

BAB IV

IMPLEMENTASI DAN EVALUASI

4.1 Kebutuhan Sistem

Implementasi sistem merupakan wujud dari analisis dan perancangan yang telah disusun dan dibuat. Untuk dapat terlaksananya implementasi sistem dengan baik, diperlukan kesiapan dari kebutuhan perangkat keras maupun perangkat lunak.

4.1.1 Kebutuhan perangkat keras

Persyaratan minimal yang dibutuhkan untuk menjalankan rancang bangun aplikasi sistem pakar ini adalah sebagai berikut:

1. Prosesor Intel Dual Core
2. *Memory* 1 GB
3. *Space Harddisk* 2 GB
4. *Monitor, Mouse, keyboard, dan printer*

4.1.2 Kebutuhan perangkat lunak

Perangkat lunak yang dibutuhkan agar mampu menjalankan aplikasi sistem pakar ini adalah:

1. Bahasa pemrograman yang digunakan adalah Visual Basic .NET 2005.
2. SQL Server 2005, untuk pengolahan *database*.
3. .Net Framework 2.0

4.2 Penjelasan Penggunaan Program

Pada bagian ini akan dijelaskan penggunaan beberapa *Form* utama dalam aplikasi sistem pakar ini, antara lain: *Form* Utama, *Form Login*, *form* Daftar User, *Form Menu*, *Form User*, *Form Cari User*, *Form* Ubah Password, *Form Treeview*, *Form* Parameter, *Form* Desain Rule, *Form* Tabel Keputusan, *Form* Solusi, *Form* Isi Solusi, *Form* Konsultasi, *Form* Hasil Konsultasi, *Form* Detil Penyakit, *Form* Laporan User, *Form* Laporan Solusi, *Form* Laporan Rule, *Form* Laporan Hasil Konsultasi dan *Form* Grafik Penyakit.

4.2.1 Form Utama

Form Utama, seperti yang terlihat pada Gambar 4.1 muncul pertama kali saat aplikasi sistem pakar ini dijalankan. Pada *form* ini, terdapat dua tombol yang dapat diakses oleh pengguna, yaitu tombol *User* Pakar untuk menampilkan *form* *Login* untuk *User* Pakar dan tombol *User* Umum untuk menampilkan *form* *Login* untuk *User* umum.



Gambar 4.1 *Form* Utama

4.2.2 Form Login User Pakar

Form Login ini ditampilkan setelah pengguna menekan tombol *User Pakar* pada *form* Utama. *Form Login* digunakan oleh pengguna pakar untuk dapat mengakses *form* selanjutnya dengan mengisi *Username* dan *Password* yang sesuai dengan hak akses yang diberikan. *Form Login* ini dapat dilihat pada Gambar 4.2.



Gambar 4.2 *Form Login* User Pakar

Hak akses yang diberikan kepada pengguna pakar dibedakan menjadi dua level, yaitu level Admin dan Pakar. Level inilah yang membedakan *forms* yang dapat diakses oleh pengguna pakar sesuai dengan hak aksesnya. Tombol *OK* untuk validasi setiap masukan yang ada dan untuk menampilkan *form Menu* sesuai dengan hak akses yang diisi. Tombol *Batal* untuk membatalkan pengisian pada *form Login* dan kembali ke *form* Utama.

4.2.3 Form Login User Umum

Form Login ini digunakan oleh *User* umum untuk masuk ke dalam sistem. *User* yang memiliki *ID* bisa langsung mengisi *form* ini, jika tidak maka *User* harus mendaftar terlebih dahulu dengan memilih *linklabel* klik disini untuk buat *user* baru. *form Login* ini dapat dilihat pada Gambar 4.3.

Gambar 4.3 *Form Login User Umum*

Pada *form Login* seperti yang terlihat pada Gambar 4.3, hak akses atau level yang diberikan kepada pengguna adalah level *User*. Level inilah yang membedakan *forms* yang dapat diakses oleh pengguna umum dengan *form* yang diakses oleh pengguna pakar. Tombol *OK* untuk validasi setiap masukan yang ada dan untuk menampilkan *form Menu*. Tombol *Batal* untuk membatalkan pengisian pada *form Login* dan kembali ke *form Utama*.

4.2.4 **Form Daftar User**

Form Daftar User ini ditampilkan saat pengguna memilih *linklabel* untuk membuat *user* baru. *Form* ini digunakan untuk membuat data *user* umum yang baru pertama kali mengakses aplikasi sistem pakar ini. Desain *form* ini dapat dilihat pada Gambar 4.4. Tombol *OK* yang terlihat pada Gambar 4.4 digunakan untuk menyimpan hasil masukan pengguna, kemudian menampilkan *form Login* untuk *user* umum. Tombol *Batal* untuk membatalkan proses daftar *user* dan kembali ke *form Login*.

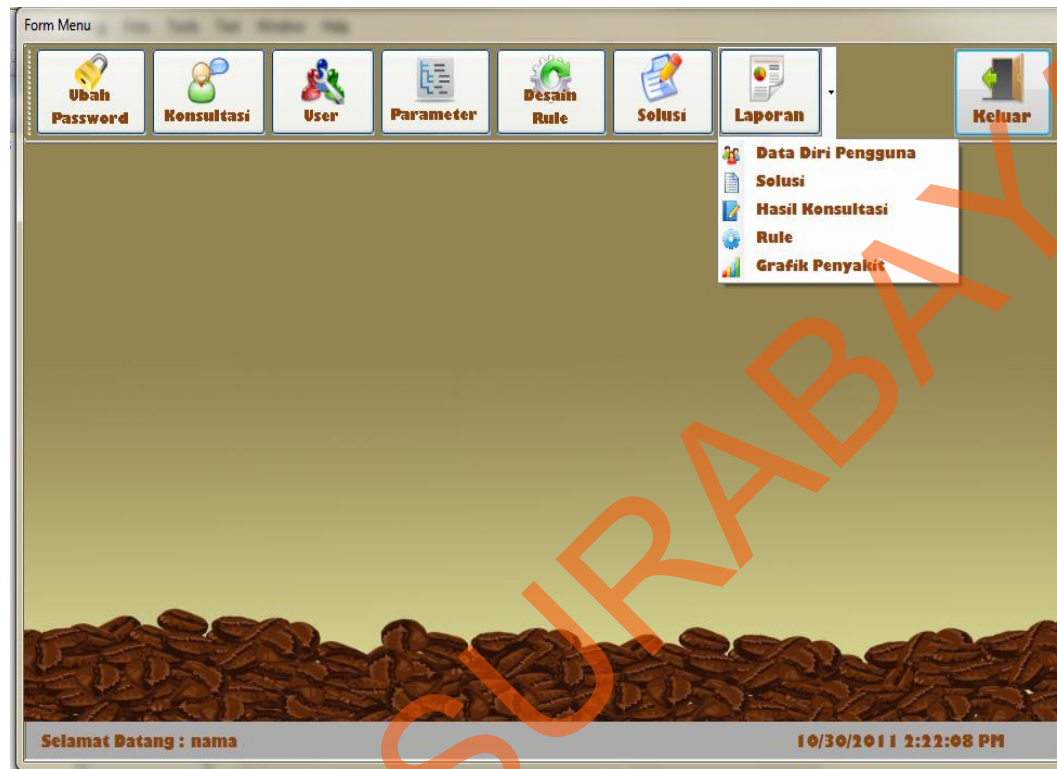
Gambar 4.4 *Form Daftar User*

4.2.5 Form Menu

Form Menu ini memiliki delapan *menu*, yaitu *menu* Ubah Password, Konsultasi, Data User, Parameter, Desain Rule, Solusi, Laporan dan Keluar. *Menu* Laporan memiliki lima sub *menu*, yaitu Laporan User, Solusi, Rule, Hasil Konsultasi dan Grafik Penyakit. Tampilan *form Menu* ini dapat dilihat pada Gambar 4.5.

Fungsi dari masing-masing *menu* adalah untuk memanggil *form* sesuai dengan menunya. *Menu* Ubah Password digunakan untuk menampilkan *form* Ubah Password, *menu* User digunakan untuk menampilkan *form* Data User, *menu* Parameter digunakan untuk menampilkan *form* Treeview, *menu* Desain Rule untuk menampilkan *form* Rule, Solusi untuk menampilkan *form* Solusi, Laporan

untuk menampilkan *form* Laporan sesuai dengan sub *menu* yang dipilih, dan *menu* Keluar untuk keluar dari *form* dan kembali ke *form* Utama.



Gambar 4.5 *Form Menu*

4.2.6 **Form Ubah Password**

Form Ubah Password ini hanya dapat diakses oleh *User* umum yang ingin mengubah *password*. *Form* ubah *password* ini dapat dilihat pada Gambar 4.6. *TextBox* ulangi *password* yang terlihat pada gambar digunakan untuk mengisi *password* yang sama dengan *password* baru yang diisi. Tombol *OK* yang terlihat pada Gambar 4.6 digunakan untuk menyimpan hasil mengubah *password*, kemudian menampilkan *form Menu*. Tombol *Batal* untuk membatalkan proses ubah *password* dan kembali ke *form Menu*.

Gambar 4.6 *Form Ubah Password*

4.2.7 Form Data User

Form Data User berfungsi untuk memelihara data-data pengguna pakar yang mengoperasikan sistem. *User ID* dihasilkan langsung oleh sistem sehingga pengguna tidak perlu memasukan *User ID* secara manual. *Form* ini dapat dilihat pada Gambar 4.7. *Form* ini memiliki lima tombol, yaitu tombol Simpan untuk menyimpan data *user* yang telah diisi, tombol Ubah untuk mengubah data *user* yang telah dipilih dari *form* Cari *User*, tombol Batal untuk membatalkan setiap masukan sebelum disimpan atau diubah, tombol Keluar untuk keluar dari *form* Data *User* dan tombol Cari untuk memanggil *form* Cari *User* seperti yang terlihat pada Gambar 4.8.

4.2.8 Form Cari User

Tampilan *Form* Cari *User* seperti yang terlihat pada Gambar 4.8 ini digunakan untuk mencari data *user* yang diubah datanya. Pencarian data *user* ini bisa dengan mengisi nama pada *textbox* nama setelah itu menekan *enter*. Data langsung disortir sesuai dengan nama yang diisi setelah itu *double* klik pada tabel untuk memilih data. Data *user* ini juga bisa dipilih langsung dari tabelnya.

Tombol Lihat Laporan digunakan untuk menampilkan *form* Laporan *User*, sedangkan tombol Batal untuk membatalkan proses cari *user* dan kembali ke *form* Data *User*.

Gambar 4.7 *Form* Data *User*

Nama	username	level	Alamat	No_Telp
Angriani An...	angie	ADMIN	Jln.Dukuh K...	08133272
Dewi Angkie	dewi	PAKAR	Jln.Dukuh K...	03151096
Hari Cahyadi	Harconk	ADMIN	Jln. Kendal ...	08573109
Herlin Mach...	herlin	PAKAR	Jln. Dukuh K...	031 56796
Marta Yakob	Martha	PAKAR	Jln. Dukuh K...	031 56716
Sonie Susanto	sonie	ADMIN	Bratang Ged...	03156716
Lendra wisesa	bozz	PAKAR	Jln.semolow...	23043040
Andrew Long	Andre	ADMIN	Jln. Kedung ...	03156716

Gambar 4.8 *Form* Cari *User*

4.2.9 Form Treeview

Form Treeview ini digunakan untuk mendesain dan membuat diagram ketergantungan (*dependency diagram*) dalam bentuk *treeview* yang merupakan langkah awal dalam membuat *rule*. Tampilan *form Treeview* ini dapat dilihat pada Gambar 4.9.

Tampilan *form Treeview* pada Gambar 4.9 memiliki empat tombol, yaitu tambah parameter, ubah parameter, hapus parameter dan keluar. Tombol Tambah parameter digunakan untuk menampilkan *form* Parameter ketika ingin menambah parameter pada *treeview* dengan memilih parameter induk (*parent node*) terlebih dahulu. Tombol ubah parameter digunakan untuk menampilkan *form* Parameter ketika ingin mengubah parameter yang dipilih pada *treeview*. Tombol hapus parameter digunakan untuk menghapus parameter yang dipilih pada *treeview*. Parameter yang dihapus tidak boleh parameter yang paling atas dan juga parameter yang masih memiliki anak atau parameter dibawahnya (*child node*).

4.2.10 Form Parameter

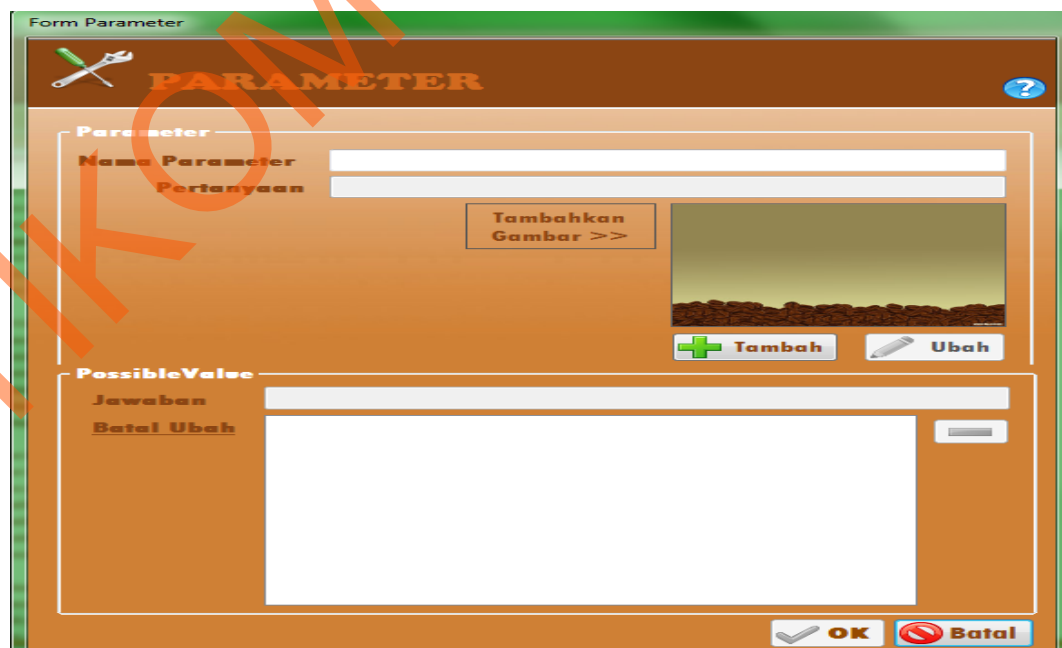
Form Parameter ini muncul setelah memilih tombol Tambah dan Ubah Parameter pada *form Treeview*. Desain *form* ini digunakan untuk menambah dan mengubah data parameter dan juga menambah, mengubah dan menghapus *value* atau jawaban pada setiap parameter yang dibuat. Desain *form* Parameter ini dapat dilihat pada Gambar 4.10.

Form ini memiliki tombol tambah yang digunakan untuk menyimpan atau menambah parameter baru. Tombol ubah digunakan untuk mengubah parameter yang dipilih dari *form Treeview*. Tombol yang bergambar kurang (-) ini digunakan untuk menghapus jawaban dengan memilih jawaban yang dihapus pada

list jawaban yang ada. Tombol *OK* digunakan untuk menyimpan hasil masukan dan kembali ke *form Treeview*, tombol batal digunakan untuk membatalkan proses dan kembali ke *form Treeview* dan *LinkLabel* Batal edit digunakan untuk membatalkan proses ubah *value*.



Gambar 4.9 *Form Treeview*



Gambar 4.10 *Form Parameter*

4.2.11 Form Desain Rule

Form ini digunakan untuk membuat *rules*. Sebelum menghasilkan *rule* yang benar *rule* harus diverifikasi terlebih dahulu. Langkah pertama yang harus dilakukan adalah memilih *set* yang diverifikasi. Setelah *set* dipilih, tekan tombol *generate decision table* untuk menampilkan *decision table* yang pada *form* *Tabel Keputusan*, kemudian pakar melakukan proses reduksi tabel. Hasil reduksi tabel ini dimunculkan pada *form* *Rule*. Setelah itu, pakar menekan tombol *generate rule* untuk menampilkan *rule* beserta jawabannya kemudian tekan tombol simpan. *Form* *Desain Rule* ini dapat dilihat pada Gambar 4.11.

