

**IMPLEMENTASI SISTEM PAKAR BERBASIS ANDROID PADA PENYAKIT  
TANAMAN LADA DENGAN MENGGUNAKAN METODE *FORWARD CHAINING***

(Skripsi)

**RIZKA ESA BASRI**



**JURUSAN ILMU KOMPUTER**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS LAMPUNG**

**2017**

## ABSTRACT

### IMPLEMENTATION OF ANDROID-BASED EXPERT SYSTEM ON THE DISEASE THAT EXIST ON PEPPER PLANTS USING FORWARD CHAINING METHOD

Disease on paper plants is undesirable for people who have paper planting because it can make the harvest are not maximum and finally paper plants will die. To solve this problem, people need knowledge about the information of the disease, symptom, and how to handling the disease. Based on this problem, we need expert system to diagnose the disease on paper plants. This system aims to plan and built Android expert system. In this research, expert system is built based on android system and java programming language as the database. The reasoning method used in this research is forward chaining. This method is used to determine which rules will be used, then it is executed, finally the process is repeated until the results are found. This system could diagnose 11 diseases of pepper plants with 31 symptoms. From the data result, using *Equivalence Partitioning* showed that the manager rule system could run according to function and system could diagnose the disease well. In addition, based on questionnaire data, this application was *user friendly* application (with the average value 85,75% or very good)

**Key Words:** *Forward Chaining, Diseases on Pepper Plants, Expert System, Android*

## ABSTRAK

# IMPLEMENTASI SISTEM PAKAR BERBASIS ANDROID PADA PENYAKIT TANAMAN LADA DENGAN MENGUNAKAN METODE *FORWARD CHAINING*

Oleh

**RIZKA ESA BASRI**

Penyakit pada tanaman lada merupakan hal yang tidak diinginkan bagi para masyarakat yang memiliki lahan penanaman lada, karena dapat menyebabkan panen tidak maksimal dan kematian pada tanaman lada. Untuk mengatasi kendala tersebut maka masyarakat membutuhkan suatu pengetahuan tentang informasi penyakit, gejala, dan penanganan untuk penyakit tersebut. Didasari kendala penyakit tanaman lada tersebut maka dibutuhkan sistem pakar yang digunakan untuk mendiagnosa penyakit pada tanaman lada. Pada penelitian ini bertujuan untuk merancang dan membangun aplikasi sistem pakar berbasis android. Sistem pakar yang dibangun pada penelitian ini berbasis *Android* menggunakan bahasa pemrograman *Java* sebagai basis data. Metode penalaran yang digunakan dalam penelitian ini yaitu *Forward Chaining*. Metode ini digunakan untuk menentukan aturan mana yang akan dijalankan, kemudian aturan tersebut dijalankan, proses diulang sampai hingga ditemukan hasil. Data yang digunakan dalam penelitian ini adalah 11 penyakit tanaman lada dengan 31 gejala. Hasil data pengujian menggunakan pengujian *Equivalence Partitioning*, menunjukkan bahwa pengelolaan aturan (*rule*) sistem dapat berjalan sesuai fungsinya dan sistem dapat mendiagnosa penyakit dengan baik. Selain itu berdasarkan data angket, aplikasi ini adalah aplikasi yang *user friendly* (dengan nilai rata-rata 85,75% / sangat baik).

**Kata kunci :** *Forward Chaining*, Penyakit Tanaman Lada, Sistem Pakar, *Android*

**IMPLEMENTASI SISTEM PAKAR BERBASIS ANDROID PADA  
PENYAKIT TANAMAN LADA DENGAN MENGGUNAKAN METODE  
*FORWARD CHAINING***

Oleh

**RIZKA ESA BASRI**

Skripsi

**Sebagai Salah Satu Syarat Untuk Memperoleh Gelar  
SARJANA KOMPUTER**

Pada

**Jurusan Ilmu Komputer  
Fakultas Matematika dan Ilmu Pengetahuan Alam**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2017**

Judul Skripsi : **IMPLEMENTASI SISTEM PAKAR  
BERBASIS ANDROID PADA PENYAKIT  
TANAMAN LADA DENGAN  
MENGUNAKAN METODE FORWARD  
CHAINING**

Nama Mahasiswa : **Rizka Esa Basri**

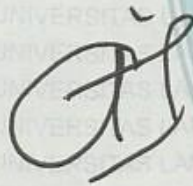
Nomor Pokok Mahasiswa : 1317051058

Jurusan : Ilmu Komputer

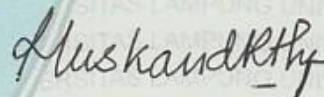
Fakultas : Matematika dan Ilmu Pengetahuan Alam

**MENYETUJUI**

1. Komisi pembimbing,

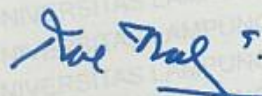


**Aristoteles, S.Si., M.Si.**  
NIP. 19810521 200604 1 002



**Dr. Ir. Suskandini Ratih, D.MP.**  
NIP. 19610502 198707 2 001

2. Mengetahui  
Ketua Jurusan Ilmu Komputer



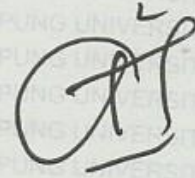
**Dr. Ir. Kurnia Muludi, M.S.Sc.**  
NIP. 19640616 198902 1 001



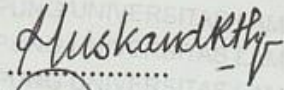
**MENGESAHKAN**

**1. Tim Penguji**

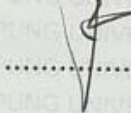
**Ketua : Aristoteles, S.Si., M.Si.** .....



**Sekretaris : Dr. Ir. Suskandini Ratih D.MP.** .....



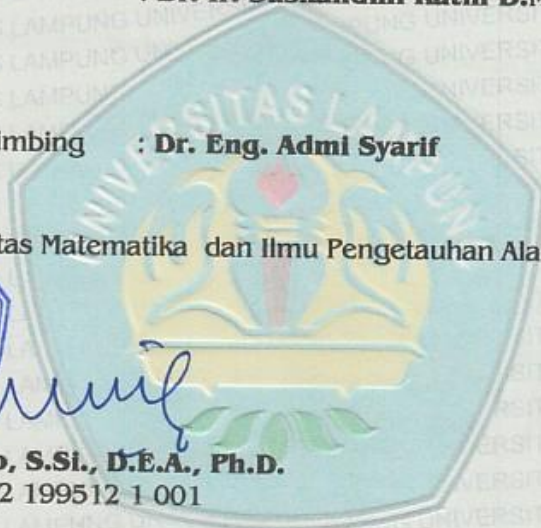
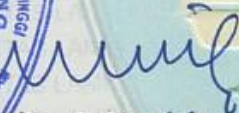
**Penguji  
Bukan Pembimbing : Dr. Eng. Admi Syarif** .....



**2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam**



**Prof. Warsito, S.Si., D.E.A., Ph.D.**  
NIP. 19710212 199512 1 001



**Tanggal Lulus Ujian Skripsi : 05 September 2017**

## PERNYATAAN

Saya yang bertanda tangan dibawah ini, menyatakan bahwa skripsi saya yang berjudul **“Implementasi Ssitem Pakar Berbasis Android Pada Penyakit Tanaman Lada Dengan Menggunakan Metode *Forward Chaining*”** merupakan karya saya sendiri dan bukan hasil karya orang lain. Semua tulisan yang tertuang di skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi saya merupakan hasil penjiplakan atau dibuat orang lain, maka saya bersedia menerima sanksi berupa pencabutan gelar yang telah saya terima.

Bandar Lampung, 11 September 2017



Rizka Esa Basri  
NPM. 1317051058

## RIWAYAT HIDUP



Penulis dilahirkan di Kotabumi Lampung Utara pada tanggal 17 Januari 1995, sebagai anak ketiga dari tiga bersaudara dari pasangan Bapak Basri dan Ibu Sri Puryani.

Penulis memulai pendidikan di Taman Kanak-Kanak Aisyiyah Bustanul Athfal pada tahun 2001, kemudian melanjutkan pendidikan Sekolah Dasar di SDN 02

Tanjung Aman Kotabumi Lampung Utara yang diselesaikan pada tahun 2007, kemudian Sekolah Menengah Pertama di SMPN 03 Kotabumi Lampung Utara yang diselesaikan pada tahun 2010, dan Sekolah Menengah Atas di SMAN 03 Kotabumi Lampung Utara yang diselesaikan pada tahun 2013.

Tahun 2013, penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung melalui jalur Seleksi Nasional Masuk Perguruan Tinggi Negeri (SNMPTN). Pada tahun 2016, penulis telah melaksanakan Kuliah Kerja Nyata (KKN) Tematik di Desa Payung Batu Kecamatan Pubian Kabupaten Lampung Tengah.



## **PERSEMBAHAN**

*Puji dan syukur saya panjatkan kepada Allah SWT atas segala berkah-Nya sehingga skripsi ini dapat terselesaikan.*

*Kupersembahkan karya ini kepada :*

*Papi dan Mami yang telah membesarkan, mendidik, memberikan doa, dukungan dan semangat untuk kesuksesan anak-anaknya. Terimakasih atas semua perjuangan, pengorbanan, kesabaran dan kasih sayang telah kalian berikan untukku.*

*Mba dan Wanda yang aku sangat sayangi Ari Febriyanti dan Sen Freddie Putra serta keluarga besar tercinta.*

*Keluarga Ilmu Komputer 2013,*

*Serta Almamater Tercinta,*

*Universitas Lampung.*

## MOTTO

*Allah SWT memfasilitasi umatnya dengan cara yang berbeda-beda, ada yang ia berikan lift dan ada yang harus memijak anak tangga satu persatu untuk mencapai kesuksesan, bahkan ada yang harus terjatuh terguling dan memulai pijakan lagi dengan rasa sakit yang didapat. Inilah hidup nikmat saja prosesnya, tinggal bagaimana umatnya bersungguh-sungguh dalam proses mencapai kesuksesan, karena tidak ada yang tidak mungkin dan tidak ada yang tidak bisa apabila kita terus berdoa dan berikhtiar.*

## SANWACANA

Puji syukur penulis panjatkan kehadirat ALLAH SWT atas berkat, rahmat dan hidayah-Nya penulis dapat menyelesaikan skripsi yang berjudul “Implementasi Sistem Pakar Berbasis Android Pada Penyakit Tanaman Lada Dengan Menggunakan Metode *Forward Chaining*”. Oleh karena itu penulis ingin mengucapkan terima kasih kepada :

1. Papi dan Mami yang tanpa lelah memberikan segala bentuk dukungan, motivasi serta senantiasa mendoakan penulis.
2. Bapak Aristoteles S.Si., M.Si., selaku dosen pembimbing utama yang telah memberikan arahan, bantuan, saran, serta waktunya selama penulis menjadi mahasiswa maupun dalam penyusunan skripsi ini.
3. Ibu Dr. Ir. Suskandini Ratih D.MP., selaku dosen pembimbing pendamping yang telah memberikan bantuan, motivasi, saran, serta waktunya dalam penyusunan skripsi ini.
4. Bapak Dr. Eng. Admi Syarif., selaku dosen penguji yang telah memberikan masukan dan saran dalam penyusunan skripsi ini.
5. Bapak Rico Andrian, S.Si., M.Kom., selaku pembimbing akademik yang telah memberikan motivasi selama penyusunan skripsi ini.
6. Bapak Dr. Ir. Kurnia Muludi, M.S.Sc., selaku Ketua Jurusan Ilmu Komputer FMIPA Universitas Lampung.

7. Bapak Didik Kurniawan, S.Si., M.T., selaku Sekretaris Jurusan Ilmu Komputer FMIPA Universitas Lampung.
8. Bapak dan Ibu dosen yang telah memberikan ilmu yang bermanfaat selama penulis berada di Jurusan Ilmu Komputer.
9. Mba Ari Febriyanti dan Wanda Sen Fredie Putra yang telah mendoakan penulis.
10. Navia Yufitasari yang selalu menemani, mengingatkan, memotivasi, membantu, dan mendoakan penulis selama pengerjaan skripsi ini.
11. Teman-teman seperjuangan di semester akhir : Mevici, Nisa, Ratu, Mita, Gita, Rina, Tika, Bunga yang telah menemani penulis di semester akhir.
12. Rio Pradipa yang telah menemani penulis selama mencari data kebutuhan pengerjaan skripsi ini
13. Seluruh pihak yang telah membantu penulis.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna, namun penulis berharap semoga skripsi ini bermanfaat bagi yang membutuhkan.

Bandar Lampung, Agustus 2017

Penulis,

Rizka Esa Basri

## DAFTAR ISI

	Halaman
<b>HALAMAN JUDUL .....</b>	<b>i</b>
<b>LEMBAR PERSETUJUAN .....</b>	<b>ii</b>
<b>LEMBAR PENGESAHAN .....</b>	<b>iii</b>
<b>LEMBAR PERNYATAAN .....</b>	<b>iv</b>
<b>RIWAYAT HIDUP .....</b>	<b>v</b>
<b>PERSEMBAHAN.....</b>	<b>vi</b>
<b>MOTTO .....</b>	<b>vii</b>
<b>SANWACANA .....</b>	<b>viii</b>
<b>DAFTAR ISI.....</b>	<b>x</b>
<b>DAFTAR GAMBAR.....</b>	<b>xiv</b>
<b>DAFTAR TABEL .....</b>	<b>xv</b>
<b>BAB I. PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah .....	4
1.3 Batasan Masalah .....	4
1.4 Tujuan Penelitian .....	5
1.5 Manfaat Penelitian .....	5
<b>BAB II. TINJAUAN PUSTAKA.....</b>	<b>6</b>
2.1 Budidaya Tanaman Lada .....	6
2.1.1 Sentra Produksi Lada Nasional.....	7
2.1.2 Perkembangan Konsumsi Per Kapita Lada Nasional .....	8
2.1.3 Penyakit Penting Tanaman Lada .....	9
2.1.3.1 Penyakit Busuk Pangkal Batang.....	9

2.1.3.2	Penyakit Keriting Daun .....	10
2.1.3.3	Penyakit Kuning Lada .....	11
2.1.3.4	Penyakit Jamur Upas Lada .....	12
2.1.3.5	Penyakit Jamur Marasmius Lada.....	12
2.1.3.6	Penyakit Mati Ujung Lada.....	13
2.1.3.7	Penyakit Bercak Daun Lada .....	13
2.1.3.8	Penyakit Karat Merah Lada.....	13
2.1.3.9	Penyakit Buah Hitam Lada.....	13
2.1.3.10	Penyakit Jamur Akar Lada .....	14
2.2	Sistem Pakar.....	14
2.2.1	Definisi Sistem Pakar.....	14
2.2.2	Ciri-Ciri Sistem Pakar.....	15
2.2.3	Manfaat SistemPakar .....	16
2.2.4	Konsep Dasar Sistem Pakar .....	17
2.2.4.1	Kepakaran .....	17
2.2.4.2	Pakar .....	17
2.2.4.3	Pengalihan Kepakaran .....	18
2.2.4.4	Inferensi .....	18
2.2.4.5	Aturan .....	18
2.2.4.6	Kemampuan Menjelaskan.....	19
2.2.5	Struktur Sistem Pakar .....	19
2.2.6	Basis Pengetahuan .....	22
2.2.7	Mekanisme Inferensi.....	23
2.2.7.1	Metode <i>Forward Chaining</i> .....	24
2.3	Metode Pengembangan Sistem .....	29
2.3.1	Metode <i>Waterfall</i> .....	29
2.3.2	<i>Unified Modelling Language</i> (UML).....	31
2.4	Teknik Pengujian Perangkat Lunak .....	35
2.4.1	<i>Equivalence Partitioning</i> .....	35



