

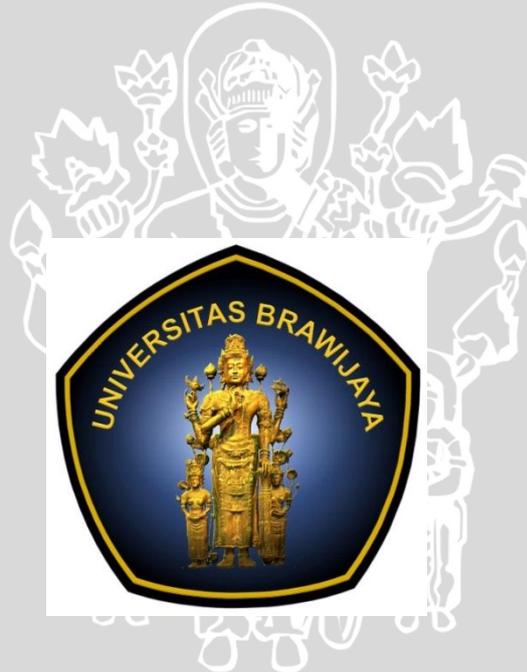
**PEMODELAN SISTEM PAKAR UNTUK IDENTIFIKASI PENYAKIT
PADA TANAMAN KEDELAI MENGGUNAKAN
METODE *FUZZY K-NEAREST NEIGHBOR***

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Romantika Mayang Asri
NIM: 125150201111015



TEKNIK INFORMATIKA
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2016

PENGESAHAN

PEMODELAN SISTEM PAKAR UNTUK IDENTIFIKASI PENYAKIT PADA TANAMAN
KEDELAI MENGGUNAKAN METODE *FUZZY K-NEAREST NEIGHBOR*

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :

Romantika Mayang Asri

NIM: 125150201111015

Skripsi ini telah diuji dan dinyatakan lulus pada
10 Maret 2016

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Nurul Hidayat, S.Pd, M.Sc
NIP: 19680430 200212 1 001

M. Ali Fauzi, S.Kom, M.Kom
NIK: 201502 890101 1 000

Mengetahui
Ketua Program Studi NamaProgramStudi

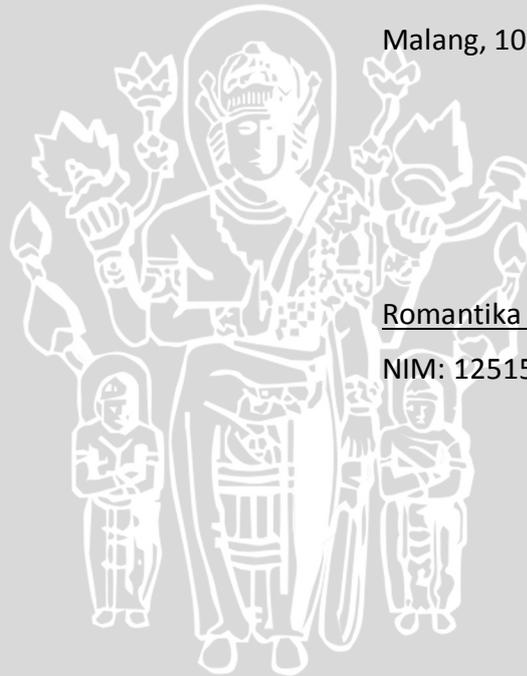
Drs. Marji, MT
NIP: 19670801 199203 1001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 10 Maret 2016



Romantika Mayang Asri

NIM: 125150201111015

KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadirat Tuhan Yang Maha Esa, yang telah melimpahkan berkat dan anugerah-Nya kepada penulis sehingga penulis dapat menyelesaikan skripsi ini yang berjudul “**PEMODELAN SISTEM PAKAR UNTUK IDENTIFIKASI PENYAKIT PADA TANAMAN KEDELAI MENGGUNAKAN METODE FUZZY K-NEAREST NEIGHBOR**” dengan baik.

Dalam penyusunan skripsi ini penulis banyak terlibat dengan berbagai pihak. Oleh karena itu, penulis ucapkan terimakasih kepada:

1. Orang tua, kakak dan adik penulis yang telah memberikan dukungan secara motivasi, doa dan material.
2. Nurul Hidayat, S.Pd, M.Sc. Selaku dosen pembimbing pertama yang telah meluangkan waktu untuk membantu dan membimbing penulis dalam pengerjaan tugas akhir ini.
3. M. Ali Fauzi, S.Kom, M.Kom. Selaku dosen pembimbing kedua yang telah meluangkan waktu untuk membantu dan membimbing penulis dalam pengerjaan tugas akhir ini.
4. Prof. (Riset). Dr. Ir, Moh. Cholil Mahfud, M.S selaku pakar tanaman kedelai yang telah meluangkan waktunya, membantu dan berbagi ilmu dalam penyelesaian tugas akhir ini.
5. Seluruh civitas akademika informatika/ilmu komputer Universitas Brawijaya terutama yang telah banyak membantu dan memberi dukungan selama penulisan skripsi ini.
6. Nenovli, Efi, A'yun, Fakhri, Iqbal, Wicky, Faisal, Mega, Qoirul dan keluarga Besar Mahasiswa Informatika/Illmu Komputer khususnya angkatan 2012.
7. Witri, Mutiara, Wati, Apip, Mbak Ninda, Windy, Anita, Vina, Audy, Ersya, Dini, Addah, Kiky, Fi'ah, Atun, Tiara, Ade atas motivasi dan nasihatnya selama ini.
8. Semua pihak yang telah membantu penulis dalam pengerjaan skripsi yang tidak bisa disebutkan satu persatu.

Dengan segala kerendahan hati, dalam penyusunan skripsi ini penulis menyadari masih terdapat banyak kekurangan dan kesalahan, maka dari itu penulis mengharapkan saran dan kritik yang bersifat membangun. Akhir kata, semoga skripsi ini dapat bermanfaat bagi kita semua, terima kasih.

Malang, 10 Maret 2016

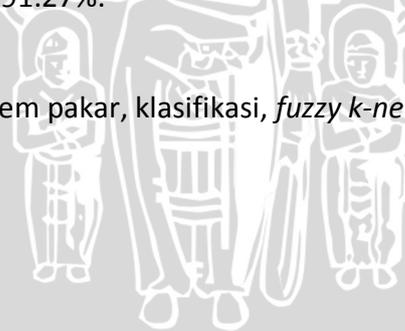
Penulis

maayaang@gmail.com

ABSTRAK

Kedelai merupakan salah satu sumber komoditas pangan utama di Indonesia yang tidak hanya berfungsi sebagai bahan baku industri pangan tetapi juga industri non-pangan. Namun untuk meningkatkan produksi kedelai bukanlah hal yang mudah. Salah satu faktor yang menghambat produksi kedelai adalah penyakit yang menyerang tanaman kedelai. Jenis penyakit yang beraneka ragam membutuhkan penanganan yang berbeda-beda. Keterbatasan pakar dan minimnya pengetahuan petani tentang gejala dan jenis penyakit yang menyerang tanaman kedelai menjadi permasalahan yang berdampak negatif pada produksi tanaman kedelai. Untuk mengatasi hal tersebut dibutuhkan suatu sistem yang dapat mengadopsi pengetahuan pakar. Pemodelan sistem pakar pada penelitian ini menggunakan metode *fuzzy k- nearest neighbor* yang merupakan metode klasifikasi yang yang digunakan untuk memprediksi data uji menggunakan nilai derajat keanggotaan terbesar dari data uji pada setiap kelas sebagai sebagai kelas hasil prediksi. Sistem pakar ini menggunakan 16 parameter gejala input dan 5 kelas penyakit. Berdasarkan hasil akurasi pengujian pengaruh nilai k klasifikasi didapatkan rata-rata nilai akurasi sebesar 97.5% , hasil pengujian pengaruh variasi jumlah data didapatkan rata-rata nilai akurasi sebesar 91.57%, hasil pengujian perbandingan untuk pengaruh nilai k klasifikasi pada metode *fuzzy k-nearest neighbor* didapatkan nilai rata-rata sebesar 97.5% dan pada metode k-nn sebesar 93.1%, hasil pengujian perbandingan untuk pengaruh variasi jumlah data pada metode *fuzzy k-nearest neighbor* didapatkan nilai rata-rata sebesar 91.57% dan pada metode k-nn sebesar 91.27%.

Kata kunci: Pemodelan sistem pakar, klasifikasi, *fuzzy k-nearest neighbor*, kedelai.



ABSTRACT

Soybeans are one of the source of main food comodities in Indonesia, which not only be functioning as raw material for the industrial food but also for non-industrial food. But to increase the production of soybeans are not easy. One of the factors that can obstruct the production of soybeans are soybeans diseases. Varied types of diseases need the different handling. The limitations of expert and lack knowledge of farmers about the symptoms and diseases that attack the soybeans become a problem that have negative impact on the production of soybeans. To overcome this problem needed the system that can adopt the expert knowledge. Expert system modeling in this study using fuzzy k-nearest neighbor method which is classification method that used to prediction the test data using the greatest degree of membership value of the test data for each class as class prediction results. This expert system uses 16 parameters input symptoms and 5 grades of disease. Based on the results of accuracy value of k classification obtained an average accuracy value of 97.5%, and the results of accuracy of variation the amount of data obtained an average accuracy value of 91.57%, the results of comparative accuracy value of k classification for fuzzy k-nearest neighbor method obtain an average accuracy value 97.5% and for k-nearest neighbor method obtain an average accuracy value 93.1% and the results of comparative accuracy of variation the amount of data for fuzzy k-nearest neighbor method obtain an average accuracy value 91.57% and for k-nearest neighbor method obtain an average accuracy value 91.27%

Keywords: *Expert system modelling, classification, fuzzy k-nearest neighbor, soybeans.*

DAFTAR ISI

PENGESAHAN.....	ii
PERNYATAAN ORISINALITAS.....	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
ABSTRACT.....	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xii
DAFTAR SOURCE CODE.....	xiv
DAFTAR LAMPIRAN.....	xv
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan Penelitian.....	3
1.4 Manfaat Penelitian.....	3
1.5 Batasan Masalah.....	3
1.6 Sistematika Pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN.....	5
2.1 Kajian Pustaka.....	5
2.2 Tanaman Kedelai.....	6
2.2.1 Sejarah Tanaman Kedelai.....	6
2.2.2 Taksonomi Tanaman Kedelai.....	10
2.2.3 Morfologi Tanaman Kedelai.....	10
2.2.4 Kegunaan dan Manfaat Tanaman Kedelai.....	12
2.2.5 Penyakit Tanaman Kedelai.....	12
2.3 Sistem Pakar.....	17
2.3.1 Definisi Sistem Pakar.....	17
2.3.2 Manfaat Sistem Pakar.....	17
2.3.3 Kekurangan Sistem Pakar.....	18
2.3.4 Ciri-ciri Sistem Pakar.....	18

2.3.5 Struktur Sistem Pakar	18
2.4 Logika Fuzzy	20
2.4.1 Pengertian Logika Fuzzy	20
2.4.2 Himpunan <i>Fuzzy</i>	21
2.5 Klasifikasi	22
2.6 Inisialisasi <i>Fuzzy</i>	22
2.7 <i>K-Nearest Neighbor</i>	23
2.8 Fuzzy <i>K-Nearest Neighbor</i>	24
2.8.1 Contoh Perhitungan Manual Fuzzy <i>K-Nearest Neighbor</i>	25
2.9 Akurasi	30
BAB 3 METODOLOGI PENELITIAN	31
3.1 Studi Literatur	31
3.2 Pengumpulan Data	32
3.3 Analisis Kebutuhan	32
3.4 Perancangan Sistem	32
3.5 Implementasi Sistem Pakar	33
3.6 Pengujian Sistem Pakar	34
3.7 Pengambilan Kesimpulan	34
BAB 4 PERANCANGAN	35
4.1 Analisa Kebutuhan Perangkat Lunak	35
4.1.1 Identifikasi Aktor	36
4.1.2 Analisa Kebutuhan Masukan	36
4.1.3 Analisa Kebutuhan Proses	36
4.1.4 Analisa Kebutuhan Keluaran	36
4.2 Perancangan Perangkat Lunak	37
4.2.1 Proses Klasifikasi	37
4.3 Perancangan Sistem	43
4.3.1 Akuisisi Pengetahuan	44
4.3.2 Basis Pengetahuan	46
4.3.3 Mesin Inferensi	47
4.3.4 Perhitungan Manual	47
4.3.5 Perancangan Antarmuka	54

4.3.6 Perancangan Pengujian.....	55
BAB 5 IMPLEMENTASI	57
5.1 Spesifikasi Sistem.....	58
5.1.1 Spesifikasi Perangkat Keras	58
5.1.2 Spesifikasi Perangkat Lunak.....	58
5.2 Batasan Implementasi.....	58
5.3 Implementasi Algoritma.....	59
5.3.1 Implementasi Fuzzy.....	59
5.3.2 Implementasi Algoritma KNN.....	66
5.3.3 Implementasi Algoritma Fuzzy KNN.....	69
5.4 Implementasi Antarmuka	70
5.4.1 Implementasi Antarmuka Halaman Beranda	70
5.4.2 Implementasi Antarmuka Halaman Diagnosa	70
5.4.3 Implementasi Antarmuka Halaman Data	70
BAB 6 PENGUJIAN dan analisis	73
6.1 Pengujian Fungsionalitas	74
6.1.1 Prosedur dan Hasil Pengujian Fungsionalitas.....	74
6.1.2 Analisis Pengujian Fungsionalitas.....	74
6.2 Pengujian Akurasi.....	74
6.2.1 Prosedur dan Hasil Pengujian Nilai K.....	75
6.2.2 Analisis Pengujian Nilai K.....	76
6.2.3 Prosedur dan Hasil Pengujian Jumlah Variasi Data.....	76
6.2.4 Analisis Pengujian Jumlah Variasi Data.....	76
6.2.5 Prosedur dan Hasil Pengujian Perbandingan Akurasi Fuzzy K- NN dan K-NN.....	77
6.2.6 Analisis Pengujian Perbandingan Akurasi Fuzzy K-NN dan K-NN	79
BAB 7 Penutup.....	83
7.1 Kesimpulan.....	83
7.2 Saran.....	83
DAFTAR PUSTAKA	84

DAFTAR TABEL

Tabel 2.1 Kajian pustaka	7
Tabel 2.2 Kandungan gizi dalam 100 g kacang kedelai	12
Tabel 2.3 Contoh data latih sistem	25
Tabel 2.4 Contoh data uji sistem	25
Tabel 2.5 Contoh data latih yang telah dinormalisasi	26
Tabel 2.6 Contoh data uji yang telah dinormalisasi	26
Tabel 2.7 Tabel hasil perhitungan jarak	27
Tabel 2.8 Hasil perhitungan jarak dari terkecil	27
Tabel 2.9 Hasil perhitungan jarak terkecil berdasarkan <i>id</i> data	27
Tabel 2.10 Hasil perhitungan jarak dari terkecil berdasarkan penyakit	28
Tabel 2.11 <i>K record</i> terdekat	28
Tabel 2.12 Hasil inialisasi <i>fuzzy</i>	29
Tabel 2.13 Hasil perhitungan jarak data uji terhadap data latih	29
Tabel 2.14 Hasil perhitungan jarak data uji terhadap latih dari yang terkecil	30
Tabel 4.1 Kebutuhan fungsional sistem	36
Tabel 4.2 Kebutuhan non fungsional	36
Tabel 4.3 Daftar nilai bobot setiap gejala	44
Tabel 4.4 Daftar kode gejala, gejala dan penyakit	46
Tabel 4.5 Data latih penyakit tanaman kedelai	47
Tabel 4.6 Data uji penyakit tanaman kedelai	47
Tabel 4.7 Normalisasi data latih penyakit tanaman kedelai	48
Tabel 4.8 Normalisasi data uji penyakit tanaman kedelai	48
Tabel 4.9 Tabel hasil perhitungan jarak	49
Tabel 4.10 Hasil perhitungan jarak dari terkecil	49
Tabel 4.11 Hasil perhitungan jarak dari yang terkecil berdasarkan id data	50
Tabel 4.12 Hasil perhitungan jarak dari terkecil berdasakan penyakit	50
Tabel 4.13 <i>K record</i> terdekat	51
Tabel 4.14 Hasil inialisasi <i>fuzzy</i>	51
Tabel 4.15 Hasil perhitungan jarak data uji terhadap data latih	52
Tabel 4.16 Hasil jarak data uji terhadap data latih dari yang terkecil	52

Tabel 4.17 Skenario pengujian fungsionalitas	56
Tabel 4.18 Contoh pengujian akurasi variasi nilai k	56
Tabel 4.19 Contoh pengujian akurasi variasi jumlah data	56
Tabel 5.1 Spesifikasi perangkat keras	58
Tabel 5.2 Spesifikasi perangkat lunak	58
Tabel 6.1 Hasil pengujian fungsionalitas.....	74
Tabel 6.2 Hasil pengujian nilai k	75
Tabel 6.3 Hasil pengujian variasi data	76
Tabel 6.4 Hasil perbandingan <i>fuzzy k-nn</i> dan <i>k-nn</i> terhadap pengaruh nilai k	77
Tabel 6.5 Hasil perbandingan <i>fuzzy k-nn</i> dan <i>k-nn</i> terhadap variasi jumlah data	78
Tabel 6.6 Data perbandingan antara diagnosa pakar, <i>f k-nn</i> dan <i>k-nn</i> terhadap nilai k	81
Tabel 6.7 Data perbandingan antara diagnosa pakar, <i>f k-nn</i> dan <i>k-nn</i> terhadap variasi jumlah data	82



DAFTAR GAMBAR

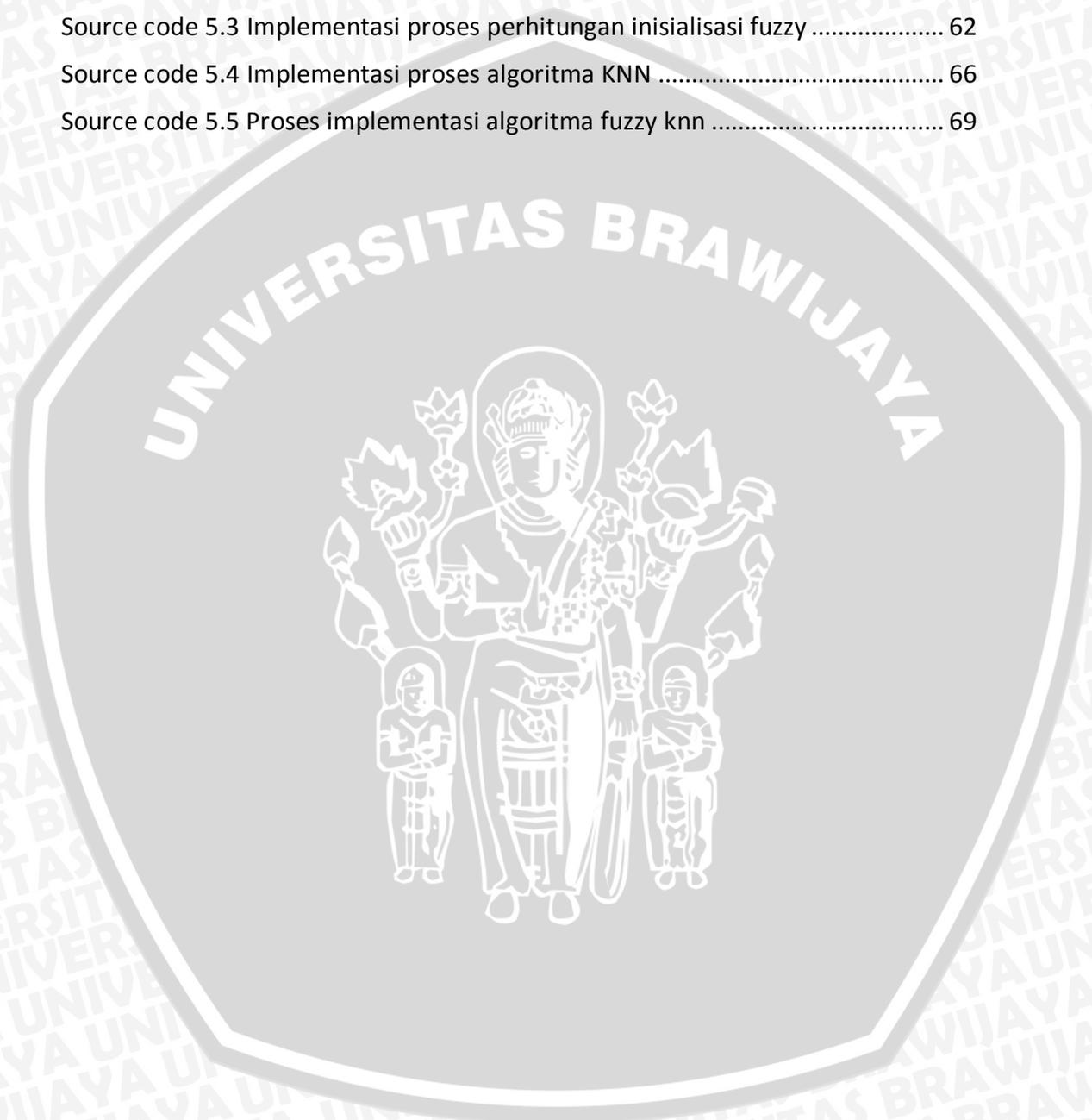
Gambar 2.1 Karat daun	13
Gambar 2.2 Pustul bakteri	14
Gambar 2.3 Virus mosaik	14
Gambar 2.4 Target spot	15
Gambar 2.5 Hawar batang.....	16
Gambar 2.6 Komponen-komponen sistem pakar	19
Gambar 3.1 Diagram alir penelitian	31
Gambar 3.2 Struktur sistem pakar diagnosa penyakit tanaman kedelai	33
Gambar 4.1 Pohon perancangan sistem.....	35
Gambar 4.2 Proses klasifikasi secara umum	37
Gambar 4.3 Proses normalisasi	38
Gambar 4.4 Proses inialisasi <i>fuzzy</i>	39
Gambar 4.5 Proses <i>euclidian distance</i> antar data latih	40
Gambar 4.6 Proses perhitungan nilai keanggotaan <i>fuzzy</i>	41
Gambar 4.7 Proses <i>K-Nearest Neighbor</i>	42
Gambar 4.8 Proses <i>Euclidian distance</i> data uji pada data latih	42
Gambar 4.9 Proses <i>Fuzzy K-Nearest Neighbor</i>	43
Gambar 4.10 Antarmuka tampilan beranda.....	54
Gambar 4.11 Antarmuka tampilan diagnosa	55
Gambar 4.12 Antarmuka tampilan data	55
Gambar 5.1 Tahapan Implementasi	57
Gambar 5.2 Implementasi antarmuka halaman beranda	71
Gambar 5.3 Implementasi antarmuka halaman diagnosa	71
Gambar 5.4 Implementasi antarmuka halaman hasil diagnosa.....	72
Gambar 5.5 Implementasi antarmuka halaman data	72
Gambar 6.1 Pohon pengujian dan analisis	73
Gambar 6.2 Grafik hasil pengujian pengaruh nilai k.....	75
Gambar 6.3 Grafik hasil pengujian variasi jumlah data	77
Gambar 6.4 Grafik hasil perbandingan F K-NN dan K-NN terhadap nilai k	78
Gambar 6.5 Grafik hasil perbandingan F K-NN dan K-NN variasi jumlah data	79

