologi untuk memahami, merancang, dan mengembangkan solusi yang efektif untuk berbagai permasalahan. Tujuan utama dari pemodelan sistem adalah untuk menciptakan representasi yang abstrak dan komprehensif tentang bagaimana suatu sistem seharusnya beroperasi.

Proses pemodelan sistem ini melibatkan serangkaian aktivitas yang sangat penting, dimulai dari pemahaman mendalam terhadap permasalahan yang ingin diselesaikan. Selanjutnya, pemodelan sistem melibatkan analisis kebutuhan sistem, yang mencakup identifikasi dan dokumentasi persyaratan, tujuan, dan fungsi utama yang harus dipenuhi oleh sistem yang akan dibangun. Dalam hal ini, use case diagram digunakan untuk menggambarkan skenario interaksi antara sistem dan pengguna, membantu dalam pemahaman yang lebih baik tentang kasus penggunaan sistem.

Selanjutnya, pemodelan sistem juga mencakup perancangan arsitektur sistem, di mana struktur, komponen, dan hubungan antar mereka dirancang dengan teliti. Activity diagram digunakan untuk menggambarkan alur kerja dan proses di dalam sistem, membantu mengidentifikasi tugas-tugas yang harus dilakukan oleh sistem dan bagaimana alur kerja tersebut berjalan.

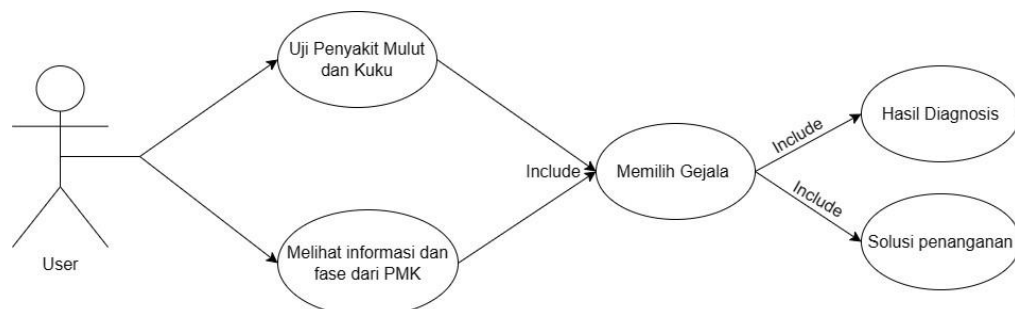
Selain itu, sequence diagram juga merupakan alat yang berguna dalam pemodelan sistem. Diagram ini digunakan untuk memvisualisasikan urutan pesan dan interaksi antar objek atau komponen sistem, membantu dalam pemahaman bagaimana komponen-komponen tersebut saling berhubungan dalam menjalankan fungsi sistem.

Selama proses pengembangan, pemodelan sistem juga dapat digunakan untuk menguji perilaku sistem sebelum tahap implementasi sebenarnya. Ini membantu dalam mengidentifikasi potensi masalah atau kesalahan dalam desain sebelum sistem menjadi operasional.

Jadi, dapat disimpulkan bahwa pemodelan sistem bukan hanya sekadar penciptaan gambaran visual, tetapi juga merupakan fondasi utama dalam pengembangan sistem yang sukses. Ini membantu tim pengembang untuk memiliki pemahaman yang mendalam tentang sistem yang akan dibangun, memastikan bahwa persyaratan dan desain sistem terpenuhi dengan baik, dan mengurangi risiko kesalahan dalam implementasi akhirnya.

3.3.1 Use Case Diagram

Diagram *Use Case* adalah gambaran grafis dari beberapa atau semua *user*, *use case*, dan interaksi diantaranya yang memperkenalkan suatu sistem dimana terdapat bagan interaksi antara pengguna dengan sistem yang dibangun. Diagram ini bertujuan untuk mempermudah proses analisa elemen penting yang dibutuhkan dalam membangun sistem.



Gambar 3.3 Use Case Diagram

Diagram *use case* di atas pada Gambar 3.5 menggambarkan hubungan antara pengguna dengan sistem yang akan dirancang.

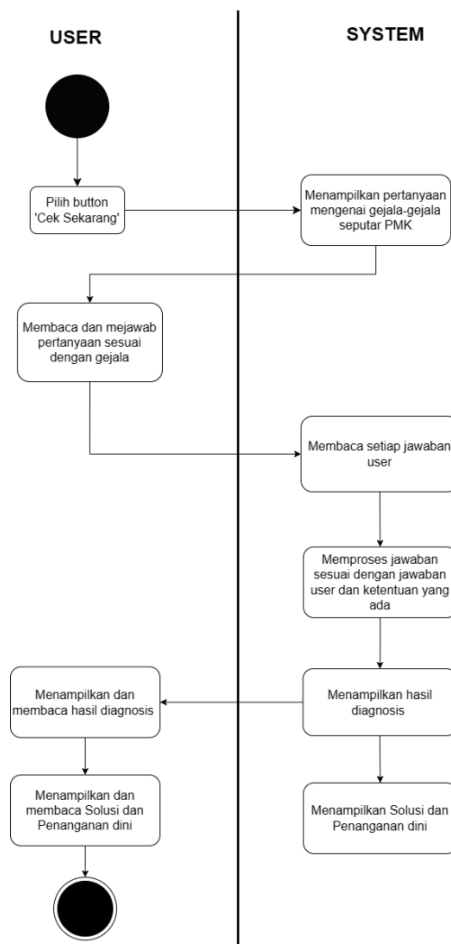
3.3.2 Activity Diagram

Activity Diagram adalah jenis diagram yang digunakan dalam pemodelan sistem untuk menggambarkan proses kerja atau aktivitas sistem secara keseluruhan, dari awal hingga akhir. Diagram ini memberikan representasi visual yang jelas tentang alur kerja sistem atau proses, termasuk tindakan, keputusan, dan aliran aktivitas yang terjadi selama proses tersebut.

Beberapa elemen penting dalam Activity Diagram meliputi:

1. Aktivitas (*Activity*): Aktivitas mewakili tugas, langkah, atau tindakan yang harus dilakukan dalam proses. Mereka direpresentasikan oleh bentuk persegi panjang dalam diagram.
2. Transisi (*Transition*): Transisi adalah panah atau anak panah yang menghubungkan aktivitas-aktivitas dalam urutan waktu yang logis. Mereka menunjukkan aliran dari satu aktivitas ke aktivitas berikutnya.
3. Keputusan (*Decision*): Keputusan direpresentasikan oleh berlian dalam diagram. Mereka menunjukkan titik di mana suatu keputusan harus dibuat dalam alur kerja, dan hasil keputusan akan memengaruhi aliran aktivitas yang selanjutnya.
4. Pemilihan (*Merge*): Pemilihan digunakan untuk menggambarkan penggabungan aliran dari beberapa aktivitas yang berbeda menjadi satu. Ini biasanya digunakan setelah keputusan diambil dan beberapa cabang alur kerja perlu digabungkan kembali.
5. Pengulangan (*Loop*): Pengulangan atau iterasi dalam alur kerja direpresentasikan dengan simbol yang menunjukkan perulangan atau pengulangan aktivitas tertentu.
6. Awal dan Akhir (*Start/End*): Awal dan akhir dari alur kerja ditandai dengan simbol oval. Awal menunjukkan titik mulai alur kerja, sedangkan akhir menunjukkan titik di mana alur kerja selesai.

