

APLIKASI SISTEM PAKAR UNTUK MENDIAGNOSIS PENYAKIT PADA TANAMAN KOPI DENGAN METODE FORWARD CHAINING

Angriani Angkie¹⁾ A. B. Tjandrarini²⁾

1) SI / Jurusan Sistem Informasi, STIKOM Surabaya, email: ang13kie@yahoo.com

2) STIKOM Surabaya, email: asteria@stikom.edu

Abstract: One of the factors that lead to low quality of coffee in Indonesia is because the coffee plants in Indonesia has always been threatened by pests and plant diseases. Lack of information about the types of diseases that attack the coffee plants, causing a lot of plants that are not handled properly. This problem will cause a lot of coffee plants that should be saved, will die or reduce its quality. If so, this problem will affect on productivity levels. the problem is, how a planters can know what kind of diseases that attack that coffee plants and take appropriate action to deal with it without an expert.

Based on that problems, the planters need a tool to help them to get an information about the types of diseases that attack their plants and provide solutions to deal with that diseases. Application of expert systems is one solution to help planters in diagnosing diseases that attack the coffee plant and able to provide solutions to take an appropriate action to deal with that diseases.

To known the results of the diagnosis, this expert system application requires the facts be symptoms of the disease to be traced by using forward chaining methods. From that facts, it will be processed to obtain a conclusion. The conclusion is the name of the disease, the cause of the disease and the solutions to take an appropriate action to deal with that diseases. With the application of this expert system, the information provided can help the planters to take action in addressing the disease of coffee plants, without having to wait or bring an expert in this field.

Keywords: Expert System, Forward Chaining, Coffee Plant Diseases

Perkembangan dunia teknologi informasi saat ini telah mempengaruhi segala aspek kehidupan manusia, bahkan di dalam bidang-bidang di luar disiplin ilmu komputer. Salah satu cabang ilmu komputer yang dapat membantu manusia adalah sistem pakar (*Expert System*). Tujuan praktis dari sistem pakar ini adalah membuat komputer semakin berguna bagi manusia. Sistem pakar dapat membantu manusia dalam membuat keputusan, mencari informasi atau solusi yang lebih akurat. Sistem pakar juga dapat diterapkan di bidang perkebunan.

Kopi (*Coffea spp. L*) merupakan salah satu komoditas perkebunan yang memiliki peran penting dalam menunjang peningkatan ekspor non migas di Indonesia. Menurut Kasubdit Tanaman Teh dan Kopi Direktorat Budidaya Tanaman Rempah dan Penyegar, Direktorat Jenderal Perkebunan Departemen Pertanian, Nyoman Sudarsana, Produktivitas kopi di Indonesia masih rendah, yakni rata-rata sebesar 700 Kg/ha/tahun, atau baru mencapai 60% dari potensi produktivitasnya. Rendahnya tingkat produktivitas dan produksi kopi karena 96% diusahakan oleh perkebunan rakyat. Bila dibandingkan dengan negara produsen utama kopi di dunia lainnya tingkat produktivitas kopinya lebih tinggi, seperti Vietnam (1.540 kg/hektare/tahun), Colombia (1.220 kg/hektare/tahun) dan Brazil (1.000 kg/hektare/tahun). Penyebabnya adalah karena rendahnya kualitas kopi yang umumnya dihasilkan oleh perkebunan rakyat (Merdeka.com, 2006). Salah satu faktor yang menyebabkan rendahnya kualitas kopi di Indonesia adalah karena perkebunan kopi di Indonesia

selalu diancam oleh hama dan penyakit tumbuhan. (Sukamto, 1998).

Kurangnya informasi yang diketahui oleh pihak perkebunan kopi tentang jenis penyakit yang menyerang tanaman kopi, menyebabkan banyak tanaman kopi yang tidak tertangani dengan benar. Hal ini mengakibatkan banyak tanaman kopi yang seharusnya bisa terselamatkan menjadi mati dan kualitas kopi tersebut menurun. Jika hal ini dibiarkan terus-menerus, maka akan berimbas pada tingkat produktifitas tanaman kopi tersebut. Timbul masalah bagaimana supaya pihak perkebunan kopi dapat mengetahui jenis penyakit yang menyerang tanaman kopi dan mengambil tindakan yang tepat untuk menangani tanaman kopi yang terserang penyakit tanpa adanya seorang ahli.

Berdasarkan permasalahan di atas, pihak perkebunan kopi membutuhkan sebuah alat bantu yang dapat memberikan informasi mengenai penyakit yang menyerang tanaman kopi dan memberikan solusi untuk menangani penyakit tersebut. Alat bantu tersebut dapat berupa sebuah aplikasi sistem pakar, dengan tujuan, aplikasi sistem pakar ini dapat membantu pihak perkebunan kopi untuk mengetahui jenis penyakit yang sedang menyerang tanaman kopi berdasarkan pada gejala-gejala penyakit yang terlihat dan juga dengan adanya aplikasi sistem pakar ini, dapat membantu memberikan solusi untuk menangani tanaman kopi yang terserang penyakit, sehingga banyak tanaman kopi yang dapat diselamatkan dari gangguan penyakit.