2.4.2 Probabilitas Klasik .....	35
2.4.3 <i>Skala Likert</i> .....	36
<b>BAB III. METODE PENELITIAN .....</b>	<b>38</b>
3.1 Waktu dan Tempat Penelitian .....	38
3.2 Perangkat Penelitian.....	38
3.3 Tahapan Penelitian.....	39
3.3.1 Studi Literatur .....	39
3.3.2 Pengumpulan Data .....	40
3.3.3 Perancangan Sistem .....	40
3.3.3.1 Perancangan UML ( <i>Unified Modelling Language</i> ).....	40
3.3.3.2 Perancangan Antarmuka.....	48
3.4 Metode Pengujian Sistem .....	54
<b>BAB IV. HASIL DAN PEMBAHASAN .....</b>	<b>58</b>
4.1 Analisa Kebutuhan Data .....	58
4.2 Representasi Pengetahuan.....	58
4.3 Implementasi Sistem .....	59
4.4 Tampilan Aplikasi Sistem Pakar Diagnosa Penyakit Tanaman Lada	60
4.4.1 Tampilan Halaman <i>Splash Screen</i> .....	60
4.4.2 Tampilan Menu Utama .....	60
4.4.3 Tampilan Menu Data Penyakit .....	61
4.4.4 Tampilan Menu Konsultasi.....	62
4.4.5 Tampilan Menu Hasil Analisa .....	62
4.4.6 Tampilan Halaman Lihat Penjelasan Penyakit .....	63
4.4.7 Tampilan Menu Bantuan.....	64
4.4.8 Tampilan Menu Tentang.....	64
4.5 Analisa Persentase Penyakit .....	65
4.6 Hasil Pengujian .....	69
4.6.1 Pengujian Fungsional.....	69
4.6.1.1 Pengujian Versi Android .....	70

4.6.1.2 Pengujian Resolusi Layar dan Densitas Layar .....	71
4.6.1.3 Pengujian <i>User Interface</i> .....	73
4.6.1.4 Pengujian Fungsi dari Menu Aplikasi .....	75
4.6.2 Pengujian Kepakaran Sistem .....	77
4.6.3 Pengujian Non Fungsional .....	80
<b>BAB V. SIMPULAN DAN SARAN.....</b>	<b>88</b>
5.1 Kesimpulan .....	88
5.2 Saran .....	88
<b>DAFTAR PUSTAKA .....</b>	<b>90</b>
<b>LAMPIRAN.....</b>	<b>93</b>

## DAFTAR GAMBAR

Gambar	Halaman
2.1 Kontribusi Produksi Lada Beberapa Kabupaten Sentra di Provinsi Lampung.....	8
2.2 Penyakit Busuk Pangkal Batang .....	10
2.3 Penyakit Keriting Daun.....	11
2.4 Penyakit Kuning Lada.....	12
2.5 Struktur Sistem Pakar.....	20
2.6 Proses <i>Forward Chaining</i> .....	25
2.7 Diagram Alir Teknik Penelusuran <i>Depth-first Search</i> .....	25
2.8 Diagram Alir Teknik Penelusuran <i>Breadth-first Search</i> .....	26
2.9 Iterasi ke-1.....	27
2.10 Iterasi ke-2.....	28
2.11 Iterasi ke-3.....	28
2.12 Metode <i>Waterfall</i> .....	29
2.13 Contoh <i>Use Case Diagram</i> .....	32
2.14 Contoh Diagram Aktivitas .....	33
2.15 Contoh Diagram Sekuensial.....	33
3.1 Diagram Alir Penelitian .....	39
3.2 <i>Use Case Diagram</i> .....	41
3.3 <i>Activity Diagram</i> Data Penyakit.....	42
3.4 <i>Activity Diagram</i> Konsultasi .....	43
3.5 <i>Activity Diagram</i> Bantuan .....	44
3.6 <i>Activity Diagram</i> Tentang .....	45

3.7	<i>Sequence Diagram</i> Data Penyakit .....	46
3.8	<i>Sequence Diagram</i> Konsultasi .....	47
3.9	<i>Sequence Diagram</i> Bantuan .....	47
3.10	<i>Sequence Diagram</i> Tentang Aplikasi .....	48
3.11	<i>Layout Splash Screen</i> .....	49
3.12	<i>Layout</i> Menu Utama.....	50
3.13	<i>Layout</i> Data Penyakit .....	50
3.14	<i>Layout</i> Konsultasi .....	51
3.15	<i>Layout</i> Sub Menu Hasil Diagnosa Penyakit .....	51
3.16	<i>Layout</i> Menu Bantuan .....	52
3.17	<i>Design Layout</i> Hasil Diagnosa Penyakit (Penjelasan Penyakit) .....	53
3.18	<i>Design Layout</i> Menu Tentang .....	54
4.1	Tampilan <i>Splash Screen</i> .....	60
4.2	Tampilan Menu Utama .....	61
4.3	Tampilan Menu Data Penyakit .....	62
4.4	Tampilan Menu Konsultasi .....	62
4.5	Tampilan Hasil Analisa.....	63
4.6	Tampilan Halaman Lihat Penjelasan Penyakit .....	64
4.7	Tampilan Menu Bantuan.....	64
4.8	Tampilan Menu Tentang .....	65
4.9	Grafik Penilaian Kuisioner Terhadap Pernyataan1 .....	83
4.10	Grafik Penilaian Kuisioner Terhadap Pernyataan2.....	83
4.11	Grafik Penilaian Kuisioner Terhadap Pernyataan3 .....	84
4.12	Grafik Penilaian Kuisioner Terhadap Pernyataan4 .....	84
4.13	Grafik Penilaian Kuisioner Terhadap Pernyataan5 .....	85
4.14	Grafik Penilaian Kuisioner Terhadap Pernyataan6.....	85
4.15	Grafik Penilaian Kuisioner Terhadap Pernyataan7 .....	86
4.16	Grafik Penilaian Kuisioner Terhadap Pernyataan8.....	87

## DAFTAR TABEL

Tabel	Halaman
Tabel 3.1 Tabel Daftar Pengujian .....	55
Tabel 4.1 Pengujian Versi Android.....	70
Tabel 4.2 Pengujian Resolusi Layar dan Densitas Layar.....	72
Tabel 4.3 Pengujian <i>User Interface</i> .....	73
Tabel 4.4 Pengujian fungsi dari menu aplikasi .....	75
Tabel 4.5 Pengujian Analisa Sistem Pakar Penyakit Pada Tanaman Lada.....	77
Tabel 4.6 Interval dan Kategori Penilaian.....	81
Tabel 4.7 Hasil Penilaian Variabel <i>Sistem Pakar</i> .....	82

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Lada (*Piper nigrum*) merupakan jenis rempah yang paling sering digunakan di Eropa dibandingkan dengan rempah-rempah lainnya. Daerah utama penghasil lada di Indonesia ialah daerah Lampung yang terkenal sebagai penghasil lada hitam atau *lampung black pepper* dan Bangka yang terkenal sebagai penghasil lada putih atau *muntok white pepper* (Munif & Sulistiawati, 2014). Menurut Wahid (1996) dalam (Balfas, 2009), tanaman lada (*Piper nigrum L.*) merupakan salah satu tanaman rempah yang penting dan banyak dibudidaya serta merupakan salah satu komoditas ekspor perkebunan andalan Indonesia. Keberhasilan pengembangan tanaman lada ditentukan oleh beberapa hal diantaranya yaitu produktivitas tanaman lada, kesesuaian daerah pengembangan, dan serangan penyebab penyakit lada.

Lampung sejak dahulu telah dikenal sebagai tanah lada, daerah yang menghasilkan lada. Produksi lada di Lampung dari tahun 2013 hingga tahun 2015 berurutan adalah 24.654, 24.783, dan 24.913 ton sedangkan luas pertanamannya adalah 47.485, 47.515, dan 47.381 ha (Direktorat Jenderal Perkebunan & Kementerian Pertanian, 2014). Produksi lada yang mengalami stagnasi diantaranya disebabkan oleh serangan patogen penyebab penyakit. Kehilangan hasil akibat serangan patogen penyebab penyakit dapat dikurangi jika pembudidaya tanaman lada mudah



mendapatkan informasi tentang cara pengendalian penyakit tanaman lada yang bersifat ramah lingkungan dan berkelanjutan dengan benar dan tepat (Manohara & Wahyuno, 2013).

B eberapa patogen penting penyebab penyakit lada seperti busuk pangkal batang lada (*Phytophthora capsici*), penyakit kuning lada (*Radopholus similis*), penyakit layu lada (*Fusarium oxysporum*) telah diinformasikan kepada masyarakat luas. Namun demikian penyakit pada tanaman lada tidak terbatas berupa tiga penyakit penting melainkan masih banyak penyakit lada lainnya. Informasi bioekologi patogen dan cara pengendalian patogen tanaman lada masih sangat perlu disebarluaskan mengingat bahwa pusat informasi yang berupa Balai Proteksi Tanaman masih sangat terbatas di Lampung dan saling berjauhan letaknya (hanya terdapat tiga Balai Proteksi Tanaman di Lampung yaitu Gading Rejo Pringsewu, Trimurjo Lampung Timur dan Semuli Raya Lampung Utara). Atas dasar ini maka masyarakat masih sukar mencari solusi pengendalian patogen lada di Lampung.

Penyakit pada tanaman lada merupakan hal yang tidak diinginkan bagi para masyarakat yang memiliki lahan penanaman lada, karena dapat menyebabkan panen tidak maksimal dan kematian pada tanaman lada. Untuk mengatasi kendala tersebut maka masyarakat membutuhkan suatu pengetahuan tentang informasi penyakit, gejala, dan penanganan untuk penyakit tersebut. Berdasarkan kepada kendala tersebut maka dibutuhkan sistem pakar penyakit pada tanaman lada sehingga dapat mengurangi waktu dan biaya untuk mencari informasi tentang lada. Informasi tersebut dapat diperoleh dengan menggunakan android yang sekarang telah umum dimiliki oleh masyarakat.

Sistem pakar yang dibuat yaitu berbasis *Android* sehingga masyarakat dapat dengan mudah berkonsultasi dimana saja asalkan *Android* terkoneksi dengan internet. Metode inferensi yang digunakan dalam pembuatan sistem ini yaitu *Forward Chaining*. Menurut (Susanti, Jusak, & Sudarmaningtyas, 2016) runut maju (*Forward Chaining*) berarti menggunakan himpunan aturan kondisi-aksi. Dalam metode ini, data digunakan untuk menentukan aturan mana yang akan dijelaskan, kemudian aturan tersebut dijalankan.

Penelitian tentang sistem pakar dengan menggunakan metode *forward chaining* yang telah dilakukan oleh (Hamdani, 2010) menghasilkan sistem pakar yang dapat mendeteksi beberapa penyakit mata pada manusia berdasarkan gejala-gejalanya. Data penyakit yang dikenali menyesuaikan rules (aturan) yang dibuat untuk dapat mencocokkan gejala-gejala penyakit mata dan memberi nilai persentase agar mengetahui nilai pendekatan jenis penyakit pasien. Beberapa gejala penyakit mata dapat menyimpulkan jenis penyakit mata sesuai persentase tertinggi. Semakin tinggi nilai persentase dari hasil penelusuran, maka menunjukkan jenis penyakit mata yang di derita oleh pasien penyakit mata. Namun pada penelitian tersebut memiliki kelemahan yaitu sistem yang dibangun masih berbasis dekstop, sehingga dalam penggunaannya kurang praktis, karena sistem dapat digunakan pada komputer *stand alone*.

Penelitian selanjutnya dilakukan oleh (Istanto & Dewa, 2016) menghasilkan sistem pakar yang dapat mendeteksi hama dan penyakit pada tanaman tomat berdasarkan gejala-gejalanya. Pada penelitian tersebut memiliki kelemahan yaitu sistem pakar yang dibangun masih berbasis web, sehingga dalam penggunaan sistemnya kurang

praktis, karena pengguna harus mengakses website yang telah dibuat dan membutuhkan koneksi yang stabil agar sistem dapat berjalan dengan lancar.

Dengan adanya Sistem Pakar ini diharapkan mampu memberikan suatu informasi yang cukup mengenai penyakit pada tanaman lada dan melakukan pengidentifikasian penyakit secara cepat, tepat, dan efisien serta dapat memberikan solusi untuk menanggulangi penyakit.

## **1.2 Rumusan Masalah**

Permasalahan yang akan dibahas dalam penelitian ini adalah bagaimana merancang dan membangun suatu sistem pakar yang dapat digunakan untuk mendiagnosa penyakit pada tanaman lada pada sistem *Android*.

## **1.3 Batasan Masalah**

Batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Sistem pakar yang dibangun berbasis *Android*
2. Penyakit yang dapat diidentifikasi sebanyak 11 penyakit dengan 31 gejala yang menyerang tanaman lada.
3. Metode penalaran yang digunakan yaitu *Forward Chaining*.

#### **1.4 Tujuan Penelitian**

Tujuan dari penelitian ini adalah merancang dan membangun aplikasi sistem pakar berbasis Android menggunakan metode penalaran *forward chaining*.

#### **1.5 Manfaat Penelitian**

Manfaat dari penelitian ini adalah sebagai berikut:

1. Mempermudah peneliti atau pembudidaya untuk mengetahui jenis gejala dari penyakit yang menyerang tanaman lada.
2. Memberikan informasi kepada peneliti atau pembudidaya untuk mengetahui bagaimana cara menangani penyakit pada tanaman lada.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Budidaya Tanaman Lada**

Menurut (Manohara & Wahyuno, 2013) merica atau lada merupakan nama atau sebutan daerah untuk tanaman yang bernama latin *Piper nigrum L* yang berasal dari India. Sebutan merica digunakan di daerah Sumatra Barat dan Sulawesi, di Jawa Tengah dan Timur disebut merico, di Jawa Barat disebut pedes, sedangkan di Bangka-Belitung dan Kalimantan dikenal dengan sebutan sahang.

Buah lada Indonesia sudah dikenal dunia sejak sebelum Perang Dunia Kedua. Saat itu daerah penghasil utama merica adalah Lampung yang memproduksi lada hitam yang dikenal dunia dengan sebutan '*Lampung black pepper*', dan Bangka-Belitung yang memproduksi merica putih yang dikenal dengan sebutan '*Muntok white pepper*'. Saat ini tanaman lada telah tersebar di Jawa, Sumatra, Kalimantan dan Sulawesi.

Menurut (Agro, 2012), lada (*Piper nigrum L.*) merupakan tanaman rempah-rempah yang potensial dan mempunyai nilai ekonomi tinggi dalam perdagangan internasional. Menurut sejarah dan sumber literatur, tanaman lada termasuk tanaman yang banyak dikembangkan di Indonesia berasal dari daerah Ghat Barat, India. Buktinya pada tahun 100-600 SM banyak koloni Hindu yang datang ke Indonesia. Diperkirakan mereka inilah yang membawa bibit lada pertama kalinya

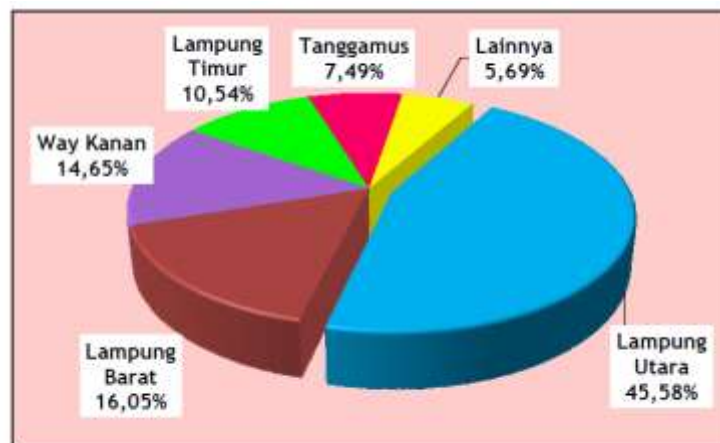
ke Indonesia. Usaha lada sekarang banyak dilirik orang. Banyak orang yang mengetahui bahwa usaha ini mempunyai prospek yang cerah. Prospek ini didukung dengan adanya pasar yang terbuka baik didalam maupun diluar negeri, serta harganya yang lumayan tinggi.