The screenshot shows the 'Form Rule' interface with the following components:

- Form Header:** 'DESAIN RULE' and 'Nama Set: Jenis Penyakit Tanaman Kopi'.
- Tabel Keputusan (Decision Table):** A table with columns: Akar, Batang, Daun, Buah, Tanaman, and Jenis Penyakit Tanaman Kopi. It lists various symptoms and their corresponding diseases.
- Rule List:** A list of logical rules generated from the table, such as 'IF Akar = Akar terthup kerak yang terdiri atas butir-butir tanah, tampak jaringan jamur berwarna coklat tua, lunak, berbau, kering AND IF Batang = normal AND IF Daun = Menguning, Layu, Gugur AND IF Buah = normal AND IF Tanaman = normal AND THEN Jenis Penyakit Tanaman Kopi = Penyakit Cendawan Akar Caklat'.
- Buttons:** 'Generate Decision Table', 'Generate Rule', 'Simpan Rule', and 'Keluar'.

Akar	Batang	Daun	Buah	Tanaman	Jenis Penyakit Tanaman Kopi
Akar terthup kerak...	normal	Menguning, Layu, ...	normal	normal	Penyakit Cendawan Akar Caklat
Akar membusuk, d...	normal	menguning, Gugur	berbau meskipun ...	tanaman kopi meny...	Penyakit Akar Putih
mengeluarkan cair...	Muncul rebahan pa...	menguning, Layu	normal	Tampak terbayu pa...	Penyakit Berek Akar
normal	muncul bercak mer...	muncul bintik mera...	normal	normal	Penyakit Berek Daun Alga
normal	Muncul bercak puti...	normal	normal	normal	Penyakit Jamur Ujap
normal	normal	Gugur, Di balik da...	normal	Tanaman Mengadi...	Penyakit Karat Daun
normal	normal	ada bercak berwar...	muncul bercak cokl...	normal	Penyakit Berek Daun
normal	normal	normal	normal	normal	Normal

Gambar 4.11 *Form* *Rule*

4.2.12 Form Tabel Keputusan

Form *Tabel Keputusan* terlihat seperti pada Gambar 4.12. *Tabel* *keputusan* (*decision table*) dibuat secara otomatis berdasarkan *set* yang telah dipilih sebelumnya pada *form* *rule*. Tombol *Reduksi* digunakan untuk melakukan proses reduksi terhadap *tabel* *keputusan*. Tombol *OK* digunakan untuk

menyimpan hasil reduksi tabel keputusan dan menutup *form* ini. Tombol *OK* menjadi aktif apabila proses reduksi telah dilakukan. Tombol *Batal* digunakan untuk menutup *form* *Tabel Keputusan* dan kembali ke *form* *Rule* tanpa menyimpan tabel keputusan.

berbuah sebelum umumnya	Munculnya bercak coklat pada bagian buah	Buah yang terkena bercak coklat menjadi busuk	Kesimpulan
Ya	Ya	Ya	Tidak teridentifikasi

Gambar 4.12 *Form* *Tabel Keputusan*

4.2.13 *Form* *Solusi*

Form *Solusi* ini digunakan untuk memasukkan penyebab penyakit, gejala-gejala penyakit dan solusi tindakan dari setiap kesimpulan yang ada. Desain *form* ini berisi nama-nama penyakit, penyebab penyakit, gejala penyakit dan tindakan pengendalian penyakit untuk tanaman kopi. Untuk memasukkan data pada *form* *Solusi*, *double* klik pada baris nama penyakit untuk menampilkan *form* *Isi Solusi* sesuai dengan nama penyakit yang dipilih. Tombol *Keluar* yang terlihat pada gambar, digunakan untuk keluar dari *form* *Solusi* dan kembali ke *form* *Menu*. *Form* *Solusi* ini dapat dilihat pada Gambar 4.13.

Form Solusi

SOLUSI

Solusi

	Nama Penyakit	Penyebab	Gejala Penyakit
▶	Penyakit Cendawan Akar Coklat	Jamur <i>Phellinus noxius</i>	a. Daunnya menguning, layu dan gugur. b. Pada akar tertutup kerak yang terdiri
	Penyakit Akar Putih	Cendawan <i>Rigidoporus microporus</i>	a. Daun menguning kemudian gugur. b. Munculnya bunga dan buah meskipun umu
	Penyakit Busuk Akar	Cendawan <i>Armillaria tabascens</i>	a. Daun menguning dan layu. b. Muncul retakan pada batang yang menandakan l
	Penyakit Jamur Upas	Jamur <i>Upasia salmonicolor</i>	a. Muncul bercak putih pada batang yang tidak terkena sinar matahari. b. Pada te
	Penyakit Karat Daun	Jamur <i>Hemileia vastatrix</i>	a. Pada sisi bawah daun terdapat bercak-bercak berwarna kuning muda kemudi
	Penyakit Bercak Daun	Jamur <i>Cercospora coffeicola</i>	a. Pada daun yang sakit timbul bercak berwarna kuning. b. Pada buahnya timbul
	Penyakit Bercak Daun Alga	Alga <i>Cephaleuros virescens</i>	a. Pada permukaan atas daun muncul bintik merah kecoklatan berbentuk cakram

** Double klik pada tabel untuk memilih data

Keluar

Gambar 4.13 *Form Solusi*

4.2.14 Form Isi Solusi

Form Isi Solusi muncul saat pengguna melakukan *double* klik pada tabel solusi yang ada pada *form Solusi* (Gambar 4.13). *Form Isi Solusi* yang terlihat pada Gambar 4.14 digunakan untuk mengisi kesimpulan atau penyebab penyakit, gejala penyakit dan solusi tindakan pengendalian dari setiap jenis penyakit pada tanaman kopi dan juga menambahkan gambar penyakit. Tombol *OK* digunakan untuk menyimpan hasil pengisian yang dilakukan, sedangkan tombol *Batal* digunakan untuk membatalkan proses dan kembali ke *form Solusi*.

Form Isi Solusi

ISI SOLUSI

Nama Penyakit Penyakit Cendawan Akar Coklat

Penyebab Jamur *Phellinus noxius*

Gejalanya :

- Daunnya menguning, layu dan gugur.
- Pada akar tertutup kerak yang terdiri atas butir-butir tanah yang melekat sangat kuat, sehingga tidak dapat terlepas.
- Di antara butir-butir tanah tersebut tampak jaringan jamur yang berwarna coklat tua sampai coklat kehitaman.
- Akar menjadi busuk, kering dan lunak.

Tindakan Pengendalian

- Melakukan pembongkaran pada tanaman yang sakit, sisa-sisa akar diambil dan dibakar
- Membuat saluran isolasi di tempat yang terinfeksi
- Melakukan peremajaan, dengan membongkar tanaman yang sudah tua hingga tidak dijumpai tunggul pohon-pohon tua.

Tambahkan Gambar >>



OK Batal

Gambar 4.14 *Form* Isi Solusi

4.2.15 Form Konsultasi

Form Konsultasi ini digunakan oleh pengguna untuk melakukan konsultasi diagnosis penyakit yang menyerang tanaman kopi. Konsultasi dilakukan dengan memilih terlebih dahulu bagian dari tanaman kopi pada *combobox* yang tersedia, kemudian pengguna dapat memilih gejala-gejala yang ditampilkan oleh sistem. Setelah itu pengguna menekan tombol Diagnosis untuk menampilkan *form* Hasil

Konsultasi yang berisi kesimpulan akhir. *Form* Konsultasi dapat dilihat pada Gambar 4.15. Tombol Keluar yang terlihat pada *form* ini digunakan untuk keluar dari *form* Konsultasi.

Gambar 4.15 *Form* Konsultasi

4.2.16 **Form Hasil Konsultasi**

Form Hasil Konsultasi digunakan untuk menampilkan hasil diagnosis penyakit sesuai dengan gejala-gejala yang telah dipilih oleh pengguna pada *form* Konsultasi (Gambar 4.15). Sistem akan menampilkan tiga kesimpulan jika gejala yang dipilih oleh pengguna berada pada ketiga jenis penyakit yang ada. *LinkLabel* yang terlihat pada *form* ini digunakan untuk menampilkan *form* Detil Penyakit

dari setiap kesimpulan yang dihasilkan. *Form* Hasil Konsultasi ini dapat dilihat pada Gambar 4.16.

Gambar 4.16 *Form* Hasil Konsultasi

4.2.17 *Form* Detil Penyakit

Form ini digunakan untuk menampilkan detil penyakit dari setiap kesimpulan yang terlihat pada *form* Hasil Konsultasi. Detil penyakit yang ditampilkan berupa penjelasan dari masing-masing penyakit, yaitu penyebabnya, gejalanya, cara pengendaliannya dan juga disertai dengan gambar. *Form* Detil penyakit ini dapat dilihat pada Gambar 4.17. Tombol OK yang terlihat pada *form* ini digunakan untuk menyimpan hasil konsultasi yang telah dilakukan dan menampilkan *form* laporan Hasil Konsultasi.

Gambar 4.17 *Form Detil Penyakit*

4.2.18 *Form Laporan User*

Form Laporan User ini digunakan untuk menampilkan semua pengguna yang menggunakan aplikasi sistem pakar ini. *Form Laporan User* ini hanya dapat dilihat oleh pengguna dengan level ADMIN. Laporan *user* ini dapat dilihat berdasarkan level pengguna, periode dan juga dapat ditampilkan berdasarkan level dan periodenya. Tombol Tampilkan digunakan untuk menampilkan laporan *user* berdasarkan pilihan yang dipilih. *Form Laporan User* ini dapat dilihat pada Gambar 4.18.

LAPORAN USER

Tampilkan Laporan Berdasarkan

Level & Periode Level: ADMIN Periode: 8/ 1/2011 S/D: 10/31/2011 **Tampilkan**

Main Report

LAPORAN DATA PENGGUNA

Level : ADMIN
Periode : 08/01/2011 S/D 10/31/2011

Nama	Alamat	No_Telp	Reg_date	level
Sonie Susanto	Bratang Gede 12/12, Surabaya	0315671653	08/10/2011	ADMIN
Andrew Long	Jln. Kedung Baruk 12/12	0315671653	09/16/2011	ADMIN
Samuel Pokey	Jln. Semampir aws 1/1	031234567	09/16/2011	ADMIN

Total Pengguna : 3

Current Page No.: 1 Total Page No.: 1 Zoom Factor: 100%

Gambar 4.18 *Form Laporan User*

4.2.19 Form Laporan Hasil Konsultasi

Form Laporan Hasil Konsultasi ini digunakan untuk menampilkan laporan hasil konsultasi yang dilakukan oleh pengguna yang sedang *login*. *Form* ini dapat dipanggil melalui *menu* Laporan dengan sub *menu* Hasil Konsultasi (Gambar 4.5). Laporan hasil konsultasi ini dapat ditampilkan berdasarkan nama penyakit atau tanggal konsultasi dan juga dapat ditampilkan berdasarkan nama dan tanggal konsultasi. Tombol Tampilkan digunakan untuk menampilkan laporan sesuai dengan pilihan yang dipilih. Desain *form* Laporan Hasil Konsultasi dapat dilihat pada Gambar 4.19.

Gejalanya	Solusi Tindakan
a. Pada sisi bawah daun terdapat bercak-bercak berwarna kuning muda kemudian berubah menjadi kuning tua. b. Pada bercak-bercak tersebut terdapat tepung berwarna jingga cerah yang terdiri atas jamur karat. c. Bercak yang tua berwarna coklat tua sampai hitam mengering.	a. Menggunakan varietas kopi yang tahan. b. Menggunakan mikrobial yang bersifat berlawanan, yaitu bakteri <i>Bacillus thuringiensis</i> dan jamur <i>Verticillium hemileiae</i> . c. Penggunaan fungisida, misalnya oksiklorida tembaga dengan aliran

Gambar 4.19 *Form* Laporan Hasil Konsultasi

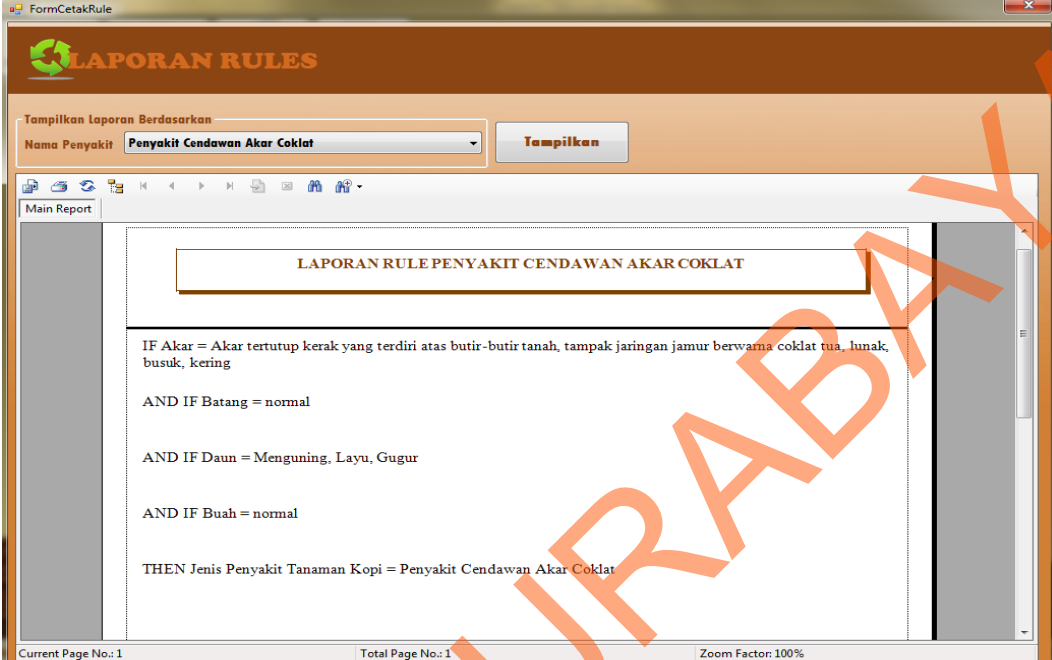
4.2.20 Form Laporan Rule

Form Laporan *rule* ini digunakan untuk menampilkan laporan *rules* atau aturan-aturan yang telah dibuat. *Form* ini dapat dipanggil melalui *menu* Laporan dengan sub *menu* Rule (Gambar 4.5). Laporan *rule* ini dapat ditampilkan berdasarkan nama penyakitnya. Tombol Tampilkan digunakan untuk menampilkan laporan sesuai dengan pilihan yang dipilih. Desain *form* Laporan Rule ini dapat dilihat pada Gambar 4.20.

4.2.21 Form Laporan Solusi

Form Laporan Solusi ini digunakan untuk menampilkan laporan penjelasan dari setiap jenis penyakit tanaman kopi. Data yang ditampilkan adalah nama penyakitnya, penyebab dan cara pengendaliannya. *Form* ini dapat dipanggil

melalui *menu* Laporan dengan sub *menu* Solusi (Gambar 4.5). Desain *form* Laporan Rule ini dapat dilihat pada Gambar 4.21



The screenshot shows a web application window titled 'FormCetakRule'. At the top, there is a header with a logo and the text 'LAPORAN RULES'. Below the header, there is a dropdown menu for 'Nama Penyakit' with the value 'Penyakit Cendawan Akar Coklat' and a 'Tampilkan' button. The main content area is titled 'Main Report' and contains the following text:

LAPORAN RULE PENYAKIT CENDAWAN AKAR COKLAT

IF Akar = Akar tertutup kerak yang terdiri atas butir-butir tanah, tampak jaringan jamur berwarna coklat tua, lunak, busuk, kering

AND IF Batang = normal

AND IF Daun = Menguning, Layu, Gugur

AND IF Buah = normal

THEN Jenis Penyakit Tanaman Kopi = Penyakit Cendawan Akar Coklat

At the bottom of the window, there is a status bar with 'Current Page No.: 1', 'Total Page No.: 1', and 'Zoom Factor: 100%'.

