

# SISTEM PAKAR BERBASIS ANDROID UNTUK MENGIDENTIFIKASI HAMA DAN PENYAKIT PADA BAWANG MERAH MENGGUNAKAN METODE FORWARD CHAINING (STUDI KASUS : PROGRAM STUDI SISTEM INFORMASI UPN “VETERAN” JAWA TIMUR)

Vangga Mahyudin Ardiansyah<sup>1)</sup>

Nur Cahyo Wibowo<sup>2)</sup>

Agung Brastama Putra<sup>3)</sup>

E-mail : <sup>1)</sup>[tugas.vangga@yahoo.co.id](mailto:tugas.vangga@yahoo.co.id), <sup>2)</sup>[bluejundi@yahoo.com](mailto:bluejundi@yahoo.com), <sup>3)</sup>[agungtama@aol.com](mailto:agungtama@aol.com)

<sup>1, 2, 3</sup> Program Studi Sistem Informasi, Fakultas Ilmu Komputer, UPN “Veteran” Jawa Timur

## Abstrak

Tanaman bawang merah tidak luput dari serangan hama dan penyakit bawang merah. Untuk itu petani harus memahami aspek hama dan penyakit bawang merah agar jerih payah mereka dari usaha tani bawang merah membawa hasil yang memuaskan. Dengan adanya kemajuan ilmu pengetahuan dan teknologi informasi saat ini, petani dapat mengetahui informasi mengenai hama dan penyakit bawang merah serta cara pengendaliannya dengan mudah. Penelitian ini menghasilkan aplikasi yang dapat mengetahui informasi tentang hama dan penyakit bawang merah dengan menggunakan metode forward chaining. Petani dapat menggunakan berbagai fitur dalam aplikasi bawang merah ini. Fitur yang pertama adalah identifikasi hama dan identifikasi penyakit. Dalam fitur ini petani akan diberikan sesi konsultasi berupa beberapa pertanyaan untuk gejala yang kemudian akan ditarik kesimpulan berupa hasil dan cara penanggulangan dari hama dan penyakit yang disimpulkan. Fitur kedua, petani dapat mengetahui cara pengendalian dari hama dan penyakit pada bawang merah.

**Kata Kunci :** Hama, Penyakit, Pakar, Forward Chaining, Bawang Merah.

## 1. PENDAHULUAN

Bawang merah yang nama latinnya *Allium scalonicus* sebenarnya bukanlah tanaman asli Indonesia, melainkan tanaman yang berasal dari Syiria, Timur Tengah. Jenis tanaman ini mampu hidup di daerah tropis maupun subtropis. Oleh karena itu, jenis tanaman ini pun sangat bagus tumbuh di Indonesia.

Sistem pakar (*expert system*) merupakan suatu program komputer cerdas yang menggunakan *knowledge* (pengetahuan) dan prosedur inferensi untuk menyelesaikan masalah yang cukup sulit sehingga membutuhkan seorang ahli untuk menyelesaikannya. Pakar nantinya akan memberikan data – data berupa fakta – fakta dan solusi. Proses yang digunakan dalam pengolahan data adalah metode inferensi forward chaining. *Forward Chaining* digunakan sebagai salah satu teknik inferensi dalam sistem pakar ini, dikarenakan data dan fakta dalam melakukan proses penelitian telah didapatkan dan dari data atau fakta tersebut dapat dibuat sebuah sistem yang akan memberikan sebuah konklusi atau solusi berdasarkan atas sekumpulan data dan fakta tersebut. Dengan menggunakan teknik inferensi ini pula peluang dalam mendapatkan suatu konklusi yang lebih spesifik dapat dengan mudah didapatkan.

Pada jaman sekarang perkembangan teknologi semakin berkembang. Salah satunya adalah handphone. Handphone selain digunakan untuk panggilan dan pesan singkat dalam perkembangannya media ini mampu dilengkapi dengan aplikasi tambahan yang berguna bagi pengguna. Salah satu aplikasinya adalah aplikasi untuk mengetahui jenis hama dan penyakit tanaman bawang merah. Aplikasi ini diharapkan dapat menyamai seorang pakar tanaman untuk mengidentifikasi hama dan penyakit pada

tanaman bawang merah sehingga dapat membantu petani mendapatkan informasi tentang jenis hama dan penyakit serta pengendalian yang diperlukan. Aplikasi ini berupa *android application* sehingga aplikasi ini dapat digunakan dimanapun dan kapanpun karena pada jaman sekarang kebanyakan orang menggunakan Android sebagai sistem operasi telepon selulernya.

## 2. LANDASAN TEORI

### 2.1 Bawang Merah

Bawang merah merupakan salah satu komoditas sayuran unggulan yang sejak lama telah diusahakan petani secara intensif. Komoditas sayuran ini termasuk ke dalam kelompok rempah tidak bersubstitusi yang berfungsi sebagai bumbu penyedap makanan serta bahan obat tradisional. Komoditas ini juga merupakan sumber pendapatan dan kesempatan kerja yang memberikan kontribusi cukup tinggi terhadap perkembangan ekonomi wilayah. Karena memiliki nilai ekonomi yang tinggi, maka pengusahaan budidaya bawang merah telah menyebar di hampir semua provinsi di Indonesia. Meskipun minat petani terhadap bawang merah cukup kuat, namun dalam proses pengusahaannya masih di temui berbagai kendala, baik kendala yang bersifat teknis maupun ekonomis (Sumarni dan Achmad, 2005).