Serangan hama dan penyebab penyakit menyebabkan produktivitas tanaman lada menjadi rendah. Kehilangan hasil akibat serangan hama dan penyebab penyakit dapat ditekan dengan melakukan budidaya anjuran yang bersifat ramah lingkungan dan berkelanjutan dengan benar dan tepat (Manohara & Wahyuno, 2013)

### **2.1.1. Sentra Produksi Lada Nasional**

Provinsi Lampung merupakan provinsi penghasil lada kedua di Indonesia yang dikenal sebagai produsen utama lada hitam (*Lampung Black Pepper*). Sebaran produksi lada di Provinsi Lampung terdapat di lima kabupaten. Kabupaten Lampung Utara menempati posisi pertama dengan produksi lada sebesar 11,24 ribu ton atau 45,58% dari total produksi lada di Lampung, diikuti oleh Kabupaten Lampung Barat (16,05%), Way Kanan (14,65%), Lampung Timur (10,54%) dan Tanggamus (7,49%). Kabupaten lainnya memberikan kontribusi kurang dari 5% (Susanti A. , 2015).





Gambar 2.1 Kontribusi Produksi Lada Beberapa Kabupaten Sentra di Provinsi Lampung 2013 (Susanti A. , 2015)

### 2.1.2. Perkembangan Konsumsi Per Kapita Lada Nasional

Permintaan lada merupakan salah satu aspek yang sangat menentukan daya saing lada Indonesia di pasar domestik maupun di pasar dunia. Perdagangan lada Indonesia umumnya lebih berorientasi ekspor dibandingkan untuk konsumsi domestik. Hasil Survei Sosial Ekonomi Nasional (SUSENAS) dari Badan Pusat Statistik menunjukkan perkembangan konsumsi lada untuk konsumsi langsung pada tahun 2002-2014 cukup berfluktuasi. Konsumsi lada di Indonesia selama periode tersebut meningkat sebesar 1,29% per tahun. Lonjakan konsumsi yang cukup signifikan terjadi pada tahun 2007, dimana konsumsi lada naik dari 0,125 kg/kapita pada tahun 2006 menjadi 0,156 kg/kapita atau naik 25,00% dibandingkan tahun sebelumnya. Konsumsi lada tertinggi terjadi pada tahun 2011 sebesar 0,162% (Susanti A. , 2015).

### **2.1.3. Penyakit Penting Tanaman Lada**

Berikut ini merupakan beberapa penyakit penyebab produktivitas tanaman lada terganggu diantaranya:

#### **2.1.3.1. Penyakit Busuk Pangkal Batang**

Penyakit ini disebabkan oleh jamur *P. capsici* yang dapat menyerang dan mematikan semua bagian tanaman lada. Serangan pada akar atau pangkal batang menyebabkan kelayuan daun mulai dari pucuk lalu ke bawah hingga tanaman mati. Akar dan batang tanaman lada yang terserang berwarna hitam. Daun yang terserang terlihat bercak di bagian tepi daun atau bentuk bulat kehitaman di bagian tengah daun. Penyebaran penyakit sangat cepat pada lingkungan yang lembab, hingga tanaman dapat mati dalam waktu dua minggu setelah muncul layunya daun. Spora disebarkan oleh angin dan air, ternak, peralatan lapangan hingga penyakit ini dapat menyebar dengan cepat. Pengendalian penyakit ini dilakukan dengan membuat parit isolasi pada tanaman terserang ukuran lebar 30 cm dan kedalaman 40 cm. Membuat saluran drainase di dalam dan di luar kebun. Pengendalian kedua adalah mengumpulkan dan membakar tanaman dan daun yang terserang. Pengendalian ke tiga adalah membiarkan tanah terbuka kena sinar matahari; disulam setelah 1-2 tahun kemudian. Untuk pencegahannya sulur tanaman lada dekat permukaan tanah di pangkas sampai ketinggian + 30 cm diatas permukaan tanah. Pengendalian selanjutnya adalah melakukan pemangkasan tajam hidup secara teratur (Direktorat Perlindungan Perkebunan, 2002).



Gambar 2.2. Penyakit Busuk Pangkal Batang (Manohara & Wahyuno, 2013)

### 2.1.3.2 Penyakit Keriting Daun (PKD)

Penyakit ini disebabkan oleh *virus* yang dapat ditularkan oleh serangga *Aphis sp*, *Planococcus sp*, *Pseudococcus sp*, dan melalui alat gunting pangkas. Gejala tanaman terserang, daun-daun berwarna hijau kekuningan, belangbelang kemudian pinggir daun menggulung kedalam atau keriting sehingga bentuk daunnya memanjang dan mengecil. Daun-daunnya terlihat jarang, tangkai buah pendek, buahnya kecil dan sedikit. Pada gejala lanjut pertumbuhan tanaman terhambat atau kerdil dan tidak berbuah. Penyebaran penyakit lebih cepat pada musim kemarau karena serangan serangga yang membawa PKD ini lebih aktif, sehingga penyebarannya lebih cepat dibandingkan musim hujan. Penyebaran penyakit ini juga terjadi melalui gunting pangkas yang digunakan pada tanaman sakit. Apabila di dalam kebun ditemukan hanya satu pohon yang menunjukkan gejala penyakit ini, harus langsung dicabut dan dikubur dalam lubang yang cukup dalam. Jangan dibiarkan berserakan di dalam kebun. Jangan menggunakan tanaman lada di areal terserang sebagai stek untuk bahan tanaman baru yang akan ditanam. Gunting atau

alat pangkas harus dipanaskan dengan korek api sebelum dipakai lagi ke tanaman berikutnya (Direktorat Perlindungan Perkebunan, 2002)



Gambar 2.3. Penyakit Keriting Daun (PKD) (Direktorat Perlindungan Perkebunan, 2002))

### **2.1.3.3 Penyakit Kuning Lada**

Penyebabnya adalah *nematoda* (cacing kecil) seperti *R. similis* dan *M. incognita*, serta jamur *Fusarium* dalam tanah yang biasanya mengandung pasir tinggi. *Nematoda* menyerang akar lada dengan menusuk dan menghisap cairan sehingga terjadi pelukaan akar. Luka tersebut dimasuki jamur dan menyebabkan matinya jaringan pembuluh akar sehingga peredaran air dan unsur hara terganggu. Gejala tanaman terserang, terhambatnya pertumbuhan tanaman terutama pada musim hujan. Warna daun dan sulur jadi kuning pucat, tetapi tidak layu dan tampak kaku. Kemudian daun menjadi kuning, lalu gugur hingga pohon gundul (Direktorat Perlindungan Perkebunan, 2002).



Gambar 2.4. Penyakit Kuning Lada (Direktorat Perlindungan Perkebunan, 2002)

#### **2.1.3.4 Penyakit Jamur Upas Lada**

Penyakit jamur upas disebabkan oleh *Upasia salmonicolor*. Jamur ini lebih sering menginfeksi tanaman lada pada saat musim penghujan gejala yang timbul daun-daun mengering, kemudian pada permukaan cabang atau batang yang sakit terdapat benang jamur berwarna putih mengkilat seperti sarang laba-laba. Jamur ini lama kelamaan akan berkembang menjadi lapisan berwarna merah jambu. Lapisan inilah yang merupakan tanda khas bagi jamur upas (Semangun, 1989).

#### **2.1.3.5 Penyakit Jamur Marasmius Lada**

Penyakit jamur marasmius disebabkan oleh jamur *Marasmius scandens* Mass. Jamur ini membentuk benang-benang putih bercabang-cabang pada batang dan cabang tanaman lada. Kemudian menyebar mencapai daun hingga membuat daun kering dan mati daun yang kering ini akan gugur dan tergantung pada cabang. Hal ini dikarenakan daun tersebut terikat oleh benang jamur yang tumbuh dicabang tanaman (Semangun, 1989).

### **2.1.3.6 Penyakit Mati Ujung Lada**

Penyakit mati ujung pada tanaman lada disebabkan oleh dua jamur yaitu jamur *Gloeosporium* dan *Fusarium*. Jamur ini menyerang pada bagian ujung-ujung cabang tanaman kemudian membuat cabang menjadi mati. Kematian cabang ini selanjutnya akan meluas kebagian pangkal tanaman lada (Semangun, 1989).

### **2.1.3.7 Penyakit Bercak Daun Lada**

Penyakit bercak daun disebabkan oleh jamur-jamur *Pestalotiopsis (Pestalotia)*, *Phyllosticta*, *Curvularia*, *Colletotrichum*, dan *Fusarium*. Jamur ini menyerang pada daun-daun lada yang dewasa. Warna bercak bervariasi dari kelabu, coklat, dan hitam. Bercak-bercak tersebut mempunyai lingkaran-lingkaran yang sepusat dan menimbulkan kerugian pada pertumbuhan tanaman lada (Semangun, 1989).

### **2.1.3.8 Penyakit Karat Merah Lada**

Penyakit karata merah disebabkan oleh *Cephaleuros virescens kunze* yang termasuk kedalam kelas gangga hijau atau (*Chlorophyta*). Patogen ini menyerang pada daun-daun dewasa tanaman lada. Gejala yang timbul daun-daun lada terdapat bercak-bercak hijau kemerahan (Semangun, 1989).

### **2.1.3.9 Penyakit Buah Hitam Lada**

Penyakit buah hitam pada tanaman lada atau yang disebut dengan *black berry disease* sangat merugikan pada pertanaman lada. Penyakit ini disebabkan oleh *Colletotrichum spp.* Jamur ini menyebabkan gugurnya bunga sebelum pembentukan buah dan menghambat perkembangan buah. Buah yang dapat berkembang mempunyai ukuran yang lebih kecil dari biasanya serta terdapat bintik-bintik hitam

kecil. Selain itu terdapat bulu-bulu keemasan yang terdiri dari seporangiofor dan sporangium *Cephaleuros virescens* Kunze. Pada daun tanaman lada terdapat titik-titik hitam. Daun yang terserang berusia muda maupun tua (Semangun, 1989).

#### **2.1.3.10 Penyakit Jamur Akar Lada**

Jamur akar menyerang pada tanaman lada yang disebabkan oleh bermacam-macam patogen. Di Indonesia sendiri belum pernah mendapat laporan terkait serangan jamur akar pada tanaman lada. Akan tetapi penyakit ini sudah menyerang pada tanaman lada di Malaysia, Serawak, dan India. Di Malaysia penyakit jamur akar disebabkan oleh *Rigidoporus microporus*, di Serawak disebabkan oleh *Ganoderma Lucidum*, dan di India diserang oleh jamur akar hitam *Rosellinia bunodes* (Semangun, 1989).

### **2.2. Sistem Pakar**

#### **2.2.1. Definisi Sistem Pakar**

Martin dan Oxman (1988) dalam (Hartati & Iswanti, 2008) mendefinisikan sistem pakar sebagai sistem berbasis komputer yang menggunakan pengetahuan, fakta, dan teknik penalaran dalam memecahkan masalah, yang biasanya hanya dapat diselesaikan oleh seorang pakar dalam bidang tertentu.

Menurut (Putra, 2016) sistem pakar adalah aplikasi berbasis komputer yang digunakan untuk menyelesaikan masalah sebagaimana yang dipikirkan oleh pakar. Pakar yang dimaksud di sini adalah orang yang mempunyai keahlian khusus yang dapat menyelesaikan masalah yang tidak dapat diselesaikan oleh orang awam. Sistem pakar dipandang berhasil ketika mampu mengambil keputusan seperti yang

dilakukan oleh pakar aslinya baik dari sisi proses pengambilan keputusannya maupun hasil keputusan yang diperoleh.

Giarratano dan Riley (2005) dalam (Hartati & Iswanti, 2008) mendefinisikan sistem pakar sebagai salah satu cabang kecerdasan buatan yang menggunakan pengetahuan-pengetahuan khusus yang dimiliki oleh seorang ahli untuk menyelesaikan suatu masalah tertentu.

Menurut Ignizio (1991) dalam (Hartati & Iswanti, 2008) mendefinisikan sistem pakar merupakan bidang yang dicirikan oleh sistem berbasis pengetahuan (*Knowledge Base System*), memungkinkan komputer dapat berfikir dan mengambil kesimpulan dari sekumpulan kaidah.

Menurut Turban dan Aronson (2001) dalam (Hartati & Iswanti, 2008) mendefinisikan sistem pakar sebagai sistem yang menggunakan pengetahuan manusia yang dimasukkan ke dalam komputer untuk memecahkan masalah-masalah yang biasanya diselesaikan oleh pakar.

### **2.2.2. Ciri-Ciri Sistem Pakar**

Ciri-ciri sistem pakar menurut (Pratama, 2015), adalah:

1. Terbatas pada domain keahlian tertentu.
2. Dapat memberikan penalaran untuk data-data yang tidak lengkap atau tidak pasti.
3. Padat penebaran optimal.
4. Dirancang untuk dapat dikembangkan secara bertahap.
5. Pengetahuan dan mekanisme penalaran (*inference*) jelas terpisah.



6. Keluaranya bersifat anjuran.
7. Sistem dapat mengaktifkan kaidah secara searah yang sesuai dituntun oleh dialog dengan user.
8. Sistem dapat mengaktifkan kaidah secara searah yang sesuai dituntun oleh dialog dengan *user*.

### **2.2.3 Manfaat Sistem Pakar**

Menurut (Pratama, 2015) sistem pakar menjadi populer karena sangat banyak kemampuan dan manfaat yang diberikannya, diantaranya:

1. Meningkatkan produktivitas, karena sistem pakar dapat bekerja lebih cepat dari pada manusia.
2. Membuat seorang yang awam bekerja seperti layaknya seorang pakar.
3. Meningkatkan kualitas dengan memberi nasihat yang konsisten dan mengurangi kesalahan.
4. Mampu menangkap pengetahuan dan kepakaran seseorang.
5. Dapat beroperasi di lingkungan yang berbahaya.
6. Memudahkan akses pengetahuan seorang pakar.
7. Handal, sistem pakar tidak pernah menjadi bosan dan kelelahan atau sakit.
8. Meningkatkan kapabilitas sistem komputer. Integrasi sistem pakar dengan sistem komputer lain membuat sistem lebih efektif dan mencakup lebih banyak aplikasi.

9. Mampu bekerja dengan informasi yang tidak lengkap atau tidak pasti.
10. Bisa digunakan sebagai media pelengkap dalam pelatihan. Pengguna pemula yang bekerja dengan sistem pakar akan menjadi lebih berpengalaman karena adanya fasilitas penjelas yang berfungsi sebagai guru.
11. Meningkatkan kemampuan untuk menyelesaikan masalah karena sistem pakar mengambil sumber pengetahuan dari banyak pakar

#### **2.2.4 Konsep Dasar Sistem Pakar**

Pengetahuan dari suatu sistem pakar mungkin dapat direpresentasikan dalam sejumlah cara. Salah satu metode yang paling umum untuk merepresentasikan pengetahuan adalah dalam bentuk tipe aturan (*rule*) **IF...THEN** (Jika...maka).