Gambar 4.20 *Form* Laporan Rule



The screenshot shows a web application window titled 'Form laporan solusi'. At the top, there is a header with a logo and the text 'LAPORAN SOLUSI'. Below the header, there is a dropdown menu for 'Nama Penyakit' with the value 'Penyakit Cendawan Akar Coklat' and a 'Tampilkan' button. The main content area is titled 'Main Report' and contains the following text:

LAPORAN PENJELASAN JENIS PENYAKIT TANAMAN KOPI

Nama Penyakit : Penyakit Cendawan Akar Coklat

Penyebabnya : Jamur *Phellinus noxius*

Cara Pengendalian

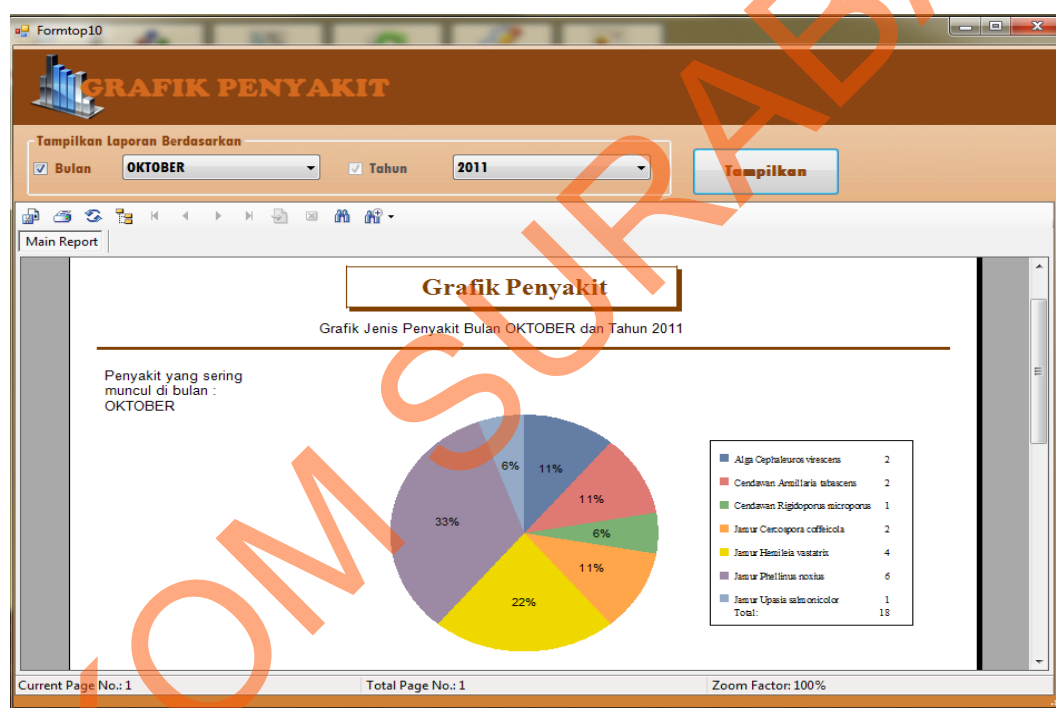
- a. Dilakukan pembongkaran pada tanaman yang sakit, sisa-sisa akar diambil dan dibakar.
- b. Membuat saluran isolasi di tempat yang terinfeksi.
- c. Melakukan peremajaan, dengan membongkar tanaman yang sudah tua hingga tidak dijumpai tunggul pohon-pohon tua.

At the bottom of the window, there is a status bar with 'Current Page No.: 1', 'Total Page No.: 4', and 'Zoom Factor: 100%'.

Gambar 4.21 *Form* Laporan Solusi

4.2.22 Form Laporan Grafik Penyakit

Form Laporan Grafik Penyakit ini digunakan untuk menampilkan laporan penyakit yang sering muncul dalam bentuk grafik. Laporan ini dapat ditampilkan per bulan dan juga per tahun. *Form* ini dapat dipanggil melalui *menu* Laporan dengan sub *menu* Grafik Penyakit (Gambar 4.5). Tombol Tampilkan digunakan untuk menampilkan laporan sesuai dengan pilihan yang dipilih. Desain *form* Laporan Grafik Penyakit ini dapat dilihat pada Gambar 4.22.



Gambar 4.22 *Form* Laporan Grafik Penyakit

4.3 Uji Coba

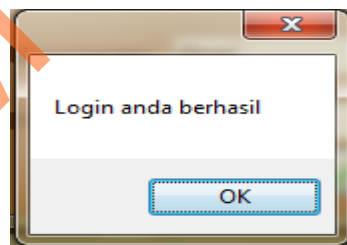
Berdasarkan pada perancangan uji coba yang terdapat pada sub bab 3.3, aplikasi ini akan diujicobakan terhadap data masukan sebenarnya. Uji coba dilakukan untuk memastikan bahwa aplikasi telah dibuat dengan benar sesuai dengan kebutuhan atau tujuan yang diharapkan. Proses uji coba ini menggunakan

black box testing untuk menguji aplikasi dengan melakukan berbagai percobaan untuk membuktikan bahwa aplikasi telah dibuat sesuai dengan tujuan. Proses uji coba dibagi menjadi dua, yaitu uji coba sistem dan uji coba kasus.

4.3.1 Uji Coba Sistem

A Uji Coba Form Login

Uji coba *form Login* ini bertujuan untuk mengecek setiap masukan yang dilakukan oleh pengguna telah sesuai atau tidak. Proses *Login* ini dilakukan dengan cara mengisi *Username* dan *Password*. Masukkan data 1 seperti yang terlihat pada Tabel 3.31, kemudian menekan tombol *OK*, maka muncul pesan “*Login* Anda berhasil”. Contoh pesan ini dapat dilihat pada Gambar 4.23. Jika *Username* yang dimasukan adalah data 2 dengan *password* yang beda, maka muncul pesan *error* “*Password* anda salah”. Contoh pesan *error* ini dapat dilihat pada Gambar 4.24.



Gambar 4.23 Pesan *Login* Valid



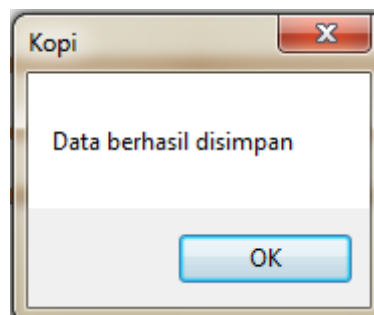
Gambar 4.24 Pesan *Error* Saat Proses *Login*

Tabel 4.1 Hasil Uji Coba *Form Login*

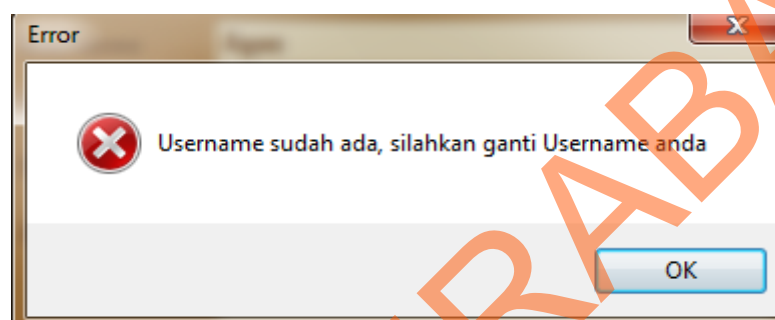
No	Tujuan	Input	Output yang diharapkan	Output system
1	Mendeskripsikan <i>Username</i> dan <i>Password</i> yang <i>valid</i>	Memasukkan data 1 seperti pada Tabel 3.31	Muncul Pesan “ <i>Login</i> anda berhasil” dan <i>menu</i> aktif sesuai dengan level pengguna	Aplikasi Sesuai. Muncul Pesan “ <i>Login</i> anda berhasil” (Gambar 4.23) dan <i>menu</i> aktif sesuai dengan level pengguna
2	Mendeskripsikan <i>Username</i> atau <i>Password</i> yang <i>invalid</i>	Memasukkan data 2 seperti pada Tabel 3.31 dengan <i>Password</i> nya diganti : believe	Muncul Pesan <i>Error</i> “ <i>Password</i> anda salah”	Aplikasi Sesuai. Muncul Pesan <i>Error</i> “ <i>Password</i> anda salah” (Gambar 4.24)

B Uji Coba Form Data User

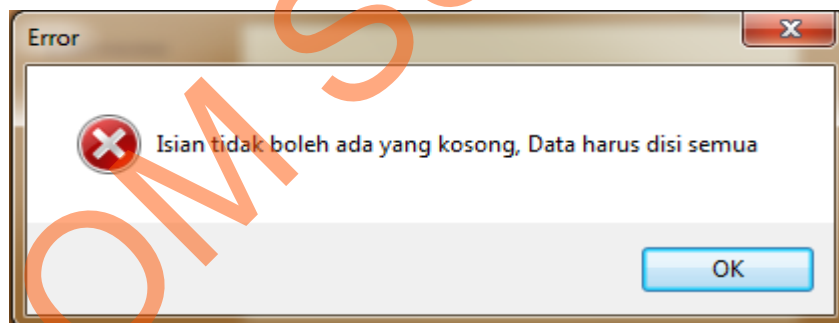
Uji coba *form* Data *User* ini bertujuan untuk mengetahui keberhasilan proses simpan dan ubah data *user* yang dapat dilakukan melalui *form* Data *User*. Untuk mengetahui proses penyimpanan berhasil, masukan Data 1 yang terlihat pada Tabel 3.33, kemudian menekan tombol Simpan, maka muncul pesan “Data berhasil disimpan”. Contoh pesan ini dapat dilihat pada Gambar 4.25. Jika pengguna mengisi *username* yang sama dengan *username* yang sudah ada, kemudian menekan tombol Simpan, maka muncul pesan *error* “*Username* sudah ada, silahkan ganti *username* anda”. Contoh pesan *error* ini dapat dilihat pada Gambar 4.26. Gambar 4.27 menampilkan pesan *error* “*Isian* tidak boleh ada yang kosong, data harus diisi semua”. Pesan *error* ini dimaksudkan ketika pada proses simpan ada *fields* yang tidak diisi atau kosong.



Gambar 4.25 Pesan Sukses Proses Simpan Data *User*



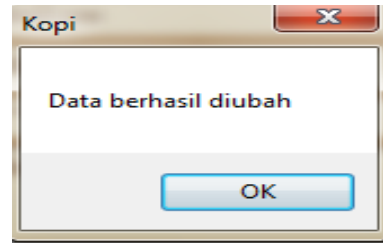
Gambar 4.26 Pesan *Error* Ketika Mengisi *Username* Yang Sama



Gambar 4.27 Pesan *Error* Saat Proses Simpan

Langkah pertama yang dilakukan untuk melakukan perubahan pada data *User* adalah tekan tombol cari untuk menampilkan *form* Cari *User* seperti yang terlihat pada Gambar 4.8, kemudian cari nama yang akan diubah datanya, klik dua kali untuk memilih data dan secara otomatis sistem akan menampilkan data yang dipilih tersebut pada *form* Data *User*. Selanjutnya, masukan data yang diubah,

kemudian tekan tombol Ubah, maka muncul pesan “Data berhasil diubah” seperti yang terlihat pada Gambar 4.28.



Gambar 4.28 Pesan Sukses Proses Ubah Data *User*

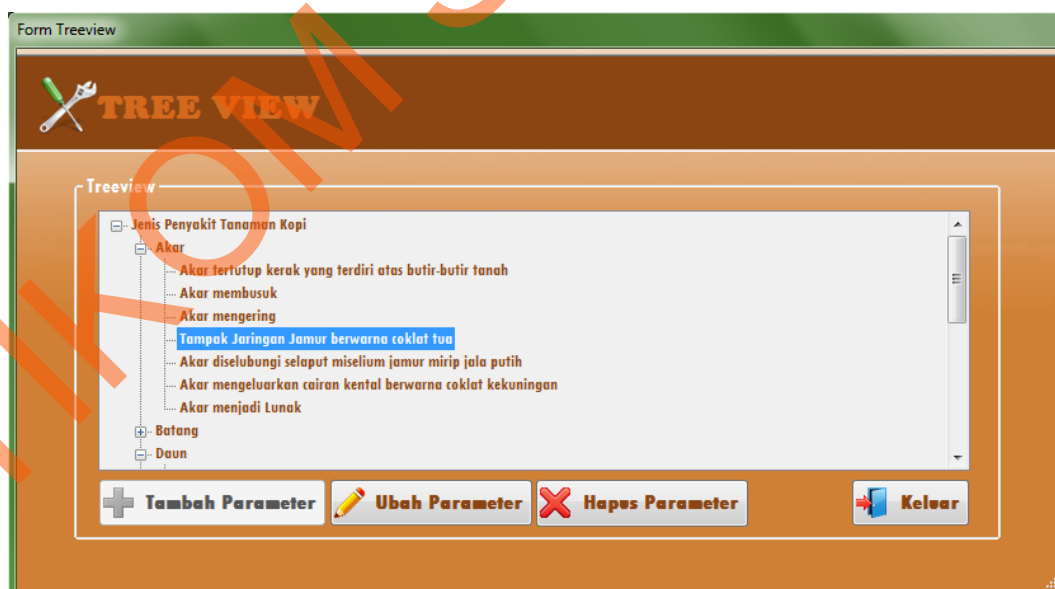
Tabel 4.2 Hasil Uji Coba *Form Data User*

No	Tujuan	Input	Output yang diharapkan	Output system
1	Menyimpan data <i>user</i> baru ke dalam <i>database</i>	Memasukkan data 1 seperti pada Tabel 3.33	Muncul Pesan “Data berhasil disimpan”	Aplikasi Sesuai. Muncul Pesan “Data berhasil disimpan” (Gambar 4.25)
2	Mengubah data <i>User</i>	Mengubah No. Telp data 1 pada Tabel 3.33 menjadi 085731096123	Muncul Pesan “Data berhasil diubah”	Aplikasi Sesuai. Muncul Pesan “Data berhasil diubah” (Gambar 4.28)
3	Menghindari ada <i>field</i> yang kosong saat proses penyimpanan	Memasukkan data kosong pada salah satu <i>field</i>	Muncul Pesan <i>Error</i> “Isian tidak boleh ada yang kosong, Data harus diisi semua”	Aplikasi Sesuai. Muncul Pesan <i>Error</i> (Gambar 4.27)
4	Menghindari pengisian <i>username</i> yang sama	Memasukan <i>Username</i> yang sama dengan <i>username</i> sebelumnya, kemudian tekan Tombol Simpan	Muncul Pesan <i>Error</i> “Username sudah ada, Silahkan ganti Username anda”	Aplikasi Sesuai. Muncul Pesan <i>Error</i> “Username sudah ada, Silahkan ganti Username anda” (Gambar 4.26)

C Uji Coba Form Treeview

Uji coba ini bertujuan untuk memeriksa fungsi pembuatan *dependency diagram* secara dinamis. Pada uji coba pertama adalah melakukan perubahan terhadap parameter yang telah ada. Langkah pertama adalah memilih nama parameter yang akan diubah seperti ditunjukkan pada Gambar 4.29. Setelah memilih parameter, tekan tombol Ubah Parameter dan akan muncul *Form Parameter* seperti pada Gambar 4.10.

Data yang dipilih pada Gambar 4.29 akan ditampilkan pada *form Parameter*, kemudian nama parameter diubah menjadi “Tampak Jaringan Jamur berwarna coklat tua”, kemudian tekan tombol Ubah pada *form Parameter*, seperti yang terlihat pada Gambar 4.30, kemudian menekan tombol *OK* untuk kembali ke *form Treeview*, dan data yang baru saja diubah akan terlihat seperti pada Gambar 4.31.



Gambar 4.29 Memilih Parameter Yang Akan Diubah Namanya

Form Parameter

PARAMETER

Parameter

Nama Parameter Pada akar Tampak Jaringan Jamur berwarna coklat tua

Pertanyaan apakah Tampak Jaringan Jamur berwarna coklat tua pada bagian akar?