Activity Diagram sangat berguna dalam merancang, mendokumentasikan, dan memahami alur kerja dalam sistem atau proses. Mereka membantu dalam mengidentifikasi tugas-tugas yang harus dilakukan, urutan tindakan yang diperlukan, dan bagaimana keputusan memengaruhi aliran kerja. Diagram ini sering digunakan dalam pengembangan perangkat lunak untuk merinci alur kerja aplikasi atau sistem yang kompleks. Selain itu, *Activity Diagram* juga membantu dalam komunikasi antara tim pengembangan dan pemangku kepentingan untuk memastikan pemahaman yang seragam tentang proses yang sedang dikembangkan atau ditingkatkan.



Gambar 3.4 Activity Diagram

Aktifitas dan interaksi yang terjadi antara pengguna dan sistem digambarkan pada *activity diagram* yang terlihat pada gambar 3.4 di atas.

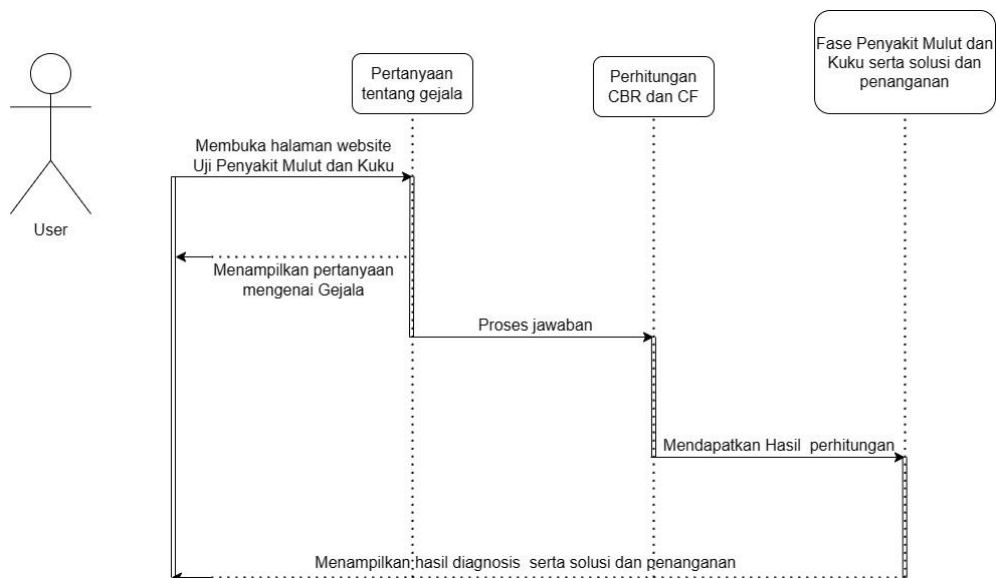
3.3.3 Sequence Diagram

Sequence Diagram adalah jenis diagram dalam pemodelan sistem yang digunakan untuk menggambarkan interaksi antara berbagai objek atau komponen dalam sebuah sistem secara urutan atau sekuensial. Diagram ini memberikan representasi visual tentang bagaimana pesan atau panggilan metode dipindahkan dari satu objek ke objek lain dalam urutan waktu tertentu. *Sequence Diagram* biasanya digunakan dalam pengembangan perangkat lunak dan desain sistem untuk memahami dan merancang komunikasi antar objek dalam sistem tersebut.

Beberapa elemen kunci dalam Sequence Diagram meliputi:

1. Objek: Setiap objek atau entitas dalam sistem direpresentasikan oleh sebuah kotak vertikal di diagram. Objek-objek ini mewakili berbagai komponen dalam sistem, seperti kelas, aktor, atau elemen-elemen lain yang terlibat dalam interaksi.
2. *Lifeline*: Garis vertikal yang menghubungkan objek dengan sekuensial waktu. Lifeline mewakili masa hidup objek selama interaksi dalam diagram.
3. Pesan: Panah horizontal yang menghubungkan objek-objek dan mewakili pesan atau panggilan metode yang dikirimkan dari satu objek ke objek lain. Pesan-pesan ini menunjukkan bagaimana objek-objek berinteraksi satu sama lain.
4. Aktivasi: Kotak kecil di sepanjang garis lifeline yang menunjukkan kapan objek aktif dalam interaksi. Aktivasi menggambarkan interval waktu ketika objek sedang menjalankan operasi atau merespons pesan.
5. Iterasi dan Perulangan: Sequence Diagram juga dapat menggambarkan perulangan atau iterasi dalam interaksi. Ini dapat digunakan untuk menunjukkan bagaimana objek-objek berulang kali berkomunikasi atau menjalankan operasi tertentu.
6. Paralelisme: Diagram ini juga dapat menggambarkan paralelisme ketika beberapa objek berinteraksi secara bersamaan dalam proses yang berbeda.

Sequence Diagram sangat berguna dalam merancang sistem yang melibatkan banyak komponen atau objek yang berinteraksi satu sama lain. Mereka membantu pengembang dan perancang sistem untuk memahami aliran logika dalam sistem, mengidentifikasi potensi masalah atau kebingungan dalam interaksi, dan memastikan bahwa komunikasi antar objek berjalan dengan benar. *Sequence Diagram* juga dapat digunakan untuk dokumentasi yang lebih baik dalam pengembangan perangkat lunak dan membantu dalam komunikasi antara anggota tim pengembangan. *Sequence diagram* dalam proses diagnosa *Penyakit mulut dan kuku* dapat dilihat pada gambar 3.5 dibawah ini.



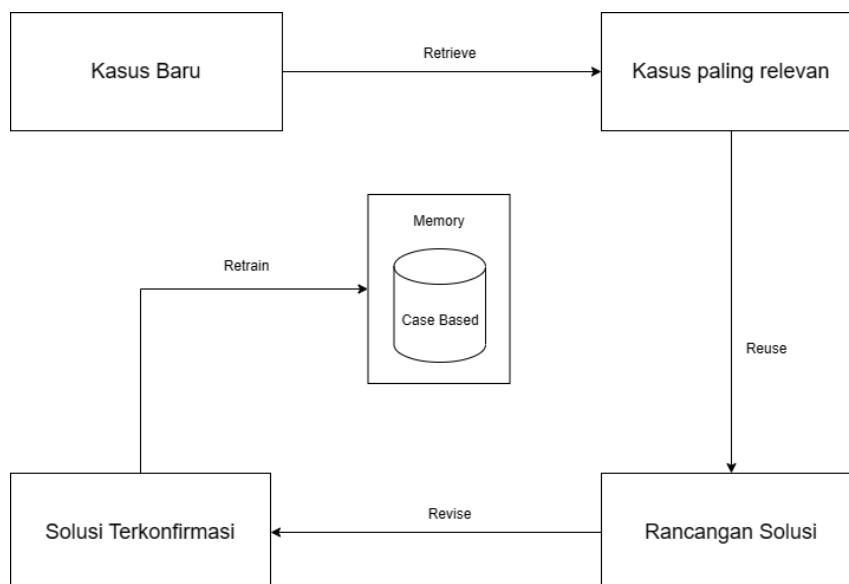
Gambar 3.5 Sequence Diagram

3.4 Flowchart

Flowchart atau diagram alir adalah salah satu jenis diagram grafis yang digunakan untuk menggambarkan aliran atau urutan proses dengan rinci. Diagram ini menggunakan simbol-simbol grafis yang mewakili berbagai tahapan kerja dari suatu sistem atau proses tertentu. Tujuan utama dari flowchart adalah untuk memberikan gambaran visual tentang bagaimana suatu proses atau sistem beroperasi.

