

ChiliCare: Prototype Aplikasi Diagnostik Penyakit Tanaman Cabai Berbasis Website Menggunakan Certainty Factor

Yesi Nur Fitriana^{1*}, Elvita Septian Sadila², Bonanza Cahya Daniswara³, Fajar Suryani⁴

¹Teknik Informatika/Fakultas Ilmu Komputer

Universitas Duta Bangsa Surakarta
1*250103089@mhs.udb.ac.id

²Teknik Informatika/Fakultas Ilmu Komputer

Universitas Duta Bangsa Surakarta
250103070@mhs.udb.ac.id

³Teknik Informatika/Fakultas Ilmu Komputer

Universitas Duta Bangsa Surakarta
250103066@mhs.udb.ac.id

⁴Teknik Informatika/Fakultas Ilmu Komputer

Universitas Duta Bangsa Surakarta
fajar_suryani@udb.ac.id

Abstrak—Indonesia merupakan salah satu penghasil cabai terbesar di Asia Tenggara, namun serangan penyakit dan hama serta keterbatasan akses terhadap tenaga ahli pertanian menyulitkan petani dalam melakukan diagnosis secara mandiri. Penelitian ini bertujuan mengembangkan ChiliCare, aplikasi diagnosis penyakit tanaman cabai berbasis website menggunakan metode *Certainty Factor* (CF). Penelitian menerapkan metode waterfall yang meliputi studi literatur, analisis kebutuhan, perancangan, implementasi, dan pengujian. Sistem dibangun menggunakan HTML5, CSS3, Bootstrap 4, dan vanilla JavaScript tanpa backend server, dengan basis pengetahuan yang mencakup 8 penyakit, 16 gejala, dan nilai CF pakar. Algoritma CF diimplementasikan pada sisi klien untuk menghasilkan tingkat keyakinan diagnosis secara real time. Pengujian menggunakan black-box testing terhadap 12 skenario menunjukkan seluruh fungsi sistem berjalan sesuai rancangan dengan tingkat keberhasilan 100%. Hasil penelitian menunjukkan bahwa ChiliCare berhasil diimplementasikan sebagai aplikasi diagnosis berbasis web yang ringan, mudah diakses, dan mampu membantu petani memperoleh diagnosis penyakit beserta tingkat keyakinannya secara mandiri.

Kata kunci— Certainty Factor, Sistem Pakar, Penyakit Tanaman Cabai, JavaScript, Diagnosis Penyakit.

Abstract— Indonesia is one of the largest chili producers in Southeast Asia, but disease and pest attacks, along with limited access to agricultural experts, make it difficult for farmers to diagnose plants independently. This study aims to develop ChiliCare, a website-based chili plant disease diagnosis application using the *Certainty Factor* (CF) method. The study applies the waterfall method, which includes literature review, needs analysis, design, implementation, and testing. The system was built using HTML5, CSS3, Bootstrap 4, and vanilla JavaScript without a backend server, with a knowledge base covering 8 diseases, 16 symptoms, and expert CF values. The CF algorithm is implemented on the client side to generate a diagnosis confidence level in real time. Testing using black-box testing on 12 scenarios showed that all system functions ran as designed with a 100% success rate. The results show that ChiliCare was successfully implemented as a lightweight, easily accessible web-based diagnosis application that can help farmers obtain disease diagnoses and their confidence levels independently.

Keywords— Certainty Factor, Expert System, Chili Plant Disease, JavaScript, Disease Diagnosis.

I. PENDAHULUAN

Perkembangan teknologi informasi telah mendorong penerapan sistem informasi di sektor pertanian, termasuk sistem pakar diagnosis penyakit tanaman berbasis aturan maupun nilai ketidakpastian [1], [2]. Cabai merupakan komoditas hortikultura bernilai ekonomi tinggi dengan tren konsumsi yang terus meningkat, dari 596,14 ribu ton (2021) menjadi 636,56 ribu ton (2022) [3], seiring permintaan tinggi terhadap cabai rawit sebagai bahan sambal [4]. Jumlah petani cabai pun terus bertambah, khususnya di

sentra produksi seperti Jawa Timur, Jawa Tengah, dan Sumatera Utara [5].

Meski demikian, produksi cabai nasional justru fluktuatif akibat cuaca ekstrem dan perubahan iklim, dengan produksi cabai rawit merah turun dari 1.544,44 ribu ton (2022) menjadi 1.506,76 ribu ton (2023) [6]. Di tingkat daerah, misalnya di Kabupaten Bantul, produksi cabai bahkan anjlok dari hampir 40.000 ton (2021) menjadi sekitar 9.000 ton (2023) akibat serangan hama patek pada lebih dari 100 hektare lahan [1].

Kondisi ini diperparah oleh minimnya pemahaman petani serta keterbatasan tenaga ahli, sehingga diagnosis manual sering terlambat dilakukan [1]. Berbagai penelitian terdahulu telah mengembangkan sistem pakar diagnosis penyakit cabai dengan metode berbeda: Certainty Factor namun belum berbasis web [7]; Certainty Factor untuk diagnosis penyakit cabai [3]; Certainty Factor berbasis web dengan cakupan gejala terbatas [5]; berbasis Android [4], [8]; forward chaining pada tanaman lain [6], [9], [10]; serta pendekatan *fuzzy* dan *Dempster-Shafer* [11]. Penelitian-penelitian tersebut masih terbatas dari sisi aksesibilitas platform, cakupan gejala/penyakit, dan penanganan ketidakpastian gejala yang saling tumpang tindih.

Berdasarkan hal tersebut, penelitian ini mengusulkan **ChiliCare**, sistem informasi berbasis website untuk diagnosis penyakit tanaman cabai menggunakan metode Certainty Factor, yang mampu merepresentasikan tingkat keyakinan pakar terhadap gejala dan hasil diagnosis [12]. Tujuan penelitian ini adalah merancang prototype **ChiliCare** yang menyediakan informasi penyakit cabai sekaligus fitur konsultasi diagnosis, sehingga dapat diakses luas oleh petani kapan pun dan di mana pun untuk mendukung diagnosis yang cepat, mandiri, dan rekomendasi penanganan yang tepat [1], tanpa harus selalu bergantung pada pakar pertanian secara langsung.