Ubah Gambar >>

+ Tambah Ubah

PossibleValue

Jawaban

Batal Ubah Ya
Tidak

OK Batal

Gambar 4.30 Tampilan *Form* Parameter Untuk Proses Ubah Parameter

Form Treeview

TREE VIEW

Treeview

Jenis Penyakit Tanaman Kopi

- Akar
 - Akar tertutup kerak yang terdiri atas butir-butir tanah
 - Akar membusuk
 - Akar mengering
 - pada akar tampak Jaringan Jamur berwarna coklat tua
 - Akar diselubungi selaput miselium jamur mirip jala putih
 - Akar mengeluarkan cairan kental berwarna coklat kekuningan

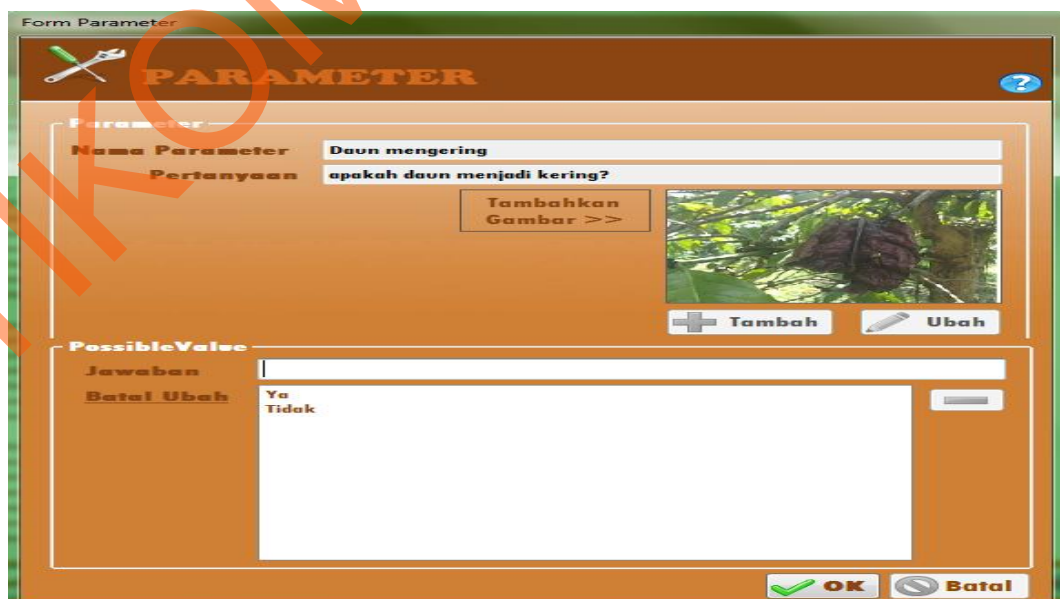
+ Tambah Parameter Ubah Parameter Hapus Parameter Keluar

Gambar 4.31 Hasil Ubah Parameter

Uji coba selanjutnya adalah melakukan penambahan parameter. Langkah pertama adalah memilih parameter yang akan dijadikan induk. Misalnya parameter Daun dipilih sebagai induk, lihat Gambar 4.32, kemudian tekan tombol Tambah Parameter dan akan muncul *form* Parameter. Kemudian isi data parameter seperti yang terlihat pada Gambar 4.33, kemudian tekan tombol Tambah pada *form* Parameter. Data yang baru saja dimasukkan muncul pada *Form Treeview* seperti pada Gambar 4.34.



Gambar 4.32 Memilih Parameter Induk

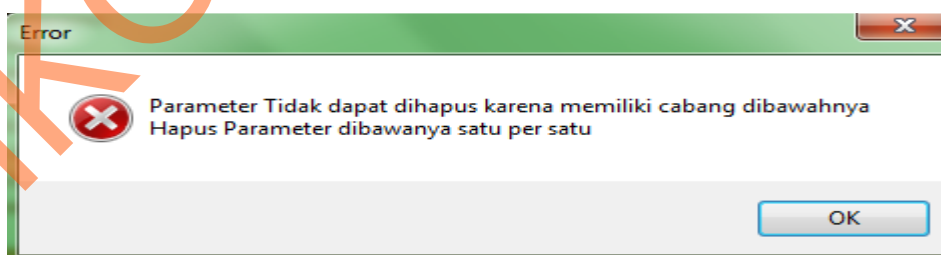


Gambar 4.33 Proses Tambah Parameter



Gambar 4.34 Hasil Penambahan Parameter

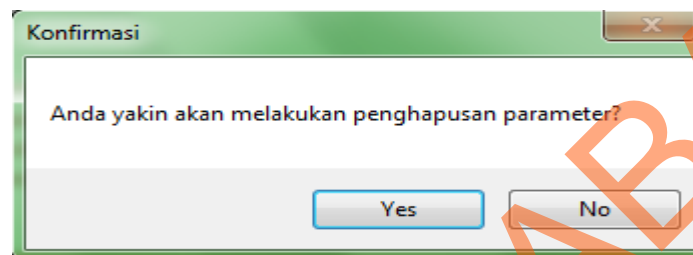
Uji coba selanjutnya adalah melakukan penghapusan salah satu parameter. Pilih parameter yang akan dihapus dan tekan tombol Hapus Parameter. Misalnya parameter yang dipilih adalah Akar (lihat Gambar 4.32), maka akan keluar pesan kesalahan seperti tampak pada Gambar 4.35 dan sistem tidak melakukan penghapusan parameter. Sistem mengeluarkan pesan kesalahan karena parameter akar yang akan dihapus memiliki anak (*child node*) parameter di bawahnya.



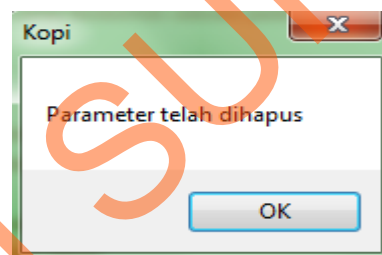
Gambar 4.35 Pesan *Error* Pada Proses Hapus Parameter Yang Memiliki Cabang Di bawahnya

Apabila parameter yang dipilih tidak memiliki cabang maka akan keluar pesan konfirmasi penghapusan seperti tampak pada Gambar 4.36. Jika memilih

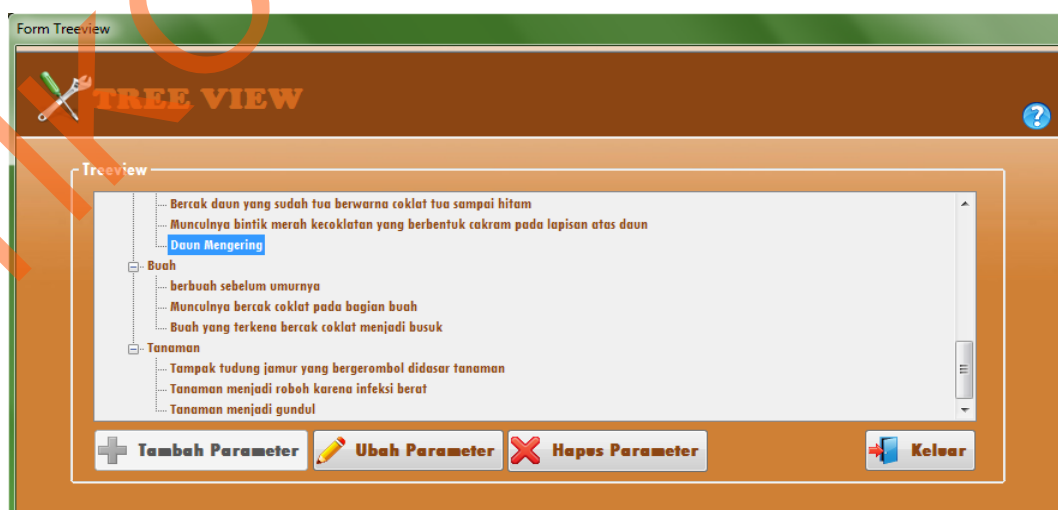
tombol *No* maka proses penghapusan tidak jadi dilakukan, tetapi jika memilih tombol *Yes* maka proses penghapusan akan dilakukan, dan muncul pesan seperti pada Gambar 4.37. Misalnya parameter yang dihapus adalah parameter Daun mengering pada Gambar 4.38 maka hasil penghapusan parameter akan tampak seperti pada Gambar 4.39.



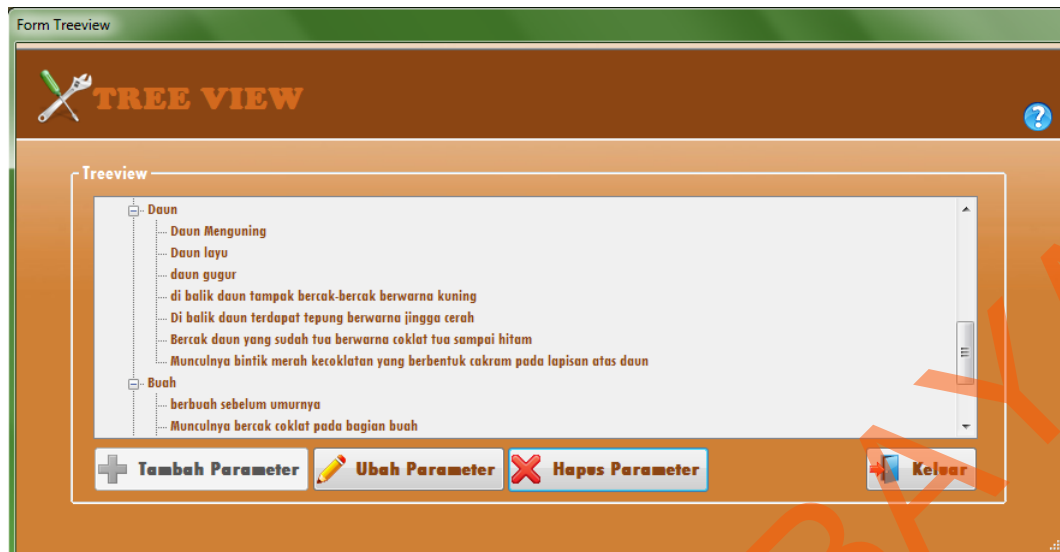
Gambar 4.36 Pesan Konfirmasi Penghapusan Parameter



Gambar 4.37 Pesan Ketika Parameter Berhasil Dihapus

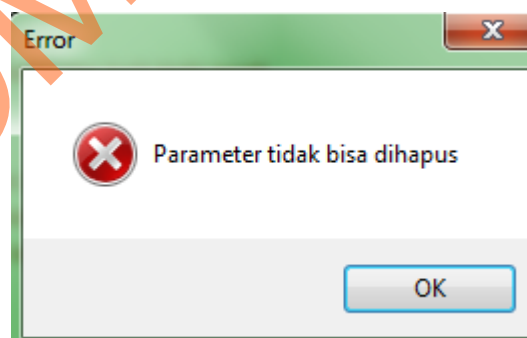


Gambar 4.38 Parameter “Daun Mengering” Yang Akan Dihapus



Gambar 4.39 Hasil Penghapusan Parameter “Daun Mengering”

Dalam proses penghapusan parameter ada pengecualian pada parameter yang berada paling atas, yaitu Jenis Penyakit Tanaman Kopi. Parameter ini tidak dapat dihapus meskipun tidak memiliki cabang di bawahnya. Apabila dilakukan penghapusan parameter Jenis Penyakit Tanaman Kopi, maka akan keluar pesan kesalahan seperti tampak pada Gambar 4.40



Gambar 4.40 Pesan *Error* Proses Hapus Parameter

Tabel 4.3 Hasil Uji Coba *Form Treeview*

No	Tujuan	Input	Output yang diharapkan	Output sistem
1	Nama Parameter akan berubah ketika melakukan perubahan nama parameter	Pilih parameter yang akan diubah kemudian tekan Ubah Parameter. Ubah data pada <i>form</i> Parameter.	Nama Parameter berubah sesuai dengan data yang telah dimasukkan.	Aplikasi sesuai. (Gambar 4.31).
2	Parameter pada <i>form Treeview</i> akan bertambah ketika ada penambahan Parameter	Pilih parameter yang akan dijadikan induk pada <i>form Treeview</i> kemudian tekan Tambah Parameter. Isikan data	Muncul parameter baru sesuai dengan data yang telah ditambahkan.	Aplikasi sesuai. (Gambar 4.34)
3	Menghindari proses hapus parameter yang memiliki cabang dibawahnya	Memilih nama parameter yang memiliki <i>child</i> kemudian tekan tombol Hapus Parameter	Muncul Pesan <i>Error</i> "Parameter tidak dapat dihapus karena memiliki cabang dibawahnya, Hapus parameter dibawahnya satu per satu"	Aplikasi sesuai. Muncul Pesan <i>error</i> seperti pada Gambar 4.35
4	Menghindari proses hapus parameter yang paling atas (nama penyakit)	Memilih nama parameter yang paling atas pada <i>treeview</i> (Jenis Penyakit Tanaman kopi) kemudian menekan tombol Hapus Parameter	Muncul Pesan <i>Error</i> "Parameter tidak dapat dihapus"	Aplikasi sesuai. Muncul Pesan <i>error</i> seperti pada Gambar 4.40
5	Menghapus parameter	Memilih parameter pada <i>treeview</i> yang akan dihapus kemudian tekan tombol Hapus Parameter	Muncul Pesan "anda yakin melakukan penghapusan parameter", jika memilih <i>No</i> , maka parameter tidak dihapus, jika memilih <i>Yes</i> , maka muncul pesan "Parameter telah Dihapus"	Aplikasi sesuai. Muncul Pesan seperti pada Gambar 4.36 dan Gambar 4.37

D Uji Coba *Form* Parameter

Uji coba *form* Parameter ini bertujuan untuk memeriksa proses tambah, ubah parameter dan juga untuk mengecek proses simpan, ubah dan hapus *value* atau jawaban yang dilakukan pada *form* Parameter ini. Pada uji coba pertama adalah melakukan penambahan parameter. Langkah pertama adalah memilih parameter yang akan dijadikan induk. Misalnya parameter Akar dipilih sebagai induk (lihat Gambar 4.32), kemudian tekan tombol Tambah Parameter dan akan muncul *form* Parameter seperti pada Gambar 4.10. Kemudian isi data parameter seperti yang terlihat pada Gambar 4.33, kemudian tekan tombol Tambah pada *form* Parameter.

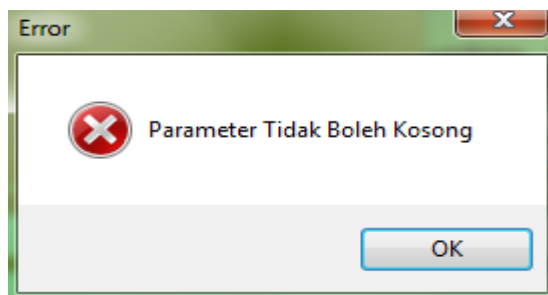
Uji coba menambah parameter dengan nama Buah, ketika menekan tombol Tambah, maka muncul pesan *error* seperti pada Gambar 4.41, karena parameter Buah sudah ada sebelumnya. Pesan *error* ini merupakan pesan *error* yang ditampilkan ketika pengguna memasukkan nama parameter yang sama dengan parameter yang sudah ada pada *Form Treeview*.



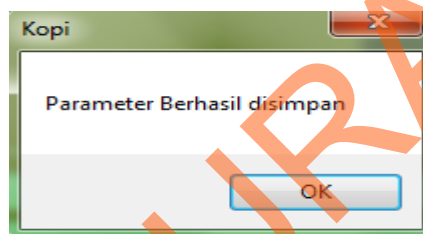
Gambar 4.41 Pesan *Error* Proses Tambah Parameter

Uji coba menambah parameter ini akan memunculkan pesan *error* seperti pada Gambar 4.42, ketika pengguna menekan tombol Tambah tanpa mengisi

nama parameter, sedangkan Gambar 4.43 merupakan tampilan pesan ketika proses tambah parameter berhasil dilakukan.

