

Case-Based Reasoning Diagnosis Penyakit Pada Tanaman Kelapa Sawit

Tursina, Rina Septiriana, Nella Yuliarni*

Program Studi Informatika, Fakultas Teknik, Universitas Tanjungpura
Pontianak, Kalimantan Barat

¹tursina@informatika.untan.ac.id, ²rinaseptiriana@informatika.untan.ac.id, ³nellayuliarni12@student.untan.ac.id

Abstract -Oil palm cultivation is one of the main agricultural commodities in Indonesia, as oil palm plants have a relatively high economic value. However, oil palm cultivation is vulnerable to disease attacks that can threaten productivity. The lack of knowledge among farmers in managing plantations and addressing these problems can exacerbate damage to plants due to disease attacks. To address this issue, this research will develop an application using a Case-Based Reasoning (CBR) approach to diagnose diseases in oil palm plants with the nearest neighbor calculation algorithm. CBR will solve new problems by reusing knowledge to solve old problems that are similar and already have solutions. The nearest neighbor algorithm is useful for calculating similarity between new cases and old cases. The data for the research involves 7 types of diseases, 20 symptoms, and 345 cases of disease attacks on oil palm plants. The application testing uses 25 test cases and 320 case bases. From the test results, the application shows an accuracy rate of 31.00, with the highest similarity value reaching 0.98. In this research, it can be seen that the generated similarity values are high but the accuracy value is low. This can be caused by several factors, such as the data of the cases, feature representation, and insufficient diagnosis, which result in high similarity values due to the assignment of values to attribute categories and attribute value proximity.

Keywords: case-based reasoning, nearest neighbor algorithm, oil palm, disease

Abstrak--Budidaya tanaman kelapa sawit adalah salah satu komoditas pertanian utama di Indonesia, hal tersebut terjadi karena tanaman kelapa sawit memiliki nilai jual ekonomi yang cukup tinggi. Namun Budidaya tanaman kelapa sawit rentan terhadap serangan penyakit yang dapat mengancam produktivitas. Kurangnya pengetahuan para petani dalam mengelola kebun dan mengatasi masalah tersebut dapat memperparah kerusakan tanaman akibat serangan penyakit. Untuk mengatasi permasalahan tersebut pada penelitian ini akan membangun sebuah aplikasi dengan pendekatan Case-Based Reasoning (CBR) untuk mendiagnosis penyakit pada tanaman kelapa sawit dengan algoritma perhitungan nearest neighbor. CBR akan menyelesaikan permasalahan baru dengan memanfaatkan kembali pengetahuan dalam menyelesaikan permasalahan lama yang memiliki kesamaan dan telah memiliki solusi. Algoritma nearest neighbor berguna untuk menghitung similaritas atau kemiripan antar kasus baru dengan kasus lama. Data untuk penelitian melibatkan 7 jenis penyakit, 20 gejala, dan 345 kasus serangan penyakit pada tanaman kelapa sawit. Pengujian aplikasi menggunakan 25 kasus uji serta 320 basis kasus. Dari hasil pengujian aplikasi menunjukkan hasil akurasi sebesar 31,00, dengan nilai similaritas tertinggi mencapai 0,98. Pada penelitian ini dapat dilihat nilai similaritas yang dihasilkan tinggi tetapi nilai akurasi rendah hal ini dapat disebabkan oleh beberapa faktor, seperti data kasus yang dimiliki, representasi fitur, serta diagnosis yang kurang, dan yang menyebabkan nilai similaritas tinggi dapat dikarenakan pemberian nilai pada kategori nilai pada atribut dan kedekatan nilai atribut.

Kata kunci: case-based reasoning, algoritma nearest neighbor, kelapa sawit, penyakit

I. PENDAHULUAN

Tanaman kelapa sawit ditanam di hampir seluruh wilayah di Indonesia terkecuali wilayah Jawa Timur, Jawa Tengah, Bali, dan Nusa Tenggara [1]. Tanaman perkebunan kelapa sawit memiliki nilai jual yang sangat baik hal tersebut terjadi dikarenakan tanaman kelapa sawit banyak diminati oleh para investor [2].

Budidaya tanaman kelapa sawit terkadang mengalami penurunan, hal ini dapat disebabkan oleh ketidaktahuan para petani dalam mengelola

kebun dan tanaman kelapa sawit dengan benar [3]. Salah satu aspek yang dapat menyebabkan penurunan produktivitas tanaman kelapa sawit, yaitu karena terserang penyakit. Penyakit tanaman kelapa sawit biasanya melibatkan infeksi atau gangguan pada tingkat sel atau jaringan tanaman[4]. Jika gejala penyakit yang dialami tanaman kelapa sawit tidak ditanggulangi dengan benar khawatirnya akan menimbulkan penambahan biaya perawatan dan kerusakan yang lebih parah pada tanaman [5]. Untuk menghadapi permasalahan dalam penanganan penyakit yang menyerang tanaman kelapa sawit,

para petani biasanya saling bertukar informasi mengenai gejala dan cara penanganan gejala tersebut. Namun dalam penyebaran informasi mengenai pengalaman-pengalaman tersebut masih sulit dan juga terdapat kesulitan dalam bertemu dengan ahli dalam bidang penanganan serangan penyakit khususnya pada tanaman kelapa sawit.

