



Department of Digital Business

**Journal of Artificial Intelligence and Digital Business (RIGGS)**

Homepage: <https://journal.ilmudata.co.id/index.php/RIGGS>

Vol. 5 No. 1 (2026) pp: 11743-11751

P-ISSN: 2963-9298, e-ISSN: 2963-914X

---

## Pengembangan Sistem Pakar Berbasis Web Untuk Diagnosa Dan Pengendalian Hama Wereng Batang Coklat Pada Tanaman Padi Dengan Menggunakan Metode *Backward Chaining*

Dimas Setiawan<sup>1</sup>, Willis Puspita Sari<sup>2</sup>

<sup>1,2</sup> Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pamulang

[dimassetiawan2137@gmail.com](mailto:dimassetiawan2137@gmail.com), [dosen02527@unpam.ac.id](mailto:dosen02527@unpam.ac.id)

### Abstrak

Wereng Batang Coklat (WBC) merupakan salah satu hama paling merugikan pada tanaman padi di Indonesia, dengan potensi kerusakan yang bisa mencapai gagal panen total. Petani selama ini masih mengandalkan cara manual dalam mendeteksi serangan hama mengandalkan pengalaman pribadi atau menunggu kedatangan penyuluh pertanian, yang tentu saja memakan waktu dan seringkali terlambat. Penelitian ini bertujuan membangun sistem pakar berbasis web yang mampu membantu petani dan penyuluh dalam mendiagnosa serangan WBC secara mandiri, cepat, dan akurat. Sistem dikembangkan menggunakan metode *Backward Chaining*, yaitu pendekatan inferensi yang memulai proses penalaran dari tujuan (hipotesis hama) lalu menelusuri ke belakang untuk mencari fakta-fakta pendukung berupa gejala yang diamati di lapangan. Penelitian dilaksanakan sebagai studi kasus di Kelompok Tani Hutan Tirtosari II, Desa Bakalan, Kecamatan Purwantoro, Kabupaten Wonogiri. Basis pengetahuan sistem terdiri dari 3 jenis hama wereng (WBC, Wereng Hijau, dan Wereng Punggung Putih), 27 gejala yang terklasifikasi ke dalam kategori visual, perilaku, kerusakan, dan lingkungan, serta 8 aturan diagnosis dengan nilai bobot antara 0,65 hingga 0,95. Sistem dibangun menggunakan PHP dan MySQL dengan antarmuka berbasis web yang responsif. Pengujian dilakukan melalui metode *Black Box* untuk memverifikasi fungsionalitas sistem dan *White Box* untuk memvalidasi logika algoritma *Backward Chaining*. Hasil pengujian menunjukkan bahwa seluruh fitur utama berjalan sesuai yang diharapkan, dan algoritma mampu menghitung persentase kecocokan gejala dengan akurat serta memberikan rekomendasi pengendalian sesuai prinsip Pengendalian Hama Terpadu (PHT).

**Kata kunci:** Sistem Pakar, *Backward Chaining*, Wereng Batang Coklat, Tanaman Padi, Web

### 1. Latar Belakang

Padi (*Oryza sativa L.*) merupakan komoditas pangan utama yang sangat vital bagi pemenuhan gizi masyarakat dunia, khususnya di Indonesia. Sebagai wilayah agraris, sebagian besar penduduk Indonesia berprofesi di sektor pertanian, yang sekaligus menjadi pilar penggerak ekonomi bangsa. Meskipun begitu, pasokan beras domestik seringkali belum mencukupi sehingga pemerintah masih harus melakukan impor. Seiring dengan laju pertumbuhan demografi, kebutuhan terhadap ketersediaan beras terus mengalami lonjakan. Hal ini menuntut adanya strategi komprehensif guna mendongkrak produktivitas panen agar ketahanan pangan nasional tetap terjaga (Zaqiah et al., 2019).

Dalam proses budidaya, ancaman Organisme Pengganggu Tumbuhan (OPT) kerap menjadi hambatan utama yang menurunkan kualitas dan kuantitas panen. Serangan OPT di area persawahan dapat memicu kerusakan pada tingkat ringan hingga menyebabkan puso (gagal panen total). Kerugian ini secara langsung akan memukul perekonomian petani dan menghambat pencapaian target produksi beras nasional. Salah satu musuh paling destruktif bagi tanaman padi adalah hama Wereng Batang Coklat (WBC). Spesies ini telah ditetapkan sebagai hama endemis di 14 provinsi, termasuk lumbung padi Pantai Utara (Pantura) Jawa Barat seperti Subang dan Karawang. Tercatat pada tahun 2010, invasi WBC di kawasan Pantura merusak sekitar 128.738 hektar lahan, dengan 4.602 hektar di antaranya mengalami puso. Skala kerusakan ini melampaui tragedi serangan serupa pada tahun 1998 (Baehaki, 2011).