#### 2.1.1 Klasifikasi Bawang Merah

Dalam klasifikasinya bawang merah termasuk famili *Amaryllidaceae*, *ordo Liliales*. Merupakan tanaman semusim, berumbi lapis (bulbus), berakar serabut dan berdaun dengan bentuk silindris. Pangkal daun bersatu membentuk batang semu yang berubah bentuk dan fungsinya, membengkak membentuk umbi lapis. (Samsudin, 1979).

- Sisik daun : Merupakan bagian umbi yang berisi makanan cadangan, yang diperlukan untuk persediaan makanan sejak mulai bertunas sampai keluar akar. Dari ketiak sisik ini dapat tumbuh umbi – umbi baru yang disebut bulbulus atau suing.
- Kuncup : Bagian umbi yang akan menghasilkan tunas baru, yang kemudian membentuk umbi – umbi baru.
- Subang : Merupakan batang yang mengkerut, tempat duduknya sisik daun.
- Akar advent : Yaitu akar serabut berupa benang – benang (*radix fibrosa*) yang terdapat di bawah subang.

### 2.2 Hama Dan Penyakit Bawang Merah

Tanaman budi daya sering mengalami gangguan dari hama dan penyakit. Demikian pula tanaman bawang merah tidak luput dari gangguan tersebut. Gangguan hama dan penyakit ini dapat mengakibatkan menurunnya produksi tanaman. Pada serangan yang hebat dapat mengakibatkan gagalnya panen.

### 2.3 Pengendalian Hama Dan Penyakit Bawang Merah

Hama dan penyakit merupakan salah satu masalah yang cukup mengganggu dalam pelestarian dan perawatan tanaman, khususnya bawang merah. Oleh karena itu, pengendalian hama dan penyakit harus dilakukan secara intensif sejak dini, tentunya tanpa menimbulkan dampak negative bagi lingkungan.

## 2.4 Sistem Pakar

Ada beberapa definisi tentang sistem pakar, antara lain seperti berikut:

1. Menurut Durkin: sistem pakar adalah suatu program komputer yang dirancang untuk memodelkan kemampuan penyelesaian masalah yang dilakukan seorang pakar.
2. Menurut Ignizio: sistem pakar adalah suatu model dan prosedur yang berkaitan, dalam suatu domain tertentu, yang mana tingkat keahliannya dapat dibandingkan dengan keahlian seorang pakar.
3. Menurut Giarratano dan Riley: sistem pakar adalah suatu sistem komputer yang bisa menyaingi atau meniru kemampuan seorang pakar.

Sistem pakar (SP) atau *expert system* (ES) adalah sistem informasi yang berisi dengan pengetahuan dari pakar sehingga dapat digunakan untuk konsultasi. Sistem pakar ini dapat berisi dengan pengetahuan (*knowledge*) dari satu atau lebih pakar. Pengetahuan dari pakar didalam sistem ini digunakan sebagai dasar oleh sistem pakar untuk menjawab pertanyaan (konsultasi).

### 2.4.1 Arsitektur Sistem Pakar

Pada dasarnya sistem pakar terdiri dari empat komponen utama, yaitu : *Knowledge Acquisition Interface* (antar muka akuisisi pengetahuan), *User Interface*, *Knowledge Base*, *Inference Engine*. Adapun penjelasan dari masing-masing komponen tersebut adalah sebagai berikut.

1. Knowledge Acquisition Interface (antar muka akuisisi pengetahuan)

Knowledge acquisition Interface mengontrol bagaimana seorang pakar dan rekayasa pengetahuan (knowledge engineer) berinteraksi dengan program untuk memasukkan pengetahuan kedalam basis pengetahuan (knowledge base). Termasuk hal-hal yang dimaksudkan untuk membantu para pakar dalam mengekspresikan pengetahuannya dalam bentuk yang sesuai yang bisa diterima oleh komputer.

Proses ini mengekspresikan pengetahuan dalam basis pengetahuan yang biasa disebut dengan akuisisi pengetahuan (knowledge acquisition). Akuisisi pengetahuan akan menjadi sangat sulit, karena aspek dari pengembangan sistem pakar ini membutuhkan waktu yang cukup banyak dan usaha yang keras.

2. User Interface

User Interface adalah bagian dari program yang berinteraksi dengan pengguna. Prompt bagi pengguna digunakan untuk informasi yang diperlukan untuk memecahkan masalah, menampilkan kesimpulan dan menjelaskan alasannya. Fasilitas dari user interface kerap kali memasukkan :

- a. Tidak menanyakan pertanyaan yang bersifat basa-basi
- b. Menjelaskan alasan atas pertanyaan yang diajukan
- c. Menyediakan dokumentasi dari referensi
- d. Mendefinisikan istilah-istilah yang bersifat teknik
- e. Laporan rekomendasi yang cukup detail
- f. Pembeneran rekomendasi
- g. Bantuan online (online help)
- h. Penyajian informasi berbentuk gambar

3. Knowledge Base

Knowledge base terdiri dari detail pengetahuan tentang domain tertentu. Basis pengetahuan sangat beda dengan basis data yang mana basis pengetahuannya terdiri dari pengetahuan eksplisit (explicit knowledge) dan pengetahuan implisit (implicit knowledge). Banyak pengetahuan dalam basis pengetahuan tidak dalam kondisi eksplisit, tetapi diproses dengan mesin inferensi (inference engine) dari pernyataan eksplisit kedalam basis pengetahuan. Hal ini membuat basis pengetahuan lebih efisien dalam penyimpanan data bila dibandingkan dengan basis data.