II. METODOLOGI PENELITIAN

Penelitian ini menggunakan pendekatan pengembangan waterfall [13], [14], dengan lima tahapan utama, yaitu studi literatur dan akuisisi pengetahuan, analisis kebutuhan sistem, perancangan, implementasi, dan pengujian. Pada tahap studi literatur dan akuisisi pengetahuan, dilakukan pengumpulan referensi terkait penyakit cabai serta penggalan pengetahuan pakar mengenai gejala dan nilai kepastian untuk metode Certainty Factor. Tahap analisis kebutuhan sistem dilakukan untuk mengidentifikasi kebutuhan fungsional dan non-fungsional sistem, sebelum dilanjutkan ke tahap perancangan basis pengetahuan, basis data, dan

antarmuka (*interface*). Tahap implementasi merupakan proses penerjemahan hasil perancangan ke dalam kode program berbasis website, dan tahap pengujian dilakukan untuk memastikan sistem berfungsi sesuai kebutuhan yang ditetapkan. Tahapan tersebut dilakukan secara sistematis untuk menghasilkan prototype aplikasi **ChiliCare** berbasis website. Gambaran tahapan penelitian dapat dilihat pada Gambar 1.



Gambar 1. Tahapan metodologi penelitian

A. Studi Literatur dan Akuisisi Pengetahuan

Tahap ini dilakukan dengan mengumpulkan referensi mengenai sistem pakar, metode Certainty Factor, penyakit tanaman cabai, serta teknologi pengembangan website. Selain itu, dilakukan akuisisi pengetahuan dari buku, jurnal, dan sumber terpercaya mengenai gejala, penyakit, serta penanganan tanaman cabai sebagai dasar dalam penyusunan kebutuhan sistem.

Metode Certainty Factor pertama kali diperkenalkan oleh Shortliffe dan Buchanan [12] untuk mengkombinasikan ketidakpastian dalam proses inferensi medis, dan kemudian banyak diadaptasi pada sistem pakar diagnosis penyakit tanaman [3][6]. Perhitungan CF dilakukan dalam dua tahap. Pertama, dihitung CF Evidence per gejala menggunakan persamaan (1):

$$CF(E) = CF(\text{Pakar}) \times CF(\text{User}) \quad (1)$$

Pada persamaan (1), $CF(E)$ menyatakan nilai keyakinan gabungan untuk satu gejala tunggal, $CF(Pakar)$ adalah bobot keyakinan pakar terhadap keterkaitan gejala dengan suatu penyakit dan $CF(User)$ adalah nilai keyakinan yang dipilih pengguna saat mengisi form diagnosis. Nilai $CF(User)$ berasal dari lima pilihan tingkat keyakinan sebagaimana ditunjukkan pada Tabel 1, mengikuti skala keyakinan yang umum diterapkan pada sistem pakar berbasis CF [5].

Tabel 1. Skala Nilai CF User

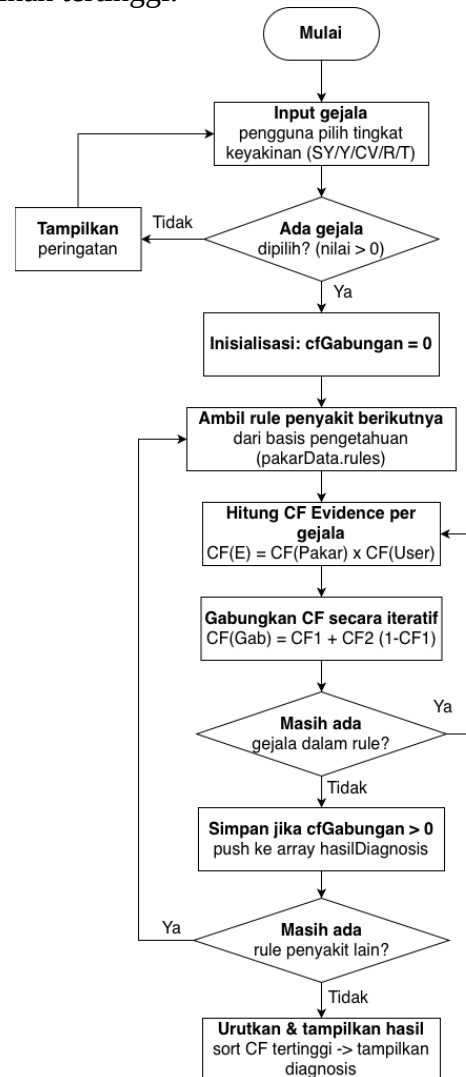
Kode	Keterangan	Nilai CF User
1	Sangat Yakin	1,0
2	Yakin	0,8
3	Cukup Yakin	0,6
4	Ragu-ragu	0,4
5	Tidak	0,0

Kedua, apabila lebih dari satu gejala relevan dengan penyakit yang sama, nilai $CF(E)$ dari setiap gejala dikombinasikan menjadi satu nilai $CF Gabungan$ menggunakan persamaan (2):

$$CF(Gabungan) = CF1 + CF2 \times (1 - CF1) \quad (2)$$

Pada persamaan (2), $CF1$ adalah nilai CF Evidence dari gejala yang telah diproses sebelumnya (atau nilai $CF(E)$ pertama jika baru satu gejala yang terpenuhi), dan $CF2$ adalah nilai $CF(E)$ dari gejala berikutnya yang akan dikombinasikan. Rumus kombinasi ini diterapkan secara iteratif untuk setiap gejala yang relevan pada suatu *rule* penyakit [6][7], sehingga apabila terdapat tiga gejala atau lebih, hasil kombinasi dua gejala pertama akan menjadi $CF1$ baru untuk dikombinasikan kembali dengan gejala ketiga, dan seterusnya. Hasil akhir $CF Gabungan$ dikalikan 100 untuk menghasilkan persentase keyakinan diagnosis yang ditampilkan kepada pengguna.

