

**SISTEM PAKAR DIAGNOSA PENYAKIT TANAMAN
KACANG PANJANG DENGAN METODE
CERTAINTY FACTOR BERBASIS WEB**

PROPOSAL TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat Penulisan Tugas Akhir pada
Sekolah Tinggi Manajemen Informatika dan Komputer
(STMIK) Palangkaraya



OLEH :

MUHAMMAD KHOIRUL

NIM C1755201064

PROGRAM STUDI TEKNIK INFORMATIKA

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA
2021**

**SISTEM PAKAR DIAGNOSA PENYAKIT TANAMAN
KACANG PANJANG DENGAN METODE
CERTAINTY FACTOR BERBASIS WEB**

PROPOSAL TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat Penulisan Tugas Akhir pada
Sekolah Tinggi Manajemen Informatika dan Komputer
(STMIK) Palangkaraya

OLEH :

MUHAMMAD KHOIRUL

NIM C1755201064

PROGRAM STUDI TEKNIK INFORMATIKA

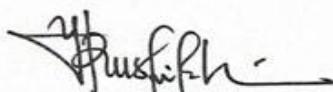
**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA
2021**

PERSETUJUAN

SISTEM PAKAR DIAGNOSA PENYAKIT TANAMAN KACANG PANJANG DENGAN METODE CERTAINTY FACTOR BERBASIS WEB

Proposal Tugas Akhir ini telah disetujui untuk diseminarkan
pada Tanggal 4 Maret 2021

Pembimbing I,



Sulistyowati, S.Kom., M.Cs.
NIK. 1982 1216 2007 002

Pembimbing II,



Sherly Jayanti, S.T., M.Cs.
NIK. 1985 0110 2012 004

Mengetahui
Rektor STMIK Palangka Raya,



Parno, M.Kom.
1969 0104 1995 105

PENGESAHAN

SISTEM PAKAR DIAGNOSA PENYAKIT TANAMAN KACANG PANJANG DENGAN METODE CERTAINTY FACTOR BERBASIS WEB

Proposal Tugas Akhir ini telah Diseminarkan, Dinilai dan Disahkan
Oleh Tim Seminar pada Tanggal 8 Maret 2021

Tim Sidang Tugas Akhir :

1. Lili Rusdiana, M.Kom.
Ketua
2. Sulistyowati, S.Kom., M.Cs.
Sekretaris
3. Sherly Jayanti, S.T., M.Cs.
Anggota



Three handwritten signatures are present on dotted lines, corresponding to the members of the final task team listed on the left. The signatures are written in black ink and are somewhat stylized.

KATA PENGANTAR

Puji syukur penulis panjatkan kehadiran Tuhan Yang Maha Esa yang telah melimpahkan karunia-Nya sehingga penulis bisa menyelesaikan proposal tugas akhir dengan judul “Sistem Pakar Diagnosa Penyakit Tanaman Kacang Panjang Dengan Metode *Certainty Factor* Berbasis Web”.

Penulis memahami, tanpa bantuan, doa dan bimbingan dari semua orang akan sangat sulit untuk menyelesaikan proposal ini. Maka dari itu penulis ingin mengucapkan terima kasih kepada :

1. Suparno, M.Kom selaku ketua STMIK Palangkaraya.
2. Sulistyowati, S.Kom., M.Cs selaku dosen pembimbing I
3. Sherly Jayanti, S.T., M.Cs selaku dosen pembimbing II
4. Orang tua yang selalu memberikan semangat dan seluruh teman-teman yang selalu membantu dan mendukung dalam penyusunan Tugas Akhir ini.

Atas segala kekurangan dan ketidaksempurnaan Tugas Akhir ini, penulis sangat mengharapkan masukan, kritik dan saran yang bersifat membangun kearah perbaikan dan penyempurnaan Tugas Akhir ini. Cukup banyak kesulitan yang penulis temui dalam penulisan skripsi ini, tetapi Alhamdulillah dapat penulis atasi dan selesaikan dengan baik.

Palangkaraya, 4 Maret 2021

Penulis

DAFTAR ISI

HALAMAN PERSETUJUAN.....	i
HALAMAN PENGESAHAN.....	ii
KATA PENGANTAR.....	iii
DAFTAR ISI.....	iv
DAFTAR TABEL.....	vi
DAFTAR GAMBAR.....	vii
DAFTAR LAMPIRAN.....	viii
BAB I. PENDAHULUAN	
1.1. Latar Belakang	1
1.2. Rumusan Masalah	3
1.3. Batasan Masalah	4
1.4. Tujuan dan Manfaat	5
a. Tujuan Penelitian	5
b. Manfaat Penelitian	6
1.5. Sistematika Penulisan.....	6
BAB II. LANDASAN TEORI	
2.1. Tinjauan Pustaka	7
2.2. Kajian Teori	9
2.2.1 Sistem Pakar	9
2.2.2 Penyakit Tanaman Kacang Panjang.....	18
2.2.3 <i>Certainty Factor</i>	23
2.2.4 MySQL.....	25
2.2.5 Notepad++.....	29
2.2.6 UML.....	29
2.2.7 PHP	35
2.2.8 Website.....	36
2.2.9 XAMPP	37

BAB III. METODE PENELITIAN

3.1 Lokasi Penelitian	38
3.2 Teknik Pengumpulan Data	38
3.3 Analisis	40
3.3.1 Analisis Proses	40
3.3.2 Analisis Kelemahan Sistem	43
3.3.3 Analisis Kebutuhan	44
3.3.4 Analisis Kelayakan Sistem	45
3.4 Desain Sistem.....	46
3.4.1 Desain Proses	46
3.4.2 Desain Perangkat Lunak.....	55
3.4.3 Desain Keamanan.....	57

DAFTAR PUSTAKA

LAMPIRAN-LAMPIRAN

DAFTAR TABEL

Tabel 1. Tabel Perbandingan Penelitian Yang Relevan	8
Tabel 2. Logika Metode Certainty Factor	24
Tabel 3. Simbol dalam Use Case Diagram	31
Tabel 4. Simbol dalam Activity Diagram.....	32
Tabel 5. Simbol dalam Sequence Diagram.....	34
Tabel 6. Sintak Variable pada PHP	36
Tabel 7. Kebutuhan Perangkat Keras	44
Tabel 8. Kebutuhan Perangkat Lunak	45
Tabel 9. Activity Diagram Login Admin	49
Tabel 10. Activity Diagram Menu Input Data Penyakit.....	51
Tabel 11. Activity Diagram Menu Data Konsultasi.....	52
Tabel 12. Spesifikasi Admin.....	53
Tabel 13. Spesifikasi Gejala	54
Tabel 14. Spesifikasi Diagnosa	54
Tabel 15. Tabel Menentukan	54

DAFTAR GAMBAR

Gambar 1.	Arsitektur Sistem Pakar	14
Gambar 2.	Use Case Diagram.....	47
Gambar 3.	Class Diagram	55
Gambar 4.	Layout Halaman Login.....	56
Gambar 5.	Layout Halaman Utama Aplikasi Untuk Admin.....	56
Gambar 6.	Layout Halaman Utama Aplikasi Untuk Operator.....	57

DAFTAR LAMPIRAN

- Lampiran 1. Surat Tugas Dosen Pembimbing
- Lampiran 2. Kartu Kegiatan Konsultasi Tugas Akhir
- Lampiran 3. Surat Tugas Penguji Seminar Tugas Akhir
- Lampiran 4. Daftar Hadir Peserta Seminar proposal Tugas Akhir
- Lampiran 5. Berita Acara Seminar Proposal Tugas Akhir

BAB I PENDAHULUAN

1.1 Latar Belakang Masalah

Semakin majunya perkembangan zaman dengan syarat pemanfaatan teknologi informasi seharusnya dapat menjadi solusi dalam masalah pelayanan publik. Teknologi informasi dapat dimanfaatkan untuk meningkatkan pelayanan menjadi lebih cepat, transparan sehingga pelayanan publik menjadi lebih efektif dan efisien.

Tanaman kacang panjang (*Vigna Sinensis L.*) merupakan komoditas sayur yang sangat potensial untuk dikembangkan, karena mempunyai nilai ekonomi yang cukup tinggi. Kacang panjang dapat dikonsumsi dalam bentuk segar maupun di olah menjadi sayur. Dalam upaya peningkatan gizi masyarakat, kacang panjang penting sebagai sumber vitamin dan mineral. Biji kacang panjang mengandung karbohidrat (70,00%), protein (17,30%), lemak (1,50%) dan air (12,20%), sehingga komoditi ini juga merupakan sumber protein nabati.

Keberadaan hama/penyakit di area pertanaman kacang panjang biasanya tidak sampai menyebabkan kegagalan panen namun dapat mengakibatkan berkurangnya hasil dan penurunan kualitas kacang panjang yang dihasilkan. Rendahnya produksi kacang panjang tersebut tidak terlepas dari adanya gangguan penyakit. Penyakit pada tanaman kacang panjang sebagian besar disebabkan oleh hama ulat, lalat kacang, tungau merah, layu sklerotium, karat daun, layu fusarium, dan bercak daun.

Penyebaran penyakit pada tanaman kacang panjang dan juga penularan secara non persisten dimana hama awalnya menghisap tanaman kacang panjang yang terinfeksi virus hanya dalam beberapa detik, kemudian hama akan menularkan virusnya dengan cepat ke tanaman berikutnya. Sebelumnya dalam mendiagnosa penyakit pada kacang panjang petani biasanya mengamati melalui gejala-gejala yang nampak pada tanaman, misalnya ketika terkena penyakit lalat kacang terdapat bintik-bintik putih sekitar tulang daun, pertumbuhan tanaman yang terserang terhambat dan daun berwarna kekuningan, pangkal batang terjadi perakaran sekunder dan membengkak

Kendala utama dalam mendiagnosa kacang panjang antara lain adalah minimnya pengetahuan petani tentang penyakit kacang panjang, keterbatasan waktu yang dimiliki para petani dalam mendeteksi penyakit serta pengambilan keputusan untuk proses penanggulangan sehingga mempermudah para petani untuk mendeteksi jenis penyakit yang menyerang tanaman kacang Panjang.

Sistem pakar adalah sistem berbasis komputer yang menggunakan pengetahuan, fakta, dan teknik penalaran dalam memecahkan masalah yang biasanya hanya dapat dipecahkan oleh pakar bidang tertentu. Kelebihan sistem pakar diantaranya adalah memungkinkan orang awam dapat mengerjakan pekerjaan para pakar (ahli). Sistem pakar dapat digunakan untuk menyimpan pengetahuan dan keahlian pakar.

Dengan menggunakan sistem pakar para petani dapat dengan mudah mendiagnosa penyakit yang menyerang tanaman kacang panjang karena sistem pakar dapat digunakan untuk menyimpan pengetahuan dan keahlian pakar, selain itu sistem pakar dapat meningkatkan kapabilitas dalam menyelesaikan masalah sehingga menghemat waktu dalam pengambilan keputusan. Salah satu pelayanan publik yang perlu menggunakan sistem pakar ini adalah Balai Pengkajian Teknologi Pertanian Kalimantan Tengah agar dapat mempermudah kinerja pelayanan publik sebagai sumber informasi untuk menyampaikan informasi ke petani mengenai diagnosa penyakit pada tanaman kacang panjang. Pengoperasian aplikasi yang dibuat oleh penulis diharapkan juga mudah digunakan dengan adanya menu bantuan/panduan penggunaan pada aplikasi. Dengan data-data yang telah dikelola maka dengan mudah mencari data yang diperlukan dengan menyaring (filter) data yang diperlukan.

Berdasarkan uraian di atas, maka penulis bermaksud melakukan sebuah penelitian dengan judul “Sistem Pakar Diagnosa Penyakit Pada Tanaman Kacang Panjang Dengan Metode *Certainty Factor* Berbasis Web”

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang ada maka dapat dirumuskan permasalahan dari penelitian ini “bagaimana merancang sebuah aplikasi **Sistem Pakar Diagnosa Penyakit Tanaman Kacang Panjang Dengan Metode *Certainty Factor* berbasis Web?**”

1.3 Batasan Masalah

Agar sistem dan penelitian sesuai dengan tujuan yang ingin dicapai maka diperlukan batasan masalah agar menjadi sistematis. Batasan-batasan masalah sistem pakar ini adalah sebagai berikut :

- a. Penggunaan dari aplikasi ini adalah para petani kacang panjang Kantor Balai Pengkajian Teknologi Pertanian Kalimantan Tengah.
- b. Pada aplikasi ini tidak terdapat derajat kepastian suatu penyakit.
- c. Objek yang diteliti adalah tanaman kacang panjang.
- d. Metode yang digunakan adalah *certainty factor*.
- e. Input dari sistem berupa 10 gejala-gejala yang ada pada tanaman kacang panjang.
- f. Output yang dihasilkan dari sistem ini berupa 5 jenis penyakit seperti Antraknose, penyakit sapu, layu bakteri dan penyakit mosaik serta bagaimana cara pencegahannya.
- g. Pembuatan aplikasi ini menggunakan bahasa pemrograman PHP dan database MYSQL.

