

IMPLEMENTASI ALGORITMA A*(STAR) PADA SISTEM PAKAR UNTUK MENDIAGNOSA PENYAKIT PADA TANAMAN JERUK (Studi Kasus : Tanaman Jeruk Rimau Gerga Lebong)

Mentari Elnaz¹, Ernawati², Asahar Johar³

^{1,2}Program Studi Teknik Informatika, Fakultas Teknik, Universitas Bengkulu.
Jl. WR. Supratman Kandang Limun Bengkulu 38371A INDONESIA
(tel: 0736-341022; fax: 0736-341022)

¹mentarielnaz70@gmail.com

²w_ier_na@yahoo.com

³asahar.johar@yahoo.com

Abstrak: Jeruk Rimau Gerga Lebong adalah salah satu komoditas utama Provinsi Bengkulu. Hama penyakit yang menyerang tanaman jeruk ini sangat bervariasi sehingga sangat dibutuhkan seorang konsultan pertanian yang mampu mendiagnosa hama penyakit tanaman jeruk. Penelitian ini bertujuan untuk menghasilkan sistem pakar berbasis dekstop yang dapat mendiagnosa hama penyakit tanaman jeruk. Bahasa pemrograman yang digunakan adalah *Borland Delphi 7* dan *XAMPP* untuk mengolah. Metode yang akan digunakan adalah Metode *A-Star*. Metode ini digunakan dalam mesin inferensi yang mengolah basis pengetahuan menjadi kesimpulan-kesimpulan yang diharapkan. *Output* dari sistem ini adalah hasil diagnosis berupa nilai kesimpulan dan informasi penyakit yang dialami oleh hama penyakit tanaman. Hasil analisis dan pengujian menunjukkan bahwa ditinjau dari segi akurasi diagnosis dengan *Best First Search (Rules)* yang mampu menghasilkan diagnosis penyakit yang akurat dengan kesimpulan hipotesis yang sama

Kata kunci : Jeruk Rimau Gerga Lebong, sistem pakar, diagnosa hama, Metode *A-Star*, *Best First Search*

Abstract: Jeruk Rimau Gerga Lebong is one of the main commodity in the Bengkulu Province. Pests that attack citrus crops are highly variable so it is needed an agricultural consultant who can diagnose pests of citrus. This research aims to produce a desktop-based expert system which can diagnose pests of

citrus. This expert system developed based on Sequential Linear method and Data Flow Diagram (DFD). The programming language used is Borland Delphi 7 and XAMPP to process. The method that will be used is the A-Star method. This method is used in the inference engine that processes the knowledge

base into the expected conclusions. The results of analysis and testing shows that in terms of the diagnosis accuracy Best First Search is capable to producing an accurate disease diagnosis with the same hypothesis conclusion

Keywords : Jeruk Rimau Gerga Lebong, expert system, diagnoses of diseases, A-Star Method, Best First Search

I. PENDAHULUAN

Salah satu jenis jeruk yang dikembangkan di Provinsi Bengkulu adalah jeruk Rimau Gerga Lebong. Jeruk tersebut merupakan komoditas unggulan Kabupaten Lebong karena mempunyai keunggulan kompetitif, yaitu buahnya berwarna kuning-oren, berbuah sepanjang tahun, ukuran buah besar 200-350 gram dan kadar sari buah tinggi [1]. Dirjen hortikultura mulai tahun 2011 telah menetapkan jeruk Rimau Gerga Lebong ini sebagai prioritas nasional, untuk dikembangkan dari yang sekarang baru sekitar 6 ha menjadi kawasan agribisnis hortikultura/jeruk dilahan tidur seluas 6000 ha lima tahun mendatang.

Masa berbunga sampai menjadi buah masak sekitar 6-7 bulan tergantung varietas jeruk. Tanaman jeruk dapat berbuah setelah berumur 3 tahun dan buah paling banyak pada tanaman yang berumur lebih dari 5 tahun. Jeruk Rimau Gerga Lebong, sudah berbuah pada umur 2 tahun [2].

Secara umum produksi buah jeruk di tingkat petani masih dilakukan secara tradisional dan belum/tidak menerapkan manajemen produksi mutu sehingga mutu buah rendah seperti kulit buah burik, kotor, tidak mulus, warna buah tidak menarik/pucat, rasa buah beragam, dan sebagainya yang menyebabkan pertumbuhan tanaman menjadi tidak optimal. Banyaknya jenis penyakit yang dapat menyerang tanaman jeruk ini serta sulitnya proses deteksi karena adanya kemiripan gejala yang ditimbulkan membuat para petani jeruk tidak

bisa menentukan langkah pencegahan dan pengendalian yang tepat untuk mengatasi penyakit tersebut. Tiap jenis penyakit memiliki langkah pencegahan dan pengendalian yang berbeda. Maka diharapkan dengan adanya kemajuan teknologi pada saat ini, penyakit yang menyerang tanaman jeruk Rimau Gerga Lebong dapat terdeteksi dengan lebih cepat.