Kesimpulan yang dihasilkan merupakan hasil penelusuran dari seperangkat data atau fakta yang

berupa gejala penyakit pada tanaman kopi. Untuk itu, metode yang dapat digunakan adalah metode *forward chaining* yang merupakan suatu metode dari *inference engine* untuk memulai penalaran atau pelacakan suatu data dari fakta-fakta yang ada menuju suatu kesimpulan (Dologite, 1993). Dari fakta-fakta yang berupa gejala penyakit pada tanaman kopi tersebut akan diolah untuk mendapatkan sebuah kesimpulan berupa hasil diagnosis penyakit yang meyerang tanaman kopi dan membantu memberikan solusi untuk melakukan tindakan pengendalian yang tepat. Kesimpulan yang dihasilkan inilah yang akan digunakan sebagai pendukung keputusan bagi pihak perkebunan kopi untuk mengambil tindakan dalam menangani tanaman kopi yang terserang penyakit.

METODE

1. Sistem Pakar

Menurut Irawan (2007), sistem pakar adalah sebuah program komputer yang mencoba meniru atau mensimulasikan pengetahuan (*knowledge*) dan ketrampilan (*skill*) dari seorang pakar pada area tertentu. Menurut Kusriani (2006), ada beberapa keuntungan yang dapat diperoleh dengan mengembangkan sistem pakar antara lain:

1. Membuat seorang yang awam dapat bekerja seperti layaknya seorang pakar.
2. Dapat bekerja dengan informasi yang tidak lengkap atau tidak pasti.
3. Meningkatkan output dan produktivitas.
4. Meningkatkan kualitas.
5. Menyediakan nasihat atau solusi yang konsisten dan dapat mengurangi tingkat kesalahan.
6. Membuat peralatan yang kompleks dan mudah dioperasikan karena sistem pakar dapat melatih pekerja yang tidak berpengalaman.
7. Sistem tidak dapat lelah atau bosan.
8. Memungkinkan pemindahan pengetahuan ke lokasi yang jauh serta memperluas jangkauan seorang pakar, dapat diperoleh dan dipakai di mana saja.

Menurut Kusriani (2006), ada beberapa kelemahan yang dapat diperoleh dengan mengembangkan sistem pakar, antara lain:

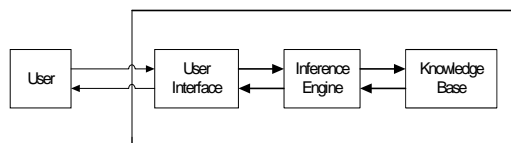
1. Daya kerja dan produktivitas manusia menjadi berkurang karena semuanya dilakukan secara otomatis oleh sistem.
2. Pengembangan perangkat lunak sistem pakar lebih sulit dibandingkan dengan perangkat lunak konvensional.
3. Biaya pembuatannya mahal, karena seorang pakar membutuhkan pembuat aplikasi untuk membuat sistem pakar yang diinginkannya.

Ciri-ciri sistem pakar adalah sebagai berikut (Kusriani, 2006):

1. Terbatas pada bidang yang spesifik.
2. Dapat memberikan penalaran untuk data-data yang tidak lengkap atau tidak pasti.
3. Dapat mengemukakan rangkaian alasan yang diberikannya dengan cara yang dapat dipahami.
4. Berdasarkan pada *rules* atau aturan-aturan tertentu.
5. Dirancang untuk dikembangkan secara bertahap.
6. Outputnya bersifat nasihat atau anjuran.

7. Output tergantung dari dialog dengan user.
8. *Knowledge base* dan *inference engine* terpisah.

Menurut Gonzales (1993) sistem pakar mempunyai 3 bagian utama, yaitu *User Interface*, *Inference Engine* dan *Knowledge Base*. Hubungan ketiga komponen tersebut dapat dinyatakan seperti pada Gambar 1.



Gambar 1. Bagian utama sistem pakar

User Interface memberikan fasilitas komunikasi antara pemakai dan sistem, memberikan berbagai fasilitas informasi dan memberikan berbagai keterangan yang bertujuan untuk membantu mengarahkan alur penelusuran masalah sampai ditemukan solusi (Andi, 2003).

Inference engine adalah bagian dari sistem pakar yang melakukan penalaran dengan menggunakan isi daftar *rule* berdasarkan urutan dan pola tertentu (Andi, 2003). Secara umum ada dua teknik utama yang digunakan dalam *inference engine* untuk menguji *rule*, yaitu penalaran maju (*forward chaining*) dan penalaran mundur (*backward chaining*).

Knowledge base merupakan inti program sistem pakar. Pengetahuan ini merupakan representasi pengetahuan dari seorang pakar. Menurut Irawan (2007), *knowledge base* bisa direpresentasikan dalam berbagai macam bentuk, salah satunya adalah bentuk sistem berbasis aturan (*ruled-based system*). *Knowledge base* tersusun atas fakta yang berupa informasi tentang obyek dan *rules* yang merupakan informasi tentang cara bagaimana membangkitkan fakta baru dari fakta yang telah diketahui.