Informasi mengenai pengalaman-pengalaman atau pengetahuan para petani dalam menangani gejala yang timbul diakibatkan serangan penyakit pada tanaman kelapa sawit dapat disimpan dalam sebuah aplikasi, teknik untuk menyimpan pengetahuan tersebut disebut *Case-Based Reasoning* (CBR) atau penalaran berbasis kasus [6][20]. CBR ialah sebuah teknik yang digunakan sebagai pemecahan masalah atau kasus baru dengan menyesuaikan solusi masalah pada kasus sebelumnya [7][8]. Penyelesaian masalah menggunakan pendekatan CBR memiliki empat tahapan utama, tahapan tersebut dikenal dengan 4R (*Retrieve, Reuse, Revise, dan Retain*) [9]. *Retrieve* merupakan tahapan di mana kasus lama dalam basis kasus yang serupa dengan kasus baru diambil dan dibandingkan. Tahapan *reuse* merupakan tahapan di mana pengetahuan dari kasus sebelumnya digunakan kembali untuk menyelesaikan masalah atau kasus baru. Tahapan *revise* merupakan tahapan di mana solusi yang diusulkan direvisi apabila diperlukan. Dan yang terakhir tahapan *retain* di mana bagian-bagian dari pengalaman baru dipertahankan untuk pemecahan masalah di masa yang akan datang.

Pada tahapan *retrieve* saat mengambil (perbandingan) kasus baru dengan basis kasus diperlukan perhitungan untuk menentukan nilai kemiripan atau similaritas kasus baru dengan kasus-kasus sebelumnya yang dapat menggunakan beberapa algoritma, salah satu yang dapat digunakan yaitu Algoritma *Nearest Neighbor*. Algoritma *nearest neighbor* adalah algoritma untuk menemukan kemiripan antara kasus baru dan kasus sebelumnya dengan memperhatikan kesesuaian bobot atribut/fitur yang ada [10]. Jika permasalahan baru dan permasalahan masa lalu (basis kasus) memiliki kesamaan yang besar, maka solusi dari kasus masa lalu tersebut berpotensi untuk diterapkan dalam menangani permasalahan baru [19][20].

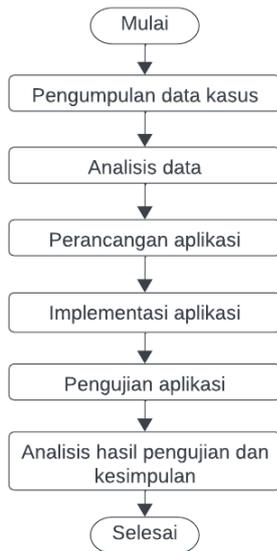
Berikut beberapa penelitian yang pernah dilakukan sebelumnya yang berhubungan dengan *case-based reasoning*. Penelitian yang dilakukan oleh [5] memiliki objek dan metode penelitian yang sama namun pada penelitian ini akan menggunakan data kasus yang berbeda, penentuan bobot yang

berbeda, serta jumlah penyakit yang di diagnosis juga berbeda. Dari hasil penelitian [5] menghasilkan aplikasi yang dapat mendiagnosis penyakit pada tanaman kelapa sawit. Dari hasil pengujian kasus yang dibuat kemiripan yang dihasilkan sebesar 0.78. Selanjutnya penelitian [11] hasil pada penelitian tersebut menghasilkan nilai akurasi sistem sebesar 80% dengan objek penelitian hama dan penyakit tanaman karet. Penelitian [12] pada penelitian tersebut menghasilkan tingkat akurasi 80%, dengan objek penelitian mengenai hama dan penyakit pada tanaman anggrek. Selanjutnya penelitian [13] penelitian tersebut menghasilkan tingkat akurasi 80%, dengan objek penelitian diagnosis hama dan penyakit tanaman nilam. Dari ketiga penelitian yang pernah dilakukan dapat dilihat bahwa penelitian dengan penerapan pendekatan CBR dapat mengatasi permasalahan-permasalahan dengan objek penelitian yang berbeda-beda. Hal ini menunjukkan fleksibilitas CBR untuk diterapkan pada berbagai konteks masalah. Hasil penelitian sebelumnya menunjukkan tingkat akurasi yang dihasilkan cukup baik hal ini membuktikan bahwa penerapan CBR mampu memberikan solusi atau rekomendasi yang sesuai dengan kasus-kasus yang terjadi sebelumnya dengan nilai rata-rata sekitar 80%. Sehingga penelitian mengenai CBR dalam konteks pertanian, terutama untuk mengatasi masalah diagnosis tanaman yang terserang penyakit dapat dilakukan atau diterapkan.

Aplikasi ini akan menghasilkan *output* atau keluaran berupa rekomendasi diagnosis penyakit yang menyerang tanaman kelapa sawit dan solusi penanganan pada kasus tersebut yang berdasarkan dari hasil *input* atau masukan gejala yang dilakukan oleh pengguna aplikasi.

II. METODE PENELITIAN

Alur penelitian yang digunakan pada penelitian ini dapat dilihat pada Gambar 1.



Gambar 1 Alur Penelitian

Adapun penjelasan dari alur penelitian yang akan dilakukan:

i) Pengumpulan data kasus: pada tahap pengumpulan data kasus akan dikumpulkan kasus-kasus yang berkaitan dengan diagnosis penyakit yang menyerang tanaman kelapa sawit. Data kasus tersebut didapat dari hasil wawancara dan observasi dengan petani kepala sawit.

ii) Analisis data: pada tahapan ini dilakukan dengan menganalisis data-data yang diperlukan untuk membuat aplikasi. Data-data tersebut yaitu jenis penyakit, gejala penyakit, dan solusi penanganan penyakit.

iii) Perancangan aplikasi: pada tahap perancangan aplikasi diawali dengan menentukan fitur berdasarkan data-data kasus yang telah dikumpulkan pada tahapan sebelumnya. Fitur yang digunakan dalam mendiagnosis penyakit yang menyerang tanaman kelapa sawit adalah berdasarkan dari gejala-gejala yang tampak pada bagian akar, buah, daun, dan batang tanaman kelapa sawit. Setelah itu dilakukan penentuan bobot dan nilai kedekatan atribut pada setiap fitur sesuai dengan algoritma yang digunakan (*nearest neighbor*) untuk menghitung nilai similaritas antara kasus baru dengan kasus sebelumnya. algoritma NN merupakan teknik yang paling banyak digunakan dengan CBR. Algoritma NN bekerja dengan cara menentukan kemiripan kasus baru (target) dengan fitur-fitur kasus pada basis kasus. Kemudian jumlah kemiripan seluruh fitur dihitung untuk memberikan ukuran kemiripan (similaritas) kasus di basis kasus dengan kasus baru [16][17]. Kemiripan kasus biasanya

dinilai dalam rentang 0 hingga 1. Nilai 0 mengindikasikan kedua kasus tersebut tidak sama secara mutlak, sementara nilai 1 mengindikasikan kedua kasus sama secara mutlak [14]. Berikut adalah rumus yang digunakan [18]:

$$Similarity(p, q) = \frac{\sum_{i=1}^n f(p_i, q_i) \times w_i}{w_i} \quad (1)$$

Keterangan rumus :

p : Merujuk pada kasus baru.