Seiring dengan perkembangan teknologi, maka dikembangkan sebuah teknologi yang mampu mengadopsi proses dan cara berpikir manusia yaitu teknologi *Artificial Intelligence* atau kecerdasan buatan. Sistem pakar merupakan program komputer yang dapat meniru proses pemikiran dan pengetahuan pakar untuk menyelesaikan suatu masalah yang spesifik. Implementasi sistem pakar banyak digunakan untuk kepentingan komersial, karena sistem pakar dipandang sebagai cara penyimpanan pengetahuan pakar dalam bidang tertentu ke dalam suatu program aplikasi, sehingga dapat memberikan keputusan dan melakukan penalaran secara cerdas. Dengan adanya teknologi seperti ini, maka penyajian informasi akan lebih cepat dan mudah.

Metode yang digunakan pada sistem pakar mendiagnosa hama penyakit Jeruk Rimau Gerga Lebong ini adalah metode *A-Star*, dengan menerapkan penelusuran *Best First Search*, dimanapadapenelusuraninimerupakanpenelusurant erbimbinguntuk mendapatkansuatu kesimpulan yang berupa nama hama penyakit, gambar hama penyakit dan solusi berdasarkan gejala-gejala yangdimasukan. Setiap gejala memiliki bobot yang telah ditentukan oleh pakar jeruk sesuai dengan pengalamannya.

Berdasarkan permasalahan diatas, perlu adanya sebuah sistem pakar yang mampu

mendeteksi jenis penyakit dengan gejala awal. Sistem ini berbasis komputer yang dirancang agar dapat mengumpulkan dan menyimpan pengetahuan dari beberapa orang pakar hama penyakit jeruk, khususnya pakar hama penyakit pada tanaman jeruk Rimau Gerga Lebong dalam memecahkan masalah. Aplikasi ini juga akan memberikan informasi mengenai cara pencegahan dan pengendalian yang tepat terhadap penyakit yang menyerang tanaman jeruk Rimau Gerga Lebong.

II. LANDASAN TEORI

A. Sistem Pakar

Sistem pakar adalah salah satu cabang dari AI yang membuat penggunaan secara luas *knowledge knowledge* yang khusus untuk menyelesaikan masalah tingkat manusia yang pakar[3].

Seorang pakar adalah seorang ahli dalam suatu bidang pengetahuan tertentu. Dengan sistem pakar ini, orang awam pun dapat menyelesaikan masalah yang cukup rumit yang sebenarnya hanya dapat diselesaikan dengan bantuan para pakar. Bagi para pakar, sistem ini juga akan membantu aktivitasnya sebagai asisten yang sangat berpengalaman. Adapun ciri-ciri sistem pakar antara lain :

1. Memiliki fasilitas informasi yang handal.
2. Mudah dimodifikasi.
3. Dapat digunakan dalam berbagai jenis komputer.
4. Memiliki kemampuan untuk belajar beradaptasi.

Secara garis besar banyak manfaat yang dapat diambil dengan adanya sistem pakar ahli [4] antara lain :

1. Memungkinkan orang awam bisa mengerjakan pekerjaan para ahli.

2. Bisa melakukan proses secara berulang secara otomatis.
3. Menyimpan pengetahuan dan keahlian dari para pakar.
4. Meningkatkan *output* dan produktivitas.
5. Meningkatkan kualitas.

Disamping memiliki beberapa keuntungan, sistem pakar juga memiliki beberapa kelemahan antara lain :

1. Biaya yang diperlukan dalam pembuatan aplikasi sistem pakar dan memeliharanya sangat mahal.
2. Sistem pakar sulit dikembangkan. Hal ini tentu saja erat kaitannya dengan ketersediaan pakar di bidangnya.
3. Sistem pakar tidak 100% bernilai benar.

B. Konsep Dasar Sistem Pakar

Sistem pakar merupakan sistem informasi berbasis komputer yang menggunakan pengetahuan pakar untuk mencapai performa keputusan tingkat dalam domain persoalan yang sempit [8]. Konsep dasar sistem pakar mencakup beberapa persoalan mendasar, antara lain apa yang dimaksud dengan keahlian, siapa yang disebut pakar, bagaimana keahlian dapat ditransfer dan bagaimana sistem bekerja.

Pada sistem pakar sudah tentu akan membutuhkan seorang pakar. Pakar merupakan orang yang memiliki pengetahuan dan metode khusus, serta kemampuan untuk menerapkan bakat tersebut dalam member nasihat dan memecahkan persoalan. Pakar bertugas menyediakan pengetahuan tentang bagaimana melaksanakan suatu tugas yang akan dijalankan oleh sistem pakar. Pakar mengetahui fakta yang mana yang penting dan memahami arti hubungan diantaranya.