Gambar 2 menunjukkan alur kerja algoritma Certainty Factor yang diimplementasikan pada sistem **ChiliCare**, mulai dari input gejala oleh pengguna hingga penampilan hasil diagnosis yang telah diurutkan berdasarkan persentase keyakinan tertinggi.



Gambar 2. Flowchart Algoritma Certainty Factor pada Sistem ChiliCare

Berdasarkan Gambar 2, alur CF dimulai dari input gejala pengguna. Sistem memvalidasi apakah terdapat minimal satu gejala yang dipilih (nilai CF User > 0). Jika tidak ada, sistem menampilkan peringatan agar pengguna memilih minimal satu gejala. Jika ada, sistem menginisialisasi $cfGabungan = 0$, lalu mengiterasi seluruh *rule* penyakit dari basis pengetahuan. Untuk setiap *rule*, $CF(E)$ per gejala dihitung menggunakan persamaan (1) dan

digabungkan secara iteratif menggunakan persamaan (2) [12]. Penyakit dengan *CF Gabungan* > 0 disimpan ke dalam larik hasil diagnosis, dan setelah seluruh *rule* diproses, hasil diurutkan berdasarkan persentase *CF* tertinggi sebelum ditampilkan kepada pengguna.

B. Analisis Kebutuhan Sistem

Tahap ini bertujuan untuk mengidentifikasi kebutuhan sistem berdasarkan kebutuhan pengguna, mencakup kebutuhan fungsional dan non fungsional. Kebutuhan fungsional meliputi, halaman beranda, halaman informasi penyakit, fitur konsultasi berbasis metode Certainty Factor, dan halaman hasil diagnosis yang menampilkan penyakit paling mungkin beserta nilai keyakinan dan rekomendasi penanganan. Kebutuhan non fungsional meliputi, kemudahan penggunaan (*usability*) agar mudah dipahami petani dengan berbagai latar belakang literasi digital, tampilan responsif agar dapat diakses melalui berbagai perangkat, dan kemudahan akses melalui web tanpa memerlukan instalasi tambahan.

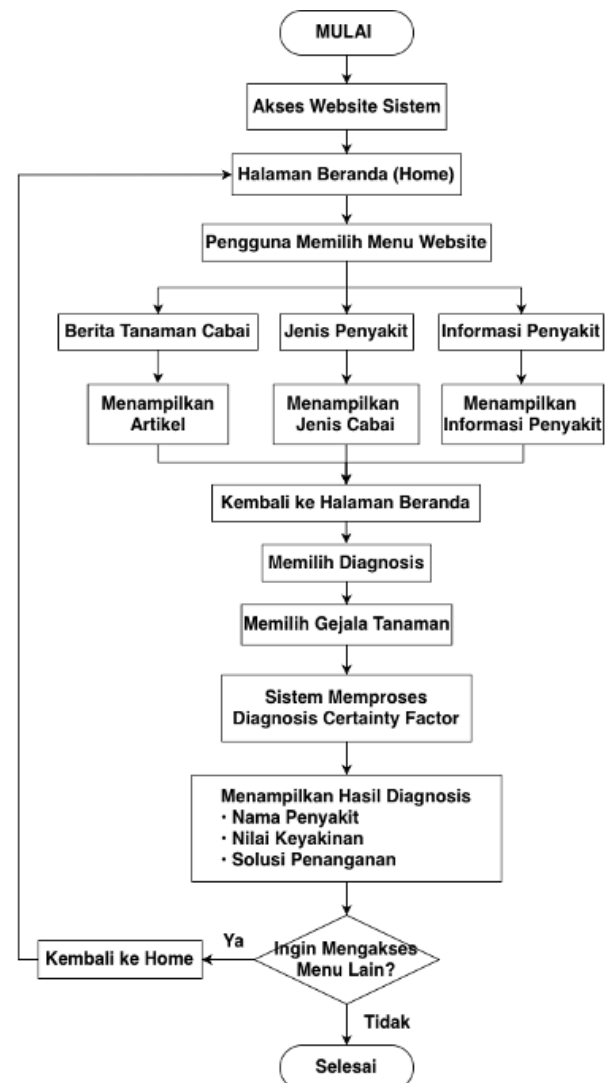
C. Perancangan

Tahap perancangan dilakukan dengan menyusun beberapa komponen utama, meliputi arsitektur sistem, desain antarmuka (*User Interface*), struktur navigasi, perancangan basis data, dan pemodelan sistem menggunakan UML (*Unified Modeling Language*). Arsitektur sistem dirancang untuk menggambarkan hubungan antar komponen dalam **ChiliCare**, mulai dari sisi pengguna (*front-end*) hingga proses pengolahan data pada sisi server (*back-end*), sehingga alur kerja sistem dapat dipahami secara menyeluruh sejak tahap awal pengembangan.

Selain itu, dirancang pula alur proses konsultasi mulai dari pemilihan gejala hingga diperolehnya hasil diagnosis, serta mekanisme penerapan metode Certainty Factor sebagai dasar perhitungan tingkat keyakinan terhadap jenis penyakit tanaman cabai yang teridentifikasi. Perancangan antarmuka disusun agar mudah digunakan oleh petani dengan berbagai tingkat pemahaman teknologi, sementara struktur navigasi dirancang agar pengguna dapat mengakses fitur diagnosis maupun informasi penyakit secara intuitif. Hasil perancangan pada

tahap ini menjadi acuan utama pada tahap implementasi dan pengembangan sistem berikutnya.