Gambar 6.6 Hasil diagnosa data ke-7 dengan nilai $k = 7$ 80



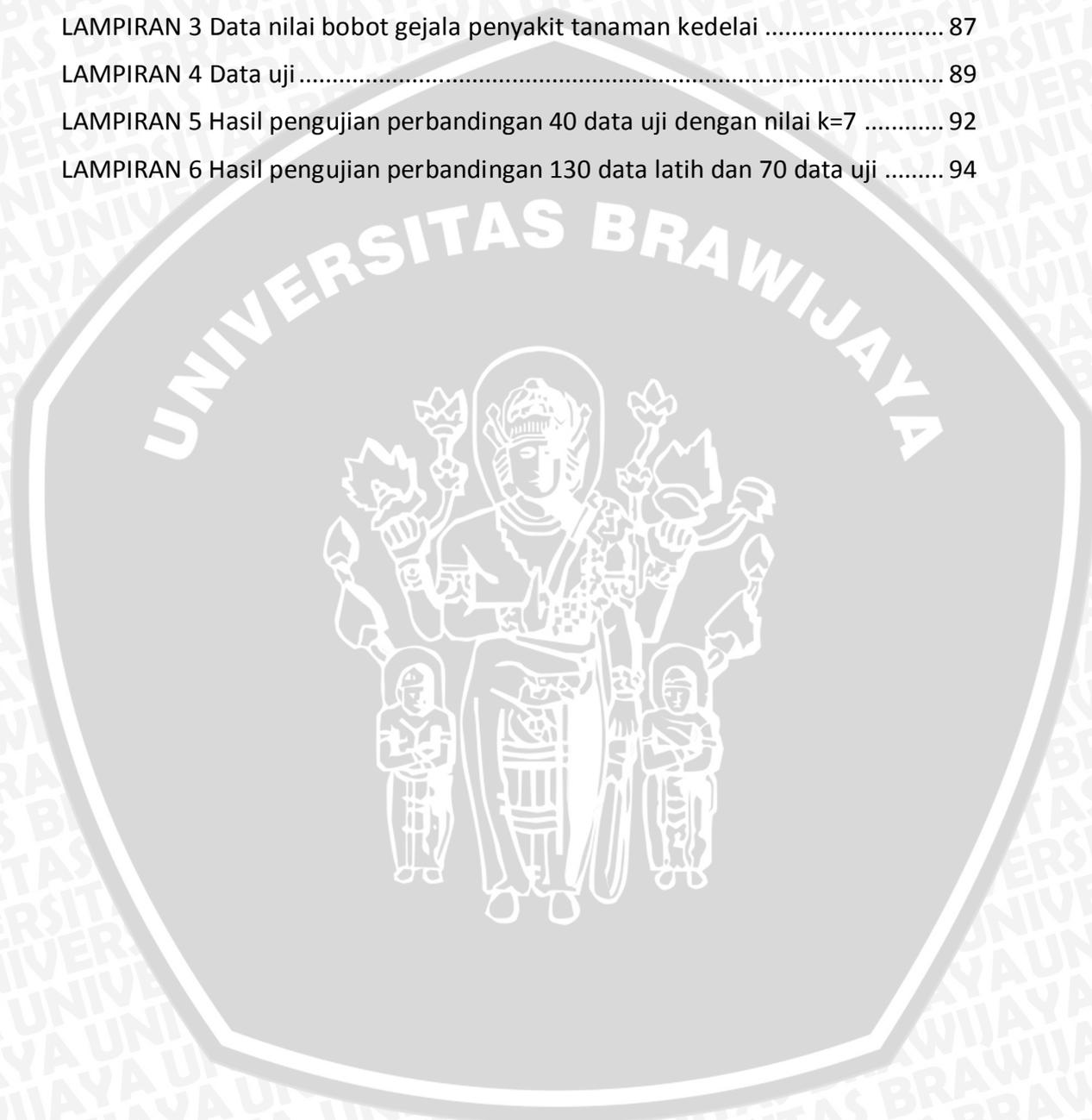
DAFTAR SOURCE CODE

Source code 5.1 Implementasi proses perhitungan normalisasi data latih	59
Source code 5.2 Implementasi proses perhitungan normalisasi data uji	61
Source code 5.3 Implementasi proses perhitungan inialisasi fuzzy	62
Source code 5.4 Implementasi proses algoritma KNN	66
Source code 5.5 Proses implementasi algoritma fuzzy knn	69



DAFTAR LAMPIRAN

LAMPIRAN 1 Hasil wawancara	85
LAMPIRAN 2 Data aturan penyakit tanaman kedelai	86
LAMPIRAN 3 Data nilai bobot gejala penyakit tanaman kedelai	87
LAMPIRAN 4 Data uji	89
LAMPIRAN 5 Hasil pengujian perbandingan 40 data uji dengan nilai $k=7$	92
LAMPIRAN 6 Hasil pengujian perbandingan 130 data latih dan 70 data uji	94



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Indonesia merupakan negara dengan tingkat pertumbuhan penduduk yang relatif besar. Menurut data yang diambil dari BPS (2015) menyatakan bahwa pada tahun 2010, jumlah penduduk Indonesia tercatat 237,6 juta jiwa. Jumlah ini bertambah sekitar 32,5 juta jiwa dari jumlah penduduk sebelumnya yang tercatat pada tahun 2000. Dengan angka pertumbuhan penduduk yang tinggi tersebut tentu membawa dampak pada perubahan kebutuhan dan produksi pangan nasional. Kebutuhan pangan akan bertambah seiring dengan penambahan jumlah penduduk. Ada 5 komoditas pangan strategis untuk pembangunan ketahanan pangan yaitu padi, jagung, kedelai, tebu dan daging sapi.

Kedelai merupakan salah satu sumber bahan komoditas pangan yang telah lama dibudidayakan di Indonesia, yang saat ini tidak hanya diposisikan sebagai bahan baku industri pangan, namun juga ditempatkan sebagai bahan baku industri non-pangan (Arnanda, et al., 2015). Kedelai juga merupakan komoditas pangan utama yang memiliki kandungan gizi tinggi, terutama kadar proteinnya yang mencapai 34 persen. Selain digunakan sebagai sumber protein nabati, kedelai juga digunakan masyarakat sebagai bahan baku utama sumber energi alternatif. Kedelai juga berperan sebagai pangan fungsional yang dapat mencegah timbulnya penyakit seperti jantung dan hipertensi. Zat isoflavon yang terdapat pada kedelai memiliki fungsi sebagai antioksidan. Beberapa produk yang dapat dihasilkan dari kedelai adalah tempe, tahu, kecap, susu dan lain lain.

Meningkatkan produksi kedelai tidaklah mudah. Banyak faktor yang dapat menghambat produksi tanaman kedelai salah satunya adalah penyakit pada tanaman kedelai. Jenis penyakit pada tanaman kedelai yang beraneka ragam membutuhkan penanganan yang berbeda-beda. Minimnya pengetahuan petani tentang gejala dan jenis penyakit yang menyerang tanaman kedelai menyebabkan petani kesulitan dalam menyelesaikan masalah penyakit pada tanaman kedelai sehingga dapat menjadi penghambat dalam produksi tanaman kedelai bahkan bisa jadi mengalami gagal panen. Keterbatasan pakar kedelai dalam mengatasi masalah identifikasi penyakit pada tanaman kedelai membuat petani menangani permasalahan tersebut dengan pengetahuan seadanya. Untuk membantu keterbatasan tersebut maka dibutuhkan sistem pakar yang mampu mengidentifikasi penyakit pada tanaman kedelai.

Sistem pakar (*expert system*) adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli (Fadli, 2010). Dalam menyelesaikan permasalahan, sistem pakar memerlukan suatu metode untuk mengolah data masukan menjadi informasi yang berguna. Pada proses klasifikasi terdapat beberapa metode antara lain *Decision tree*, *Bayesian*, *Fuzzy*, *Neural Network*, *Support Vector Machine (SVM)* dan *K-Nearest Neighbor*.

Metode *K-Nearest Neighbour* (K-NN) merupakan metode klasifikasi data yang cara kerjanya relatif lebih sederhana bila dibandingkan dengan metode klasifikasi lainnya. Namun pada metode K-NN kurang sesuai untuk pengklasifikasian yang cenderung mengatur jumlah data latih yang memiliki nilai mayoritas. Karena prinsip mayoritas K-NN mengklasifikasikan data baru berdasarkan data mayoritas kelas yang ada pada sejumlah K tetangga terdekat dari data baru tersebut. Oleh James M. Keller ditemukan suatu metode baru untuk mengatasi permasalahan ini, yaitu digunakannya metode Fuzzy K-NN yang merupakan gabungan dari metode *fuzzy* dengan K-NN (Wisdianto, Ridok dan Rahman, 2013). Namun pada Metode *Fuzzy K-Nearest Neighbor* memiliki kelemahan yaitu antara sejumlah k tetangga dianggap sama pentingnya padahal belum tentu, hal tersebut diatasi dengan perhitungan derajat keanggotaan data pada tiap kelas pada persamaan 2-4. Selain itu pada data latih belum dapat diketahui kekuatan keanggotaannya pada sebuah kelas dan untuk mengatasi hal tersebut menggunakan inisialisasi *fuzzy* (Keller, Gray dan Givens, 1985).

Penelitian sebelumnya tentang sistem pakar untuk menentukan penyakit pada tanaman kedelai telah dilakukan oleh Sofa Zainudin, Nurul Hidayat dan Arief Andy Soebroto dengan menggunakan metode *Modified K-Nearest Neighbor*. Berdasarkan penelitian tersebut Algoritma *Modified K-Nearest Neighbour* (MKNN) dapat diimplementasikan untuk klasifikasi data morfologi guna menentukan penyakit yang menyerang pada tanaman kedelai dengan tingkat akurasi rata-rata maksimum yang didapat adalah 92,74% dengan nilai k=3 (Zainudin, Hidayat & Soebroto, 2013). Penelitian sebelumnya mengenai metode *Fuzzy K-NN* telah dilakukan oleh Syela Ukmala tentang identifikasi penyakit pada tanaman tomat. Pengujian akurasi yang telah dilakukan terhadap 30 data uji, didapatkan nilai rata-rata akurasi sistem sebesar 85,4% akurasi sistem dan nilai akurasi maksimum sebesar 100% dengan nilai k=5 (Ukmala, 2015).

Berdasarkan pada uraian latar belakang tersebut, peneliti mengusulkan judul skripsi "*Pemodelan Sistem Pakar Untuk Identifikasi Penyakit Pada Tanaman Kedelai Menggunakan Metode Fuzzy K-Nearest Neighbor*". Sistem ini diharapkan mampu memberikan informasi mengenai jenis penyakit pada tanaman kedelai berdasarkan pada gejala-gejala yang diinputkan. Sehingga dengan menggunakan metode *Fuzzy K-NN* ini diharapkan dapat memberikan penanganan penyakit tanaman kedelai secara tepat.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, maka dapat diambil rumusan masalah sebagai berikut:

1. Bagaimana memodelkan sistem pakar untuk identifikasi penyakit pada tanaman kedelai menggunakan metode *fuzzy k-nearest neighbor*.
2. Bagaimana hasil pengujian sistem pakar untuk identifikasi penyakit pada tanaman kedelai menggunakan metode *fuzzy k-nearest neighbor*.

1.3 Tujuan Penelitian

Adapun tujuan diadakan penelitian ini adalah:

1. Memodelkan sistem pakar untuk identifikasi penyakit pada tanaman kedelai menggunakan metode *fuzzy k-nearest neighbor*.
2. Menguji sistem pakar untuk identifikasi penyakit pada tanaman kedelai menggunakan metode *fuzzy k-nearest neighbor*.

1.4 Manfaat Penelitian

Manfaat yang bisa diperoleh dari penelitian ini yaitu sistem diharapkan mampu mengidentifikasi penyakit pada tanaman kedelai, sehingga dapat membantu petani kedelai dalam mengambil tindakan yang tepat dalam penanganan penyakit kedelai.

1.5 Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Proses identifikasi berdasarkan 16 gejala yang nampak pada daun, biji dan batang.
2. Output sistem adalah hasil identifikasi 5 jenis penyakit pada tanaman kedelai, yaitu Karat Daun, Pustul Bakteri, Virus Mosaik, Target Spot dan Hawar Batang.
3. Data latih berjumlah 160 data diambil pada tahun 2015 berdasarkan penelitian di Balai Pengkajian Teknologi Pertanian Malang dan dikonsultasikan dengan Prof. (Riset). Dr. Ir, Moh. Cholil Mahfud, M.S sebagai pakar penyakit tanaman kedelai.
4. Sistem dibangun secara *offline*.
5. Pengujian yang dilakukan adalah pengujian fungsionalitas dan pengujian akurasi yang meliputi pengujian pengaruh nilai k klasifikasi, pengaruh variasi jumlah data dan perbandingan antara metode *Fuzzy K-NN* dan *K-NN*.

1.6 Sistematika Pembahasan

Sistematika pembahasan penelitian adalah sebagai berikut:

BAB I PENDAHULUAN

Bab ini berisi tentang latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, dan sistematika pembahasan.

BAB II LANDASAN KEPUSTAKAAN

Bab ini berisi uraian dan penjelasan tentang kajian pustaka dari penelitian sebelumnya tentang implementasi metode *Fuzzy K-NN*, dasar teori mengenai tanaman kedelai, sistem pakar, logika *fuzzy*, klasifikasi, inialisasi *fuzzy*, metode *K-NN*, metode *Fuzzy K-NN*, dan akurasi.

BAB III METODOLOGI PENELITIAN

Bab ini membahas tentang tahapan metode penelitian yang dilakukan yaitu studi literatur, pengumpulan data, analisis kebutuhan, perancangan sistem, implementasi sistem pakar, pengujian sistem pakar dan pengambilan kesimpulan.

BAB IV PERANCANGAN SISTEM

Bab ini membahas tentang tahapan perancangan sistem untuk identifikasi penyakit tanaman kedelai yang meliputi analisa kebutuhan perangkat lunak, perancangan perangkat lunak dan perancangan sistem pakar.

BAB V IMPLEMENTASI SISTEM

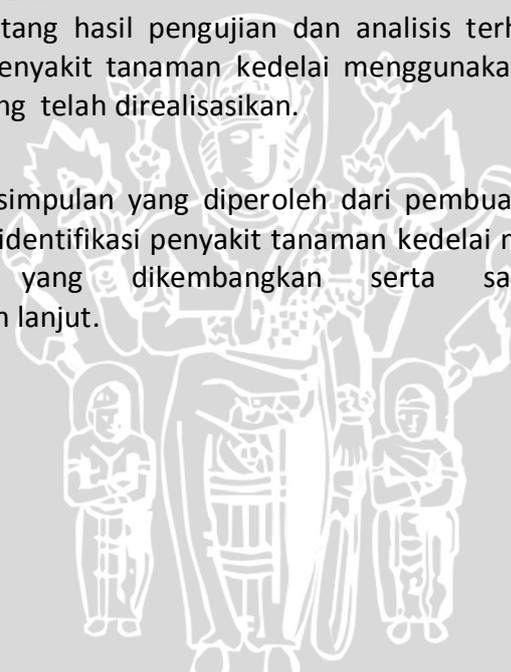
Bab ini membahas tentang implementasi sistem pakar diagnosa penyakit tanaman kedelai menggunakan metode *fuzzy k-nearest neighbor* berdasarkan pada proses perancangan pada tahapan sebelumnya. Tahapan implementasi meliputi spesifikasi sistem, batasan implementasi, implementasi algoritma dan implementasi antarmuka.

BAB VI PENGUJIAN SISTEM

Bab ini memuat tentang hasil pengujian dan analisis terhadap sistem pakar untuk identifikasi penyakit tanaman kedelai menggunakan metode *fuzzy k-nearest neighbor* yang telah direalisasikan.

BAB VII PENUTUP

Bab ini memuat kesimpulan yang diperoleh dari pembuatan dan pengujian sistem pakar untuk identifikasi penyakit tanaman kedelai menggunakan *fuzzy k-nearest neighbor* yang dikembangkan serta saran-saran untuk pengembangan lebih lanjut.



BAB 2 LANDASAN KEPUSTAKAAN

Pada bab ini berisikan kajian pustaka dan dasar teori. Kajian pustaka pada penelitian ini menjelaskan gambaran umum tentang perbandingan penelitian terdahulu yang menggunakan metode yang sama dengan penelitian ini. Sedangkan dasar teori berisikan teori yang menunjang penelitian ini dari berbagai sumber sebagai landasan penelitian yang dilakukan. Dasar teori yang diperlukan berdasarkan latar belakang dan rumusan masalah adalah tanaman kedelai, sistem pakar, logika *fuzzy*, klasifikasi, inisialisasi *fuzzy*, metode K-NN, metode *Fuzzy K-NN*, dan akurasi.

2.1 Kajian Pustaka

Kajian pustaka pada penelitian ini adalah membandingkan penelitian ini dengan penelitian sebelumnya pada objek berbeda dengan metode yang sama. Dimana metode dan objek tersebut akan menjadi acuan peneliti dalam melakukan penelitian.

Penelitian pertama dilakukan oleh Yanita Selly Meristika dengan judul “Perbandingan *K-Nearest Neighbor* dan *Fuzzy K-Nearest Neighbor* pada Diagnosis Penyakit Diabetes Melitus”. Penelitian ini membandingkan antara *K-Nearest Neighbor* (K-NN) dan *Fuzzy K-Nearest Neighbor* (FK-NN) menggunakan data diabetes Indian Pima. Data inputan dalam penelitian ini berupa jumlah hamil, 2 jam PP (OGTT), tekanan diastolik, ketebalan kulit trisep (TSFT), 2 jam serum insulin (INS), indeks masa badan, (IMB), riwayat diabetes keluarga (DPF) dan usia. Proses pada penelitian ini yaitu melakukan normalisasi nilai setiap atribut, proses klasifikasi menggunakan K-NN, proses klasifikasi dengan FK-NN dengan memilih nilai derajat keanggotaan terbesar pada target output untuk ditentukan kelas target yang baru. Penelitian ini menghasilkan output dari diagnosa resiko Diabetes Melitus pada sistem yaitu positif DM (1) dan Negatif DM (0) (Meristika, 2013).

Penelitian kedua dilakukan oleh Rofika dengan judul “Sistem pakar Diagnosa Penyakit Kulit Pada Anak Menggunakan Metode *Fuzzy K-Nearest Neighbor*”. Data input yang digunakan pada penelitian ini berupa gejala penyakit kulit pada anak sesuai dengan data pasien. Terdapat 14 macam gejala penyakit dalam penelitian ini, diantaranya adalah gatal-gatal, kulit meradang, melepuh, kulit bersisik, panas pada area yang terinfeksi, muncul gelembung nanah, demam, nyeri saat ditekan, muncul gelembung berisi air, batuk/pilek, nyeri kepala, perih, bengkak, dan mata merah. Penelitian ini menghasilkan output berupa diagnosa jenis penyakit kulit pada anak yaitu Cacar air (Varisela), Skabies, Campak (Morbili), Dermatitis (Eksim), Herpes, Furunkel (Abses) dan solusi pengobatannya (Rofika, 2015).

Penelitian ketiga dilakukan oleh Syela Ukmala dengan judul “Pemodelan Sistem Pakar Untuk Identifikasi Penyakit Pada Tanaman Tomat Menggunakan Metode *Fuzzy K-Nearest Neighbor*”. Data input yang digunakan berdasarkan pada 15 gejala yang nampak pada daun, batang dan buah. Proses klasifikasi dalam penelitian ini terdiri dari tiga proses utama yaitu Normalisasi nilai parameter gejala pada data

latih dan data uji, Proses K-NN menggunakan jarak *euclidean distance* antara data uji dengan data latih dan selanjutnya mayoritas kelas pada sejumlah k (tetangga terdekat) digunakan untuk menentukan data uji yang masuk dalam salah satu kelas prediksi dan Proses FK-NN dengan pemberian nilai keanggotaan pada data uji setelah melalui proses K-NN, sehingga setiap data uji mempunyai nilai relatif terhadap setiap kelas dimana kelas dengan nilai keanggotaan terbesar yang menjadi kelas hasil *fuzzy k-nearest neighbour*. Data output yang dihasilkan dalam penelitian ini berupa penyakit pada tanaman tomat yaitu bercak coklat, busuk daun, layu fusarium, layu bakteri, mozaik dan keriting (Ukmala, 2015).

Sistem yang akan dibangun penulis dalam penelitian ini berupa sistem pakar yang memanfaatkan metode *Fuzzy K-Nearest Neighbor* dalam membantu mendiagnosa penyakit pada tanaman kedelai berdasarkan 16 gejala penyakit yang menyerang tanaman kedelai. Output yang dihasilkan dalam penelitian ini adalah penyakit tanaman kedelai yaitu karat daun, pustul bakteri, virus mozaik, target spot dan hawar batang serta cara menanggulangnya. Penjelasan mengenai perbandingan objek, metode serta output ditunjukkan pada Tabel 2.1.

2.2 Tanaman Kedelai

Dalam sub bab ini akan dijelaskan mengenai teori-teori yang berkaitan dengan tanaman kedelai meliputi sejarah tanaman kedelai, taksonomi tanaman kedelai, morfologi tanaman kedelai, kegunaan dan manfaat tanaman kedelai dan penyakit tanaman kedelai.

2.2.1 Sejarah Tanaman Kedelai

Tanaman kacang kedelai sering disebut dengan *Sapi dari Negeri Cina* atau *Cow from China*. Menurut sejarah pada masa lalu, kacang kedelai diproduksi di China sebagai pengganti susu sapi. Oleh karena itu, sebagian kalangan menyebut bahwa kedelai berasal dari daratan China. Budidaya kacang kedelai pertama kali dilakukan sekitar abad 11 SM di China Utara. Kemudian penyebarannya meluas antara lain ke Korea dan Jepang.

Jenis kacang kedelai liar (*Glycine ussuriensis*) yang pada masa lampau terdapat di Cina, Manchuria dan Korea dianggap kacang kedelai tertua yang menurunkan kacang kedelai yang sekarang dibudidayakan. Hal tersebut menunjukkan bahwa kacang kedelai berasal dari Asia Timur dan dalam perkembangan selanjutnya menyebar ke berbagai negara di dunia.

Di Indonesia, kacang kedelai mulai dikenal pada abad ke-16 di pulau Jawa. Kemudian penyebaran kacang kedelai semakin meluas setelah Jung Hun melakukan penanaman komoditas ini di kawasan kapur selatan Jawa Tengah pada tahun 1853. Setelah itu, tanaman kedelai berkembang di Bandung pada tahun 1855 dan di pulau Jawa dan Bali sekitar tahun 1935. Penanaman kacang kedelai semakin meluas di kawasan nusantara (Rukmana dan Yudirachman, 2014).