1.4 Tujuan dan Manfaat

- a. Tujuan Penelitian

Tujuan yang ingin dicapai oleh peneliti adalah merancang dan membangun sistem untuk mendiagnosa penyakit pada tanaman kacang panjang.

b. Manfaat penelitian

Manfaat yang diharapkan dalam pembuatan sistem ini adalah sebagai berikut:

- 1) Memberikan solusi dan mempermudah untuk mendeteksi penyakit yang ada pada tanaman kacang panjang.
- 2) Dapat meningkatkan hasil panen kacang Panjang.

1.5 Sistematika Penulisan

Sistematika dari penulisan tugas akhir ini dapat dijelaskan sebagai berikut:

BAB I PENDAHULUAN

Bab ini berisi tentang latar belakang masalah, rumusan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan

BAB II LANDASAN TEORI

Pada bab ini membahas tentang tinjauan pustaka berisi penelitian yang Relevan, susunan kajian teori disesuaikan dengan tema tugas akhir.

BAB III ANALISIS DAN DESAIN SISTEM

Pada bab ini menjelaskan tentang subyek penelitian, metode penelitian yang digunakan. Perangkat yang digunakan dan pembangunan system yang dibuat.

BAB IV IMPLEMENTASI DAN PEMBAHASAN

Bab ini menjelaskan tentang deskripsi, analisis sistem, desain sistem, implemtasi dan pembahasan, tampilan progrann, serta pengujian sistem

BAB V KESIMPULAN DAN SARAN

Bab ini berisi tentang kesimpulan-kesimpulan program yang telah dibuat serta saran agar laporan ini makin sempurna

BAB II LANDASAN TEORI

2.1 Tinjauan Pustaka

Dalam suatu penelitian diperlukan dukungan hasil-hasil penelitian yang telah ada sebelumnya yang berkaitan dengan penelitian serupa.

Berikut hasil-hasil penelitian yang relevan dan perbandingan penelitian yang telah ada sebelumnya yang serupa dengan penelitian yang sedang dilakukan yang disajikan dalam bentuk tabel.

Tabel 1. Tabel perbandingan penelitian yang relevan

No.	Penulis/Tahun	Topik Penelitian	Metode	Hasil	Perbedaan
1.	Titiek Harmanti / 2013	Sistem Pakar Penelusuran Penyakit Pada Tanaman Kacang Panjang (Vigna Sesquipedalis)	Forward Chaining, DFD, Visual Basic	Sistem ini dapat mengidentifikasi penyakit pada kacang panjang ini berisikan tentang penyakit kacang panjang, dan hasilnya mendapatkan jenis penyakit dan pengendaliannya	Penulis menggunakan metode <i>certainty factor</i> , desain sistem menggunakan UML dan bahasa pemrograman PHP, sedangkan penelitian yang relevan menggunakan metode Forward chaining, DFD dan Visual Basic serta yang diteliti objeknya adalah kacang panjang
2.	Aris Widyanto / 2013	Sistem Pakar ini dibuat untuk Mendiagnosa Hama dan Penyakit Tanaman Pepaya	Forward Chaining, DFD, Visual Basic	Sistem ini dapat mendiagnosa penyakit pada tanaman pepaya dan mendapatkan hasil yang akurat	Penulis menggunakan metode <i>certainty factor</i> , desain sistem menggunakan UML dan bahasa pemrograman PHP, sedangkan penelitian yang relevan menggunakan metode Forward chaining, DFD dan Visual Basic serta yang diteliti objeknya adalah pepaya.
3.	Bella Anita Agustin Maulina dan Harrison D.S. / 2016	Sistem Pakar ini buat untuk mendiagnosis Hama dan Penyakit Tanaman Kacang Tanah	Backward Chaining, DFD, dan Borland Delphi 7	Sistem ini dapat memudahkan dalam mengetahui gejala dan penyebab adanya hama	Penulis menggunakan metode <i>certainty factor</i> , desain sistem menggunakan UML dan bahasa

				dan penyakit tanaman kacang tanah secara cepat dan akurat	pemrograman PHP, sedangkan penelitian yang relevan menggunakan metode backward chaining, DFD dan Borland Delphi 7 serta yang diteliti objeknya adalah kacang tanah
--	--	--	--	---	--

2.2 Kajian Teori

2.2.1 Sistem Pakar

Menurut (Rahmat Hidayat, 2014) dalam bukunya Perancangan Sistem Pakar, beberapa definisi sistem pakar menurut beberapa ahli yaitu sebagai berikut.

1. Menurut Durkin : Sistem pakar adalah suatu program komputer yang dirancang untuk memodelkan kemampuan penyelesaian masalah yang dilakukan seorang pakar.
2. Menurut Ignizo : 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 menyamai atau meniru kemampuan seorang pakar.

Bentuk umum sistem pakar adalah suatu program yang dibuat berdasarkan suatu set aturan yang menganalisis informasi (biasanya diberikan oleh pengguna suatu sistem) mengenai suatu kelas masalah spesifik serta analisis matematis dari masalah tersebut. Tergantung

dari desainnya, sistem pakar juga mampu merekomendasikan suatu rangkaian tindakan pengguna untuk dapat menerapkan koreksi. Sistem ini memanfaatkan kapabilitas penalaran untuk mencapai suatu simpulan.

a. Manfaat Sistem Pakar

Sistem pakar menjadi sangat populer karena sangat banyak kemampuan dan manfaat yang diberikannya, diantaranya adalah :

- 1) Meningkatkan produktivitas.
- 2) Membuat orang awam bekerja seperti layaknya seorang pakar.
- 3) Mampu menangkap pengetahuan dan kepakaran seseorang.
- 4) Memudahkan akses pengetahuan seorang pakar.
- 5) Meningkatkan kapabilitas sistem komputer. Integrasi sistem pakar dengan sistem komputer lain membuat sistem lebih efektif dan mencakup lebih banyak aplikasi.
- 6) Mampu bekerja dengan informasi yang tidak lengkap atau tidak pasti.
- 7) 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.

- 8) Meningkatkan kemampuan untuk menyelesaikan masalah karena sistem pakar mengambil sumber pengetahuan dari banyak pakar.

b. Kekurangan Sistem Pakar

Selain manfaat, ada juga beberapa kekurangan yang ada pada sistem pakar, diantaranya adalah :

- 1) Biaya yang diperlukan untuk membuat, memelihara, dan mengembangkannya sangat mahal.
- 2) Sulit dikembangkan.
- 3) Sistem pakar tidak 100% benar.
- 4) Pendekatan oleh setiap pakar untuk suatu situasi atau problem bisa berbeda - beda, meskipun sama - sama benar.
- 5) Kurangnya rasa percaya pengguna dapat menghalangi pemakaian sistem pakar.

c. Ciri – Ciri Sistem Pakar

Ciri – Ciri dari Sistem Pakar adalah sebagai berikut :

- 1) Terbatas pada domain keahlian tertentu.
- 2) Dapat memberikan penalaran untuk data – data yang tidak lengkap atau tidak pasti.
- 3) Dapat menjelaskan alasan – alasan dengan cara yang dapat dipahami.
- 4) Bekerja berdasarkan kaidah / rule tertentu.
- 5) Mudah dimodifikasi

- 6) Basis pengetahuan dan mekanisme inferensi terpisah
- 7) Keluarannya bersifat anjuran
- 8) System dapat mengaktifkan kaidah secara searah yang sesuai, dituntun oleh dialog dengan pengguna.

d. Area Permasalahan Sistem Pakar

Biasanya sistem pakar menyentuh area permasalahan sebagai berikut:

1) Interpretasi

Yaitu pengambilan keputusan dari hasil observasi, diantaranya : pengawasan, pengenalan ucapan, analisis citra, interpretasi sinyal, dan beberapa analisis kecerdasan

2) Prediksi

Memprediksi akibat-akibat yang dimungkinkan dari situasi-situasi tertentu, diantaranya peramalan, prediksi demografis, peralaman ekonomi, prediksi lalu lintas, estimasi hasil militer, pem&wan, atau peramalan keuangan.

3) Diagnosis

Menentukan sebab malfungsi dalam situasi kompleks yang didasarkan pada gejala-gejala yang teramati, diantaranya : medis, elektronis, mekanis, dan diagnosis perangkat lunak

4) Desain

Menentukan konfigurasi komponen-komponen sistem yang cocok dengan tujuan-tujuan kinerja tertentu dan kendala-

kendala tertentu, diantaranya : layout sirkuit, perancangan bangunan

5) Perencanaan

Merencanakan serangkaian tindakan yang akan dapat mencapai sejumlah tujuan dengan kondisi awal tertentu, diantaranya : perencanaan keuangan, komunikasi, militer, pengembangan politik, routing dan manajemen proyek.

6) Monitoring

Membandingkan tingkah laku suatu sistem yang teramati dengan tingkah laku yang diharapkan darinya, diantaranya: Computer Aided Monitoring System

7) Debugging dan repair

Menentukan dan mengimplementasikan cara-cara untuk mengatasi malfungsi, diantaranya memberikan resep obat terhadap suatu kegagalan.

8) Instruksi

Melakukan instruksi untuk diagnosis, debugging dan perbaikan kinerja.

9) Kontrol

Mengatur tingkah laku suatu environment yang kompleks seperti kontrol terhadap interpretasi-interpretasi, prediksi, perbaikan, dan monitoring kelakuan system

10) Seleksi

Mengidentifikasi pilihan terbaik dari sekumpulan (list) kemungkinan.

11) Simulasi

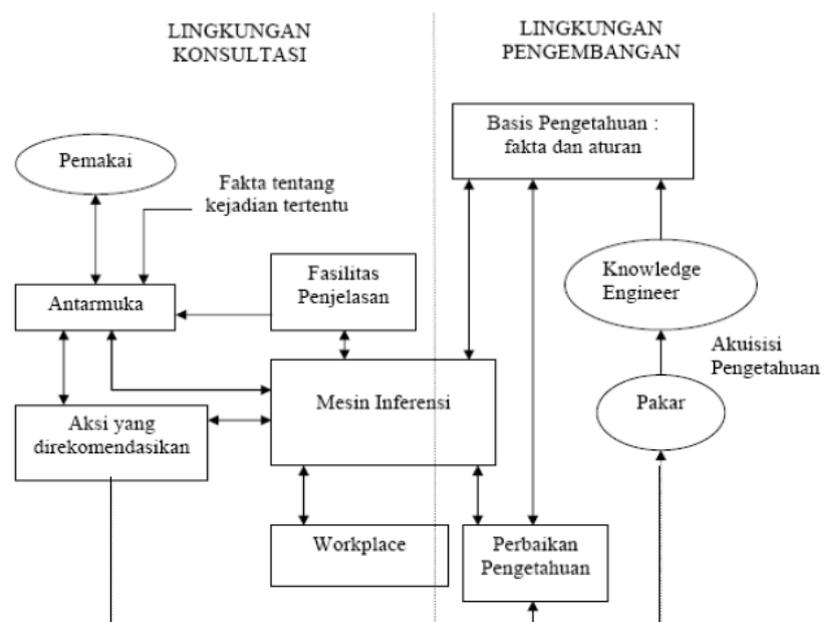
Pemodelan interaksi antara komponen-komponen sistem.

e. Struktur Sistem Pakar

Bagian utama Sistem Pakar

- 1) Lingkungan Pengembangan (*Development Environment*)
- 2) Lingkungan Konsultasi (*Consultation Environment*)

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.



Gambar 1. Arsitektur Sistem Pakar

a) Antarmuka Pengguna (*User Interface*)

Antarmuka merupakan mekanisme yang digunakan oleh pengguna dan sistem pakar untuk berkomunikasi. Antarmuka menerima informasi dari pemakai dan mengubahnya ke dalam bentuk yang dapat diterima oleh sistem. Selain itu antarmuka juga menerima dari sistem dan menyajikannya ke dalam bentuk yang dapat dimengerti oleh pemakai.

b) Basis Pengetahuan

Basis pengetahuan mengandung pengetahuan untuk pemahaman formulasi dan penyelesaian masalah.

Basis pengetahuan terdiri dari 2 elemen dasar, yaitu:

- 1) Fakta : informasi tentang obyek dalam area permasalahan tertentu.
- 2) Aturan : informasi tentang cara bagaimana memperoleh fakta baru dari fakta yang telah diketahui.

c) Akuisisi Pengetahuan (*Knowledge Acquisition*)

Akuisisi pengetahuan adalah akumulasi transfer dan transformasi keahlian dalam menyelesaikan masalah dari sumber pengetahuan ke dalam program komputer.

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.