Biasanya, pakar manusia mampu melaksanakan hal-hal seperti berikut :

1. Mengenali dan merumuskan persoalan.
2. Memecahkan persoalan dengan cepat dan tepat.
3. Menjelaskan solusi tersebut.
4. Belajar dari pengalaman.
5. Menyusun ulang pengetahuan.
6. Rendah hati (menyadari keterbatasan seseorang).

C. Struktur Sistem Pakar

Sistem pakar terdiri dari 2 lingkungan, yaitu lingkungan pengembangan dan lingkungan konsultasi [5]. Lingkungan pengembangan digunakan untuk membangun komponen dan masukkan pengetahuan ke dalam basis pengetahuan. Lingkungan konsultasi digunakan oleh seseorang yang bukan ahli untuk memperoleh pengetahuan dan nasihat pakar.

Tiga komponen utama dalam sistem pakar adalah pengetahuan, mesin inferensi dan antarmuka pengguna. Sistem pakar berinteraksi dengan pengguna dapat pula berisi komponen tambahan seperti berikut ini.

1. Subsistem akuisisi pengetahuan
2. *Blackboard* (tempat kerja)
3. Subsistem penjelasan (*justifier*)
4. Sistem perbaikan pengetahuan

D. Metode *Best First Search*

Metode *Best First Search* termasuk dalam kategori pencarian *heuristic*.

Heuristic adalah suatu metode pencarian terbimbing untuk mendapatkan suatu kesimpulan.

Metode *best first search* merupakan kombinasi dari metode *depth first search* dan *breadth first search* dengan mengambil kelebihan dari kedua metode tersebut. Informasi *heuristic*

akan membantu proses dalam proses pencarian berikut ini :

1. Memutuskan simpul mana yang akan diperluas berikutnya.
2. Memperluas simpul, yaitu memutuskan penyukses mana yang akan dihasilkan.
3. Memutuskan simpul mana yang akan dipotong dari ruang masalah.

Tujuan menggabungkan dua teknik *search* ini adalah untuk menelusuri satu jalur saja pada satu saat, tapi dapat berpindah ketika jalur lain terlihat lebih menjanjikan dari jalur yang sedang ditelusuri. Untuk mendapatkan jalur yang menjanjikan adalah dengan memberikan skala prioritas pada setiap *state* saat dihasilkan dengan fungsi *heuristic*. Berikut ini Algoritma *best first search* :

1. Node terdekat dengan keadaan tujuan, sebagaimana ditentukan oleh $h'(n)$ diperluas dahulu.
2. Ciri utamanya:
Cari penyelesaian secara cepat.
Jangan selalu mencari penyelesaian terbaik, karena itu mengevaluasi pilihan terbaik segera, tidak pilihan jangka panjang.
Dapat ke sasaran yang tidak sesuai untuk permulaan yang salah.
3. Perluas sebuah simpul yang sudah pasti simpul akhir. Mirip pencarian *depth first search* (mengikuti sebuah lintasan untuk sebuah solusi atau simpul akhir).

E. Metode A* Search

Metode A* *search* merupakan salah satu contoh algoritma pencarian yang cukup populer di dunia. Algoritma A-Star adalah salah satu algoritma yang menggunakan fungsi biaya. Algoritma A-Star

memeriksa kelayakan biaya yang diperlukan untuk mencapai suatu simpul dari sebuah simpul lain.

A* merupakan algoritma yang pertama kali dikembangkan oleh Nils Nilsson pada tahun 1964 berdasarkan algoritma *Dijkstra*. Saat itu, algoritma ini dinamakan algoritma A1. Pada tahun 1967 Bertram Raphael mengembangkan lebih jauh algoritma ini dan menyebutnya A2, namun tidak dapat membuktikan keunggulannya dibandingkan algoritma sebelumnya. Kemudian pada tahun 1968 Peter E menunjukkan bukti keoptimalan algoritma A2 dibandingkan dengan A1. Kemudian algoritma A2 dinyatakan sebagai algoritma paling optimal untuk kasus tersebut, dan diganti namanya menjadi A*. Secara umum *depth-first search* (DFS) dan *breadth-first search* (BFS) adalah dua kasus spesial dari metode A*. Algoritma *Dijkstra* merupakan kasus yang paling special dari A*, di mana $h'(n) = 0$ untuk semua n . Berdasarkan waktu, algoritma ini lebih baik daripada algoritma *Dijkstra* dengan pencarian *heuristic* [6].

Beberapa terminologi dasar yang terdapat pada algoritma ini adalah *starting point*, simpul (nodes) A, *open list*, *closed list*, harga (*cost*), halangan (*unwalkable*). Berikut ini rumus A* untuk sistem pakar.