2. Forward Chaining

Runut maju (*forward chaining*) berarti menggunakan himpunan aturan kondisi-aksi. Dalam metode ini, data digunakan untuk menentukan aturan mana yang akan dijalankan, kemudian aturan tersebut dijalankan. Mungkin proses menambahkan data ke memori kerja. Proses diulang sampai ditemukan suatu hasil (Kusriani, 2006).

3. Verifikasi

Verifikasi merupakan sekumpulan aktifitas yang memastikan suatu sistem apakah telah berlaku dalam kondisi yang ditetapkan. Verifikasi itu sendiri terdiri dari dua proses, pertama memeriksa pelaksanaan sistem, kedua memeriksa konsistensi dan kelengkapan dari basis pengetahuan. Verifikasi dijalankan ketika ada penambahan atau perubahan pada *rule*, karena *rule* tersebut sudah ada pada sistem. Tujuan verifikasi adalah untuk memastikan adanya kecocokan antara sistem dengan apa yang dikerjakan sistem (*rule base*) dan juga untuk memastikan bahwa sistem itu terbebas dari kesalahan. Berikut ini adalah beberapa metode

pemeriksaan *rules* dalam suatu basis pengetahuan (Gonzales, 1993).

1. Redundant Rules
Redundant rules terjadi jika dua *rules* atau lebih mempunyai *premise* dan *conclusion* yang sama.
2. Conflicting Rules
Conflicting rules terjadi jika dua *rules* atau lebih mempunyai *premise* yang sama, tetapi mempunyai *conclusion* yang berlawanan.
3. Subsumed Rules
Subsumed rules terjadi jika *rules* tersebut mempunyai *constraint* yang lebih atau kurang tetapi mempunyai *conclusion* yang sama.
4. Circular Rules
Circular rules adalah suatu keadaan dimana terjadinya proses perulangan dari suatu *rule*. Ini dikarenakan suatu *premise* dari salah satu *rule* merupakan *conclusion* dari *rule* yang lain, atau kebalikannya.
5. Unnecessary if Condition
Unnecessary if Condition terjadi jika dua *rules* atau lebih mempunyai *conclusion* yang sama, tetapi salah satu dari *rule* tersebut mempunyai *premise* yang tidak perlu dikondisikan dalam *rule* karena tidak mempunyai pengaruh apapun.
6. Dead-end Rules
Dead-end rules adalah suatu *rule* yang *conclusion*-nya tidak diperlukan oleh *rule* lainnya.
7. Missing Rules
Missing rules merupakan suatu aturan yang ditandai dengan fakta yang tidak pernah digunakan dalam proses *inference engine*.
8. Unreachable Rules
Unreachable rules merupakan suatu aturan yang gejalanya tidak akan pernah benar.

4. Rule Base System

Langkah awal yang dilakukan dalam menerjemahkan suatu bidang ilmu ke dalam sistem berbasis aturan adalah melalui *block diagram* (diagram blok). *Block diagram* merupakan susunan dari *rules* yang terdapat di dalam sebuah bidang ilmu (Dologite, 1993). Dengan membuat *block diagram* di dalam sistem pakar, maka dapat diketahui urutan kerja sistem dalam mencari keputusan.

Langkah Selanjutnya adalah, membuat *dependency diagram*. Menurut Dologite (1993), *dependency diagram* adalah suatu relasi yang menunjukkan hubungan atau ketergantungan antara inputan jawaban, aturan-aturan (*rules*), nilai dan rekomendasi yang dibuat oleh *prototype* sistem berbasis pengetahuan.

Decision table adalah sebuah tabel yang menyajikan nilai-nilai pada hasil rekomendasi sistem berbasis pengetahuan. *Reduced decision table* adalah pembuatan tabel yang nilai-nilainya didapat dari *decision table* yang sudah direduksi (Dologite, 1993). Setelah didapatkan nilai dari *decision table*, nilai tersebut direduksi untuk mendapatkan nilai dari kondisi terakhir.

Analisis Dan Perancangan Sistem

Kopi merupakan tanaman yang termasuk *Genus Coffea* dari *Family Rubiaceae* ini adalah salah satu dari tiga bahan minuman non-alkoholik (kopi, teh dan coklat). Produksi kopi sebagian besar berasal dari Benua Amerika, yaitu Amerika Selatan dan Amerika Tengah. Sejak tahun 1990-an produksi kopi di Benua Afrika Nampak semakin merosot, sedangkan di Benua Asia semakin meningkat. Jenis-jenis kopi komersial yang sekarang diusahakan di Indonesia, yaitu Robusta dan Arabika. (Yahmadi, 2007).