3.4.1 Diagram Case Based Reasoning

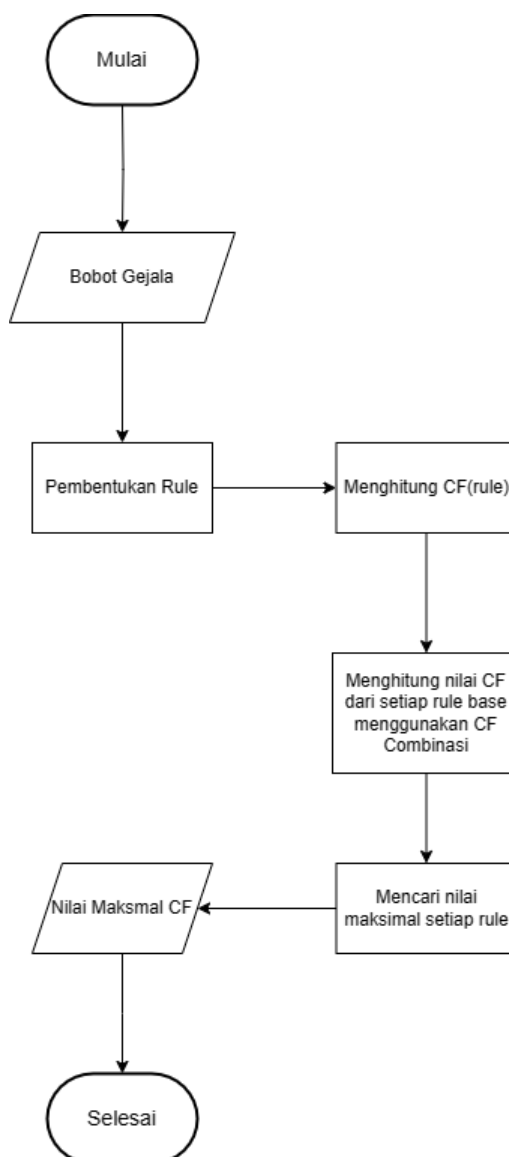
Adapun diagram alir (flowchart) dari tahapan kerja algoritma *Case Based Reasoning* dapat dilihat pada gambar 3.6 berikut ini.



Gambar 3.6 diagram Case Based Reasoning

Pada Gambar 3.6, langkah pertama yang dilakukan adalah menemukan kasus baru. Dalam langkah ini data akan disimpan ke dalam basis kasus data untuk digunakan dimasa depan. Untuk masuk ketahap selanjutnya, digunakan tahap *retrieve* yaitu mencari dan mengambil kasus-kasus yang relevan atau hampir sama dari basis kasus yang ada. Basis kasus berisi pengalaman sebelumnya yang disimpan dalam bentuk pengetahuan yang dapat diakses. Pada tahap *Retrieve*, sistem CBR mencari kasus-kasus serupa yang telah diambil dari basis kasus berdasarkan kesamaan dengan masalah yang sedang dihadapi. Setelah itu, untuk masuk ketahap rancangan solusi, digunakanlah *reuse*, dimana setelah kasus-kasus relevan telah ditemukan, pada tahap ini akan memberikan solusi atau pengetahuan yang terdapat dalam kasus tersebut. Sistem akan melakukan modifikasi atau *revise* terhadap solusi yang ada untuk membuatnya cocok dengan konteks masalah baru. Setelah solusi cocok dan terkonfirmasi, selanjutnya akan masuk tahap *retain*, yaitu sistem akan menyimpan kasus dan solusi yang dapat digunakan di masa depan.

3.4.2 Flowchart Certainty Factor

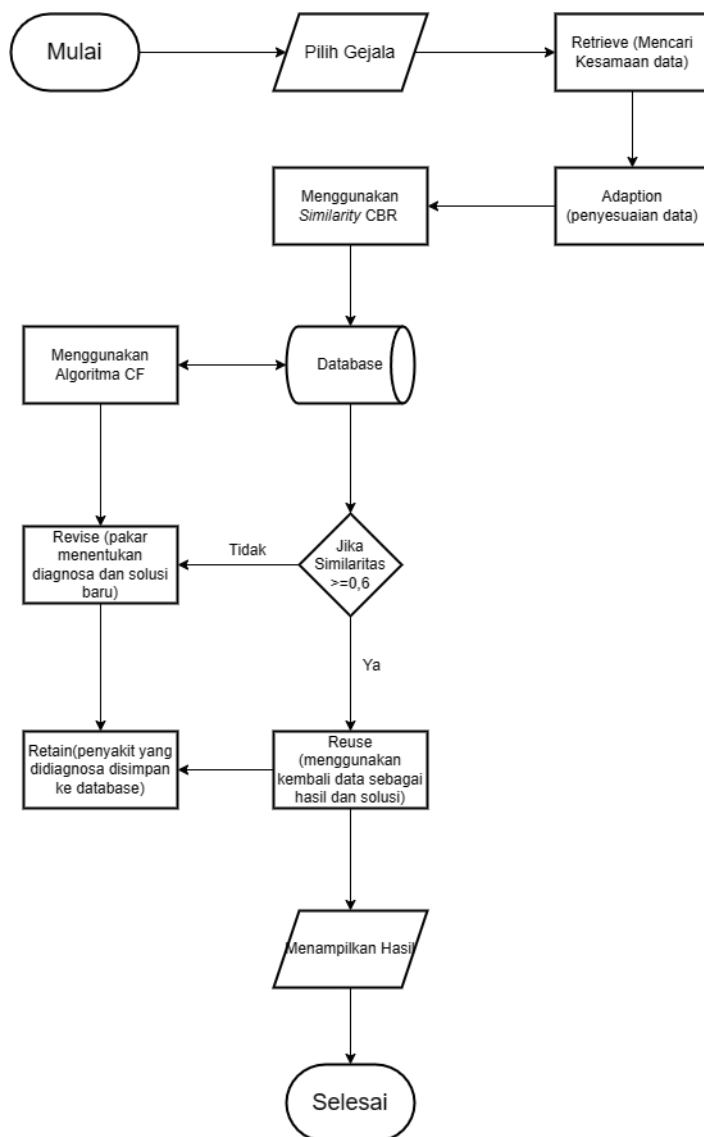


Gambar 3.7 Flowchart Certainty Factor

Gambar 3.7 di atas memperlihatkan diagram alir proses perhitungan menggunakan metode CF. Pada tahap awal akan dilakukan pembuatan rule berdasarkan bobot gejala yang diinput oleh user. Kemudian, akan dihitung rule CF dari setiap rule base sehingga didapatkan nilai maksimal setiap rule, kemudian akan didapatkan nilai maksimal CF.

3.4.3 Flowchart Sistem

Gambar 3.8 di bawah ini merupakan *flowchart* sistem atau cara kerja dari sistem yang telah dibangun.