Fungsi f' sebagai estimasi fungsi evaluasi terhadap node n :

$$f'(n) = g'(n) + h'(n)$$

$f'(n)$ = Nilai pembobotan dari setiap penyakit

$g'(n)$ = Jumlah pembobotan dari setiap gejala pada suatu penyakit

$h'(n)$ = Jumlah pembobotan dari gejala yang tidak terpilih. Untuk $h'(n) = 0$

F. Model Pengembangan *Waterfall*

Model sekuensial linier merupakan salah satu dari metode yang digunakan untuk pengembangan sistem. Sekuensial linier sering disebut juga dengan “siklus kehidupan klasik” atau “model *waterfall*”. Model sekuensial linier mengusulkan sebuah pendekatan kepada perkembangan perangkat lunak sistematis dan sekuensial yang mulai pada tingkat dan kemajuan sistem pada seluruh analisis, desain, kode, pengujian dan pemeliharaan [1].

G. *Data Flow Diagram* (DFD)

Data Flow Diagram (DFD) atau diagram arus data adalah suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem. Penggunaan DFD sangat membantu dalam memahami sistem secara logika, terstruktur, dan jelas.

Menurut De Marco Dan Yourdan, DFD terdiri dari beberapa komponen. Berikut komponen-komponen yang terdapat dalam DFD [7].

1. Entitas luar. Entitas luar atau terminator merupakan elemen di luar sistem yang akan memberikan masukan (*source*) atau menerima keluaran (*sink*) dari sistem.
2. Proses. Proses menggambarkan apa yang dilakukan oleh sistem. Proses disimbolkan dengan lingkaran yang akan memodifikasi data masukan menjadi keluaran.
3. Arus data (*data flow*). Arus data disimbolkan dengan garis beserta anak panah yang merepresentasikan bagaimana data dialirkan dari entitas luar ke proses, proses ke entitas luar, proses ke proses, atau proses ke penyimpanan data.
4. Penyimpanan data (*data store*). *Data store* berkaitan dengan penyimpanan-penyimpanan,

seperti *file* atau *database* yang berkaitan dengan penyimpanan secara komputerisasi.

H. Bahasa Pemrograman Borland Delphi

Delphi dikembangkan oleh perusahaan perangkat lunak yang bernama Borland dari bahasa-bahasa pemrograman, seperti *Turbo C* dan *Turbo Pascal*. *Delphi* adalah aplikasi database yang memakai bahasa pascal yang berorientasi objek. Pemrograman Berorientasi Objek adalah perluasan dari pemrograman terstruktur yang mengutamakan pemakaian ulang program dan enkapsulasi data berdasarkan fungsinya [8].

Borland Delphi adalah bahasa pemrograman berbasis Microsoft Windows yang didesain untuk dapat memanfaatkan fasilitas Microsoft Windows dengan optimal, khususnya Microsoft Windows 9x, Microsoft Windows 2000, Microsoft ME, Microsoft Windows XP dan Microsoft Windows NET [9]. Adapun Komponen-komponen Delphi diantaranya terdiri dari *project*, *form*, *unit*, *program*, *property*, *event* dan *method*.

III. METODOLOGI

Langkah-langkah yang dilakukan dalam penelitian ini adalah:

1. Studi Pustaka

Studi pustaka dilakukan dengan cara mengumpulkan informasi-informasi yang diperlukan dari berbagai sumber-sumber seperti buku-buku ilmiah, laporan penelitian, karangan-karangan ilmiah, tesis dan disertasi, peraturan-peraturan, ketetapan-ketetapan, buku tahunan, ensiklopedia, dan sumber-sumber tertulis baik cetak maupun elektronik.

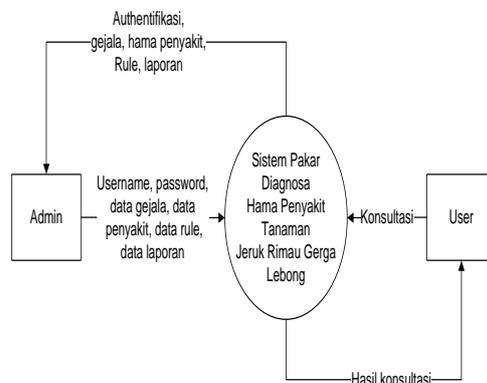
2. Wawancara kepada Kepala Laboratorium BPTPH Bengkulu. Wawancara yang akan

dilakukan diharapkan akan memberikan informasi detail tentang hama penyakit yang menyerang tanaman jeruk Rimau Gerga Lebong serta cara pengendaliannya

IV. ANALISIS DATA DAN PERANCANGAN

Sistem ini dibangun dengan dua pemakai yaitu *user* dan *admin*. *User* dapat melakukan konsultasi tanpa harus melakukan *login*, namun hanya dapat mengakses menu cara penggunaan menu *about* dan halaman yang terdapat pada menu *user* yaitu halaman konsultasi, halaman hasil diagnosa dan halaman hasil perhitungan. Sedangkan *admin* harus melakukan *login* terlebih dahulu untuk masuk ke menu *admin*. Pada menu *admin* terdapat halaman *input* data penyakit, *input* data gejala, *input* data *rule* dan riwayat untuk mencetak hasil dari konsultasi.