Adapun penyakit-penyakit yang sering menyerang tanaman kopi di Indonesia adalah (Joko dan Wibisono, 2006 serta Widodo dan Sutiyoso, 2009):

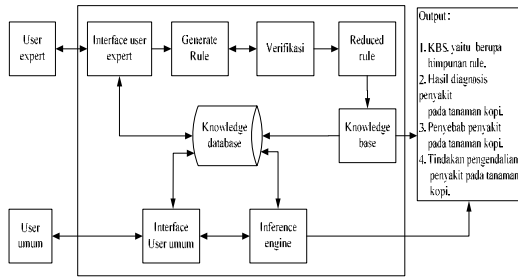
1. Penyakit Cendawan Akar Coklat
2. Penyakit Akar Putih
3. Penyakit Busuk Akar
4. Penyakit Jamur Upas
5. Penyakit Karat Daun
6. Penyakit Bercak daun
7. Penyakit Bercak daun alga

Berdasarkan keterangan dari Kasubdit Tanaman Teh dan Kopi Direktorat Budidaya Tanaman Rempah dan Penyegar, Direktorat Jenderal Perkebunan Departemen Pertanian, Nyoman Sudarsana, tingkat produktivitas kopi di Indonesia masih rendah, yakni rata-rata sebesar 700 Kg/ha/tahun, atau baru mencapai 60% dari potensi produktivitasnya. Rendahnya tingkat produktivitas dan produksi kopi karena 96% diusahakan oleh perkebunan rakyat. Penyebabnya adalah karena rendahnya kualitas kopi yang sebagian besar dihasilkan oleh perkebunan rakyat. Salah satu faktor yang menyebabkan rendahnya kualitas kopi di Indonesia adalah karena perkebunan kopi di Indonesia selalu diancam oleh hama dan penyakit tumbuhan.

Kurangnya informasi yang diketahui oleh pihak perkebunan kopi tentang jenis penyakit yang menyerang tanaman kopi, menyebabkan banyak tanaman kopi yang tidak tertangani dengan benar. Hal ini mengakibatkan banyak tanaman kopi yang seharusnya bisa terselamatkan menjadi mati atau menjadikan kualitas kopi tersebut menurun. Jika hal ini dibiarkan terus-menerus, maka akan berimbas pada tingkat produktivitas tanaman kopi tersebut. Timbul masalah bagaimana supaya pihak perkebunan kopi dapat mengetahui jenis penyakit yang menyerang tanaman kopi mereka dan mengambil tindakan serta melakukan pencegahan yang tepat tanpa adanya seorang ahli.

Berdasarkan permasalahan di atas, pihak perkebunan kopi membutuhkan sebuah alat bantu yang dapat memberikan informasi mengenai penyakit yang menyerang tanaman kopi dan memberikan solusi untuk menangani penyakit tersebut. Alat bantu tersebut dapat berupa sebuah aplikasi sistem pakar, dengan harapan aplikasi sistem pakar ini dapat membantu memberikan kesimpulan berupa hasil diagnosis jenis penyakit yang menyerang tanaman kopi dan membantu memberikan solusi untuk mengambil sebuah tindakan yang tepat dalam menangani tanaman kopi yang terserang penyakit.

1. Desain Arsitektur



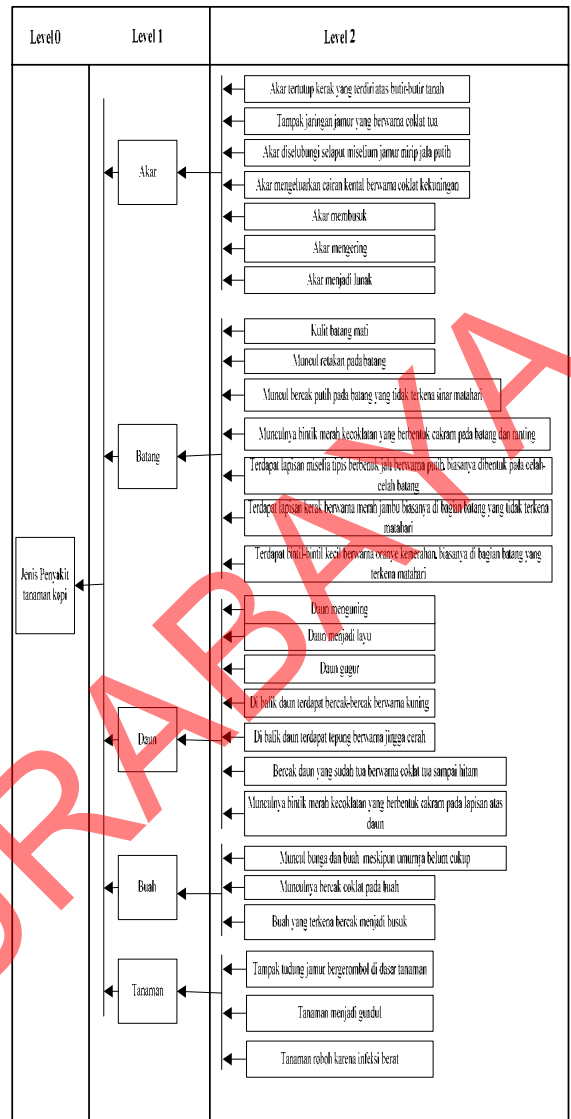
Gambar 2. Desain Arsitektur Aplikasi Sistem Pakar untuk Diagnosis Penyakit pada Tanaman Kopi

Penjelasan dari Gambar 2. adalah sebagai berikut:

1. *Interface user expert*: suatu media yang digunakan oleh *user expert* (pakar) untuk memasukkan parameter *rules*.
2. *Generate rule*: proses ini dilakukan untuk membangkitkan *rules*.
3. *Verifikasi*: yaitu melakukan proses verifikasi pada *rules* yang dibangun dan dimasukan ke dalam sistem dengan tujuan untuk mendapatkan *rules* yang benar sesuai dengan teori verifikasi.
4. *Reduced Rule*: *rules* yang telah diverifikasi.
5. *Knowledge base*: merupakan kumpulan pengetahuan dan *rules* yang diperoleh sistem selama proses berlangsung.
6. *Knowledge database*: digunakan untuk mengembangkan *knowledge base* apabila ada perubahan pada *rules* yang ada.
7. *Interface user umum*: digunakan oleh *user umum* untuk berinteraksi dengan sistem.
8. *Inference engine*: mekanisme inferensi yang digunakan adalah metode runut maju (*forward chaining*).
9. *knowledge base* yang telah dibuat oleh *user expert*.
10. *Output* yang dihasilkan adalah himpunan *rule*, hasil diagnosis penyakit, penyebab penyakit dan tindakan pengendalian penyakit pada tanaman kopi.

2. Perancangan Block Diagram

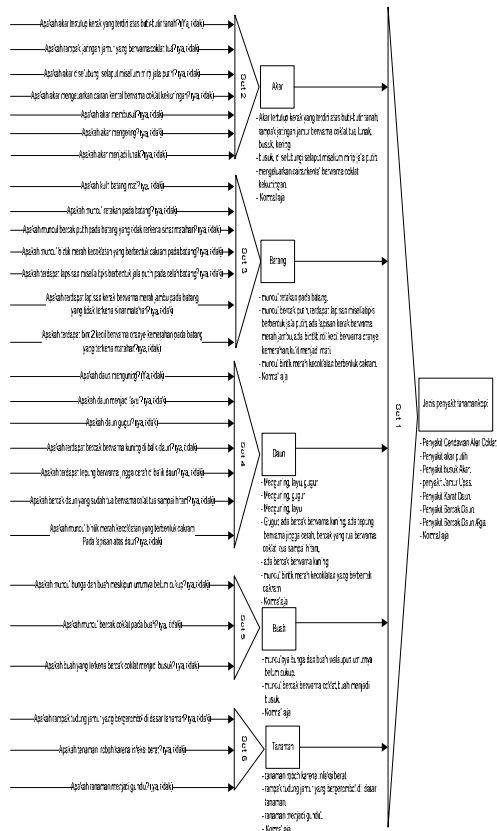
Block diagram yang terlihat pada Gambar 3. Memiliki tiga level, yaitu level 0, level 1, dan level 2. Level 0 berisi hasil akhir berupa jenis penyakit yang menyerang tanaman kopi. Pada level 1 terdapat 5 parameter, yaitu akar, batang, daun, buah, dan tanaman. Terakhir pada level 2 dijelaskan sub parameter dari masing-masing parameter berupa gejala-gejala penyakit yang menyerang tanaman kopi.



Gambar 3. Block Diagram Aplikasi Sistem Pakar Untuk Diagnosis Penyakit pada Tanaman Kopi

3. Perancangan Dependency Diagram

Dependency diagram seperti yang terlihat pada Gambar 4. memiliki enam set. Set 1 yang merupakan konklusi akhir yang merupakan jenis penyakit dari tanaman kopi. Set 2 merupakan set dari parameter akar, set 3 merupakan set dari parameter batang, set 4 merupakan set dari parameter daun, set 5 merupakan set dari parameter buah dan set 6 merupakan set dari parameter tanaman.



Gambar 4. *Dependency Diagram* Aplikasi Sistem Pakar untuk Diagnosis Penyakit pada Tanaman Kopi

4. Perancangan Decision Table

Tabel 1. *Decision Table Rule Set 5, Buah*

No	berbuah meskipun umurnya belum cukup (A)	Munculnya bercak coklat pada buah (B)	Buah yang terkena bercak menjadi busuk (C)	Buah
1	Ya	Ya	Ya	Tidak teridentifikasi
2	Ya	Ya	Tidak	Tidak teridentifikasi
3	Ya	Tidak	Ya	Tidak teridentifikasi
4	Ya	Tidak	Tidak	A
5	Tidak	Ya	Ya	B, C
6	Tidak	Ya	Tidak	Tidak teridentifikasi
7	Tidak	Tidak	Ya	Tidak teridentifikasi
8	Tidak	Tidak	Tidak	Normal

Tabel 1. menunjukkan salah satu perancangan *decision table* aplikasi sistem pakar untuk mendiagnosis penyakit pada tanaman kopi dengan *set 5*, yaitu *set* parameter buah berdasarkan pada perancangan *dependency diagram*. *Decision table* ini memiliki rangkaian aturan akhir yang terkait dengan tiga kondisi. Munculnya bunga dan buah meskipun umurnya belum cukup adalah kondisi pertama yang memiliki dua nilai, yaitu ya atau tidak. Munculnya bercak coklat pada buah adalah kondisi kedua yang memiliki dua nilai, yaitu ya atau tidak. Buah yang terkena bercak menjadi busuk adalah kondisi ketiga yang memiliki dua nilai, yaitu ya atau tidak. Perancangan *reduced decision table* berdasarkan *decision table* pada Tabel 1. menghasilkan parameter seperti yang dapat dilihat pada Tabel 2.