Tabel 2.1 Kajian pustaka

No	Judul	Objek dan Input Kriteria	Metode dan Proses	Output dan Hasil
1.	Perbandingan <i>K-Nearest Neighbor</i> dan <i>Fuzzy K-Nearest Neighbor</i> pada Diagnosis Penyakit <i>Diabetes Melitus</i>	<p>Objek: Penyakit Diabetes Melitus</p> <p>Input kriteria: jumlah hamil, 2 jam PP (OGTT), tekanan diastolik, ketebalan kulit trisep (TSFT), 2 jam serum insulin (INS), indeks massa badan (IMB), riwayat diabetes keluarga (DPF), dan usia.</p>	<p>Metode :</p> <ul style="list-style-type: none"> • <i>K-Nearest Neighbor</i> • <i>Fuzzy K-Nearest Neighbor</i> <p>Proses KNN : normalisasi data latih dan data uji, menghitung jarak <i>record</i> data uji terhadap data latih menggunakan rumus jarak euclidian, mengurutkan data berdasarkan jarak terkecil, menentukan <i>k record</i> terdekat, dipilih kelas yang paling dominan.</p> <p>Proses FKNN : normalisasi data latih dan data uji, menghitung jarak <i>record</i> data uji terhadap data latih, mengurutkan data berdasarkan jarak terkecil, menentukan <i>k record</i> terdekat, mencari nilai keanggotaan untuk tiap kelas <i>j</i>, kemudian menghitung nilai keanggotaan data uji pada masing-masing kelas. Dipilih nilai keanggotaan terbesar</p>	<p>Output : positif Diabetes Melitus (1) dan negatif Diabetes Melitus (0).</p> <p>Hasil : F K-NN = 98% pada k=15 K-NN = 96% pada k=11</p>
2.	Sistem Pakar Diagnosa Penyakit Kulit Pada Anak	<p>Objek : penyakit kulit yang menyerang anak.</p>	<p>Metode : <i>Fuzzy K-Nearest Neighbor</i></p>	<p>Output : diagnosa jenis penyakit kulit yang menyerang anak (Cacar</p>

	Menggunakan Metode <i>Fuzzy K-Nearest Neighbor</i>	<p>Input kriteria:</p> <ul style="list-style-type: none"> • Gatal-gatal • Kulit meradang • Melepuh • Kulit bersisik • Panas pada area yang terinfeksi • Muncul gelembung nanah • Demam • Nyeri saat ditekan • Muncul gelembung berisi air • Batuk pilek • Mata merah • Nyeri kepala • Perih pada area yang terinfeksi • Bengkak 	<p>Proses:</p> <ul style="list-style-type: none"> • <i>User</i> memasukkan data gejala penyakit kulit yang menyerang anak • Sistem akan menghitung jarak menggunakan rumus <i>Euclidean Distance</i> • Sistem akan mengurutkan nilai berdasarkan k tetangga terdekat yaitu dari nilai terkecil ke nilai terbesar • Sistem akan menghitung nilai keanggotaan data terhadap suatu kelas menggunakan rumus <i>Fuzzy K-NN</i> • Sistem akan memberikan hasil diagnosa penyakit kulit pada anak berdasarkan nilai keanggotaan terbesar. 	<p>air (Varisela), Skabies, Campak (Morbili), Dermatitis (Eksim), Herpes dan Furunkel (Abses))</p> <p>Hasil: Nilai akurasi maksimum 96.67%</p>
3.	Pemodelan Sistem Pakar Untuk Identifikasi Penyakit Pada Tanaman Tomat Menggunakan Metode <i>Fuzzy K-Nearest Neighbour</i>	<p>Obyek: Tanaman Tomat</p> <p>Input kriteria : 15 gejala penyakit yang terdapat pada daun, batang dan buah.</p>	<p>Metode: <i>Fuzzy K-Nearest Neighbor</i></p> <p>Proses: normalisasi data latih dan data uji, menghitung jarak record data uji terhadap data latih, mengurutkan data berdasarkan</p>	<p>Output: diagnosa penyakit busuk daun, bercak coklat, layu fusarium, layu bakteri, mozaik dan keriting.</p> <p>Hasil: rata-rata nilai akurasi sebesar 85.4% dan nilai akurasi</p>

			jarak terkecil, menentukan k record terdekat, mencari nilai keanggotaan untuk tiap kelas j, kemudian menghitung nilai keanggotaan data uji pada masing-masing kelas. Dipilih nilai keanggotaan terbesar	terbaik sebesar 100% pada k=5
4.	Sistem Pakar Untuk Identifikasi Diagnosa Penyakit Pada Tanaman Kedelai Dengan Metode Fuzzy K-Nearest Neighbor	<p>Obyek : tanaman kedelai.</p> <p>Input kriteria : 16 gejala penyakit yang terdapat pada daun, batang dan biji</p>	<p>Metode : Fuzzy K-Nearest Neighbor</p> <p>Proses : Normalisasi data latih dan data uji, Inialisasi fuzzy dengan menghitung jarak Euclidian antar data lati kemudian diurutkan berdasarkan penyakit dari yang terkecil setelah itu menentukan k record inialisasi terdekat dan menghitung nilai inialisasi fuzzy, Kemudian masuk ke proses KNN dengan menghitung jarak Euclidian data uji terhadap data latih, mengurutkan data berdasarkan nilai Euclidian terkecil, menentukan k record klasifikasi terdekat, menghitung derajat keanggotaan data baru terhadap masing-masing kelas, Memilih kelas yang memiliki nilai keanggotaan terbesar sebagai hasil.</p>	<p>Output: diagnosa jenis penyakit pada tanaman kedelai (karat daun, pustul bakteri, virus moaik, target spot dan hawar batang)</p>

Sumber: (Meristika,2013), (Rofika,2015), (Ukmala, 2015)

2.2.2 Taksonomi Tanaman Kedelai

Pada awalnya, kacang kedelai dikenal dengan beberapa nama botani, yaitu *Glycine soja* dan *Soja max*. Namun pada tahun 1948 telah disepakati bahwa nama botani yang dapat diterima dalam istilah ilmiah adalah *Glycine max* (L.) Merill. Menurut Rukmana dan Yudirachman (2014), klasifikasi tanaman kedelai adalah sebagai berikut:

Kingdom	: Plantae (Tumbuhan)
Subkingdom	: Tracheobionta (Tumbuhan berpembuluh)
SuperDivisi	: Spermatophyta (Menghasilkan biji)
Divisi	: Magnoliophyta (Tumbuhan berbunga)
Kelas	: Magnoliopsida (Berkeping dua, dikotil)
Ordo	: Fabales
Famili	: Fabaceae (Suku polong-polongan)
Genus	: <i>Glycine</i>
Spesies	: <i>Glycine max</i> (L.) Merill atau <i>Glycine soya</i> Benth.

2.2.3 Morfologi Tanaman Kedelai

Morfologi tanaman kedelai didukung oleh komponen utamanya, yaitu akar, daun, batang, polong, dan biji sehingga pertumbuhannya bisa optimal.

1. Akar

Akar kedelai muncul dari belahan kulit biji yang muncul di sekitar misofil yang kemudian tumbuh dengan cepat ke dalam tanah, kotiledon yang terdiri dari dua keping akan terangkat ke permukaan tanah akibat dari pertumbuhan yang cepat dari hipokotil. Sistem perakaran kedelai terdiri dari akar tunggang dan akar sekunder yang tumbuh dari akar tunggang. Selain itu kedelai juga sering membentuk akar adventif yang tumbuh dari bagian bawah hipokotil.

Pertumbuhan dari akar tunggang dapat mencapai panjang sekitar 2 m atau lebih pada kondisi yang optimal, namun umumnya hanya tumbuh pada kedalaman lapisan tanah olah yang tidak terlalu dalam, sekitar 30-50 cm. Sedangkan akar serabut dapat tumbuh pada kedalaman tanah sekitar 20-30 cm. Akar serabut ini mula-mula tumbuh di dekat ujung akar tunggang, sekitar 3-4 hari setelah berkecambah dan akan semakin bertambah banyak dengan pembentukan akar-akar muda yang lain.

2. Daun

Tanaman kedelai mempunyai dua bentuk daun yaitu *stidia kotiledon* yang tumbuh saat tanaman masih berbentuk kecambah dengan dua helai daun tunggal dan daun bertangkai tiga (*trifoliolate leaves*) yang tumbuh selepas masa pertumbuhan. Bentuk daun kedelai ada dua, yaitu bulat (oval) dan lancip (lanceolate). Bentuk daun diperkirakan mempunyai korelasi dengan potensi produksi biji. Daun mempunyai stomata yang berjumlah antara 190-320 buah/m². Daun mempunyai bulu dengan warna cerah dan jumlahnya bervariasi. Panjang bulu pada daun kedelai bisa mencapai 1 mm dan lebar 0,0025 mm. Kepadatan

bulu bervariasi, tergantung varietas, tetapi biasanya antara 3- 20 buah/mm² . Jumlah bulu pada varietas berbulu lebat, dapat mencapai 3- 4 kali lipat dari varietas yang berbulu normal. Lebat-tipisnya bulu pada daun kedelai berkaitan dengan tingkat toleransi varietas kedelai terhadap serangan jenis hama tertentu.

3. Batang

Hipokotil pada proses perkecambahan kedelai merupakan bagian batang, mulai dari pangkal akar sampai kotiledon. Hipokotil dan dua keping kotiledon yang masih melekat pada hipokotil akan menerobos ke permukaan tanah. Bagian batang kecambah yang berada di atas kotiledon tersebut disebut epikotil. Pertumbuhan batang tanaman kedelai dibedakan menjadi dua yaitu tipe determinate dan indeterminate. Perbedaan sistem pertumbuhan batang ini berdasarkan pada keberadaan bunga pada pucuk batang. Pertumbuhan batang tipe determinate ditunjukkan dengan batang yang tidak tumbuh lagi saat tanaman mulai berbunga. Sementara pertumbuhan batang tipe indeterminate dicirikan bila pucuk batang tanaman masih bisa tumbuh daun, walaupun tanaman sudah mulai berbunga. Selain itu, ada varietas hasil persilangan yang mempunyai tipe batang mirip keduanya sehingga dikategorikan sebagai semi-determinate atau semiindeterminate.

4. Polong

Polong kedelai pertama kali terbentuk sekitar 7-10 hari setelah munculnya bunga pertama. Panjang polong muda sekitar 1 cm. Jumlah polong yang terbentuk pada setiap ketiak tangkai daun sangat beragam, antara 1-10 buah dalam setiap kelompok. Pada setiap tanaman, jumlah polong dapat mencapai lebih dari 50, bahkan ratusan. Kecepatan pembentukan polong dan pembesaran biji akan semakin cepat setelah proses pembentukan bunga berhenti. Ukuran dan bentuk polong menjadi maksimal pada saat awal periode pemasakan biji. Hal ini kemudian diikuti oleh perubahan warna polong, dari hijau menjadi kuning kecoklatan pada saat masak.

Di dalam polong terdapat biji yang berjumlah 2-3 biji. Setiap biji kedelai mempunyai ukuran bervariasi, mulai dari kecil (sekitar 7-9 g/100 biji), sedang (10-13 g/100 biji), dan besar (>13 g/100 biji). Bentuk biji bervariasi, tergantung pada varietas tanaman, yaitu bulat, agak gepeng, dan bulat telur. Namun pada umumnya sebagian besar biji berbentuk bulat telur.

5. Biji

Biji kedelai terbagi menjadi dua bagian utama, yaitu kulit biji dan janin (embrio). Pada kulit biji terdapat bagian yang disebut puser (hilum) yang berwarna coklat, hitam, atau putih. Pada ujung hilum terdapat mikofil, berupa lubang kecil yang terbentuk pada saat proses pembentukan biji. 8 Warna kulit biji bervariasi, mulai dari kuning, hijau, coklat, hitam, atau kombinasi campuran dari warna-warna tersebut. Biji kedelai tidak mengalami masa dormansi sehingga setelah proses pembijian selesai, biji kedelai dapat langsung ditanam. Namun demikian, biji tersebut harus mempunyai kadar air berkisar 12-13%.

2.2.4 Kegunaan dan Manfaat Tanaman Kedelai

Kacang kedelai merupakan komoditas pertanian yang sangat penting karena multiguna, antara lain dapat dikonsumsi langsung dan juga digunakan sebagai bahan baku pembuatan agroindustri seperti tempe, tahu, taoco, kecap, susu kedelai dan untuk keperluan industri pakan ternak (Rukmana dan Yudirachman, 2014). Berikut adalah beberapa manfaat yang diperoleh dari tanaman kedelai:

Biji kacang kedelai dapat diolah untuk dijadikan tepung (susu kedelai, campuran kue, campuran roti, campuran kue, dsb), minyak (minyak goreng, mentega, sabun, plastic dan kosmetik) dan makanan (biji muda dapat direbus menjadi sayuran dan biji tua dapat diolah menjadi aneka makana seperti tempe, tahu, taoco, kecap).

Kacang kedelai mengandung sumber protein nabati yang efisien dan murah serta mengandung gizi dan nutrisi yang lengkap. Setiap 100 g bahan kacang kedelai mengandung protein 30-34% sehingga apabila seseorang tidak dapat makan daging karena alasan tertentu, maka kebutuhan protein dapat dipenuhi dari kacang kedelai. Kandungan gizi dalam tiap 100 g kacang kedelai disajikan dalam Tabel 2.2.

Tabel 2.2 Kandungan gizi dalam 100 g kacang kedelai

Komposisi Gizi	Kandungan Gizi	
	Kedelai Basah	Kedelai Kering
Kalori (kal.)	286,00	331,00
Protein (g)	30,20	34,90
Lemak (g)	15,60	18,10
Karbohidrat (g)	30,10	34,80
Kalsium (mg)	196,00	227,00
Fosfor (mg)	506,00	585,00
Zat Besi (mg)	6,90	8,00
Vitamin A (S.I)	95,00	110,00
Vitamin B ₁ (mg)	0,93	1,07
Air (g)	20,00	10,00
Bagian yang dapat dimakan (%)	100,00	100,00

Sumber : (Rukmana dan Yudirachman, 2014)

2.2.5 Penyakit Tanaman Kedelai

Penyakit pada tanaman kedelai dapat menghambat pertumbuhan tanaman kedelai sehingga produksi kedelai pun juga akan terhambat. Berikut adalah jenis penyakit yang sering menyerang tanaman kedelai dan menghambat hasil panen

menurut Badan Penelitian dan Pengembangan Pertanian Kementerian Pertanian (2013):

1. Karat Daun

Penyebab penyakit karat daun adalah jamur *Pakhsopora pachyrhizi*. Gejala serangan pada penyakit ini yang pertama adalah berupa bercak-bercak berisi uredia (badan duah yang memproduksi spora). Bercak tersebut kemudian berkembang ke daun-daun di atasnya dengan bertambahnya umur tanaman. Bercak terutama terdapat pada permukaan bawah daun. Warna bercak coklat kemerahan seperti warna karat. Bentuk bercak umumnya bersudut banyak berukuran sampai 1mm. Bercak juga terlihat pada bagian batang dan tangkai daun.



Gambar 2.1 Karat daun

Sumber : Balai Pengkajian Teknologi Pertanian Jawa Timur (2012)

Epidemi didorong oleh panjangnya waktu daun dalam kondisi basah dengan temperatur kurang dari 28 °C. Perkecambahan spora dan penetrasi spora membutuhkan air bebas dan terjadi pada suhu 8-28 °C. Uredia muncul 9-10 hari setelah infeksi, dan urediniospora diproduksi setelah 3 minggu. Kondisi lembab yang panjang dan periode dingin dibutuhkan untuk menginfeksi daun-daun dan sporulasi. Penyebaran urediniospora dibantu oleh hembusan angin pada waktu hujan. Patogen ini tidak ditularkan melalui benih. Pengendalian penyakit karat daun dapat dilakukan dengan cara menanam varietas tahan dan penggunaan aplikasi fungisida mankoseb, triadimefon, bitertanol, difenokonazol. Penyakit karat daun pada tanaman kedelai ditunjukkan pada Gambar 2.1.

2. Pustul Bakteri

Penyebab penyakit pustul bakteri adalah *Xanthomonas axonopodis pv glycines*. Gejala awal berupa bercak kecil berwarna hijau pucat, tampak pada kedua permukaan daun, menonjol pada bagian tengah lalu menjadi bisul warna coklat muda atau putih pada permukaan bawah daun. Gejala ini sering dikacaukan dengan penyakit karat kedelai. Tetapi bercak karat lebih kecil dan sporanya

terlihat jelas. Bercak bervariasi dari titik kecil sampai besar tak beraturan, berwarna kecoklatan. Bercak kecil bersatu membentuk daerah nekrotik yang mudah robek oleh angin sehingga daun berlubang-lubang; pada infeksi berat menyebabkan daun gugur.

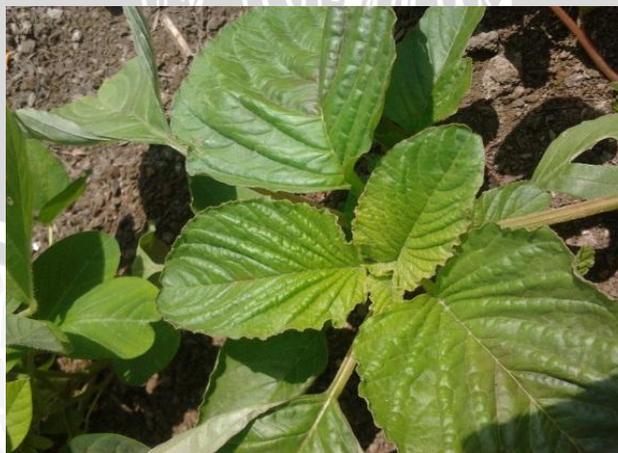
Bakteri bertahan pada biji, sisa-sisa tanaman, dan di daerah perakaran. Beberapa gulma, *Dolichos biflorus*, buncis subspecies tertentu, dan kacang tunggak bias menjadi inang. Bakteri menyebar melalui air hujan/hembusan angin pada waktu hujan. Bakteri masuk ke tanaman melalui lubang-lubang alami dan luka pada tanaman. Pengendalian dilakukan dengan cara menanam benih bebas patogen, membenamkan sisa tanaman terinfeksi dan hindari rotasi dengan buncis dan kacang tunggak. Penyakit pustule bakteri pada tanaman kedelai ditunjukkan pada Gambar 2.2.



Gambar 2.2 Pustul bakteri

Sumber : Balai Pengkajian Teknologi Pertanian Jawa Timur (2014)

3. Virus Mosaik



Gambar 2.3 Virus mosaik

Sumber : Balai Pengkajian Teknologi Pertanian Jawa Timur (2014)

Gejala serangan pada penyakit ini adalah tulang daun pada daun yang masih muda menjadi kurang jernih. Selanjutnya daun berkerut dan mempunyai gambaran mosaik dengan warna hijau gelap di sepanjang tulang daun. Tepi daun sering mengalami klorosis. Tanaman yang terinfeksi SMV ukuran biji mengecil dan jumlah biji berkurang sehingga hasil biji turun. Bila penularan virus terjadi pada tanaman berumur muda, penurunan hasil berkisar antara 50-90%. Penyakit virus mosaik pada tanaman kedelai ditunjukkan pada Gambar 2.3.

SMV dapat menginfeksi tanaman kacang-kacangan: kedelai, buncis, kacang panjang, kapri (*Pisum sativum*), orok-orok (*Crotalaria spp.*) dan berbagai jenis kara (*Dolichos lablab*, *Canavalia enciformis*, *Mucuna sp.*). Virus SMV tidak aktif pada suhu 55-70 °C dan tetap infeksi pada daun kedelai kering selama 7 hari pada suhu 25-33 °C. Partikel SMV sukar dimurnikan karena cepat mengalami agregasi.

Pengendalian penyakit ini dapat dilakukan dengan cara mengurangi sumber penularan virus, menekan populasi serangga vektor dan menanam varietas toleran.

4. Target Spot

Penyebab penyakit hawar batang adalah *Corynespora cassiicola*. Gejala penyakit target spot adalah bercak coklat kemerahan timbul pada daun, batang, polong, biji, hipokotil, dan akar, dengan diameter 10-15 mm. Kadang-kadang mengalami sonasi, yaitu membentuk lingkaran seperti pada papan tembak (target). Patogen bertahan pada batang, akar, biji, dan mampu bertahan didalam tanah yang tidak diusahakan selama lebih dari 2 tahun. Infeksi hanya terjadi bila kelembaban udara relatif 80% atau lebih atau terjadi air bebas diatas daun.



Gambar 2.4 Target spot

Sumber : Balai Pengkajian Teknologi Pertanian Jawa Timur (2012)

Cuaca kering menghambat pertumbuhan jamur pada daun dan akar. Infeksi pada batang dan akar terjadi pada awal fase pertumbuhan tanaman. Gejala terlihat pada 3 minggu setelah tanaman tumbuh. Suhu tanah optimal untuk menginfeksi dan perkembangan penyakit selanjutnya adalah 15-18 °C. Pada 20 °C gejala penyakit tidak terlalu parah dan akar terbentuk normal. Patogen dapat hidup dan menyerang bermacam-macam tumbuhan (kosmopolitan), dan di

Negara tropis keberadaannya sangat melimpah. Pengendalian penyakit ini dapat dilakukan dengan cara perawatan benih terutama pada biji terinfeksi, membenam sisa tanaman terinfeksi dan aplikasi fungisida benomil, klorotalonil, kaptan. Penyakit target spot pada tanaman kedelai ditunjukkan pada Gambar 2.4.

5. Hawar Batang

Penyebab hawar batang adalah *Sclerotium rolfsii*. Gejala serangan penyakit ini adalah Infeksi yang terjadi pada pangkal batang atau sedikit dibawah permukaan tanah berupa bercak coklat muda yang cepat berubah menjadi coklat tua/warna gelap, meluas sampai ke hipokotil. Gejala layu mendadak merupakan gejala pertama yang timbul. Daun-daun yang terinfeksi mula-mula berupa bercak bulat berwarna merah sampai coklat tua, kemudian mongering dan sering menempel pada batang mati. Gejala khas patogen ini adalah miselium putih yang terbentuk pada pangkal batang, sisa daun, dan pada tanah di sekeliling tanaman sakit. Miselium tersebut menjalar ke atas batang sampai beberapa sentimeter.



Gambar 2.5 Hawar batang

Sumber : Balai Pengkajian Teknologi Pertanian Jawa Timur (2014)

Tanaman kedelai peka terhadap jamur ini sejak mulai tumbuh sampai pengisian polong. Kondisi lembab dan panas memacu perkembangan miselium yang kemudian hilang bila keadaan berubah menjadi kering. Pada keadaan lembab sekali akan terbentuk sklerotia yang berbentuk bulat seperti biji sawi dengan diameter 1-1,5 mm. Karena mempunyai lapisan dinding yang keras, sklerotium dapat dipakai untuk mempertahankan diri terhadap kekeringan, suhu tinggi dan hal lain yang merugikan. Penyakit banyak terjadi tetapi jarang berakibat serius, namun pernah mengakibatkan penurunan hasil cukup tinggi pada kedelai yang ditanam secara monokultur atau rotasi pendek dengan tanaman yang peka.