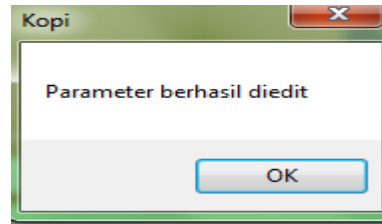


Gambar 4.42 Pesan *Error* Proses Tambah Parameter Yang Kosong



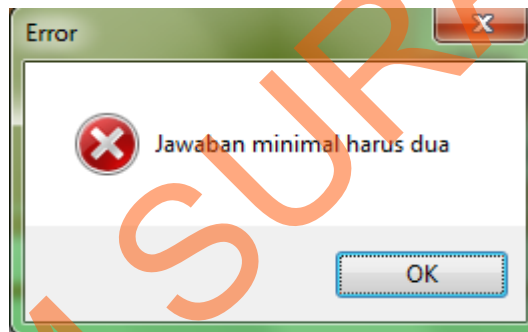
Gambar 4.43 Pesan Sukses Proses Tambah Parameter

Uji coba selanjutnya adalah melakukan proses Ubah parameter. Langkah pertama adalah memilih nama parameter yang akan diubah seperti ditunjukkan pada Gambar 4.29. Setelah memilih parameter, tekan tombol Ubah Parameter dan akan muncul *form* Parameter seperti pada Gambar 4.10. Parameter yang dipilih pada Gambar 4.29 akan ditampilkan pada *form* Parameter, kemudian nama parameter diubah menjadi “pada akar tampak jaringan jamur berwarna coklat tua”, kemudian tekan tombol Ubah pada *form* Parameter, seperti yang terlihat pada Gambar 4.30, maka muncul pesan seperti yang terlihat pada Gambar 4.44.



Gambar 4.44 Pesan Sukses Proses Ubah Parameter

Berdasarkan *test case id 3* pada Tabel 3.36, sistem tidak mengizinkan penyimpanan parameter yang hanya memiliki satu *value* atau jawaban. Apabila pengguna menekan tombol *OK* dengan hanya mengisi satu jawaban saja, maka akan keluar pesan *error* seperti terlihat pada Gambar 4.45.



Gambar 4.45 Pesan *Error* Proses Tambah *Value* Atau Jawaban

Berdasarkan *test case id 5* pada Tabel 3.36, sistem tidak mengizinkan penyimpanan jawaban yang sama dengan jawaban sebelumnya. Apabila pengguna menambah jawaban yang sama seperti yg terlihat pada Gambar 4.46 maka akan keluar pesan *error* seperti terlihat pada Gambar 4.47.

Form Parameter

PARAMETER

Parameter

Nama Parameter berbuah sebelum umurnya

Pertanyaan Apakah tanaman berbuah sebelum umurnya?

Ubah Gambar >>

Tambah Ubah

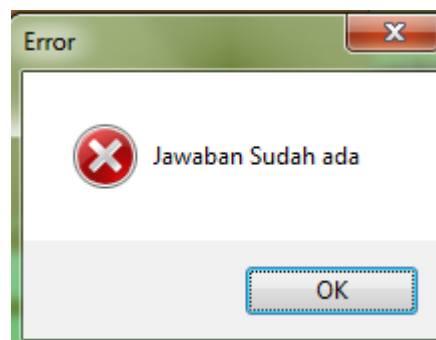
PossibleValue

Jawaban Ya

Ya
Tidak

OK Batal

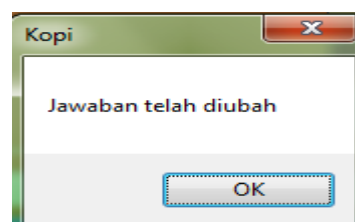
Gambar 4.46 Tampilan Proses Menambah Jawaban Yang Sama



Gambar 4.47 Pesan *Error* Proses Tambah Jawaban Yang Sama

Uji coba untuk mengubah jawaban, yaitu dengan memilih jawaban mana yang akan diubah pada pilihan jawaban yang ada seperti yang terlihat pada Gambar 4.48, misalnya parameter yang diubah jawabannya adalah parameter Buah dengan jawabannya “berbuah meskipun umurnya belum cukup” menjadi “berbuah sebelum umurnya”. Kemudian tekan *enter* untuk mengubah jawaban tersebut. Jika berhasil maka muncul pesan seperti yang terlihat pada Gambar 4.49 dan perubahannya dapat dilihat pada Gambar 4.50.

Gambar 4.48 Tampilan Jawaban Sebelum Diubah



Gambar 4.49 Pesan Sukses Proses Ubah Jawaban

Form Parameter

PARAMETER

Parameter

Nama Parameter Buah

Pertanyaan

Ubah Gambar >>

+ Tambah Ubah

PossibleValue

Jawaban

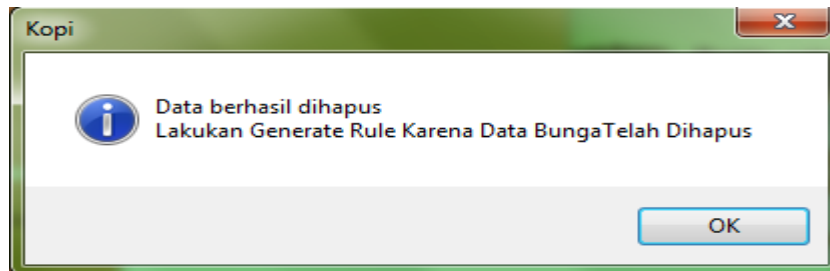
Batal Ubah

berbuah sebelum umurnya
muncul bercak coklat pada buahnya, buah menjadi busuk setelah k
Tidak teridentifikasi
normal

OK Batal

Gambar 4.50 Hasil Proses Ubah Jawaban

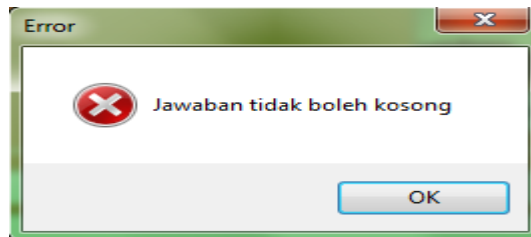
Uji coba untuk menghapus jawaban sama seperti mengubah jawaban, yaitu dengan memilih jawaban mana yang akan dihapus pada pilihan jawaban yang ada seperti yang terlihat pada Gambar 4.50, misalnya jawaban yang akan dihapus adalah “normal”, kemudian tekan tombol yang bergambar tanda kurang (-) untuk menghapus jawaban tersebut. Jika berhasil, maka muncul pesan seperti yang terlihat pada Gambar 4.51 dan perubahannya dapat dilihat pada Gambar 4.52.



Gambar 4.51 Pesan Sukses Proses Hapus Jawaban

Gambar 4.52 Hasil Proses Hapus Jawaban

Berdasarkan *test case id 8* pada Tabel 3.36, sistem tidak mengizinkan isian jawaban yang kosong ketika menekan tombol *OK*. Apabila pengguna menekan tombol *OK* tanpa mengisi jawaban maka akan keluar pesan *error* seperti terlihat pada Gambar 4.53.

Gambar 4.53 Pesan *Error* Jika Ada Jawaban Yang KosongTabel 4.4 Hasil Uji Coba *Form* Parameter

No	Tujuan	Input	Output yang diharapkan	Output Sistem
1	Menambah parameter baru	Pilih parameter yang akan dijadikan induk pada <i>treeview</i> , kemudian tekan tombol Tambah Parameter masukan data parameter baru pada <i>field</i> nama parameter dan pertanyaan kemudian tekan tombol Tambah	Muncul Pesan “Parameter berhasil disimpan”	Aplikasi sesuai. Muncul Pesan “Parameter berhasil disimpan” (Gambar 4.43).
2	Menghindari parameter kosong saat penyimpanan	Memasukkan data kosong pada <i>field</i> nama parameter kemudian tekan tombol Tambah	Muncul Pesan <i>Error</i> “Parameter tidak boleh kosong”	Aplikasi sesuai. Muncul Pesan <i>Error</i> (Gambar 4.42).
3	Menghindari jawaban kurang dari dua	Memasukkan satu jawaban pada <i>field</i> jawaban, kemudian tekan tombol <i>OK</i>	Muncul Pesan <i>Error</i> “Jawaban minimal harus dua”	Aplikasi sesuai. Muncul Pesan (Gambar 4.45).
4	Menghindari memasukkan Parameter yang sama	Mengisi nama parameter yang sama atau yang sudah ada dalam <i>Database</i> kemudian menekan tombol Tambah	Muncul Pesan <i>Error</i> “nama Parameter tidak boleh sama dengan nama parameter yang lainnya”	Aplikasi sesuai. Muncul Pesan <i>Error</i> (Gambar 4.41)

Tabel 4.4 Hasil Uji Coba *Form* Parameter (Lanjutan)

No	Tujuan	Input	Output yang diharapkan	Output Sistem
5	Menghindari memasukkan jawaban yang sama	Menambah jawaban yang sama atau sudah ada pada <i>field</i> Option Jawaban kemudian tekan <i>enter</i>	Muncul Pesan <i>Error</i> “Jawaban sudah ada”	Aplikasi sesuai. Muncul Pesan <i>Error</i> (Gambar 4.47)
6	Mengubah Jawaban	Memilih jawaban pada option jawaban dan menekan tombol edit	Muncul pesan “Jawaban telah diubah”	Aplikasi berhasil. Muncul Pesan <i>Error</i> (Gambar 4.49)
7	Mengubah data parameter	Memilih nama parameter pada <i>treeview</i> yang akan diubah kemudian tekan tombol Ubah Parameter, ubah nama kemudian tekan tombol Ubah	Muncul Pesan “Parameter berhasil diedit”	Aplikasi sesuai. Muncul Pesan <i>Error</i> (Gambar 4.44)
8	Menghindari option jawaban yang kosong ketika menekan tombol <i>OK</i>	Menekan tombol <i>OK</i> tanpa mengisi Jawaban	Muncul Pesan <i>Error</i> “Jawaban Tidak Boleh Kosong”	Aplikasi sesuai. Muncul Pesan <i>Error</i> (Gambar 4.53)
9	Menghapus Jawaban	Memilih jawaban pada option jawaban, kemudian tekan tombol yang bergambar tanda kurang (-)	Muncul pesan “Data telah dihapus”	Aplikasi berhasil. Muncul Pesan seperti pada Gambar 4.51

E Uji Coba Form Rule

Uji coba ini bertujuan untuk mencoba kemampuan sistem melakukan *generate rule* secara otomatis dengan menekan tombol Generate Rule. Sebelum melakukan *generate rule*, harus membuat *reduced decision table* terlebih dahulu

dengan menekan tombol Generate Decision Tabel untuk menampilkan *form* Tabel Keputusan dan melakukan proses reduksi tabel. Tabel yang telah direduksi inilah yang akan dijadikan sebagai *rule* dengan menekan tombol Generate Rule. Langkah pertama yang harus dilakukan adalah memilih nama set yang akan diisi tabel keputusannya. Sebagai contoh pilih set Buah seperti terlihat pada Gambar 4.54.

The screenshot shows a software window titled "DESAIN RULE" with a "Form Rule" header. A dropdown menu labeled "Nama Set" is set to "Buah". Below it is a "Tabel Keputusan" (Decision Table) with the following data:

berbuah sebelum umumnya	Munculnya bercak coklat pada bagian buah	Buah yang terkena bercak coklat menjadi busuk	Buah
Tidak	Ya	Ya	muncul bercak coklat pada buahnya, buah menjadi busuk setelah kena bercak
Ya	Tidak	Tidak	berbuah sebelum umumnya
Tidak	Tidak	Tidak	normal

To the right of the table is an empty "Rule" column. At the bottom of the window, there are four buttons: "Generate Decision Table", "Generate Rule", "Simpan Rule", and "Keluar".

Gambar 4.54 Reduksi Tabel Set Buah

Pada gambar di atas, terlihat sebuah tabel dari set Buah yang telah di reduksi sebelumnya dari *form* Tabel Keputusan. Kolom *rule* akan terisi, ketika pakar menekan tombol Generate Rule, yang hasilnya dapat dilihat pada Gambar 4.55. Berdasarkan *test case id* 1 pada Tabel 3.37, sistem menghindari proses penyimpanan sebelum melakukan *generate rule*, maka tombol Simpan Rule yang terlihat menjadi tidak aktif (lihat Gambar 4.54). Tombol Simpan Rule ini akan menjadi aktif setelah menekan tombol Generate Rule (lihat Gambar 4.55).

Form Rule

DESAIN RULE

Nama Set: Buah

Tabel Keputusan

berbuah sebelum umumnya	Munculnya bercak coklat pada bagian buah	Buah yang terkena bercak coklat menjadi busuk	Buah
Tidak	Ya	Ya	muncul bercak coklat pada buahnya, buah menjadi busuk setelah kena bercak
Ya	Tidak	Tidak	berbuah sebelum umumnya
Tidak	Tidak	Tidak	normal

Rule

IF Munculnya bercak coklat pada bagian buah = Ya
AND IF Buah yang terkena bercak coklat menjadi busuk = Ya
AND THEN Buah = muncul bercak coklat pada buahnya, buah menjadi busuk setelah kena bercak

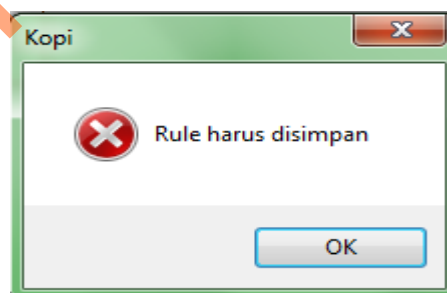
IF Munculnya bercak coklat pada bagian buah = Tidak
AND IF Buah yang terkena bercak coklat menjadi busuk = Tidak
AND THEN Buah = berbuah sebelum umumnya

IF Munculnya bercak coklat pada bagian buah = Tidak
AND IF Buah yang terkena bercak coklat menjadi busuk = Tidak
AND THEN Buah = normal

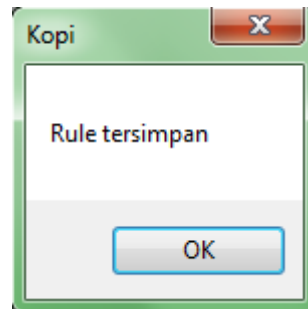
Generate Decision Table Generate Rule Simpan Rule Keluar

Gambar 4.55 Hasil Generate Rule

Berdasarkan *test case id 2* pada Tabel 3.37, sistem tidak mengizinkan keluar dari *form Rule* sebelum melakukan penyimpanan. Apabila pakar menekan tombol Keluar sebelum menyimpan *rule*, maka akan muncul pesan *error* seperti pada Gambar 4.56. Jika proses simpan *rule* berhasil maka akan muncul pesan seperti yang terlihat pada Gambar 4.57.