q : Mengacu pada kasus lama yang ada.

n : Menunjukkan jumlah atribut setiap kasus.

i : Merupakan atribut individu yang berkisar dari 1 hingga n.

f : Adalah fungsi kesamaan dari atribut i kasus p dan kasus q.

w : Adalah bobot yang diberikan pada atribut ke-i.

Selanjutnya dilanjutkan perancangan alur kerja CBR dalam aplikasi dan alur kerja dari aplikasi tersebut.

iv) Implementasi : setelah melakukan perancangan maka akan dilakukan tahap implementasi dengan pembuatan antarmuka aplikasi dan pemrograman aplikasi dengan penerapan konsep CBR. Untuk alur dari penerapan konsep CBR pada aplikasi dapat dilihat pada Gambar 2 alur kerja aplikasi yang menjelaskan dari awal data kasus baru di-*input*-kan - proses *retrieve* - proses *reuse* - proses *revise* - proses *retain*.

v) Pengujian aplikasi: pada tahap pengujian ini akan dilakukan pengecekan terhadap aplikasi yang dibangun apakah dapat melakukan keseluruhan tahapan CBR. Selain itu, pengujian dilakukan dengan membandingkan hasil diagnosis oleh pakar dengan hasil yang dihasilkan oleh aplikasi.

vi) Analisis hasil pengujian dan kesimpulan : pada tahap ini akan dilakukan analisis mengenai aplikasi yang telah dibuat dan dilakukan pengambilan kesimpulan yang dirumuskan berdasarkan tahapan-tahapan yang telah dilakukan sebelumnya apakah aplikasi yang dirancang dan dibangun menggunakan CBR mampu mendiagnosis penyakit yang menyerang tanaman kelapa sawit.

III. HASIL DAN PEMBAHASAN

A. Pengumpulan data kasus

Data kasus yang digunakan pada penelitian ini didapatkan dari hasil wawancara dengan Bapak Herkulanus, S.P (petani kelapa sawit), studi literatur, dan observasi langsung ke kebun milik narasumber, Bapak Herkulanus, S.P. Dari hasil pengumpulan data kasus yang telah dilakukan terkumpul sebanyak tujuh jenis penyakit, dua puluh gejala untuk

penyakit, dan 345 kasus yang disebabkan oleh penyakit. Dari 345 kasus yang ada, 25 kasus dipilih secara acak dan digunakan sebagai kasus uji dan 320 kasus digunakan sebagai basis kasus.

B. Analisis Data

Dalam membangun aplikasi dengan penerapan CBR, tahapan pertama yang dilakukan adalah menentukan fitur kasus yang akan digunakan. Fitur yang akan digunakan pada aplikasi CBR diagnosis penyakit pada tanaman kelapa sawit terdiri dari 20 fitur (gejala), fitur-fitur tersebut ialah:

1. Akar membusuk (F01).
2. Pada pangkal batang terbentuk badan buah jamur warna orange muda-coklat (F02).
3. Getah keluar dari tempat yang terinfeksi (F03).
4. Tanaman tumbuh tidak normal dan lemah (F04).
5. Terdapat benang-benang jamur yang berwarna putih mengkilap pada tandan buah (F05).
6. Buah busuk (F06).
7. Daun tua menjadi layu dan patah (F07).
8. Daun tua menjadi layu (F08).
9. Daun patah (F09).
10. Daun muda banyak yang tidak membuka (F10).
11. Daun berwarna hijau pucat (F11).
12. Jaringan pada kuncup membusuk (F12).
13. Kuncup berwarna kecoklatan (F13).
14. Di tepi dan ujung daun terdapat bercak-bercak coklat tua (F14).
15. Bercak-bercak kuning pada daun (F15).
16. Helai daun mulai dari pertengahan sampai ujung pelepah kecil-kecil, atau tidak ada sama sekali (F16).
17. Daun-daun bengkok (F17).
18. Tampak bercak-bercak lonjong berwarna kuning pada daun dan ditengahnya terdapat warna coklat (F18).
19. Daun tampak kering (F19).
20. Daun kekuning-kuningan (F20).

Berdasarkan hasil wawancara dari 20 data fitur (gejala) yang digunakan semua memiliki nilai sama penting (1). hal ini dikarenakan setiap fitur memberikan kontribusi yang setara terhadap hasil diagnosis. Data yang digunakan untuk mendiagnosis penyakit terdiri dari tujuh jenis penyakit. Penyakit-penyakit tersebut ialah :

1. Penyakit busuk tandan (*bunch rot*)
2. Penyakit busuk pangkal batang (*basal stem rot* atau *ganoderma*)
3. Penyakit busuk kuncup (*spear rot*)
4. Penyakit tajuk (*crown disease*)

5. Penyakit garis kuning (*patch yellow*)
6. Penyakit akar (*blast disease*)
7. Penyakit anthracnose

Selanjutnya kasus-kasus yang sudah dikumpulkan akan direpresentasikan ke dalam bentuk tabel, dapat dilihat pada Tabel I di mana terdapat 20 fitur yaitu F01 sampai F20 dan 320 kasus yaitu K01 sampai K320.