Pengendalian pada penyakit ini dapat dilakukan dengan cara memperbaiki pengolahan tanah dan drainase dan perawatan benih dengan fungisida. Penyakit hawar batang pada tanaman kedelai ditunjukkan pada Gambar 2.5.

2.3 Sistem Pakar

Pada bagian ini dijelaskan mengenai Sistem Pakar yang terdiri dari beberapa bagian, diantaranya adalah Definisi Sistem Pakar, Manfaat Sistem Pakar, Kekurangan Sistem Pakar, Ciri-ciri Sistem Pakar dan Struktur Sistem Pakar.

2.3.1 Definisi Sistem Pakar

Sistem pakar merupakan cabang dari kecerdasan buatan, yaitu sebuah sistem yang menggunakan pengetahuan seorang pakar yang kemudian pengetahuan tersebut dimasukkan ke dalam sebuah komputer dan kemudian digunakan untuk menyelesaikan masalah-masalah (Sutojo, 2011).

Sistem pakar (*expert system*) merupakan suatu sistem yang dapat mengadopsi kemampuan pakar untuk menyelesaikan masalah dalam suatu bidang keahlian tertentu. Sistem pakar yang baik dirancang agar dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja dari para ahli. Dengan sistem pakar ini, orang awampun dapat menyelesaikan masalah yang cukup rumit yang sebenarnya hanya dapat diselesaikan dengan bantuan para ahli. Bagi para ahli, sistem pakar ini juga akan membantu aktivitasnya sebagai asisten yang sangat berpengalaman (Fadli, 2010).

2.3.2 Manfaat Sistem Pakar

Sistem pakar memiliki kemampuan dan manfaat, hal tersebut yang membuat sistem pakar menjadi sangat populer. Menurut Sutojo (2011), manfaat dari sistem pakar antara lain:

1. Meningkatkan produktivitas, karena sistem pakar dapat bekerja lebih cepat daripada manusia.
2. Membuat seorang yang awam bekerja layaknya seorang pakar.
3. Meningkatkan kualitas dengan memberi nasehat 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 bosan dan kelelahan.
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 pasti.
10. Bisa digunakan sebagai media pelengkap dalam pelatihan. Pengguna pemula yang bekerja dengan sistem pakar akan menjadi lebih berpengalaman karena adanya fasilitas penjelas.

11. Meningkatkan kemampuan untuk menyelesaikan masalah karena sistem pakar mengambil sumber pengetahuan dari banyak pakar.

2.3.3 Kekurangan Sistem Pakar

Berikut merupakan kekurangan dari sistem pakar menurut Sutojo (2011), antara lain:

1. Biaya yang sangat mahal untuk membuat dan memeliharanya.
2. Sulit dikembangkan karena keterbatasan keahlian dan ketersediaan pakar.
3. Sistem pakar tidak 100% bernilai benar.

2.3.4 Ciri-ciri Sistem Pakar

Berikut merupakan ciri-ciri dari sistem pakar menurut Sutojo (2011), antara lain:

1. Terbatas pada domain keahlian tertentu.
2. Dapat memberikan penalaran untuk data-data yang 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. Keluarnya bersifat anjuran.
8. Sistem dapat mengaktifkan kaidah secara searah yang sesuai, dituntun oleh dialog dengan pengguna.

2.3.5 Struktur Sistem Pakar

Sistem pakar terdiri dari 2 bagian penting, yaitu : lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*). Lingkungan pengembang digunakan oleh pembuat sistem pakar untuk pembangunan sistem pakar baik dari segi pembangunan komponen maupun basis pengetahuan. Lingkungan konsultasi digunakan oleh seseorang bukan ahli untuk berkonsultasi (Sutojo, 2011).

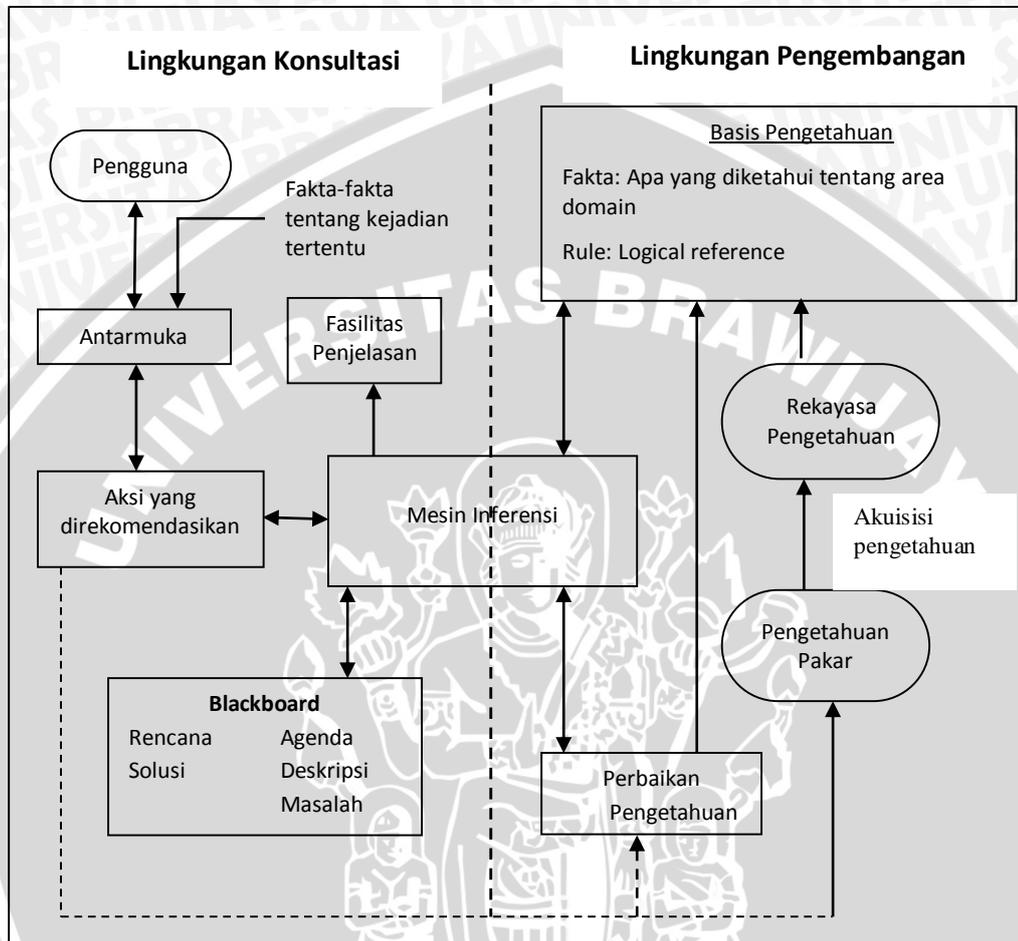
Komponen komponen dalam sistem pakar ditunjukkan pada Gambar 2.6. Penjelasan tiap komponen adalah sebagai berikut:

1. Akuisisi Pengetahuan

Akuisisi pengetahuan merupakan subsistem yang digunakan untuk memasukkan pengetahuan dari seorang pakar dengan cara merekayasa pengetahuan agar bisa diproses oleh komputer dan menaruhnya ke dalam basis pengetahuan.

2. Basis Pengetahuan

Basis pengetahuan berisi pengetahuan yang diperlukan untuk memahami, memformulasikan dan menyelesaikan masalah. Basis pengetahuan terdiri dari 2 bentuk, yaitu fakta dan rule.



Gambar 2.6 Komponen-komponen sistem pakar

Sumber: Sutojo (2011)

3. Mesin Inferensi

Mesin inferensi merupakan sebuah program yang berfungsi untuk memandu proses penalaran terhadap suatu kondisi berdasarkan pada basis pengetahuan yang ada, memanipulasi dan mengarahkan kaidah, model, dan fakta yang disimpan dalam basis pengetahuan untuk mencapai solusi atau kesimpulan. Mesin inferensi menggunakan strategi pengendalian yang berfungsi sebagai panduan arah dalam melakukan proses penalaran. Ada tiga teknik pengendalian yang digunakan yaitu forward chaining, backward chaining dan gabungan dari kedua teknik tersebut.

4. Blackboard (Daerah Kerja)

Blackboard digunakan untuk merekam hasil sementara yang akan dijadikan sebagai keputusan dan untuk menjelaskan sebuah masalah yang sedang terjadi.

Ada tiga keputusan yang dapat direkam, yaitu rencana, agenda, dan solusi. Rencana merupakan cara bagaimana menghadapi suatu masalah, agenda berupa aksi-aksi yang akan dieksekusi, dan solusi berupa aksi yang akan dibangkitkan.

5. Antarmuka Pengguna

Antarmuka digunakan sebagai media komunikasi antara pengguna dan sistem pakar. Pada bagian ini akan terjadi dialog antara sistem dengan pengguna.

6. Subsistem Penjelasan

Subsistem penjelasan berfungsi memberi penjelasan kepada pengguna, bagaimana kesimpulan dapat diambil. Kemampuan tersebut sangat penting bagi pengguna untuk mengetahui proses pemindahan keahlian pakar maupun dalam pemecahan masalah.

7. Sistem Perbaikan Pengetahuan

Kemampuan memperbaiki kemampuan dari seorang pakar digunakan untuk menganalisis pengetahuan, mempelajari kesalahan masa lalu dan memperbaikinya. Sehingga dengan cara ini akan menghasilkan basis pengetahuan yang lebih baik dan penalaran yang lebih efektif.

8. Pengguna (User)

Pengguna sistem pakar pada umumnya bukan seorang pakar, namun orang yang membutuhkan solusi atau pelatihan dari berbagai permasalahan yang ada.

2.4 Logika Fuzzy

Dalam subbab ini akan dijelaskan mengenai teori logika *fuzzy* meliputi pengertian logika *fuzzy* dan himpunan *fuzzy*.

2.4.1 Pengertian Logika Fuzzy

Logika *fuzzy* merupakan suatu cara yang tepat untuk memetakan suatu ruang input ke dalam suatu ruang output. Logika *fuzzy* diperkenalkan pertama kali oleh Lotfi A Zadeh, seorang professor dari University of California pada tahun 1965. Zadeh memperkenalkan teori yang memiliki obyek-obyek dari himpunan *fuzzy* yang memiliki batasan yang tidak presisi dalam keanggotaan dalam himpunan *fuzzy*, bukan dalam bentuk logika benar (*true*) atau salah (*false*), namun dinyatakan dalam derajat (*degree*).

Ada beberapa alasan mengapa orang menggunakan logika *fuzzy* (Kusumadewi dan Purnomo, 2010) antara lain:

1. Konsep logika *fuzzy* mudah dimengerti. Konsep matematis yang mendasari penalaran *fuzzy* sangat sederhana dan mudah dimengerti.
2. Logika *fuzzy* sangat fleksibel.
3. Logika *fuzzy* memiliki toleransi terhadap data-data yang tidak tepat.

4. Logika *fuzzy* mampu memodelkan fungsi-fungsi nonlinear yang sangat kompleks.
5. Logika *fuzzy* dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.
6. Logika *fuzzy* dapat bekerjasama dengan teknik-teknik kendali secara konvensional.
7. Logika *fuzzy* didasarkan pada bahasa alami.

2.4.2 Himpunan Fuzzy

Menurut Kusumadewi dan Purnomo (2010), pada himpunan tegas (*crisp*), nilai keanggotaan suatu item x dalam suatu himpunan A , yang sering ditulis dengan $\mu_A[x]$, memiliki 2 kemungkinan yaitu:

1. Satu (1), yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan.
2. Nol (0), yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan.

Jika dalam himpunan *crisp* nilai keanggotaan hanya ada Nol (0) dan satu (1), maka pada himpunan *fuzzy* nilai keanggotaan terletak antara rentang 0 sampai 1. Himpunan *fuzzy* memiliki 2 atribut, yaitu:

1. Linguistik, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti: KECIL, SEDANG, BESAR.
2. Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variable seperti: 15, 25, 50, dsb.

Menurut Kusumadewi dan Purnomo (2010), ada beberapa hal yang perlu diketahui dalam memahami sistem *fuzzy*, yaitu:

1. Variabel Fuzzy

Variabel *fuzzy* merupakan variable yang hendak dibahas dalam suatu sistem *fuzzy*, misal: umur, temperature, dsb.

2. Himpunan Fuzzy

Himpunan *fuzzy* merupakan suatu grup yang mewakili suatu kondisi tertentu dalam suatu variabel *fuzzy*.

3. Semesta Pembicaraan

Semesta pembicaraan merupakan keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variable *fuzzy*. Semesta pembicaraan merupakan himpunan bilangan *real* yang senantiasa bertambah secara monoton dari kiri ke kanan. Nilai semesta pembicaraan dapat berupa bilangan positif maupun bilangan negatif. Ada kalanya nilai semesta pembicaraan ini tidak dibatasi batas atasnya.

4. Domain

Domain dalam himpunan *fuzzy* merupakan keseluruhan nilai yang diijinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*. Nilai domain dapat berupa bilangan positif maupun negatif.

2.5 Klasifikasi

Klasifikasi merupakan proses untuk menyatakan suatu objek ke dalam salah satu kategori yang sudah didefinisikan sebelumnya. Tujuan dari klasifikasi adalah *record-record* yang sebelumnya belum masuk dalam kategori dapat dinyatakan kelasnya secara akurat (Bertalya, 2009). Model klasifikasi digunakan untuk:

- Pemodelan deskriptif sebagai perangkat penggambaran untuk membedakan objek-objek dari kelas berbeda.
- Pemodelan prediktif digunakan untuk memprediksi label klas untuk record yang tidak diketahui atau tidak dikenal.

Pada proses klasifikasi terdapat beberapa metode antara lain *Decision tree*, *Bayesian*, *Fuzzy*, *Neural Network*, *Support Vector Machine (SVM)* dan *K-Nearest Neighbor*.

2.6 Inisialisasi Fuzzy

Menurut James M. Keller, Michael R. Gray dan Givens James A (1985), ada tiga teknik inisialisasi yang digunakan dalam fuzzy, yaitu:

1. Inisialisasi *Crisp*
2. Inisialisasi Eksponensial
3. Inisialisasi *fuzzy 3-nearest neighbor*

Proses inisialisasi *fuzzy* digunakan untuk mengetahui nilai keanggotaan untuk setiap kelas pada data latih. Pada penelitian ini menggunakan inisialisasi *fuzzy 3-nearest neighbor* karena menurut James M. Keller, Michael R. Gray dan Givens James A (1985) dianggap memiliki nilai klasifikasi yang paling baik diantara ketiga teknik inisialisasi yang telah ditunjukkan. Perhitungan untuk inisialisasi *fuzzy* dapat dilihat pada Persamaan (2-1).

$$u_{ij} = \begin{cases} 0,51 + \left(\frac{n_j}{K}\right) * 0,49 & , \text{jika } j = i \\ \left(\frac{n_j}{K}\right) * 0,49 & , \text{jika } j \neq i \end{cases} \dots\dots\dots (2-1)$$

Dimana:

- u_{ij} : nilai keanggotaan kelas i pada vektor j
- n_j : jumlah anggota kelas j pada suatu *dataset* K
- K : banyaknya tetangga terdekat
- j : kelas target



Adapun langkah-langkah dari inialisasi *fuzzy* menurut Cui Yan (2012) adalah sebagai berikut:

1. Normalisasi setiap *record* pada data latih menggunakan persamaan 2-2.
2. Menghitung nilai jarak antar data latih menggunakan persamaan *Euclidian distance* yang ditunjukkan pada persamaan 2-3.
3. Mengurutkan hasil jarak secara ascending (dari yang terkecil ke yang terbesar).
4. Menentukan nilai *k record* inialisasi terdekat.
5. Menghitung inialisasi *fuzzy* menggunakan persamaan 2-1.

2.7 K-Nearest Neighbor

Algoritma *K-Nearest Neighbor* (KNN) adalah suatu metode yang digunakan untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Metode *K- NN* mengklasifikasikan data baru yang belum diketahui kelasnya dengan memilih data sejumlah *k* yang terletak dekat dengan data baru tersebut. *Class* terbanyak dari data terdekat yang sejumlah *k* tersebut kemudian dipilih sebagai *class* yang diprediksi untuk data yang baru. Nilai *k* umumnya ditentukan dalam jumlah ganjil untuk menghindari munculnya jumlah jarak yang sama dalam proses pengklasifikasian. Dekat atau jauhnya tetangga dihitung berdasarkan jarak *Euclidian* (Meristika, 2013). *K-Nearest Neighbor* menggunakan algoritma *supervised* dimana hasil dari sampel uji yang baru diklasifikasikan berdasarkan mayoritas dari kategori pada *K-NN*. Tujuan dari algoritma ini adalah mengklasifikasikan objek baru berdasarkan atribut dan sampel latih (Wijaya, 2014).

Adapun langkah langkah dalam metode *K-Nearest Neighbor* adalah sebagai berikut:

1. Normalisasi

Langkah awal yang dilakukan pada metode *K-Nearest Neighbor* adalah melakukan normalisasi dari setiap *record*. Hal tersebut dilakukan karena atribut cenderung memiliki nilai rentang yang bervariasi. Normalisasi yang dilakukan dalam penelitian ini adalah *Min-max Normalization*. Untuk menghitung normalisasi dapat dilihat pada Persamaan 2-2 (Meristika, 2013).

$$V' = \frac{V - \min_A}{\max_A - \min_A} \dots\dots\dots (2-2)$$

Dimana:

- V' : hasil normalisasi yang nilainya berkisar antara 0 dan 1
- V : nilai atribut A yang akan dinormalisasi
- \min_A : nilai minimum dari suatu atribut A
- \max_A : nimali maksimum dari suatu atribut A

2. Menghitung jarak *Euclidian*



Setelah melakukan normalisasi data, dilakukan perhitungan untuk mengetahui jarak terdekat antara data uji dengan data latih menggunakan persamaan *Euclidian Distance* yang ditunjukkan pada Persamaan 2-3.

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^n (a_r(x_i) - a_r(x_j))^2} \dots\dots\dots (2-3)$$

Dimana:

- $d(x_i, x_j)$: jarak *Euclidian*
- (x_i) : *record* ke – i
- (x_j) : *record* ke – j
- (a_r) : data ke – r

3. Mengurutkan berdasarkan nilai *Euclidian distance*

Setelah mendapatkan hasil semua perhitungan jarak *Euclidian*, data diurutkan berdasarkan jarak terkecil untuk menunjukkan kemiripan data uji terhadap data latih yang paling dekat. Semakin kecil nilai jarak maka semakin dekat kemiripan kelas data uji.

4. Menentukan k *record* klasifikasi terdekat

Setelah data diurutkan, diambil sebanyak K tetangga terdekat untuk memprediksi label kelas dari record baru menggunakan label kelas tetangga.

5. Target Output merupakan kelas yang mayoritas.

2.8 Fuzzy K-Nearest Neighbor

Fuzzy K-Nearest Neighbor merupakan metode klasifikasi yang menggabungkan antara teknik *fuzzy* dengan *K-Nearest Neighbor*. Algoritma ini merupakan metode klasifikasi yang digunakan untuk memprediksi data uji menggunakan nilai derajat keanggotaan terbesar dari data uji pada setiap kelas sebagai sebagai kelas hasil prediksi. Proses *Fuzzy K-Nearest Neighbor* sama dengan proses 1 – 4 dari proses KNN, namun pada proses kelima dilakukan dengan perhitungan nilai keanggotaan dari data yang baru pada masing-masing kelas menggunakan Persamaan 2-4.

$$u_i(x) = \frac{\sum_{j=1}^K u_{ij}(1/\|x-x_j\|^{\frac{2}{m-1}})}{\sum_{j=1}^K (1/\|x-x_j\|^{\frac{2}{m-1}})} \dots\dots\dots (2-4)$$

Dimana:

- $u_i(x)$: nilai keanggotaan data x ke kelas c_i
- K : jumlah tetangga terdekat yang digunakan
- $x - x_j$: selisih jarak data x ke data x_j dalam k tetangga terdekat
- m : bobot pangkat (*weight exponent*) yang besarnya $m > 1$

Adapun Langkah-langkah dari perhitungan *Fuzzy K-Nearest Neighbor* adalah sebagai berikut:



1. Normalisasi data
2. Inisialisasi fuzzy
3. Menghitung jarak *Euclidian* data uji terhadap data latih
4. Mengurutkan berdasarkan nilai *Euclidian* terkecil
5. Menentukan k record terdekat
6. Menghitung derajat keanggotaan data baru terhadap masing-masing kelas menggunakan persamaan 2-4.
7. Memilih kelas yang memiliki nilai keanggotaan terbesar sebagai hasil.

2.8.1 Contoh Perhitungan Manual Fuzzy K-Nearest Neighbor

Contoh perhitungan dengan metode *Fuzzy K-Nearest Neighbor* ini dilakukan dengan melakukan *fuzzy* terlebih dahulu. Terdapat 10 data latih dan satu data uji dengan 6 atribut data gejala yaitu G1 sampai dengan G6 dan dua diagnosa penyakit yaitu penyakit 0 dan 1. Data latih yang digunakan pada contoh perhitungan ditunjukkan pada Tabel 2.3, sedangkan data uji yang digunakan pada contoh perhitungan ditunjukkan pada Tabel 2.4.

Tabel 2.3 Contoh data latih sistem

No	G1	G2	G3	G4	G5	G6	Diagnosa
1	6	148	72	33.6	0.627	50	1
2	1	85	66	26.6	0.351	31	0
3	8	183	64	23.3	0.672	32	1
4	1	89	66	28.1	0.167	21	0
5	0	137	40	43.1	2.288	33	1
6	5	116	74	25.6	0.201	30	0
7	3	78	50	31.0	0.248	26	1
8	2	197	70	30.5	0.158	53	1
9	4	110	92	37.6	0.191	30	0
10	10	168	74	38.0	0.537	34	1

Tabel 2.4 Contoh data uji sistem

G1	G2	G3	G4	G5	G6	Diagnosa
3	126	88	39.3	0.704	27	?

Langkah pertama adalah melakukan normalisasi data menggunakan persamaan min-max (2-2). Berikut ini adalah contoh perhitungan normalisasi pada atribut G1 data pertama.



$$\begin{aligned}
 V &= 6 \\
 \text{max} &= 10 \\
 \text{min} &= 0 \\
 \text{range} &= 10-0 = 10
 \end{aligned}$$

Setelah diketahui nilai *max*, *min* dan *range* kemudian dihitung dengan persamaan (2-2).

$$\begin{aligned}
 V' &= \frac{V - \text{min}_A}{\text{max}_A - \text{min}_A} \\
 &= \frac{6 - 0}{10} = 0.6
 \end{aligned}$$

Dari perhitungan tersebut didapat nilai normalisasi *record* pertama pada atribut Hamil sebesar 0.6. Proses yang sama juga dilakukan pada seluruh *record* semua atribut. Tabel 2.5 menunjukkan data latih yang telah dinormalisasi dan Tabel 2.6 menunjukkan data uji yang telah dinormalisasi.