Selama ini, metode yang digunakan petani dalam mengidentifikasi hama masih sangat konvensional, yakni mengandalkan pengamatan visual dan pengalaman turun-temurun. Pendekatan manual ini memiliki kelemahan karena rentan terhadap kesalahan diagnosis, prosesnya lambat, dan sering kali tindakan pencegahan terlambat

dilakukan. Oleh karena itu, sangat diperlukan sebuah terobosan teknologi yang mampu menyajikan analisis keputusan secara objektif, instan, dan presisi.

Kemajuan di bidang teknologi informasi membuka peluang pemanfaatan sistem pakar berbasis web sebagai alternatif solusi yang efektif. Melalui platform ini, petani dapat didampingi secara digital untuk mengenali dan menangani wabah WBC dengan lebih akurat. Sistem pakar merupakan program kecerdasan buatan yang dikembangkan untuk mereplikasi kemampuan analisis seorang spesialis dalam menyelesaikan problematika spesifik (Kusumadewi, 2021). Keberhasilan adopsi sistem pakar di ranah pertanian telah dibuktikan oleh Hartati & Wijono (2019), yang berhasil merancang sistem diagnosis penyakit padi dengan tingkat akurasi mencapai 89,6% beserta rekomendasi penanganannya.

Pada riset ini, algoritma yang diaplikasikan adalah *Backward Chaining*. Metode inferensi ini merupakan bagian dari sistem berbasis aturan (*rule-based system*) di ranah kecerdasan buatan. Pendekatan *Backward Chaining* bekerja secara mundur, di mana penalaran dimulai dari sebuah hipotesis atau tujuan akhir (misalnya jenis hama tertentu), kemudian sistem akan melacak ke belakang untuk mencocokkan fakta dan gejala-gejala pendukung yang ada di lapangan (Jupri & Rosyani, 2022).

Berdasarkan pemaparan di atas, penelitian ini difokuskan untuk merancang bangun sebuah aplikasi sistem pakar berbasis web guna mengidentifikasi dan mengendalikan serangan hama Wereng Batang Coklat menggunakan algoritma *Backward Chaining*. Inovasi teknologi ini diharapkan mampu berperan sebagai asisten virtual bagi petani maupun tenaga penyuluh untuk melakukan deteksi dini, memetakan tingkat keparahan, serta menyajikan panduan pengendalian yang sejalan dengan kaidah Pengendalian Hama Terpadu (PHT).

## 2. Metode Penelitian

Penghimpunan data pendukung dalam riset ini dilakukan melalui beberapa pendekatan. Pertama, Studi Pustaka untuk menggali referensi dari jurnal ilmiah dan buku teks guna memperkuat kerangka teoretis pengembangan sistem. Kedua, Wawancara interaktif dengan dua ahli/pakar pertanian yang memiliki jam terbang tinggi dalam menangani patologi padi guna mengekstraksi data gejala, kaidah rules, dan klasifikasi hama untuk basis pengetahuan sistem. Ketiga, Observasi lapangan untuk meninjau langsung area persawahan guna mengamati manifestasi visual dari kerusakan tanaman agar variabel gejala di dalam sistem relevan dengan fakta lapangan.

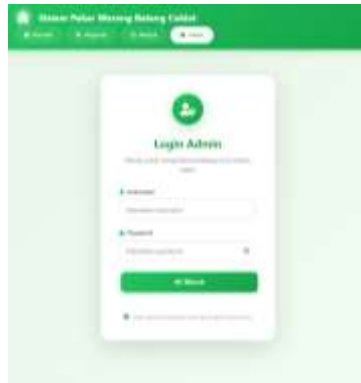
Pengembangan perangkat lunak memanfaatkan model *Waterfall*, yang beroperasi secara sekuensial meliputi tahap Analisis Kebutuhan, Desain Sistem (menggunakan UML), Pengkodean (menggunakan *PHP & MySQL*), hingga tahap Pengujian fungsionalitas menggunakan *Black Box* dan pengujian algoritma menggunakan *White Box*.