Sistem dibangun dengan pendekatan terstruktur. Diagram level 0 atau juga biasa disebut dengan diagram konteks merupakan diagram tertinggi dari DFD. Diagram ini menggambarkan hubungan sistem dengan lingkungan di sekitarnya. Berikut ini adalah gambar diagram level 0 pada Sistem Pakar Untuk Mendiagnosa Penyakit Pada Tanaman Jeruk:



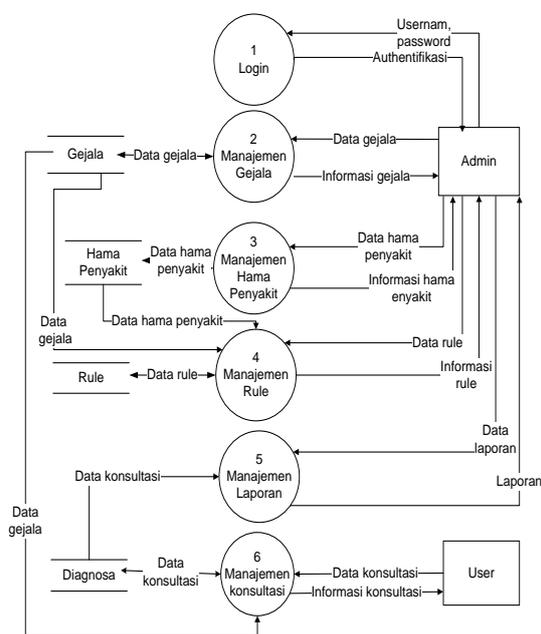
Gambar 4.1 Diagram Level 0

Pada sistem ini terdapat dua entitas yang merupakan pihak atau pengguna yang berinteraksi terhadap sistem yang akan dibangun. Entitas pada system ini yaitu *admin* dan *user*. Masing-masing entitas memiliki wewenang dan hak akses yang berbeda-beda.

Admin merupakan orang yang memiliki wewenang penuh atas sistem. *Admin* berwenang dalam pengolahan database sistem. Untuk dapat mengakses sistem, *admin* harus *login* terlebih dahulu dengan memasukkan *username* dan *password*. Kemudian sistem akan memberikan *authentifikasi* dan *admin* akan masuk ke dalam halaman *admin*. *Admin* dapat mengelola data gejala, data hama penyakit, data rule dan dapat melihat laporan.

User tidak perlu melakukan *login* terlebih dahulu untuk mendapatkan hak aksesnya. *User* merupakan bagian yang berhubungan langsung dengan sistem. *User* dapat melakukan konsultasi, dan sistem akan memberikan informasi hasil konsultasi.

Diagram level 1 merupakan diagram turunan dari diagram konteks atau diagram level 0. Ada enam proses utama dalam sistem ini, yaitu: 1) *Login*, 2) *Manajemen Gejala*, 3) *Manajemen Hama Penyakit*, 4) *Manajemen Rule*, 5) *Manajemen Laporan* dan 6) *Manajemen Konsultasi*. Hal ini seperti terlihat pada Gambar 4.2:



Gambar 4.2 Diagram Level 1

- a. **Proses 1 Login**
Proses *login* hanya dilakukan oleh *admin* untuk mendapatkan hak akses sistem. *Admin* harus memasukkan *username* dan *password*. Kemudian sistem akan mengauthentifkasi proses *login* tersebut.
- b. **Proses 2 Manajemen Gejala**
Proses manajemen gejala dilakukan oleh *admin* untuk mengelola data gejala. Manajemen gejala akan mengelola data gejala. Selain itu pada manajemen ini juga mengelola data bobot dari setiap gejala yang disimpan ke dalam *database* gejala
- c. **Proses 3 Manajemen Hama Penyakit**
Proses manajemen hama penyakit dilakukan oleh *admin* untuk mengelola data hama penyakit. Manajemen hama penyakit akan mengelola data hama penyakit. Selain itu pada manajemen ini juga mengelola data gambar dari setiap hama penyakit, dan solusi dalam mengurangi/memberantas hama penyakit

tanaman yang disimpan ke dalam *database* hama penyakit.

d. Proses 4 Manajemen *Rule*

Proses manajemen *rule* dilakukan oleh *admin* untuk mengelola data *rule*. Manajemen *rule* akan mengelola data yang diterima dari *database* gejala dan *database* hama penyakit kemudian disimpan ke dalam *database rule*.

e. Proses 5 Laporan

Proses laporan dilakukan oleh *admin* untuk mengelola data laporan. Laporan akan mengelola data yang diterima dari *database* diagnosa.