Tabel 2.5 Contoh data latih yang telah dinormalisasi

No	G1	G2	G3	G4	G5	G6
1	0.6	0.588	0.677	0.515	0.220	0.763
2	0.1	0.059	0.581	0.165	0.091	0.263
3	0.8	0.882	0.548	0	0.241	0.289
4	0.1	0.092	0.581	0.24	0.004	0
5	0	0.496	0.161	0.99	1	0.316
6	0.5	0.319	0.710	0.115	0.020	0.237
7	0.3	0	0.323	0.385	0.042	0.132
8	0.2	1	0.645	0.36	0	0.842
9	0.4	0.269	1	0.715	0.015	0.237
10	1	0.756	0.710	0.735	0.178	0.342

Tabel 2.6 Contoh data uji yang telah dinormalisasi

G1	G2	G3	G4	G5	G6
0.3	0.403	0.935	0.8	0.256	0.158

Langkah kedua adalah menghitung jarak antar data latih menggunakan persamaan (2-3), berikut ini merupakan perhitungan jarak data latih 1 terhadap data latih 2.

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^n (a_r(x_i) - a_r(x_j))^2}$$



$$= \sqrt{(0.6 - 0.1)^2 + (0.588 - 0.059)^2 + (0.677 - 0.581)^2 + (0.515 - 0.165)^2 + (0.220 - 0.091)^2 + (0.763 - 0.263)^2} = 0.964$$

Proses perhitungan jarak dilanjutkan ke semua *record* pada data latih menggunakan persamaan (2-3). Hasil Perhitungan jarak ditunjukkan pada Tabel 2.7.

Tabel 2.7 Tabel hasil perhitungan jarak

	1	2	3	4	5	6	7	8	9	10
1	0	0.964	0.796	1.100	1.292	0.749	1.005	0.640	0.778	0.646
2	0.964	0	1.104	0.289	1.374	0.503	0.422	1.132	0.787	1.285
3	0.796	1.104	0	1.146	1.580	0.706	1.137	0.936	1.142	0.793
4	1.100	0.289	1.146	0	1.415	0.548	0.394	1.250	0.761	1.288
5	1.292	1.374	1.580	1.415	0	1.521	1.296	1.484	1.402	1.453
6	0.749	0.503	0.706	0.548	1.521	0	0.613	0.992	0.676	0.928
7	1.005	0.422	1.137	0.394	1.296	0.613	0	1.273	0.814	1.182
8	0.640	1.132	0.936	1.250	1.484	0.992	1.273	0	1.092	1.061
9	0.778	0.787	1.142	0.761	1.402	0.676	0.814	1.092	0	0.848
10	0.646	1.285	0.793	1.288	1.453	0.928	1.182	1.061	0.848	0

Hasil perhitungan jarak yang telah diurutkan dari yang terkecil ditunjukkan pada Tabel 2.8. Hasil pengurutan jarak tersebut kemudian diurutkan berdasarkan *id* datanya yang ditunjukkan pada Tabel 2.9.

Tabel 2.8 Hasil perhitungan jarak dari terkecil

1	2	3	4	5	6	7	8	9	10
0	0	0	0	0	0	0	0	0	0
0.640	0.289	0.706	0.289	1	0.503	0.394	1	0.676	1
0.646	0.422	0.793	0.394	1.296	0.548	0.422	0.936	0.761	0.793
0.749	0.503	1	0.548	1.374	0.613	0.613	0.992	1	0.848
0.778	0.787	0.936	0.761	1.402	0.676	0.814	1.061	0.787	0.928
0.796	1	1.104	1	1.415	0.706	1	1.092	0.814	1.061
0.964	1.104	1.137	1.146	1.453	1	1.137	1.132	0.848	1.182
1.005	1.132	1.142	1.250	1.484	0.928	1.182	1.250	1.092	1.285
1.100	1.285	1.146	1.288	1.521	0.992	1.273	1.273	1.142	1.288
1.292	1.374	1.580	1.415	1.580	1.521	1.296	1.484	1.402	1.453

Tabel 2.9 Hasil perhitungan jarak terkecil berdasarkan *id* data

1	2	3	4	5	6	7	9	10
1	2	3	4	5	6	7	9	10
8	4	6	2	1	2	4	6	1

10	7	10	7	7	4	2	4	3
6	6	1	6	2	7	6	1	9
9	9	8	9	9	9	9	2	6
3	1	2	1	4	3	1	7	8
2	3	7	3	10	1	3	10	7
7	8	9	8	8	10	10	8	2
4	10	4	10	6	8	8	3	4
5	5	5	5	3	5	5	5	5

Setelah didapat id data yang telah diurutkan dari yang terkecil seperti yang ditunjukkan pada Tabel 2.9, langkah selanjutnya mengurutkan berdasarkan kelas penyakit. Pada Tabel 2.10 ditunjukkan pengurutan berdasarkan kelas penyakit.

Tabel 2.10 Hasil perhitungan jarak dari terkecil berdasarkan penyakit

1	2	3	4	5	6	7	8	9	10
1	0	1	0	1	0	1	1	0	1
1	0	0	0	1	0	0	1	0	1
1	1	1	1	1	0	0	1	0	1
0	0	1	0	0	1	0	0	1	0
0	0	1	0	0	0	0	1	0	0
1	1	0	1	0	1	1	0	1	1
0	1	1	1	1	1	1	0	1	1
1	1	0	1	1	1	1	0	1	0
0	1	0	1	0	1	1	1	1	0
1	1	1	1	1	1	1	1	1	1

Langkah ketiga adalah menentukan *k record* inialisasi terdekat dari perhitungan jarak antar data latih pada Tabel 2.10. Misal nilai $k = 3$, maka diambil 3 *k* terdekat pada data latih. Dalam Tabel 2.10 ditunjukkan tabel dengan warna merah merupakan *k record* yang terpilih. *K record* yang terpilih ditunjukkan pada Tabel 2.11.

Tabel 2.11 K record terdekat

1	2	3	4	5	6	7	8	9	10
1	0	0	0	1	0	0	1	0	1
1	1	1	1	1	0	0	1	0	1
0	0	1	0	0	1	0	0	1	0

Langkah keempat setelah proses penentuan *k* terdekat, dilakukan proses inialisasi *fuzzy* dengan menggunakan persamaan (2-1). Contoh perhitungan merupakan data 9 yang seharusnya masuk ke penyakit nol dicari nilai

keanggotaannya di kelas 1 dan kelas 0. Sehingga dapat diperoleh nilai keanggotaan data 9 pada kelas 1 sebesar 0.1633 dan pada kelas 0 sebesar 0.8367. Tabel 2.12 merupakan tabel hasil inisialisasi *fuzzy*.

$$u_{0(1)} = \left(\frac{1}{3}\right) * 0.49$$

$$= 0.1633$$

$$u_{0(0)} = 0.51 + \left(\frac{2}{3}\right) * 0.49$$

$$= 0.8367$$

Tabel 2.12 Hasil inisialisasi *fuzzy*

	1	2	3	4	5	6	7	8	9	10
1	0.837	0.163	0.837	0.163	0.837	0.163	0.510	0.837	0.163	0.837
0	0.163	0.837	0.163	0.837	0.163	0.837	0.490	0.163	0.837	0.163

Langkah kelima setelah proses inisialisasi *fuzzy* selesai adalah masuk ke Proses *K-Nearest Neighbor*. Perhitungan K-NN dimulai dengan mengetahui jarak data uji disetiap *record* data latih yang telah dinormalisasi menggunakan rumus *Euclidian distance* yang ditunjukkan pada persamaan (2-3). Contoh perhitungan merupakan jarak data uji pada data pertama sehingga didapat nilai 0.7998.

$$d(x_i, x_j) = \sqrt{(0.3 - 0.6)^2 + (0.403 - 0.588)^2 + (0.935 - 0.677)^2 + (0.8 - 0.515)^2 + (0.256 - 0.220)^2 + (0.158 - 0.763)^2} = 0.7998$$

Proses perhitungan jarak dilanjutkan antara data uji ke semua *record* data latih. Hasil perhitungan jarak ditunjukkan pada Tabel 2.13.

Tabel 2.13 Hasil perhitungan jarak data uji terhadap data latih

Record	Jarak	Target
1	0.7998	1
2	0.8523	0
3	1.1344	1
4	0.8153	0
5	1.1454	1
6	0.7933	0
7	0.8701	1
8	1.0852	1
9	0.3220	0
10	0.8425	1

Setelah itu hasil perhitungan jarak pada Tabel 2.13 diurutkan dari yang terkecil. Hasil perhitungan jarak data uji terhadap data latih yang telah diurutkan dari yang terkecil ditunjukkan pada Tabel 2.14.

Tabel 2.14 Hasil perhitungan jarak data uji terhadap latih dari yang terkecil

Record	Jarak	Target
9	0.3220	0
6	0.7933	0
1	0.7998	1
4	0.8153	0
10	0.8425	1
2	0.8523	0
7	0.8701	1
8	1.0852	1
3	1.1344	1
5	1.1454	1

Langkah keenam adalah menghitung *k record* klasifikasi terdekat, misal $k = 3$. Dalam Tabel 2.14 telah diketahui 3 *k record* berwarna merah yang terpilih yaitu 9, 6 dan 1. Langkah ketujuh masuk dalam proses *Fuzzy K-Nearest Neighbor* dengan mencari nilai keanggotaan data yang baru pada masing masing kelas menggunakan persamaan (2-8), pada perhitungan manual ini $m=2$.

$$u_1 = \frac{0.1633 \left(\left(\frac{1}{0.3220} \right)^{\frac{2}{2-1}} \right) + 0.1633 \left(\left(\frac{1}{0.7933} \right)^{\frac{2}{2-1}} \right) + 0.8367 \left(\left(\frac{1}{0.7998} \right)^{\frac{2}{2-1}} \right)}{\left(\left(\frac{1}{0.3220} \right)^{\frac{2}{2-1}} \right) + \left(\left(\frac{1}{0.7933} \right)^{\frac{2}{2-1}} \right) + \left(\left(\frac{1}{0.7998} \right)^{\frac{2}{2-1}} \right)}$$

$$= 0.2456$$

$$u_0 = \frac{0.8367 \left(\left(\frac{1}{0.3220} \right)^{\frac{2}{2-1}} \right) + 0.8367 \left(\left(\frac{1}{0.7933} \right)^{\frac{2}{2-1}} \right) + 0.1633 \left(\left(\frac{1}{0.7998} \right)^{\frac{2}{2-1}} \right)}{\left(\left(\frac{1}{0.3220} \right)^{\frac{2}{2-1}} \right) + \left(\left(\frac{1}{0.7933} \right)^{\frac{2}{2-1}} \right) + \left(\left(\frac{1}{0.7998} \right)^{\frac{2}{2-1}} \right)}$$

$$= 0.7544$$

Berdasarkan hasil perhitungan terdapat dua nilai keanggotaan, untuk menentukan kelas target, yang dipilih adalah nilai keanggotaan terbesar yaitu 0.7544, sehingga kelas target diagnosanya sama dengan 0.

2.9 Akurasi

Akurasi merupakan ukuran kedekatan antara hasil pengukuran terhadap angka sebenarnya. Perhitungan tingkat akurasi dihitung menggunakan persamaan (2-5).

$$akurasi = \frac{\sum \text{data uji benar}}{\sum \text{jumlah h total data uji}} \times 100\% \dots\dots\dots (2-5)$$

BAB 3 METODOLOGI PENELITIAN

Pada bab ini akan menjelaskan tentang metodologi penelitian pada sistem pakar diagnosa penyakit pada tanaman kedelai menggunakan metode *Fuzzy K-NN*. Tahapan dari metodologi penelitian yang dilakukan dalam penelitian ini meliputi studi literatur, pengumpulan data, analisis kebutuhan, perancangan sistem, implementasi sistem pakar, pengujian sistem pakar dan pengambilan kesimpulan. Untuk memberi kemudahan dalam memberikan menjelaskan tentang metodologi yang digunakan, maka digunakan diagram alir yang ditunjukkan pada Gambar 3.1.



Gambar 3.1 Diagram alir penelitian

3.1 Studi Literatur

Studi literatur yang digunakan dalam penelitian ini yaitu dengan cara mengumpulkan dan mempelajari berbagai macam literatur yang berhubungan dengan pemodelan sistem pakar diagnosa penyakit pada tanaman kedelai dengan metode *Fuzzy K-NN*. Sumber yang diperoleh dari studi literatur ini berupa buku, karya tulis ilmiah, jurnal, internet, penjelasan yang diberikan oleh pakar, dosen pembimbing, dan rekan – rekan mahasiswa. Adapun teori – teori yang dipelajari meliputi sistem pakar, tanaman kedelai beserta penyakit tanaman kedelai, logika fuzzy, metode K-NN, metode *Fuzzy K-NN* dan akurasi.



3.2 Pengumpulan Data

Pengumpulan data mengenai penyakit tanaman kedelai diperoleh dari Balai Pengkajian Teknologi Pertanian Jawa Timur yang beralamat di Jl. Raya Karangploso km 4 Malang. Data yang diperoleh berupa gejala yang tampak pada penyakit tanaman kedelai dengan bobot tertentu. Untuk pembobotan gejala dikonsultasikan dengan Prof. (Riset). Dr. Ir. Moh. Cholil Mahfud, M.S. sebagai pakar penyakit tanaman kedelai. Data yang diperoleh dijadikan sebagai data latih untuk pembelajaran identifikasi penyakit tanaman kedelai dengan melakukan klasifikasi menggunakan metode *fuzzy k-nearest neighbor*.

3.3 Analisis Kebutuhan

Pada tahap ini akan dijelaskan mengenai kebutuhan yang diperlukan dalam membangun sistem pakar diagnosa penyakit pada tanaman kedelai. Berikut ini merupakan analisis kebutuhan pada penelitian ini.

1. Kebutuhan perangkat keras yang meliputi:
Laptop dengan *Processor Intel(R) Core™ i3-2367M CPU @ 1.40GHz*
2. Kebutuhan perangkat lunak yang meliputi:
 - Sistem Operasi Windows 7
 - Google Chrome versi 39.0.2171.65
 - XAMPP versi 1.8.0
 - Netbeans IDE 7.2
3. Kebutuhan data yang meliputi:
 - Data gejala dan solusi pengobatan penyakit pada tanaman kedelai

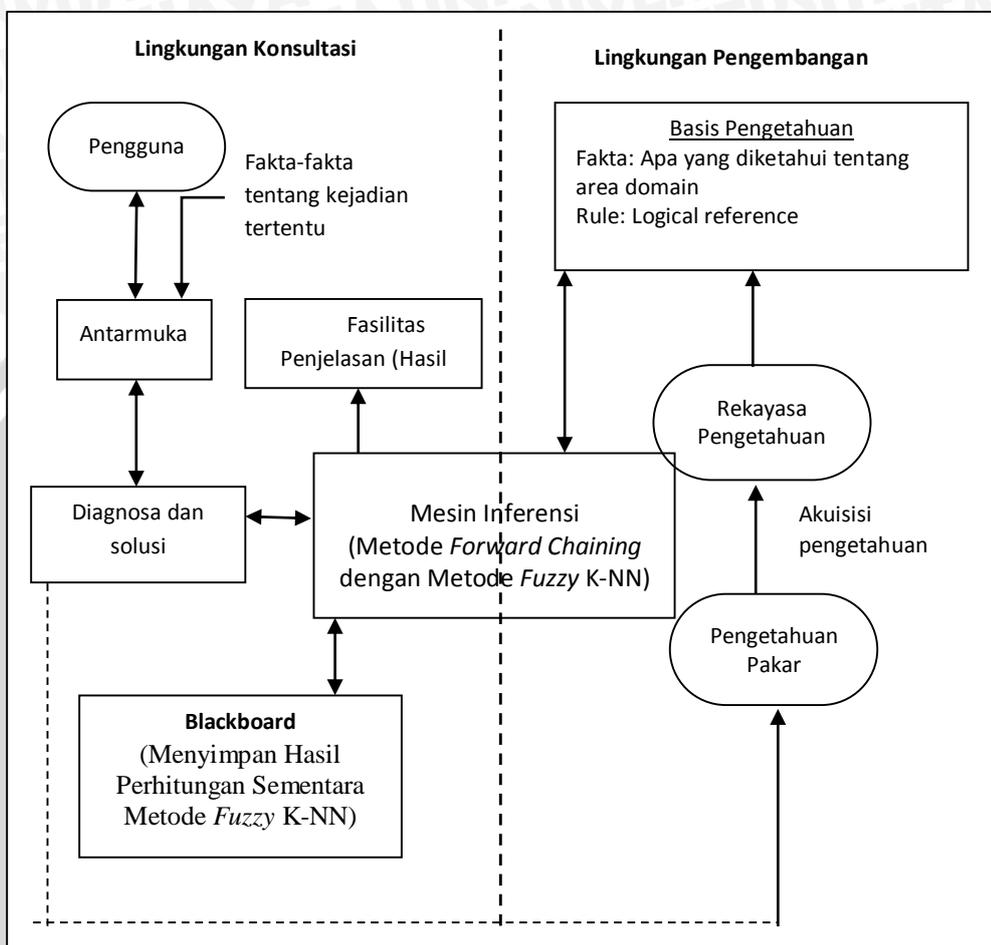
3.4 Perancangan Sistem

Pada tahap perancangan sistem ini akan dijelaskan mengenai langkah kerja sistem secara keseluruhan baik dari segi model maupun arsitektur sistem yang dibangun. Sistem perangkat lunak yang akan dibangun adalah sebuah aplikasi yang dapat mengidentifikasi penyakit pada tanaman kedelai berdasarkan gejala. Perancangan sistem ini bertujuan untuk mempermudah dalam implementasi dan pengujian sistem.

Struktur sistem pakar diagnosa penyakit pada tanaman kedelai ditunjukkan pada Gambar 3.2. Pada Gambar 3.2 merupakan struktur sistem pakar dari komponen-komponen sistem yang dibangun. Sistem pakar diagnosa penyakit pada tanaman kedelai ini menggunakan metode *Fuzzy K-NN*. Pengguna merupakan orang awam yang membutuhkan informasi mengenai permasalahan penyakit pada tanaman kedelai. Antarmuka memiliki fungsi untuk menjembatani antara pengguna dan sistem pakar. Antarmuka menampilkan halaman web yang akan digunakan oleh pengguna untuk melakukan diagnosa penyakit kedelai beserta solusinya.

Data yang dibutuhkan dalam sistem pakar diagnosa penyakit tanaman kedelai disimpan kedalam basis pengetahuan dan kemudian akan dilakukan penalaran

oleh mesin inferensi dengan menggunakan metode *forward chaining*. Perhitungan data akan diproses dengan menggunakan metode *Fuzzy K-NN* sehingga dapat memberikan hasil diagnosa sistem berupa jenis penyakit kedelai. Sedangkan *blackboard* berfungsi sebagai penyimpanan hasil perhitungan sementara dari metode *Fuzzy K-NN*.



Gambar 3.2 Struktur sistem pakar diagnosa penyakit tanaman kedelai

3.5 Implementasi Sistem Pakar

Implementasi sistem pakar yang akan dibangun menggunakan bahasa pemrograman HTML dan PHP, basis data MySQL dan alat pendukung yang lain. Implementasi dari sistem meliputi:

1. Pembuatan antarmuka pengguna berupa halaman web.
2. Penerapan metode *Fuzzy K-NN* ke dalam sistem pakar yang telah dibangun dengan menggunakan bahasa PHP.
3. Menginputkan data latih ke dalam *database* MySQL pada server *localhost* (XAMPP) yang digunakan untuk mengolah sistem.
4. *Output* yang diperoleh berupa hasil diagnosa penyakit pada tanaman kedelai dan solusi.

3.6 Pengujian Sistem Pakar

Pada tahap ini dilakukan pengujian terhadap sistem yang telah dibuat. Pengujian dilakukan dengan menggunakan pengujian fungsionalitas sistem dan pengujian akurasi sistem. Pengujian fungsionalitas sistem dilakukan dengan menggunakan metode *blackbox testing* untuk memeriksa apakah sistem telah berjalan sesuai dengan parameter yang telah ditentukan. Sedangkan pengujian akurasi sistem menggunakan variasi nilai k dan variasi jumlah data.

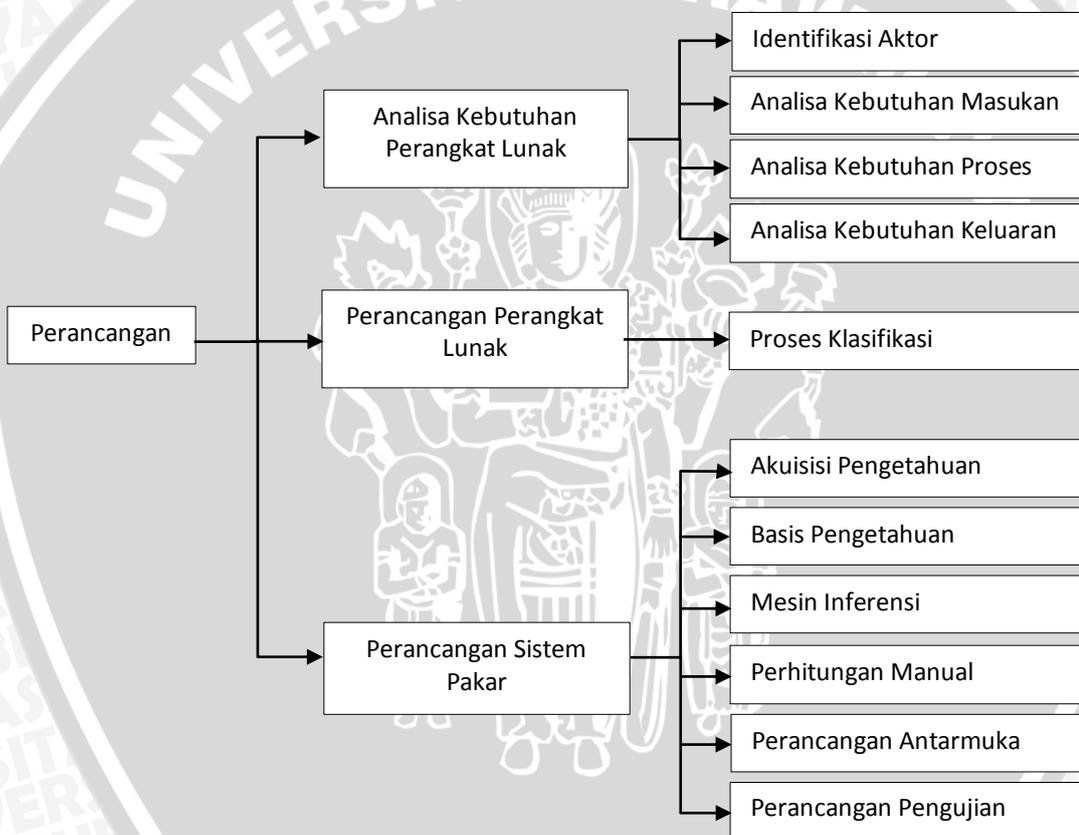
3.7 Pengambilan Kesimpulan

Pada tahap terakhir adalah pengambilan kesimpulan. Kesimpulan mengenai semua tahapan yang telah dilalui serta saran yang berkenaan dengan hasil yang telah dicapai dalam penelitian. Kesimpulan diambil dari tahap perancangan hingga analisis dan pengujian sistem. Saran berguna untuk memperbaiki kesalahan untuk pengembangan lebih lanjut serta rekomendasi dari penulis.