Metode akuisisi pengetahuan :

- 1) Wawancara : metode yang paling banyak digunakan, yang melibatkan pembicaraan dengan pakar secara langsung dalam suatu wawancara.
- 2) Analisis protocol : merupakan suatu metode akuisisi pengetahuan dimana pakar diminta untuk melakukan suatu pekerjaan dan mengungkapkan proses pemikirannya dengan menggunakan kata-kata. (direkam, ditulis, dan dianalisis).
- 3) Observasi pada pekerjaan pakar : merupakan suatu metode akuisisi dengan cara merekam dan mengobservasi Sesuatu pekerjaan dalam bidang tertentu yang dilakukan oleh pakar.
- 4) Induksi aturan dari contoh : Induksi adalah suatu proses penalaran dari khusus ke umum. Suatu sistem induksi aturan diberi contoh-contoh dari suatu masalah yang hasilnya telah diketahui. Setelah diberikan beberapa

contoh, sistem induksi aturan tersebut dapat membuat aturan yang benar untuk kasus-kasus contoh. Selanjutnya aturan dapat digunakan untuk menilai kasus lain yang hasilnya tidak diketahui.

d) Mesin/ Motor Inferensi (*Inference Engine*)

Komponen ini mengandung mekanisme pola pikir dan penalaran yang digunakan oleh pakar dalam menyelesaikan suatu masalah.

Mesin inferensi merupakan program komputer yang memberikan metodologi untuk penalaran tentang informasi yang ada dalam basis pengetahuan dan dalam workplace dan untuk memformulasikan kesimpulan.

e) *Workplace / Blackboard*

Workplace merupakan area dari sekumpulan memori kerja (*working memory*) yang digunakan untuk merekam kejadian yang sedang berlangsung termasuk keputusan sementara.

Ada 3 keputusan yang dapat direkam:

- a) Rencana : bagaimana menghadapi masalah
 - b) Agenda : aksi-aksi yang potensial yang sedang menunggu untuk dieksekusi
 - c) Solusi : calon aksi yang akan dibangkitkan
- f) Fasilitas Penjelasan

Adalah komponen tambahan yang akan meningkatkan kemampuan sistem pakar. Digunakan untuk melacak respond dan memberikan penjelasan tentang kelakuan sistem pakar secara interaktif melalui pertanyaan.

g) Perbaiki Pengetahuan

Pakar memiliki kemampuan untuk menganalisa dan meningkatkan kinerjanya serta kemampuan untuk belajar dari kinerjanya. Kemampuan tersebut adalah penting dalam pembelajaran terkomputerisasi, sehingga program akan mampu menganalisis penyebab kesuksesan dan kegagalan yang dialaminya dan juga mengevaluasi apakah pengetahuan-pengetahuan yang ada masih cocok untuk digunakan di masa mendatang.

2.2.2 Penyakit Tanaman Kacang Panjang

Kacang panjang (*Vigna sinensis L.*) merupakan tanaman hortikultura jenis kacang-kacangan yang dibudidayakan secara luas di Indonesia. Kandungan protein, lemak, karbohidrat, vitamin A, C, dan mineral terutama pada polong yang masih muda menyebabkan kacang panjang memiliki nilai ekonomi tinggi serta banyak dikonsumsi oleh masyarakat di Indonesia.

Beberapa penyakit yang menyerang tanaman kacang panjang diantaranya layu cendawan (*Fusarium sp*), Antraknosa (*Colletotricum lindemuthianum*), puru akar (*Meloidogyne sp*),

penyakit sapu (*Cowpea Witches-broom Virus/Cowpea Stunt Virus*), layu bakteri (*Pseudomonas solanacearum*) dan penyakit mosaik yang disebabkan oleh *Bean common mosaic virus* (BCMV) dan *Cowpea aphid borne mosaic virus* (CABMV) (Anwar et al., 2005). Penyakit *mosaik vein* banding yang dilaporkan oleh Damayanti et al., (2009) disebabkan oleh *Bean common mosaic virus* (BCMV) dan *Cucumber mosaic cucumovirus* (CMV).

a. Penyebab penyakit tanaman kacang panjang

Hama adalah organisme yang dianggap merugikan dan tak diinginkan dalam kegiatan sehari-hari manusia. Walaupun dapat digunakan untuk semua organisme, dalam praktik istilah ini paling sering dipakai hanya kepada hewan.

Dalam pertanian, hama adalah organisme pengganggu tanaman yang menimbulkan kerusakan secara fisik, dan ke dalamnya praktis adalah semua hewan yang menyebabkan kerugian dalam pertanian.

- a) Lalat kacang (*Ophiomya phaseoli* Tryon)
- b) Kutu daun (*Aphis cracivora* Koch)
- c) Ulat grayak (*Spodoptera litura* F.)
- d) Penggerek biji (*Callosobruchus maculatus* L)
- e) Ulat penggerek polong (*Maruca testalis*)
- f) Tunggau merah (*Tetranychus cinnabarinus* Boisduval, *T. bimaculatus* Harv, *T. telatius* dan *T. cucurbitacearum*)

b. Jenis-jenis penyakit pada tanaman kacang Panjang

Beberapa jenis penyakit yang ada pada tanaman kacang Panjang yaitu:

1) Antraknose

Penyebab: jamur *Colletotricum lindemuthianum*. Gejala: serangan dapat diamati pada bibit yang baru berkecamabah, semacam kanker berwarna coklat pada bagian batang dan keping biji. Pengendalian: dengan rotasi tanaman, perlakuan benih sebelum ditanam dengan fungisida Dithane M-45 dan Cupravit OB 21 0,1-0,2% dan membuang rumput-rumput dari sekitar tanaman.

2) Penyakit mozaik

Penyebab: virus Cowpea Aphid Borne Virus/CAMV. Gejala: pada daun-daun muda terdapat gambaran mozaik yang warnanya tidak beraturan. Penyakit ditularkan oleh vektor kutu daun. Pengendalian: dengan menggunakan benih yang sehat dan bebas virus, disemprot dengan insektisida yang efektif untuk kutu daun dan tanaman yang terserang dicabut dan dibakar.

3) Penyakit sapu

Penyebab: virus Cowpea Witches-broom Virus/Cowpea Stunt Virus. Gejala: pertumbuhan tanaman terhambat, ruas-ruas (buku-buku) batang sangat pendek, tunas ketiak memendek

dan membentuk "sapu". Penyakit ditularkan kutu daun.

Pengendalian: sama dengan pengendalian penyakit mosaik.

4) Layu bakteri

Penyebab: bakteri *Pseudomonas solanacearum* E.F. Smith.

Gejala: tanaman mendadak layu dan serangan berat menyebabkan tanaman mati. Pengendalian: dengan rotasi tanaman, perbaikan drainase dan mencabut tanaman yang mati.

c. Gejala penyakit tanaman kacang panjang

Beberapa gejala yang ada pada tumbuhan kacang panjang yaitu:

- 1) Bercak kecil berwarna putih pada permukaan daun kemudian bercak membesar dan berubah menjadi cokelat bertepung di kelilingi warna kuning atau bercincin cokelat
- 2) Serangan dapat diamati pada bibit yang baru berkecambah, semacam kanker berwarna cokelat pada bagian batang dan keping biji.
- 3) Pada daun-daun muda terdapat gambaran mosaik yang warnanya tidak beraturan. Penyakit ditularkan oleh vektor kutu daun.
- 4) Pertumbuhan tanaman terhambat, ruas-ruas (buku-buku) batang sangat pendek, tunas ketiak memendek dan membentuk "sapu". Penyakit ditularkan kutu daun

- 5) Tanaman mendadak layu dan serangan berat menyebabkan tanaman mati.
- 6) Terdapat bintik-bintik putih sekitar tulang daun, pertumbuhan tanaman yang terserang terhambat dan daun berwarna kekuningan, pangkal batang terjadi perakaran sekunder dan membengkak. Pengendalian: dengan cara pergiliran tanaman yang bukan dari famili kacang-kacangan dan penyemprotan dengan insektisida Orthene 75 SP 1 cc/liter.
- 7) Pertumbuhan terlambat karena hama mengisap cairan sel tanaman dan penurunan hasil panen. Kutu bergerombol di pucuk tanaman dan berperan sebagai vektor virus. Pengendalian: dengan rotasi tanaman dengan tanaman bukan famili kacang-kacangan dan penyemprotan insektisida Furadan 3G dan Carbofuran 80 kg/ha.
- 8) Daun berlubang dengan ukuran tidak pasti, serangan berat di musim kemarau, juga menyerang polong. Pengendalian: dengan peraikan kultur teknis, rotasi tanaman, penanaman serempak, perangkap hama kimiawi dan insektisida Suoracide 0,1-0,2%.
- 9) Biji dirusak berlubang-lubang, hancur sampai 90%. Pengendalian: dengan membersihkan dan memusnahkan sisa-sisa tanaman tempat persembunyian hama. Benih

kacang panjang diberi perlakuan minyak jagung 10 cc/kg biji.

- 10) Larva menyerang bunga yang sedang membuka, kemudian memakan polong. Pengendalian: dengan rotasi tanaman dan menjaga kebersihan kebun dari sisa-sisa tanaman. Disemprot dengan insektisida yang efektif seperti Sevin pada konsentrasi 0,1%-0,2%.

2.2.3 *Certainty Factor*

Menurut (Dewi, 2013). *Certainty factor* adalah suatu metode untuk membuktikan apakah suatu fakta itu pasti ataukah tidak pasti yang berbentuk metric yang biasanya digunakan dalam sistem pakar. Metode ini sangat cocok untuk sistem pakar yang mendiagnosis sesuatu yang belum pasti.

Teori *Certainty Factor* adalah faktor kepastian (*certainty factor*) menyatakan kepercayaan dalam sebuah kejadian (fakta atau hipotesa) berdasar bukti atau penilaian pakar. *Certainty factor* menggunakan suatu nilai untuk mengasumsikan derajat keyakinan seorang pakar terhadap suatu data. Secara umum Teori *Certainty Factor* ditulis dalam suatu interval: *Certainty factor* menggunakan suatu nilai untuk mengasumsikan derajat keyakinan seorang pakar terhadap suatu data.

$$CF[H,E] = MB[H,E] - MD[H,E]$$

Keterangan :

CF (H,E) = *certainty factor* hipotesa yang dipengaruhi oleh *evidence e* diketahui dengan pasti.

$MB(H,E)$ = *measure of belief* terhadap hipotesa H, jika diberikan evidence E (antara 0 dan 1).

$MD(H,E)$ = *measure of disbelief* terhadap evidence H, jika diberikan evidence E (antara 0 dan 1) Certainty factor untuk kaidah premis tunggal.

$$CF[H,E]_1 = CF[H] * CF[E]$$

Certainty Factor untuk kaidah dengan kesimpulanyang serupa (similarly concluded rules):

$$CF_{combine}CF[H,E]_{1,2} = CF[H,E]_1 + CF[H,E]_2 * [1 - CF[H,E]_1]$$

$$CF_{combine}CF[H,E]_{old,3} = CF[H,E]_{old} + CF[H,E]_3 * (1 - CF[H,E]_{old})$$

Tabel 2 : Logika Metode certainty factor

No	Keterangan	Nilai User
1	Tidak	0
2	Tidak tahu	0,2
3	Sedikit yakin	0,4
4	Cukup yakin	0,6
5	Yakin	0,8
6	Sangat yakin	1

Nilai 0 menunjukkan bahwa pengguna konsultasi menginformasikan bahwa user tidak mengalami gejala seperti yang ditanyakan oleh sistem. Semakin pengguna konsultasi yakin bahwa gejala tersebut memang dialami manusia, maka semakin tinggi pula hasil prosentase keyakinan yang diperoleh. Proses penghitungan presentase keyakinan diawali dengan pemecahan sebuah kaidah yang memiliki premis majemuk, menjadi kaidah-kaidah yang memiliki premis tunggal. Kemudian masing-masing aturan baru dihitung certainty

factornya, sehingga diperoleh nilai certainty factor untuk masing-masing aturan, kemudian nilai certainty factor tersebut dikombinasikan.

a) Kelebihan metode *Certainty Factor* adalah:

- 1) Metode ini cocok dipakai dalam sistem pakar untuk mengukur sesuatu apakah pasti atau tidak pasti dalam mendiagnosa penyakit sebagai salah satu contohnya.
- 2) Perhitungan dengan menggunakan metode ini dalam sekali hitung hanya dapat mengolah 2 data saja sehingga keakuratan data dapat terjaga.

b) Kekurangan metode *Certainty Factor* adalah:

- 1) Ide umum dari pemodelan ketidakpastian manusia dengan menggunakan numer metode *certainty factor* biasanya diperdebatkan. Sebagian orang akan membantah pendapat bahwa formula untuk metode *certainty factor* diatas memiliki sedikit kebenaran.
- 2) Metode ini hanya dapat mengolah ketidakpastian/kepastian hanya 2 data saja. Perlu dilakukan beberapa kali pengolahan data untuk data yang lebih dari 2 buah.