TABEL I
REPRESENTASI BASIS KASUS PENYAKIT DAN KASUS BARU
PENYAKIT

Kasus	Gejala							
	F01	F02	F05	...	F17	F18	F19	F20
K01	1	1	1	..	0	0	0	0
K02	1	1	0	..	0	0	0	0
K03	1	0	1	..	0	0	0	0
K04	1	0	1	..	0	0	0	0
K05	1	0	0	..	0	0	0	0
...
K316	0	0	0	...	0	0	0	0
K317	0	0	0	...	1	0	0	0
K318	0	0	0	...	0	1	0	0
K319	0	0	0	...	0	0	1	0
K320	0	0	0	...	0	0	0	1
Kbaru	1	0	1	...	0	0	0	0

Keterangan :

0 yang mengindikasikan bahwa gejala tersebut tidak terlihat.

1 yang menunjukkan bahwa gejala tersebut terlihat.

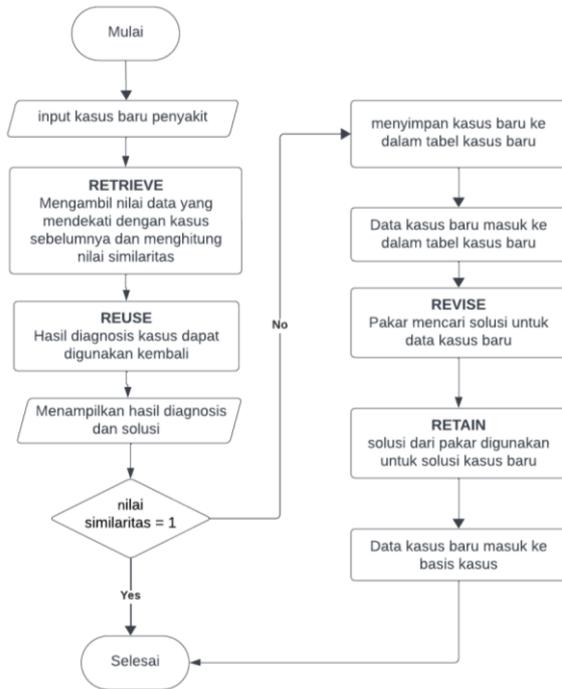
Kode kasus : K01, K02, K03, K04, ..., K320.

C. Perancangan

Pada tahap perancangan akan dilakukan perancangan alur kerja aplikasi, diagram konteks, DFD level 0.

1) Alur Kerja Aplikasi

Perancangan alur kerja aplikasi dengan pendekatan *case-based reasoning* dapat dilihat pada Gambar 2.

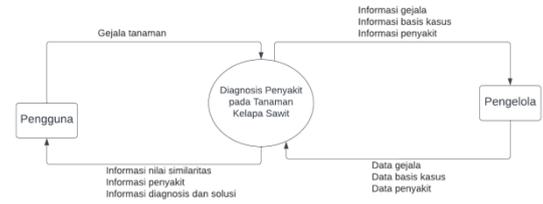


Gambar 2 Alur Kerja Aplikasi

Gambar 2 menunjukkan skema alur kerja aplikasi yang menggunakan pendekatan *case-based reasoning* untuk mendiagnosis penyakit pada tanaman kelapa sawit. Pengguna diminta memasukkan gejala yang muncul pada bagian tertentu dari tanaman (akar, batang, daun, atau buah). Aplikasi kemudian mencari data kasus serupa di dalam basis kasus, menilai tingkat kemiripan dengan kasus baru yang dimasukkan. Pada tahap ini akan menggunakan algoritma *nearest neighbor* untuk menghitung nilai kemiripan kasus di mana setiap kasus dalam basis kasus akan dibandingkan dan dihitung kemiripannya dengan kasus baru, yang kemudian diurutkan dari nilai terendah hingga tertinggi [15]. Kasus dengan nilai similaritas tertinggi dianggap paling mirip dengan kasus baru. Nilai similaritas berkisar antara 0 hingga 1, di mana nilai 1 menandakan kemiripan sempurna. Jika nilai similaritas adalah 1, kasus baru dianggap sudah ada dalam basis dan dapat dihapus. Namun, jika nilai similaritas kurang dari 1, kasus baru akan disimpan dalam tabel kasus baru. Selanjutnya adalah tahap *revise*, di mana kasus baru akan ditinjau oleh seorang pakar. Setelah ditinjau, solusi untuk kasus baru akan dicari dan disimpan kembali ke dalam basis kasus, tahap ini disebut tahap *retain*. Hasil akhir dari aplikasi ini berupa diagnosis penyakit pada tanaman kelapa sawit beserta saran penanganannya.

2) Diagram Konteks Aplikasi

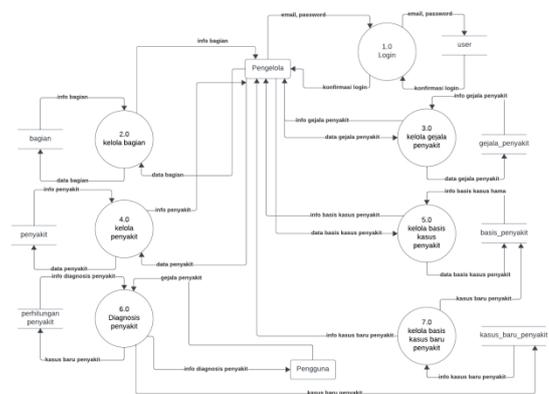
Diagram konteks adalah representasi grafis yang menunjukkan asal dan tujuan data yang akan diolah dalam suatu aplikasi.



Gambar 3 Diagram Konteks Aplikasi

Pada Gambar 3 dapat dilihat aplikasi ini terdapat dua *entitas*, yaitu pengguna dan pengelola. Pengguna dapat melakukan *input* gejala yang dialami tanaman dan mendapatkan informasi mengenai diagnosis penyakit yang menyerang tanaman, solusi penanganan, dan nilai similaritas. Pengelola memiliki peran untuk mengelola data yang terdapat dalam aplikasi seperti data basis kasus, bagian tanaman, kasus baru, penyakit, dan gejala penyakit.