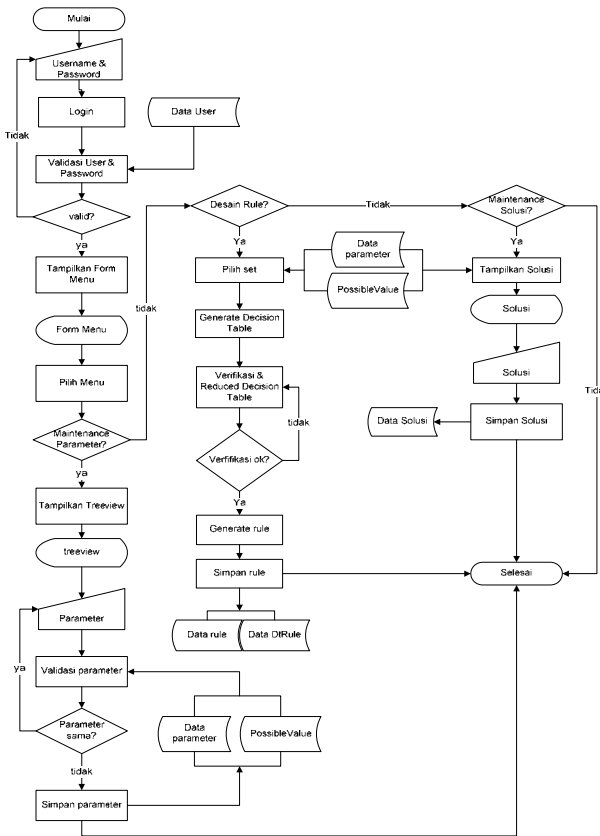
Tabel 2. *Reduced decision table rule set 5*

No	berbuah meskipun umurnya belum cukup (A)	Munculnya bercak coklat pada buah (B)	Buah yang terkena bercak menjadi busuk (C)	Buah
1	Ya	Tidak	Tidak	A
2	Tidak	Ya	Ya	B, C

5. Perancangan Diagram Alir

Diagram alir sistem untuk pakar ini menggambarkan proses-proses yang dilakukan oleh pakar. Proses untuk memasukkan setiap parameter yang sesuai dengan permasalahan, seperti gejala-gejala penyakit, jenis-jenis penyakit dan cara pengendaliannya dan membuat *rule*. Diagram alir sistem untuk pakar dapat dilihat pada Gambar 5.

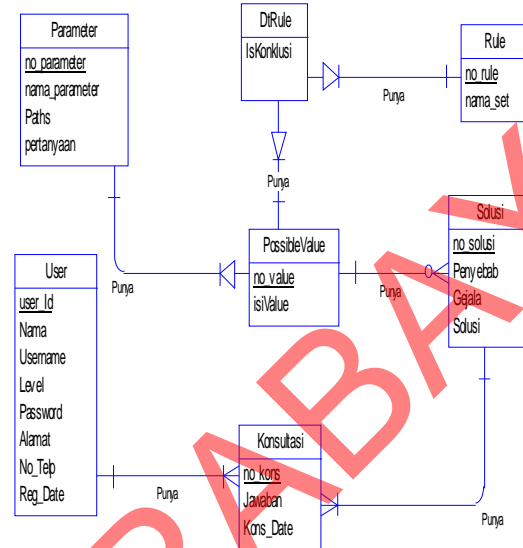
Diagram alir sistem untuk *user* ini menggambarkan jalannya sistem pada desain *user* dalam melakukan konsultasi sampai mendapatkan hasil diagnosis penyakit dan tindakan pengendalian pada tanaman kopi. Diagram alir untuk *user* dapat dilihat pada Gambar 6.



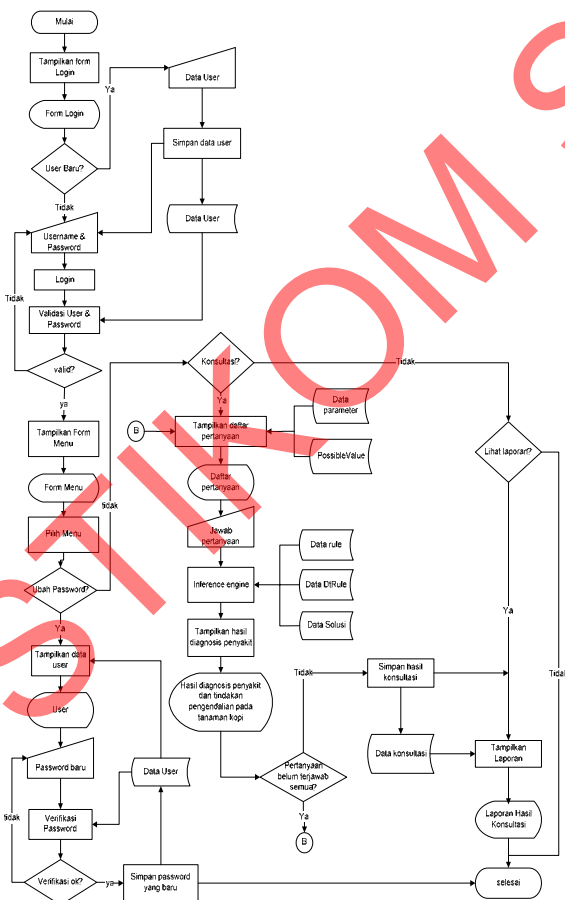
Gambar 5. Diagram Alir Sistem untuk User Pakar