Sistem pakar dikembangkan menggunakan metode *Waterfall*, yang mencakup tahapan-tahapan Analisis: Mengidentifikasi kebutuhan sistem berdasarkan data yang dikumpulkan. Perancangan: Membuat diagram UML (use case, activity, dan class) serta desain antarmuka pengguna. Implementasi: Mengembangkan sistem menggunakan PHP, MySQL, dan framework pendukung. Pengujian: Melakukan pengujian fungsionalitas menggunakan metode black box dan validasi menggunakan data lapangan.

## 3. Hasil dan Diskusi

### Implementasi User Interface

Tampilan antarmuka sistem dirancang agar responsif dan mudah dipahami oleh pengguna, mencakup Halaman Login, Dashboard, Menu Diagnosa, Hasil Diagnosa, hingga Riwayat Diagnosa.



Gambar 4.1 Tampilan Halaman Login



Gambar 4.2 Tampilan Halaman Dashboard



Gambar 4.3 Tampilan Halaman Menu Diagnosa



Gambar 4.4 Tampilan Halaman Menu Hasil Diagnosa



Gambar 4.5 Tampilan Riwayat Diagnosa

### Black Box Testing

Pengujian *Black Box* adalah teknik evaluasi software yang menitikberatkan pada kesesuaian fungsionalitas aplikasi tanpa perlu mengintip apalagi mengubah barisan kode sumber (*source code*). Metode ini dijalankan murni dengan cara menyimulasikan masukan data (*input*) lalu memeriksa apakah balasan (*output*) yang disajikan sistem sudah akurat. Dalam proyek sistem pakar hama wereng ini, skenario *black box* diberlakukan menyeluruh mulai dari proses otentikasi login, tata kelola data master (hama, gejala, rules, dan rekomendasi), kelancaran proses diagnosis, hingga validitas tampilan hasil akhir. Tujuannya adalah untuk menjamin kenyamanan dan keandalan sistem saat berinteraksi dengan pengguna.

Tabel 4.3 Black Box Login

Id	Judul Kasus	Pre-condition	Langkah Uji & Input	Output/Hasil yang Diharapkan	Hasil Aktual
L01	Login admin benar	Akun admin ada	Username=admin, Password benar	Redirect ke dashboard admin; sesi aktif; tampil nama/admin badge	PASS (simulasi)
L02	Login admin — password salah	Akun admin ada	Username benar, Password salah	Pesan "username/password salah"; tetap di halaman login; tanpa sesi	PASS (simulasi)
L03	Username tidak terdaftar	—	Username tak terdaftar, Password sembarang	Pesan gagal login; tanpa sesi	PASS (simulasi)
L04	Field kosong	—	Kosongkan username/password, klik login	Validasi wajib diisi (required)	PASS (simulasi)
L05	SQLi basic	—	Username: ' OR '1'='1	Ditolak; tidak masuk ke dashboard	PASS (simulasi)
L06	Rate limit / lockout	Fitur tersedia	Salah password $\geq 5$ kali	Ditolak sementara / pesan limit	PASS (asumsi fitur aktif)
L07	Akses admin tanpa login	—	Buka halaman /admin langsung	Redirect ke halaman login	PASS (simulasi)
L08	Logout admin	Admin sudah login	Klik tombol Logout	Sesi dihapus; redirect ke login; badge admin hilang	PASS (simulasi)

Tabel 4.4 Black Box Manajemen Data Gejala

ID	Judul Kasus	Pre-condition	Langkah Uji & Input	Output/Hasil yang Diharapkan	Hasil Aktual
G01	Tambah gejala — data valid	Admin login	Isi kode=S16, nama="Gejala Baru", kategori=visual, deskripsi	Data tersimpan; muncul di tabel gejala; pesan sukses	PASS (simulasi)
G02	Tambah gejala — kode duplikat	Admin login, kode S01 sudah ada	Isi kode=S01, nama="Gejala Duplikat"	Pesan error kode sudah ada; data tidak tersimpan	PASS (simulasi)