2.2.4 MySQL

Menurut (Arief, 2011) MySQL merupakan salah satu jenis database server yang sangat terkenal dan banyak dibangun untuk membangun aplikasi web yang menggunakan database sebagai sumber pengolah datanya.

MySQL adalah salah satu jenis *database* server yang sangat terkenal dan banyak digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengelolaan datanya.

MySQL merupakan database yang pertama kali didukung oleh bahasa pemrograman script untuk internet (PHP dan Perl). MySQL dan PHP dianggap pasangan software pengembangan aplikasi *web* yang ideal. MySQL lebih sering di gunakan untuk membangun aplikasi berbasis *web*, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman script PHP.

a. *Data Definition Language* (DDL)

DDL merupakan perintah-perintah yang biasa digunakan administrator database untuk mendefinisikan skema dan subskema database.

Data Definition Language (DDL) mempunyai fungsi utama untuk mendefinisikan data dalam database secara logika, diantaranya yaitu digunakan untuk mendefinisikan karakteristik dari record (meliputi nama, tipe dan lebar dari field), untuk menentukan kunci field, menyediakan cara untuk menentukan hubungan dengan data di *file* lain, untuk mengubah struktur dari record, untuk menampilkan struktur dari record. DDL digunakan untuk mendefinisikan, mengubah, serta menghapus basis data dan objek-objek yang diperlukan dalam basis data, misalnya tabel. Perintah yang termasuk DDL:

- 1) *CREATE*, untuk membuat, termasuk diantaranya membuat database dan tabel baru. Contoh sintaks sebagai berikut :

```
CREATE TABLE peserta (No SMALLINT UNSIGNED NOT
NULL AUTO_INCREMENT, Nama CHAR(30) NOT NULL,
BidangStudi ENUM('TS','WD') NOT NULL, PRIMARY
KEY (No), INDEX (Nama, BidangStudi)).
```

- 2) *ALTER*, untuk mengubah struktur tabel yang telah dibuat. Contoh sintaksnya sebagai berikut ;

```
ALTER TABLE peserta ADD (JenisKelamin CHAR(10));
```

- 3) *DROP*, untuk menghapus database dan tabel. Contoh sintaksnya sebagai berikut :

```
DROP TABLE peserta;
```

```
DROP INDEX No;
```

b. *Data Control Language (DCL)*

DCL merupakan merupakan perintah-perintah yang digunakan untuk mengontrol data. Perintah yang termasuk DCL:

- 1) *GRANT* » untuk memberikan hak atau izin akses oleh administrator server kepada user. Contoh sintaksnya sebagai berikut :

```
GRANT ALL PRIVILEGES ON database_name.* TO
'myuser'
IDENTIFIED BY 'mypassword';
```

2) *REVOKE* » untuk menghilangkan atau mencabut hak akses yang telah diberikan kepada user oleh administrator. Contoh sintaksnya sebagai berikut :

```
REVOKE hak_akses ON nama_database.nama_tabel
FROM user;
```

c. *Data Manipulation Language (DML)*

DML merupakan merupakan perintah-perintah yang memungkinkan pengguna melakukan akses dan manipulasi data sebagaimana yang telah diorganisasikan sebelumnya dalam model data yang tepat, Data Manipulation Language digunakan untuk memanipulasi database yang telah didefinisikan dengan DDL.

Perintah yang termasuk DML:

1) *INSERT* » untuk menyisipkan atau memasukan dalam tabel.

Contoh sintaksnya sebagai berikut :

```
INSERT INTO Pelajar VALUES ('00311217','Wempi
Satria','02-JAN-1982','1','Laki-laki');
```

2) *UPDATE* » untuk memperbaharui data lama menjadi data terkini. Contoh sintaksnya sebagai berikut :

```
UPDATE Pelajar SET No_Induk = '00311216' ,Nama =
'Wati'
```

```
WHERE No_Induk ='00311210' and Nama = 'Satria';
```

3) *DELETE* » untuk menghapus data dari tabel. Contoh sintaksnya sebagai berikut :

```
DELETE FROM Pelajar WHERE No_Induk = '00311211';
```

4) *SELECT* » untuk mengambil data atau menampilkan data dari satu tabel atau beberapa tabel. Contoh sintaks sebagai berikut :

```
SELECT a.No_Induk, a>Nama, b.Kode, b>Nama, c.NI_Angka
FROM Pelajar a, Mata_Pelajaran b, Nilai c; WHERE
a.No_Induk=c.No_Induk and b.Kode=c.kode;
```

2.2.5 Notepad++

Menurut (Madcom, 2016) “Notepad++ adalah sebuah text editor yang sangat berguna dalam membuat program. Notepad++ menggunakan komponen Scintilla untuk menampilkan teks dan berkas kode sumber berbagai bahasa pemrograman yang berjalan di atas sistem operasi M. Windows”.

Selain manfaat dan kemampuannya menangani banyak bahasa pemrograman, Notepad ++ juga dilisensikan sebagai perangkat free. Jadi, setiap orang yang menggunakannya tidak perlu mengeluarkan biaya untuk membeli aplikasi ini karena sourceforge.net sebagai layanan yang memfasilitasi Notepad ++ membebaskannya untuk digunakan.

2.2.6 UML (*Unified Modeling Language*)

Menurut (Adi, 2013) *UML (Unified Modeling Language)* adalah “bahasa” pemodelan untuk sistem atau perangkat lunak yang berparadigma ‘Berorientasi Objek’. Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami.

UML (Unified Modeling Language) merupakan metode pemodelan secara visual sebagai sarana untuk merancang atau membuat *software* berorientasi objek. *UML* adalah salah satu *tool/model* untuk merancang pengembangan *software* yang berbasis

object oriented. UML sendiri juga memberikan standar penulisan sebuah sistem *blue print*, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema *database*, dan komponen-komponen yang diperlukan dalam sistem *software*. UML adalah sebuah bahasa standar untuk pengembangan sebuah *software* yang dapat menyampaikan bagaimana membuat dan membentuk model-model, tetapi tidak menyampaikan apa dan kapan model yang seharusnya dibuat yang merupakan salah satu proses implementasi pengembangan *software*. UML mendefinisikan diagram-diagram sebagai berikut :

1) *Use Case Diagram*

Use Case Diagram menggambarkan apa saja aktifitas yang dilakukan oleh suatu sistem dari sudut pandang pengamatan luar. yang menjadi persoalan itu apa yang dilakukan bukan bagaimana melakukannya. *Use Case Diagram* dekat kaitannya dengan kejadian-kejadian. Kejadian atau skenario merupakan contoh apa yang terjadi ketika seseorang berinteraksi dengan system. *Use Case Diagram* berguna dalam tiga hal :

- a) Menjelaskan fasilitas yang ada
- b) Memudahkan komunikasi dengan klien
- c) Membuat tes dari kasus-kasus yang muncul

Tabel 3. Simbol dalam *Use Case Diagram*

Notasi	Nama	Keterangan
	<i>Actor</i>	peran yang pengguna mainkan ketika berinteraksi dengan <i>Use Case</i> .
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Include</i>	Menspesifikasikan bahwa <i>Use Case</i> sumber secara eksplisit
	<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
	<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas

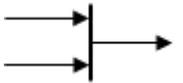
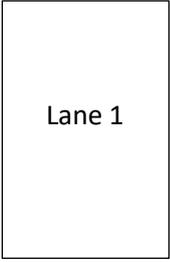
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya.
	<i>Note</i>	Elemen fisik yang ada saat aplikasi dijalankan

2) Activity Diagram

Activity diagram berfokus pada aktifitas-aktifitas yang terjadi yang terkait dalam suatu proses tunggal. Jadi dengan kata lain, diagram ini menunjukkan bagaimana aktifitas-aktifitas tersebut bergantung satu sama lain. *Activity diagram* dapat dibagi menjadi beberapa jalur kelompok yang menunjukkan obyek yang mana yang bertanggung jawab untuk suatu aktifitas.

Tabel 4. Simbol dalam *Activity Diagram*

Notasi	Nama	Keterangan
	<i>Start Point</i>	Dimulainya alur kerja suatu sistem dalam activity diagram dinotasikan dengan solid.
	<i>End Point</i>	Merepresentasikan diakhirnya alur kerja suatu sistem dalam activity diagram. Dinotasikan dengan lingkaran solid dengan lingkaran di luarnya.

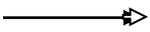
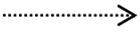
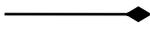
	<i>Activity</i>	Merepresentasikan performa dari beberapa tingkah laku di dalam alur kerja, dinotasikan dengan segiempat
	<i>Fork</i> (Percabangan)	<i>Fork</i> ; Digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel
	<i>Join</i> (Penggabungan)	<i>Join</i> , digunakan untuk menggabungkan dua kegiatan paralel menjadi satu.
	<i>Decision</i>	Menentukan kapan alur dalam aktivitas menjadi bercabang.
	<i>Swimlane</i>	Sebuah cara untuk mengelompokkan <i>activity</i> berdasarkan <i>actor</i> . <i>Actor</i> (Mengelompokkan <i>activity</i> dalam sebuah urutan yang sama)

3) Class Diagram

Diagram Class memberikan pandangan secara luas dari suatu sistem dengan menunjukkan kelas-kelasnya dan hubungan mereka. *Diagram Class* bersifat statis menggambarkan hubungan apa yang terjadi bukan apa yang terjadi jika mereka berhubungan.

Simbol-simbol yang ada pada *diagram class* dapat dilihat pada tabel 5.

Tabel 5. Simbol dalam *Sequence Diagram*

Notasi	Nama	Keterangan
1	2	3
	<i>Asosiasi/ Assosiation</i>	Hubungan statis antar kelas. Asosiasi menggambarkan kelas yang memiliki atribut berupa kelas lain, atau kelas yang harus mengetahui eksistensi kelas lain. Asosiasi biasanya disertai dengan <i>multiplicity</i>
	Asosiasi berarah/ Directed assosiation	Asosiasi dengan makna kelas yang satu digunakan oleh kelas yang lain. Asosiasi berarah juga biasanya disertai dengan <i>multiplicity</i> .
	<i>Generalization</i>	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus) atau untuk menyatakan hubungan <i>inheritance</i> .
	<i>Despendency/ Kebergantungan</i>	Relasi antar kelas dengan makna kebergantungan antar kelas.
	<i>Agregation / Agregasi</i>	Hubungan yang menyatakan bahwa suatu kelas menjadi atribut bagi kelas lain.
	<i>Composition / Komposisi</i>	Bentuk khusus dari agregasi dimana kelas yang menjadi bagian diciptakan setelah kelas menjadi <i>whole</i> dibuat. Misal kelas <i>whole</i> dihapus, maka kelas yang menjadi part ikut musnah.

	<i>Realization</i>	Hubungan antar kelas dimana sebuah kelas memiliki keharusan untuk mengikuti aturan yang ditetapkan oleh kelas lainnya.
---	--------------------	--

2.2.7 PHP (*Hypertext Proprocessor*)

Menurut (Sanjani Lukman, 2014), PHP adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Karena PHP merupakan *server-side scripting* maka *sintaks* dan perintah-perintah PHP akan di esekusi di *server* kemudian hasilnya dikirimkan ke *browser* dalam format HTML.

PHP dirancang untuk membentuk halaman *web* yang dinamis, yaitu halaman *web* yang dapat membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data kehalaman *web*. PHP termasuk dalam *Open Source Product*. Sehingga *source code* PHP dapat diubah dan di distribusikan secara bebas. PHP juga dapat berjalan pada berbagai *web server* seperti IIS (*Internet Information Server*), PWS (*Personal Web Server*), *Apache*, *Xitami*.

PHP juga mampu lintas *platform*. Artinya PHP dapat berjalan di banyak system operasi yang beredar saat ini, diantaranya: Sistem Operasi Windows (semua versi), Linux, Mac OS, Solaris. Salah satu keunggulan yang dimiliki oleh PHP adalah kemampuannya untuk melakukan koneksi keberbagai macam *software* system manajemen basis data atau *Database Management System* (DBMS), sehingga dapat menciptakan suatu halaman *web* dinamis.

Untuk mengenal sintak dalam PHP, tentu saja juga harus sudah mengenal algoritma dalam pemrograman, karena pada dasarnya semua bahasa pemrograman menggunakan algoritma yang sama. Mulai dari pengenalan *variable*, proses pengulangan (*looping*), dan menghasilkan keluaran/*output*. Sintak PHP selalu dimulai dengan `<? Atau <?php` dan diakhiri dengan `?>`. dan di dalam Sintak *php* juga dapat kita sisipkan kode *html* dengan menambahkan perintah *echo* “”;dalam PHP, setiap *variable* diberi tanda *dollar* (\$). Contoh pengenalan *variable* didalam PHP yang dilihat pada Tabel 6.