D. Implementasi



Gambar 3. Flowchart Alur Sistem ChiliCare

Berbeda dengan penelitian sejenis yang menggunakan arsitektur *client-server* dengan PHP dan basis data MySQL serta pemodelan UML dengan tiga aktor (admin, petani, dan pakar) [1], **ChiliCare** diimplementasikan sepenuhnya pada sisi klien tanpa backend server [13]. Seluruh basis pengetahuan disimpan dalam berkas JavaScript (*data.js*) berbentuk objek yang memuat larik penyakit, gejala, dan *rules*, sehingga sistem dapat berjalan hanya dengan membuka berkas HTML pada peramban tanpa

memerlukan instalasi server maupun basis data relasional. Gambar 3 menunjukkan alur sistem **ChiliCare** secara keseluruhan, mulai dari pengguna mengakses website hingga proses navigasi antar menu dan proses diagnosis.

E. Pengujian

Pengujian sistem menggunakan metode *black-box testing* yang mengikuti kerangka *user acceptance testing* untuk memvalidasi kesesuaian setiap halaman dengan kebutuhan pengguna [14]. Pengujian dilakukan per halaman agar cakupan skenario fungsional pada setiap bagian sistem dapat ditelusuri secara rinci.

III. HASIL DAN PEMBAHASAN

A. Analisis Kebutuhan

Analisis kebutuhan dilakukan untuk mengidentifikasi kebutuhan pengguna dalam proses diagnosis penyakit tanaman cabai. Berdasarkan analisis tersebut, **ChiliCare** dirancang dengan satu aktor utama, yaitu pengguna, yang dapat mengakses informasi penyakit, melakukan konsultasi berdasarkan gejala yang dipilih, serta melihat hasil diagnosis.

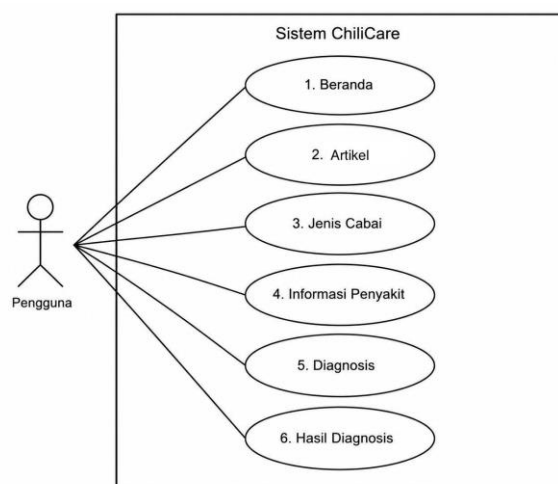
Analisis juga menunjukkan bahwa sistem memerlukan beberapa fitur utama, yaitu halaman Home, Berita, Jenis Cabai, Informasi, Diagnosis, dan Hasil Diagnosis. Selain memenuhi kebutuhan fungsional, sistem dirancang berbasis website agar dapat diakses melalui berbagai perangkat menggunakan browser dengan antarmuka yang sederhana dan mudah digunakan. Hasil analisis ini menjadi dasar dalam proses perancangan dan implementasi *prototype ChiliCare*.

B. Perancangan Sistem

Perancangan sistem dilakukan berdasarkan hasil analisis kebutuhan yang telah diperoleh pada tahap sebelumnya. Tujuan dari tahap ini adalah menghasilkan rancangan aplikasi yang dapat memenuhi kebutuhan pengguna serta menjadi acuan dalam proses implementasi *prototype* berbasis website.

Berdasarkan hasil analisis, aplikasi **ChiliCare** dirancang dengan satu aktor utama, yaitu pengguna, yang dapat mengakses fitur beranda, berita, jenis cabai, informasi penyakit, diagnosis, dan hasil diagnosis. Untuk menggambarkan alur kerja sistem, perancangan dilakukan menggunakan *Unified Modeling Language* (UML) yaitu menggunakan *Use Case Diagram*.

Use Case Diagram digunakan untuk menggambarkan hubungan antara pengguna dengan fitur-fitur yang tersedia pada aplikasi **ChiliCare**. Berdasarkan hasil analisis kebutuhan, aplikasi memiliki satu aktor, yaitu pengguna, yang dapat mengakses seluruh fitur sistem. Pengguna dapat membuka halaman beranda, berita, jenis cabai, informasi penyakit, diagnosis, dan hasil diagnosis. Diagram ini memberikan gambaran fungsi utama aplikasi serta menjadi dasar dalam proses perancangan sistem dan pengembangan *prototype*.



Gambar 4. Use Case Diagram ChiliCare

C. Desain Interface Sistem

Perancangan antarmuka dilakukan melalui penyusunan *wireframe* untuk menentukan struktur informasi dan tata letak setiap halaman sebelum proses implementasi. *Wireframe* mencakup lima halaman utama **ChiliCare**, yaitu Home, Berita, Jenis Cabai, Informasi, dan Diagnosis, yang selanjutnya diimplementasikan dengan identitas visual berupa logo, palet warna merah-putih, dan ilustrasi tanaman cabai.

1. Rancangan Halaman Home



Gambar 5. Wireframe Halaman Home

Gambar 5 merupakan *wireframe* home menempatkan navigasi utama (Home, Berita, Jenis Cabai, Informasi) beserta tombol aksi utama Diagnosis pada bagian kanan atas agar mudah ditemukan. Bagian hero menampilkan judul **ChiliCare**, deskripsi singkat, serta dua tombol Diagnosis Sekarang dan Cara Kerja, diikuti bagian Cara Kerja yang menjelaskan tiga langkah penggunaan sistem.

2. Rancangan Halaman Berita



Gambar 6. Wireframe Halaman Berita

Gambar 6 adalah *wireframe* berita menampilkan rancangan hero artikel utama pada bagian atas berupa gambar besar dengan judul, penulis, dan tanggal publikasi, disusul strip berita berjalan (*running text*), kemudian dua kolom berisi daftar Berita Terbaru dan Trending di bagian bawah.