Gambar 4.56 Pesan Error Sebelum Melakukan Penyimpanan Rule

Gambar 4.57 Pesan Sukses Proses Simpan *Rule*Tabel 4.5 Hasil Uji Coba *Form Rule*

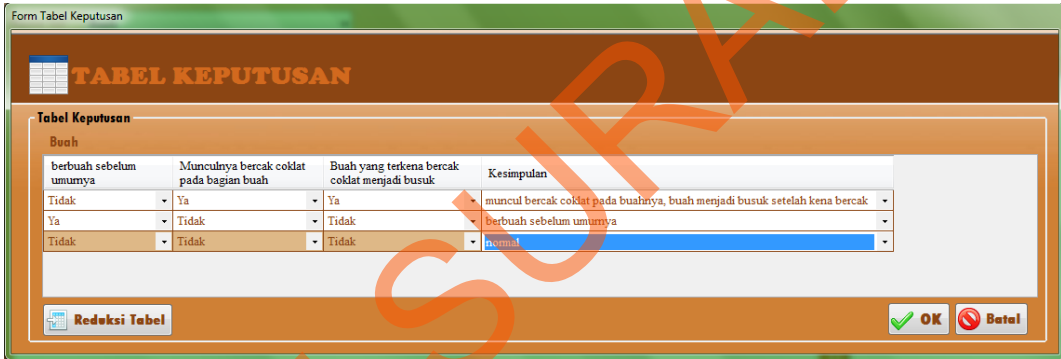
No	Tujuan	Input	Output yang diharapkan	Output Sistem
1	Menghindari simpan <i>rule</i> tanpa menekan tombol Generate Rule	Tidak menekan tombol Generate Rule untuk menghasilkan Rule-Rule	Tombol Simpan Rule tidak aktif	Aplikasi sesuai. Tombol Simpan Rule tidak Aktif (Gambar 4.54)
2	Keluar <i>form rule</i> tanpa melakukan penyimpanan rule	Tekan tombol keluar sebelum menyimpan hasil generate rule.	Muncul pesan <i>error</i> "Rule Harus Disimpan".	Aplikasi sesuai. muncul pesan <i>error</i> seperti Gambar 4.56

F Uji Coba Form Tabel Keputusan

Form Tabel Keputusan ini digunakan untuk memasukkan konklusi pada *decision table* (tabel keputusan) dan mereduksi *decision table* tersebut. Langkah pertama yang harus dilakukan adalah, memilih *set* Buah pada *form rule* dan menekan tombol *generate decision table*, maka muncul *form* Tabel Keputusan seperti Gambar 4.58, jika pengguna menekan tombol Reduksi Tabel, maka sistem menghasilkan reduksi tabel seperti yang terlihat pada Gambar 4.59.



NO	berbuah sebelum umumnya	Munculnya bercak coklat pada bagian buah	Buah yang terkena bercak coklat menjadi busuk	Kesimpulan
1	Ya	Ya	Ya	Tidak Teridentifikasi
2	Ya	Ya	Tidak	Tidak Teridentifikasi
3	Ya	Tidak	Ya	Tidak Teridentifikasi
4	Ya	Tidak	Tidak	berbuah sebelum umumnya
5	Tidak	Ya	Ya	muncul bercak coklat pada buahnya, buah menjadi busuk set...
6	Tidak	Ya	Tidak	Tidak Teridentifikasi
7	Tidak	Tidak	Ya	Tidak Teridentifikasi
8	Tidak	Tidak	Tidak	Normal

Gambar 4.58 *Decision Table* (Tabel Keputusan) Set Buah


TABEL KEPUTUSAN

Tabel Keputusan

Buah

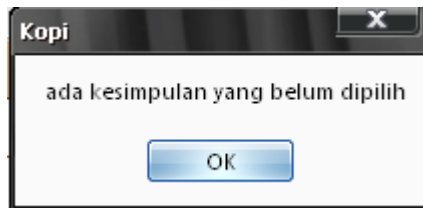
berbuah sebelum umumnya	Munculnya bercak coklat pada bagian buah	Buah yang terkena bercak coklat menjadi busuk	Kesimpulan
Tidak	Ya	Ya	muncul bercak coklat pada buahnya, buah menjadi busuk setelah kena bercak
Ya	Tidak	Tidak	berbuah sebelum umumnya
Tidak	Tidak	Tidak	normal

Reduksi Tabel

OK Batal

Gambar 4.59 *Reduction Decision Table* Set Buah

Berdasarkan *test case id 2* pada Tabel 3.38, sistem akan menampilkan pesan *error* seperti yang terlihat pada Gambar 4.60, jika pengguna tidak mengisi semua konklusi atau kesimpulan yang ada dan menekan tombol Reduksi Tabel. Sesuai dengan *test case id 3* dan *4* pada Tabel 3.38, tombol OK dan Batal menjadi *disable* (tidak aktif) sebelum proses reduksi tabel (lihat Gambar 4.58). Tombol OK menjadi aktif setelah proses reduksi berhasil dan Tombol Reduksi Tabel menjadi tidak aktif (lihat Gambar 4.59).

Gambar 4.60 Pesan *Error* Proses Reduksi TabelTabel 4.6 Hasil Uji Coba *Form* Tabel Keputusan

No	Tujuan	Input	Output yang diharapkan	Output Sistem
1	Menghasilkan <i>reduced decision table</i>	Mengisi konklusi dengan nama set yang terpilih kemudian menekan tombol Reduksi	Menghasilkan <i>reduced decision table</i>	Aplikasi sesuai. Gambar 4.59
2	Menghindari proses <i>reduced decision table</i> tanpa mengisi konklusi	Menekan tombol Reduksi tanpa mengisi konklusi	Muncul Pesan <i>Error</i> "ada kesimpulan yang belum dipilih"	Aplikasi sesuai. Muncul pesan seperti Gambar 4.60
3	Menghindari menekan tombol OK dan Batal sebelum mereduksi tabel	Mengisi konklusi dan tidak menekan tombol Reduksi	Tombol OK dan Batal tidak aktif.	Aplikasi sesuai. (Gambar 4.58)
4	Menghindari menekan tombol Reduksi setelah proses reduksi tabel dilakukan	Mengisi semua konklusi dan Menekan tombol Reduksi Tabel	Tombol OK aktif dan Tombol Reduksi tidak aktif	Aplikasi sukses (Gambar 4.59)

G Uji Coba Form Solusi

Uji coba pada *form* Solusi ini adalah untuk memastikan proses pengisian data solusi telah dilakukan dengan benar. Langkah pertama yang harus dilakukan adalah memilih data solusi yang diubah datanya. Nama penyakit yang akan diisi solusinya adalah "Cendawan Akar Coklat" (Gambar 4.61). Masukkan data pada Tabel 3.39 sesuai dengan *fields* yang ada, kemudian tekan tombol OK (Gambar 4.62). Jika berhasil maka sistem akan menampilkan pesan "Data solusi berhasil

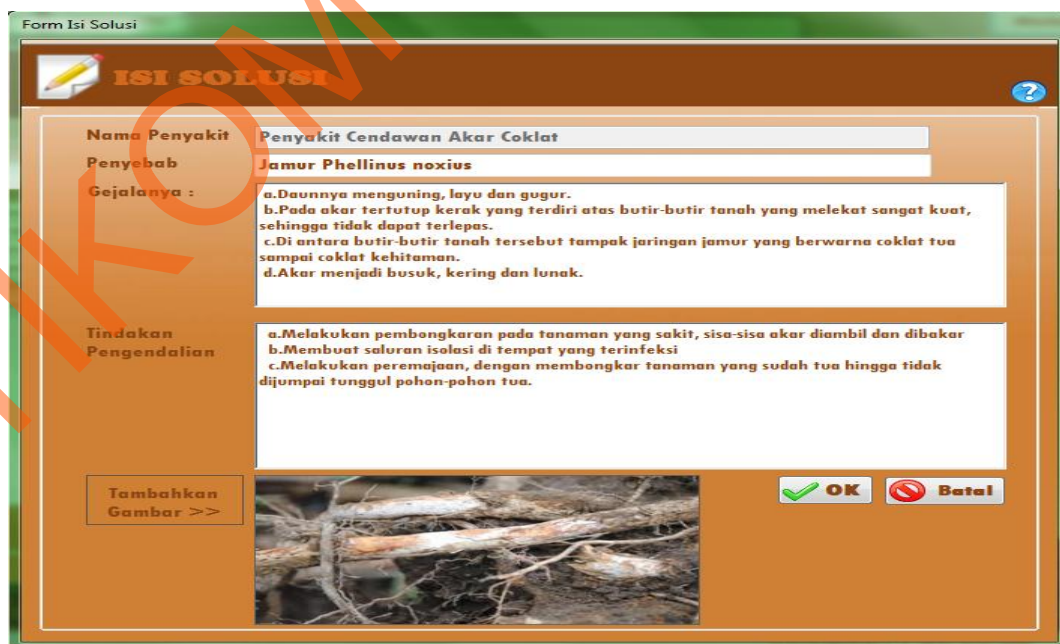
diedit” dan solusinya terisi sesuai dengan data yang dimasukkan (Gambar 4.63).

Pesan “Data solusi berhasil diedit” ini dapat dilihat pada Gambar 4.64.

Berdasarkan *test case id 3* pada Tabel 3.40, sistem akan menampilkan pesan *error* “Isian tidak boleh ada yang kosong” karena pengguna menekan tombol OK ketika ada *fields* yang kosong (tidak diisi). Contoh pesan *error* ini dapat dilihat pada Gambar 4.65.



Gambar 4.61 Pilih Nama Penyakit



Gambar 4.62 Isi Solusi Dari Nama Penyakit “Cendawan Akar Coklat”

Form Solusi

SOLUSI

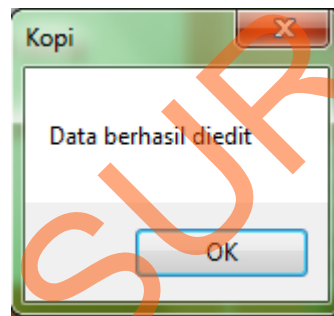
Solusi

Nama Penyakit	Penyebab	Gejala Penyakit
Penyakit Cendawan Akar Coklat	Jamur <i>Phellinus noxius</i>	a. Daunnya menguning, layu dan gugur. b. Pada akar tertutup kerak yang terdiri atas butir-butir tanah yang melekat.
Penyakit Akar Putih	Cendawan <i>Rigidoporus microporus</i>	a. Daun menguning kemudian gugur. b. Munculnya bunga dan buah meskipun umurnya belum cukup. c. Akar membusuk.
Penyakit Busuk Akar	Cendawan <i>Armillaria tabescens</i>	a. Daun menguning dan layu. b. Muncul retakan pada batang yang menandakan kekurangan air. c. Akar tampak membusuk.
Penyakit Jamur Upas	Jamur <i>Upasia salmonicolor</i>	a. Muncul bercak putih pada batang yang tidak terkena sinar matahari. b. Pada tahap awal atau tahap sarang laba-laba.
Penyakit Karat Daun	Jamur <i>Hemileia vastatrix</i>	a. Pada sisi bawah daun terdapat bercak-bercak berwarna kuning muda kemudian berubah menjadi kuning tua. b. Pada sisi atas daun terdapat bercak-bercak berwarna coklat.
Penyakit Bercak Daun	Jamur <i>Cercospora coffeicola</i>	a. Pada daun yang sakit timbul bercak berwarna kuning. b. Pada buahnya timbul bercak berwarna coklat, biasanya di bagian bawah.
Penyakit Bercak Daun Alga	Alga <i>Cephaleuros virescens</i>	a. Pada permukaan atas daun muncul bintik merah kecoklatan berbentuk cakram yang lama-kelamaan melebar dan menggugurkan daun.

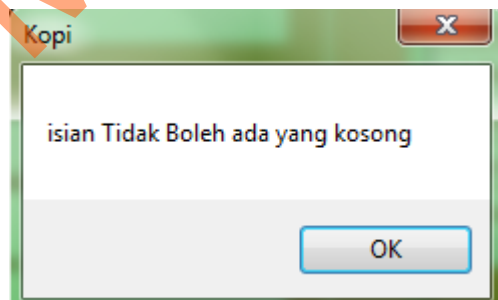
** Double klik pada tabel untuk memilih data

Keluar

Gambar 4.63 Hasil Isi Solusi Nama Penyakit “Cendawan Akar Coklat”



Gambar 4.64 Pesan Sukses Proses Edit Data Solusi



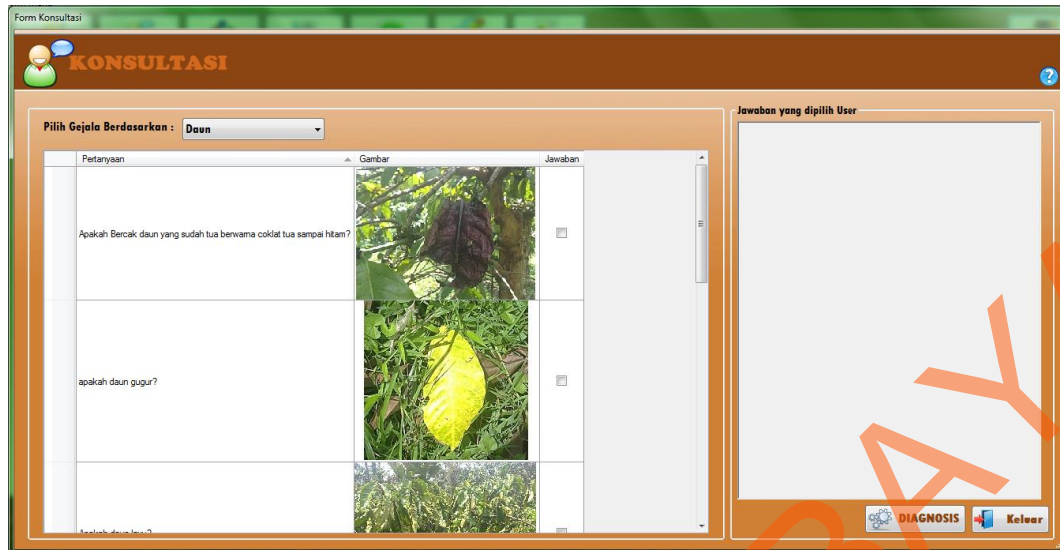
Gambar 4.65 Pesan *Error* Proses Edit Data Solusi

Tabel 4.7 Hasil Uji Coba *Form* Isi Solusi

No	Tujuan	Input	Output yang diharapkan	Output Sistem
1	Mengisi data Solusi	Memilih nama penyakit “Cendawan Akar Coklat “ pada <i>form Solusi</i> , isi data pada Tabel 3.39 kemudian tekan tombol OK	Solusi terisi sesuai dengan data yang dimasukkan	Aplikasi sesuai. (Gambar 4.63)
2	Menyimpan hasil pengisian data solusi	Memilih nama penyakit pada <i>form Solusi</i> , isi data pada Tabel 3.39 pada <i>form</i> Isi Solusi, tekan tombol OK	Muncul Pesan “Data berhasil diedit”	Aplikasi sesuai. Muncul pesan sukses (Gambar 4.64)
3	Menghindari <i>field</i> yang kosong ketika menekan tombol OK	Menekan tombol OK tanpa mengisi <i>field</i> yang ada	Muncul pesan <i>Error</i> “Isian tidak boleh ada yang kosong”	Aplikasi sesuai. Muncul pesan <i>error</i> (Gambar 4.65)

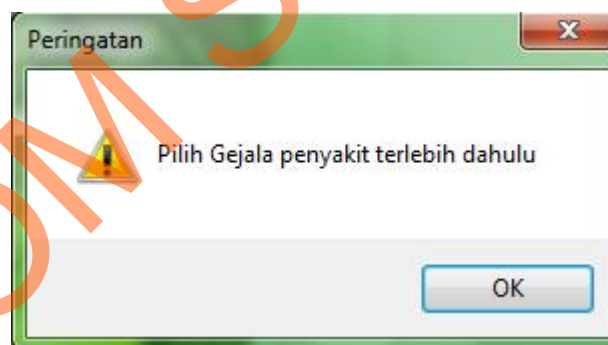
H Uji Coba Form Konsultasi

Uji coba *form* ini bertujuan untuk memeriksa kemampuan sistem dalam melakukan proses diagnosis penyakit pada tanaman kopi. Agar dapat melakukan diagnosis penyakit, pengguna harus memilih gejala-gejala penyakit yang ada kemudian menekan tombol Diagnosis. Berdasarkan *test case id* 2 pada Tabel 3.41, sistem akan menyaring (*filter*) daftar gejala penyakit sesuai pilihan pada *combobox* yang ada, misalnya Daun, maka sistem akan menampilkan daftar pertanyaan yang berhubungan dengan daun (lihat Gambar 4.66).



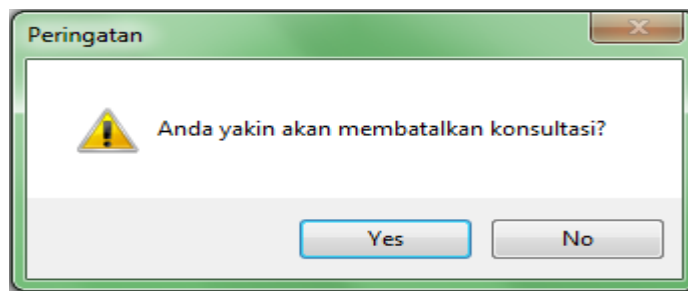
Gambar 4.66 Tampilan Proses Menyaring Pertanyaan

Berdasarkan *test case id 1* pada tabel 3.41, sistem akan menampilkan pesan *error* karena pengguna menekan tombol Diagnosis tanpa memilih gejala penyakit. Pesan *error* ini dapat dilihat pada Gambar 4.67.