6. Entity Relationship Diagram (ERD)

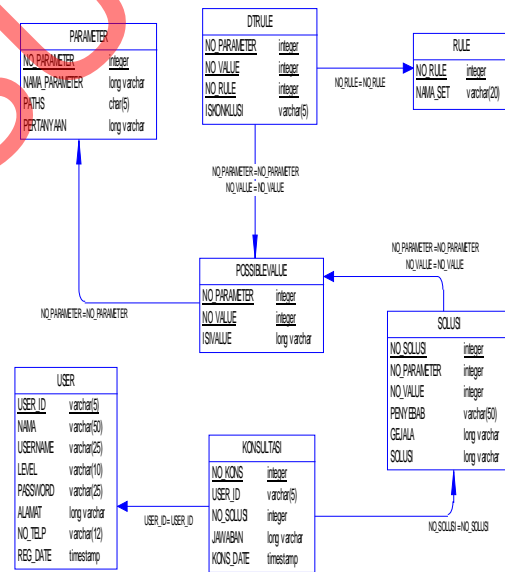
ERD digunakan untuk menggambarkan hubungan tabel-tabel yang digunakan dalam sistem. Dalam laporan ini ERD disajikan dalam bentuk *Conceptual Data Model (CDM)* dan *Physical Data Model (PDM)*. Model CDM dan PDM dari aplikasi ini dapat dilihat pada Gambar 7 dan Gambar 8.



Gambar 7. CDM Aplikasi Sistem Pakar untuk Diagnosis Penyakit Kopi



Gambar 6. Diagram Alir Sistem untuk User Umum



Gambar 8. PDM Aplikasi Sistem Pakar untuk Diagnosis Penyakit Kopi

HASIL DAN PEMBAHASAN

Pada tahap ini diuraikan hasil dan pembahasan aplikasi Sistem pakar untuk mendiagnosis penyakit pada tanaman kopi. Hasil yang dibahas adalah proses untuk mengenerate tabel keputusan dan melakukan reduksi pada tabel tersebut, kemudian hasil reduksi tabel tersebut akan digenerate untuk menghasilkan *rule* yang akan digunakan sebagai fakta

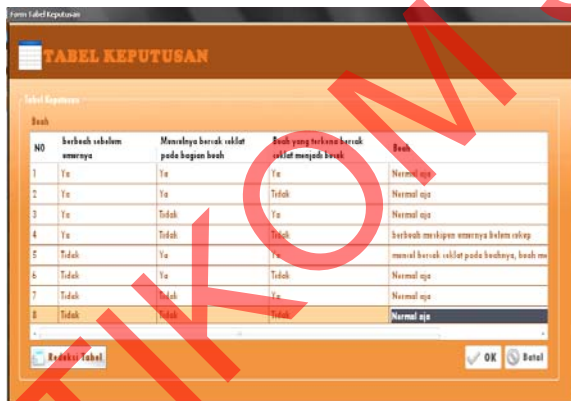
dalam proses untuk mendiagnosis penyakit pada tanaman kopi dengan metode *forward chaning*.

Gambar 9. merupakan tampilan dari form Desain Rule, yang akan digunakan untuk menghasilkan *rule*. Langkah pertama yang dilakukan adalah pilih nama setnya, kemudian menekan tombol *generate decision table* untuk menampilkan *form* Tabel Keputusan, seperti yang terlihat pada Gambar 10. Lakukan proses isi konklusinya dan mereduksi tabel keputusannya.

Hasil reduksi tabel ini dapat dilihat pada Gambar 11. Kemudian menekan tombol OK untuk menampilkan kembali *form* Desain Rule. Langkah selanjutnya adalah menekan tombol *generate rule* untuk menghasilkan *rule* dalam bentuk sistem berbasis aturan yaitu bentuk IF-THEN. Hasil *generate rule* ini dapat dilihat pada Gambar 12. selanjutnya melakukan penyimpanan *rule*.



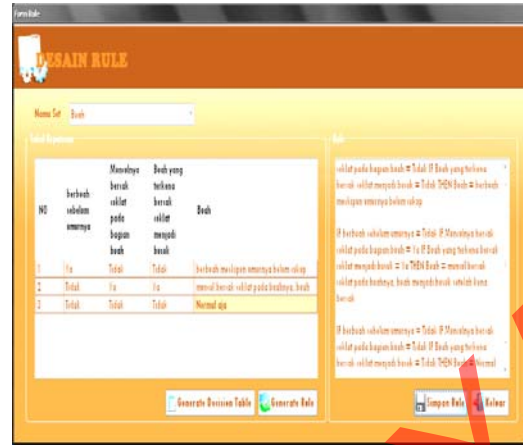
Gambar 9. Form Desain Rule



Gambar 10. Form Tabel Keputusan



Gambar 11. Hasil Reduksi Tabel



Gambar 12. Hasil Generate Rule

Uji coba kasus yang digunakan adalah penyakit Cendawan Akar Coklat. Data uji coba yang digunakan untuk melakukan konsultasi dapat dilihat pada Tabel 3.