Tabel 6. Sintak *Variable* pada PHP

No	Sintak	Keterangan
1	<code><?</code>	Awal sintak php
2	<code>\$nama="Budi";</code>	<i>variable</i> nama bernilai string Budi
3	<code>\$usia="15";</code>	<i>Variable</i> usia bernilai integer 12
4	<code>echo "Namaku \$nama, dan usiaku \$usia";</code>	Menampilkan nilai dari <i>variable</i> \$nama dan \$usia
5	<code>?></code>	Akhir sintak php

2.2.8 Website

Menurut (Arief, 2011) *website* adalah salah satu aplikasi yang berisikan dokumen-dokumen multimedia (teks, gambar, suara, animasi, video) didalamnya yang menggunakan protokol HTTP (*HyperText Transfer Protocol*) dan untuk mengaksesnya menggunakan perangkat lunak yang disebut *browser*.

Berdasarkan pendapat yang dikemukakan diatas dapat disimpulkan *website* adalah aplikasi yang berisikan dokumen-dokumen yang tersimpan dalam *server* dan diakses melalui *browser*.

2.2.9 Xampp

Menurut (Buana, 2014), “XAMPP adalah perangkat lunak opensource yang diunggah secara gratis dan bisa dijalankan di semua semua operasi seperti windows, linux, solaris, dan mac”.

Di dalam folder utama xampp, terdapat beberapa folder penting yang perlu diketahui. Untuk lebih memahami setiap fungsinya, Anda dapat melihat penjelasannya sebagai berikut:

Apache	:	Folder utama dari Apache Webserver
Htdocs	:	Folder utama untuk menyimpan data data latihan web, baik PHP maupun HTML biasa
Manual	:	Berisi subfolder yang di dalamnya terdapat manual program dan database, termasuk manual PHP dan MySQL
MySQL	:	Folder utama untuk database MySQL Server. PHP Folder utama untuk program PHP

BAB III METODE PENELITIAN

3.1 Lokasi Penelitian

Penulis membuat proposal penelitian mengenai Sistem Pakar Diagnosa Penyakit Tanaman Kacang Panjang dengan Metode *Certainty Factor* ini berfungsi sebagai media diagnosa penyakit dan cara pencegahan pada tanaman kacang panjang. Lokasi penelitian sebagai bahan penulis mengetahui penyakit dan pencegahan yang sering terjadi pada tanaman kacang panjang dilakukan di Kantor Balai Pengkajian Teknologi Pertanian Kalimantan Tengah.

3.2 Teknik Pengumpulan Data

Penulis menggunakan beberapa tahapan atau metode dalam melakukan penelitian untuk menyusun proposal tugas akhir ini, yaitu :

a. Studi Pustaka

Pada tahap ini penulis mengumpulkan beberapa proposal maupun jurnal penelitian dari sumber lain dengan judul yang menyerupai judul penelitian sebagai sumber referensi yang relevan untuk proposal penelitian yang sedang dilakukan.

b. Metode Dokumentasi

Metode dokumentasi adalah salah satu metode pengumpulan data kualitatif dengan melihat atau menganalisis dokumen-dokumen yang dibuat oleh subjek sendiri atau oleh orang lain oleh subjek.

Dokumentasi merupakan salah satu cara yang dapat dilakukan peneliti kualitatif untuk mendapatkan gambaran dari sudut pandang

subjek melalui suatu media tertulis dan dokumen lainnya yang ditulis atau dibuat langsung oleh subjek yang bersangkutan.

Dengan metode ini, peneliti mengumpulkan data dari dokumen yang sudah ada, sehingga penulis dapat memperoleh catatan-catatan yang berhubungan dengan penelitian. Metode dokumentasi ini dilakukan untuk mendapatkan data-data yang belum didapatkan melalui metode observasi dan wawancara.

c. Metode Wawancara

Pada tahap ini penulis melakukan wawancara berupa pertanyaan pada pihak lokasi penelitian yaitu di kantor Balai Pengkajian Teknologi Pertanian Kalimantan Tengah mengenai penyakit-penyakit yang sering terjadi pada tanaman kacang panjang dan cara pencegahannya.

d. Kuisisioner

Kuisisioner adalah teknik pengumpulan data yang dilakukan dengan cara memberi seperangkat pertanyaan tertulis kepada responden untuk dijawabnya, dapat diberikan secara langsung atau melalui pos atau internet. Jenis angket ada dua, yaitu tertutup dan terbuka. Kuisisioner yang digunakan dalam hal ini adalah kuisisioner tertutup yakni kuisisioner yang sudah disediakan jawabannya, sehingga responden tinggal memilih dan menjawab secara langsung.

3.3 Analisis

3.3.1 Analisis Proses

a. Metode Pengembangan Perangkat Lunak

Adapun proposal penelitian yang sedang dilakukan penulis menerapkan *prototype* sebagai metode pengembangan perangkat lunak karena metode proses pembuatan sistem yang dibuat secara terstruktur dan memiliki beberapa tahap – tahap yang harus dilalui pada pembuatannya. Namun jika tahap final dinyatakan bahwa sistem yang telah dibuat belum sempurna atau masih memiliki kekurangan, maka sistem akan dievaluasi kembali dan akan melalui proses dari awal.

Penulis menerapkan *prototyping* sebagai metode pengembangan sistem karena dalam proses interative yang melibatkan hubungan kerja yang dekat antara perancang dan pengguna.

Selain itu untuk memodelkan sebuah perangkat lunak dibutuhkan beberapa tahapan dalam proses pengembangannya, tahapan inilah yang akan menentukan keberhasilan dari sebuah *software* itu, tahapan – tahapan dalam model *prototype* adalah sebagai berikut :

1) Pengumpulan Kebutuhan

Pada penelitian yang dilakukan di Kantor Balai Pengkajian Teknologi Pertanian Kalimantan Tengah,

penulis melakukan beberapa teknik pengumpulan data kebutuhan, diantaranya :

a) *One-on-one Interview*

Pada teknik one-on-one interview penulis melakukan wawancara maupun pertanyaan pada pihak lokasi penelitian mengenai sistem transaksi yang diterapkan saat ini dan apa saja yang dibutuhkan pada sistem dan yang perlu dikembangkan pada aplikasi yang akan dibangun.

b) *Kuisoner*

Pada teknik kuisoner penulis mengumpulkan data dengan menggunakan lembar teks berupa kuisoner mengenai persyaratan pembangunan sistem, persetujuan dari penulis maupun masukan dari pihak lokasi penelitian dan masukan dari penulis.

Berdasarkan teknik pengumpulan kebutuhan di atas yaitu *one-on-one interview* dan *questionnaires* maka hasil yang diperoleh yaitu dapat membantu penulis untuk menentukan fitur dan menu-menu apa saja yang akan digunakan dalam aplikasi berdasarkan kebutuhan dan pengembangan sistem transaksi.

2) Membangun *Prototyping*

Dalam tahap ini membangun *prototyping* dengan membuat perancangan sementara yang berfokus pada penyajian kepada pengguna aplikasi, diantaranya contoh *interface*, *input* dan *output*, beserta fitur-fitur lain yang telah ditentukan.

3) Evaluasi *Prototyping*

Dalam tahap ini evaluasi dilakukan oleh pengguna aplikasi pada lokasi penelitian apakah sistem yang sudah dibangun sesuai dengan sistem yang diterapkan maupun dikembangkan. Jika hasil dari evaluasi pengguna sudah selesai, maka akan dilanjutkan ke pengkodean sistem, jika tidak maka sistem direvisi kembali ke tahap pengumpulan kebutuhan.

4) Mengkodekan Sistem

Dalam tahap ini sistem yang sudah disepakati oleh pembuat maupun pengguna maka akan dilanjutkan ke bahasan pemrograman yang sesuai dengan pembuat lampiran.

Dalam penelitian ini pembuat menggunakan bahasa pemrograman *PHP* dalam membangun aplikasi dan *MySQL* sebagai *database* maupun *server* penyimpanan data.

5) Menguji Sistem

Setelah sistem sudah menjadi perangkat lunak yang siap dipakai, maka harus di uji coba terlebih dahulu sebelum digunakan. Pengujian ini dilakukan dengan *white box*, *black box*, *basic patch*, pengujian arsitektur, dan lain-lain.

6) Evaluasi Sistem

Dalam tahap ini evaluasi dilakukan oleh pengguna aplikasi pada lokasi penelitian apakah sistem yang sudah dibangun sesuai dengan sistem yang disepakati terlebih dahulu. Jika hasil dari evaluasi pengguna sudah sesuai maka dapat dilanjutkan ke bagian penggunaan aplikasi, jika tidak maka sistem direvisi kembali ke tahap pengkodean sistem untuk menganalisis dan memperbaiki kesalahan yang terjadi ataupun adanya ketidaksesuaian pada aplikasi.

7) Menggunakan Sistem

Perangkat lunak yang telah diuji dan diterima pengguna siap untuk digunakan.

3.3.2 Analisis Kelemahan Sistem

Karena diagnosa penyakit menggunakan referensi buku dan informasi internet maka memakan banyak waktu dalam proses pencairan, sehingga pihak kantor kesulitan mengatasi masalah data tersebut. Selain itu data petani yang meminta hasil konsultasi hanya

diinput atau dicatat secara manual kemungkinan besar akan terjadi kesalahan dalam menuliskan data tersebut.

3.3.3 Analisis Kebutuhan

Kebutuhan sistem haruslah sesuai dengan kondisi dan kemampuan pengguna, maka dari itu penulis yang juga adalah sebagai pembuat atau membangun program ikut serta melibatkan pengguna dalam mencari dan menganalisis kebutuhan-kebutuhan sistem yang menunjang dalam proses perancangan serta membangun Sistem Pakar Diagnosa Penyakit Pada Tanaman Kacang Panjang Dengan Metode *Certainty Factor* Berbasis Web. Adapun kebutuhan sistem yang diperlukan sebagai berikut :

a. Kebutuhan Perangkat Keras

Dalam kebutuhan perangkat keras (*hardware*) yang penulis gunakan dalam pembuatan aplikasi sebagai berikut.

Tabel 7. Kebutuhan Perangkat Keras

No.	Perangkat Keras	Spesifikasi (Type)
1.	Laptop	Asus
2.	Processor	Intel® Celeron® CPU N2840 @ 2.16 GHz
3.	Memory	4 GB DDR3L
4.	Disk	500 GB
5.	Graphic	Intel® HD Graphics

b. Kebutuhan Perangkat Lunak

Dalam kebutuhan perangkat lunak (*software*) yang penulis gunakan dalam pembuatan aplikasi sebagai berikut.

Tabel 8. Kebutuhan Perangkat Lunak

No.	Perangkat Lunak	Sepesifikasi
1.	Notepad++	Sebagai perangkat lunak dalam pembuatan aplikasi sistem diagnosa penyakit tanaman kacang panjang
2.	MySQL	Sebagai perangkat lunak penyimpan database

c. Kebutuhan Informasi

Penulis menggunakan beberapa tahapan atau metode dalam melakukan penelitian untuk mendapatkan informasi pada penyusunan proposal tugas akhir ini diantaranya yaitu wawancara dan studi pustaka.

d. Kebutuhan Pengguna

Untuk pengguna sistem pada aplikasi ini terdiri dari 2 pengguna yaitu admin dan operator.