Konsep dasar dari suatu sistem pakar mengandung beberapa unsur atau elemen, yaitu (Putra, 2016):

##### **2.2.4.1 Kepakaran**

Kepakaran merupakan suatu pengetahuan yang diperoleh dari pelatihan, membaca, dan pengalaman. Kepakaran inilah yang memungkinkan para ahli dapat mengambil keputusan lebih cepat dan lebih baik daripada seseorang yang bukan pakar.

##### **2.2.4.2 Pakar**

Pakar adalah seseorang yang mempunyai pengetahuan, pengalaman, dan metode khusus serta mampu menerapkannya untuk memecahkan masalah atau memberi nasihat. Seorang pakar harus mampu menjelaskan dan mempelajari hal-hal baru yang berkaitan dengan topik permasalahan, jika perlu harus mampu menyusun

kembali pengetahuan-pengetahuan yang didapatkan, dan dapat memecahkan aturan-aturan serta menentukan relevansi kepakarannya.

#### **2.2.4.3 Pengalihan Kepakaran**

Tujuan dari sistem pakar adalah memindahkan kepakaran dari seorang pakar ke dalam komputer, kemudian ditransfer kepada orang lain yang bukan pakar. Proses ini melibatkan empat kegiatan, yaitu:

1. Akuisisi pengetahuan (dari pakar atau sumber lain)
2. Representasi pengetahuan (pada komputer)
3. Inferensi pengetahuan
4. Pemindahan pengetahuan ke pengguna

#### **2.2.4.4 Inferensi**

Inferensi adalah sebuah prosedur yang mempunyai kemampuan dalam melakukan penalaran. Inferensi ditampilkan pada suatu komponen yang disebut mesin inferensi yang mencakup prosedur-prosedur mengenai pemecahan masalah. Semua pengetahuan yang dimiliki oleh seorang pakar disimpan pada basis pengetahuan oleh sistem pakar. Tugas mesin inferensi adalah mengambil kesimpulan berdasarkan basis pengetahuan yang dimilikinya.

#### **2.2.4.5 Aturan**

Kebanyakan aplikasi sistem pakar komersial adalah sistem yang berbasis *rule* (*rule-based system*), yaitu pengetahuan disimpan terutama dalam bentuk *rule*, sebagai prosedur-prosedur pemecahan masalah.

#### **2.2.4.6 Kemampuan Menjelaskan**

Fasilitas lain dari sistem pakar adalah kemampuannya untuk menjelaskan saran atau rekomendasi yang diberikannya. Penjelasan dilakukan dalam subsistem yang disebut subsistem penjelasan (*explanation*). Bagian dari sistem ini memungkinkan sistem untuk memeriksa penalaran yang dibuatnya sendiri dan menjelaskan operasi-operasinya.

#### **2.2.5 Struktur Sistem Pakar**

Sistem pakar dibagi menjadi empat bentuk (Putra, 2016):

1. Berdiri Sendiri

Sistem pakar jenis ini merupakan perangkat lunak yang berdiri sendiri tidak bergabung dengan software lainnya.

2. Tergabung

Sistem pakar jenis ini merupakan bagian program yang terkandung di dalam suatu algoritma (konvensional), atau merupakan program dimana di dalamnya memanggil algoritma subrutin lain (konvensional).

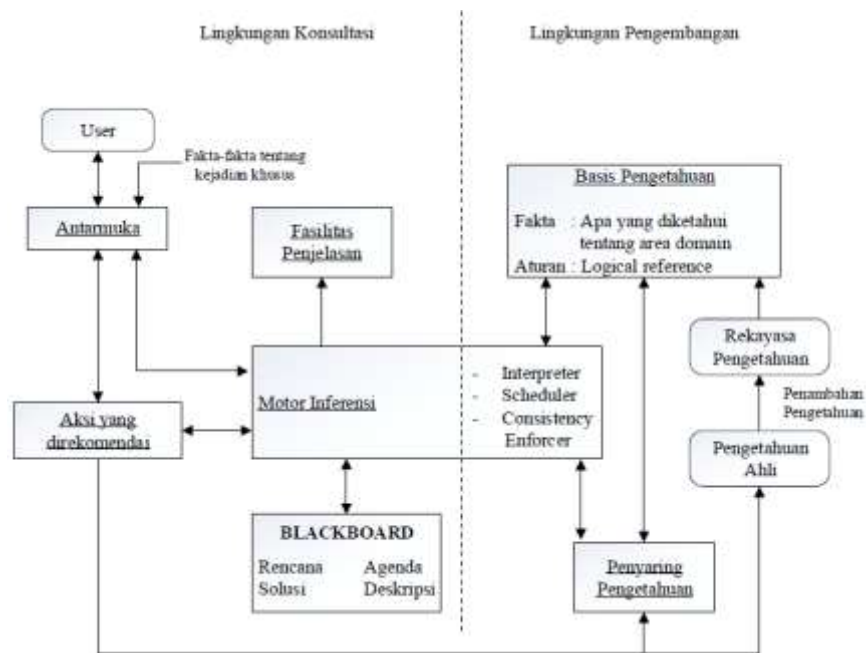
3. Menghubungkan ke perangkat lunak lain

Bentuk ini biasanya merupakan sistem pakar yang menghubungkan ke suatu paket program tertentu, misalnya DBMS.

4. Sistem Mengabdikan

Sistem pakar merupakan bagian dari komputer khusus yang dihubungkan dengan suatu fungsi tertentu. Misalnya sistem pakar yang digunakan untuk membantu menganalisis data radar.

Sistem pakar disusun oleh dua bagian utama, yaitu lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*) (Putra, 2016). Lingkungan pengembangan sistem pakar digunakan untuk memasukkan pengetahuan pakar ke dalam lingkungan sistem pakar, sedangkan lingkungan konsultasi digunakan oleh pengguna yang bukan pakar guna memperoleh pengetahuan pakar. Komponen-komponen sistem pakar dalam kedua bagian tersebut dapat dilihat dalam Gambar 2.5



Gambar 2.5. Struktur Sistem (Putra, 2016)

Komponen-komponen yang terdapat dalam sistem pakar antara lain, sebagai berikut:

### 1. Antarmuka Pengguna (*User Interface*)

Sistem pakar juga menyediakan komunikasi antara sistem dan pemakainya, yang disebut sebagai antarmuka. Antarmuka yang efektif dan ramah (*user-friendly*)

penting sekali terutama bagi pemakai yang tidak ahli dalam bidang yang diterapkan pada sistem pakar (Hartati & Iswanti, 2008).

## 2. Basis Pengetahuan

Basis pengetahuan mengandung pengetahuan untuk pemahaman, formulasi, dan penyelesaian masalah. Komponen sistem pakar ini disusun atas dua elemen dasar, yaitu fakta dan aturan. Fakta merupakan informasi tentang obyek dalam area permasalahan tertentu, sedangkan aturan merupakan informasi tentang cara bagaimana memperoleh fakta baru dari fakta yang telah diketahui (Putra, 2016).

## 3. Mesin Inferensi

Komponen ini mengandung mekanisme pola pikir dan penalaran yang digunakan oleh pakar dalam menyelesaikan suatu masalah. Mesin inferensi adalah program komputer yang memberikan metodologi untuk penalaran tentang informasi yang ada dalam basis pengetahuan dan dalam *workplace*, dan untuk memformulasikan kesimpulan (Putra, 2016).

## 4. Akuisisi Pengetahuan

Akuisisi pengetahuan adalah akumulasi, transfer, dan transformasi keahlian pemecahan masalah dari pakar atau sumber pengetahuan terdokumentasi ke program komputer, untuk membangun atau memperluas basis pengetahuan. Dalam tahap ini *knowledge engineer* berusaha menyerap pengetahuan untuk selanjutnya ditransfer ke dalam basis pengetahuan. Pengetahuan diperoleh dari pakar, dilengkapi dengan buku, basis data, laporan penelitian, dan pengalaman pemakai (Putra, 2016).

### 5. *Blackboard* (Tempat Kerja)

*Blackboard* adalah area kerja memori tempat pendeskripsian masalah yang diberikan oleh data input, digunakan juga untuk perekaman hipotesis dan keputusan sementara. Tiga tipe keputusan dapat direkam dalam *blackboard*, yaitu rencana (bagaimana mengatasi persoalan), agenda (tindakan potensial sebelum eksekusi), dan solusi (hipotesis kandidat dan arah tindakan alternatif yang telah dihasilkan sistem sampai dengan saat ini) (Putra, 2016).

### 6. Fasilitas Penjelasan Sistem

Fasilitas penjelasan sistem merupakan bagian dari sistem pakar yang memberikan penjelasan tentang bagaimana program dijalankan, apa yang harus dijelaskan kepada pemakai tentang suatu masalah, memberikan rekomendasi kepada pemakai, mengakomodasi kesalahan pemakai dan menjelaskan bagaimana suatu masalah terjadi (Putra, 2016).

### 7. Perbaikan Pengetahuan

Pakar manusia memiliki sistem perbaikan pengetahuan, yakni mereka dapat menganalisis pengetahuannya sendiri dan kegunaannya, belajar darinya, dan meningkatkannya untuk konsultasi mendatang (Putra, 2016).

## **2.2.6 Basis Pengetahuan**

Basis pengetahuan berisi pengetahuan-pengetahuan dalam menyelesaikan masalah, tentu di dalam domain tertentu. Ada dua bentuk pendekatan basis pengetahuan yang umum digunakan, yaitu:

### 1. Penalaran berbasis aturan (*Rule-Based Reasoning*)

Pengetahuan direpresentasikan dengan menggunakan aturan berbentuk :**IF-THEN**. Bentuk ini digunakan apabila kita memiliki sejumlah pengetahuan pakar pada suatu permasalahan tertentu, dan si pakar dapat menyelesaikan masalah tersebut secara berurutan. Disamping itu juga digunakan apabila dibutuhkan penjelasan tentang langkah-langkah pencapaian solusi.

### 2. Penalaran berbasis kasus (*Case-Based Reasoning*)

Basis pengetahuan akan berisi solusi-solusi yang telah dicapai sebelumnya, kemudian akan diturunkan suatu solusi untuk keadaan yang sekarang. Bentuk ini akan digunakan apabila *user* menginginkan untuk tahu lebih banyak lagi pada kasus-kasus yang hampir sama. Selain itu, bentuk ini juga digunakan apabila kita telah memiliki sejumlah situasi atau kasus tertentu dalam basis pengetahuan (Pratama, 2015).

#### **2.2.7 Mekanisme Inferensi**

Mekanisme Inferensi adalah bagian dari sistem pakar yang melakukan penalaran dengan menggunakan isi daftar aturan berdasarkan urutan dan pola tertentu. Selama proses konsultasi antar sistem dan pemakai, mekanisme inferensi menguji aturan satu demi satu sampai kondisi aturan itu benar. Dalam penalaran maju, aturan-aturan diuji satu demi satu dalam urutan tertentu. Urutan itu mungkin berupa urutan pemasukan aturan ke dalam basis aturan atau juga urutan lain yang ditentukan oleh pemakai. Saat setiap urutan diuji, sistem pakar akan mengevaluasi apakah kondisinya benar atau salah. Jika kondisinya benar, maka aturan itu disimpan kemudian aturan berikutnya diuji. Sebaliknya kondisinya salah, aturan itu tidak

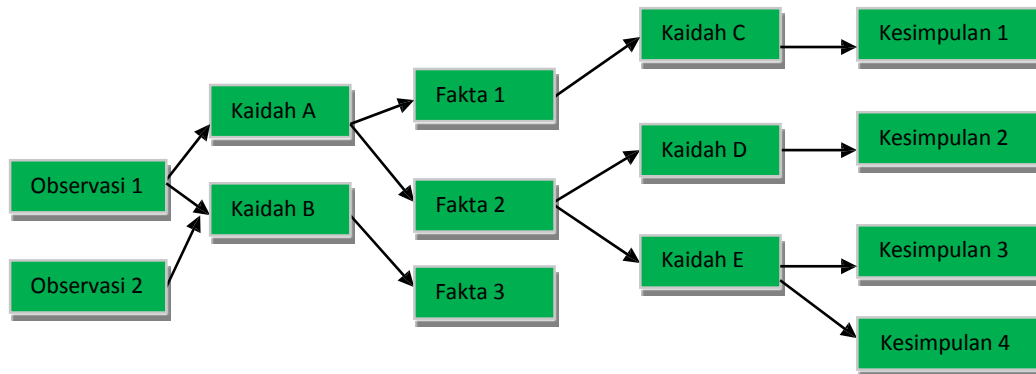


disimpan dan aturan berikutnya diuji. Proses ini akan berulang sampai seluruh basis aturan teruji dengan berbagai kondisi (Sinaga, 2014).

### **2.2.7.1 Metode *Forward Chaining***

Mesin inferensi dalam *forward chaining* menggunakan informasi yang ditentukan oleh *user* untuk memindahkan logika *and* dan *or* sampai ditentukannya objek. Bila mesin inferensi tidak dapat menentukan objek maka akan meminta informasi lain. Oleh karena itu, untuk mencapai objek harus memenuhi semua aturan. Salah satu metode yang dapat diterapkan dalam sistem pakar adalah metode *forward chaining*. *Forward chaining* disebut juga penalaran dari bawah ke atas. Suatu rantai yang dicari atau dilewati/dilintasi dari suatu permasalahan untuk memperoleh solusinya disebut dengan *forward chaining*. Cara lain menggambarkan *forward chaining* ini adalah dengan penalaran dari fakta menuju konklusi yang didapat dari fakta (Sinaga, 2014).

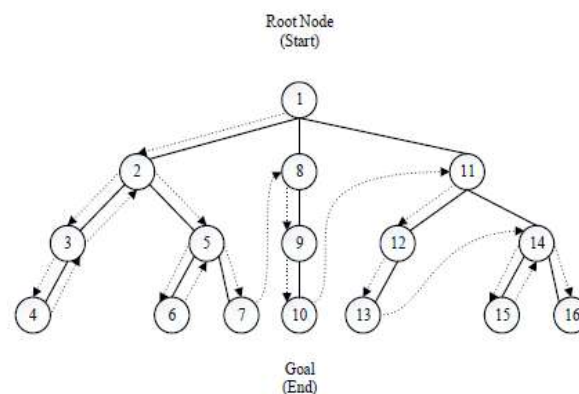
Runut maju berarti menggunakan himpunan aturan kondisi-aksi. Dalam metode ini, data digunakan untuk menentukan aturan mana yang akan dijalankan, kemudian aturan tersebut dijalankan. Mungkin proses menambahkan data ke memori kerja, Proses diulang sampai ditemukan suatu hasil. Pelacakan ke depan adalah pendekatan yang dimotori data (*data-driven*). Dalam pendekatan ini pelacakan dimulai dari informasi masukan, dan selanjutnya mencoba menggambarkan kesimpulan. Pelacakan ke depan mencari fakta yang sesuai dengan bagian **IF** dari aturan **IF-THEN** (Pratama, 2015). Proses *forward chaining* disajikan pada Gambar 2.6.