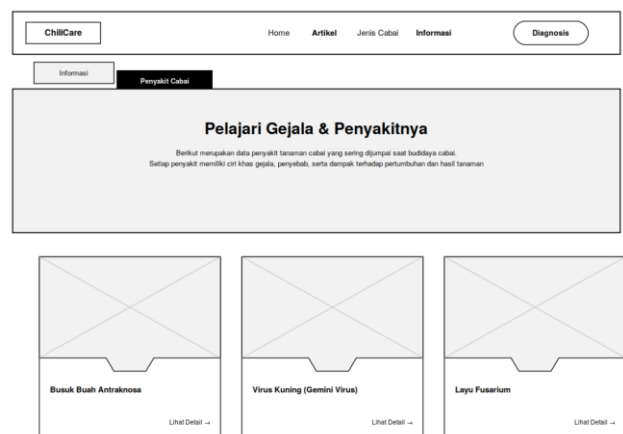
3. Rancangan Halaman Jenis Cabai



Gambar 7. Wireframe Halaman Jenis Cabai

Wireframe Jenis Cabai pada Gambar 7 memuat hero berisi deskripsi singkat mengenai keragaman spesies *Capsicum* beserta tombol Lihat Jenis, dilanjutkan bagian Mengenal Jenis-Jenis Cabai yang menjelaskan klasifikasi lima spesies budidaya utama.

4. Rancangan Halaman Informasi



Gambar 8. Wireframe Halaman Informasi

Wireframe Informasi pada Gambar 8 menampilkan rancangan informasi menggunakan tampilan tab (Informasi dan Penyakit Cabai) menyerupai folder arsip, dengan deskripsi singkat pada bagian atas dan tiga kartu ringkasan penyakit (nama penyakit serta tautan Lihat Detail) pada bagian bawah.

5. Rancangan Halaman Diagnosis

Form Diagnosis Penyakit Tanaman Cabai

Keterangan Pilihan: 1/1 Sangat Terjadi 2/2 Terjadi 3/3 Cukup Terjadi 4/4 Banyak Terjadi 5/5 Sangat Banyak Terjadi

- Apakah Terjadi kerak matang/ragam/merah/merah/merah pada buah?
- Apakah Buah menjadi keriput, mengering, atau membusuk lunak?
- Apakah Muncul titik hitam (bukan busuk) pada pangkal buah?
- Apakah Buah menjadi pucat dan gigitan ke busuk seluruhnya?
- Apakah Pucuk dan batang daun berubah menjadi kuning terang (klorosis)?
- Apakah Tulang daun membusuk dan menggigit ke atas?
- Apakah Pertumbuhan tanaman menjadi lambat dan sulit berbuah?
- Apakah Tanaman mengalami kelutihan mulai dari bagian bawah yang perlahan memutar ke atas?
- Apakah Warna jaringan pembuluh pada akar dan batang berubah menjadi putih?
- Apakah Tanaman layu secara tiba-tiba (badai) dan belum mengering / masih hijau?
- Apakah Pangkal batang terakus lunak, busuk, dan berbau tajam?
- Apakah Terjadi kerak busuk lunak, busuk, dan busuk pada daun?
- Apakah Daun yang berumur tua layu dan mengering?
- Apakah Pertumbuhan busuk dan terdapat busuk busuk busuk?
- Apakah Daun berubah menjadi merah putih busuk, mengering, dan busuk?
- Apakah Daun mengering ke busuk, mengering, dan terdapat busuk putih / paku hitam?

Legenda:

- ChiliCare**: Sistem yang digunakan untuk membantu petani cabai dalam diagnosis penyakit.
- Profil**: Nama, Jenis Cabai, Informasi.
- Salah**: Tidak ada jawaban yang benar.

Gambar 9. Wireframe Halaman Diagnosis

Wireframe Diagnosis pada gambar 9 menampilkan rancangan halaman diagnosis yang berisi judul Form Diagnosis Penyakit Tanaman Cabai beserta legenda keterangan pilihan (1/2/3/4/5), dan daftar pertanyaan gejala yang masing-masing dilengkapi lima opsi tingkat keyakinan dalam bentuk tombol sejajar, yang pada implementasi akhir diubah menjadi komponen sliding pill radio.

D. Implementasi Sistem

Tahap implementasi sistem dilakukan untuk merealisasikan hasil perancangan menjadi sebuah aplikasi yang dapat digunakan oleh pengguna. Pada penelitian ini, sistem **ChiliCare** diimplementasikan sebagai platform digital yang menyediakan informasi budidaya cabai, identifikasi penyakit, serta fasilitas diagnosis berdasarkan gejala yang dipilih oleh pengguna. Implementasi meliputi pembangunan antarmuka pengguna, integrasi basis pengetahuan, serta penerapan mekanisme inferensi sehingga seluruh fitur yang telah dirancang dapat berjalan sesuai dengan kebutuhan fungsional sistem. Pada subbab ini dijelaskan implementasi setiap halaman utama aplikasi, meliputi halaman beranda (home), artikel, informasi penyakit, serta fitur diagnosis beserta hasil yang ditampilkan kepada pengguna. Adapun hasil implementasi *prototype ChiliCare* dapat dilihat pada Gambar 9-13.

1. Halaman Home

Halaman Home merupakan tampilan awal **ChiliCare** yang menyediakan akses ke seluruh fitur utama. Halaman ini menampilkan slogan "Solusi Cerdas Petani Cabai" serta tombol Diagnosis Sekarang dan Cara Kerja sebagai akses menuju fitur diagnosis. Implementasi halaman Home ditunjukkan pada Gambar 9.



Gambar 10. Implementasi Halaman Home

2. Halaman Artikel

Halaman Artikel berfungsi sebagai media edukasi yang menyajikan informasi terkini mengenai budidaya cabai. Halaman ini menampilkan artikel unggulan, artikel terbaru, dan berita populer yang bersumber dari media terpercaya. Implementasi halaman artikel ditunjukkan pada Gambar 11. .