Gambar 3.8 Flowchart Sistem

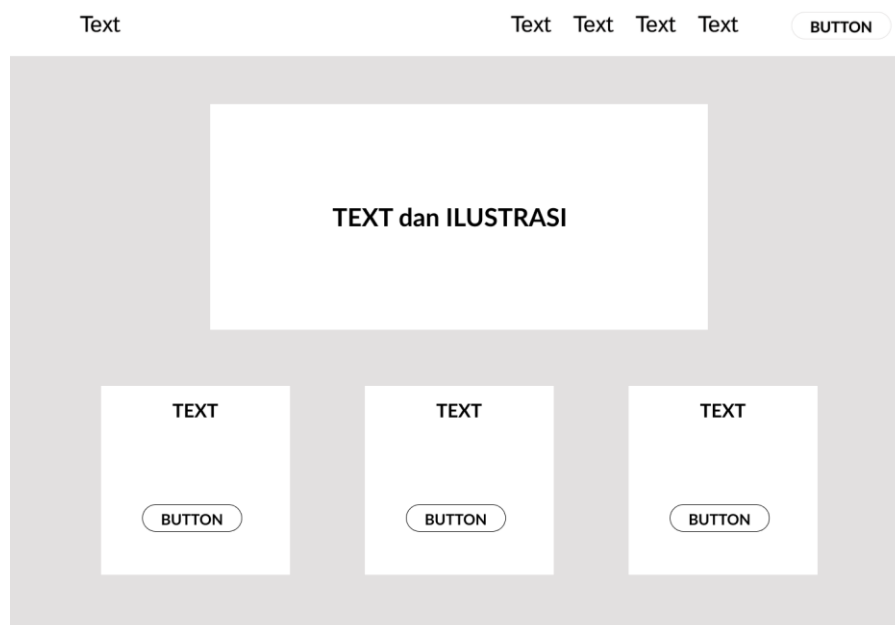
3.5 Perancangan *Interface*

Perancangan *Interface* merupakan salah satu hal yang sangat penting dilakukan sebelum aplikasi dibuat sehingga membentuk aplikasi yang mudah dan nyaman digunakan oleh *user*.

1. Rancangan Halaman Beranda

Halaman beranda terlihat pada gambar 3.9 di bawah, pada halaman beranda *user* bisa melihat ada tiga *card*, yaitu tentang, Informasi Penyakit, dan Konsultasi.

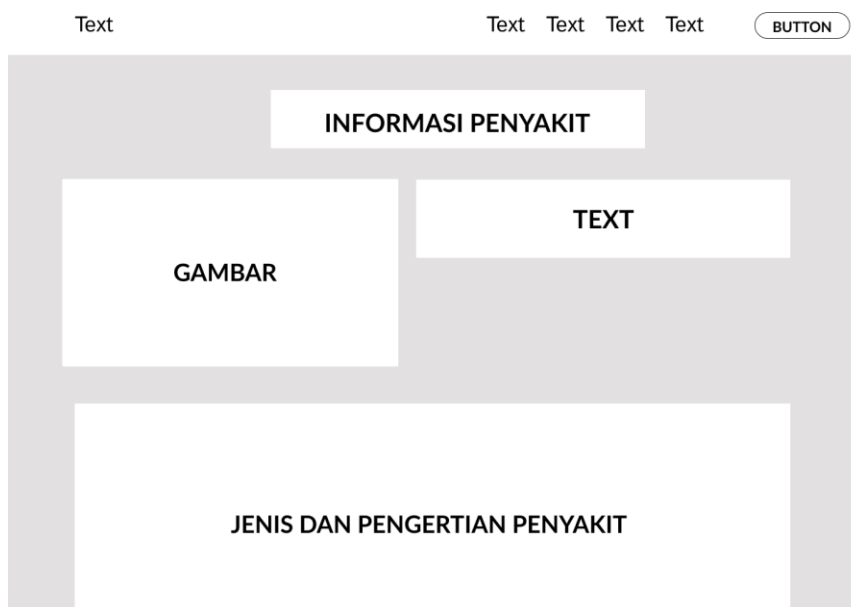
Pada *card* tentang, akan berisi tentang sistem pakar PMK, *card* Informasi Penyakit berisi info tingkatan penyakit mulut dan kuku serta solusi sesuai dengan tingkatannya. Dan di *card* Konsultasi akan langsung mengarah ke halaman Konsultasi dimana berisi *user* akan memilih gejala-gejala dari penyakit mulut dan kuku itu sendiri yang sesuai dengan hewan ternak.



Gambar 3.9 Rancangan Halaman Beranda

2. Rancangan Halaman Info

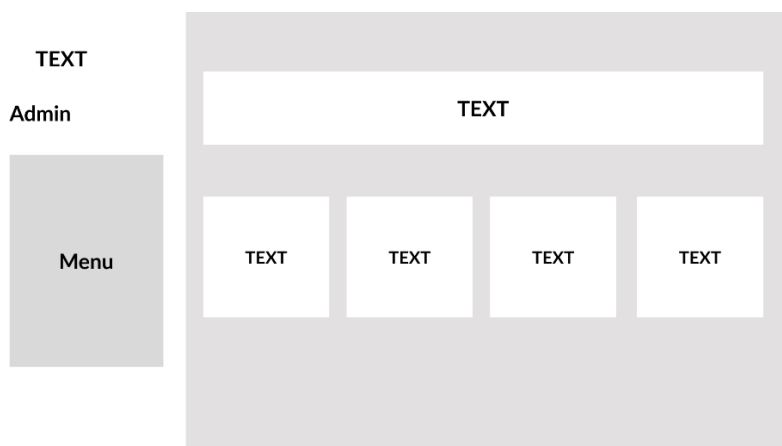
Halaman info berisi informasi singkat mengenai ketiga tingkatan Penyakit mulut dan kuku. Akan terdapat tabel dimana pada tabel ini akan berisi tingkatan penyakit mulut dan kuku serta penjelasan singkat. Rancangan halaman info dapat dilihat pada gambar 3.10 di bawah.



Gambar 3.10 Rancangan Halaman Info

3. Halaman Admin

Pada sistem ini, terdapat halaman admin. Dimana admin dapat menambah, menghapus, mengurangi serta mengedit data-data yang ada di sistem tersebut yang dapat memudahkan dalam pengerjaan sistem. Seperti yang dapat dilihat pada gambar 3.11 dibawah ini.



Gambar 3.11 Rancangan Halaman Admin

4. Halaman Konsultasi

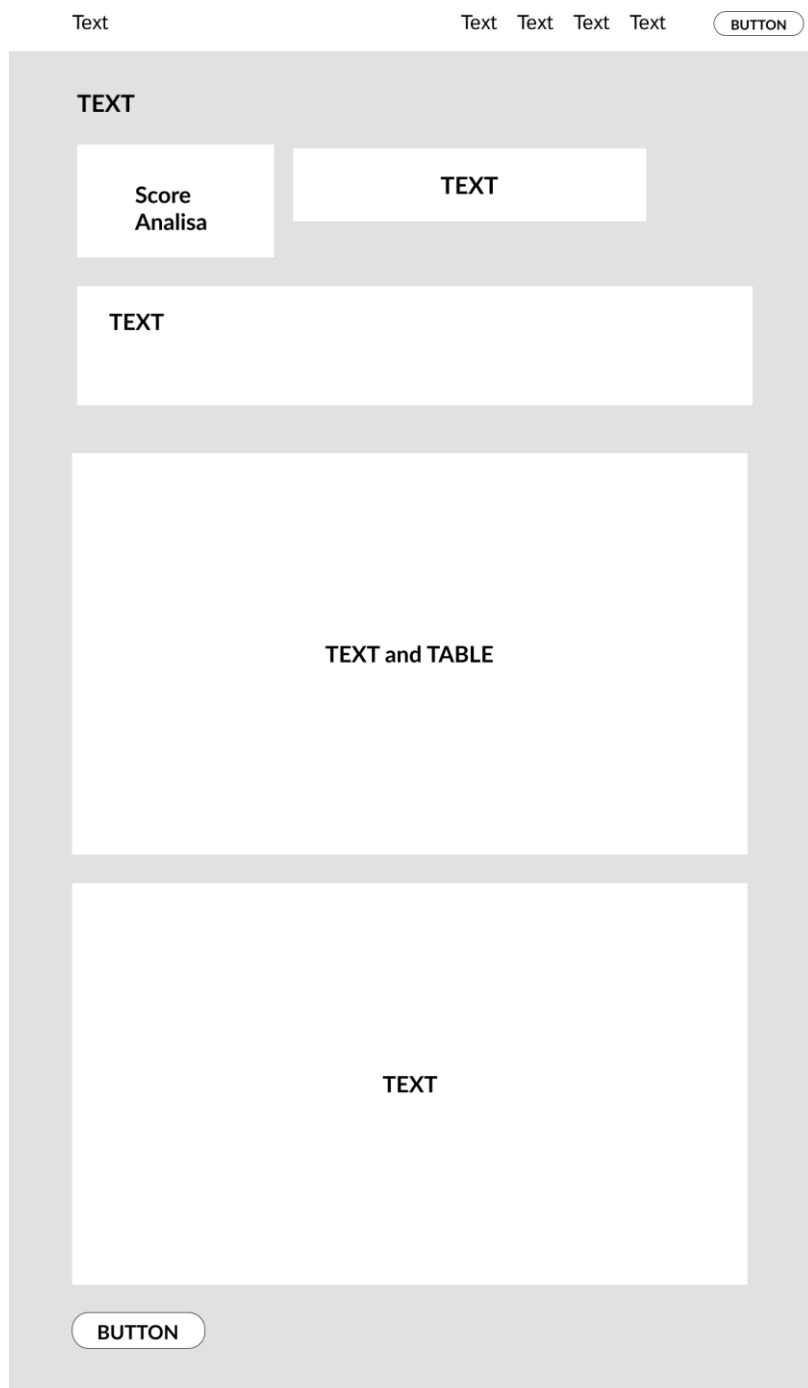
Pada halaman konsultasi yang ditunjukkan pada gambar 3.12, *user* dapat memilih gejala sesuai yang dirasakannya. *User* wajib menjawab semua pertanyaan yang ada, jika *user* tidak mengisi salah satu pertanyaan, maka *user* tidak akan bisa lanjut dan mengetahui hasil akhirnya. Setelah memilih gejala yang sesuai, klik *button* Konsultasi untuk melihat hasilnya.