Gambar 2.6. Proses *Forward Chaining* (Pratama, 2015)

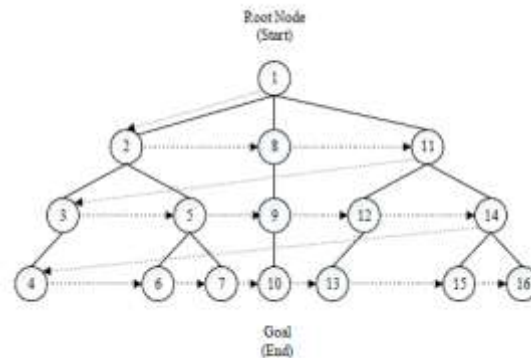
Metode inferensi tersebut dipengaruhi oleh tiga macam penelusuran, yaitu:

1. *Depth-first search*, melakukan penelusuran kaidah secara mendalam dari simpul akar bergerak menurun ke tingkat dalam yang berurutan. Diagram alir *depth-first search* disajikan pada Gambar 2.7



Gambar 2.7. Diagram Alir Teknik Penelusuran *Depth-first Search* (Pratama, 2015)

2. *Breadth-first search*, bergerak dari simpul akar, simpul yang ada pada setiap tingkat diuji sebelum pindah ke tingkat selanjutnya. Diagram alir *Breadth-first search* disajikan pada Gambar 2.8.



Gambar 2.8. Diagram Alir Teknik Penelusuran *Breadth-first Search*

(Pratama, 2015)

3. *Best-first search*, bekerja berdasarkan kombinasi kedua metode sebelumnya.

Menurut (Pratama, 2015) metode *forward chaining* merupakan teknik pencarian yang dimulai dengan fakta yang diketahui, kemudian mencocokkan fakta-fakta tersebut dengan bagian **IF** dari *rules IF-THEN*. Bila ada fakta yang cocok dengan bagian **IF**, maka *rule* tersebut dieksekusi. Bila sebuah *rule* dieksekusi, maka sebuah fakta baru (bagian **THEN**) ditambahkan ke dalam database. Setiap kali pencocokan berhenti bila tidak ada lagi *rule* yang bisa dieksekusi.

Contoh:

Misalkan diketahui sistem pakar menggunakan 5 buah *rule* berikut:

**R1 : IF (Y AND D) THEN Z**

**R2 : IF ( X AND B AND E) THEN Y**

**R3 : IF A THEN X**

**R4 : IF C THEN L**

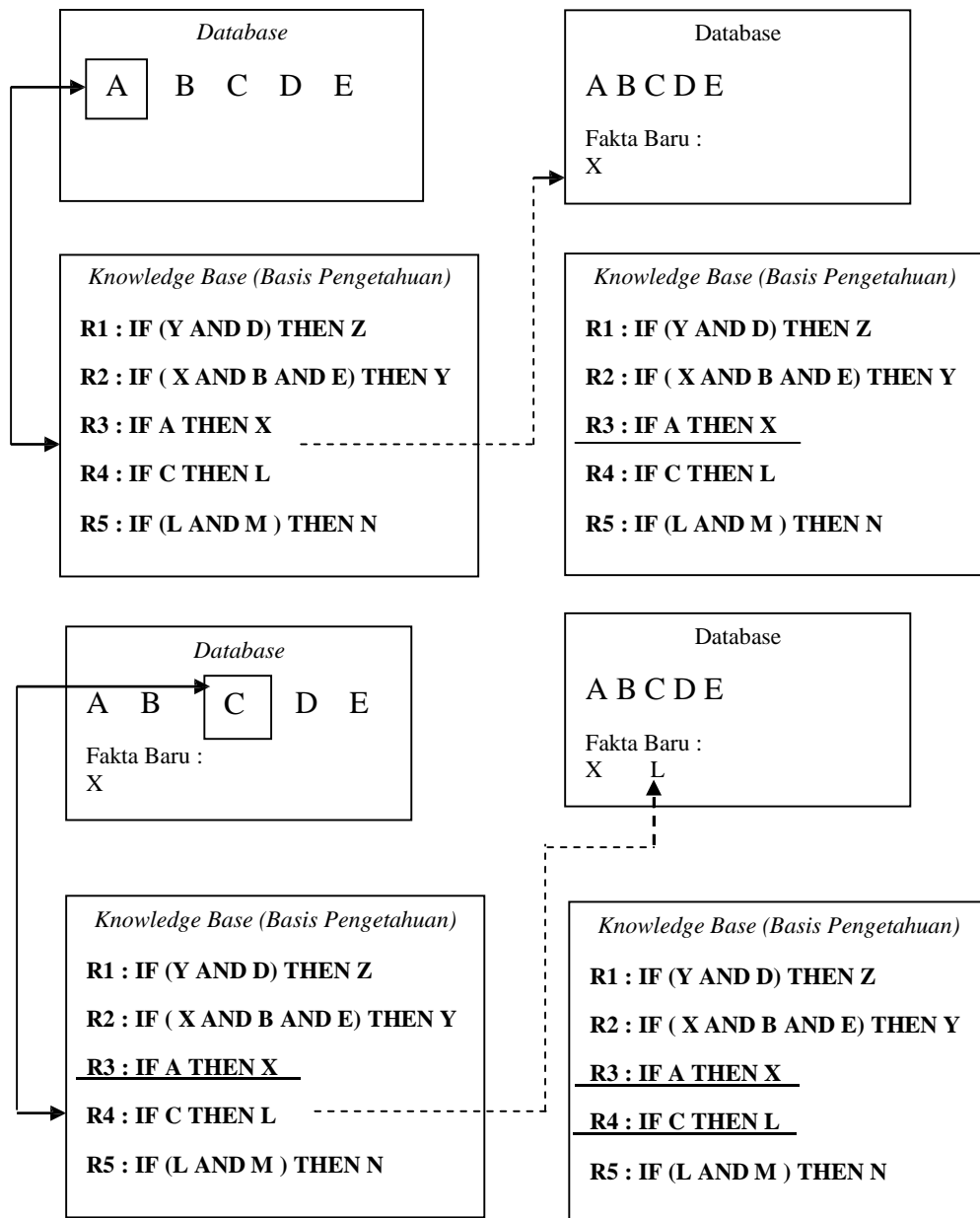
**R5 : IF (L AND M) THEN N**

Fakta-fakta : A, B, C, D, dan E bernilai benar

Goal : menentukan apakah Z bernilai benar atau salah

**Iterasi ke-1**

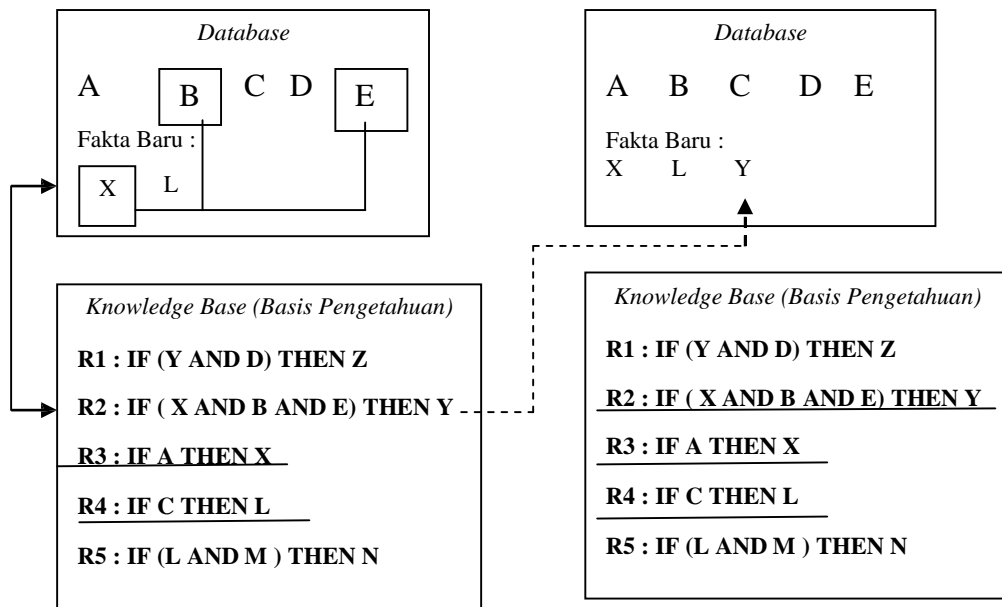
Iterasi ke-1 dapat dilihat pada Gambar 2.9.



Gambar 2.9. Iterasi ke-1

### Iterasi ke-2

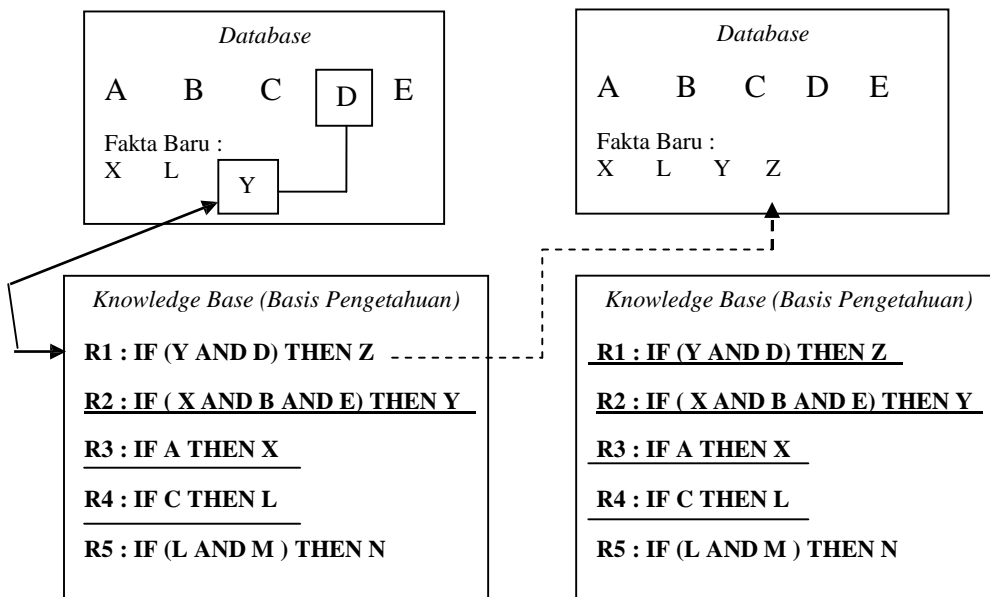
Iterasi ke-2 dapat dilihat pada Gambar 2.10.



Gambar 2.10. Iterasi ke-2

### Iterasi ke-3

Iterasi ke-3 dapat dilihat pada Gambar 2.11.



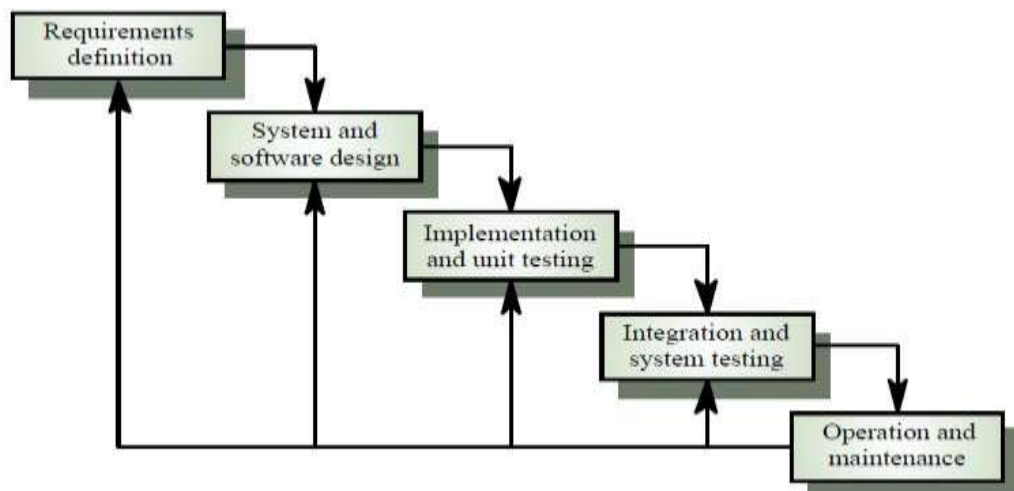
Gambar 2.11. Iterasi ke-3

Sampai di sini proses dihentikan karena sudah tidak ada lagi *rule* yang bisa dieksekusi. Hasil pencarian adalah Z bernilai benar (lihat database di bagian fakta baru).

### 2.3. Metode Pengembangan Sistem:

#### 2.3.1 Metode *Waterfall*

Metode *System Development Life Cycle* (SDLC) proses yang terus mengalir ke bawah seperti air terjun, sering juga disebut model sekuensial linier atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*). Model *Waterfall* disajikan pada Gambar 2.12 (Ruliansyah, 2016).



Gambar 2.12. Metode *Waterfall* (Ladjamuddin, 2006)

Tahapan pengembangan sistem dengan Metode *Waterfall* dijelaskan sebagai berikut:

1. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menganalisis kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

2. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

3. Pembuatan kode program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Pengujian fokus pada perangkat lunak secara fungsional dan meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung (*Support*) atau pemeliharaan (*Maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang ada, tapi tidak untuk membuat perangkat lunak baru.

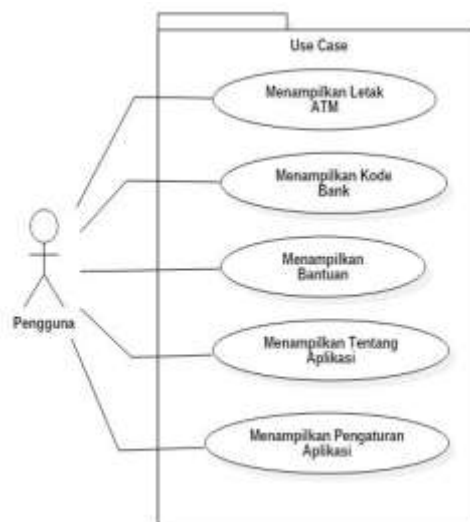
### ***2.3.2 Unified Modeling Language (UML)***

Menurut (Ruliansyah, 2016), UML adalah keluarga notasi grafis yang didukung oleh meta-model tunggal, yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek (OO). Definisi ini merupakan definisi yang sederhana. Pada kenyataannya, pendapat orang-orang tentang UML berbeda satu sama lain. Hal ini dikarenakan oleh sejarahnya sendiri dan oleh perbedaan persepsi tentang apa yang membuat sebuah proses rancang-bangun perangkat lunak efektif. UML lahir dari penggabungan banyak bahasa permodelan grafis berorientasi objek yang berkembang pesat pada akhir 1980-an dan awal 1990-an. UML dibuat oleh Grady Booch, James Rumbaugh, dan Ivar Jacobson di bawah bendera *Rational Software Corp*. UML menyediakan notasi-notasi yang membantu memodelkan sistem dari berbagai perspektif. UML tidak hanya digunakan dalam pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan. UML dideskripsikan oleh beberapa diagram, yaitu sebagai berikut:



## 1. Use Case Diagram

*Use caseDiagram* digunakan untuk menggambarkan sistem dari sudut pandang pengguna sistem tersebut (*user*), sehingga pembuatan *use case diagram* lebih dititik beratkan pada fungsionalitas yang ada pada sistem, bukan berdasarkan alur atau urutan kejadian. Sebuah *use case diagram* merepresentasikan sebuah interaksi antara aktor dengan sistem. Contoh *use case* dapat dilihat pada Gambar 2.13.