Gambar 11. Implementasi Halaman Berita

Berdasarkan Gambar 11, halaman artikel menampilkan artikel dalam bentuk *featured post* dan daftar artikel berbentuk *grid* yang dilengkapi gambar, judul, serta ringkasan isi. Pengguna dapat mengklik gambar atau judul artikel untuk melihat informasi secara lengkap.

3. Halaman Jenis Cabai

Halaman Jenis Cabai menyajikan informasi mengenai lima spesies cabai, yaitu *Capsicum annum*, *Capsicum frutescens*, *Capsicum chinense*, *Capsicum baccatum*, dan *Capsicum pubescens*. Informasi disajikan dalam bentuk *card* yang memuat gambar, nama ilmiah, nama lokal, dan deskripsi singkat. Pengguna dapat memilih tombol Pelajari untuk melihat informasi lebih lengkap mengenai spesies yang dipilih. Implementasi halaman Jenis Cabai ditunjukkan pada Gambar 12.



Gambar 12. Implementasi Halaman Jenis Cabai

4. Halaman Informasi

Halaman Informasi menyajikan daftar penyakit dan hama yang umum menyerang tanaman cabai dalam bentuk *card* yang dilengkapi gambar, nama penyakit atau hama, serta tombol Lihat Detail. Halaman ini menggunakan tab Informasi dan Penyakit Cabai dengan tampilan bergaya folder arsip berwarna merah gradasi yang memuat delapan kartu penyakit. Ketika pengguna memilih tombol Lihat Detail, sistem menampilkan informasi lengkap yang meliputi deskripsi, gejala, penyebab, bagian tanaman yang diserang, dan cara penanganan. Implementasi halaman Informasi ditunjukkan pada Gambar 13.



Gambar 13. Implementasi Halaman Informasi

5. Halaman Diagnosis

Halaman Diagnosis digunakan untuk melakukan proses identifikasi penyakit tanaman cabai berdasarkan gejala yang dipilih oleh pengguna. Setiap gejala dinilai menggunakan skala tingkat keyakinan dari 1 (Tidak) , 2 (Ragu-ragu), 3 (Cukup Yakin), 4 (Yakin), 5 (Sangat Yakin). Setelah seluruh gejala diisi, pengguna

dapat menekan tombol *Analisis Diagnosis* untuk memproses data menggunakan metode Certainty Factor (CF) sehingga sistem dapat menampilkan hasil diagnosis beserta tingkat keyakinannya. Implementasi tampilan halaman Diagnosis ditunjukkan pada Gambar 14.

Gambar 14. Implementasi Halaman Diagnosis

Sebagai ilustrasi mekanisme perhitungan, digunakan contoh diagnosis penyakit Antraknosa (P01). Pada contoh ini, pengguna memilih gejala G01 dengan tingkat keyakinan 4 (Yakin) (CF User = 0,8) dan gejala G02 dengan tingkat keyakinan 3 (Cukup Yakin) (CF User = 0,6). Berdasarkan basis pengetahuan sistem, nilai CF Pakar untuk gejala G01 adalah 0,8, sedangkan untuk gejala G02 adalah 0,7. Dengan demikian, variabel yang digunakan dalam perhitungan terdiri atas penyakit P01, gejala G01 dan G02, nilai CF Pakar, serta nilai CF User. Selanjutnya,

nilai Certainty Factor dihitung menggunakan Persamaan (1) [5], [7].

$$CF(E)1 = 0,8 \times 0,8 = 0,64$$

$$CF(E)2 = 0,7 \times 0,6 = 0,42$$

Selanjutnya, kombinasi dari kedua nilai CF tersebut dihitung nilai CF Gabungan menggunakan persamaan (2) seperti yang ditunjukkan dibawah ini :

$$CF \text{ Gabungan} = 0,64 + 0,42 \times (1 - 0,64) = 0,7912$$

Sistem menampilkan diagnosis Antraknosa dengan keyakinan 79,12% beserta deskripsi dan rekomendasi penanganan. Penyakit lain dengan $CF > 0$ ditampilkan sebagai kemungkinan alternatif, memberikan informasi yang lebih komprehensif dibandingkan sistem *rule based* murni yang hanya menghasilkan satu kesimpulan [1].

Untuk memperlihatkan variasi hasil pada kedelapan penyakit/hama, Tabel 2 menyajikan ilustrasi perhitungan CF Gabungan dengan skenario setiap gejala pada *rule* masing-masing penyakit dipilih pada tingkat keyakinan yang sama, yaitu 4 (Yakin) (CF User = 0,8), sehingga perbedaan hasil akhir semata-mata mencerminkan perbedaan nilai CF Pakar pada basis pengetahuan. Karena setiap penyakit memiliki gejala kunci yang berbeda (lihat Tabel 1), gejala yang digunakan pada tiap baris disesuaikan dengan *rule* basis pengetahuan penyakit yang bersangkutan.

Tabel 2. Ilustrasi Perhitungan CF Gabungan untuk Kedelapan Penyakit/Hama

No	ID	Penyakit / Hama	CF(E) per Gejala (CF User = 0,8)	CF Gabungan
1	P01	Antraknosa (Busuk Buah)	G01: 0,64; G02: 0,56	84,16%
2	P02	Virus Kuning (Gemini Virus)	G05: 0,64; G06: 0,72; G07: 0,48	94,76%
3	P03	Layu Fusarium	G08: 0,72; G09:	93,15%

No	ID	Penyakit / Hama	CF(E) per Gejala (CF User = 0,8)	CF Gabungan
			0,64; G07: 0,32	
4	P04	Layu Bakteri	G10: 0,72; G09: 0,48; G11: 0,56	93,59%
5	P05	Bercak Daun (Cercospora)	G12: 0,72; G13: 0,56	87,68%
6	P06	Hama Thrips	G14: 0,72; G15: 0,64; G07: 0,32	93,15%
7	P07	Hama Kutu Daun (Aphids)	G03: 0,72; G02: 0,40; G04: 0,64	80,96%
8	P08	Hama Lalat Buah	G03: 0,72; G02: 0,40; G04: 0,64	93,95%