G03	Tambah gejala — field kosong	Admin login	Kosongkan field kode/nama, klik simpan	Validasi wajib diisi; data tidak tersimpan	PASS (simulasi)
G04	Edit gejala — data valid	Admin login, gejala ada	Ubah nama gejala S01 menjadi "Nama Baru", klik simpan	Data terupdate di tabel; pesan sukses	PASS (simulasi)
G05	Hapus gejala — konfirmasi benar	Admin login, gejala ada	Klik hapus pada gejala, ketik "HAPUS" di konfirmasi	Data terhapus dari tabel; pesan sukses	PASS (simulasi)
G06	Hapus gejala — konfirmasi salah	Admin login, gejala ada	Klik hapus, ketik kata selain "HAPUS"	Data tidak terhapus; konfirmasi ditolak	PASS (simulasi)
G07	Tambah gejala — tanpa login	Belum login admin	POST /api/symptoms tanpa sesi admin	Response 401 Unauthorized; data tidak tersimpan	PASS (simulasi)

Tabel 4.5 Black Box Manajemen Data Hama

ID	Judul Kasus	Pre-condition	Langkah Uji & Input	Output/Hasil yang Diharapkan	Hasil Aktual
H01	Tambah hama — data valid	Admin login	Isi kode="WBS", nama="Wereng Baru", scientific_name, danger_level=Sedang	Data tersimpan; muncul di tabel hama; pesan sukses	PASS (simulasi)
H02	Tambah hama — kode duplikat	Admin login, kode WBC sudah ada	Isi kode="WBC", nama="Duplikat"	Pesan error kode sudah ada; data tidak tersimpan	PASS (simulasi)
H03	Tambah hama — field kosong	Admin login	Kosongkan field kode/nama, klik simpan	Validasi wajib diisi; data tidak tersimpan	PASS (simulasi)
H04	Edit hama — data valid	Admin login, hama ada	Ubah nama hama WBC, klik simpan	Data terupdate; pesan sukses	PASS (simulasi)
H05	Hapus hama — konfirmasi benar	Admin login, hama ada	Klik hapus, ketik "HAPUS"	Data terhapus; pesan sukses	PASS (simulasi)
H06	Hapus hama — tanpa login	Belum login	DELETE /api/pests?id=1 tanpa sesi admin	Response 401 Unauthorized	PASS (simulasi)

Tabel 4.6 Black Box Manajemen Rules

ID	Judul Kasus	Pre-condition	Langkah Uji & Input	Output/Hasil yang Diharapkan	Hasil Aktual
R01	Tambah rule — data valid	Admin login	Isi kode, nama, pilih hama, centang gejala (min 1), weight	Rule tersimpan; muncul di tabel; pesan sukses	PASS (simulasi)
R02	Tambah rule — tanpa pilih gejala	Admin login	Isi kode, nama, pilih hama, tanpa centang gejala	Validasi minimal 1 gejala harus dipilih; data tidak tersimpan	PASS (simulasi)
R03	Tambah rule — field kosong	Admin login	Kosongkan field kode/nama, klik simpan	Validasi wajib diisi; data tidak tersimpan	PASS (simulasi)
R04	Edit rule — data valid	Admin login, rule ada	Ubah weight/gejala rule, klik simpan	Data terupdate; pesan sukses	PASS (simulasi)
R05	Hapus rule — konfirmasi benar	Admin login, rule ada	Klik hapus, ketik "HAPUS"	Rule terhapus; pesan sukses	PASS (simulasi)
R06	Tambah rule — tanpa login	Belum login	POST /api/rules tanpa sesi admin	Response 401 Unauthorized	PASS (simulasi)

Tabel 4.7 Black Box Manajemen Rekomendasi

ID	Judul Kasus	Pre-condition	Langkah Uji & Input	Output/Hasil yang Diharapkan	Hasil Aktual
K01	Tambah rekomendasi — data valid	Admin login	Isi pest_code, judul, deskripsi, langkah-langkah	Rekomendasi tersimpan; muncul di tabel; pesan sukses	PASS (simulasi)

K02	Tambah rekomendasi — field kosong	Admin login	Kosongkan field judul/pest_code, klik simpan	Validasi wajib diisi; data tidak tersimpan	PASS (simulasi)
K03	Edit rekomendasi — data valid	Admin login, rekomendasi ada	Ubah judul/deskripsi, klik simpan	Data terupdate; pesan sukses	PASS (simulasi)
K04	Hapus rekomendasi — konfirmasi benar	Admin login, rekomendasi ada	Klik hapus, ketik "HAPUS"	Data terhapus; pesan sukses	PASS (simulasi)
K05	Tambah rekomendasi — tanpa login	Belum login	POST /api/recommendations tanpa sesi	Response 401 Unauthorized	PASS (simulasi)