3.3.4 Analisis Kelayakan Sistem

Adapun beberapa kriteria analisis kelayakan sistem dalam penelitian ini meliputi sebagai berikut :

a. Kelayakan Teknologi

Kelayakan teknologi yang diberikan aplikasi ini adalah sebagai perangkat lunak yang dapat memberikan kemudahan untuk membantu pengguna atau instansi pada tempat penelitian dalam mendiagnosa penyakit pada tanaman kacang panjang.

b. Kelayakan Hukum

Untuk kelayakan hukum aplikasi ini semua menggunakan perangkat lunak yang gratis atau open source sehingga tidak ada masalah dengan pelanggaran hukum atau software bajakan. Perangkat lunak yang dimaksud dalam pembuatan aplikasi ini yaitu seperti Notepad++ dan *database* MySQL.

c. Kelayakan Operasional

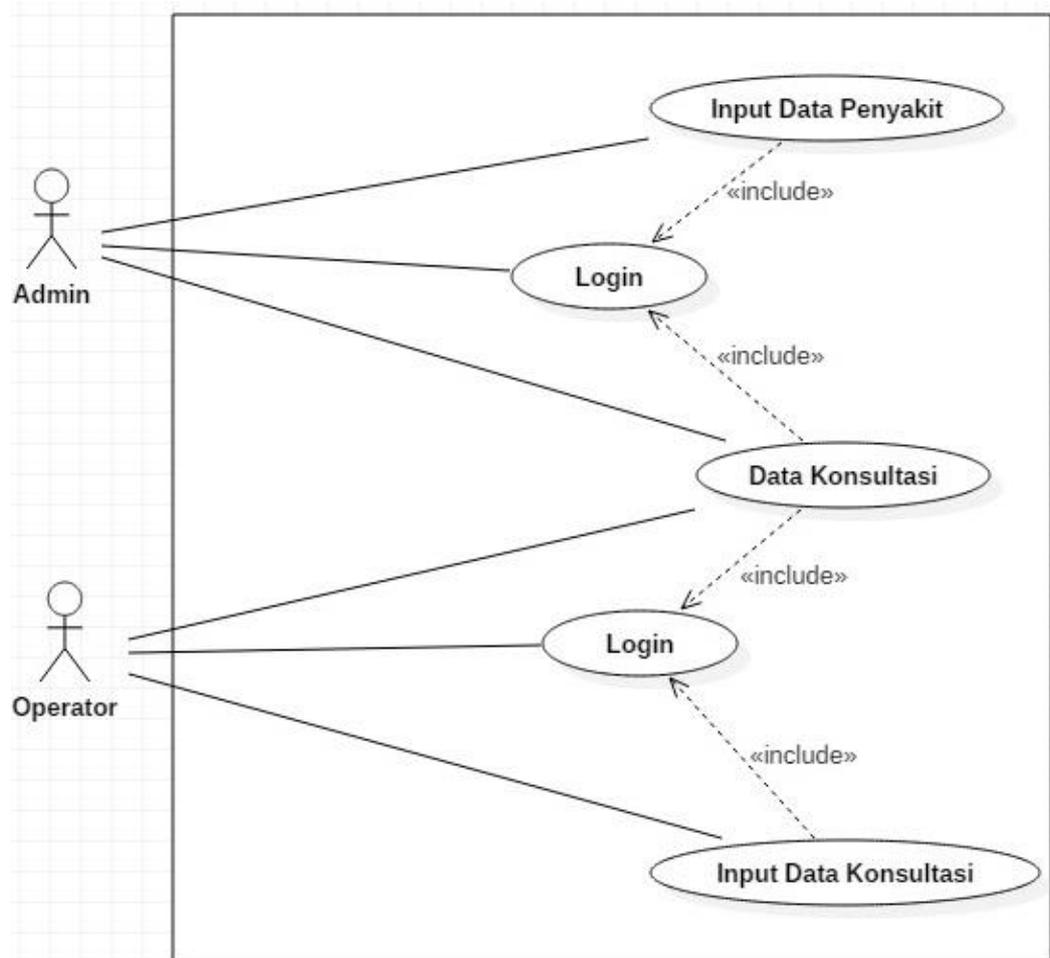
Mengenai kelayakan operasional, aplikasi yang dibangun dirancang lebih sederhana agar pengguna aplikasi dapat lebih mudah memahami saat menggunakan aplikasi tersebut sesuai dengan hak akses dan perangkat masing-masing.

3.4 Desain Sistem

3.4.1 Desain Proses

a. *Use Case* Diagram

Diagram *use case* disini merupakan gambaran dari *user* yang menggunakan sistem dan perilaku *user* terhadap system



Gambar 2. Use Case Diagram

Pada gambar 2 di atas dijelaskan bahwa terdapat 2 kategori pengguna yaitu *admin* dan *operator* yang mana disetiap pengguna tersebut memiliki hak akses masing-masing. Berikut penjelasan mengenai masing-masing kategori dan rancangan dari *use case* diagram di atas.

Admin : *Admin* bersifat *full* akses dimana *admin* bisa mengakses menu yang tidak bisa dilakukan oleh *operator* seperti menginput data penyakit, dan data konsultasi.

Operator : Operator memiliki peran akses terbatas yang hanya dapat melakukan tindakan sebagai operator seperti melakukan proses datas konsultasi dan input data konsultasi.

Pada gambar *use case* diagram di atas dijelaskan bahwa admin dan operator harus *login* terlebih dahulu sesuai dengan nama pengguna dan kata sandi masing-masing yang sebelumnya sudah ditentukan atau didaftarkan.

Jika *login* sebagai *admin* maka dapat membuka menu *input* data penyakit dan hasil konsultasi, seperti gambar *use case* di atas.

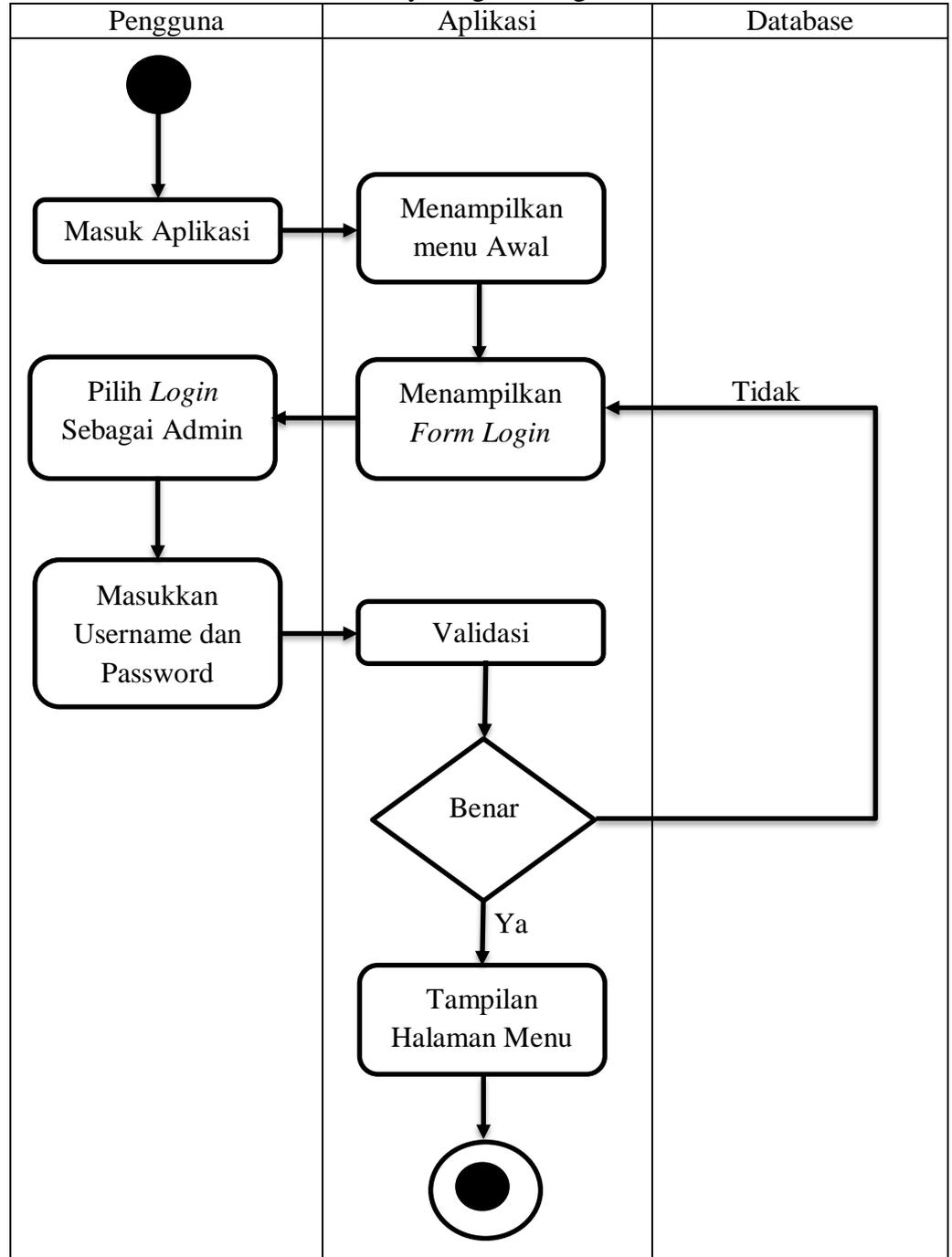
Jika *login* sebagai operator maka seperti diagram *use case* di atas hanya dapat melakukan proses konsultasi dan hasil konsultasi.

b. *Activity Diagram*

Activity diagram disini merupakan gambaran alur proses atau cara kerja sistem. Pada diagram ini digambarkan aktivitas-aktivitas apa saja yang dikerjakan oleh sebuah sistem.

1) Activity Diagram Menu Login Admin

Tabel 9. Activity Diagram Login Admin

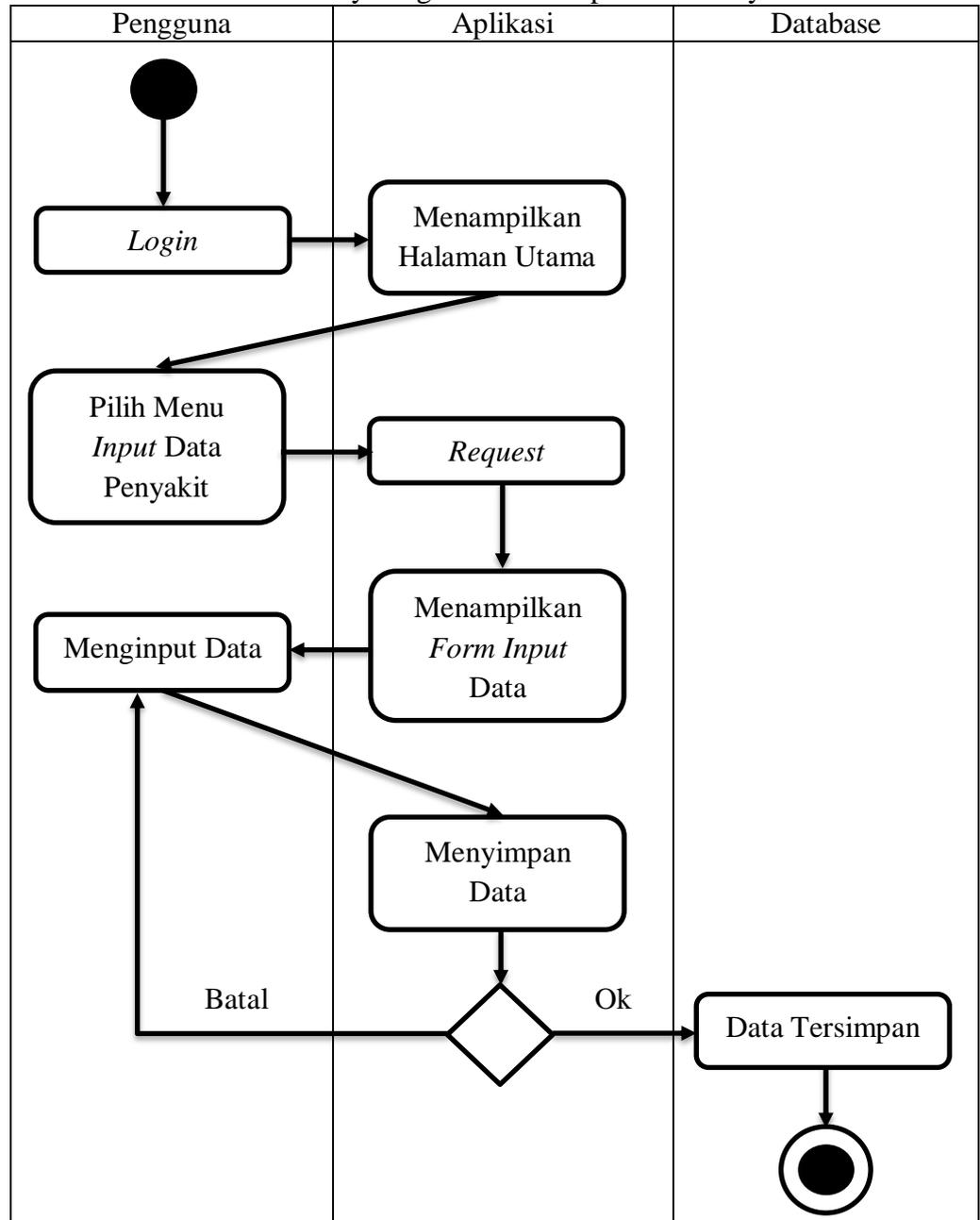


Pada tabel 9 activity diagram menu login di atas dijelaskan bahwa saat admin atau operator membuka aplikasi

akan diarahkan ke halaman login kemudian mengisi nama pengguna dan kata sandi secara otomatis masuk ke form halaman utama apabila nama pengguna dan kata sandi benar. Apabila nama pengguna dan kata sandi salah, maka tampilan aplikasi akan kembali ke form login.