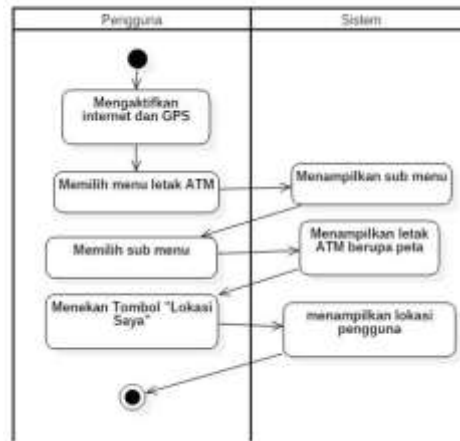


Gambar 2.13. Contoh *use case diagram* (Erika, 2015)

## 2. Activity Diagram

Diagram aktivitas atau *Activity Diagram* menggambarkan aliran fungsionalitas sistem. Pada tahap pemodelan bisnis, diagram aktivitas dapat digunakan untuk menunjukkan aliran kerja bisnis (*business work flow*) serta dapat digunakan untuk menggambarkan aliran kejadian (*flow of events*) dalam *use case*. Aktivitas dalam diagram dipresentasikan dengan bentuk bujur sangkar bersudut tidak lancip, yang di dalamnya berisi langkah-langkah apa saja yang terjadi dalam aliran kerja. Ada kedudukan awal (*start state*) yang menunjukkan

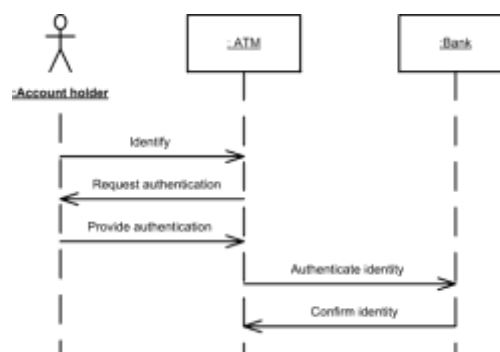
dimulainya aliran kerja, dan sebuah kedudukan akhir (*end state*) yang menunjukkan akhir diagram, titik keputusan dipresentasikan dengan *diamond*. Contoh diagram aktivitas dapat dilihat pada Gambar 2.14 (Ruliansyah, 2016).



Gambar 2.14. Contoh diagram aktivitas (Erika, 2015).

### 3. Sequence Diagram

Diagram sekuensial atau *sequence diagram* digunakan untuk menunjukkan aliran fungsionalitas dalam *use case*. Misalkan, pada *use case* “menarik uang” mempunyai beberapa kemungkinan, seperti penarikan uang secara normal, percobaan penarikan uang tanpa kecukupan ketersediaan dana, penarikan dengan penggunaan PIN yang salah, dan lainnya. Contoh diagram sekuensial dapat dilihat pada Gambar 2.15 (Ruliansyah, 2016)



Gambar 2.15. Contoh diagram sekuensial (Ruliansyah, 2016)

#### 4. Class Diagram

Class adalah sebuah spesifikasi yang akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi). Class Diagram menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain seperti pewarisan, asosiasi, dan lain-lain.

Class memiliki tiga area pokok:

1. Nama (Class Name)
2. Atribut
3. Metode (Operations)

Pada UML, class digambarkan dengan segi empat yang dibagi beberapa bagian. Bagian atas merupakan nama dari *class*. Bagian yang tengah merupakan struktur dari *class* (atribut) dan bagian bawah merupakan sifat dari *class* (metode/operasi).

Atribut dan metode dapat memiliki salah satu sifat berikut:

1. *Private*, tidak dapat dipanggil dari luar *class* yang bersangkutan.
2. *Protected*, hanya dapat dipanggil oleh *class* yang bersangkutan dan *class* lain yang mewarisinya.
3. *Public*, dapat dipanggil oleh *class* lain (Putra, 2016).

## 2.4. Teknik Pengujian Perangkat Lunak

### 2.4.1. *Equivalence Partitioning*

*Equivalence Partitioning* (EP) merupakan metode *Black Box testing* yang membagi domain masukan dari program ke dalam kelas-kelas sehingga *test case* dapat diperoleh. *Equivalence Partitioning* berusaha untuk mendefinisikan kasus uji yang menemukan sejumlah jenis kesalahan, dan mengurangi jumlah kasus uji yang harus dibuat. Kasus uji yang didesain untuk *Equivalence Partitioning* berdasarkan pada evaluasi dari kelas ekuivalensi untuk kondisi masukan yang menggambarkan kumpulan keadaan yang valid atau tidak. Kondisi masukan dapat berupa spesifikasi nilai numerik, kisaran nilai, kumpulan nilai yang berhubungan atau kondisi boolean (Erika, 2015).

### 2.4.2. Probabilitas Klasik

Menurut (Putra, 2016) probabilitas merupakan suatu cara kuantitatif yang berhubungan dengan ketidakpastian yang telah ada. Probabilitas klasik disebut juga *a priori probability* karena berhubungan dengan suatu permainan (*games*) atau sistem. Seperti yang telah disebutkan sebelumnya, istilah *a priori* berarti “sebelum”. Probabilitas ini dianggap sebagai suatu jenis permainan seperti pelemparan dadu, permainan kartu, dan pelemparan koin.

Rumus umum untuk probabilitas klasik di definisikan sebagai peluang  $P(A)$  dengan  $n$  adalah banyaknya kejadian,  $n(A)$  merupakan banyaknya hasil mendapatkan  $A$ .

Frekuensi relative terjadinya  $A$  adalah  $\frac{n(A)}{n}$  maka:

$$P(A) = \frac{n(A)}{n} \dots\dots\dots(1)$$

Probabilitas klasik ini digunakan untuk mendapatkan peluang kemungkinan

penyakit, sehingga untuk menghitung persentase penyakit adalah :

$$\text{Persentase (A)} = P(A) \times 100\% \dots\dots\dots (2)$$

Tabel persentase penyakit disajikan pada Tabel 2.1.

Tabel 2.1 Tabel Kemungkinan Penyakit

<b>Kondisi</b>	<b>Persentase</b>
Pasti Tidak	<10%
Tidak Tahu	10-19%
Hampir Mungkin	20 – 39%
Mungkin	40 – 59%
Kemungkinan Besar	60 - 79%
Hampir Pasti	80 – 99%
Pasti	100%

### **2.4.3 Skala *Likert***

Menurut (Agustianti, 2016), Metode ini merupakan metode penskalaan pernyataan sikap yang menggunakan

distribusi respons sebagai dasar penentuan nilai skalanya. Nilai skala setiap pernyataan tidak ditentukan oleh derajat favourable masing-masing, akan tetapi ditentukan oleh distribusi respons setuju dan tidak setuju dari sekelompok responden yang bertindak sebagai kelompok uji coba.

Skala Likert, yaitu skala yang berisi lima tingkat preferensi jawaban dengan

pilihan sebagai berikut : 1= sangat setuju; 2 = tidak setuju; 3 = ragu-ragu atau

netral; 4 = setuju; 5 = sangat setuju. Persentase penilaian berdasarkan kriteria skala likert akan diperoleh dengan rumus aritmatika mean, yaitu:

$$P = \frac{Xi}{n \times N} \times 100 \% \quad \dots\dots\dots (3)$$

Keterangan :

P = Persentase pernyataan

Xi = Nilai kuantitatif total

n = Jumlah responden

N = Nilai kategori pernyataan terbaik

Selanjutnya, penentuan interval per kategori digunakan rumus sebagai berikut :

$$I = \frac{100\%}{K} \quad \dots\dots\dots (4)$$

Keterangan :

I = Interval

K = Kategori interval

## **BAB III**

### **METODE PENELITIAN**

#### **3.1 Waktu Dan Tempat Penelitian**

Penelitian ini dilakukan di Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam, Laboratorium Penyakit Hama Tanaman Jurusan Agroteknologi Fakultas Pertanian Universitas Lampung dan Balitro Natar. Waktu penelitian dilakukan pada bulan November 2016 sampai dengan dengan selesai.

#### **3.2 Perangkat Penelitian**

Alat pendukung yang digunakan dalam penelitian ini adalah sebagai berikut:

##### **A. Perangkat Keras**

Nootebook dengan spesifikasi:

Processor : Intel(R) Celeron(R) CPU 1007U @ 1.500GHz (2 CPUs),

RAM : 4096 MB

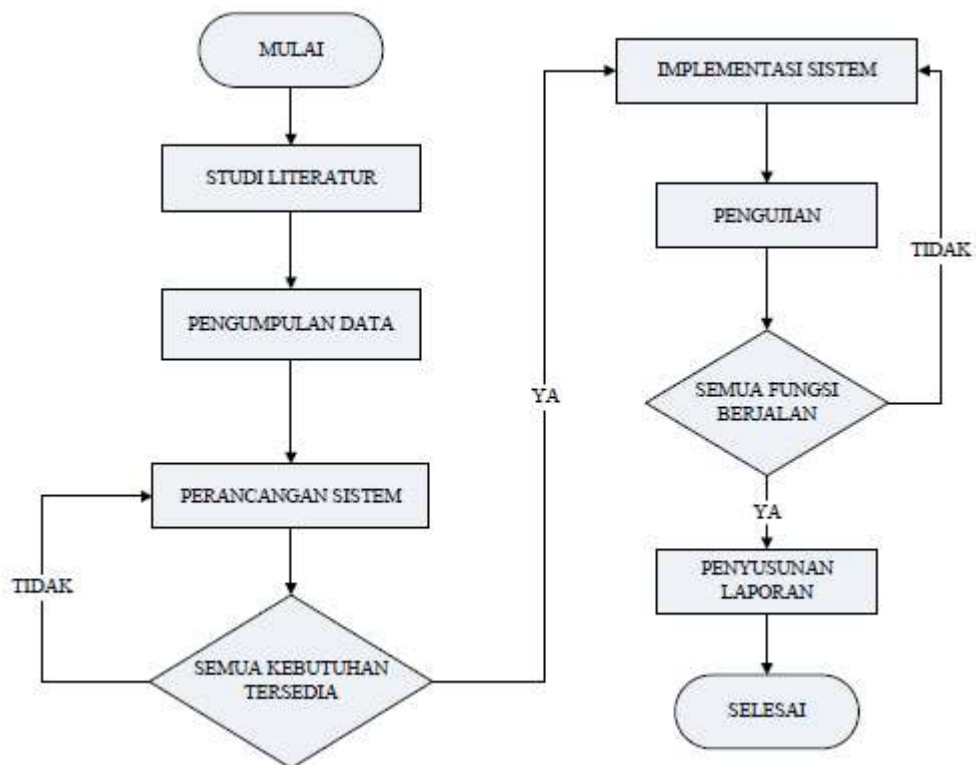
Hardisk : 500 GB

##### **B. Perangkat Lunak**

- Sistem Operasi Windows 7 64bit
- *Text Editor* Notepad++
- Adobe Photoshop CS 6
- Android Studio

### 3.3 Tahapan Penelitian

Tahapan penelitian yaitu tahapan yang akan dilakukan peneliti untuk mempermudah dalam melakukan penelitian. Desain penelitian sistem pakar penyakit pada tanaman lada dengan metode *forward chaining* digambarkan pada Gambar 3.1



Gambar 3.1. Diagram Alir Penelitian

#### 3.1.1 Studi Literatur

Studi literatur dilakukan dengan cara mempelajari aspek-aspek yang berkaitan dengan penelitian ini, diantaranya adalah mencari jenis-jenis penyakit tanaman lada, gejala-gejala dari penyakit tersebut dan solusinya, mempelajari metode *forward chaining*. Data-data yang digunakan dalam studi literatur didapat dengan



cara mengumpulkan jurnal, penelusuran internet dan buku yang berkaitan dengan topik.

### **3.3.2 Pengumpulan Data**

Tahapan ini mengumpulkan data-data terkait penyakit pada tanaman lada. Data yang dikumpulkan berasal dari hasil konsultasi dengan pakar dan literatur-literatur tentang penyakit tanaman lada. Data-data yang dikumpulkan disusun menjadi basis aturan yang akan digunakan dalam sistem pakar.

### **3.3.3 Perancangan Sistem**

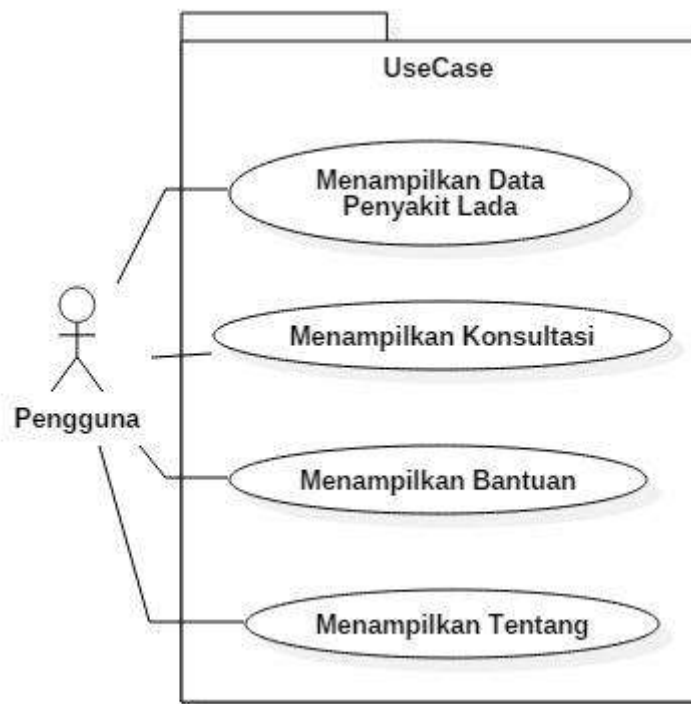
Perancangan sistem adalah tahap setelah analisis dari siklus pengembangan sistem. Perancangan sistem di sini berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi. Perancangan sistem menentukan bagaimana suatu sistem akan menyelesaikan apa yang mesti diselesaikan. Tahap ini termasuk mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem sehingga setelah dilakukan instalasi akan benar-benar memuaskan rancang bangun yang telah ditetapkan pada akhir tahap analisis sistem.

#### **3.3.3.1 Perancangan UML (*United Modelling Language*)**

Pemodelan (*modeling*) adalah tahap merancang perangkat lunak sebelum melakukan tahap pembuatan program (koding). Pada penelitian ini, perancangan sistem dilakukan dengan memodelkan permasalahan dalam bentuk diagram-diagram UML sebagai berikut:

## 1. Use Case Diagram

*Use case* Diagram dibawah ini menggambarkan sistem dari sudut pandang pengguna sistem tersebut (*user*), sehingga pembuatan *use case* diagram ini lebih dititik beratkan pada fungsionalitas yang ada pada sistem, bukan berdasarkan alur atau urutan kejadian. Pada aplikasi ini pengguna dapat melakukan 4 interaksi antara lain Data Penyakit, Konsultasi, Bantuan, Tentang aplikasi. *Use case* diagram aplikasi Sistem Pakar Diagnosa Penyakit Tanman Lada dapat dilihat pada Gambar 3.2.