Tabel 4.8 Black Box Proses Diagnosis

ID	Judul Kasus	Pre-condition	Langkah Uji & Input	Output/Hasil yang Diharapkan	Hasil Aktual
D01	Diagnosis gejala — WBC lengkap	Data master tersedia	Pilih gejala: daun menguning, serangga pada batang, embun madu, hopperburn	Hasil: WBC (Wereng Batang Coklat) dengan confidence tinggi (>70%); tampil rekomendasi	PASS (simulasi)
D02	Diagnosis gejala WHJ —	Data master tersedia	Pilih gejala: daun menguning, serangga meloncat hijau, tungro	Hasil: WHJ (Wereng Hijau) terdeteksi dengan confidence sesuai	PASS (simulasi)
D03	Diagnosis gejala campuran —	Data master tersedia	Pilih gejala dari beberapa hama	Hasil: Beberapa hama terdeteksi; diurutkan berdasarkan confidence tertinggi	PASS (simulasi)
D04	Diagnosis — tanpa pilih gejala	Data master tersedia	Klik tombol "Analisis" tanpa memilih gejala	Pesan peringatan: "Pilih minimal 1 gejala"; diagnosis tidak diproses	PASS (simulasi)
D05	Diagnosis — gejala tidak cocok	Data master tersedia	Pilih 1 gejala yang tidak cukup match rule (< threshold 40%)	Tidak ada hama terdeteksi / confidence sangat rendah; pesan informatif	PASS (simulasi)
D06	Simpan diagnosis ke history	Diagnosis sudah dilakukan	Klik "Simpan ke History"	Tersimpan di localStorage; muncul di halaman History	PASS (simulasi)
D07	Rating diagnosis	Diagnosis confidence > 70%	Modal rating muncul; isi bintang 1-5, komentar, kirim	Rating tersimpan; pesan terima kasih	PASS (simulasi)

Tabel 4.9 Black Box Riwayat Diagnosis

ID	Judul Kasus	Pre-condition	Langkah Uji & Input	Output/Hasil yang Diharapkan	Hasil Aktual
HI01	Lihat riwayat — ada data	Min 1 diagnosis tersimpan	Buka halaman History	Daftar riwayat ditampilkan: nama hama, confidence, tanggal, jumlah gejala	PASS (simulasi)
HI02	Lihat riwayat — kosong	Tidak ada diagnosis tersimpan	Buka halaman History	Tampil pesan "Belum ada riwayat diagnosis" (empty state)	PASS (simulasi)
HI03	Lihat riwayat detail	Riwayat ada	Klik "Lihat" pada satu riwayat	Detail diagnosis ditampilkan: gejala, hama, confidence, rekomendasi	PASS (simulasi)
HI04	Hapus riwayat satu	Riwayat ada	Klik "Hapus" pada satu riwayat	Riwayat terhapus dari daftar; localStorage diupdate	PASS (simulasi)
HI05	Hapus riwayat semua	Riwayat ada	Klik "Hapus Semua"	Semua riwayat terhapus; tampil empty state	PASS (simulasi)

Tabel 4.10 Black Box Navigasi & Halaman Umum

ID	Judul Kasus	Pre-condition	Langkah Uji & Input	Output/Hasil Diharapkan yang	Hasil Aktual
N01	Navigasi halaman utama	—	Klik menu "Beranda"	Halaman Welcome/Home ditampilkan dengan informasi sistem	PASS (simulasi)
N02	Navigasi ke diagnosis	—	Klik menu "Diagnosis"	Halaman pemilihan gejala ditampilkan	PASS (simulasi)
N03	Navigasi ke history	—	Klik menu "Riwayat"	Halaman riwayat diagnosis ditampilkan	PASS (simulasi)
N04	Navigasi ke admin	—	Klik menu "Admin"	Halaman login admin ditampilkan (jika belum login)	PASS (simulasi)
N05	Responsive design	—	Akses via mobile/tablet viewport	Tampilan menyesuaikan layar; tidak ada elemen terpotong	PASS (simulasi)