BAB 4 PERANCANGAN

Pada bab ini membahas tentang perancangan pada sistem “Pemodelan Sistem Pakar Untuk Identifikasi Penyakit Pada Tanaman Kedelai Menggunakan Metode *Fuzzy K-Nearest Neighbor*”. Pohon perancangan pada pemodelan sistem pakar ini meliputi tiga tahapan yaitu analisa kebutuhan perangkat lunak, perancangan perangkat lunak dan perancangan sistem pakar. Analisa kebutuhan perangkat lunak terdiri dari identifikasi actor, analisa kebutuhan masukan, analisa kebutuhan proses dan analisa kebutuhan keluaran. Perancangan perangkat lunak berupa proses klasifikasi. Perancangan sistem pakar terdiri dari akuisisi pengetahuan, basis pengetahuan, mesin inferensi, perhitungan manual, perancangan antarmuka dan perancangan pengujian. Pohon perancangan pemodelan sistem pakar pada penelitian ini ditunjukkan pada Gambar 4.1.



Gambar 4.1 Pohon perancangan sistem

4.1 Analisa Kebutuhan Perangkat Lunak

Tahap awal perancangan sistem pakar diagnosa penyakit pada tanaman kedelai menggunakan metode *fuzzy k-nearest neighbor* berupa analisa kebutuhan perangkat lunak yang terdiri dari identifikasi actor, analisa kebutuhan masukan, analisa kebutuhan proses dan analisa kebutuhan keluaran.

4.1.1 Identifikasi Aktor

Aktor merupakan pengguna yang berperan dalam menjalankan sistem pakar. Sistem pakar ini dirancang untuk satu tipe pengguna umum. Pengguna umum dapat melakukan proses diagnosa penyakit dan melihat informasi yang ada pada sistem.

4.1.2 Analisa Kebutuhan Masukan

Pada tahap analisa kebutuhan masukan ini terdiri dari kebutuhan fungsional dan kebutuhan non fungsional. Analisa kebutuhan fungsional ditunjukkan pada Tabel 4.1.

Tabel 4.1 Kebutuhan fungsional sistem

ID	Kebutuhan
KF_01	Sistem mampu menampilkan halaman utama
KF_02	Sistem mampu menampilkan halaman diagnosa
KF_03	Sistem dapat menampilkan hasil diagnosa penyakit tanaman kedelai
KF_04	Sistem dapat menampilkan halaman data
KF_05	sistem dapat menampilkan halaman tentang kami

Selain itu terdapat daftar kebutuhan non-fungsional yang digunakan untuk mengetahui spesifikasi dari kebutuhan sistem. Daftar kebutuhan non fungsional ditunjukkan pada Tabel 4.2.

Tabel 4.2 Kebutuhan non fungsional

Kebutuhan	Keterangan
<i>Availability</i>	Sistem dapat diakses oleh pengguna umum selama 24 jam
<i>Response Time</i>	Sistem memiliki respon yang cepat dalam melakukan pengolahan data

4.1.3 Analisa Kebutuhan Proses

Analisa kebutuhan proses pada sistem ini berupa proses penalaran. Pada sistem pakar diagnosa penyakit tanaman kedelai ini sistem akan melakukan proses penalaran dalam menentukan jenis penyakit tanaman kedelai berdasarkan gejala-gejala yang telah diinputkan oleh pengguna.

4.1.4 Analisa Kebutuhan Keluaran

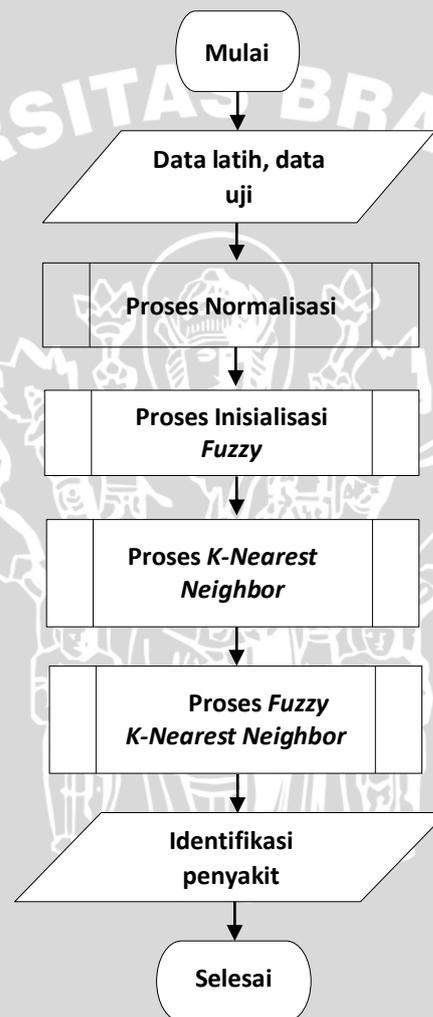
Analisa kebutuhan keluaran pada sistem ini berupa hasil diagnosa yaitu jenis penyakit pada tanaman kedelai dengan menggunakan metode *fuzzy k-nearest neighbor*. Hasil diagnosa tersebut berdasarkan gejala-gejala yang telah diinputkan

oleh pengguna. Tampilan output terdiri dari penyakit kedelai, solusi dan hasil perhitungan proses penalaran.

4.2 Perancangan Perangkat Lunak

Perancangan perangkat lunak menjelaskan tentang hubungan antar komponen dalam membangun sebuah sistem dengan metode yang telah ditentukan. Perancangan perangkat lunak pada sistem ini terdiri dari proses klasifikasi berupa alur jalannya program atau *flowchart*.

4.2.1 Proses Klasifikasi

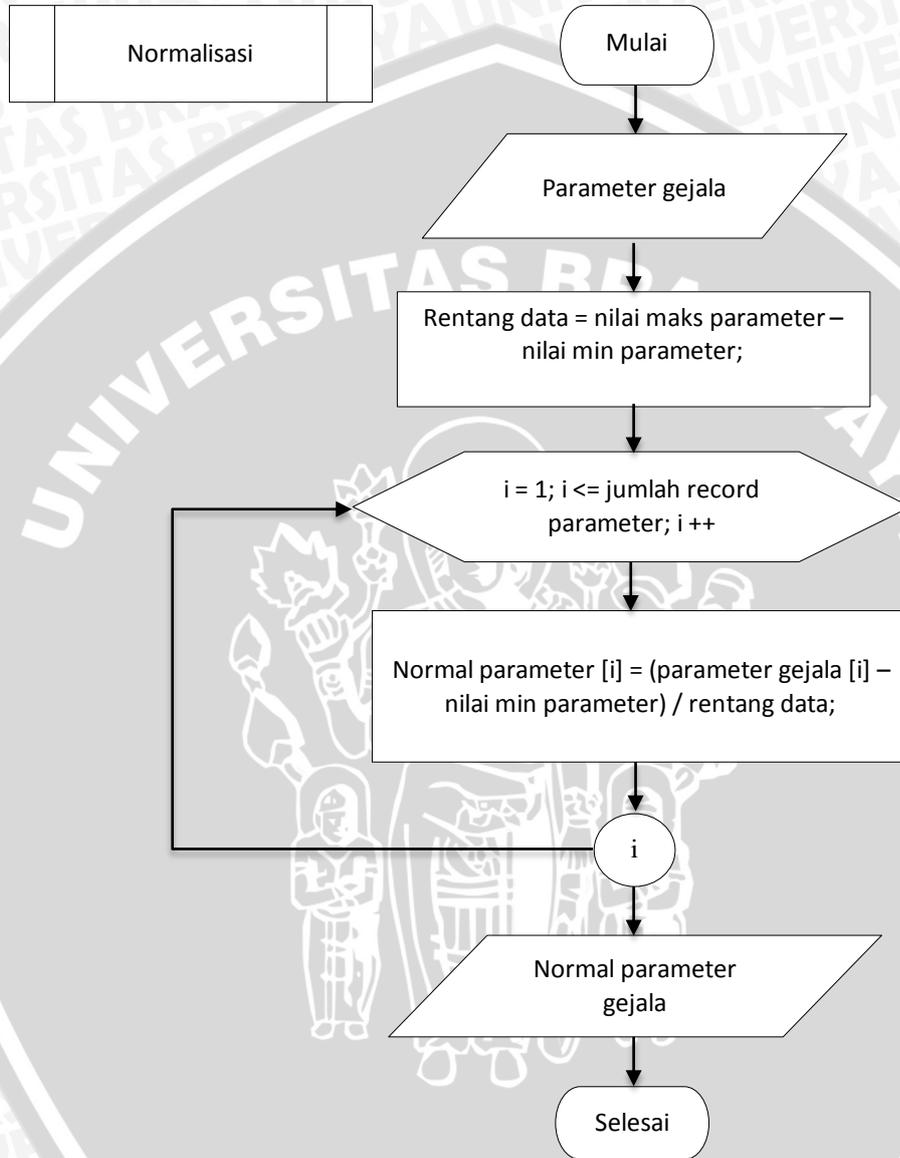


Gambar 4.2 Proses klasifikasi secara umum

Pada Gambar 4.2 ditunjukkan proses klasifikasi secara umum dalam pemrosesan data latih dan data uji gejala penyakit tanaman kedelai menggunakan metode *fuzzy k-nearest neighbor*. Terdapat 4 proses utama yaitu Normalisasi, Inisialisasi Fuzzy, Proses *K-Nearest Neighbor* dan Proses *Fuzzy K-Nearest Neighbor*.

4.2.1.1 Proses Normalisasi

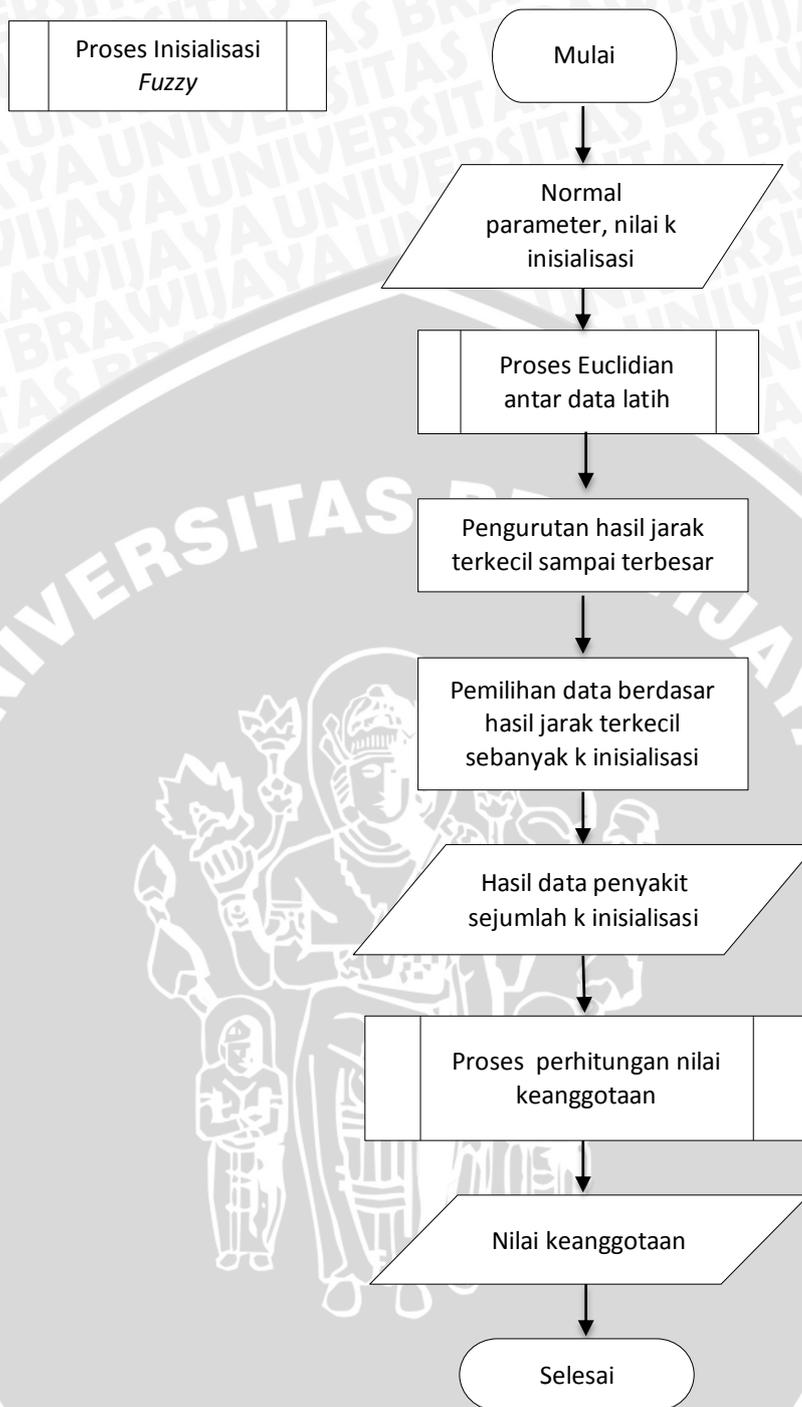
Proses Normalisasi pada tahap ini dilakukan terhadap setiap nilai parameter gejala penyakit tanaman kedelai pada data latih maupun data uji. Normalisasi data latih dan uji menggunakan persamaan (2-2). Proses normalisasi data ditunjukkan pada Gambar 4.3.



Gambar 4.3 Proses normalisasi

4.2.1.2 Proses Inisialisasi Fuzzy

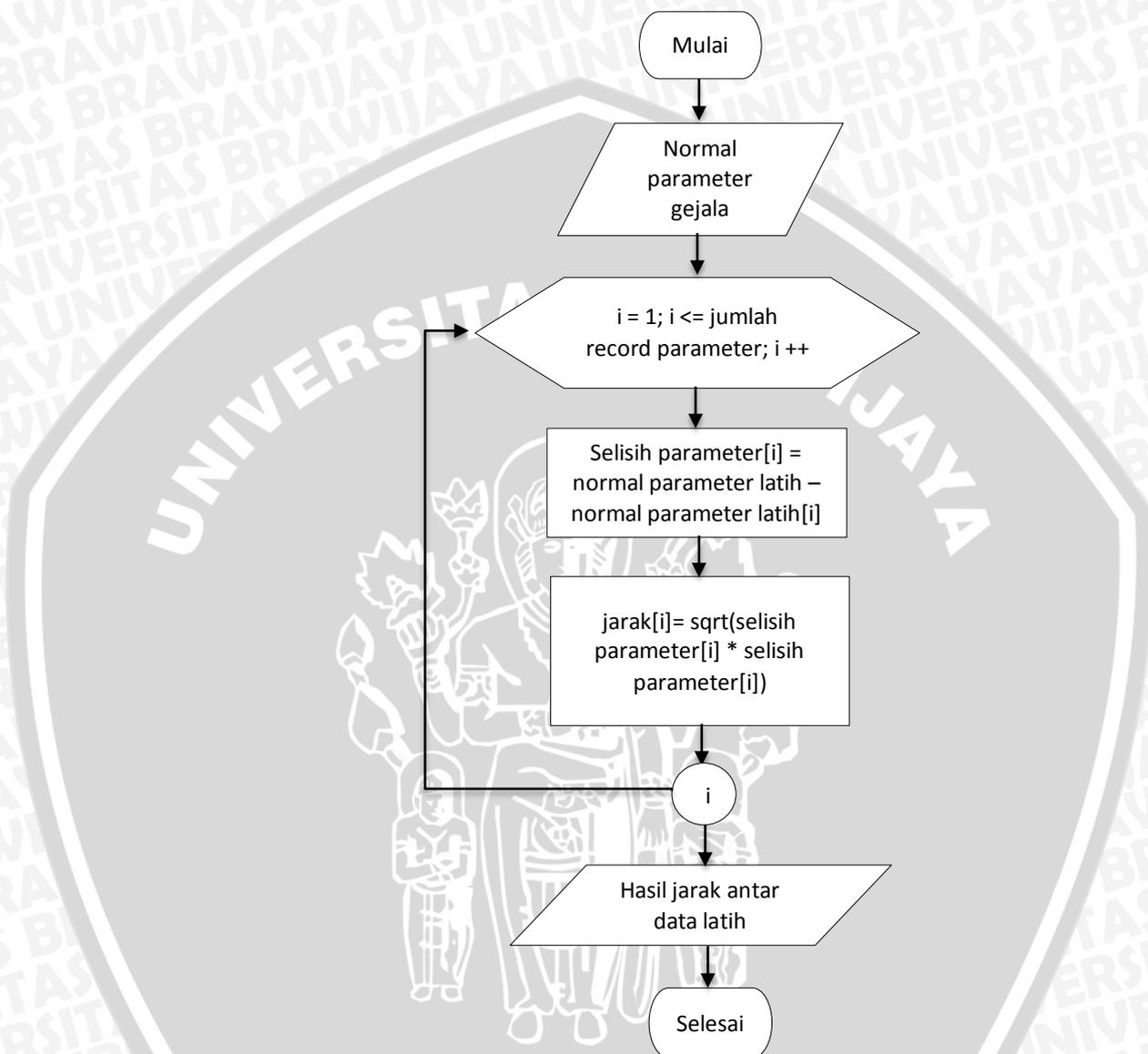
Proses Inisialisasi fuzzy digunakan untuk menghitung nilai keanggotaan pada tiap kelas, perhitungan inisialisasi fuzzy menggunakan persamaan (2-1). Proses inisialisasi fuzzy ditunjukkan pada Gambar 4.4.



Gambar 4.4 Proses inialisasi fuzzy

Dalam proses inialisasi *fuzzy* terdapat 2 proses perhitungan yaitu perhitungan jarak *Euclidian* antar data latih menggunakan persamaan (2-3). Input dari proses perhitungan jarak ini adalah parameter yang telah dinormalisasi dalam proses normalisasi dan outputnya adalah hasil jarak antar data latih. Proses perhitungan jarak *Euclidian* antar data latih ditunjukkan pada Gambar 4.5.

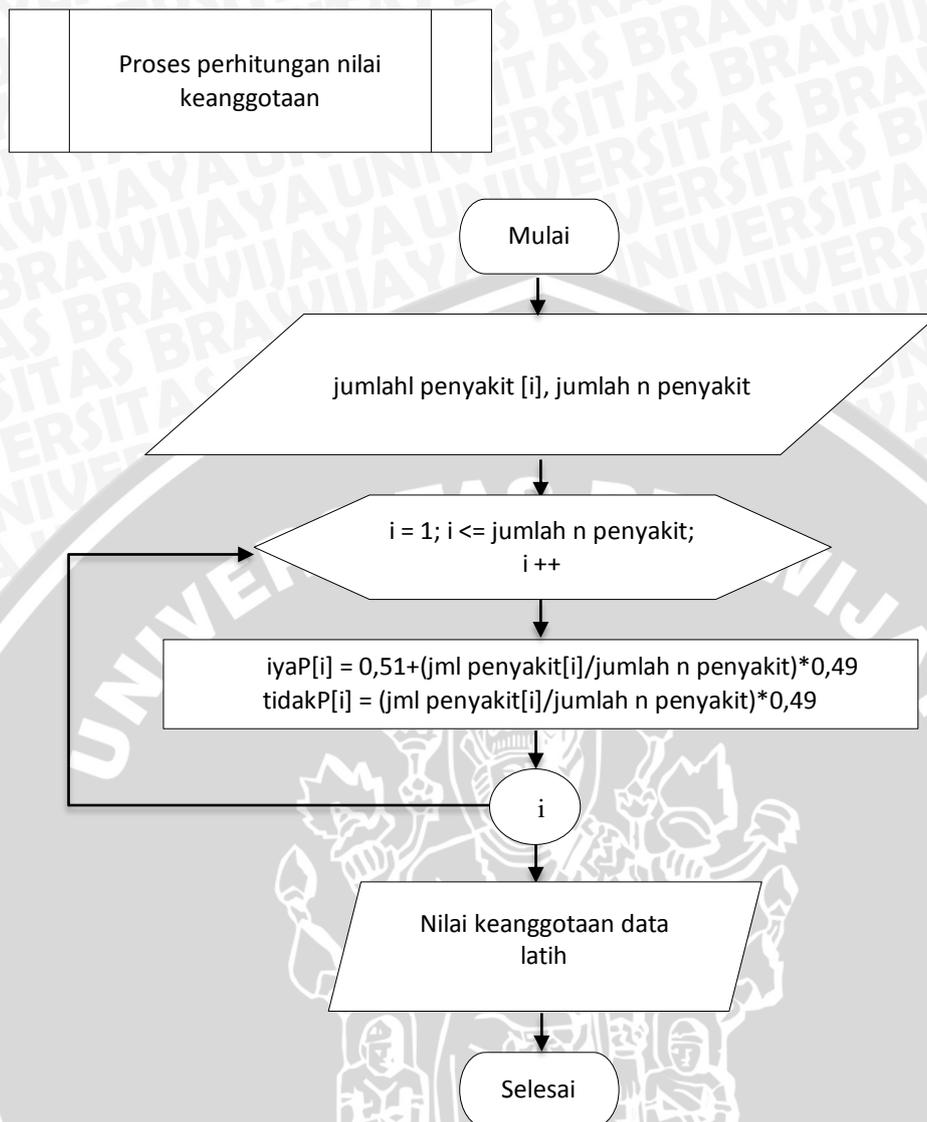
Proses Euclidian antar data latih



Gambar 4.5 Proses *euclidian distance* antar data latih

Selain perhitungan jarak antara data latih terdapat proses perhitungan nilai keanggotaan *fuzzy* menggunakan persamaan (2-1). Input dari proses ini adalah jumlah penyakit dan jumlah n penyakit yang telah ditentukan sebanyak k dan hasil jarak antar data latih yang didapat pada proses perhitungan jarak antar data latih. Proses perhitungan nilai keanggotaan *fuzzy* ditunjukkan pada Gambar 4.6.