3) Data Flow Diagram Level 0



Gambar 4 DFD Level 0 Aplikasi

Pada Gambar 4 adalah diagram level 0 dari aplikasi yang akan dibangun, terdapat dua eksternal yaitu pengguna dan pengelola. Terdapat tujuh proses yang akan terjadi pada aplikasi, proses *login*, proses kelola bagian tanaman, proses kelola gejala penyakit, proses kelola jenis penyakit, proses kelola basis kasus penyakit, proses diagnosis penyakit, dan proses kelola basis kasus baru penyakit, serta terdapat tujuh *data store* yaitu *user*, bagian, gejala penyakit, penyakit, basis penyakit, perhitungan penyakit, dan kasus baru penyakit.

D. Implementasi (hasil perancangan)

Halaman beranda admin adalah halaman pertama yang tampil saat admin berhasil *login* aplikasi. Daftar menu yang tersedia di halaman admin dapat ditemukan dalam Tabel II, dan tampilan antarmuka halaman admin pada Gambar 5.

TABEL II

DAFTAR MENU PENGELOLA DAN FUNGSI

Menu	Sub Menu	Fungsi
Penyakit	-	Menu ini berisi kode, nama, dan solusi dari hama. Pada menu ini terdapat fungsi tambah, edit, dan hapus data hama serta solusi nya.
Gejala	Penyakit	Menu ini berisi kode dan keterangan gejala penyakit. Pada menu ini terdapat fungsi tambah, edit, dan hapus data gejala.
	Bagian	Menu ini berisi nama bagian kelapa sawit yang terserang penyakit. Pada menu ini terdapat fungsi tambah, edit, dan hapus.
Basis Kasus	Penyakit	Menu ini menampilkan basis kasus penyakit.
	Gejala Penyakit	Digunakan untuk menambah, mengedit dan menghapus basis kasus penyakit.
Kasus Baru	Penyakit	Menu ini digunakan untuk melakukan proses revise, retain, dan hapus pada kasus baru.
	Beranda User	Menu ini digunakan untuk mengedit halaman awal pengguna.
Pengaturan	Kelapa Sawit	Menu ini digunakan untuk menambah, mengedit, dan menghapus isi halaman kelapa sawit yang ada di tampilan pengguna.



Gambar 5 Halaman Uama Pengelola

Halaman awal pengguna adalah halaman pertama yang ditampilkan saat pengguna membuka aplikasi diagnosis penyakit pada tanaman kelapa sawit. Pada halaman awal pengguna, terdapat empat opsi menu, yaitu beranda, kelapa sawit, penyakit, dan tentang saya. Fungsi dari setiap menu dapat ditemukan dalam Tabel III dan dapat dilihat pada Gambar 6.

TABEL III

MENU PENGGUNA DAN FUNGSI

Nama Menu	Keterangan/Fungsi
Beranda	Halaman utama pengguna.
Kelapa Sawit	Halaman ini berisi pengetahuan singkat mengenai tanaman kelapa sawit
Penyakit	Halaman ini berguna untuk melakukan diagnosis penyakit yang menyerang tanaman kelapa sawit.
Tentang Saya	Melihat biodata singkat tentang pengembang aplikasi



Gambar 6 Halaman Utama Pengguna

E. Pengujian

Pada pengujian ini akan dilakukan pengecekan terhadap aplikasi yang telah dibangun apakah dapat melakukan keseluruhan tahapan *case based reasoning* yaitu tahapan *retrieve*, tahapan *reuse*, tahapan *revise* dan tahapan *retain* sesuai dengan tahapan yang ada.

Selanjutnya pengujian kedua dilakukan dengan memperhatikan kesesuaian hasil diagnosis yang telah dilakukan oleh pakar dan yang dihasilkan oleh aplikasi terhadap 25 kasus uji penyakit. Pengujian perbandingan ini dilakukan untuk mengetahui keberhasilan aplikasi dalam mendiagnosis penyakit yang menyerang tanaman kelapa sawit.

1. Pengujian tahapan CBR

Pada pengujian tahapan *case-based reasoning* dilakukan dengan pengecekan seluruh tahapan (*retrieve - reuse - revise - retain*) yang ada di aplikasi dengan cara memasukkan kasus baru dan melihat alur kerja tahapan apakah sudah sesuai atau belum.

Tahap *retrieve* diuji dengan cara melacak dan mengambil kasus-kasus yang mirip. Hal ini dilakukan dengan membandingkan kasus *input* dengan sejumlah kasus yang ada dalam basis kasus. Proses yang terdapat pada tahapan *retrieve*, yaitu: pengguna meng-*input*-kan data gejala penyakit (F01, F03, F05, dan F07) dan menekan tombol diagnosis. Gambar 7 merupakan proses *input* kasus baru.

- N = Bobot atribut F07 → 1
- O = Kedekatan F08 (kasus baru : 0, & K76: 0) → 1
- P = Bobot atribut F08 → 1
- Q = Kedekatan F09 (kasus baru : 0, & K76:0) → 1
- R = Bobot atribut F09 → 1
- S = Kedekatan F10 (kasus baru : 0, & K76: 0) → 1
- T = Bobot atribut F10 → 1
- U = Kedekatan F11 (kasus baru : 0, & K76: 0) → 1
- V = Bobot atribut F11 → 1
- W = Kedekatan F12 (kasus baru : 0, & K76: 0) → 1
- X = Bobot atribut F12 → 1
- Y = Kedekatan F13 (kasus baru : 0, & K76: 0) → 1
- Z = Bobot atribut F13 → 1
- AA = Kedekatan F14 (kasus baru : 0, & K76: 0) → 1
- AB = Bobot atribut F14 → 1
- AC = Kedekatan F15 (kasus baru : 0, & K76: 0) → 1
- AD = Bobot atribut F15 → 1
- AE = Kedekatan F16 (kasus baru : 0, & K76: 0) → 1
- AF = Bobot atribut F16 → 1
- AG = Kedekatan F17 (kasus baru : 0, & K76: 0) → 1
- AH = Bobot atribut F17 → 1
- AI = Kedekatan F18 (kasus baru : 0, & K76: 0) → 1
- AJ = Bobot atribut F18 → 1
- AK = Kedekatan F19 (kasus baru : 0, & K76:0) → 1
- AL = Bobot atribut F19 → 1
- AM = Kedekatan F20 (kasus baru : 0, & K76: 0) → 1
- AN = Bobot atribut F20 → 1