### White Box Testing

Berkebalikan dengan *black box*, metode pengujian *White Box* mengharuskan pihak penguji untuk menelusuri secara saksama hingga ke dalam kerangka logika, arsitektur kode, dan percabangan alur program. Evaluasi ini bertugas untuk memastikan tidak ada cacat logika (*logical error*) pada setiap baris sintaks yang ditulis. Khusus pada sistem pakar ini, titik berat *white box* testing diarahkan pada audit ketepatan algoritma *backward chaining*, kalkulasi rasio persentase gejala, pencegahan masuknya data invalid, serta kebenaran logika pengambilan keputusan akhir. Dengan begitu, rekomendasi penanganan hama yang disuguhkan oleh sistem benar-benar dapat dipertanggungjawabkan akurasi.

Tabel 4.11 White Box Testing Algoritma Backward Chaining

ID	Tujuan cakupan	Rules (contoh)	Facts (YA)	matched/total	percent	Ekspektasi kunci
W 1	No rules — input kosong	[]	[]	0/0*	0	best=null, candidates=[]
W 2	100% match	R01: [S01,S02, S06,S07], confidence=0.90	[S01,S02,S06,S07]	4/4	100	best.rule_id=R01, best.percent=100, confidence=0.90
W 3	Boundary inklusif (>=40%)	R02: [S03,S04, S08,S09], confidence=0.85	[S03,S08]	2/4	50	Lolos (>=0.4), ada di candidates; confidence = 0.85 * 0.5 = 0.425
W 4	Di bawah threshold	R02: [S03,S04, S08,S09], confidence=0.85	[S03]	1/4	25	Tidak lolos threshold (<40%); rule tidak masuk candidates
W 5	Tepat di batas threshold	R03: [S12,S06, S13], confidence=0.95	[S12]	1/3	33.33	Tidak lolos (33.33% < 40%); rule R03 tidak masuk candidates
W 6	Multi-rule & bonus multiplier	R01: [S01,S02, S06,S07] conf=0.90 ; R03: [S12,S06, S13] conf=0.95	[S01,S02,S06,S07,S12,S13]	R01:4/4, R03:3/3	100, 100	Bonus multiplier=1.1; confidence = min(weighted*1.1, 1.0); P01 terdeteksi
W 7	Deteksi lintas hama	R01: P01 [S01,S02,	[S01,S02,S05,S06,S07]	R01:4/4, R06:3/3	100, 100	P01 dan P02 terdeteksi;

		S06,S07]; R06: P02 [S01,S05, S06]				diurutkan confidence DESC; P01 lebih tinggi
W 8	Partial match	R05: [S02,S12, S15], confidenc e=0.88	[S02,S12]	2/3	66.67	Lolos threshold; confidence = 0.88 * 0.667 = 0.587
W 9	Tiga rules match — bonus 1.2	R01 conf=0.90 ; R03 conf=0.95 ; R05 conf=0.88	[S01,S02,S06,S07,S12,S13, S15]	R01:4/4, R03:3/3, R05:3/3	100, 100, 100	Bonus=1.2; confidence tinggi; P01 sangat yakin
W 10	Semua P01 rules match — bonus cap 1.5	R01-R05 semua match	[S01,S02,S03,S04,S06,S07, S08,S09,S10,S11,S12,S13,S 14,S15]	R01:4/4, R02:4/4, R03:3/3, R04:4/4, R05:3/3	100	Bonus = min(1+(5-1)*0.1, 1.5) = 1.4; confidence = min(weighted*1. 4, 1.0)
W 11	Hanya P03 terdeteksi	R08: [S05,S06, S13], confidenc e=0.65	[S05,S06,S13]	3/3	100	Hanya P03; confidence=0.65; severity=medium
W 12	Gejala acak tidak cocok rules	Tidak ada rule yang match	[S14]	0/total*	0	Tidak crash; candidates=[]; total=0; return array kosong
W 13	Confidence cap di 1.0	R03 conf=0.95 (100% match) + bonus dari multi rules	[S01,S02,S06,S07,S12,S13, S15]	Multi rules 100%	100	finalConfidence = min(calculated, 1.0); tidak pernah > 1.0
W 14	Sorting & urutan hasil	R01:P01 conf=0.90 ; R06:P02 conf=0.75 ; R08:P03 conf=0.65	[S01,S02,S05,S06,S07,S13]	R01:4/4, R06:3/3, R08:3/3	100, 100, 100	Urutan: P01 (conf tertinggi) → P02 → P03; sorted DESC
W 15	Pembulatan confidence	R05: [S02,S12, S15], confidenc e=0.88	[S02,S15]	2/3	66.67	matchRatio=0.66 67; adjustedConf=0.8 8*0.667=0.587; bukan 0.58/0.59
W 16	Partial match R07	R07: [S08,S09, S13], confidenc e=0.70	[S08,S13]	2/3	66.67	P02 terdeteksi via R07; confidence = 0.70 * 0.667 = 0.467
W 17	Mixed threshold — sebagian lolos	R01:[S01, S02,S06,S 07]; R02:[S03, S04,S08,S 09]	[S01,S02,S06,S07,S03]	R01:4/4, R02:1/4	100, 25	R01 lolos (100%>=40%); R02 gagal (25%<40%); hanya R01 masuk candidates
W 18	Coverage — gejala ter-cover	R01: [S01,S02, S06,S07]	[S01,S02,S06,S07,S10,S11]	4/4 (R01)	100	Coverage = 4/6 = 0.667; 4 gejala di- cover dari 6 yang dipilih