Basis pengetahuan dapat terdiri dari beberapa tipe pengetahuan dan proses mendapatkan pengetahuan untuk basis pengetahuan kerap kali berbeda antara yang satu dengan yang lainnya. Hal ini tergantung dari tipe pengetahuan.

4. Inference Engine

*Inference engine* secara umum memproses *rule* untuk memberikan alasan dan menggambarkan kesimpulan-kesimpulan yang tidak dapat dilihat tetapi dapat diproses dari basis pengetahuan.

Mesin inferensi dapat dikatakan sama dengan query dalam sistem basis data. Tugas dari modul inferensi adalah mengeksplorasi basis pengetahuan untuk mencari solusi dari permasalahan tertentu.

*Inference engine* mampu memberikan alasan simbolik, bukan hanya alasan yang bersifat matematik. Bentuk *inference engine* diperkenankan untuk dikembangkan dalam berbagai variasi, tergantung dari berbagai faktor, termasuk bagaimana cara merepresentasikan pengetahuan

Terdapat 2 metode dalam *inference engine*, yaitu metode forward chaining dan metode *backward chaining*.

a. Forward chaining

Forward chaining adalah pendekatan data-driven yang dimulai dari informasi yang tersedia atau ide dasar, kemudian mencoba menarik kesimpulan.

Data aturan kesimpulan

A = 1 IF A = 1 AND B = 2

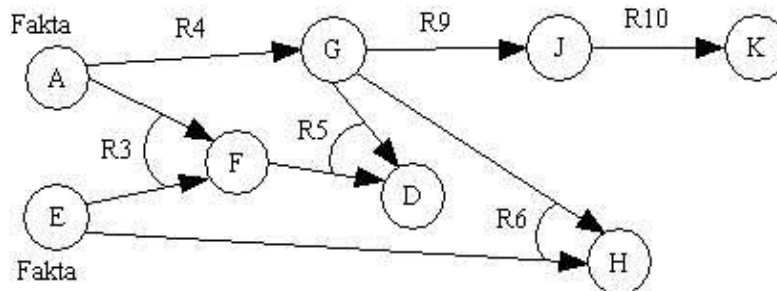
B = 2 THEN C = 3 C = 3

Contoh

IF Akar tanaman rusak.

AND Terdapat telur – telur ulat pada rerumputan.

THEN Terserang hama ulat grayak.



Gambar 1. Cara Kerja Mesin Inferensi Forward chaining

b. Backward chaining

Backward chaining adalah pendekatan *goal-driven* yang dimulai dari harapan apa yang akan terjadi (hipotesis) dan kemudian mencari bukti yang mendukung (atau berlawanan) dengan harapan. Sering, hal ini menentukan perumusan dan pengujian hipotesis sementara (subhipotesis).

Subtujuan aturan tujuan

A = 1 IF A = 1 AND B = 2

B = 2 THEN C = 3 C = 3

Misal : A dan B adalah gejala dan C adalah hama/penyakit

1 = Akar tanaman rusak

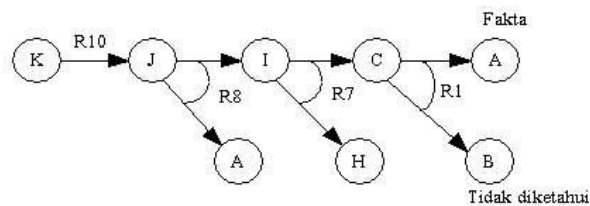
2 = Terdapat telur – telur ulat pada rerumputan

3 = Hama ulat grayak

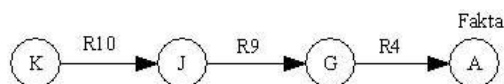
Contoh

Padi terserang hama ulat grayak.

IF Akar tanaman rusak AND Terdapat telur – telur ulat pada rerumputan.



Contoh Backward Chaining gagal



Contoh Backward Chaining berhasil

**Gambar 2.** Cara Kerja Mesin Inferensi Backward Chaining

### 2.5 Definisi Mobile Technology

Perangkat *mobile* memiliki banyak jenis dalam hal ukuran, desain dan *layout*, tetapi mereka memiliki kesamaan karakteristik yang sangat berbeda dari sistem *desktop*, diantaranya :

a. Ukuran yang kecil

Perangkat *mobile* memiliki ukuran yang kecil. Konsumen menginginkan perangkat yang terkecil untuk kenyamanan dan mobilitas mereka.

b. Memory yang terbatas

Perangkat *mobile* juga memiliki *memory* yang kecil, yaitu *primary* (RAM) dan *secondary* (disk). Pembatasan ini adalah salah satu faktor yang mempengaruhi penulisan program untuk berbagai jenis dari perangkat ini.