The screenshot displays a web interface for a consultation page. At the top, there are four 'Text' labels and a 'BUTTON' button. Below this, the main content area is titled 'Text' and contains six identical question blocks. Each block is titled 'PERTANYAAN TENTANG GEJALA' and includes a text input field with the placeholder 'Pilih tingkatan' and a 'V' button. The form is designed to collect user input for a diagnosis.

Gambar 3.12 Halaman Cek

5. Halaman Hasil Diagnosis

Halaman ini merupakan halaman terakhir dari aplikasi yang telah dibuat, dimana pada halaman ini berfungsi untuk menampilkan hasil diagnosis serta solusi sesuai dengan tingkatannya berdasarkan gejala - gejala yang telah diinput oleh pengguna atau *user*. Pada halaman ini juga terdapat persenan dari hasil gejala tersebut serta besaran dari setiap faktor yang ada. Halaman ini juga terdapat *box* saran dimana *user* bisa memasukkan gejala-gejala lain yang belum ada di sistem tersebut sebagai tambahan atau masukan untuk sistem. Halaman hasil diagnosis ditunjukkan pada gambar 3.13 di bawah ini.



Gambar 3.13 Halaman Hasil Diagnosis

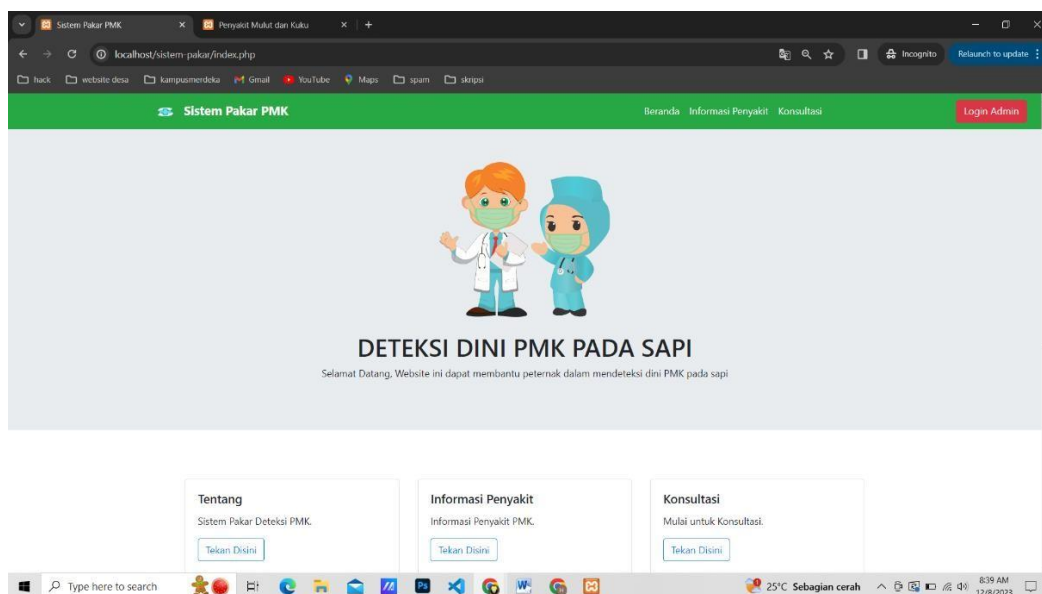
BAB 4

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Pada bab ini merupakan proses penerapan dari rancangan yang sudah dirancang dari bab sebelumnya, tujuannya adalah untuk melihat sistem telah berjalan dan sesuai dengan harapan.

4.1.1 Halaman Beranda

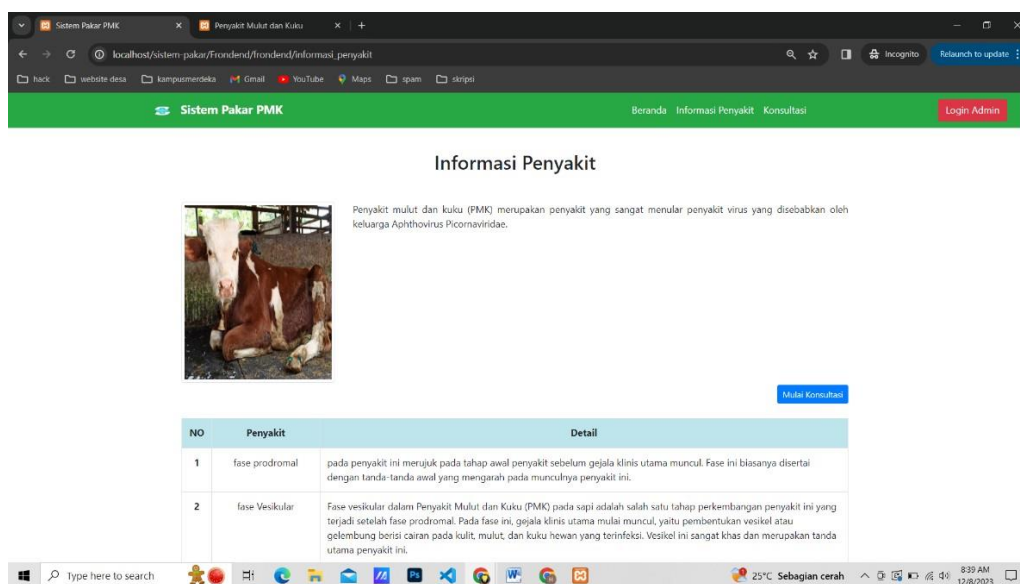


Gambar 4.1 Halaman Beranda

Gambar 4.1 menampilkan halaman utama pada aplikasi, dimana pada halaman ini user bisa melihat penjelasan singkat, ilustrasi, *button* yaitu *button* ‘Mulai Konsultasi’.