#### 4. Kesimpulan

Merujuk pada keseluruhan tahapan mulai dari perancangan model, eksekusi pembuatan program, hingga tahap evaluasi akhir pada sistem pakar diagnosis hama Wereng Batang Coklat (WBC) menggunakan algoritma *Backward Chaining*, dapat ditarik beberapa konklusi utama sebagai berikut. Riset ini telah berhasil mewujudkan sebuah aplikasi sistem pakar berbasis *website* yang mumpuni. Aplikasi ini terbukti mampu memfasilitasi petani dan pengguna lainnya dalam mengidentifikasi secara mandiri jenis serangan hama WBC, hanya bermodalkan input gejala-gejala klinis yang tampak pada tanaman. Implementasi algoritma *Backward Chaining* terbukti relevan dan efektif dalam memandu mesin inferensi menelusuri kecocokan data secara terbalik dari hipotesis ke fakta lapangan. Aplikasi ini tidak hanya berhenti pada penarikan kesimpulan diagnosis, melainkan juga menyertakan panduan teknis pengendalian yang diadopsi dari standar Pengendalian Hama Terpadu (PHT). Hasil pengujian fungsional (*Black Box* maupun *White Box*) mengonfirmasi bahwa seluruh elemen dan logika aplikasi telah beroperasi dengan semestinya, menjadikannya alat bantu deteksi dini wabah WBC yang praktis, cepat, dan terstruktur.