c. Daya proses yang terbatas

Sistem *mobile* tidaklah setangguh *desktop*. Ukuran, teknologi dan biaya adalah beberapa faktor yang mempengaruhi status dari sumber daya ini. Seperti *harddisk* dan RAM, pengguna dapat menggunakannya dalam ukuran yang pas dengan sebuah kemasan kecil.

d. Mengonsumsi daya yang rendah

Perangkat *mobile* menghabiskan sedikit daya dibandingkan dengan mesin *desktop*. Perangkat ini harus menghemat daya karena mereka berjalan pada keadaan dimana daya yang disediakan dibatasi oleh baterai-baterai.

e. Kuat dan dapat diandalkan

Karena perangkat *mobile* selalu dibawa kemana saja, mereka harus cukup kuat untuk menghadapi benturan-benturan, gerakan, dan sesekali tetesan-tetesan air. Akhir – akhir ini sudah banyak perangkat *mobile* yang sudah tahan banting, kebanyakan dari perangkat *mobile* yang tahan banting ini berasal dari China negara dengan populasi manusia terbesar didunia.

f. Konektivitas yang terbatas

Perangkat *mobile* memiliki *bandwith* rendah, beberapa dari mereka bahkan tidak tersambung. Kebanyakan dari mereka menggunakan koneksi *wireless*.

g. Masa hidup yang pendek

Perangkat-perangkat konsumen ini menyala dalam hitungan detik kebanyakan dari mereka selalu menyala. Coba ambil kasus sebuah *handphone*, mereka *booting* dalam hitungan detik dan kebanyakan orang tidak mematikan *handphone* mereka bahkan ketika malam hari.

### 2.6 Android

Pada tahun 2005 Google mengakuisisi *Android Inc* yang pada saat itu dimotori oleh Andy Rubin, Rich Miner, Nick Sears, dan Chris White. Yang kemudian pada tahun itu juga memulai membangun *platform* Android secara intensif.

Kemudian pada tanggal 12 November 2007 Google bersama *Open Handset Alliance (OHA)* yaitu konsorsium perangkat mobile terbuka, merilis Google Android SDK, setelah mengumumkannya seminggu sebelumnya. Dan sambutannya sangat luar biasa, hampir semua media berita tentang IT dan *Programming* membritakan tentang dirilisnya Android SDK (*Software Development Kit*).



Gambar 3. Android Timeline

### 2.6.1 Android Studio

*Android Studio* adalah sebuah IDE yang bisa digunakan untuk pengembangan aplikasi *Android*, dan dikembangkan oleh *Google*. *Android Studio* merupakan pengembangan dari *Eclipse* IDE, dan dibuat berdasarkan IDE *Java* populer, yaitu *IntelliJ IDEA*. *Android Studio* direncanakan untuk menggantikan *Eclipse* ke depannya sebagai IDE resmi untuk pengembangan aplikasi *Android*.

Sebagai pengembangan dari *Eclipse*, *Android Studio* mempunyai banyak fitur-fitur baru dibandingkan dengan *Eclipse* IDE. Berbeda dengan *Eclipse* yang menggunakan *Ant*, *Android Studio* menggunakan *Gradle* sebagai *build environment*. Fitur-fitur lainnya adalah sebagai berikut :

- Menggunakan *Gradle-based build system* yang fleksibel.
- Bisa mem-build multiple APK .
- Template support untuk *Google Services* dan berbagai macam tipe perangkat.
- Layout editor yang lebih bagus.
- Built-in support untuk *Google Cloud Platform*, sehingga mudah untuk integrasi dengan *Google Cloud Messaging* dan *App Engine*.
- Import library langsung dari *Maven repository*.

### 2.7 Unified Modelling Language (Uml)

Dalam suatu proses pengembangan *software*, analisa dan rancangan telah merupakan terminologi yang sangat tua. Pada saat masalah ditelusuri dan spesifikasi dinegosiasikan, dapat dikatakan bahwa kita berada pada tahap rancangan. merancang adalah menemukan suatu cara untuk menyelesaikan masalah, salah satu tool/model untuk merancang pengembangan *software* yang berbasis *object-oriented* adalah UML. Alasan mengapa UML digunakan adalah, pertama, *scalability* dimana objek lebih mudah dipakai untuk menggambarkan sistem yang besar dan komplek. Kedua, *dynamic modeling*, dapat dipakai untuk pemodelan sistem dinamis dan *real time*. Sebagaimana dalam tulisan pertama, menjelaskan konsep mengenai obyek, OOA&D (*Obyek Oriented Analyst/Design*) dan pengenalan UML, maka dalam tulisan kedua ini lebih ditekankan pada cara bagaimana UML digunakan dalam merancang sebuah pengembangan *software* yang disertai gambar atau contoh dari sebuah aplikasi.

## 3 ANALISIS DAN PERANCANGAN

### 3.1 Spesifikasi Sistem

Perancangan program ini dimaksudkan untuk mendapatkan suatu hasil, yaitu berupa aplikasi identifikasi hama dan penyakit pada bawang merah dengan metode *forward chaining*.

Sistem pakar untuk mengidentifikasi hama dan penyakit pada bawang merah untuk memperoleh fakta sebenarnya mengenai gejala-gejala hama dan penyakit pada bawang merah, terdapat sesi konsultasi yang merupakan sesi tanya jawab antara pengguna atau petani dengan sistem pakar. Dari gejala - gejala tersebut dapat ditentukan jenis hama dan penyakit dan serta mengendalikan atau penanggulangan masalah hama dan penyakit tersebut.