$$Sim(p, q) = \frac{\sum_{i=1}^n f(p_i, q_i) \times w_i}{w_i}$$

$$Sim(p, q) = \frac{19.5}{20} = 0.98$$

2. Kasus baru penyakit dan basis kasus penyakit (K148)

Berikut perhitungan nilai similaritas:

$$p = [1,0,1,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

$$q = [1,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

$$Sim(p, q) = \frac{\sum_{i=1}^n f(p_i, q_i) \times w_i}{w_i}$$

$$Sim(p, q) = \frac{19.5}{20} = 0.98$$

3. Kasus baru penyakit dan basis kasus penyakit (K178)

Berikut perhitungan nilai similaritas:

$$p = [1,0,1,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

$$q = [1,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

$$Sim(p, q) = \frac{\sum_{i=1}^n f(p_i, q_i) \times w_i}{w_i}$$

$$Sim(p, q) = \frac{19.5}{20} = 0.98$$

4. Kasus baru penyakit dan basis kasus penyakit (K181)

Berikut perhitungan nilai similaritas:

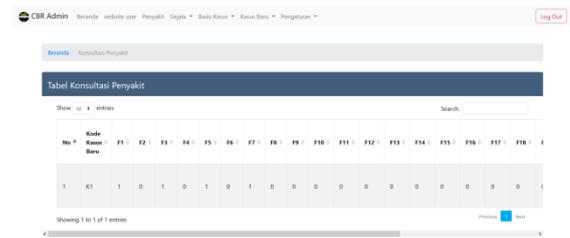
$$p = [1,0,1,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

$$q = [1,0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

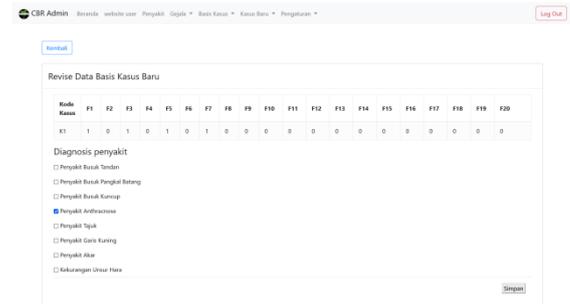
$$Sim(p, q) = \frac{\sum_{i=1}^n f(p_i, q_i) \times w_i}{w_i}$$

$$Sim(p, q) = \frac{19.5}{20} = 0.98$$

Selanjutnya pada tahap *revise* kasus baru yang di-*input*-kan tadi akan disimpan ke *database* sebagai kasus baru yang akan diproses pada tahap *revise*, jika nilai kurang dari satu. Pada tahapan *revise* akan dilakukan perubahan terhadap solusi yang dihasilkan aplikasi dan akan diperbaiki oleh pakar. Pada tahap ini hanya mengubah hasil diagnosis tanpa mengganti fitur (gejala) yang sudah ada. Tahap *revise* kasus baru pada Gambar 9 dan Gambar 10.

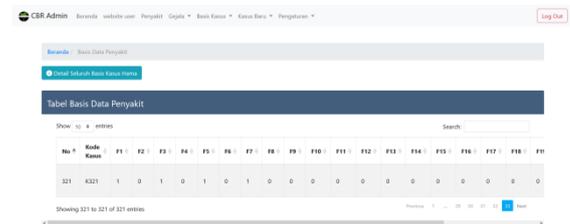


Gambar 9 Data tersimpan di basis kasus baru



Gambar 1 Tahapan *revise*

Pada tahapan ini proses menggabungkan dan menyimpan solusi kasus baru dengan diagnosis yang sudah diperbaiki ke dalam basis kasus penyakit yang ada. Tahapan *retain* dilihat di Gambar 11.



Gambar 2 Tahapan *retain*

Hasil uji tahap *case-based reasoning* menunjukkan aplikasi memiliki kemampuan dalam melaksanakan 4R.

2. Hasil Perbandingan Antara Hasil Pakar Dengan Hasil Aplikasi

Pengujian kedua dengan membandingkan hasil yang diberikan oleh pakar dengan hasil diagnosis aplikasi. Hasil perbandingan kesesuaian diagnosis dapat dilihat pada Tabel VI dan Tabel VII.

TABEL VI
HASIL PENGUJIAN KASUS 1

Kode Baru	Kasus	Gejala	Diagnosis	Sim	Kesesuaian Hasil
	F01, F03, F05, F07				
Hasil Pakar/Fakta	penyakit akar, penyakit busuk pangkal batang, dan penyakit busuk tandan				
Hasil Aplikasi					
No.	Kode Basis Kasus	Gejala	Diagnosis	Sim	Kesesuaian Hasil
1.	K76	F03, F05, F07	penyakit busuk tanda, penyakit tajuk, dan penyakit busuk pangkal batang	0,98	Tidak Sesuai
2.	K148	F01, F05, F07	penyakit akar, penyakit tajuk, dan penyakit busuk tandan	0,98	Tidak Sesuai
3.	K178	F01, F03, F05	penyakit busuk pangkal batang, penyakit tajuk, dan penyakit busuk tandan	0,98	Sesuai
4.	K181	F01, F03, F07	penyakit busuk pangkal batang, dan penyakit akar	0,98	Tidak Sesuai

Pada Tabel VI dapat dilihat bahwa hasil kesesuaian kasus yang dihasilkan oleh aplikasi dengan diagnosis pakar terdapat satu kasus yang sesuai dengan diagnosis pakar dan tiga kasus lainnya dikatakan tidak sesuai karena kekurangan/kelebihan satu penyakit dari hasil sesungguhnya. Sehingga akurasi untuk pengujian kasus 1 yaitu $\frac{1}{4} \times 100 = 25$.