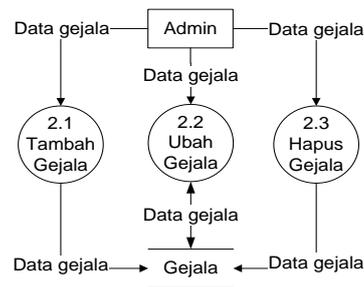
f. Proses 6 Konsultasi

Proses konsultasi dilakukan oleh *user* tanpa harus *login* terlebih dahulu. Pada awal proses konsultasi *user* harus menjawab pertanyaan yang diajukan system mengenai gejala-gejala apa saja yang dialami dan kemudian disimpan kedalam *database user*, kemudian proses konsultasi akan menerima data dari *database* hama penyakit, kemudian disimpan ke dalam *database* diagnosa.

Diagram level 2 merupakan diagram turunan dari diagram level 1. Diagram ini menjelaskan proses dari diagram level 1 yang masih bisa diturunkan. Pada diagram level 1 proses yang masih bisa dijabarkan adalah proses 2, proses 3, proses 4 dan proses 6.

a. Diagram Level 2 Proses 2 Manajemen Gejala

Diagram level 2 proses 2 ini dapat dilihat pada Gambar 4.4 berikut.

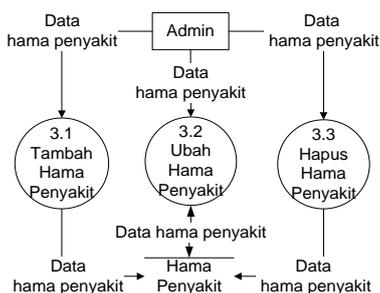


Gambar 4.3 Diagram Level 2 Proses 2 Manajemen Gejala

Berdasarkan gambar 4.3, diagram level 2 proses 2 manajemen klasifikasi memiliki 3 buah proses. Proses pertama yaitu proses tambah gejala, pada proses ini *admin* dapat menambah data gejala hama penyakit jeruk Rimau Gerga Lebong dan disimpan ke dalam *database* gejala. Proses kedua yaitu proses ubah gejala, pada proses ini *admin* dapat mengubah data gejala yang ada di *database*. Proses ketiga yaitu proses hapus gejala, pada proses ini *admin* dapat menghapus data gejala.

b. Diagram Level 2 Proses 3 Manajemen Hama Penyakit

Diagram level 2 proses 3 ini dapat dilihat pada Gambar 4.4 berikut.



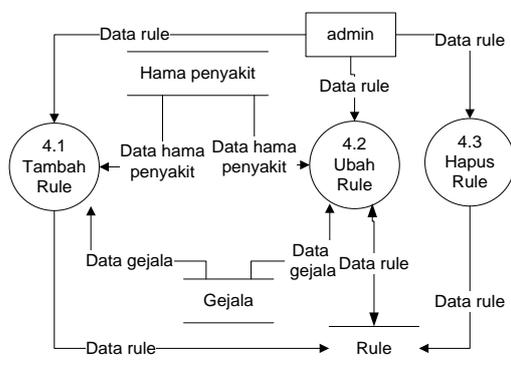
Gambar 4.4 Diagram Level 2 Proses 3 Manajemen Hama Penyakit

Berdasarkan gambar 4.4, diagram level 2 proses 3 manajemen hama penyakit memiliki 3 buah proses. Proses pertama yaitu proses tambah hama penyakit, pada proses ini *admin*

dapat menambah data hama penyakit jeruk Rimau Gerga Lebong dan disimpan ke dalam *database* hama penyakit. Proses kedua yaitu proses ubah hama penyakit, pada proses ini *admin* dapat mengubah data hama penyakit yang ada di *database*. Proses ketiga yaitu proses hapus hama penyakit, pada proses ini *admin* dapat menghapus data hama penyakit.

c. Diagram Level 2 Proses 4 Manajemen *Rule*

Diagram level 2 proses 4 ini dapat dilihat pada Gambar 4.5 berikut.

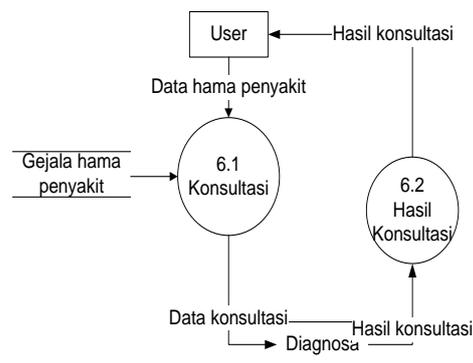


Gambar 4.5 Diagram Level 2 Proses 4 Manajemen *Rule*

Berdasarkan gambar 4.6, diagram level 2 proses 3 manajemen *rule* memiliki 4 buah proses. Proses pertama yaitu proses tambah *rule*, pada proses ini *admin* dapat menambah data *rule* yang mengambil data gejala dari *database* gejala dan data hama penyakit dari *database* hama penyakit kemudian disimpan ke dalam *database* *rule*. Proses kedua yaitu proses ubah *rule*, pada proses ini *admin* dapat mengubah data *rule* yang ada di *database*. Proses ketiga yaitu proses hapus *rule*, pada proses ini *admin* dapat menghapus data *rule*.