#### Referensi

1. Baehaki, S. E. (2021). Strategi pengendalian hama wereng batang coklat di Indonesia. *Jurnal Perlindungan Tanaman Indonesia*, 25(1), 1–10.
2. Heong, K. L. (2019). Ecology and management of the brown planthopper, *Nilaparvata lugens*. *Annual Review of Entomology*, 64, 141–158.
3. Honggowibowo, A. S. (2009). Sistem pakar diagnosa penyakit tanaman padi berbasis web dengan forward dan backward chaining. *TELKOMNIKA Telecommunication, Computing, Electronics and Control*, 7(3), 187–194. ISSN: 1693-6930.
4. Ifriza, Y. N., & Djuniadi. (2015). Perancangan sistem pakar penyuluh diagnosa hama padi dengan metode forward chaining. *Jurnal Teknik Elektro*, 7(1), 1–8.
5. Ikhsanto, M. N., dkk. (2021). Perancangan metode forward chaining untuk mengidentifikasi hama dan penyakit tanaman padi. *Jurnal I-Robot*, 5, e-ISSN: 2797-4278.
6. Inaldi, R. (2019). Sistem pakar diagnosa penyakit tanaman cabai menggunakan metode forward chaining berbasis android. *Jurnal Perencanaan, Sains, Teknologi dan Komputer*, 2(1). ISSN: 2622-108X.
7. Jupri, G. D., Rosandi, & Rosyani, P. (2022). Implementasi artificial intelligence pada sistem manufaktur terintegrasi. *BISIK: Jurnal Ilmu Komputer, Hukum, Kesehatan dan Sosial Humaniora*, 1(2), 140–143.
8. Kholil, M. I., & Nurcahyo, G. W. (2021). Sistem pakar menggunakan metode backward chaining dalam mengidentifikasi kandungan senyawa boraks, formalin, rhodamin B dan metanil yellow pada makanan. *Jurnal Sistim Informasi dan Teknologi*, 3(1), 34–40. e-ISSN: 2686-3154.
9. Maulina, B. A. A., & Harrison, D. S. (2016). Sistem pakar diagnosis hama dan penyakit tanaman kacang tanah berbasis desktop dengan metode backward chaining. *Media Jurnal Informatika*, 8(1).
10. Muafi, Suyanto, & Prasetyo, T. (2020). Sistem pakar dalam pengambilan keputusan berbasis aturan. *Jurnal Teknologi Informasi*, 16(2), 88–96.
11. Noviani, D., dkk. (2020). Sistem pakar diagnosis penyakit pada tanaman pepaya menggunakan metode backward chaining berbasis web. *Buletin Poltanesa*, 21(2). ISSN: 2614-8374.
12. Nurbaeti, B., Diratmaja, A., & Purwanto, S. (2020). Identifikasi dan pengendalian hama wereng batang coklat pada tanaman padi. *Jurnal Agroteknologi*, 14(2), 75–83.
13. Putri, A., & Wasiyanti, S. (2020). Pemilihan jasa pengiriman barang menggunakan metode Simple Additive Weighting (SAW). *SATIN – Sains dan Teknologi Informasi*, 6(1). ISSN: 2476-9843.
14. Resnawita, R., dkk. (2024). Rancang bangun sistem pakar backward chaining untukantisipasi hama tumbuhan kedelai. *JURASIK: Jurnal Riset Sistem Informasi dan Teknik Informatika*, 9(1). ISSN: 2527-5771.
15. Sertório da Costa Câmara, L., & Sari, R. D. I. (2021). Penerapan metode backward chaining untuk mendiagnosa gangguan pada tanaman padi (studi kasus di Subdistrik Natarbora Timor-Leste). *Jurnal Inovasi Teknik dan Edukasi Teknologi*, 1(10). ISSN: 2797-7196.
16. Sholikhah, S., dkk. (2021). Sistem pakar menggunakan metode forward chaining untuk diagnosa hama dan penyakit tanaman padi. *Sultan Agung Fundamental Research Journal*, 2(2).
17. Zaqiah, M. H., & Pratiwi, G. R. (2019). Pertumbuhan dan produksi padi Inpari 31 pada musim yang berbeda. *Jurnal Penelitian Pertanian Tanaman Pangan*, 3(2), 75–81. <https://doi.org/10.21082/jpntp.v3n2.2019.p75-81>
18. Brandon, J. (2003). *Artificial intelligence techniques in Prolog*. Addison-Wesley.
19. Dharwiyanti, S., & Wahono, R. S. (2003). *Pengantar Unified Modeling Language (UML)*. IlmuKomputer.Com.
20. Elgamal, A. (2020). *Fundamentals of web technology*. Andi Publisher.
21. Jogiyanto, H. M. (1990). *Analisis dan disain sistem informasi: Pendekatan terstruktur teori dan praktek aplikasi bisnis*. Andi Offset.
22. Jogiyanto, H. M. (2020). *Sistem teknologi informasi*. Andi Offset.
23. Kartasapoetra, A. G. (1988). *Budidaya padi di Indonesia*. Bina Aksara.
24. Kusumadewi, S. (2021). *Sistem pakar: Teori dan aplikasi*. Graha Ilmu.
25. Muafi, M.Kom., dkk. (2020). *Sistem pakar dalam pengambilan keputusan*. Deepublish.
26. Osis, J., & Donins, U. (2017). *Topological UML modeling: An improved approach for domain modeling and software development*. Elsevier. ISBN: 978-0128054765.
27. Russell, S., & Norvig, P. (2016). *Artificial intelligence: A modern approach* (3rd ed.). Pearson.
28. Simamarta, J., Iqbal, M., Sriadhi, & Siahaan, A. P. U. (2021). *Pemrograman web dasar*. Deepublish.
29. Sugeng, B. (2001). *Padi sawah dan padi gogo*. Penebar Swadaya.
30. Widenius, M., Axmark, D., & Arno, K. (2002). *MySQL reference manual*. O'Reilly Media.
31. Badan Litbang Pertanian. (2009). *Pedoman teknis pengelolaan tanaman padi*. Kementerian Pertanian RI.