TABEL VII
HASIL PENGUJIAN KASUS 2

Kode Baru	Kasus	Gejala	Diagnosis	Sim	Kesesuaian Hasil
	F01, F04, F06, F07				
Hasil Pakar/Fakta	penyakit akar, penyakit tajuk, dan penyakit busuk tanda				
Hasil Aplikasi					

No.	Kode Basis Kasus	Gejala	Diagnosis	Sim	Kesesuaian Hasil
1.	K79	F04, F06, F07	penyakit tajuk dan penyakit busuk tandan	0,98	Tidak Sesuai
2.	K149	F01, F06, F07	penyakit busuk tandan, penyakit akar, dan penyakit tajuk	0,98	Sesuai
3.	K182	F01, F04, F07	penyakit akar dan penyakit tajuk	0,98	Tidak Sesuai

Pada Tabel VII dapat dilihat bahwa hasil kesesuaian kasus yang dihasilkan oleh aplikasi dengan diagnosis pakar terdapat satu kasus yang sesuai dengan diagnosis pakar dan dua kasus lainnya dikatakan tidak sesuai karena kekurangan/kelebihan satu penyakit dari hasil sesungguhnya. Sehingga akurasi untuk pengujian kasus ke-2 yaitu $\frac{1}{3} \times 100 = 33,33$.

F. Analisis Hasil Pengujian

1. Pengujian tahapan CBR

Dari hasil pengujian, keseluruhan tahapan CBR dapat dilakukan dengan sesuai dimulai dari mendapatkan kasus-kasus yang mirip, menggunakan kembali kasus-kasus yang ada dan mencoba untuk menyelesaikan kasus baru, mengubah dan mengadopsi solusi yang ditawarkan (jika perlu), dan menggabungkan kasus baru yang solusinya telah diperbaharui ke dalam basis kasus.

2. Pengujian perbandingan hasil diagnosis

Dari total 25 kasus uji penyakit dapat dilihat hasil perbandingan diagnosis pakar dengan hasil diagnosis aplikasi serta hasil akurasi dari setiap pengujian pada Tabel VIII.

TABEL VIII
HASIL PERHITUNGAN AKURASI SETIAP KASUS UJI

Kasus Uji	Kasus sesuai	Kasus terpilih	Perhitungan	Hasil
K01	1	4	$\frac{1}{4} \times 100$	25
K02	1	3	$\frac{1}{3} \times 100$	33,33
K03	1	5	$\frac{1}{5} \times 100$	20
K04	1	5	$\frac{1}{5} \times 100$	20
K05	1	5	$\frac{1}{5} \times 100$	20
		↓↓↓		
K20	2	3	$\frac{2}{3} \times 100$	66,67
K21	1	3	$\frac{1}{3} \times 100$	33,33
K22	2	7	$\frac{2}{7} \times 100$	28,58

Kasus Uji	Kasus sesuai	Kasus terpilih	Perhitungan	Hasil
K23	1	7	$\frac{1}{7} \times 100$	14,29
K24	2	6	$\frac{2}{6} \times 100$	33,33
K25	1	4	$\frac{1}{4} \times 100$	25
Total				31,00

Keterangan:

- Kasus sesuai = kasus yang memiliki diagnosis yang sama dengan hasil diagnosis pakar (hasil sesungguhnya/fakta).
- Kasus terpilih = kasus yang dikatakan mirip dengan kasus baru (kasus uji) dengan nilai similaritas tertinggi.

Hasil pengujian kesesuaian pakar dan aplikasi CBR dengan menggunakan algoritma *nearest neighbor* pada Tabel VIII dengan basis kasus sebanyak 320 kasus, dari total 25 kasus uji, aplikasi memberikan hasil diagnosis kasus terpilih lebih dari satu basis kasus (kasus terpilih). Namun dari kasus-kasus yang terpilih tersebut hasil diagnosis dengan hasil yang diberikan oleh pakar tidak jauh berbeda. Pada table pengujian terdapat keterangan sesuai dan tidak sesuai. Diagnosis aplikasi dikatakan sesuai apabila hasilnya sama persis seperti diagnosis pakar, sedangkan keterangan tidak sesuai dapat terjadi karena hasil diagnosis aplikasi kekurangan atau kelebihan satu jenis penyakit yang sesuai dengan diagnosis pakar. Hasil akurasi yang dihasilkan aplikasi dalam mendiagnosis penyakit kelapa sawit menghasilkan tingkat akurasi sebesar 31,00 dengan nilai similaritas tertinggi yang dihasilkan sebesar 0,98. Pada penelitian ini dapat dilihat nilai similaritas yang dihasilkan tinggi tetapi nilai akurasi rendah hal ini berarti meski pun kasus yang sedang diproses memiliki kemiripan yang tinggi dengan kasus-kasus yang ada dalam basis kasus, namun pendekatan CBR pada kasus ini tidak dapat menghasilkan solusi atau prediksi atau rekomendasi yang akurat atau sesuai dengan keadaan sebenarnya. Tingkat akurasi yang rendah dapat disebabkan oleh beberapa faktor, seperti data kasus yang dimiliki, representasi fitur, serta diagnosis yang kurang, dan yang menyebabkan nilai similaritas tinggi dapat dikarenakan pemberian nilai pada kategori nilai pada atribut dan kedekatan nilai atribut, kedua hal ini dapat berpengaruh pada perhitungan similaritas menggunakan algoritma *nearest neighbor*. Nilai akurasi dalam penelitian ini digunakan sebagai