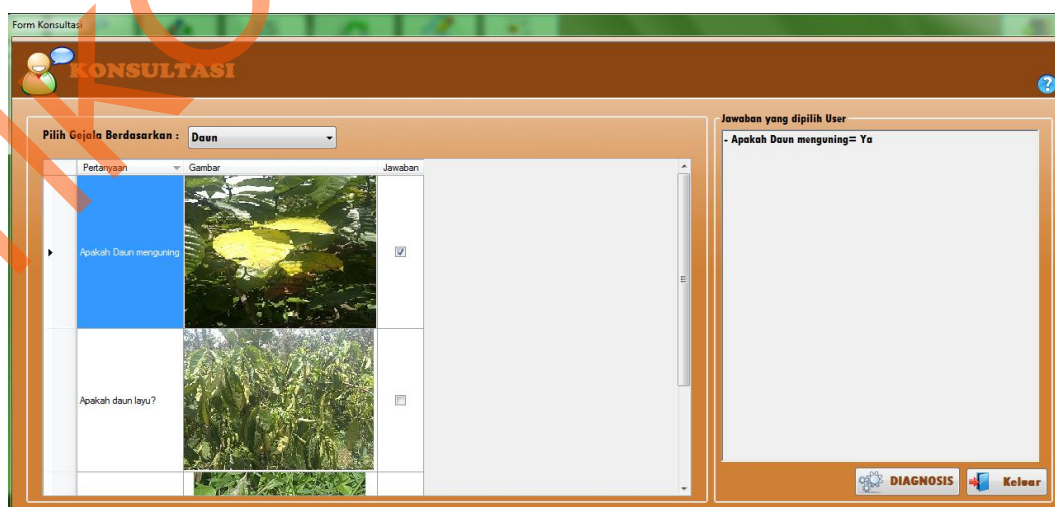
Gambar 4.67 Pesan *Error* Proses Diagnosis

Berdasarkan *test case id 3* pada tabel 3.41, sistem akan menampilkan pesan peringatan ketika pengguna menekan tombol Keluar. Pesan peringatan ini dapat dilihat pada Gambar 4.68. Jika pengguna menjawab *Yes*, maka sistem akan membatalkan proses konsultasi dan keluar dari *form* ini, jika *No*, maka pengguna dapat melanjutkan konsultasinya.



Gambar 4.68 Pesan Peringatan Ketika Menekan Tombol Keluar

Berdasarkan *test case id 4* pada Tabel 3.41 dilakukan proses diagnosis penyakit pada tanaman kopi. Langkah pertama adalah pengguna harus menjawab pertanyaan pada tanaman kopi. Langkah pertama adalah pengguna harus menjawab pertanyaan yang diberikan dengan memilih pada *checkbox* yang tersedia. Misalnya pengguna memilih pertanyaan “Apakah Daun Menguning?”, kemudian menekan tombol Diagnosis, maka sistem akan menampilkan *form* Hasil Konsultasi untuk melihat hasil diagnosis penyakit sesuai dengan gejala yang telah dipilih. Gambar 4.69 adalah proses dimana pengguna memilih pertanyaan Apakah daun Menguning?, sedangkan Gambar 4.70 adalah hasil diagnosisnya. Hasil diagnosis pada Gambar 4.70 memiliki tiga nama penyakit, karena pada ketiga penyakit tersebut terdapat gejala “Daun Menguning”.



Gambar 4.69 Proses Menjawab Pertanyaan

Gambar 4.70 Hasil Diagnosis Penyakit Pada Tanaman Kopi

Berdasarkan *test case id 5* pada Tabel 3.41, sistem menampilkan *form* Detil Penyakit untuk melihat detil dari penyakit yang telah didiagnosis dan untuk melihat *form* Detil Penyakit, pengguna harus memilih *linklabel* klik untuk melihat detil penyakit (Lihat Gambar 4.70). Tampilan *form* Detil penyakit ini dapat dilihat pada Gambar 4.71.

Gambar 4.71 Tampilan Detil Penyakit

Tombol OK yang terlihat pada Gambar 4.71 digunakan untuk menyimpan hasil konsultasi yang ada dan menampilkan *form* Laporan Hasil konsultasi yang telah dilakukan. Tampilan *form* hasil Konsultasi ini dapat dilihat pada Gambar 4.72.

The screenshot shows a software window titled 'Tampilkan Laporan Berdasarkan'. It contains a form for a consultation report. The form is titled 'LAPORAN HASIL KONSULTASI' and is generated by 'Dewi Angkie' on '09/05/2011'. The form includes the following information:

- Tanggal Konsultasi :** 09/05/2011
- Gejala yang dipilih :**
 - Akar tumbuh kerak yang terdiri atas butir-butir tanah = YA
 - Akar membusuk = YA
 - Akar mengering = YA
 - Tangkai Seringgus jamur berwarna coklat tua = YA
 - Akar menjadi lunak = YA
 - Daun menguning = YA
 - Daun layu = YA
 - daun gugur = YA
- Kesimpulannya**
 - Nama Penyakit : Penyakit Cendawan Akar Coklat
 - Penyebabnya : Jamur *Phellinus noxius*
- Gejalanya**
 - a. Daunnya menguning, layu dan gugur.
 - b. Pada akar tertutup kerak yang terdiri atas butir-butir tanah yang melekat sangat kuat, sehingga tidak dapat terlepas.
 - c. Di antara butir-butir tanah tersebut tampak janggut jamur yang berwarna coklat tua sampai coklat kehitaman.
 - d. Akar menjadi busuk kering dan lunak.
- Solusi Tindakan**
 - a. Dilakukan pembongkaran pada tanaman yang sakit, sisa-sisa akar diambil dan dibakar.
 - b. Membuat saluran isolasi di tempat yang terinfeksi.
 - c. Melakukan peremajaan, dengan membongkar tanaman yang sudah tua hingga tidak dipupai tanggal polen-pohon tua.

At the bottom of the window, it shows 'Current Page No.: 1', 'Total Page No.: 2', and 'Zoom Factor: 75%'.

Gambar 4.72 Tampilan *Form* Laporan Hasil Konsultasi

Tabel 4.8 Hasil Uji Coba *Form* Konsultasi

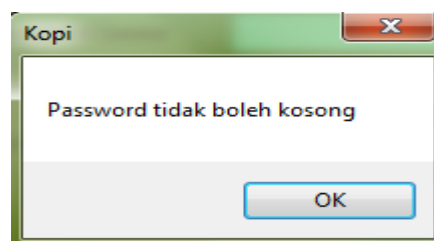
No	Tujuan	Input	Output yang diharapkan	Output Sistem
1	Menghindari proses diagnosis sebelum memilih gejala penyakit	Menekan tombol Diagnosis tanpa memilih gejala penyakit	Muncul Pesan <i>Error</i> "Pilih Gejala penyakit terlebih dahulu"	Aplikasi sesuai. Muncul pesan <i>error</i> (Gambar 4.67)
2	Menyaring daftar pertanyaan yang bersesuaian	Memilih pada <i>comboBox</i> yang tersedia	Daftar pertanyaan akan tersaring sesuai dengan pilihan.	Aplikasi sesuai. Gambar 4.66
3	Menampilkan pesan saat menekan tombol Keluar setelah memilih gejala penyakit	Memilih gejala-gejala penyakit pada <i>form</i> Konsultasi dan menekan tombol Keluar	Muncul Pesan <i>error</i> "Anda yakin akan membatalkan konsultasi?" jika menjawab <i>yes</i> maka keluar dari <i>form</i> , jika <i>cancel</i> maka melanjutkan proses konsultasi.	Aplikasi sesuai. Gambar 4.68

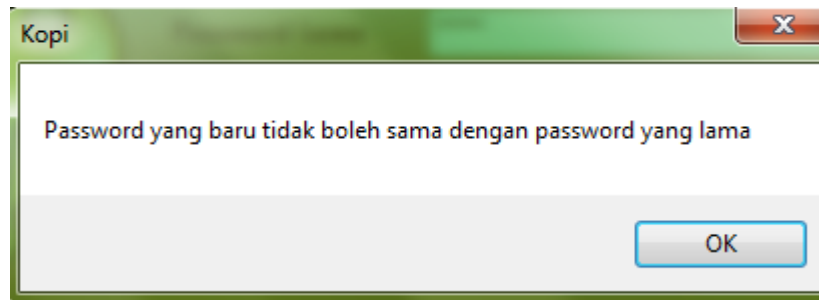
Tabel 4.8 Hasil Uji Coba *Form* Konsultasi (Lanjutan)

No	Tujuan	Input	Output yang diharapkan	Output Sistem
4	Melakukan proses Diagnosis penyakit	Memilih pertanyaan dan menekan tombol diagnosis	Menghasilkan diagnosis penyakit berupa nama penyakit sesuai dengan gejala-gejala yang dipilih.	Aplikasi sesuai. Gambar 4.69 dan Gambar 4.70
5	Menampilkan <i>Form</i> detil penyakit beserta dengan penjelasannya	menekan <i>link</i> Lihat Detil pada <i>form</i> Hasil Konsultasi	Form Detil Penyakit beserta dengan penjelasannya	Aplikasi sesuai. Gambar 4.71
6	Menampilkan laporan Hasil konsultasi ketika memilih OK	Memilih Tombol OK pda <i>form</i> detil Penyakit	Laporan yang ditampilkan adalah laporan dari hasil konsultasi yang baru saja dilakukan.	Aplikasi sesuai. Gambar 4.72

I Uji Coba Form Ubah Password

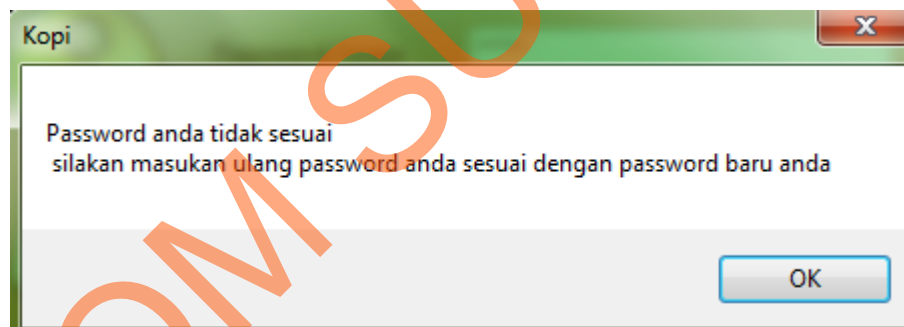
Form ini digunakan untuk mengubah *password* dari *user* umum yang ingin mengubah *password*. Uji coba ini dilakukan untuk mengecek setiap proses ubah *password* yang dilakukan oleh pengguna. Sesuai dengan *test case id* 1 pada Tabel 3.42, sistem akan menampilkan pesan *error* seperti yang terlihat pada Gambar 4.73, saat pengguna menekan tombol OK ketika ada *field* yang kosong. Pesan *error* yang terlihat pada Gambar 4.74 merupakan pesan peringatan kepada pengguna karena memasukan *password* baru yang sama persis dengan *password* yang lama.

Gambar 4.73 Pesan *Error* Proses Ubah *Password*

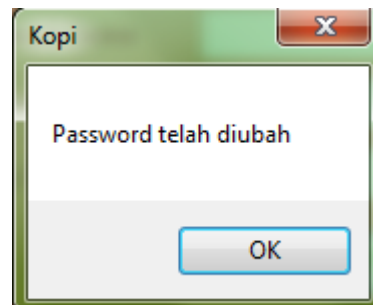


Gambar 4.74 Pesan *Error* Proses Ubah *Password*

Berdasarkan dengan *test case id 3* pada Tabel 3.42, sistem akan menampilkan pesan *error* karena data yang diisi pada *field* ulang password tidak sama dengan data pada *field* password baru. Pesan *error* ini dapat dilihat pada Gambar 4.75 dan pesan pada Gambar 4.76 adalah pesan sukses ketika proses ubah *password* dilakukan dengan benar.



Gambar 4.75 Pesan *Error* Proses Ubah *Password*



Gambar 4.76 Pesan Sukses Ubah *Password*

Tabel 4.9 Hasil Uji Coba *Form* Ubah *Password*

No	Tujuan	Input	Output yang diharapkan	Output Sistem
1	Menghindari <i>field</i> yang kosong ketika mengubah <i>password</i>	Menekan tombol OK tanpa mengisi <i>fields</i> yang ada	Muncul Pesan <i>Error</i> “Password tidak boleh kosong”	Aplikasi sesuai. Muncul pesan <i>error</i> . (Gambar 4.73)
2	Menghindari memasukan <i>password</i> baru sama dengan <i>password</i> yang lama	Mengisi <i>password</i> yang baru sama persis dengan <i>password</i> yang lama, kemudian tekan tombol OK	Muncul Pesan <i>Error</i> “password yang baru tidak boleh sama dengan password yang lama”	Aplikasi Sukses memunculkan pesan <i>error</i> . (Gambar 4.74)
3	Menghindari memasukan <i>password</i> yang beda pada <i>field</i> ulangi <i>password</i> dengan <i>password</i> pada <i>field</i> <i>password</i> baru	Mengisi data pada <i>field</i> Ulangi <i>Password</i> yang tidak sesuai dengan <i>Password</i> baru kemudian menekan tombol OK	Muncul Pesan <i>Error</i> “Password anda tidak sesuai, silakan masukan ulang password sesuai dengan password baru anda”	Aplikasi Sukses memunculkan pesan <i>error</i> . (Gambar 4.75)
4	Proses ubah <i>password</i> berhasil dilakukan	Mengisi <i>password</i> sesuai dengan <i>fields</i> yang ada, kemudian tekan tombol OK.	Muncul pesan “Password telah diubah”	Aplikasi Sukses memunculkan pesan sukses. (Gambar 4.76)

4.3.2 Uji Coba Kasus Jenis Penyakit Tanaman Kopi

Uji coba kasus pada jenis penyakit tanaman kopi ini dilakukan untuk menguji aplikasi sistem pakar ini dalam mendiagnosis penyakit yang ada dengan menggunakan metode *forward chaining* dan dapat menampilkan dengan benar hasil diagnosis nya serta dapat memberikan solusi untuk tindakan pengendaliannya. Uji coba dilakukan dengan menjawab setiap pertanyaan yang ditampilkan, dengan cara memilih pertanyaan yang sesuai dengan gejala penyakit yang dimaksud. Jenis penyakit tanaman kopi yang diuji coba pada bab ini adalah penyakit Cendawan Akar Coklat, penyakit Akar Putih dan penyakit Busuk Akar.

A Uji Coba Kasus untuk Penyakit Cendawan Akar Coklat

Pada uji coba berikut ini dilakukan pengujian sistem dalam mendiagnosis penyakit tanaman kopi untuk penyakit “Cendawan Akar Coklat” sesuai dengan data uji coba pada Tabel 3.43. Uji coba dilakukan dengan memilih pertanyaan pada *form* Konsultasi (lihat Gambar 4.77) sesuai dengan data pada Tabel 3.43 dan hasil uji coba diagnosisnya dapat dilihat pada Gambar 4.78 dan detail penyakitnya dapat dilihat pada Gambar 4.79.