d. Diagram Level 2 Proses 6 Konsultasi

Diagram level 2 proses 6 ini dapat dilihat pada Gambar 4.6 berikut



Gambar 4.6 Diagram Level 2 Proses 6 Konsultasi

Berdasarkan gambar 4.6, diagram level 2 proses 6 konsultasi memiliki 2 buah proses. Proses pertama yaitu proses konsultasi, pada proses ini *user* melakukan konsultasi dengan memilih gejala yang dialami tanaman jeruk Rimau Gerga Lebong dari *database* hama penyakit kemudian dilakukan proses perhitungan dan disimpan ke dalam *database* diagnosa. Proses kedua yaitu proses hasil konsultasi, pada proses ini *user* menerima hasil konsultasi dari *database* diagnosa.

V. HASIL DAN PEMBAHASAN

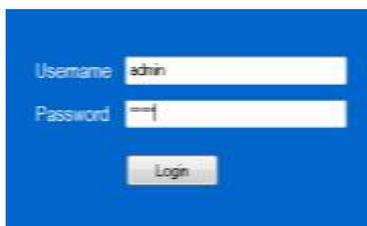
Setelah dilakukan analisis dan perancangan sistem, selanjutnya adalah tahap implementasi. Berikut ini adalah hasil implementasi pada sistem:

1. Hasil Implementasi Halaman Utama



5.1 Halaman Utama

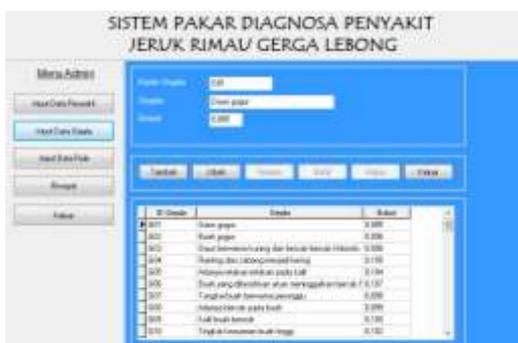
2. Hasil Implementasi Halaman *Login Admin*



5.2 Halaman Login Admin

Pada gambar 5.2, sebelum *admin* masuk ke sistem terlebih dahulu melakukan *login* dengan mengisi *username* dan *password*.

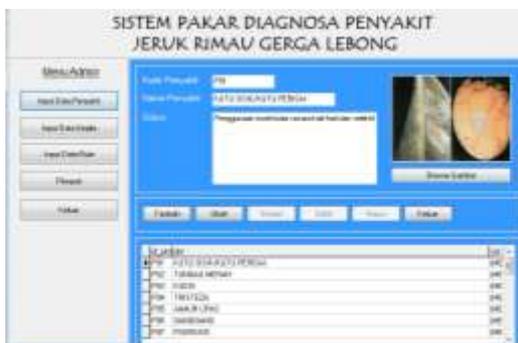
3. Hasil Implementasi Halaman *Input* Data Penyakit



5.3 Halaman *Input* Data Penyakit

Pada gambar 5.3, halaman *input* data penyakit adalah halaman yang digunakan untuk mengelola data hama penyakit tanaman jeruk Rimau Gerga Leborg dengan memasuki data nama penyakit dan solusinya serta gambar setiap penyakit.

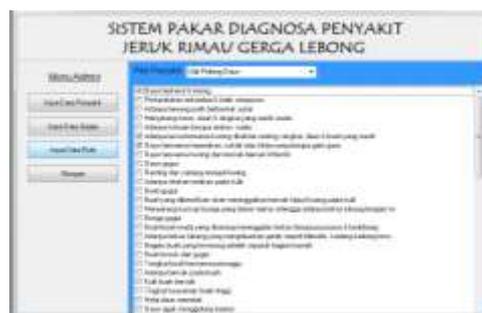
4. Hasil Implementasi Halaman *Input* Data Gejala



5.4 Halaman *Input* Data Gejala

Pada gambar 5.4, halaman *input* data gejala adalah halaman yang digunakan untuk mengelola data gejala. *Admin* dapat menambah, mengubah, dan menghapus data gejala. Selain itu *admin* juga harus mengisi bobot dari setiap gejala.

5. Hasil Implementasi Halaman *Input* Data Rule



5.5 Halaman *Input* Data Penyakit

Pada gambar 5.5, halaman *input* data rule digunakan untuk mengelola data rule antara gejala dan penyakit. *Admin* dapat menambah, mengubah, dan menghapus data rule.