### 3.2 Perancangan Alur Aplikasi

Aplikasi sistem pakar identifikasi hama dan penyakit bawang merah ini terdapat beberapa interface, antara lain:

#### 1. Halaman awal aplikasi

Yaitu tampilan awal saat user mengakses program aplikasi ini. Pada tampilan ini terdapat beberapa menu awal yaitu : menu identifikasi hama, identifikasi penyakit, penanggulangan dan login admin.

#### 2. Menu identifikasi penyakit

Jika user mengakses menu ini, user akan diberikan pertanyaan – pertanyaan tentang gejala penyakit dan user diminta untuk memberi jawaban ya atau tidak dengan mengklik tombol (*RadioButton*) kemudian pilih *submit*. Setelah itu sistem akan mengajukan pertanyaan lagi dan begitu seterusnya sesuai dengan alur rule atau aturan dalam proses identifikasi ini. Kemudian akan keluar hasil serta penanggulangan.

#### 3. Menu identifikasi Hama

Hampir sama dengan identifikasi penyakit. Jika user mengakses menu ini, user akan diberikan pertanyaan – pertanyaan tentang gejala hama dan user diminta untuk memberi jawaban ya atau tidak dengan mengklik tombol (*RadioButton*) kemudian pilih *submit*. Setelah itu sistem akan mengajukan pertanyaan lagi dan begitu seterusnya sesuai dengan alur rule atau aturan dalam proses identifikasi ini. Kemudian akan keluar hasil serta penanggulangan.

#### 4. Cara penanggulangan

Berisikan tentang cara penanggulangan dari semua hama dan penyakit dari bawang merah

#### 5. Login Admin

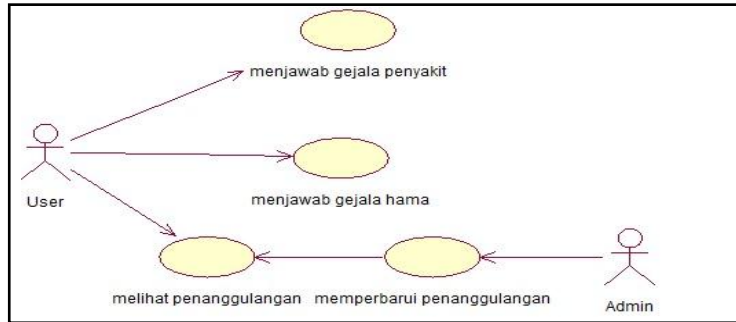
Hanya *admin* yang dapat mengakses menu ini. Setelah login admin dapat mengelola keterangan dari penanggulangan pada aplikasi ini.

### 3.3 Analisis Sistem

Dalam perencanaan sistem terdapat *use case diagram*, *activity diagram*, *class diagram* dan *sequence diagram* untuk memperjelas jalannya program.

#### 3.3.1 Use Case Diagram Identifikasi Hama Dan Penyakit Bawang Merah

*Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case diagram* dapat menggambarkan fungsi umum dari sebuah program beserta dengan keterlibatan actor yang terkait dalam sub – sub sistem untuk pengaksesan program secara maksimal, use juga dapat menggambarkan kerja seorang Admin secara menyeluruh pada pengaturan sistem. Berikut ini use case diagram dari sistem yang akan dibuat:



**Gambar 4.** Use Case Diagram Identifikasi Hama dan Penyakit

Pada diagram di atas terdiri dari 2 aktor dan 4 use case. Alur ini dimulai dari ketika user membuka aplikasi Sistem Pakar Untuk mengidentifikasi Hama dan Penyakit Bawang Merah. Di dalam menu identifikasi hama atau penyakit user di beri pertanyaan gejala hama atau penyakit., Dan di dalam menu login admin berisi pengelolaan penanggulangan yang dapat digunakan hanya admin.

**3.4 Perancangan Rule Base**

*Rule base* adalah sistem yang digunakan sebagai cara menyimpan dan memanipulasi pengetahuan untuk diwujudkan dalam suatu informasi yang dapat membantu dalam menyelesaikan berbagai permasalahan. Pada pengembangan *rule base* yang akan dipresentasikan dalam bentuk tabel keputusan dan pohon keputusan untuk hama dan penyakit bawang merah.

**3.4.1 Tabel Keputusan**

Tabel keputusan (decision table) adalah tabel yang digunakan sebagai alat bantu untuk menyelesaikan logika dalam program. Algoritma yang berisi keputusan bertingkat yang banyak sekali sangat sulit untuk digambarkan langsung dengan structured English atau pseudocode dan dapat dibuat terlebih dahulu dengan menggunakan tabel keputusan. Dengan demikian tabel keputusan efektif digunakan bilamana kondisi yang akan diseleksi didalam program jumlahnya cukup banyak dan rumit.

Pada Tabel 1 menunjukkan relasi dari gejala dan gangguan yang ditandai dengan ✓. A1 sampai A6 merupakan nama hama bawang merah sedangkan B1 sampai B17 merupakan gejala hama bawang merah. C1 sampai C7 merupakan nama penyakit bawang merah sedangkan D1 sampai D20 merupakan gejala penyakit bawang merah.