Form Konsultasi

KONSULTASI

Pilih Gejala Berdasarkan : Daun

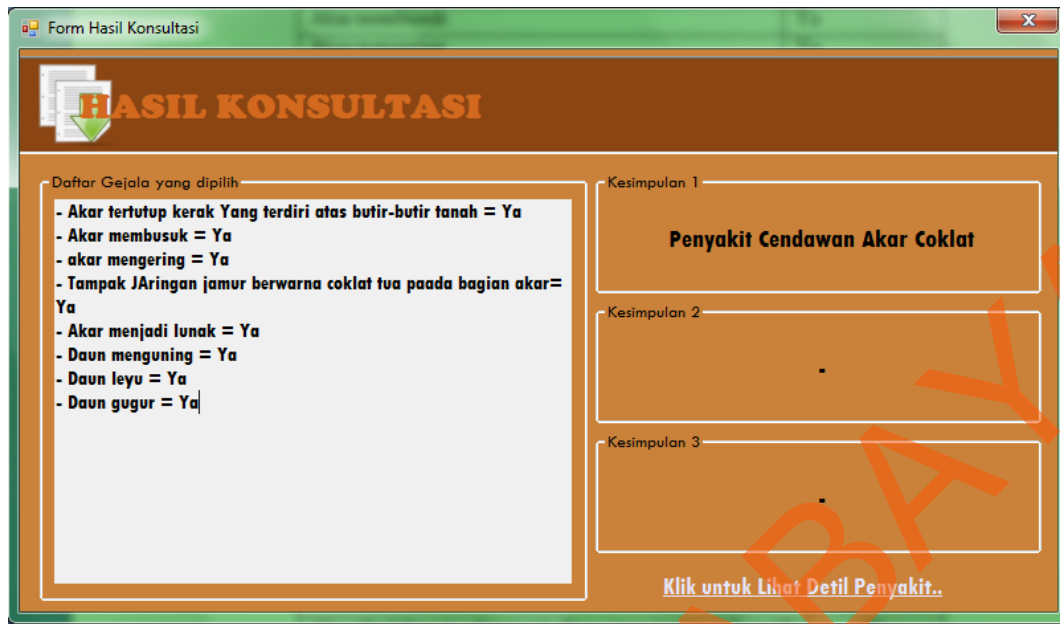
Pertanyaan	Gambar	Jawaban
Apakah Daun menguning?		<input checked="" type="checkbox"/>
Apakah daun layu?		<input checked="" type="checkbox"/>
apakah daun gugur?		<input checked="" type="checkbox"/>

Jawaban yang dipilih User

- Apakah Akar tertutup kerak yang terdiri atas butir-butir tanah = Ya
- Apakah akar membusuk? = Ya
- apakah akar mengering = Ya
- Apakah Tampak Jaringan Jamur berwarna coklat tua pada bagian akar? = Ya
- Apakah Akar menjadi lunak? = Ya
- Apakah Daun menguning = Ya
- Apakah daun layu? = Ya
- apakah daun gugur? = Ya

DIAGNOSIS Keluar

Gambar 4.77 Form Konsultasi Uji Coba Kasus Untuk Penyakit “Cendawan Akar Coklat”



Gambar 4.78 Hasil Uji Coba Diagnosis Penyakit “Cendawan Akar Coklat”



Gambar 4.79 Hasil Detil Penyakit “Cendawan Akar Coklat”

B Uji Coba Kasus untuk Penyakit Akar Putih

Uji coba berikut ini dilakukan pengujian terhadap sistem dalam mendiagnosis penyakit tanaman kopi untuk penyakit “Akar Putih” sesuai dengan data uji coba pada Tabel 3.44. Uji coba dilakukan dengan memilih pertanyaan pada

form Konsultasi (Lihat Gambar 4.80) sesuai dengan data pada Tabel 3.44 dan hasil uji coba diagnosisnya dapat dilihat pada Gambar 4.81 dan detil penyakitnya dapat dilihat pada Gambar 4.82.

Gambar 4.80 *Form* Konsultasi Uji Coba Kasus Untuk Penyakit “Akar Putih”

Gambar 4.81 Hasil Uji Coba Diagnosis Penyakit “Akar Putih”

Form Detil Penyakit

DETIL PENYAKIT

Kesimpulan 1

Penyakit Akar Putih

Kesimpulan 2

Nama Penyakit

Kesimpulan 3

Nama Penyakit

PENYEBABNYA : Cendawan Rigidoporus microporus

GEJALANYA :

a. Daun menguning kemudian gugur.

b. Munculnya bunga dan buah meskipun umumnya belum cukup.

c. Akar membusuk diselubungi selaput miselium jamur mirip jala putih.

d. Infeksi berat membuat pohon roboh karena akar membusuk.

SOLUSINYA :

a. Lakukan sanitasi lahan sampai benar-benar bersih dari serasah dan tonggak ataupun akar yang tersisa dalam tanah.

b. Tebang dan cabut akar pohon yang terserang.

c. Sebelum ditanami, lahan disanitasi dengan fungisida berbahan aktif triadimenfon seperti Bayleton 250 EC, Anvil 50 SC yang mengandung heksakonazol dan Calxon 750 EC dengan bahan aktif tidemorf.

d. Buat parit antar pohon untuk mengisolasi penyakit.

OK

Gambar 4.82 Hasil Detil Penyakit “Akar Putih ”

C Uji Coba Kasus untuk Penyakit Busuk Akar

Pada uji coba berikut ini dilakukan pengujian sistem dalam mendiagnosis penyakit tanaman kopi untuk penyakit “Busuk Akar” sesuai dengan data uji coba pada Tabel 3.45. Uji coba dilakukan dengan memilih pertanyaan pada *form* Konsultasi (Lihat Gambar 4.83) sesuai dengan data pada Tabel 3.45 dan hasil uji coba diagnosisnya dapat dilihat pada Gambar 4.84 dan detil penyakitnya dapat dilihat pada Gambar 4.85

Form Konsultasi

KONSULTASI

Pilih Gejala Berdasarkan : **Tanaman**

Pertanyaan	Gambar	Jawaban
Apakah tanaman menjadi gundul?		<input checked="" type="checkbox"/>

Jawaban yang dipilih User

- Apakah Muncul retakan pada batang? = Ya
- Apakah Akar mengeluarkan cairan kental berwarna coklat kekuningan? = Ya
- Apakah Daun menguning? = Ya
- Apakah daun layu? = Ya
- Apakah tanaman menjadi gundul? = Ya

DIAGNOSIS **Keluar**

Gambar 4.83 *Form* Konsultasi Uji Coba Kasus Untuk Penyakit “Busuk Akar”

Gambar 4.84 Hasil Uji Coba Diagnosis Penyakit “Busuk Akar”

Gambar 4.85 Hasil Detil Penyakit “Busuk Akar”

Berdasarkan hasil uji coba sistem dan kasus di atas, aplikasi ini juga diuji cobakan kepada ahli sistem, ahli isi dan pengguna umum. Selain melakukan uji coba, setiap pengguna juga diberi angket yang mengandung jawaban dari pertanyaan mengenai aplikasi, dan setiap pengguna dapat memberikan penilaian

terhadap aplikasi yang dibuat, sesuai dengan hasil uji coba yang dilakukan. Tujuan dari pengisian angket ini adalah untuk menguji kebenaran aplikasi dengan yang sebenarnya. Adapun pertanyaan dan hasil jawaban yang diberikan dapat dilihat pada lampiran yang akan disertakan pada laporan tugas akhir ini. Berikut ini adalah hasil pengolahan data angket yang telah dilakukan sesuai dengan bidangnya:

1. Angket Pengguna Umum

Angket ini digunakan untuk pengguna yang melakukan uji coba aplikasi ketika melakukan konsultasi. Total pengguna yang mengisi angket ini berjumlah 6 orang. Pengolahan data pada angket ini menggunakan bentuk skala *Likert* (Riduwan, 2005).

Tabel 4.10 Hasil Olah Data Angket Untuk Pengguna Umum

No	Pertanyaan	Penilaian					Jumlah
		5	4	3	2	1	
1	Bagaimana fitur-fitur yang disediakan oleh aplikasi?	10	12	3	0	0	25
2	Apakah tampilan dari menu konsultasi sudah jelas?	15	4	6	0	0	25
3	Apakah proses untuk menjawab pertanyaan yang ditampilkan sudah jelas?	0	24	0	0	0	24
4	Bagaimana dengan hasil diagnosis yang diberikan? Apakah sudah jelas?	0	24	0	0	0	24
5	Apakah informasi yang terdapat dalam form Hasil konsultasi sudah jelas?	5	8	9	0	0	22
6	Apakah informasi yang terdapat di dalam form detil penyakit sudah jelas ?	0	16	6	0	0	22
7	Bagaimana dengan laporan hasil konsultasinya? Apakah sudah jelas?	15	8	3	0	0	26
8	Bagaimana dengan tulisan yang ditampilkan?	5	16	3	0	0	24
9	menurut anda, bagaimana tampilan dari aplikasi ini secara keseluruhan?	0	16	6	0	0	22
Jumlah Skor Hasil Pengumpulan data						214	
Persentase Hasil Uji Coba						79.26 %	

Skor penilaian yang diperoleh dari masing-masing pertanyaan didapat dari:

(jumlah pengguna) x (skala nilai).

Misalnya : untuk pertanyaan no.1

Jumlah pengguna yang menjawab skala nilai 5 adalah 2 orang, maka

skor penilaiannya = $2 \times 5 = 10$

Jumlah Skor tertinggi = (skala tertinggi = 5) x (jumlah pertanyaan= 9) x (total pengguna=6) adalah 270

Jika jumlah skor hasil pengumpulan data = 214

Maka untuk mengetahui persentasi hasil uji coba didapat dari:

(Jumlah skor hasil pengumpulan data / jumlah skor tertinggi) x 100%

Persentasi hasil uji coba = $(214 / 270) \times 100\% = 79.26 \%$

2. Angket Pakar

Angket ini digunakan untuk pakar yang melakukan uji coba aplikasi ketika melakukan maintenance parameter, solusi dan proses generate *rule*. Pakar yang mengisi angket ini berjumlah 2 orang, hasil jawaban yang diberikan dapat dilihat pada lampiran pada laporan tugas akhir ini. Pengolahan data pada angket ini juga menggunakan bentuk skala pengukuran *rating scale*.

Jumlah Skor tertinggi = (skala tertinggi = 5) x (jumlah pertanyaan= 13) x (total pengguna=2) adalah 130

Jika jumlah skor hasil pengumpulan data = 100

Maka untuk mengetahui persentasi hasil uji coba didapat dari:

(Jumlah skor hasil pengumpulan data / jumlah skor tertinggi) x 100%

Persentasi hasil uji coba = $(100 / 130) \times 100\% = 76.92 \%$

Tabel 4.11 Hasil Olah Data Angket Untuk Pakar

No	Pertanyaan	Penilaian					Jumlah
		5	4	3	2	1	
1	Bagaimana fitur-fitur yang disediakan oleh aplikasi?	0	8	0	0	0	8
2	Apakah menu yang disediakan untuk menambah, mengubah dan menghapus gejala penyakit sudah jelas?	0	4	3	0	0	7
3	Apakah tampilan menu yang disediakan di form rule sudah jelas?	0	4	0	2	0	6
4	Apakah langkah-langkah dalam membuat rule sudah jelas?	0	4	0	2	0	6
5	Bagaimana dengan tampilan menu dari form Solusi?	5	4	0	0	0	9
6	Apakah tampilan dari menu konsultasi sudah jelas?	5	0	3	0	0	8
7	Apakah proses untuk menjawab pertanyaan yang ditampilkan sudah jelas?	0	8	0	0	0	8
8	Bagaimana dengan hasil diagnosis yang diberikan? Apakah sudah jelas?	0	8	0	0	0	8
9	Apakah informasi yang terdapat dalam form Hasil konsultasi sudah jelas?	0	0	6	0	0	6
10	Apakah informasi yang terdapat di dalam form detail penyakit sudah jelas ?	10	0	0	0	0	10
11	Bagaimana dengan laporan hasil konsultasinya? Apakah sudah jelas?	5	4	0	0	0	9
12	Bagaimana dengan tulisan yang ditampilkan?	0	4	3	0	0	7
13	menurut anda, bagaimana tampilan dari aplikasi ini secara keseluruhan?	0	8	0	0	0	8
Jumlah Skor Hasil Pengumpulan data							100
Persentase hasil uji coba							76.92%

3. Angket Programmer

Angket ini digunakan oleh *programmer* untuk menguji fungsi dari setiap menu dan fitur-fitur yang disediakan oleh aplikasi apakah sudah berfungsi sebagaimana mestinya. *Programmer* yang mengisi angket ini berjumlah 5 orang, hasil jawaban yang diberikan dapat dilihat pada lapiran pada laporan tugas akhir ini. Pengolahan data pada angket ini menggunakan tipe pengukuran skala *guttman* (Riduwan, 2005)

Tabel 4.12 Hasil Olah Data Angket Untuk *Programmer*

No	Pertanyaan	Penilaian (%)	
		Ya	Tidak
1	Apakah menu yang terdapat di dalam setiap form berfungsi sebagaimana mestinya?	100	0
2	Apakah aplikasi dapat menambah pengguna baru?	100	0
3	Apakah aplikasi dapat mengubah data pengguna?	100	0
4	Apakah aplikasi dapat membuat dependency diagram secara lengkap?	100	0
5	Apakah aplikasi dapat melakukan perubahan data parameter yang terdapat pada form treeview?	100	0
6	Apakah aplikasi dapat menghapus parameter pada treeview yang tidak memiliki cabang dibawahnya dan tidak berada paling atas?	100	0
7	Apakah aplikasi dapat melakukan pengisian solusi pada kesimpulan akhir yang ada?	100	0
8	Apakah aplikasi dapat merubah solusi yang telah ada pada kesimpulan?	100	0
9	Apakah aplikasi dapat melakukan proses diagnosis?	100	0
10	Apakah aplikasi dapat menampilkan kesimpulan dari proses diagnosis?	100	0
11	Apakah aplikasi dapat mencetak laporan hasil diagnosis/konsultasi?	100	0
12	Apakah aplikasi dapat mencetak daftar rule?	100	0
13	Apakah aplikasi dapat mencetak data solusi?	100	0
14	Apakah aplikasi dapat mencetak data pengguna?	100	0
15	Apakah aplikasi dapat mencetak grafik penyakit?	100	0
16	Apakah secara umum kegunaan aplikasi ini sudah jelas?	100	0
17	Apakah desain aplikasi sudah baik?	100	0
18	Apakah tulisan yang ada pada aplikasi dapat dibaca dengan jelas?	80	20
Rata-rata persentase hasil uji coba		98.89	1.11

Berdasarkan hasil pengolahan data angket yang telah diberikan kepada masing-masing ahli, menghasilkan persentase nilai sebagai berikut:

- a. Angket yang diberikan kepada pengguna umum dengan 9 pertanyaan, menghasilkan persentase nilai hasil uji coba sebesar 79.26 % .
- b. Angket yang diberikan kepada pakar dengan 13 pertanyaan, menghasilkan persentase nilai keberhasilan penggunaan aplikasi sebesar 76.92 %.

- c. Angket yang diberikan kepada *programmer* dengan 18 pertanyaan, menghasilkan persentase nilai keberhasilan penggunaan aplikasi sebesar 98.89 %.

4.4 Evaluasi

Berdasarkan hasil uji coba sistem dan uji coba kasus yang dilakukan terhadap aplikasi sistem pakar ini, terdapat beberapa kelebihan dan kekurangan. Kelebihan dari aplikasi yang dibuat adalah:

1. Aplikasi mampu membuat tabel keputusan secara otomatis sesuai dengan kombinasi value yang ada.
2. Aplikasi mampu melakukan reduksi tabel keputusan dan melakukan *generate rule*.
3. Aplikasi mampu menyortir pertanyaan yang bersesuaian, sehingga dapat mencegah pengguna dalam memilih jawaban yang tidak sesuai (asal-asalan).
4. Dari hasil uji coba kasus, terlihat bahwa aplikasi mampu mendiagnosis penyakit dengan benar dan memberikan solusi untuk tindakan pengendaliannya.
5. Dari persentase nilai hasil uji coba aplikasi yang menggunakan angket, terlihat bahwa aplikasi sistem pakar ini dapat digunakan dan dapat memberikan informasi diagnosis penyakit pada tanaman kopi sesuai dengan yang diharapkan.

Kekurangan dari aplikasi sistem pakar ini adalah, proses untuk diagnosis penyakitnya hanya menggunakan metode *forward chaining*. Proses *generate rule* hanya menghasilkan *rule* dengan *assignment operator* sama dengan (=) dan *logical operator AND*.