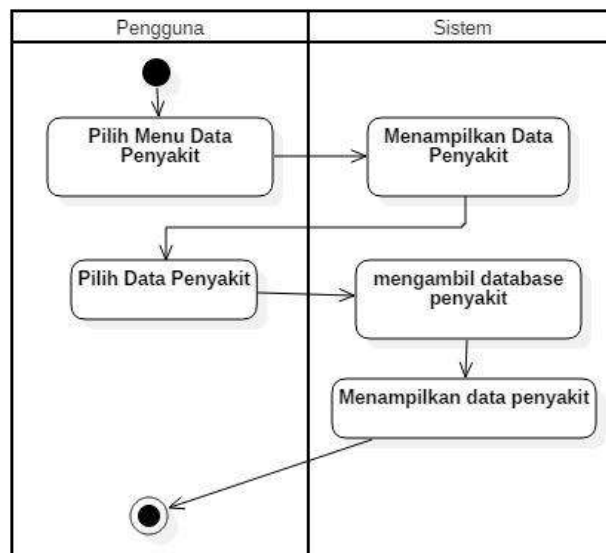
Gambar 3.2. *Use Case Diagram*

## 2. Activity Diagram

*Activity* diagram menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam satu operasi sehingga dapat juga untuk aktivitas lainnya. Diagram ini sangat mirip dengan *flowchart* karena memodelkan *workflow* dari satu aktivitas ke aktivitas lainnya atau dari aktivitas ke status. Pada aplikasi Sistem Pakar Diagnosa Penyakit Tanaman Lada terdapat 4 (empat) *activity* diagram yaitu sebagai berikut:

### a. Activity Diagram Data Penyakit

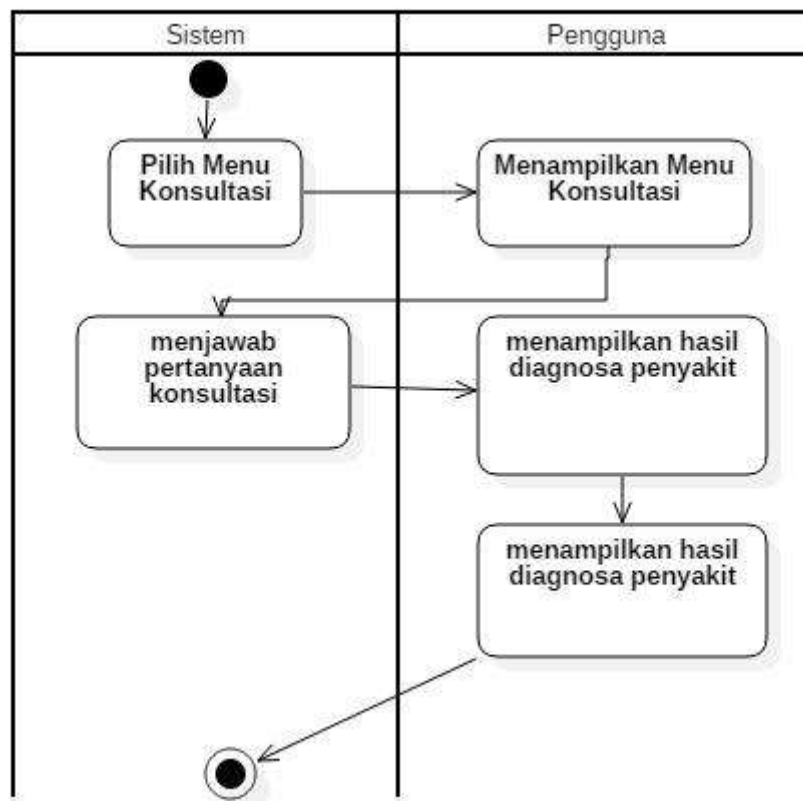
*Activity* diagram Data Penyakit dimulai dengan pengguna memilih menu “Data Penyakit”, kemudian sistem akan menampilkan daftar penyakit pada tanaman lada. Pengguna dapat mengetikkan nama Penyakit yang akan dicari, jika sudah pilih *button* “cari”. *Activity* diagram Data Penyakit dilihat pada Gambar 3.3.



Gambar 3.3. Activity Diagram Data Penyakit

b. *Activity* Diagram Konsultasi

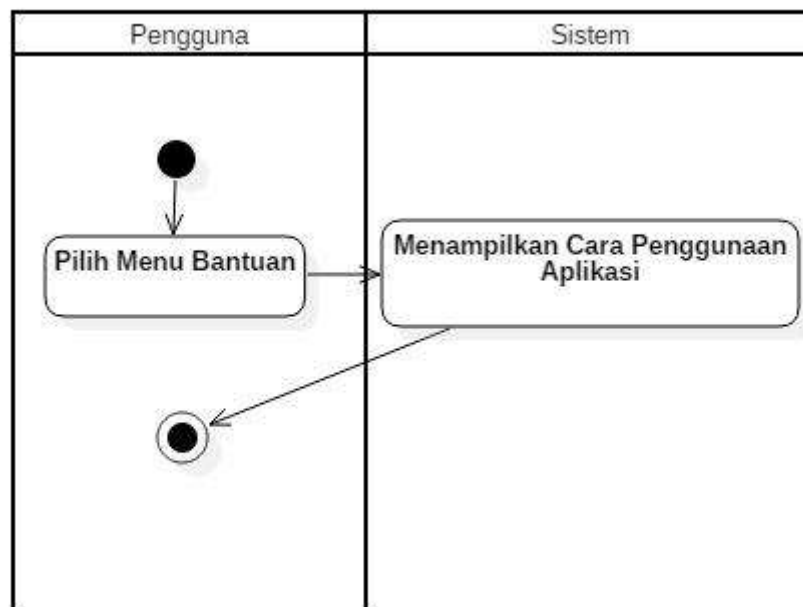
*Activity* diagram Konsultasi dimulai dengan pengguna memilih menu Konsultasi, selanjutnya sistem akan menampilkan kolom konsultasi gejala dan pengguna menjawab pertanyaan gejala penyakit pada tanaman lada. Sistem mencari database penyakit lada dan menampilkan hasil diagnosa. *Activity* diagram Konsultasi disajikan pada Gambar 3.4.



Gambar 3.4. *Activity* Diagram Konsultasi

c. *Activity Diagram Bantuan*

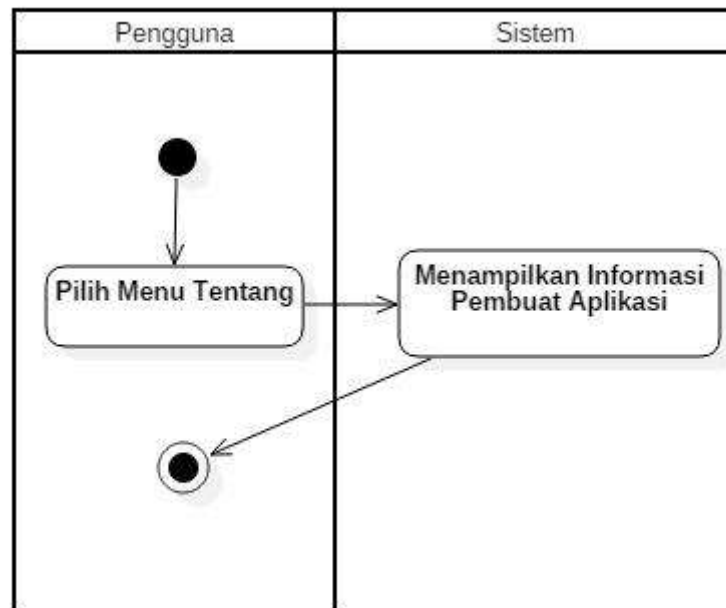
*Activity diagram bantuan* dimulai dengan pengguna memilih menu “Bantuan”, kemudian sistem akan menampilkan informasi yang berkaitan dengan cara penggunaan aplikasi Sistem Pakar Penyakit Tanaman Lada. *Activity diagram bantuan* disajikan pada Gambar 3.5.



Gambar 3.5. *Activity Diagram Bantuan*

d. *Activity Diagram Tentang*

*Activity diagram tentang aplikasi* dimulai dengan pengguna memilih menu “Tentang”, kemudian sistem menampilkan informasi mengenai aplikasi Sistem Pakar Penyakit Tanaman Lada. *Activity diagram tentang* dapat dilihat pada Gambar 3.6.



Gambar 3.6. *Activity Diagram Tentang*

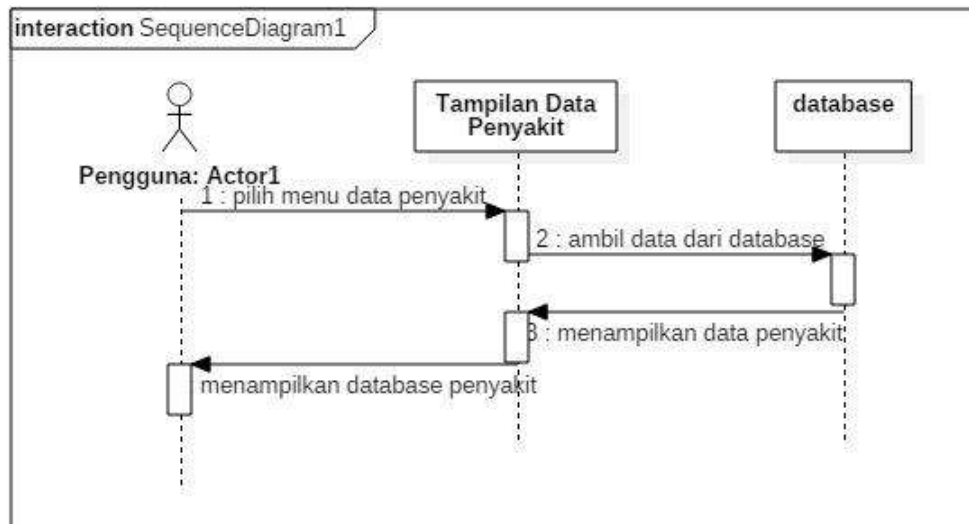
### 3. *Sequence Diagram*

Menggambarkan interaksi antara sejumlah objek dalam urutan waktu. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi sistem. Pada aplikasi Sistem Pakar Diagnosa Penyakit Tanaman Lada terdapat 4 (empat) *sequence diagram* yaitu sebagai berikut:

#### a. *Sequence Diagram Data Penyakit*

Untuk dapat memilih menu “Data Penyakit ” pengguna harus memilih aplikasi Sistem Pakar Penyakit Tanaman Lada, kemudian otomatis akan menuju halaman *splash screen*, selanjutnya akan muncul menu utama aplikasi, dan pengguna dapat memilih menu “Data Penyakit”, selanjutnya sistem akan menampilkan daftar penyakit tanaman lada. Kemudian pengguna dapat

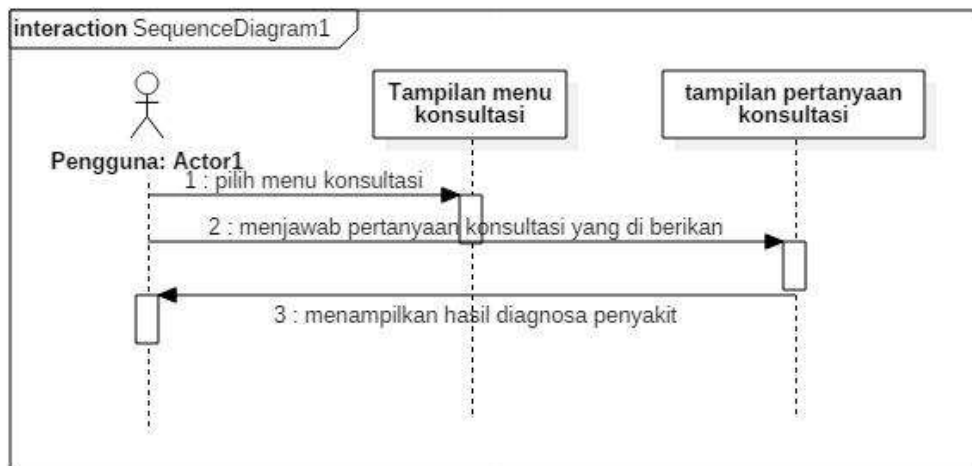
mengetikkan nama penyakit yang akan dicari lalu sistem akan otomatis mencari data penyakit yang telah disimpan di database kemudian menampilkan data penyakit lada sesuai perintah yang di ketikkan sebelumnya. Untuk lebih jelasnya *sequence* diagram Data Penyakit dapat dilihat pada Gambar 3.7.



Gambar 3.7. *Sequence* Diagram Data Penyakit

#### b. *Sequence* Diagram Konsultasi

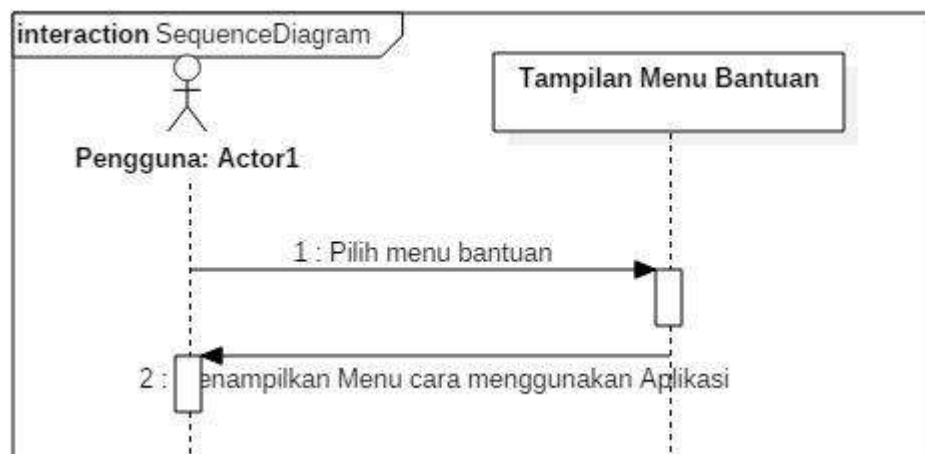
Untuk dapat memilih menu “Konsultasi” pengguna harus memilih aplikasi Sistem Pakar Penyakit Tanaman Lada, kemudian otomatis akan menuju halaman *splash screen*, selanjutnya akan muncul menu utama aplikasi, dan pengguna dapat memilih menu “Konsultasi”, maka sistem akan menampilkan pertanyaan terkait gejala pada tanaman lada, dan menjawab pertanyaan gejala penyakit pada tanaman lada dan sistem akan memproses database kemudian menampilkan hasil diagnosa penyakit, gejala dan solusi. *Sequence* diagram Konsultasi dapat dilihat pada Gambar 3.8.



Gambar 3.8. *Sequence* Diagram Konsultasi

c. *Sequence* Diagram Bantuan

Untuk dapat memilih menu “Bantuan” pengguna harus memilih aplikasi Sistem Pakar Penyakit Tanaman Lada, selanjutnya pengguna dapat memilih menu “Bantuan” untuk mengetahui informasi mengenai cara penggunaan aplikasi, maka sistem akan menampilkan informasi tersebut. *Sequence* diagram bantuan disajikan pada Gambar 3.9.

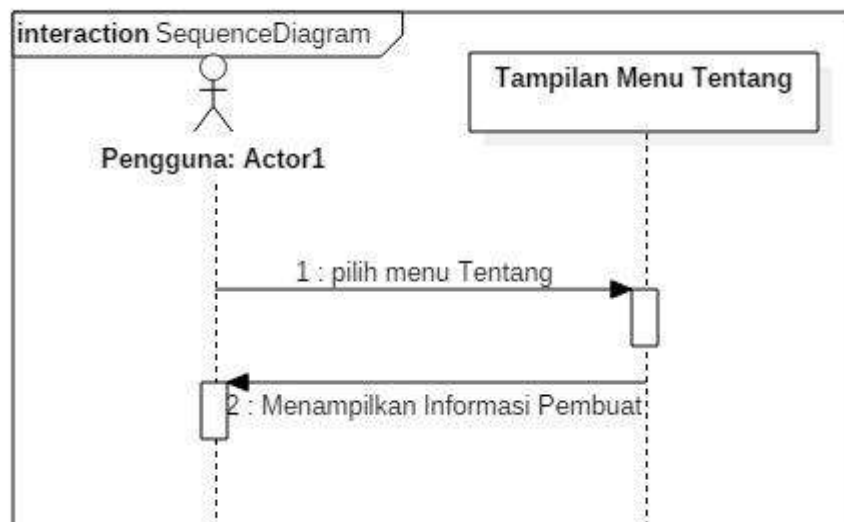


Gambar 3.9. *Sequence* Diagram Bantuan



#### d. *Sequence* Diagram Tentang

Untuk dapat memilih menu “Tentang” pengguna harus memilih aplikasi Sistem Pakar Penyakit Tanaman Lada, selanjutnya pengguna dapat memilih menu “Tentang” untuk mengetahui mengenai informasi aplikasi, maka sistem akan menampilkan informasi tentang aplikasi Sistem Pakar Diagnosa Penyakit Tanaman Lada. *Sequence* diagram tentang aplikasi disajikan pada Gambar 3.10.