**Tabel 1.** Tabel Keputusan Hama Bawang Merah

Kode Gejala	Kode Gangguan					
	A1	A2	A3	A4	A5	A6
B1		✓		✓		✓
B2	✓	✓				
B3			✓	✓		
B4		✓				
B5	✓					
B6	✓					
B7	✓					
B8		✓				
B9			✓			
B10			✓			
B11			✓			
B12				✓		
B13					✓	
B14					✓	
B15					✓	

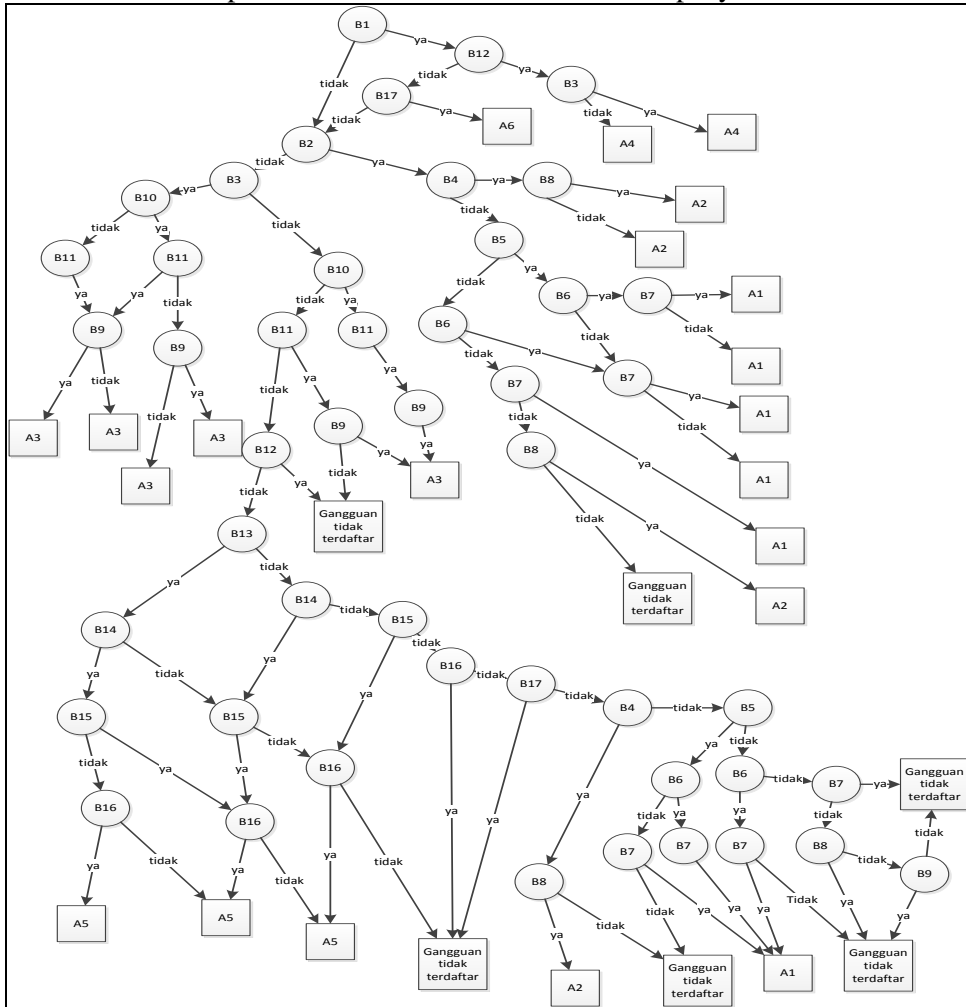


B16					✓	
B17						✓

**3.4.2 Pohon Keputusan**

Proses pada pohon keputusan adalah mengubah bentuk tabel menjadi model pohon, mengubah model pohon menjadi rule, dan menyederhanakan rule. Pohon Keputusan juga berguna untuk mengeksplorasi data, menemukan hubungan tersembunyi antara sejumlah calon variabel input dengan sebuah variabel target.

Pada Gambar 5 akan menunjukkan tentang pohon keputusan yang nantinya akan dijadikan acuan untuk pembuatan rule identifikasi hama dan penyakit



**Gambar 5.** Pohon Keputusan Identifikasi Hama Bawang Merah

**3.4.3 Kaidah Produksi**

Pohon keputusan yang dihasilkan digunakan sebagai acuan dalam menyusun kaidah. Atribut di dalam tabel keputusan menjadi premis di dalam kaidah yang direpresentasikan secara kaidah produksi. Berikut ini gambaran dari kaidah produksi yang dihasilkan dari pohon keputusan.

Kaidah 1 : IF B1 AND B12 AND B3 THEN A4

Kaidah 2 : IF B1 AND B17 THEN A6

Kaidah 3 : IF D1 AND D8 AND D9 AND D20 THEN C4

Kaidah 4 : IF D6 AND D11 AND D18 THEN C6

**3.4.4 Perhitungan Prosentase**

Adapun langkah perhitungan prosentase hama dan penyakit bawang merah adalah sebagai berikut:

$$P = \frac{M}{N} \times 100\% \quad (1)$$

Dimana:

P = Prosentase hama atau penyakit

M = Jumlah gangguan yang dipilih

N = Jumlah gangguan yang ada [1]

Sebagai contoh, jumlah gangguan yang dipilih ada 3 yaitu (D1) Timbul bercak daun putih keabuan, (D8) Daun mengecil dan berwarna gelap, (D9) Daun patah secara serentak maka hasilnya (C4) Antraknosa.