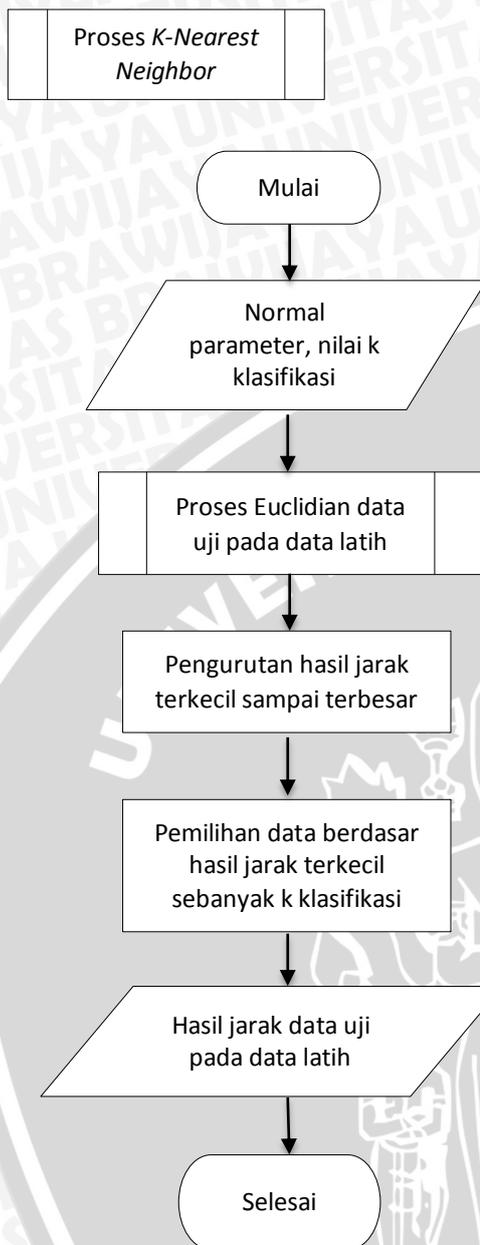




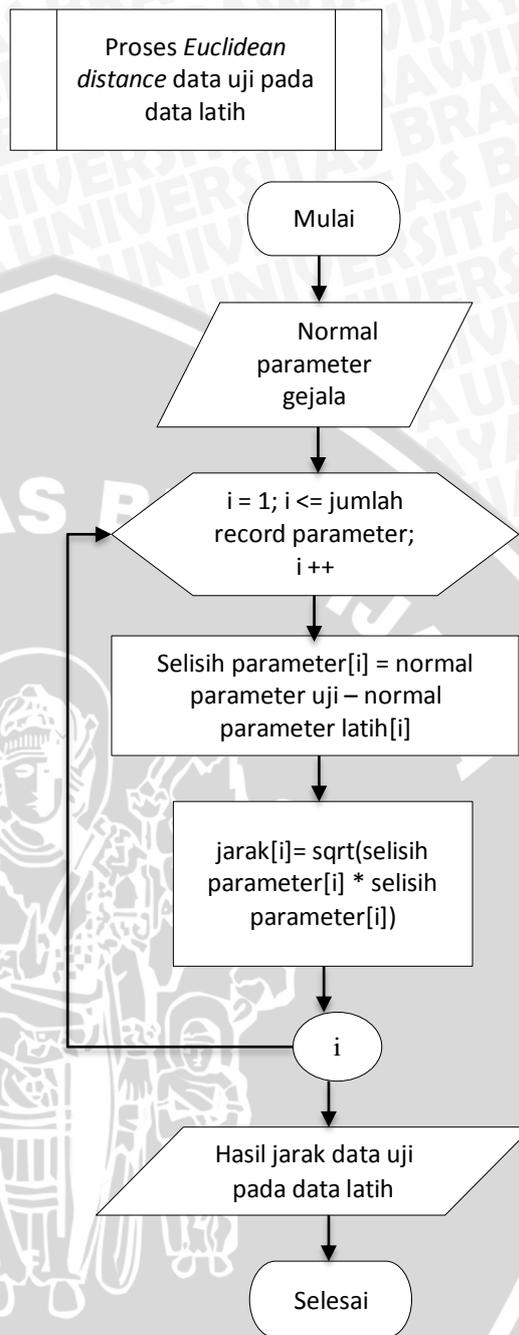
Gambar 4.6 Proses perhitungan nilai keanggotaan *fuzzy*

4.2.1.3 Proses *K-Nearest Neighbor*

Proses *K-Nearest Neighbor* dimulai dengan melakukan perhitungan jarak data uji pada data latih menggunakan persamaan (2-3). Input dari perhitungan jarak adalah parameter yang telah dinormalisasikan dan juga nilai *k*, setelah diketahui hasil jarak kemudian diurutkan dari yang terkecil sampai yang terbesar. Setelah diurutkan, ditentukan nilai hasil jarak sebanyak *k* yang kemudian akan digunakan sebagai inputan dalam proses *fuzzy k-nearest neighbor*. Proses *k-nearest neighbor* ditunjukkan pada Gambar 4.7.



Gambar 4.7 Proses *K-Nearest Neighbor*



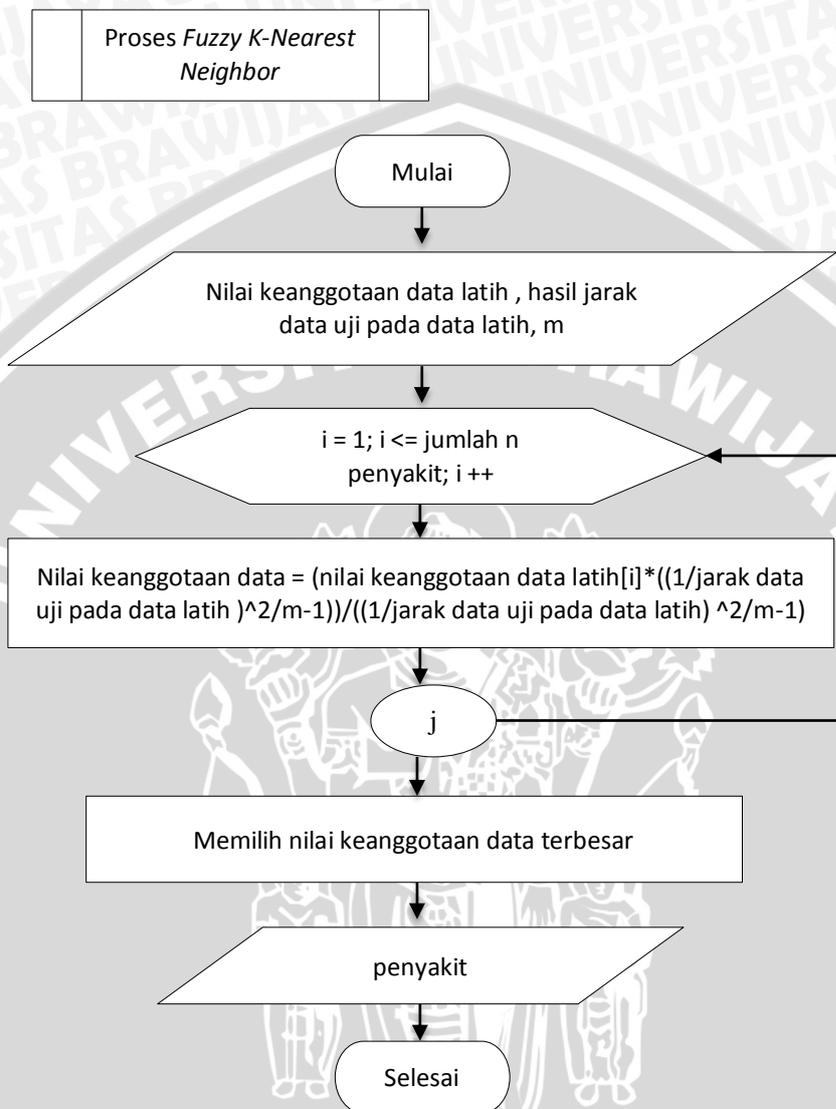
Gambar 4.8 Proses *Euclidian distance* data uji pada data latih

Gambar 4.8 merupakan proses perhitungang jarak data uji pada setiap *record* data latih menggunakan persamaan (2-3). Perhitungan jarak pada proses K-NN sama dengan perhitungan jarak pada proses inialisasi *fuzzy*, yang membedakan hanya inputan data yang digunakan.

4.2.1.4 Proses *Fuzzy K-Nearest Neighbor*

Proses *fuzzy k-nearest neighbor* merupakan proses terakhir untuk mengetahui diagnosa penyakit pada sistem pakar diagnosa penyakit tanaman kedelai. Input

dalam proses ini adalah nilai keanggotaan yang didapat dari proses inialisasi fuzzy dan hasil jarak data uji pada data latih yang didapat dari proses *K-Nearest Neighbor* serta nilai m. Output dalam proses ini adalah penyakit tanaman kedelai. Proses fuzzy *k-nearest neighbor* ditunjukkan pada Gambar 4.9.



Gambar 4.9 Proses *Fuzzy K-Nearest Neighbor*

4.3 Perancangan Sistem

Perancangan sistem ini menjelaskan mengenai perancangan sistem yang sesuai dengan arsitektur sistem pakar yang telah ditunjukkan pada gambar 3.2. Perancangan sistem pakar terdiri dari proses akuisisi pengetahuan, basis pengetahuan, mesin inferensi, *blackboard*, fasilitas penjelas dan rancangan antarmuka.

Pada penelitian ini sistem yang akan dibangun adalah sistem pakar yang diagnosa penyakit tanaman kedelai yang meneraprkan metode *Fuzzy K-NN* untuk menentukan hasil dari diagnosa. Teknik penelusuran jawaban dari proses

perhitungan ini menggunakan metode *forward chaining*. Metode *Forward Chaining* merupakan metode pengambilan keputusan yang melakukan penalaran dari suatu masalah hingga ditemukannya solusi. Permasalahan yang ada dalam penelitian ini berupa gejala-gejala penyakit yang menyerang tanaman kedelai, dari gejala-gejala tersebut kemudian dapat diambil kesimpulan berupa diagnosa penyakit yang menyerang tanaman kedelai.

4.3.1 Akuisisi Pengetahuan

Akuisisi pengetahuan digunakan untuk memindahkan suatu keahlian pakar dalam penyelesaian masalah dari sumber pengetahuan ke dalam komputer. Pada tahap ini *knowledge engineer* berusaha mendapatkan pengetahuan yang selanjutnya akan ditransfer kedalam basis pengetahuan. Pengetahuan tersebut dapat diperoleh dari pakar, referensi dari buku maupun internet, basisdata, laporan observasi dan pengalaman. Dalam penelitian ini akuisisi pengetahuan yang digunakan meliputi:

1. Wawancara

Wawancara merupakan teknik pengumpulan data untuk mendapatkan informasi secara langsung dari sumbernya melalui percakapan Tanya jawab. Tujuan dari wawancara dalam penelitian ini adalah untuk memperoleh wawasan dari pakar untuk domain masalah penyakit pada tanaman kedelai. Pada wawancara ini *knowledge engineer* mengumpulkan semua informasi tentang gejala-gejala penyakit tanaman kedelai dan pada setiap gejala akan diberikan nilai berdasarkan pembobotan yang diberikan oleh pakar. Dalam penelitian ini wawancara dilakukan di Balai Pengkajian Teknologi Pertanian (BPTP) Jawa Timur, Jl. Raya Karangploso, Km. 4, Kota Malang, Jawa Timur dengan pakar yaitu Prof. (Riset). Dr. Ir, Moh. Cholil Mahfud, M.S.

2. Aturan

Tahap aturan ini dijadikan sebagai acuan pembuatan aturan basis pengetahuan untuk melakukan diagnosa penyakit tanaman kedelai. Dalam tahap ini peneliti mengkaji hasil wawancara dengan pakar mengenai tingkat keutamaan suatu gejala pada penyakit-penyakit yang menjadi alternatif dalam sistem menggunakan nilai bobot. Daftar nilai bobot setiap gejala pada penelitian ini ditunjukkan pada Tabel 4.3.

Tabel 4.3 Daftar nilai bobot setiap gejala

No	Gejala	Keterangan Gejala	Opsi	Bobot
1	Bercak berisi uredia pada daun	Gejala Utama	Tidakada	0
			Sedikit	0.2
			Banyak	1
2	Bentuk bercak bersudut banyak	Gejala Pendukung	Tidakada	0
			Sedikit	0.15
			Banyak	0.75

3	Bercak kecil, agak bulat, kering, berwarna coklat dikelilingi warna kuning	Gejala Sekunder	Tidakada	0
			Sedikit	0.1
			Banyak	0.5
4	Bercak hijau pucat pada daun	Gejala Utama	Tidakada	0
			Sedikit	0.2
			Banyak	1
5	Bercak coklat muda atau putih pada daun	Gejala Pendukung	Tidakada	0
			Sedikit	0.15
			Banyak	0.75
6	Bercak kecil menyebabkan daun berlubang	Gejala Pendukung	Tidakada	0
			Sedikit	0.15
			Banyak	0.75
7	Daun berkerut	Gejala Pendukung	Tidakada	0
			Sedikit	0.15
			Banyak	0.75
8	Daun memiliki gambaran mosaik	Gejala Utama	Tidakada	0
			Sedikit	0.2
			Banyak	1
9	Tepi daun mengalami klorosis	Gejala Pendukung	Tidakada	0
			Sedikit	0.15
			Banyak	0.75
10	Biji Mengecil	Gejala Sekunder	Tidakada	0
			Sedikit	0.1
			Banyak	0.5
11	Bentuk lingkaran seperti papan tembak pada daun	Gejala Utama	Tidakada	0
			Sedikit	0.2
			Banyak	1
12	Biji bercak coklat kemerahan	Gejala Pendukung	Tidakada	0
			Sedikit	0.15
			Banyak	0.75
13	Bercak coklat kemerahan pada batang	Gejala Sekunder	Tidakada	0
			Sedikit	0.1
			Banyak	0.5
14	Miselium di pangkal batang	Gejala Utama	Tidakada	0
			Sedikit	0.2
			Banyak	0.1
15	Bercak coklat pada pangkal dan berkembang cepat	Gejala Pendukung	Tidakada	0
			Sedikit	0.15
			Banyak	0.75
16	Tanaman layu	Gejala Sekunder	Tidakada	0
			Sedikit	0.1
			Banyak	0.5

Sumber: [Pakar]

4.3.2 Basis Pengetahuan

Basis pengetahuan merupakan inti dari sistem pakar karena berisi representasi pengetahuan pakar yang menjadi acuan. Dalam penelitian ini terdapat 16 gejala dengan 5 penyakit. Daftar kode gejala, gejala dan penyakit pada penelitian ini ditunjukkan pada Tabel 4.4.

Tabel 4.4 Daftar kode gejala, gejala dan penyakit

Kode Gejala	Gejala	Penyakit		
G1	Bercak berisi uredia pada daun	Karat		
G2	Bentuk bercak bersudut banyak	Karat		
G3	Bercak kecil, agak bulat, kering, berwarna coklat dikelilingi warna kuning	Karat	Target Spot	Hawar Batang
G4	Bercak hijau pucat pada daun	Pustul Bakteri		
G5	Bercak coklat muda atau putih pada daun	Pustul Bakteri		
G6	Bercak kecil menyebabkan daun berlubang	Pustul Bakteri		
G7	Daun berkerut	Virus Mosaik		
G8	Daun memiliki gambaran mosaik	Virus Mosaik		
G9	Tepi daun mengalami klorosis	Virus Mosaik		
G10	Biji Mengecil	Virus Mosaik	Pustul bakteri	
G11	Bentuk lingkaran seperti papan tembak pada daun	Target Spot		
G12	Biji bercak coklat kemerahan	Target Spot		
G13	Bercak coklat kemerahan pada batang	Target Spot	Karat	
G14	Miselium di pangkal batang	Hawar Batang		
G15	Bercak coklat pada pangkal dan berkembang cepat	Hawar Batang		
G16	Tanaman layu	Hawar Batang		

Sumber: [Pakar]

4.3.3 Mesin Inferensi

Mesin inferensi berfungsi dalam melakukan penalaran atau penelusuran jawaban dengan menggunakan aturan (*rule*) berdasarkan pola tertentu yang disimpan dalam basis pengetahuan untuk mendapatkan suatu solusi. Metode penelusuran menggunakan metode *forward chaining*, dimulai dari sekumpulan fakta-fakta tentang suatu gejala yang diinputkan oleh pengguna melalui sistem, kemudian dilakukan proses pelacakan input nilai gejala dengan menggunakan perhitungan *fuzzy k-nearest neighbor*. Kesimpulan dari perhitungan akhir berupa diagnosa penyakit tanaman kedelai.

4.3.4 Perhitungan Manual

Perhitungan manual merupakan gambaran secara umum mengenai perhitungan metode dalam sistem. Perhitungan manual ini menggunakan 10 data latih dan 3 data uji. Dalam perhitungan manual berikut merupakan perhitungan untuk mengetahui diagnosa penyakit pada data uji pertama. Data latih penyakit tanaman kedelai ditunjukkan pada Tabel 4.5 dan Data uji penyakit tanaman kedelai ditunjukkan pada Tabel 4.6.

Tabel 4.5 Data latih penyakit tanaman kedelai

Data	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	G16	Diag.
1	0.2	0.75	0.5	0	0	0	0	0	0	0	0	0	0.1	0	0	0	1
2	1	0.75	0.5	0	0	0	0.75	0	0	0	0	0.75	0.5	0	0.75	0.5	1
3	0	0	0	0.2	0.15	0.75	0.15	1	0.15	0.5	0.2	0.75	0	0	0	0	2
4	0.2	0.75	0	0.2	0.75	0	0.75	0	0	0	0.2	0	0.1	0	0	0.5	2
5	0	0	0	0	0	0	0.15	1	0.75	0	0	0	0	0	0	0	3
6	0	0	0	0	0	0	0.15	1	0.15	0.5	0	0	0	0	0	0	3
7	0	0	0.1	0	0	0	0	0	0	0	0.2	0.75	0.1	0	0	0	4
8	0	0	0	1	0	0	0	0.2	0	0	1	0	0.5	0	0	0.5	4
9	0	0	0.1	0	0	0	0	0	0	0	0	0	0	0.2	0.75	0.1	5
10	0	0	0.5	0	0	0	0	1	0.75	0.5	1	0.75	0	1	0.75	0.5	5

Tabel 4.6 Data uji penyakit tanaman kedelai

Data	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	G16	Diag.
1	0	0	0	0	0	0	0	0	0	0.1	0.2	0	0.1	0.2	0.75	0	?
2	0	0	0	0	0	0	0.15	1	0.15	0	0	0	0.5	0	0	0.1	?
3	0.2	0	0	0.2	0	0	0.15	0	0	0	0	0.15	0	0	0.15	0.1	?

Langkah pertama adalah melakukan normalisasi data menggunakan persamaan min-max (2-2). Berikut ini adalah contoh perhitungan normalisasi pada data latih atribut G1 data 1.

$$\begin{aligned} V &= 1 \\ \max &= 1 \\ \min &= 0 \\ \text{range} &= 1-0 = 1 \end{aligned}$$

Setelah diketahui nilai *max*, *min* dan *range* kemudian dihitung dengan persamaan (2-2).

$$\begin{aligned} V' &= \frac{V - \min_A}{\max_A - \min_A} \\ &= \frac{1 - 0}{1} = 1 \end{aligned}$$

Dari perhitungan tersebut didapat nilai normalisasi data latih pertama pada atribut G1 sebesar 1. Proses yang sama juga dilakukan di setiap data pada semua atribut gejala. Data latih yang telah dinormalisasi ditunjukkan pada Tabel 4.7 dan Data uji pertama yang telah dinormalisasi ditunjukkan pada Tabel 4.8.

Tabel 4.7 Normalisasi data latih penyakit tanaman kedelai

Data	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	G16	Diag .
1	1	1	1	0	0	0	0	0	0	0	0	0	0.2	0	0	0	1
2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
3	0	0	0	1	1	1	0	0	0	0.2	0	0	0	0	0	0	2
4	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	2
5	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	3
6	0	0	0	0	0	0	1	1	0.2	1	0	0	0	0	0	0	3
7	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	4
8	0	0	1	0	0	0	0	0	0	0	1	0.2	1	0	0	0	4
9	0	0	0.2	0	0	0	0	0	0	0	0	0	0	1	1	1	5
10	0	0	0.2	0	0	0	0	0	0	0	0	0	0	1	1	0.2	5

Tabel 4.8 Normalisasi data uji penyakit tanaman kedelai

G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	G16	Diag .
0	0	0	0	0	0	0	0	0	0.2	0.2	0	0.2	0.2	1	0	?

Langkah selanjutnya masuk dalam proses *fuzzy*, terdapat beberapa proses yang pertama adalah menghitung jarak antar data latih yang telah dinormalisasi menggunakan persamaan (2-3), berikut ini merupakan perhitungan jarak data latih 1 terhadap data latih 2.

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^n (a_r(x_i) - a_r(x_j))^2}$$

$$= \sqrt{(0.2 - 1)^2 + (1 - 1)^2 + (1 - 1)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 1)^2 + (0.2 - 1)^2 + (0 - 0)^2 + (0 - 1)^2 + (0 - 1)^2} = 2.298$$

Proses perhitungan jarak dilanjutkan ke semua data pada data latih menggunakan persamaan (2-3). Hasil Perhitungan jarak ditunjukkan pada Tabel 4.9.

Tabel 4.9 Tabel hasil perhitungan jarak

Data	1	2	3	4	5	6	7	8	9	10
1	0	2.298	2.506	2.020	2.030	2.040	1.649	2.392	1.673	3.013
2	2.298	0	3.130	2.315	3.105	3.111	2.514	2.835	2.514	3.000
3	2.506	3.130	0	2.720	1.939	1.456	1.800	2.653	2.307	2.530
4	2.020	2.315	2.720	0	2.408	2.417	2.263	2.236	2.209	3.280
5	2.030	3.105	1.939	2.408	0	1.281	1.778	2.383	1.778	2.653
6	2.040	3.111	1.456	2.417	1.281	0	1.789	2.392	1.789	2.585
7	1.649	2.514	1.800	2.263	1.778	1.789	0	2.088	1.470	2.706
8	2.392	2.835	2.653	2.236	2.383	2.392	2.088	0	2.182	2.939
9	1.673	2.514	2.307	2.209	1.778	1.789	1.470	2.182	0	2.631
10	3.013	3.000	2.530	3.280	2.653	2.585	2.706	2.939	2.631	0

Hasil perhitungan jarak yang telah diurutkan dari yang terkecil ditunjukkan pada Tabel 4.10.

Tabel 4.10 Hasil perhitungan jarak dari terkecil

1	2	3	4	5	6	7	8	9	10
0	0	0	0	0	0	0	0	0	0
1.649	2.298	1.456	2.020	1.281	1.281	1.470	2.088	1.470	2.530
1.673	2.315	1.800	2.209	1.778	1.456	1.649	2.182	1.673	2.585
2.020	2.514	1.939	2.236	1.778	1.789	1.778	2.236	1.778	2.631
2.030	2.514	2.307	2.263	1.939	1.789	1.789	2.383	1.789	2.653
2.040	2.835	2.506	2.315	2.030	2.040	1.800	2.392	2.182	2.706
2.298	3.000	2.530	2.408	2.383	2.392	2.088	2.392	2.209	2.939
2.392	3.105	2.653	2.417	2.408	2.417	2.263	2.653	2.307	3.000
2.506	3.111	2.720	2.720	2.653	2.585	2.514	2.835	2.514	3.013

3.013	3.130	3.130	3.280	3.105	3.111	2.706	2.939	2.631	3.280
-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

Hasil perhitungan jarak yang telah diurutkan pada Tabel 4.10 kemudian disesuaikan berdasarkan id datanya. Hasil perhitungan jarak dari yang terkecil berdasarkan id data ditunjukkan pada Tabel 4.11. Setelah itu diurutkan berdasarkan kelas penyakitnya. Hasil perhitungan jarak dari yang terkecil berdasarkan kelas penyakit ditunjukkan pada Tabel 4.12.