4.1.2 Halaman Info

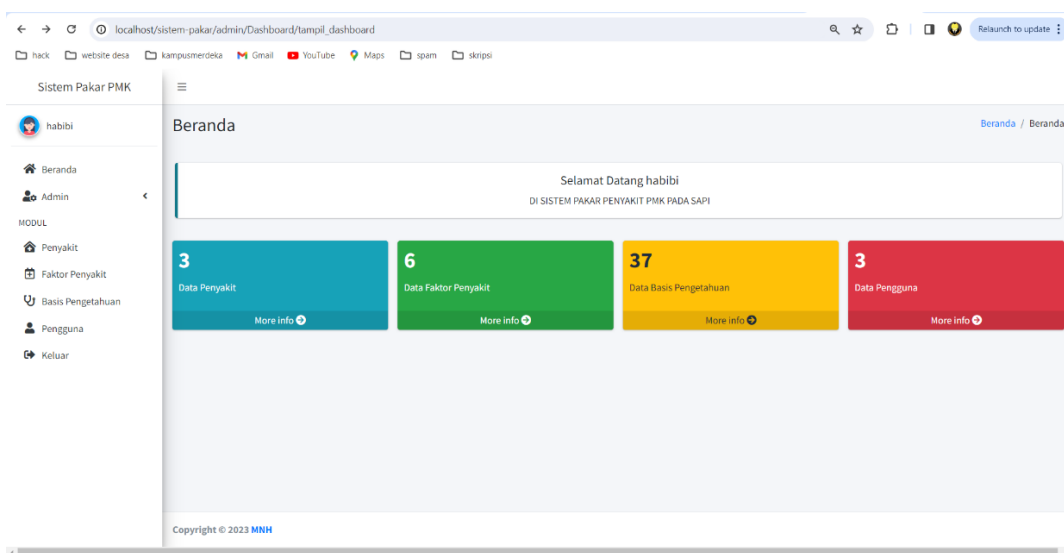
Pada Gambar 4.2 akan berisi info-info mengenai tingkatan penyakit mulut dan kuku dan juga terdapat tombol ‘Konsultasi sekarang’ dimana tombol ini akan mengarah ke halaman pengecekan penyakit mulut dan kuku yang ditunjukkan gambar di atas telah diinput oleh pengguna.



Gambar 4.2 Halaman Info

4.1.3 Halaman Admin

Pada Gambar 4.3 dibawah ini, merupakan halaman admin, dimana pada halaman ini, admin dapat memasukkan, mengubah serta menghapus penyakit atau gejala baru.



Gambar 4.3 Halaman Admin

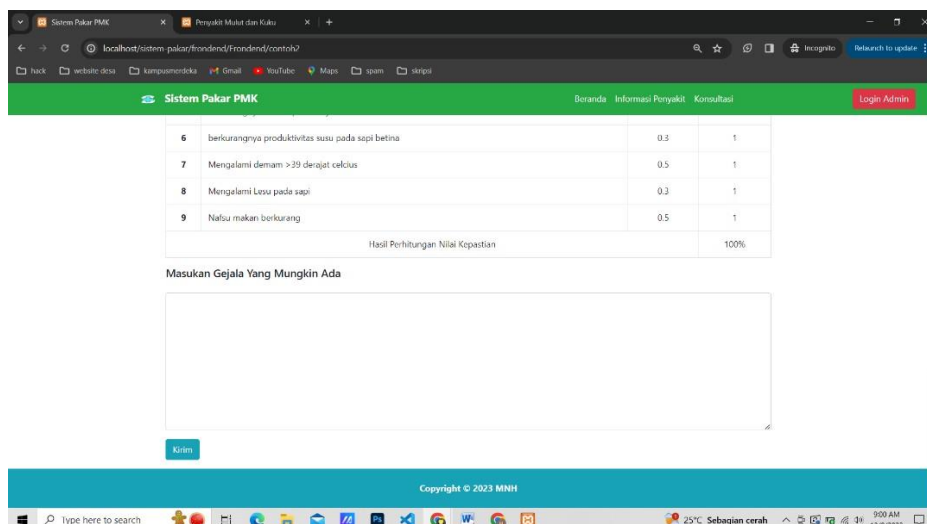
4.14 Halaman Cek Sekarang

Gambar 4.4 Halaman Cek Sekarang

Halaman Cek Sekarang, Gambar 4.4 berisi pertanyaan yang berisi gejala-gejala dan radio button dimana pengguna akan memilih sesuai dengan yang dirasakan. Setelah memilih gejala, pengguna akan menginputkan hasil dan diarahkan ke halaman hasil.

4.15 Halaman Hasil

Pada halaman ini, seperti pada Gambar 4.5, *user* dapat melihat hasil diagnosa dan solusinya. Pada halaman ini *user* juga dapat melihat persenan gejala yang ada serta bisa memasukkan gejala baru yang belum ada di halaman website tersebut.



Gambar 4.5 Halaman Hasil

4.2 Hasil Pengujian

Tahap ini merupakan tahap untuk menguji dan menganalisis sistem untuk mengetahui apakah sistem dapat berjalan dengan baik sesuai dengan yang sudah dirancang. Pada pengujian ini, dilakukan uji coba terhadap 15 data uji yang berasal dari pakar dengan membandingkan hasil diagnosis pakar tersebut dengan hasil diagnosis sistem.

Tabel 4.1 Hasil Pengujian Sistem kepada Pakar

| No. | Gejala yang Dialami | Diagnosis Pakar | Diagnosis Sistem | Hasil |
|-----|-------------------------------------|-----------------------|------------------|--------------|
| 1. | G01,G02,G03 | Fase Prodromal | Fase Prodromal | Sesuai |
| 2. | G01,G02,G03,G04,G05,G07,G09 | Fase Vesikular | Fase Vesikular | Sesuai |
| 3. | G01,G02,G03,G04,G05,G06,G07,G08,G09 | Fase Prodromal | Fase Prodromal | Sesuai |
| 4. | G01,G02,G03,G05 | Fase Prodromal | Fase Prodromal | Sesuai |
| 5. | G01,G03,G04,G05 | Fase Vesikular | Fase Vesikular | Sesuai |
| 6. | G03,G04,G05 | Fase Prodromal | Fase Prodromal | Sesuai |
| 7. | G01,G02,G03,G04,G05 | Fase Vesikular | Fase Vesikular | Sesuai |
| 8. | G03,G04 | Fase Prodromal | Fase Prodromal | Sesuai |
| 9. | G03,G04,G05,G06,G07,G08 | Fase Ulseratif | Fase Ulseratif | Sesuai |
| 10. | G02,G04,G05,G07 | Fase Vesikular | Fase Vesikular | Sesuai |
| 11. | G02,G04,G05,G06,G07 | Fase Ulseratif | Fase Ulseratif | Sesuai |
| 12. | G02,G04,G06 | Belum Teridentifikasi | Tidak Terdeteksi | Tidak Sesuai |

| | | | | |
|-----|-------------------------|--------------------------|------------------|--------------|
| 13. | G02,G04,G05,G06,G08,G09 | Fase Ulseratif | Fase Ulseratif | Sesuai |
| 14. | G02,G03, G05 | Fase Prodromal | Fase Prodromal | Sesuai |
| 15. | G02,G03,G06 | Belum Teridentifikasi | Tidak terdeteksi | Tidak Sesuai |

Dari hasil pengujian terhadap 15 data uji pada tabel 4.1 diatas, maka akurasi perbandingan antara hasil diagnosis sistem dengan hasil diagnosis oleh pakar dapat dihitung sebagai berikut.

$$Akurasi = \frac{\text{Banyak data uji yang sesuai}}{\text{Total data uji}} \times 100\%$$

$$= \frac{13}{15} \times 100\% = 86\%$$

Dari hasil perhitungan nilai akurasi terhadap pengujian sistem pakar ini, didapatkan nilai akurasi sebesar 86% dan membuktikan bahwa sistem pakar diagnosis Penyakit mulut dan kuku dapat berjalan dengan baik.