Langkah perhitungannya sebagai berikut:

$$P = \frac{M}{N} \times 100\%$$

$$P = \frac{3}{4} \times 100\%$$

P = 75% untuk penyakit antraknosa

## 4 IMPLEMENTASI DAN UJICOBA

### 4.1 Implementasi

Implementasi merupakan tahap pengembangan rancangan menjadi kode program. Pada awal bagian ini dijabarkan spesifikasi perangkat keras dan lunak pada mana program diimplementasikan. Bagian utama implementasi adalah penjabaran rancangan kelas menjadi kelas yang ditulis dalam sintaks Bahasa Pemrograman Java. Di samping itu disajikan juga tampilan *interface* aplikasi setelah diimplementasikan pada telepon genggam Lenovo A6000 PLUS

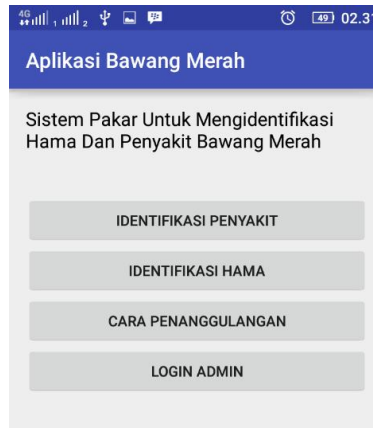
### 4.2 Implementasi Desain Interface

Setelah selesai membuat perancangan desain *interface* yang dilakukan pada bab sebelumnya, maka pada bab ini akan diimplementasikan desain *interface* yang telah dibuat pada bab sebelumnya. Pada bab sebelumnya telah dijelaskan bahwa aplikasi ini memiliki sebuah form yang berisi beberapa komponen.

#### 4.2.1 Form Menu

Berdasarkan perancangan, tampilan form menu merupakan tampilan utama yang menampilkan form Identifikasi penyakit, form identifikasi hama, form cara penanggulangan dan form login admin.

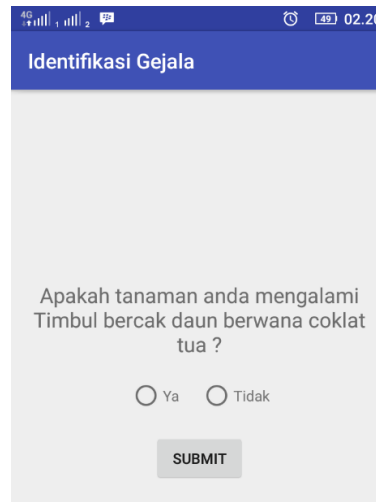
1. *Button* Identifikasi penyakit berfungsi untuk menampilkan pertanyaan konsultasi berupa gejala – gejala untuk penyakit bawang merah
2. *Button* Identifikasi hama berfungsi untuk menampilkan pertanyaan konsultasi berupa gejala – gejala untuk hama bawang merah
3. *Button* cara penanggulangan berfungsi untuk *user* melihat cara penanggulangan berupa *ListView*



Gambar 6. Form Utama Aplikasi Bawang Merah

#### 4.2.2 Form Pertanyaan Gejala

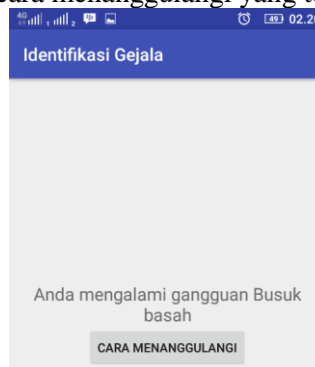
Pada form pertanyaan gejala merupakan form dari *button* identifikasi hama dan *button* identifikasi penyakit. Isi dari form ini merupakan pertanyaan gejala dan *user* akan memilih jawaban antara ya atau tidak menggunakan *RadioButton*. Setelah itu *user* mengkonfirmasi jawaban menggunakan *button submit* kemudian akan diproses menuju pertanyaan selanjutnya.



Gambar 7. Form Pertanyaan Gejala

#### 4.2.3 Form Hasil Hama Dan Penyakit

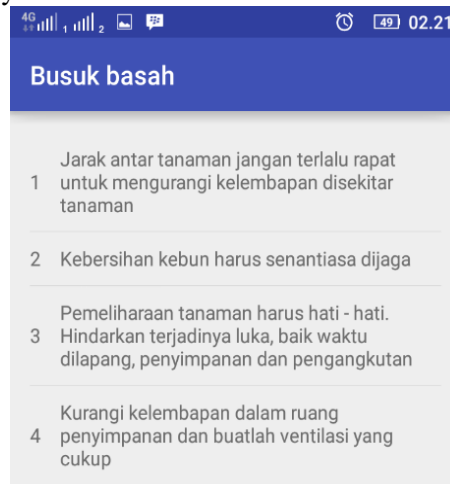
Setelah *user* menjawab pertanyaan gejala maka akan muncul hasil dari gejala – gejala yang telah dijawab. Hasil ini berupa jenis hama atau penyakit pada bawang merah. Selain muncul hasil, *user* dapat mengetahui cara penanggulangan hama atau penyakit dari bawang merah melalui *button cara menanggulangi yang tepat* di bawah hasil.