2) Activity Diagram Menu Input Data Penyakit

Tabel 10. Activity Diagram Menu Input Data Penyakit

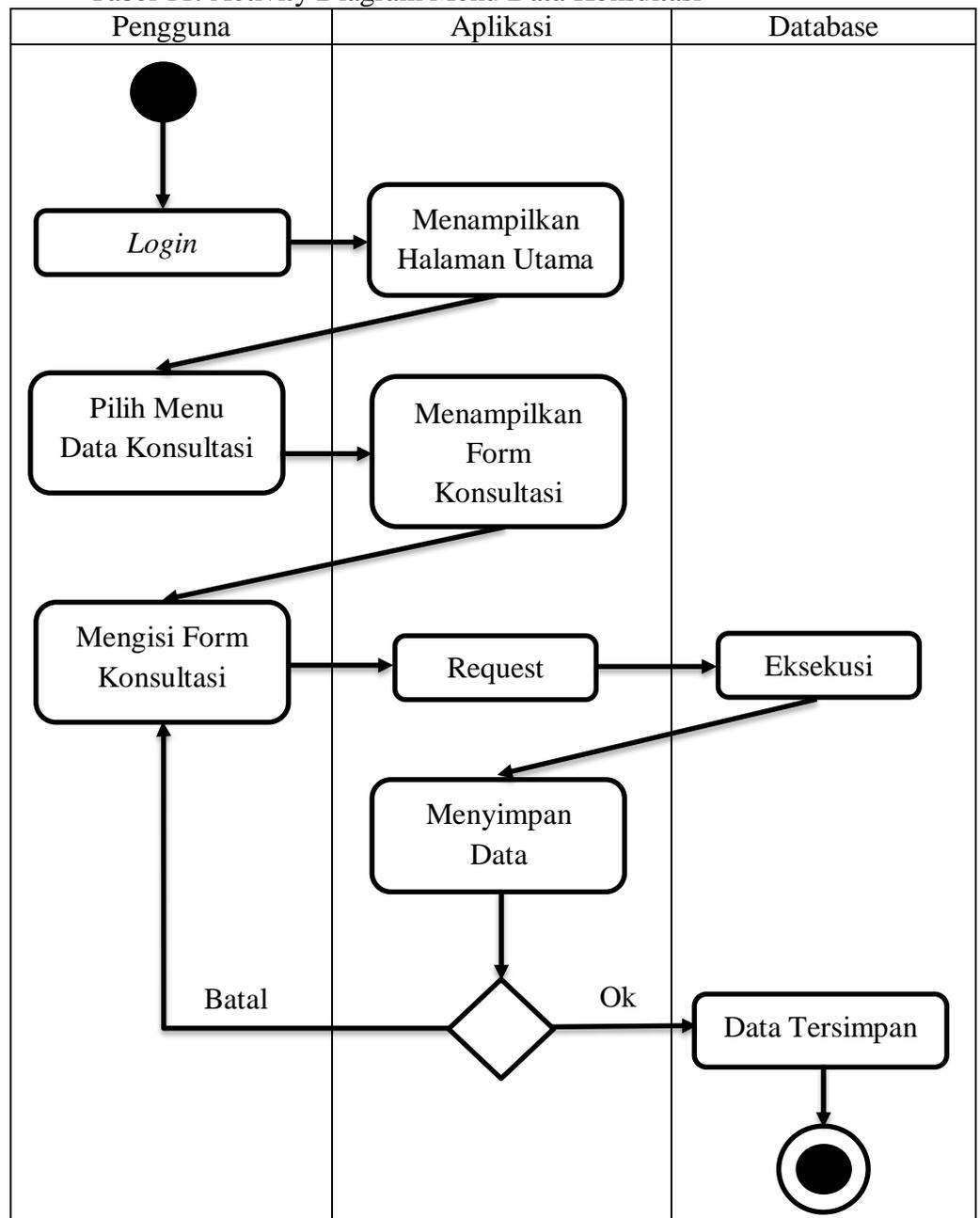


Pada tabel 10 activity diagram di atas dijelaskan bahwa saat Admin menambah data penyakit dengan cara memilih menu data penyakit. Sistem kemudian akan menampilkan halaman

data penyakit. Kemudian admin memilih input data penyakit. Pada halaman ini admin mengisi *form* sampai selesai dan menekan tombol simpan.

3) Activity Diagram Menu Data Konsultasi

Tabel 11. Activity Diagram Menu Data Konsultasi



Pada tabel 11 activity diagram di atas dijelaskan bahwa saat Admin menambah data pengguna dengan cara memilih menu data pengguna. Sistem kemudian akan menampilkan halaman data pengguna. Kemudian admin memilih tambah data pengguna. Pada halaman ini admin mengisi *form* sampai selesai dan menekan tombol simpan.

c. Tabel Basis Data

Database tb_ta.sql adalah database yang dibuat untuk menyimpan data-data yang akan digunakan dalam proses pengolahan data admin, data gejala, data diagnosa, dan data input penyakit dan data konsultasi. Berikut adalah desain dari tabel-tabel yang di buat di dalam database tb_ta.sql

1) Admin

Admin merupakan tabel yang menyimpan data admin yang digunakan untuk login admin pada sistem. Berikut spesifikasi dari admin:

Tabel 12. Spesifikasi admin

Kolom	Jenis	Tak ternilal
Admin_Username	Varchar (30)	Tidak
Admin_Password	Varchar (30)	Tidak

2) Desain Tabel Gejala

Tabel gejala adalah tabel yang berisi gejala penyakit tanaman kacang Panjang. Berikut spesifikasinya:

Tabel 13. Spesifikasi gejala

Kolom	Jenis	Tak Ternilai
Id_gejala	Int (11)	Tidak
Kode_gejala	Varchar (30)	Tidak
Nama_gejala	Varchar (30)	Tidak

3) Desain Tabel diagnosa

Tabel diagnosa adalah tabel yang berisi diagnosa penyakit tanaman kacang. Berikut spesifikasinya:

Tabel 14. Spesifikasi diagnosa

Kolom	Jenis	Tak Ternilai
Id_diagnosa	Int (11)	Tidak
Kode_diagnosa	Varchar (30)	Tidak

4) Desain Tabel Menentukan

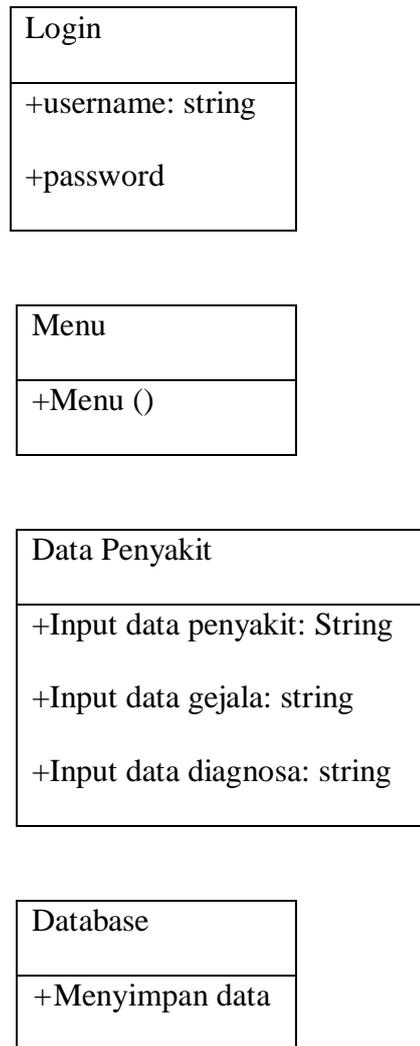
Tabel menentukan ini berisi tentang pertanyaan seputar penyakit tanaman kacang Panjang.

Tabel 15. Tabel Menentukan

Kolom	Jenis	Tak ternilai
Id_penyakit	Int (11)	Tidak
Id_gejala	Int (11)	Tidak
Id_diagnosa	Int (11)	Tidak
Kode_penyakit	Int (11)	Tidak
Kode_gejala	Int (11)	Tidak
Kode_diagnosa	Int (11)	Tidak

d. *Class Diagram*

Class Diagram adalah salah satu jenis diagram yang paling berguna di UML, hal ini karena dapat dengan jelas memetakan struktur sistem tertentu dengan memodelkan kelas, atribut, operasi serta hubungan antar objek.

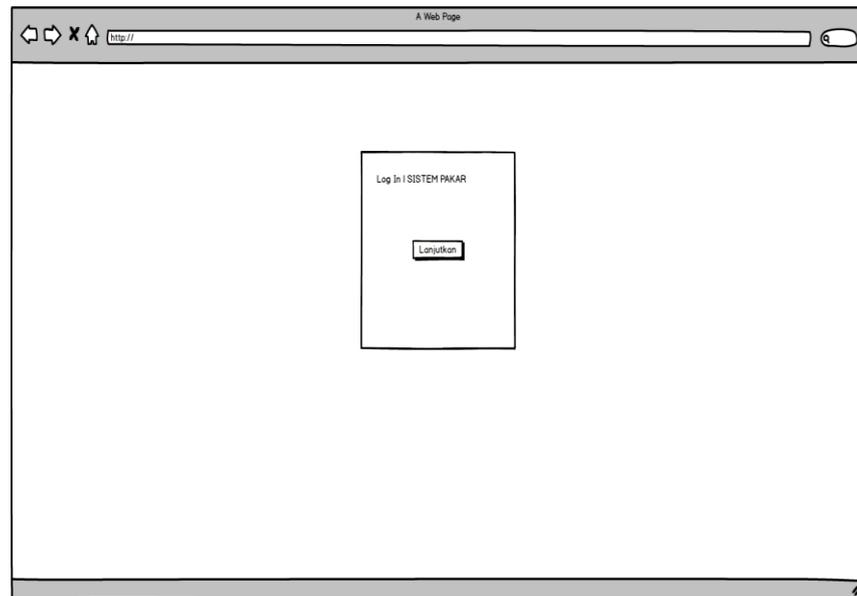


Gambar 3. Class Diagram

3.4.2 Desain Perangkat Lunak

a. Tampilan Halaman *Login*

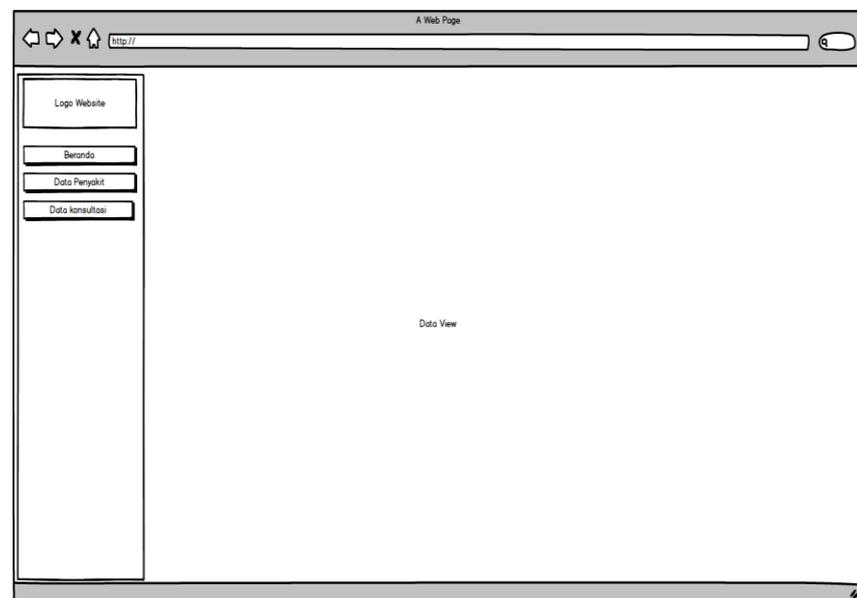
Di halaman *login*, pengguna harus mengisi nama pengguna dan kata sandi untuk masuk ke halaman menu utama atau beranda.



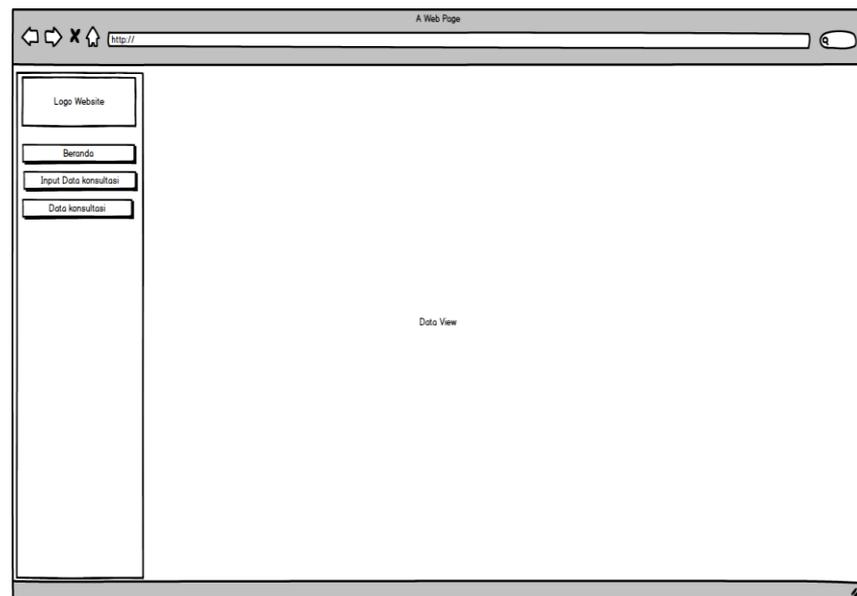
Gambar 4. Layout Halaman Login

b. Tampilan Menu Utama

Sebelum login, pengguna akan ditampilkan menu awal dengan menu bar di pojok kiri.