BAB 5

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil penelitian yang berjudul “Sistem Pakar Deteksi Penyakit Mulut Dan Kuku Pada Hewan Ternak Sapi Dengan Case Based Reasoning Dan Certainty Factor” didapatkan beberapa kesimpulan yaitu :

1. Sistem pakar yang dibangun sebagai informasi umum dan tindakan awal pencegahan terjadinya penyakit penyakit mulut dan kuku sudah dapat berjalan dan memberikan hasil diagnosa sesuai dengan gejala yang diberikan pengguna.
2. Sistem pakar ini terdiri dari 6 gejala, 3 tingkatan penyakit, serta 15 kali pengujian sistem dengan menghasilkan akurasi sebesar 86%
3. Pengumpulan data dan informasi tentang gejala-gejala penyakit mulut dan kuku dan solusi penanganan dini didapatkan dari hasil wawancara terhadap pakar, yaitu dokter hewan dan beberapa data juga didapatkan oleh *user*.

5.2 Saran

Saran penulis yang diberikan agar mengembangkan penelitian selanjutnya, yaitu :

1. Sistem pakar ini dibangun dengan mengimplementasikan algoritma *Case Based Reasoning* dan *Certainty Factor* diharapkan penelitian selanjutnya dapat menggunakan algoritma lain dalam membangun sistem pakar untuk Penyakit mulut dan kuku.
2. Sistem pakar bisa dirancang dan digunakan pada Mobile.
3. Penelitian selanjutnya dapat menggunakan metode serta algoritma yang sama dengan kasus yang berbeda sehingga dapat memberikan varian baru bagi sistem pakar.

DAFTAR PUSTAKA

- Ahmed, B., Megersa, L., Mulatu, G., Siraj, M., & Boneya, G. (2020). Seroprevalence and associated risk factors of foot and mouth disease in cattle in West Shewa Zone, Ethiopia. *Veterinary medicine international*, 2020.
- Alim, S., Lestari, P. P., & Rusliyawati, R. (2020). 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), 26-31.
- Ayu Jayanti, dkk (2020) Sistem Pakar Pendeteksi Kerusakan Kamera DSLR Dengan Metode Case Based Reasoning Dan Certainty Factor. Seminar Informatika Aplikatif Polinema, Politeknik Negeri Malang. ISSN Publishing
- Bani, A. U., & Asruddin, A. (2022). Pendekteksian Penyakit Mulut san Kuku Pada Sapi dengan Menerapkan Metode Naïve Bayes. *Journal of Computer System and Informatics (JoSYC)*, 3(4), 264-268
- Kom, Y. M. (2019). Sistem Pakar Diagnosa Penyakit pada Sapi Berbasis Android. *NUANSA INFORMATIKA*, 13(1).
- Li Chaozhia et. Al (2023) Using Cased Based Reasoning for Automated Safety Risk Management In Construction Industry. School of Civil Engineering, Southeast University Rd, China. Elsevier Publishing
- Rachman Rizal (2021) Implementasi Case Based Reasoning Mendiagnosa Penyakit Stroke Menggunakan algoritma Probabilistic Symmetric. Vol.8 No.1,10-16. *Jurnal Informatika*, Universitas Adhirajasa Reswara Sanjaya. ISSN: 2355-6579

Rohma, M. R., Zamzami, A., Utami, H. P., Karsyam, H. A., & Widianingrum, D. C. (2022). Kasus penyakit mulut dan kuku di Indonesia: epidemiologi, diagnosis penyakit, angka kejadian, dampak penyakit, dan pengendalian. *CONFERENCE_PROCEEDING_SERIES*, 3, 15-22.

Samhan, L. F., Alfarra, A. H., & Abu-Naser, S. S. (2021). Expert System for Knee Problems Diagnosis.

Rubin Stuart H., Hanini Maria R., et. al. (2020) A case-based reasoning system for supervised classification problems in the medical field. *Laboratoire LCSI*.

LAMPIRAN

Pseudo code

Function StartSession():

// Implementasi untuk memulai sesi (start session)

Function CBR(\$userInput):

listGejala = Retrieve 'list_gejala' from Session

listKondisi = Retrieve 'list_kondisi' from Session

gejala = FindSimilarity(\$userInput)

penyakit = {

1: {

'gejala': [

{'kode_gejala': 1, 'nilai_cf': 0.5},

{'kode_gejala': 2, 'nilai_cf': 0.5},

// Data penyakit lainnya...

]

},

// Data penyakit lainnya...

}

penyakitTerpilih = null

kodePenyakit = null

For each penyakitInd in penyakit:

If \$userInput == ArrayColumn(penyakit[penyakitInd]['gejala'], 'nilai_cf'):

penyakitTerpilih = ArrayColumn(penyakit[penyakitInd]['gejala'], 'nilai_cf')

kodePenyakit = penyakitInd

If Not (\$kodePenyakit and \$penyakitTerpilih):

Return {

'kode_penyakit': \$kodePenyakit,

'hasil_akhir': null

}

For each userInputInd in \$userInput:

\$userInput[userInputInd] = \$userInput[userInputInd] * \$penyakitTerpilih[userInputInd]

cfCombine = null

For each userInputInd in \$userInput:

If \$userInputInd == 0:

\$cfCombine = \$userInput[userInputInd]

Else:

\$cfCombine = \$cfCombine + \$userInput[userInputInd] * (1 - \$cfCombine)

```

Return {
    'kode_penyakit': $kodePenyakit,
    'hasil_akhir': $cfCombine * 100
}

```

```

Function FindSimilarity($gejala):
    defaultValue = ArrayFilledWithFalse(Count($gejala))

```

```

    For each gejalaInd in $gejala:
        defaultValue[(int) $gejalaInd - 1] = true

```

```

    Return defaultValue

```

Code System

```

<?php
defined('BASEPATH') or exit('No direct script access allowed');

class Frondend extends CI_Controller
{
    function __construct()
    {
        parent::__construct();
        $this->load->model('frondend/M_frondend');
        $this->load->library('form_validation');
    }

    public function index()
    {
        $this->load->view("frondend/index");
    }

    public function biodata()
    {
        $data = [
            "get_no_user" => $this->M_frondend->get_no_user()
        ];

        $this->load->view("frondend/biodata", $data);
    }

    // Menampilkan pertanyaan rawan
    public function konsultasi()
    {
        $data = [
            "data_kuisoner" => $this->M_frondend->
            >get_data_kuisoner()
        ];

        $this->load->view("frondend/konsultasi", $data);
    }
}

```

```

// Menampilkan pertanyaan aman
public function konsultasi2()
{
    $data = [
        "data_kuisoner" => $this->M_fronddend-
>get_data_kuisoner2()
    ];

    $this->load->view("frondend/konsultasi2", $data);
}

public function hasil_konsultasi()
{
    $this->load->view("frondend/hasil_konsultasi");
}

public function informasi_penyakit()
{
    $this->load->view("frondend/informasi_penyakit");
}

public function tambah_data_biodata()
{
    $show = $this->M_fronddend;
    $id_user = $_POST["id_user"];
    $this->session->set_userdata('id_user', $id_user);

    if ($show->simpan_biodata()) {
        // $this->session->set_flashdata('success', 'Berhasil
Menambah Data Faktor Penyakit');
        redirect(site_url('frondend/Fronddend/konsultasi'));
    }
}

// Method ini untuk save data pertanyaan aman
public function save_cbr()
{
    $data2 = array();
    $index = 0; // Set index array awal dengan 0
    $id_faktor = $_POST["id_faktor"];
    $nilai_cf_user = $_POST["cf_user"];
    $nilai_bobot = $_POST["bobot_faktor"];

    foreach ($id_faktor as $data_id_faktor) { // Kita buat
perulangan berdasarkan id_order sampai data terakhir
        if ($nilai_cf_user[$index]==0) continue;
        array_push($data2, array(
            'id_user' => $this->session-
>userdata('id_user'),
            'id_pengetahuan' => $data_id_faktor,
            'nilai_cf' =>
$nilai_bobot[$index], //index nilai masing" cf atau user
            'nilai_user' => $nilai_cf_user[$index],