**Gambar 8.** Form Hasil Hama dan Penyakit

**4.2.4 Form Hasil Penanggulangan**

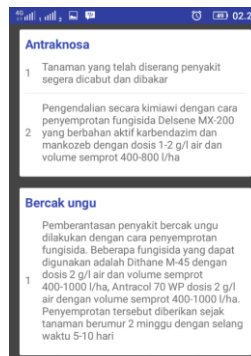
Form ini berisikan hasil tentang cara penanggulangan hama atau penyakit pada bawang merah. Form ini dapat dilihat setelah *user* menerima hasil dari pertanyaan yang telah dijawab. Seperti contoh penyakit busuk basah, maka akan keluar cara penanggulangan dari penyakit busuk basah.



**Gambar 9.** Form Hasil Penanggulangan

**4.2.5 Form Cara Penanggulangan**

Pada Form ini berisi bagaimana cara menanggulangi hama dan penyakit bawang merah. Form ini menampilkan penanggulangan berupa *ListView* untuk *user*. *User* dapat menscroll kebawah atau keatas untuk melihat penanggulangan hama atau penyakit bawang merah.



**Gambar 10.** Form Cara Penanggulangan

**5. PENUTUP**

**5.1 KESIMPULAN**

Berdasarkan hasil uji coba dari bab sebelumnya, maka dapat ditarik kesimpulan sebagai berikut:

1. Petani yang ingin mengetahui informasi mengenai hama dan penyakit bawang merah dapat menggunakan fitur identifikasi hama dan identifikasi penyakit yang terdapat pada menu aplikasi dengan cara konsultasi sedangkan cara pengendalian didapat setelah proses konsultasi selesai dan juga terdapat pada menu yaitu cara pengendalian.
2. Merancang *rule base* hama dan penyakit pada tanaman bawang merah dalam penggunaan sistem pakar dengan cara membuat tabel keputusan dan pohon keputusan kemudian akan menghasilkan *rule base* hama dan penyakit pada bawang merah.

**5.2 SARAN**

Berdasarkan hasil penelitian dan pembahasan serta kesimpulan yang telah diuraikan diatas, selanjutnya dapat disarankan sebagai berikut :

1. Admin tidak hanya memperbarui cara penanggulangan tetapi juga untuk memperbarui gejala, hama dan penyakit bawang merah.
2. Perbaikan untuk informasi seharusnya dapat dilakukan, agar pengguna lebih mengetahui dengan cara memberikan gambar tentang hama dan penyakit pada bawang merah.

## 6. DAFTAR RUJUKAN

- [1] Baur Gregory R. dan Pigford D.V. 1990. “*Expert System For Business : Concept and Implementations*”, Boyd & Fraser Publishing Company, Boston.
- [2] Juman, Kundang K. 2013. “Java” <http://kundang.weblog.esaunggul.ac.id/2013/09/> diakses pada tanggal 5 September 2015.
- [3] Kendall, Kendall. 2003. “Analisis dan Perancangan Sistem Jilid 1”. PT Prenhallindo. Jakarta.
- [4] Merlina, Nita. Hidayat, Rahmat. 2012. ”Perancangan Sistem Pakar”. Ghalia Indonesia. Jakarta
- [5] Nugroho, Adi. 2008. “Pemrograman Java Menggunakan IDE Eclipse Callisto”. Penerbit ANDI. Yogyakarta
- [6] Nuriya, Tri Ulfa. 2009. “Sistem Pakar Berbasis Mobile Untuk Mendeteksi Hama Dan Penyakit Pada Tanaman Bawang Merah”. Skripsi. Jurusan Teknologi Informatika. Universitas Pembangunan Nasional “Veteran” Jawa Timur. Surabaya.
- [7] Pratama, Wirandha Ryan dkk. 2013. “Rancang Bangun Aplikasi Sistem Pakar Untuk Menentukan Penyakit Pada Tanaman Kedelai”. Program Studi / Jurusan Sistem Informasi STMIK STIKOM. Surabaya
- [8] Sumarni, N. dan Achmad Hidayat. 2005. “Budidaya Bawang Merah”. Balai Penelitian Tanaman Sayuran. Jakarta Selatan.
- [9] Samsudin. 1979. “Bawang Merah”. Binacipta. Bandung.
- [10] Semangun, Haryono. 2004. “Penyakit – Penyakit Tanaman Hortikultura Di Indonesia”. Gajah Mada University Press. Yogyakarta.
- [11] Wahana Komputer. 2013. ”Shortcourse Android Programming with Eclipse”. Penerbit ANDI. Yogyakarta.
- [12] Whitten, Jeffery L. dkk. 2004. “Metode desain dan analisis sistem. edisi ke-6”. ANDI. Yogyakarta.
- [13] Yaqin, Ainul dan Ema Utami. 2014. “Sistem Pakar Pemetaan Kelas Siswa LBB “abc” Menggunakan Metode Forward Chaining”. Magister Teknik informatika STMIK AMIKOM. Yogyakarta.
- [14] Yuswanto. 2006. “Pemrograman Dasar Visual Basic .NET”. Prestasi Pustakaraya. Jakarta.