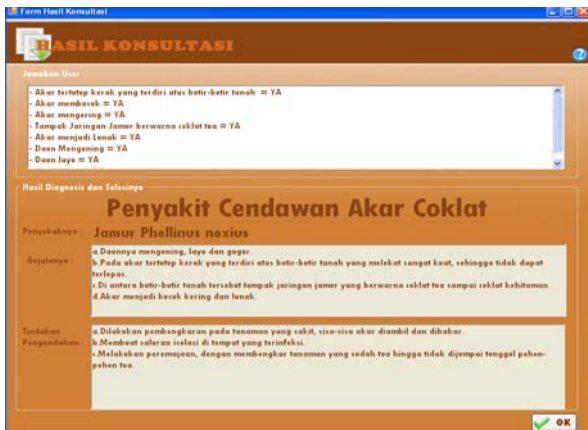
Tabel 3. Data Uji coba Kasus penyakit Cendawan Akar Coklat

Nama Parameter	Jawaban
Akar tertutup kerak yang terdiri atas butir-butir tanah	Ya
Akar membusuk	Ya
Akar mengering	Ya
Tampak Jaringan Jamur berwarna coklat tua pada akar	Ya
Akar menjadi Lunak	Ya
Daun Menguning	Ya
Daun layu	Ya
daun gugur	Ya

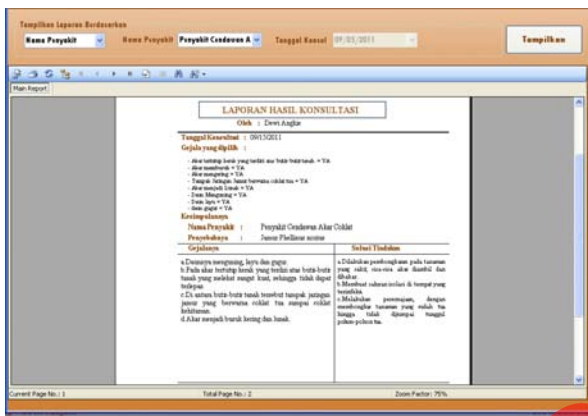
Berdasarkan data uji coba pada Tabel 3. Uji coba dilakukan dengan memilih pertanyaan yang ditampilkan pada *form* Konsultasi sesuai dengan data uji tersebut. Proses konsultasi ini dapat dilihat pada Gambar 13 Setelah melakukan konsultasi pengguna harus menekan tombol diagnosis untuk menghasilkan diagnosis penyakit dengan menggunakan metode *forward chaning*. Hasil diagnosis penyakitnya dapat dilihat pada Gambar 14 dan Gambar 15 adalah laporan hasil konsultasinya.



Gambar 13. Tampilan form konsultasi



Gambar 14. Hasil Diagnosis penyakit



Gambar 15. Form Laporan Hasil Konsultasi

SIMPULAN

Beberapa kesimpulan yang didapatkan dari sistem ini, yaitu:

1. Aplikasi yang telah dibuat dapat digunakan untuk mendiagnosis penyakit pada tanaman kopi dengan metode *forward chaining*.
2. Aplikasi yang telah dibuat dapat memberikan solusi tindakan pengendalian sesuai dengan hasil diagnosis penyakitnya.

RUJUKAN

- Andi, 2003. *Pengembangan Sistem Pakar Menggunakan Visual Basic*, Yogyakarta: Andi Offset.
- Dologite, D. G. 1993. *Developing Knowledge-Based System Using VP-Expert*. New York: Macmillan Publishing Company.
- Gonzalez, A. J dan Dankel D. D. 1993. *The Engineering of Knowledge-based System*. New Jersey: Prentice Hall inc.
- Irawan, Jusak. 2007. *Buku Pegangan Kuliah Sistem Pakar*. Surabaya: Sekolah Tinggi Manajemen Informatika & Teknik Komputer Surabaya.
- Joko, S dan Wibisono, I. I. 2006. *Hama dan Penyakit Tanaman Perkebunan*. Yogyakarta: PT Citra Aji Parama.
- Kusrini. 2006. *Sistem Pakar (Teori dan Aplikasi)*. Yogyakarta: Andi Offset.
- Merdeka.com. 2006. *Produktivitas Kopi Indonesia Baru 60%*. (Online). (<http://www.merdeka.com/ekonomi/nasional/pr-oduktivitas-kopi-indonesia-baru-60-5njwvwm.html>, diakses tanggal 24 Februari 2011).
- Sukanto, S. 1998. *Pengelolaan Penyakit Tanaman kopi. Kumpulan Materi Pelatihan Pengelolaan Organisme Pengganggu Tanaman Kopi*. PUSLIT KOKA.
- Widodo dan Sutyoso Y. Vol 09. *Hama & Penyakit Tanaman. Deteksi Dini & Penanggulangan*. Depok: PT Trubus Swadaya.
- Yahmadi, Mudrig. 2007. *Rangkaian Perkembangan dan Permasalahan Budidaya & Pengolahan Kopi di Indonesia*. Surabaya: Asosiasi Export Kopi Indonesia Jatim.