```

```

        ));

        $index++;
    }

    $this->M_fronddend->save_batch($data2); // Memasukan Data
Pertanyaan Ke Database

    redirect(base_url("frondend/Fronddend/contoh"));

}

// Method ini untuk save data pertanyaan rawan
public function save_cbr2()
{
    $data2 = array();
    $index = 0; // Set index array awal dengan 0
    $id_faktor = $_POST["id_faktor"];
    $nilai_cf_user = $_POST["cf_user"];
    $nilai_bobot = $_POST["bobot_faktor"];

    foreach ($id_faktor as $data_id_faktor) { // Kita buat
perulangan berdasarkan id_order sampai data terakhir

        array_push($data2, array(
            'id_user' => $this->session-
>userdata('id_user'),
            'id_pengetahuan' => $data_id_faktor,
            'nilai_cf' =>
$nilai_bobot[$index], //index nilai masing" cf atau user
            'nilai_user' => $nilai_cf_user[$index],
        ));

        $index++;
    }

    $this->M_fronddend->save_batch($data2); // Memasukan Data
Pertanyaan Ke Database

    $this->db->select('*');
    $this->db->from('tabel_result_cbr a');
    $this->db->join('tabel_pengetahuan b', 'a.id_pengetahuan =
b.id_pengetahuan');
    $this->db->where('b.id_penyakit', 3);
    $this->db->where('a.id_user', $this->session-
>userdata('id_user'));
    $this->db->where('a.nilai_user', "Pilih Kondisi");
    $cek_jumlah_data = $this->db->get()->num_rows();

    // memeriksa apakah pertanyaan yang di pilih kurang dari 2
atau tidak
    if ($cek_jumlah_data > 4) {
        //Jika Pertanyaan hanya di isi 1
        $id_user = $this->session->userdata('id_user');
    }
}

```

```

        $sql = "DELETE t1 FROM tabel_result_cbr t1
        JOIN tabel_pengetahuan t2 ON t1.id_pengetahuan =
t2.id_pengetahuan
        WHERE t2.id_penyakit = '3' and t1.id_user =
'$id_user' ";

        $this->db->query($sql); // Eksekusi Query SQL

        $this->session->set_flashdata('kurang', 'Silahkan
Pilih Minimum 2 Pertanyaan'); //buat session warning
        redirect(site_url('frondend/Frondend/konsultasi2'));
// dikembalikan pada view konsultasi 2

    } else {
        // Jika Pertanyaan Diisi lebih dari 2
        $where = [
            "id_user" => $this->session-
>userdata('id_user'),
            "nilai_user" => "Pilih Kondisi"
        ];

        $this->db->delete("tabel_result_cbr", $where); //
menghapus data yang tidak diisi oleh user

        redirect(base_url("frondend/Frondend/contoh"));
    }
}

public function contoh()
{
    if (!$this->session->userdata('id_user')) {
        redirect(base_url("frondend/Frondend/biodata"));
    }

    $this->load->view("frondend/contoh");
}

public function contoh2()
{
    if (!$this->session->userdata('id_user')) {
        redirect(base_url("frondend/Frondend/biodata"));
    }

    $id_user = $this->session->userdata('id_user');
    //$id_penyakit = 3;
    //$id_user = "USR0008";
    $cbrAll = $this->M_frondend->get_data_cbr_all($id_user);
    $list_factor_id = array_column($cbrAll, 'id_pengetahuan');
    $penyakit = $this->M_frondend-
>get_data_penyakit_using_cbr($list_factor_id);
    $max = null;

```

```

        foreach($penyakit as $in => $p){
            $this->db->from('tabel_pengetahuan');
            $this->db->where('id_penyakit', $p['id_penyakit']);
            $jumlah_factor_keseluruhan = $this->db->get()-
>num_rows();

            $penyakit[$in]['skor_kemiripan'] =
($p['jumlah_faktor'] / $jumlah_factor_keseluruhan) *100;
            if ($max === null){
                $max = $in;
            } else if ($penyakit[$in]['skor_kemiripan'] >
$penyakit[$max]['skor_kemiripan']){
                $max = $in;
            }
        }

        $id_penyakit = $penyakit[$max]['id_penyakit'];

        $skor_kepastian = null;

        foreach($cbrAll as $in=>$c){
            if ($in == 0){
                continue;
            }
            if ($skor_kepastian == null){
                $CF1 = $cbrAll[$in-1]->nilai_user *
$cbrAll[$in-1]->nilai_cf;
            } else{
                $CF1 = $skor_kepastian;
            }
            $CF2 = $cbrAll[$in]->nilai_user * $cbrAll[$in]-
>nilai_cf;

            $skor_kepastian = $CF1 + $CF2 * (1 - $CF1);
        }

        $skor_kepastian *=100;

        $cbrAll2 = $this->M_fronddend->get_data_cbr_all2($id_user);
        $data = [
            "skor_analisa" => $penyakit[$max]['skor_kemiripan'],
            "Data_cbr_all" => $cbrAll2,
            'penyakit'=>$penyakit,
            'skor_kepastian'=>$skor_kepastian,
            "nilai_cf_all" => $this->M_fronddend-
>get_data_cf_all($id_penyakit, $id_user),
            "solusi" => $this->M_fronddend-
>get_data_cf_solusi($id_penyakit, $id_user),
            'max'=>$max
        ];
        $this->load->view("fronddend/contoh2", $data);
        // $this->session->unset_userdata('id_user'); // Menghapus
        Session id_user
    }

    public function save_masukan(){
        if (!$this->session->userdata('id_user')) {
            redirect(base_url("fronddend/Fronddend/biodata"));
        }
    }

```

```
$id_user = $this->session->userdata('id_user');  
$masukan = $_POST['masukan'];  
$this->db->where('id_user', $id_user);  
$this->db->update('tabel_user', [  
    'masukan'=>$masukan  
]);  
redirect(base_url("frondend/Frondend/biodata"));  
}  
}
```

Foto bersama pakar



Bukti hasil wawancara pakar

Fase Prodromal

1. Suhu tubuh meningkat (rectal $> 39,2$)
2. Nafsu makan menurun
3. Hewan ternak lesu

Fase Vesikular

1. Suhu tinggi
2. Nafsu makan berkurang
3. lesu
4. Kemampuan berjalan berkurang
5. Air liur melimpah
6. Muncul bopuh

Fase Ulseratif

1. Suhu tinggi tidak mau
2. Nafsu makan ~~berkurang~~
3. lesu
4. tidak dapat berdiri
5. Air liur
6. Pecah bopuh

Solusi

Fase prodromal

1. Obat penurunan panas
2. Antibiotik
3. Vitamin

fase II dan III

1. Obat Anti Implanasi
2. Obat mencegah infeksi sekunder atau datang lambat

drh. Anwar, M.Pt



1 Solusi

Foto bersama peternak