IV. KESIMPULAN DAN SARAN

Berdasarkan hasil pengujian, keseluruhan tahapan *case-based reasoning* dapat dilakukan dengan sesuai dimulai dari tahapan *retrieve*, *reuse*, *revise*, dan *retain*. Berdasarkan hasil pengujian perbandingan hasil pakar

dan aplikasi. Aplikasi diagnosis penyakit pada tanaman kelapa sawit dari sebanyak 25 kasus uji penyakit menghasilkan nilai akurasi sebesar 31,00 dengan nilai similaritas tertinggi 0,98. Berdasarkan hasil penelitian yang dilakukan dapat disimpulkan bahwa algoritma *nearest neighbor* yang digunakan dalam aplikasi dapat memberikan hasil diagnosis penyakit pada tanaman kelapa sawit berdasarkan gejala-gejala yang ada. Namun untuk perhitungan kasus yang terpilih dapat diperbaiki lagi agar kasus yang terpilih atau hasil diagnosis hanya menampilkan satu kasus yang benar-benar mirip atau sesuai.

REFERENSI

- [1] Juliardi, S. E., & Fachrudin, H. T. (2022). Penilaian Perkebunan Kelapa Sawit. Merdeka Kreasi Group.
- [2] Fauzi, Y., Widyastuti, Y. E., Satyawibawa, I., & Paeru, R. H. (2012). Kelapa sawit. Penebar Swadaya Grup.
- [3] Rasywir, E., Sinaga, R., & Pratama, Y. (2020). Evaluasi pembangunan sistem pakar penyakit tanaman sawit dengan metode deep neural network (DNN). Jurnal media informatika budidarma, 4(4), 1206-1215.
- [4] Kamus Besar Bahasa Indonesia (KBBI), "Penyakit". [Online]. Available : <https://kbbi.kemdikbud.go.id/entri/penyakit>. [Accessed, 17 April 2023].
- [5] Sihaloho, T. P., Tarigan, W., Siallagan, S., & Simbolon, F. H. (2022). Model Case Based Reasoning Dalam Mendiagnosa Penyakit Kelapa Sawit. Jurnal Mnemonic, 5(2), 178-183.
- [6] Riyanto, I. R. (2021). Diagnosa Penyakit Hepatitis Menggunakan Metode Sorensen Coefficient. JATISI (Jurnal Teknik Informatika dan Sistem Informasi), 8(3), 1307-1321.
- [7] Pal, S. K., Dillon, T. S., & Yeung, D. S. (Eds.). (2012). Soft computing in case based reasoning. Springer Science & Business Media.
- [8] López, B. (2022). Case-based reasoning: a concise introduction. Springer Nature.
- [9] Warman, I. (2019, March). Implementation of case-based reasoning and nearest neighbor similarity for peanut disease diagnosis. In Journal of Physics: Conference Series (Vol. 1196, No. 1, p. 012053). IOP Publishing.
- [10] Kusriani & Emha Taufiq Luthfi. (2009). Algoritma Data Mining. Yogyakarta : C.V Andi Offset.
- [11] Sulistiani, H., Darwanto, I., & Ahmad, I. (2020). Penerapan Metode Case Based Reasoning dan K-Nearest Neighbor untuk

- Diagnosa Penyakit dan Hama pada Tanaman Karet. JEPIN (Jurnal Edukasi Dan Penelitian Informatika), 6(1), 23-28.
- [12] Aldo, D., Nur, Y. S. R., Lanyak, A. C. F., Hulqi, F. Y. A., & Hikmah, R. N. (2022). Penerapan Metode Case Base Reasoning Dalam Diagnosa Penyakit dan Hama pada Tanaman Hortikultura. *Building of Informatics, Technology and Science (BITS)*, 4(2), 1111-1122.
- [13] Adawiyah, R. (2018). Case based reasoning diagnosis hama dan Penyakit Tanaman Nilam. *INTENSIF: Jurnal Ilmiah Penelitian dan Penerapan Teknologi Sistem Informasi*, 2(1), 57-67.
- [14] Ikhsan, Z., Yaherwandi, Y., Efendi, S., Rezki, D., Umami, I. M., & Suhendra, D. (2020). Pemberdayaan masyarakat Nagari Silago Kabupaten Dharmasraya melalui teknologi budidaya tanaman kelapa sawit. *Jurnal Hilirisasi IPTEKS*, 3(1), 10-19.
- [15] Merawati, N. L. P., & Hartati, S. (2018). Sistem rekomendasi topik skripsi menggunakan metode case based reasoning. *Jurnal Ilmiah Teknologi Infomasi Terapan*, 4(3).
- [16] Vedayoko, L. G., Sugiharti, E., & Muslim, M. A. (2017). Expert system diagnosis of bowel disease using case based reasoning with nearest neighbor algorithm. *Scientific Journal of Informatics*, 4(2), 134-142.
- [17] Simon, H., Simon, M., & Ajayi, I. H. (2019). Artificial intelligence model for control of plants diseases using case-based reasoning with nearest neighbor technique. *Int J Eng Sci Comput*, 9(9), 23780-23784.
- [18] Warman, I. (2019, March). Implementation of case-based reasoning and nearest neighbor similarity for peanut disease diagnosis. In *Journal of Physics: Conference Series* (Vol. 1196, No. 1, p. 012053). IOP Publishing.
- [19] Zhai, Z., Martínez, J. F., Martínez, N. L., & Díaz, V. H. (2020). Applying case-based reasoning and a learning-based adaptation strategy to irrigation scheduling in grape farming. *Computers and Electronics in Agriculture*, 178, 105741.
- [20] Smiti, A., & Nssibi, M. (2020). Case Based Reasoning Framework for COVID-19 Diagnosis. *Ingénierie des Systèmes d Inf.*, 25(4), 469-474.