Gambar 5. Layout Halaman Utama Aplikasi untuk Admin



Gambar 6. Layout Halaman Utama Aplikasi untuk Operator

3.4.3 Desain Keamanan

Desain keamanan dengan menggunakan sistem login (login, juga biasa disebut sebagai log in, log on, signon, sign on, signin, sign in) adalah proses untuk mengakses sebuah sistem dengan memasukkan identitas dari akun pengguna dan kata sandi untuk mendapatkan hak akses dalam menggunakan sistem. Untuk melakukan login ke sistem biasanya membutuhkan akun pengguna dan kata sandi. Akun pengguna dan kata sandi tersebut harus tepat dan keduanya adalah pasangan yang tidak bisa dipisahkan dan kata sandi dapat diubah sesuai dengan kebutuhan.

DAFTAR PUSTAKA

- Adi, N., 2013. REKAYASA PERANGKAT LUNAK BERORIENTASI OBJEK DENGAN METODE USDP. *Andi Offset*, Volume 2, pp. 35-40.
- Agustin Maulina, B. A. & D.S, H., 2016. SISTEM PAKAR DIAGNOSIS HAMA DAN PENYAKIT TANAMAN KACANG TANAH BERBASIS DESKTOP DENGAN METODE BACKWARD CHAINING. *Media Jurnal Informatika*, Volume 8, pp. 25-32.
- Arief, 2011. PEMROGRAMAN WEB DINAMIS MENGGUNAKAN PHP DAN MYSQL. *Andi Offset*, Volume 1, pp. 28-32.
- Buana, 2014. SISTEM PENILAIAN TUGAS AKHIR BERBASIS WEB DI FAKULTAS TEKNIK. *Universitas Islam Majapahit*, Volume 1, pp. 1-8.
- Dewi, A., 2013. JURNAL SISTEM PAKAR DIAGNOSA KERUSAKAN PADA PRINTER DENGAN MENGGUNAKAN METODE CERTAINTY FACTOR. *Universitas Islam Negeri Sunan Gunung Djati*, Volume 2, pp. 21-25.
- Harmanti, T., 2013. SISTEM PAKAR PENELUSURAN PENYAKIT PADA TANAMAN KACANG PANJANG (VIGNA SESQUIPEDALIS). *Jurnal Sistemasi*, Volume 2, pp. 23-29.
- Madcom, 2016. PEMROGRAMAN PHP DAN MYSQL UNTUK PEMULA. *Andi*, Volume 1, pp. 23-30.
- Rahmat Hidayat, N. M., 2014. PERANCANGAN SISTEM PAKAR. *Ghalia Indonesia*, Volume 2, pp. 23-25.
- Sanjani Lukman, A., 2014. JURNAL SISTEM INFORMASI (RANCANG BANGUN SISTEM INFORMASI PENGGAJIAN PEGAWAI DAN REMUNERASI JASA MEDIS PADA RUMAH SAKIT BEDAH). *STMIK STIKOM Surabaya*, Volume 1, pp. 32-36.
- Widyanto, A., 2014. SISTEM PAKAR UNTUK MENDIAGNOSA HAMA DAN PENYAKIT TANAMAN PEPAYA MENGGUNAKAN METODE FORWARD CHAINING. *Universitas Dian Nuswantoro*, Volume 1, pp. 30-35.

LAMPIRAN

Lampiran 1. Surat Tugas Dosen Pembimbing

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA**
Jl. G. Obos No.114 Telp. 0536-3225515 Fax. 0536-3225515 Palangkaraya
surel (email) : humas@stmikplk.ac.id – laman (website) : www.stmikplk.ac.id

SURAT TUGAS
No.182/STMIK-3.C.2/AU/X/2020

Ketua Program Studi Teknik Informatika Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Palangkaraya menugaskan nama- nama tersebut di bawah ini :

1. Nama : Sulistyowati, S.Kom., M.Cs.
NIK : 198212162007002
Sebagai Pembimbing I Dalam Pembuatan Program
2. Nama : Sherly Jayanti, S.T., M.Cs.
NIK : 198501102012004
Sebagai Pembimbing II Dalam Penulisan Tugas Akhir

Untuk membimbing Tugas Akhir mahasiswa :

Nama : Muhammad Khoirul
NIM : C1755201064
Program Studi : TEKNIK INFORMATIKA (55201)
Tanggal Daftar : 12 September 2020
Judul Tugas Akhir : Sistem Pakar Diagnosa Penyakit Tanaman Kacang Panjang Dengan Menggunakan Metode Certainty Factor Berbasis Web

Demikian surat ini dibuat agar dapat dipergunakan sebagaimana mestinya dan dilaksanakan dengan penuh tanggung jawab.

Palangka Raya, 10 Oktober 2020
Ketua Program Studi,

Hotmian Sitohang, M.Kom.
NIK. 198503282008002

Tembusan :

1. Pembimbing I dan II
2. Mahasiswa yang bersangkutan
3. Arsip

Lampiran 2. Kartu Kegiatan Konsultasi Tugas Akhir



**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA**
 Jl. G. Obos No.114 Telp. 0536-3225515 Fax. 0536-3225515 Palangkaraya
 surel (email) : humas@stmikpk.ac.id – laman (website) : www.stmikpk.ac.id

**KARTU KEGIATAN KONSULTASI
TUGAS AKHIR**

Nama Mahasiswa : MUHAMMAD KHOIRUL.....
 NIM : C175520106A.....
 Tanggal Persetujuan Judul :
 Judul Tugas Akhir : Sistem Pakar Diagnosa Penyakit Tanaman.....
 Karang Panjang dengan menggunakan metode
 Certainty Factor Berbasis web.....

No.	Tanggal Konsultasi		Uraian	Tanda Tangan
	Terima	Kembali		
			Tempat penelitian harus ada di latar belakang lalu alasnya hrs disebutkan Tulisannya.	ST
			banyak tulisan asing tidak dicetak miring tolong di cek. Bahasan mengenai certainty factor ada dimana. Jenis Penyakit apa saja sebutkan. Tahun diurutkan berdasarkan tahun tercama.	
			Cek tulisan I, pd penelitian relevan Topik itu bkn Judul. Masukkan Desain UMLnya yg lengkap. ditentukan proses yg ada di UML nya lengkap. class Diagram & Rane Databasenya	ST ST
	3/21-2021		- Pada latar belakang Lemahkan alasan tentang diagnosa di pulitik untuk pengguna sosh palar.	ST
	15/01-2021		- Penulisan sumber pengutipan kayak teori, seminar dan pedoman penulisan TA.	ST
	25/02-2021		- Lengkap berkas laporan yang diperlukan (baik penelitian, bukti wawancara/observasi, dll) ACC usulan proposal TA!	ST
	2/02-2021		Stahkan di perbaiki u/ Desain nya, Desain proses, data base dan class nya	ST
	3/02-2021		Stahkan daftar Summar	ST

Lampiran 3. Surat Tugas Penguji Seminar Tugas Akhir

 **SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA**
Jl. G. Obos No.114 Telp.0536-3224593, 3225515 Fax.0536-3225515 Palangkaraya
email : humas@stmikpik.ac.id - website : www.stmikpik.ac.id

SURAT TUGAS
PENGUJI SEMINAR PROPOSAL TUGAS AKHIR
No.44/STMIK-3.C.2/AK/III/2021

Ketua Program Studi Teknik Informatika Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Palangkaraya menugaskan kepada nama-nama berikut :

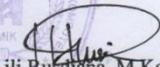
1. Nama : Lili Rusdiana, M.Kom
NIK : 198707282011007
Sebagai Ketua
2. Nama : Sulistyowati, S.Kom., M.Cs.
NIK : 198212162007002
Sebagai Sekretaris
3. Nama : Sherly Jayanti, S.T., M.Cs.
NIK : 198501102012004
Sebagai Anggota

Tim Penguji Seminar Proposal Tugas Akhir :

Nama : Muhammad Khoirul
NIM : C1755201064
Hari/Tanggal : Senin, 8 Maret 2021
Waktu : 10.00 WIB
Judul Proposal : Sistem Pakar Diagnosa Penyakit Tanaman Kacang Panjang Dengan Menggunakan Metode Certainty Factor Berbasis Web

Demikian surat ini dibuat agar dapat dipergunakan sebagaimana mestinya dan dilaksanakan dengan penuh tanggung jawab.

Palangka Raya, 6 Maret 2021
Ketua Program Studi Teknik Informatika


Lili Rusdiana, M.Kom
NIK. 198707282011007

Tembusan :

1. Dosen Penguji
2. Mahasiswa yang Bersangkutan
3. Arsip Prodi

Lampiran 4. Daftar Hadir Peserta Seminar Proposal Tugas Akhir

**DAFTAR HADIR PESERTA
SEMINAR PROPOSAL TUGAS AKHIR**

1. Nama Penyaji : MUHAMMAD KHORUL
 2. Hari/ Tanggal : Senin, 08-Maret 2021
 3. Waktu : 10.00 WIB - Selesai
 4. Judul Proposal : Sistem Pakar Diagnosa Penyakit tanaman
 kacang panjang dengan metode Certainty factor
 berbasis web

No.	Nama Mahasiswa	NIM	Tanda Tangan
1	Wibodo Sindu	C1755201033	
2	Kevin Johannes	C17552010	
3	Jamalul Inan	C1755201052	
4	Putra .A. Makhuda	C1755201067	
5	M. kahfi A	C1755201076	
6	Kevin Johannes	C1755201070	
7	Rahmad Arif Kurniawan	C1755201015	
8	Irwan	C1855201017	
9	Al Fahmi	C1657201057	
10	Ramadinato Bagaskoro	C1755201010	
11	Jusep Valentino	C1755201050	
12	Octora	C1855201006	
13	Novi Vebriyanti	C1755201065	
14	Martin	C1755201079	
15			
16			
17			
18			
19			
20			
21			
22			
23			
24			
25			
26			
27			
28			
29			
30			

Palangka Raya, 08 Maret 2021.....

Mengetahui :
Ketua Tim Penguji,

Ulil Pusdiana

Mahasiswa Penyaji,

MUHAMMAD KHORUL

Lampiran 5. Berita Acara Seminar Proposal Tugas Akhir



SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA
Jl. G. Obos No.114 Telp.0536-3224593, 3225515 Fax.0536-3225515 Palangkaraya
email : humas@stmikplk.ac.id - website : www.stmikplk.ac.id

**BERITA ACARA
SEMINAR PROPOSAL TUGAS AKHIR**

Periode (Bulan) : maret Tahun 2021

1. Hari/Tanggal Seminar : senin 108 - maret 2021
2. Waktu (Jam) : 08.00 WIB sampai dengan WIB
3. Nama Mahasiswa : MUHAMMAD HIDIRUL
4. Nomor Induk Mahasiswa : 175520064
5. Program Studi : Teknik Informatika
6. Tahun Angkatan : 2017
7. Judul Tugas Akhir : SISTEM PAPAN DIAGNOSA PENYAKIT TAPAKAN KECAPG PANJANG DENGAN METODE CERTAINTY FACTOR BERBASIS WEB
8. Dosen Penguji :

Nama	Nilai	Tanda Tangan
1. Lili Rusdiana, M.Kom	<u>2</u>	
2. Sulistyowati, S.Kom, M.LS	<u>2</u>	
3. Sberly Jayanti, S.T., M.UC	<u>2</u>	
9. Hasil Ujian : **LULUS / TIDAK LULUS*)** NILAI = 81,2
Dengan Perbaikan/ Tanpa Perbaikan*)
10. Catatan Penting :
 1. Lama Perbaikan : 14 hari (Maks. 15 hari)
 2. Jika lebih dari 15 hari s/d 1 (satu) bulan dikenakan sanksi berupa denda sebesar Rp. 300.000,- (Tiga ratus ribu rupiah), dan jika lebih dari 1 (satu) bulan dikenakan denda Rp. 600.000,- (Enam Ratus ribu rupiah) per bulan dari tanggal ujian
 3. Jika lebih dari 3 (tiga) bulan dari tanggal ujian maka hasil ujian dibatalkan dan wajib mengajukan judul dan pembimbing baru. Wajib membayar Denda dan membayar biaya seminar ulang.

Palangka Raya, senin, 08 - maret 2021

Mengetahui :
Ketua Program Studi Teknik Informatika,

Lili Rusdiana, M.Kom.
NIK. 198707282011007

Ketua Penguji,

Lili Rusdiana
NIK.

Tembusan :

1. Arsip Prodi Teknik Informatika
 2. Mahasiswa yang bersangkutan
- Dibawa saat konsultasi perbaikan dengan dosen penguji

*) Coret yang tidak perlu