6. Hasil Implementasi Halaman Riwayat



5.6 Halaman Riwayat

Pada gambar 5.6, halaman riwayat digunakan untuk mencetak laporan hasil diagnosa.

7. Hasil Implementasi Halaman Penggunaan



5.7 Halaman Penggunaan

Pada gambar 5.7, halaman penggunaan digunakan untuk memudahkan *user* dalam menjalankan sistem pakar diagnosa hama penyakit Jeruk Rimau Gerga Lebong.

8. Hasil Implementasi Halaman Konsultasi



5.8 Halaman Konsultasi

Pada gambar 5.8, halaman konsultasi digunakan *user* untuk melakukan diagnosa. *User*

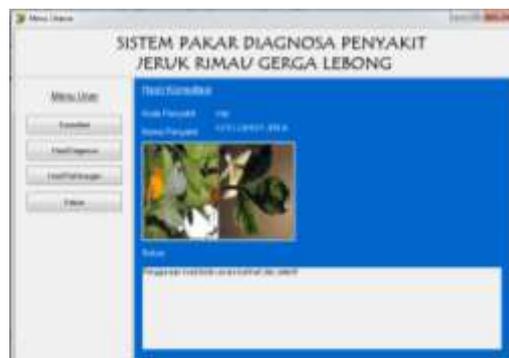
diminta untuk menjawab pertanyaan yang diajukan oleh sistem dengan memilih opsi ya atau tidak sesuai gejala hama penyakit yang dialami oleh tanaman *user*. Sistem akan secara otomatis mengeluarkan pertanyaan yang saling berkaitan antar gejala dan mengarah ke setiap penyakit,

karena dirancang berdasarkan penelusuran *best first search*.

Dimana penelusuran itu termasuk kedalam pencari *an heuristic* (terbimbing)

sehingga hasil *output* nya berupa narikan kesimpulan sesuai dengan data yang telah dimasukkan. Selanjutnya *user* memilih tombol diagnosa

9. Hasil Implementasi Halaman Hasil Konsultasi



5.9 Halaman Hasil Konsultasi

Pada gambar 5.9, halaman ini digunakan untuk melihat hasil konsultasi berupa nama penyakit, gambar penyakit dan solusi untuk mengatasi penyakit tersebut.

10. Hasil Implementasi Halaman Hasil Perhitungan



5.10 Halaman Hasil Perhitungan

Pada gambar 5.10, halaman hasil perhitungan ini digunakan untuk melihat perhitungan diagnosa dengan menerapkan fungsi metode $A^*(Star)$ pada sistem.

11. Hasil Implementasi Halaman *About*



5.11 Halaman *About*

Pada gambar 5.11, halaman *About* digunakan sebagai informasi mengenai sistem pakar diagnosa hama penyakit tanaman jeruk Rimau Gerga Lebong.

VI. KESIMPULAN

Berdasarkan analisa perancangan sistem, implementasi, dan pengujian sistem, maka dapat disimpulkan bahwa sistem ini mampumendiagnosahamapenyakitjerukRimauGer galebongsesuaidengangejalanyamenggunakanmet ode *A-Star* danpenelusuran *Best First Search*

VII SARAN

Berdasarkan analisa perancangan sistem, implementasi, dan pengujian sistem, maka untuk pengembangan penelitian selanjutnya penulis mengharapkan aplikasi ini dapat terus dikembangkan lebih lanjut dalam metode pemeriksaan atau diagnosa hama penyakit tanaman jeruk. Diharapkan untuk dapat menggunakan metode pemeriksaan atau diagnosa penyakit selain metode *A-Star*.

REFERENSI

- [1] Sommerville. (2003). *Rekayasa Perangkat Lunak. Edisi 6, diterjemahkan oleh Hanum Y.* Jakarta: Erlangga.
- [2] Suwanto, A. A. (2009). *Analisis pengembangan pertanian organik di Kabupaten Magelang.* Semarang.
- [3] Armi, M. (2005). *Konsep Dasar Sistem Pakar.* Yogyakarta: CV Andi.
- [4] Kusumadewi. (2003). *Artificial Intelligent (Teknik dan aplikasinya).*
- [5] Turban, E. (2005). *Metodologi Penelitian Survei.* Yogyakarta: LP3ES.
- [6] Riftadi, M. (2007). *Variasi Penggunaan Fungsi Heuristik Dalam Pengaplikasian Algoritma A*.* [Online]. Available: http://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2006-2007/Makalah_2007/MakalahSTMik2007-069.pdf
- [7] Wijayaning, N. (2009). *Modul Praktikum Rekayasa Perangkat Lunak.* Yogyakarta: Universitas Islam Indonesia.
- [8] Martina, I. (2004). *36 Jam Belajar Komputer. Pemrograman Borland Delphi 7.* Elex Media Komputindo.
- [9] Budiman, T. (2008). *Diktat Pemrograman Delphi.* Garut: Lingkungan Amik Garut.