Berdasarkan Tabel 2, seluruh penyakit dan hama pada basis pengetahuan **ChiliCare** menghasilkan rentang tingkat keyakinan antara 80,96% hingga 94,76% pada skenario tingkat keyakinan pengguna yang seragam, yaitu Yakin (CF User = 0,8). Penyakit dengan nilai CF Pakar yang lebih tinggi dan konsisten pada setiap gejalanya, seperti Virus Kuning (P02) dan Hama Lalat Buah (P08), menghasilkan nilai CF Gabungan yang lebih tinggi dibandingkan penyakit lain yang memiliki kombinasi nilai CF Pakar lebih rendah, seperti Hama Kutu Daun (P07). Hasil tersebut menunjukkan bahwa metode Certainty Factor mampu menghasilkan tingkat keyakinan diagnosis yang berbeda sesuai dengan bobot keyakinan pakar pada setiap gejala. Dengan demikian, mekanisme CF Gabungan pada **ChiliCare** dapat memberikan hasil diagnosis yang proporsional berdasarkan kombinasi nilai CF Pakar dan tingkat keyakinan pengguna, sejalan dengan penerapan metode Certainty Factor pada penelitian sejenis [15].

F. Pengujian Sistem

Pengujian sistem menggunakan metode *black-box testing* yang mengikuti kerangka *user acceptance testing* untuk memvalidasi kesesuaian setiap halaman dengan kebutuhan pengguna [14]. Pengujian dilakukan per halaman agar cakupan skenario fungsional pada setiap bagian sistem dapat ditelusuri secara rinci, sebagaimana disajikan pada Tabel 3.

Tabel 3. Hasil Pengujian Fungsionalitas Sistem ChiliCare

No	Halaman	Skenario Pengujian	Output yang Diharapkan	Status
1	Home	Akses index.html / klik menu Home pada navbar	Hero section, navigasi utama, dan tombol Diagnosis tampil dengan tampilan responsif	Valid
2	Home	Klik tombol "Diagnosis Sekarang" / tombol Diagnosis pada navbar	Pengguna dialihkan ke halaman diagnosis.html	Valid
3	Berita	Akses halaman Berita	Hero artikel utama, strip berita berjalan, serta daftar Berita Terbaru dan Trending tampil lengkap	Valid
4	Jenis Cabai	Akses halaman Jenis Cabai / klik tombol "Lihat Jenis"	Penjelasan lima spesies cabai beserta gambar pendukung tampil	Valid
5	Informasi	Klik menu Informasi	Tab Penyakit Cabai dengan card folder 8 penyakit/hama tampil lengkap	Valid
6	Diagnosis	Klik tautan "Lihat Detail" pada salah satu kartu penyakit	Pengguna dialihkan ke halaman detail penyakit yang dipilih	Valid
7	Diagnosis	Buka halaman Diagnosis	16 gejala berhasil dirender dari basis pengetahuan (data.js)	Valid
8	Diagnosis	Klik opsi tingkat keyakinan (1/2/3/4/5) pada sliding pill	Pill meluncur ke posisi opsi yang dipilih dan nilai CF User tersimpan secara real-time	Valid
9	Diagnosis	Pilih kombinasi gejala tertentu (mis. G01 & G02), klik Analisis	Sistem menampilkan diagnosis penyakit beserta persentase CF Gabungan	Valid
10	Diagnosis	Klik Analisis tanpa memilih gejala apapun	Muncul peringatan agar memilih minimal satu gejala	Valid
11	Diagnosis	Pilih kombinasi gejala yang tidak cocok dengan <i>rule</i> manapun	Muncul pesan "Diagnosis Tidak Ditemukan"	Valid

12	Diagnosis	Klik tombol "Diagnosis Ulang"	Konfirmasi muncul, seluruh pilihan gejala dan hasil diagnosis dikembalikan ke kondisi awal	Valid
----	-----------	-------------------------------------	---	-------

Berdasarkan Tabel 3, pengujian fungsional dilakukan terhadap 12 skenario pada lima halaman utama **ChiliCare**, yaitu Home, Berita, Jenis Cabai, Informasi, dan Diagnosis. Seluruh skenario menghasilkan status Valid, yang menunjukkan bahwa setiap fitur telah berjalan sesuai dengan rancangan dan menghasilkan keluaran yang diharapkan. Pengujian mencakup fungsi navigasi, penyajian informasi, proses pemilihan gejala, perhitungan Certainty Factor, hingga penanganan kondisi tanpa input, diagnosis tidak ditemukan, dan fitur reset.

Hasil pengujian menunjukkan bahwa *prototype ChiliCare* telah berhasil diimplementasikan dan seluruh fungsi utama dapat dijalankan sebagaimana mestinya. Fitur diagnosis mampu memproses gejala yang dipilih pengguna dan menampilkan hasil diagnosis beserta tingkat keyakinannya sesuai mekanisme metode Certainty Factor. Meskipun demikian, pengujian pada penelitian ini masih terbatas pada aspek fungsional menggunakan *black-box testing*, sehingga pengujian *usability* oleh pengguna serta validasi akurasi hasil diagnosis terhadap pakar dapat dilakukan pada penelitian selanjutnya.

IV. KESIMPULAN

Berdasarkan hasil penelitian dan pembahasan, *prototype ChiliCare* berhasil dirancang dan diimplementasikan sebagai aplikasi *front-end* murni berbasis HTML5, CSS3, Bootstrap 4, dan *vanilla* JavaScript, tanpa ketergantungan pada server *backend* maupun basis data relasional. *Prototype* ini terdiri atas lima halaman utama, yaitu Home, Berita, Jenis Cabai, Informasi, dan Diagnosis, dengan antarmuka yang dirancang sederhana dan intuitif agar mudah digunakan oleh petani tanpa latar belakang teknis. Seluruh logika sistem, termasuk mekanisme diagnosis, berjalan sepenuhnya di sisi klien dengan memanfaatkan basis pengetahuan yang tertanam

dalam berkas JavaScript. Pengujian *black-box* terhadap 12 skenario pada kelima halaman tersebut menunjukkan bahwa seluruh fitur berfungsi sesuai rancangan, dengan tingkat validitas mencapai 100%.