Gambar 3.10. *Sequence* Diagram Tentang Aplikasi

#### 3.3.3.2 Perancangan Antarmuka

Perancangan antarmuka merupakan proses penggambaran bagaimana sebuah tampilan (*interface*) sistem dibentuk. Aplikasi Sistem Pakar Penyakit Tanaman Lada dirancang dengan tampilan yang *user friendly*, sehingga diharapkan dapat mempermudah pengguna dalam menggunakan aplikasi ini. Dalam Aplikasi ini terdapat beberapa *layout* atau form antara lain:

### 1. *Layout Splash Screen*

*Splash Screen* adalah form yang ditampilkan diawal ketika aplikasi/program dijalankan. Aplikasi Sistem Pakar Penyakit Tanaman Lada menggunakan *splash screen* yang muncul sepersekian detik pada saat pertama membuka aplikasi. *Splash screen* di sini dimaksudkan sebagai estetika untuk menunjukan identitas aplikasi saja, tanpa fungsi lainnya. Perancangan *layout splash screen* aplikasi dapat dilihat pada Gambar 3.11.



Gambar 3.11. *Layout Splash Screen*

### 2. *Layout Menu Utama*

Menu utama berisikan menu-menu pilihan yang dapat digunakan oleh pengguna pada aplikasi. Menu yang terdapat pada menu utama yaitu: menu Data Penyakit, menu Konsultasi, menu Bantuan, dan menu Tentang aplikasi. Perancangan *layout* menu utama aplikasi dapat dilihat pada Gambar 3.12.



Gambar 3.12. *Layout* Menu Utama



Gambar 3.13. *Design Layout*  
Data Penyakit

### 3. *Layout* Menu Data Penyakit

Pada menu Data Penyakit ketika pengguna memilih menu ini, pengguna dapat langsung melihat daftar nama penyakit tanaman lada dan mencari jenis penyakit tanaman lada, dengan menekan kolom *teks search* pengguna harus memasukkan nama penyakit kemudian tekan tombol “cari” untuk melakukan perintah. Perancangan *layout* menu data penyakit aplikasi dapat dilihat pada Gambar 3.13.

### 4. *Layout* Menu Konsultasi

Ketika pengguna memilih menu “Konsultasi”, maka akan muncul sub menu dari Konsultasi. Sub-sub menu pada *layout* “Konsultasi” memiliki rincian sebagai berikut:

a. *Layout* Sub Menu Konsultasi

Pengguna/pembudidaya menjawab pertanyaan yang disediakan, kemudian jawaban pengguna disimpan untuk di proses. Data digunakan untuk menampilkan pertanyaan yang terkait dengan jawaban yang diberikan pengguna dan menyimpan aturan analisa kemungkinan penyakit yang diderita tanaman lada pengguna/pembudidaya. Pengguna tinggal menekan *button* “YA” atau “TIDAK” untuk menjawab beberapa pertanyaan yang diajukan

oleh sistem. Perancangan *layout* sub menu konsultasi gejala dapat dilihat pada Gambar 3.14.



Gambar 3.14. *Design Layout* Sub Menu Konsultasi Gejala



Gambar 3.15. *Design Layout* Sub Menu Hasil Diagnosa

b. *Layout* Sub Menu Hasil Diagnosa

Pada sub menu ini menampilkan hasil diagnosa penyakit yang diderita dan memberikan informasi tentang penyakit tersebut, serta memberikan solusi penanganannya. Pengguna dapat melihat apa penyakit yang di derita tanaman lada dan cara penanganannya secara rinci dengan menekan tombol Detail. Perancangan *layout* Sub Menu Hasil Diagnosa Penyakit dan penjelasan secara rinci tentang solusi penanganannya dapat dilihat pada Gambar 3.15 dan 3.16.



Gambar 3.16. *Design Layout* Menu Bantuan

5. *Layout* Menu Bantuan

Ketika pengguna memilih menu “Bantuan” pengguna akan melihat informasi mengenai cara penggunaan aplikasi Sistem Pakar Penyakit Pada Tanaman Lada. Perancangan *layout* menu Bantuan dapat dilihat pada Gambar 3.17.



Gambar 3.17. *Design Layout* Hasil Diagnosa Penyakit (Penjelasan Penyakit)

#### 6. *Layout* Menu Tentang

Ketika pengguna memilih menu “Tentang” pengguna akan melihat informasi mengenai aplikasi Sistem Pakar Penyakit Pada Tanaman Lada. Pada layout ini terdapat 3 (tiga) *Button* yang dapat pengguna gunakan untuk memberikan kritik dan saran melalui email developer, google play dan fanpage developer. Perancangan *layout* menu Tentang dapat dilihat pada Gambar 3.18



Gambar 3.18. *Design Layout* Tentang

### 3.4 Metode Pengujian Sistem

Pengujian sistem dimaksudkan untuk menguji semua elemen–elemen perangkat lunak yang dibuat apakah sudah sesuai dengan yang diharapkan. Pendekatan kasus uji dalam penelitian ini adalah pengujian black box dengan metode *Equivalence Partitioning* (EP). Pengujian ini dilakukan dengan membagi domain masukan dari program ke dalam kelas-kelas sehingga test case dapat diperoleh. EP berusaha untuk mendefinisikan kasus uji yang menemukan sejumlah jenis kesalahan, dan mengurangi jumlah kasus uji yang harus dibuat. EP didasarkan pada premis masukan dan keluaran dari suatu komponen yang dipartisi ke dalam kelas-kelas, menurut spesifikasi dari komponen tersebut, yang akan diperlakukan sama (ekuivalen) oleh komponen tersebut. Pada pengujian ini harus diyakinkan bahwa masukan yang sama akan menghasilkan

respon yang sama pula. Alasan menggunakan metode EP pada pengujian aplikasi Sistem Pakar Penyakit Pada Tanaman Lada ini adalah karena metode ini dapat digunakan untuk mencari kesalahan pada fungsi yang diberikan ke aplikasi dan dapat mengetahui kesalahan pada interface aplikasi sehingga dapat mengurangi masalah terhadap nilai masukan.

Berikut ini merupakan rancangan daftar pengujian yang disajikan pada Tabel 3.1.

Tabel 3.1 Tabel Daftar Pengujian

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang Diharapkan
1	Versi Android	Pengujian kompatibilitas versi <i>operatif system</i> android	Pengujian pada android versi 4.1 (Jelly Bean)	Kompatibel dengan android versi 4.1 (Jelly Bean)
			Pengujian pada android versi 4.4 (Kit-Kat)	Kompatibel dengan android versi android versi 4.4 (Kit-Kat)
			Pengujian pada android Versi 5.0 (Lollipop)	Kompatibel dengan android Versi 5.0 (Lollipop)
2	Resolusi Layar dan Intebstas Layar	Pengujian Resolusi Layar dan Densitas Layar pada android	Pengujian pada android dengan resolusi 4 inch	Tampilan terlihat baik pada android dengan resolusi 4 inch
			Pengujian pada android dengan resolusi 4,5 inch	Tampilan terlihat sesuai atau baik pada android dengan resolusi 4,5 inch
			Pengujian pada android dengan resolusi 4,7 inch	Tampilan terlihat sesuai atau baik pada android dengan resolusi 4,7 inch



Tabel 3.1 Tabel Daftar Pengujian (Lanjutan)

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang Diharapkan
			Pengujian pada android dengan resolusi 5 inch	Tampilan terlihat sesuai atau baik pada android dengan resolusi 5 inch
			Pengujian pada android dengan resolusi 5,5 inch	Tampilan terlihat sesuai atau baik pada android dengan resolusi 5,5inch
3	<i>User Interface</i>	Pengujian pada icon PAKAR LADA	Klik icon Pakar Lada pada perangkat android pengguna	Menampilkan layout splash screen
		Pengujian pada menu utama PAKAR LADA	Klik tombol menu "Data Penyakit"	Klik tombol menu "Data Penyakit"
			Klik tombol menu "Konsultasi"	Menampilkan layout Konsultasi
			Klik tombol menu "Bantuan"	Menampilkan layout Bantuan
			Klik tombol menu "Tentang"	Menampilkan layout Tentang
4	Fungsi layout Data Penyakit	Pengujian Pencarian database Penyakit Lada	Klik tombol menu "Cari Penyakit"	Menampilkan kolom teks search dan layout masukan
			Klik tombol "Cari"	Menampilkan daftar nama penyakit

Tabel 3.1 Tabel Daftar Pengujian (Lanjutan)

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang Diharapkan
5	Fungsi layout Konsultasi	Pengujian Pada layout Konsultasi	Klik tombol menu "Konsultasi"	Menampilkan Layout Pertanyaan yang disediakan sistem
			Klik Sub Menu Hasil Diagnosa Penyakit	Menampilkan hasil diagnosa penyakit yang diderita tanaman lada dan memberikan informasi tentang penyakit tersebut, serta memberikan solusi penanganannya.
6	Fungsi pada menu Bantuan	Pengujian pada menu Bantuan	Klik tombol "Bantuan"	Menampilkan Layout mengenai cara menggunakan aplikasi
7	Fungsi pada menu Tentang	Pengujian pada menu Tentang	Klik icon "Email"	Menampilkan halaman sosial media pengembang
			Klik icon "Sosial Media"	Menampilkan halaman sosial media pengembang
			Klik icon "Google Play"	Masuk ke akun halaman google play untuk aplikasi Pakar Lada

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1. Kesimpulan**

Berdasarkan hasil penelitian yang telah dilakukan dapat disimpulkan beberapa hal sebagai berikut:

1. Sistem Pakar Lada masuk dalam kategori “Sangat Baik” dengan total rata-rata yang diperoleh pada variabel Sistem Pakar Lada sebesar 85,75%.
2. Telah berhasil dibangun Aplikasi PAKAR LADA yang dibuat sebagai sarana penunjang untuk pembudidaya tanaman lada dalam konsultasi masalah penyakit lada.
3. Sistem pakar yang dibangun dapat memberikan hasil diagnosa berdasarkan fakta-fakta yang telah diberikan.
4. Aplikasi dapat membantu pembudidaya dalam mendiagnosa penyakit lada dan memberikan solusi terkait penyakit yang diderita layaknya seorang pakar.
5. Berdasarkan dari hasil data pengujian *Equivalence Partitioning*, hasil penilaian pengguna didapat bahwa aplikasi dapat mendiagnosa penyakit dengan sangat baik.

#### **5.2. Saran**

Beberapa saran yang diberikan setelah dilakukan penelitian ini untuk pengembangan lebih lanjut adalah sebagai berikut:

1. Penambahan data-data jenis penyakit, dan gejala serta perbaikan dan penyempurnaan pertanyaan gejala secara lebih detail.
2. Pengembangan selanjutnya dapat menambahkan data hama pada tanaman lada.
3. Pengembangan selanjutnya dapat membuat aplikasi berjalan di versi Android terbaru dan pengembangan proses diagnosa menggunakan gambar.
4. Aplikasi ini nantinya dapat dikembangkan sehingga kompatibel pada *platform* selain android, seperti iOS, dan Windows Phone.

## DAFTAR PUSTAKA

- Agro, M. (2012). *Budidaya Tanaman Lada Ramah Lingkungan Berbasis Teknologi Bio~FOB*. Bogor: CV Meori Agro.
- Agustianti, R. M. (2016). *Sistem Pakar Identifikasi Jenis Ikan Famili Cyprinidae Menggunakan Metode Forward Chaining dan Classic Probability (Skripsi)*. Bandar Lampung: Fakultas MIPA Universitas Lampung.
- Balfas, R. (2009). Status Penelitian Serangga Vektor Penyakit Kerdil Pada Tanaman Lada. *Perspektif*.
- Direktorat Jenderal Perkebunan , & Kementerian Pertanian. (2014). *Statistik Perkebunan Indonesia*. Jakarta: Direktorat Jenderal Perkebunan.
- Direktorat Perlindungan Perkebunan. (2002). *Musuh Alami, Hama dan Penyakit Tanaman Lada*. Jakarta: Departemen Pertanian.
- Erika, O. R. (2015). *Aplikasi Pencarian Letak ATM Berbasis Android Dengan GIS di Kota Bandar Lampung (Skripsi)*. Bandar Lampung: Fakultas MIPA Universitas Lampung.
- Hamdani. (2010). Sistem Pakar Untuk Diagnosa Penyakit Mata Pada Manusia. *Jurnal Informatika Mulawarman*.
- Hartati, S., & Iswanti, S. (2008). *Sistem Pakar dan Pengembangannya*. Yogyakarta: Graha Ilmu.

- Istanto, A., & Dewa, W. (2016). Sistem Pakar Untuk Mendiagnosa Hama dan Penyakit Tanaman Tomat Menggunakan Metode Forward Chaining . *Jurnal Teknologi Informasi*.
- Ladjamuddin, A.-B. b. (2006). *Rekayasa Perangkat Lunak*. Yogyakarta: Graha Ilmu.
- Manohara, D., & Wahyuno, D. (2013). *Pedoman Budidaya Merica*. Bogor: Balai Penelitian Tanaman Rempah dan Obat (Balitro).
- Munif, A., & Sulistiawati, I. (2014). Pengelolaan Penyakit Kuning pada Tanaman Lada oleh Petani di Wilayah Bangka. *Fitopatologi Indonesia*.
- Pratama, A. A. (2015). *Sistem Pakar Diagnosa Penyakit Pada Ikan Budidaya Air Tawar Dengan Metode Forward Chaining (Skripsi)*. Bandar Lampung: Fakultas MIPA Universitas Lampung.
- Putra, A. P. (2016). *Sistem Pakar Diagnosa Pada Ikan Air Tawar Dengan Metode Forward Chaining Berbasis Android (Skripsi)*. Bandar Lampung: Fakultas MIPA Universitas Lampung.
- Ruliansyah, B. (2016). *Pengembangan Sistem Informasi Geografis (GIS) Penyedia Pelayanan Kesehatan Berbasis Android (Skripsi)*. Bandar Lampung: Fakultas MIPA Universitas Lampung.
- Semangun, H. (1989). *Penyakit-penyakit Tanaman Perkebunan di Indonesia*. Yogyakarta: Universitas Gajah Mada.
- Sinaga, M. D. (2014). Sistem Pakar Mendeteksi Penyakit Tanaman Terong Belanda dengan Menggunakan Metode Forward Chaining. *Jatisi*.

Susanti, A. (2015). *Outlook Lada Komuditas Pertanian Subsektor Perkebunan*.

Jakarta: Pusat Data dan Sistem Informasi Pertanian Sekretariat Jenderal -  
Kementerian Pertanian.

Susanti, R., Jusak, & Sudarmaningtyas, P. (2016). Sistem Pakar untuk Menentukan

Penyakit Pada Tanaman Cokelat. *JSIKA*.