Meskipun unggul dari segi arsitektur yang ringan dan dapat diakses langsung melalui peramban tanpa instalasi server maupun koneksi internet setelah aset dimuat, *prototype* ini masih memiliki sejumlah keterbatasan. Basis pengetahuan sistem tertanam secara statis dalam kode sumber, sehingga penambahan atau pembaruan data belum dapat dilakukan secara dinamis melalui antarmuka, melainkan harus melalui modifikasi kode secara manual. Riwayat konsultasi pengguna juga hanya tersimpan sementara di *Session Storage* dan hilang setelah sesi peramban berakhir. Selain itu, *prototype* ini baru diuji secara fungsional melalui pendekatan *black-box* dan belum melalui pengujian *usability* langsung bersama petani sebagai pengguna akhir, sehingga kemudahan penggunaan sistem di lapangan belum sepenuhnya terukur.

Untuk penelitian selanjutnya, disarankan agar *prototype* ini dikembangkan menjadi aplikasi web yang lebih dinamis dengan mengintegrasikan layanan *backend-as-a-service* seperti Firebase atau Supabase, disertai panel administrasi agar pengelolaan data dapat dilakukan melalui antarmuka tanpa mengubah kode sumber. Selain itu, disarankan pula konversi sistem menjadi *Progressive Web App* (PWA) agar dapat diakses secara *offline* di lapangan, serta pelaksanaan *usability testing* bersama petani menggunakan metode *System Usability Scale* (SUS) sebelum sistem dirilis secara lebih luas.

REFERENSI

- [1] A. Srirahayu dan F. Suryani, "Prototype sistem informasi diagnosis cabai berbasis android," Seminar Nasional AMIKOM Surakarta (SEMNAS), hlm. 219-225, 2023.
- [2] V. Edo dan A. K. Wijayanto, "Sistem Pakar Diagnosis Hama & Penyakit Tanaman Pangan," IPB, Bogor, 2015.
- [3] R. Mujahiddin, Zaeniah, dan B. Imran, "Rancang bangun sistem pakar diagnosa penyakit pada tanaman cabai dengan metode certainty factor," J. Kecerdasan Buatan dan Teknol. Inf., vol. 2, no. 1, hlm. 11-19, 2023.
- [4] M. D. Ronaldo dan I. Maziidi, "Sistem pakar diagnosis penyakit tanaman cabai menggunakan metode certainty factor berbasis android," JKBTI, vol. 3, no. 2, 2024.

- [5] S. Holifah dkk., "Sistem pakar diagnosa penyakit tanaman cabai berbasis web menggunakan certainty factor," *J. Inf. dan Komput.*, vol. 11, no. 1, hlm. 58-64, 2023.
- [6] N. A. Muslimah, R. Y. Bakti, dan T. Wahyuni, "Sistem pakar mendiagnosa penyakit pada tanaman menggunakan metode forward chaining," *Aurus J. Sains dan Teknol.*, vol. 2, no. 2, 2023.
- [7] T. Winanto, Y. R. W. Utami, dan S. H. Fitriasih, "Sistem pakar diagnosa hama dan penyakit tanaman cabai besar menggunakan metode certainty factor," *J. Ilm. SINUS*, vol. 15, no. 2, hlm. 13-24, 2017.
- [8] Suradi dkk., "Sistem pakar diagnosa hama dan penyakit tanaman cabai berbasis android dengan metode forward chaining," *J. Teknol. dan Komputer (JTEK)*, vol. 3, no. 1, hlm. 259-264, 2023.
- [9] M. A. Fitria dan D. C. Febrianto, "Penerapan sistem pakar untuk diagnosa penyakit dan hama tanaman cabai dengan metode forward chaining," *Sainteks, Lemb. Penelitian dan Pengabdian Masyarakat (LPPM) Universitas Muhammadiyah Purwokerto*, vol. 16, no. 2, 2019.
- [10] B. A. Nusantara dan F. A. Suciono, "Sistem pakar diagnosa hama dan penyakit pada jamur tiram menggunakan metode forward chaining," *Journal of Information Technology and Computer Science*.
- [11] Muliadi, I. Budiman, M. A. Pratama, dan A. Sofyan, "Fuzzy dan Dempster-Shafer pada sistem pakar diagnosa penyakit tanaman cabai," *Kumpulan J. Ilmu Komputer*, vol. 4, no. 2, hlm. 209-222, 2017.
- [12] E. H. Shortliffe dan B. G. Buchanan, "A model of inexact reasoning in medicine," *Math. Biosci.*, vol. 23, no. 3-4, hlm. 351-379, 1975.
- [13] F. Ramadhan dkk., "Rancang bangun sistem informasi pertanian berbasis web menggunakan JavaScript dan JSON," *J. Teknol. dan Manajemen Inf.*, vol. 9, no. 2, hlm. 112-121, 2023.
- [14] F. C. Hudi dan C. M. Karyanti, "Pengujian fungsionalitas sistem informasi berbasis web menggunakan metode user acceptance testing," *J. Ilm. Komputasi*, vol. 22, no. 4, hlm. 553-560, 2024.
- [15] S. Aisah, A. S. Yazid, D. P. Wijaya, dan D. Dianti, "Sistem Pakar Diagnosa Hama dan Penyakit Tanaman Cabai Menggunakan Metode Certainty Factor Berbasis Web," *Jurnal Informatika Teknologi dan Sains (Jinteks)*, vol. 8, no. 1, hlm. 1-9, Feb. 2026.