Tabel 4.11 Hasil perhitungan jarak dari yang terkecil berdasarkan id data

1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10
7	1	6	1	6	5	9	7	7	3
9	4	7	9	7	3	1	9	1	6
4	7	5	8	9	7	5	4	5	9
5	9	9	7	3	9	6	5	6	5
6	8	1	2	1	1	3	1	8	7
2	10	10	5	8	8	8	6	4	8
8	5	8	6	4	4	4	3	3	2
3	6	4	3	10	10	2	2	2	1
10	3	2	10	2	2	10	10	10	4

Tabel 4.12 Hasil perhitungan jarak dari terkecil berdasakan penyakit

1	2	3	4	5	6	7	8	9	10
1	1	2	2	3	3	4	4	5	5
4	1	3	1	3	3	5	4	4	2
5	2	4	5	4	2	1	5	1	3
2	4	3	4	5	4	3	2	3	5
3	5	5	4	2	5	3	3	3	3
3	4	1	1	1	1	2	1	4	4
1	5	5	3	4	4	4	3	2	4
4	3	4	3	2	2	2	2	2	1
2	3	2	2	5	5	1	1	1	1
5	2	1	5	1	1	5	5	5	2

Selanjutnya menentukan *k record* klasifikasi terdekat dari perhitungan jarak antar data latih, pada Tabel 4.12 tabel berwarna merah merupakan 3 *k record* klasifikasi terdekat, maka diambil 3 *k* inialisasi terdekat pada data latih. Pada Tabel 4.13 ditunjukkan *record* yang terpilih.

Tabel 4.13 K record terdekat

1	2	3	4	5	6	7	8	9	10
4	1	3	1	3	3	5	4	4	2
5	2	4	5	4	2	1	5	1	3
2	4	3	4	5	4	3	2	3	5

Setelah proses penentuan k terdekat, dilakukan proses inialisasi *fuzzy* dengan menggunakan persamaan (2-1). Tabel 4.14 merupakan tabel hasil inialisasi *fuzzy*. Contoh perhitungan menggunakan data pertama yang seharusnya masuk ke dalam kelas penyakit 1 dicari nilai keanggotaannya pada setiap kelas.

Nilai keanggotaan data 1 pada kelas 1

$$u_{1(1)} = 0.51 + \left(\frac{0}{3}\right) * 0.49$$

$$= 0.510$$

Nilai keanggotaan data 1 pada kelas 2

$$u_{1(2)} = \left(\frac{1}{3}\right) * 0.49$$

$$= 0.163$$

Nilai keanggotaan data 1 pada kelas 3

$$u_{1(3)} = \left(\frac{0}{3}\right) * 0.49$$

$$= 0$$

Nilai keanggotaan data 1 pada kelas 4

$$u_{1(4)} = \left(\frac{1}{3}\right) * 0.49$$

$$= 0.163$$

Nilai keanggotaan data 1 pada kelas 5.

$$u_{1(2)} = \left(\frac{1}{3}\right) * 0.49$$

$$= 0.163$$

Tabel 4.14 Hasil inialisasi *fuzzy*

	1	2	3	4	5	6	7	8	9	10
1	0.510	0.673	0	0.163	0	0	0.163	0	0.163	0
2	0.163	0.163	0.510	0.510	0	0.163	0	0.163	0	0.163
3	0	0	0.327	0	0.673	0.673	0.163	0	0.163	0.163
4	0.163	0.163	0.163	0.163	0.163	0.163	0.510	0.673	0.163	0
5	0.163	0	0	0.163	0.163	0	0.163	0.163	0.673	0.673



Setelah proses inialisasi *fuzzy* selesai, selanjutnya masuk ke Proses *K-Nearest Neighbor*. Perhitungan K-NN dimulai dengan mengetahui jarak data uji yang telah dinormalisasi yang ditunjukkan pada Tabel 4.8 di setiap *record* data latih yang telah dinormalisasi yang ditunjukkan pada Tabel 4.7 menggunakan rumus *Euclidian distance* yang ditunjukkan pada persamaan (2-3). Contoh perhitungan merupakan perhitungan jarak data uji ke data latih pertama.

$$d(x_i, x_j) = \sqrt{(0 - 0.2)^2 + (0 - 1)^2 + (0 - 1)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0.2 - 0)^2 + (0.2 - 0)^2 + (0 - 0)^2 + (0.2 - 0.1)^2 + (0.2 - 0.2)^2 + (1 - 0.75)^2 + (0 - 0)^2}$$

$$= 1.7776$$

Proses perhitungan jarak dilanjutkan antara data uji pertama ke semua *record* data latih. Setelah itu hasil perhitungan jarak diurutkan dari yang terkecil. Hasil perhitungan jarak data uji terhadap data latih ditunjukkan pada Tabel 4.14 dan jarak data uji terhadap data latih yang telah diurutkan dari yang terkecil ditunjukkan pada Tabel 4.15.

Tabel 4.15 Hasil perhitungan jarak data uji terhadap data latih

Record	Jarak	Target
1	1.7776	1
2	2.6000	1
3	2.2091	2
4	2.2716	2
5	1.7889	3
6	1.6852	3
7	1.4560	4
8	2.0976	4
9	0.4472	5
10	2.6382	5

Tabel 4.16 Hasil jarak data uji terhadap data latih dari yang terkecil

Record	Jarak	Target
9	0.4472	5
7	1.4560	4
6	1.6852	3
1	1.7776	1
5	1.7889	3



8	2.0976	4
3	2.2091	2
4	2.2716	2
2	2.6000	1
10	2.6382	5

Langkah selanjutnya adalah menghitung k klasifikasi *record* terdekat, misal k = 3. Dalam tabel 4.15 telah diketahui 3 *record* yang terpilih yaitu 9, 7 dan 6.

Setelah diketahui k *record* terdekat, langkah selanjutnya masuk dalam proses *Fuzzy K-Neasrest Neighbor* dengan mencari nilai keanggotaan data yang baru pada masing masing kelas menggunakan persamaan (2-8), pada perhitungan manual ini m=2.

$$u_1 = \frac{0.163 \left(\left(\frac{1}{0.4472} \right)^{\frac{2}{2-1}} \right) + 0.163 \left(\left(\frac{1}{1.4560} \right)^{\frac{2}{2-1}} \right) + 0 \left(\left(\frac{1}{1.6852} \right)^{\frac{2}{2-1}} \right)}{\left(\left(\frac{1}{0.4472} \right)^{\frac{2}{2-1}} \right) + \left(\left(\frac{1}{1.4560} \right)^{\frac{2}{2-1}} \right) + \left(\left(\frac{1}{1.6852} \right)^{\frac{2}{2-1}} \right)}$$

$$= 0.1535$$

$$u_2 = \frac{0 \left(\left(\frac{1}{0.4472} \right)^{\frac{2}{2-1}} \right) + 0 \left(\left(\frac{1}{1.4560} \right)^{\frac{2}{2-1}} \right) + 0.163 \left(\left(\frac{1}{1.6852} \right)^{\frac{2}{2-1}} \right)}{\left(\left(\frac{1}{0.4472} \right)^{\frac{2}{2-1}} \right) + \left(\left(\frac{1}{1.4560} \right)^{\frac{2}{2-1}} \right) + \left(\left(\frac{1}{1.6852} \right)^{\frac{2}{2-1}} \right)}$$

$$= 0.0099$$

$$u_3 = \frac{0.163 \left(\left(\frac{1}{0.4472} \right)^{\frac{2}{2-1}} \right) + 0.163 \left(\left(\frac{1}{1.4560} \right)^{\frac{2}{2-1}} \right) + 0.673 \left(\left(\frac{1}{1.6852} \right)^{\frac{2}{2-1}} \right)}{\left(\left(\frac{1}{0.4472} \right)^{\frac{2}{2-1}} \right) + \left(\left(\frac{1}{1.4560} \right)^{\frac{2}{2-1}} \right) + \left(\left(\frac{1}{1.6852} \right)^{\frac{2}{2-1}} \right)}$$

$$= 0.1942$$

$$u_4 = \frac{0.163 \left(\left(\frac{1}{0.4472} \right)^{\frac{2}{2-1}} \right) + 0.510 \left(\left(\frac{1}{1.4560} \right)^{\frac{2}{2-1}} \right) + 0.163 \left(\left(\frac{1}{1.6852} \right)^{\frac{2}{2-1}} \right)}{\left(\left(\frac{1}{0.4472} \right)^{\frac{2}{2-1}} \right) + \left(\left(\frac{1}{1.4560} \right)^{\frac{2}{2-1}} \right) + \left(\left(\frac{1}{1.6852} \right)^{\frac{2}{2-1}} \right)}$$

$$= 0.1914$$

$$u_5 = \frac{0.673 \left(\left(\frac{1}{0.4472} \right)^{\frac{2}{2-1}} \right) + 0.163 \left(\left(\frac{1}{1.4560} \right)^{\frac{2}{2-1}} \right) + 0 \left(\left(\frac{1}{1.6852} \right)^{\frac{2}{2-1}} \right)}{\left(\left(\frac{1}{0.4472} \right)^{\frac{2}{2-1}} \right) + \left(\left(\frac{1}{1.4560} \right)^{\frac{2}{2-1}} \right) + \left(\left(\frac{1}{1.6852} \right)^{\frac{2}{2-1}} \right)}$$

$$= 0.5913$$

Berdasarkan hasil perhitungan terdapat lima nilai keanggotaan, untuk menentukan kelas target, yang dipilih adalah nilai keanggotaan terbesar yaitu 0.5913, sehingga data uji pertama yang ditunjukkan pada Tabel 4.6 masuk dalam diagnosa penyakit 5.

4.3.5 Perancangan Antarmuka

Antarmuka sistem pakar merupakan sarana penghubung dan komunikasi antara pengguna dengan sistem. Sistem pakar dalam penelitian ini dirancang untuk dapat menampilkan halaman beranda, halaman diagnosa, halaman data dan halaman tentang kami.

4.3.5.1 Antarmuka Tampilan Beranda

Pada antarmuka beranda berisi gambar *slide show* produk tanaman kedelai, tentang info tanaman kedelai secara umum dan penyakit-penyakit yang menyerang tanaman kedelai lebih tapenya penyakit yang menghambat produksi tanaman kedelai. Antarmuka beranda ditunjukkan pada Gambar 4.10.

4.3.5.2 Antarmuka Tampilan Diagnosa

Pada antarmuka tampilan diagnosa berisi pertanyaan berupa gejala-gejala penyakit yang akan diinputkan oleh pengguna. Pertanyaan tersebut memiliki opsi jawaban yaitu tidak ada, sedikit dan banyak. Setelah mengisi seluruh pertanyaan gejala, pengguna menekan tombol lihat diagnosa untuk mengetahui hasil diagnosa dari gejala-gejala yang telah diinputkan. Antarmuka tampilan diagnosa ditunjukkan pada Gambar 4.11.

Sistem Pakar Diagnosa Tanaman Kedelai		
Beranda	Diagnosa	Data
Slide Show Produk Kedelai		
Tentang kedelai		
Penyakit – Penyakit Kedelai		

Gambar 4.10 Antarmuka tampilan beranda

Sistem Pakar Diagnosa Tanaman Kedelai		
Beranda	Diagnosa	Data
<div style="border: 1px solid black; padding: 20px; margin: 10px auto; width: 80%;"> <p style="text-align: center;">Kuisiner Gejala-gejala</p> </div>		
<div style="border: 1px solid black; padding: 5px; margin: 10px auto; width: 15%;"> <p>Diagnosa</p> </div>		

Gambar 4.11 Antarmuka tampilan diagnosa

4.3.5.3 Antamuka Tampilan Data

Antarmuka tampilan data berisi data latih dan data uji yang digunakan dalam penelitian ini. Antarmuka tampilan data ditunjukkan pada Gambar 4.12.

Sistem Pakar Diagnosa Tanaman Kedelai		
Beranda	Diagnosa	Data
<div style="border: 1px solid black; padding: 20px; margin: 10px auto; width: 80%;"> <p style="text-align: center;">Data Latih</p> </div>		
<div style="border: 1px solid black; padding: 20px; margin: 10px auto; width: 80%;"> <p style="text-align: center;">Data Uji</p> </div>		

Gambar 4.12 Antarmuka tampilan data

4.3.6 Perancangan Pengujian

Pada tahap ini dilakukan perancangan pengujian terhadap sistem yang telah dibuat. Pengujian dilakukan dengan menggunakan pengujian fungsionalitas sistem

dan pengujian akurasi sistem. Pengujian fungsionalitas adalah pengujian untuk mengetahui apakah sistem yang dibangun telah sesuai dengan kebutuhan-kebutuhan sistem yang telah ditentukan. Skenario pengujian fungsionalitas ditunjukkan pada Tabel 4.16.

Tabel 4.17 Skenario pengujian fungsionalitas

No	Skenario Pengujian	Hasil yang Diharapkan
1.	Mengklik menu beranda	Sistem menampilkan halaman beranda
2.	Mengklik menu diagnosa	Sistem menampilkan halaman diagnosa
3.	Mengklik tombol diagnosa	Sistem menampilkan halaman hasil diagnosa
4.	Mengklik menu data	Sistem menampilkan halaman data

Pengujian akurasi adalah untuk mengetahui jumlah kecocokan data antara hasil diagnosa sistem dengan hasil diagnosa pakar. Pengujian akurasi dilakukan melalui tiga tahap yaitu pengujian variasi nilai k, pengujian akurasi variasi jumlah data dengan menggunakan nilai k terbaik dari pengujian variasi nilai k dan pengujian perbandingan metode *fuzzy* k-nn dan k-nn. Contoh pengujian akurasi nilai k ditunjukkan pada Tabel 4.17 dan contoh pengujian akurasi variasi jumlah data ditunjukkan pada Tabel 4.18.

Tabel 4.18 Contoh pengujian akurasi variasi nilai k

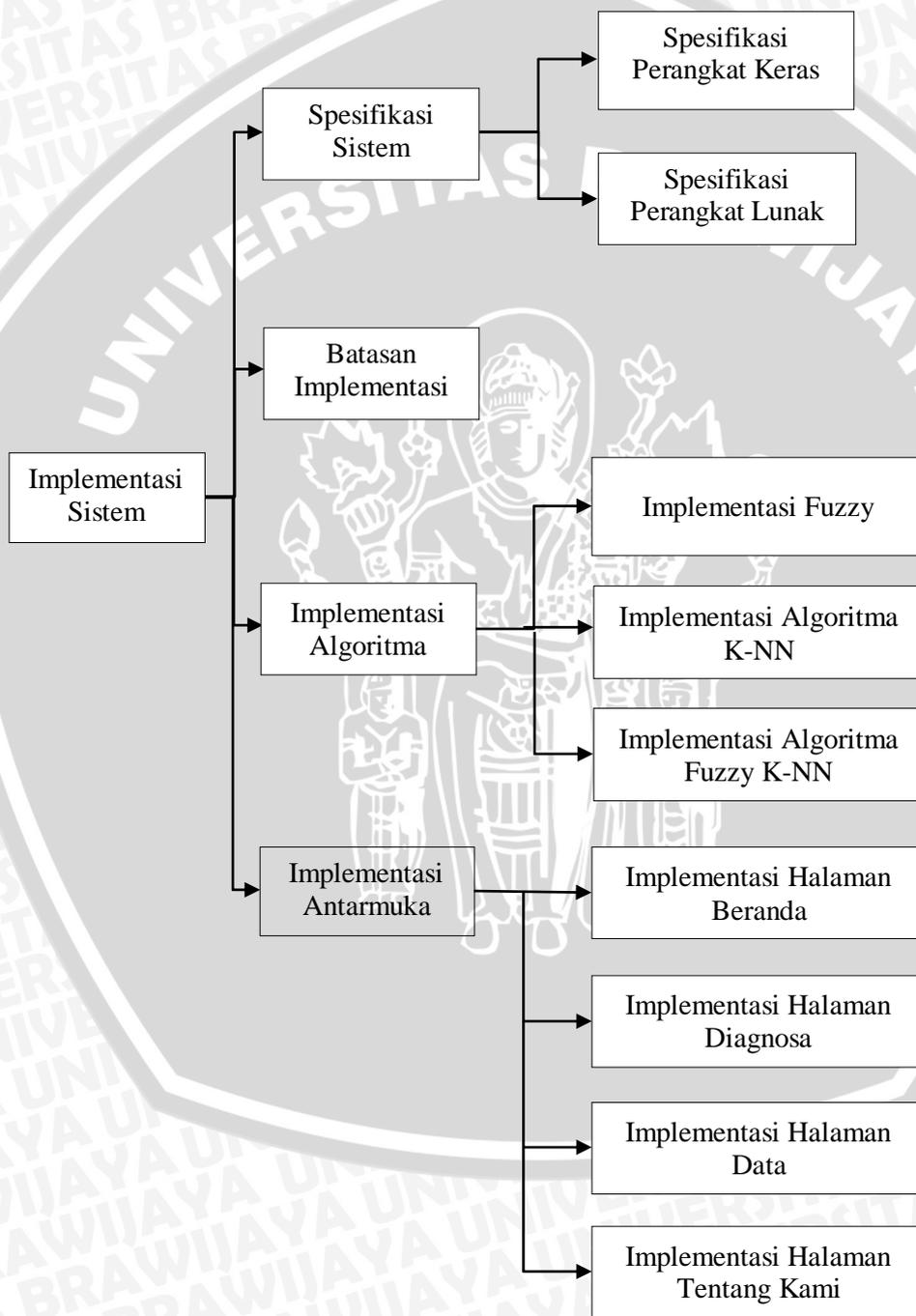
Nilai K	Akurasi (%) 160 data latih dan 40 data uji
3	(%)
5	(%)
7	(%)
Rata-rata	(%)

Tabel 4.19 Contoh pengujian akurasi variasi jumlah data

Nilai K	Akurasi Variasi Jumlah Data (%)				Rata-rata
	160 data latih dan 40 data uji	130 data latih dan 70 data uji	100 data latih dan 100 data uji	70 data latih dan 130 data uji	
k	(%)	(%)	(%)	(%)	(%)

BAB 5 IMPLEMENTASI

Pada bab ini akan membahas tentang implementasi sistem pakar diagnosa penyakit tanaman kedelai berdasarkan pada proses perancangan sistem yang telah dibuat pada tahapan sebelumnya. Pembahasan dalam tahap ini meliputi spesifikasi sistem, batasan implementasi, implementasi algoritma dan implementasi antarmuka. Tahapan implementasi ditunjukkan pada Gambar 5.1.



Gambar 5.1 Tahapan Implementasi

5.1 Spesifikasi Sistem

Hasil dari analisa kebutuhan perangkat lunak yang terdapat pada bab perancangan menjadi acuan dalam pembuatan implementasi sistem. Spesifikasi sistem meliputi spesifikasi perangkat keras dan spesifikasi perangkat lunak. Spesifikasi perangkat keras merupakan alat-alat yang digunakan dalam pembuatan sistem pakar diagnosa penyakit tanaman kedelai, sedangkan spesifikasi perangkat lunak merupakan *tools* atau aplikasi yang mendukung dalam pembuatan sistem pakar ini.

5.1.1 Spesifikasi Perangkat Keras

Spesifikasi perangkat keras dalam pembuatan sistem pakar diagnosa penyakit pada tanaman kedelai terdiri dari komponen-komponen yang ditunjukkan pada Tabel 5.1.

Tabel 5.1 Spesifikasi perangkat keras

Nama Komponen	Spesifikasi
Processor	Intel(R) Core (TM) i3-2367M CPU @ 1.40 GHz
Memori (RAM)	4.00 GB

5.1.2 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak terdiri dari sistem operasi pendukung dalam pembuatan sistem pakar diagnosa penyakit pada tanaman kedelai yang ditunjukkan pada Tabel 5.2.

Tabel 5.2 Spesifikasi perangkat lunak

Sistem Operasi	Microsoft Windows 7 (64-bit)
Bahasa Pemrograman	HTML dan PHP
Tools Pemrograman	Netbeans IDE 7.2
DBMS	XAMPP Control Panel v3.2.1
Browser	Google Chrome dan Mozilla Firefox

5.2 Batasan Implementasi

Batasan implementasi pada pembangunan sistem pakar diagnosa penyakit pada tanaman kedelai menggunakan metode *fuzzy k- nearest neighbor* adalah sebagai berikut:

1. Sistem yang dibangun berbasis web dengan menggunakan bahasa pemrograman PHP.
2. Data-data yang digunakan dalam sistem pakar disimpan dalam database MySQL, namun didalam sistem ini tidak dapat menambah data gejala dengan

alasan yang didasarkan pada pernyataan pakar bahwa hanya terdapat 5 penyakit utama pada kedelai.

3. *Input* yang diterima oleh sistem berupa gejala-gejala penyakit tanaman kedelai yang dimasukkan oleh pengguna umum.
4. Menerapkan metode *Fuzzy K-NN* pada proses perhitungan penentuan hasil diagnosa.
5. *Output* yang diterima oleh pengguna adalah jenis penyakit, solusi dan hasil perhitungan dengan menggunakan metode *fuzzy k-nn*.

5.3 Implementasi Algoritma

Implementasi algoritma pada pemodelan sistem pakar untuk identifikasi penyakit pada tanaman kedelai menggunakan metode *fuzzy k-nearest neighbor* terdiri dari implementasi *fuzzy*, implementasi algoritma K-NN dan implementasi algoritma *fuzzy K-NN*.

5.3.1 Implementasi Fuzzy

Implementasi inisialisasi *fuzzy* pada pemodelan sistem pakar ini terdiri dari proses perhitungan normalisasi data dan proses perhitungan inisialisasi *fuzzy*.

5.3.1.1 Implementasi Proses Perhitungan Normalisasi Data

Implementasi proses perhitungan normalisasi data dibagi menjadi dua yaitu normalisasi data latih dan normalisasi data uji. Perhitungan normalisasi menggunakan persamaan 2-2. Implementasi proses perhitungan normalisasi data latih ditunjukkan pada Source code 5.1 dan implementasi proses perhitungan normalisasi data uji ditunjukkan pada Source code 5.2.

Source code 5.1 Implementasi proses perhitungan normalisasi data latih

```

1. <?php
2. $truncate=mysql_query("TRUNCATE table
   normal_data_latih");
3. $query = 'Select * from data_latih';
4. $hasil = mysql_query($query, $conn);
5. $id = 1;
6. while ($row = mysql_fetch_array($hasil, MYSQL_NUM)) {
7. $G1[$id] = $row[1]; $G2[$id] = $row[2];
8. $G3[$id] = $row[3]; $G4[$id] = $row[4];
9. $G5[$id] = $row[5]; $G6[$id] = $row[6];
10. $G7[$id] = $row[7]; $G8[$id] = $row[8];
11. $G9[$id] = $row[9]; $G10[$id] = $row[10];
12. $G11[$id] = $row[11]; $G12[$id] = $row[12];
13. $G13[$id] = $row[13]; $G14[$id] = $row[14];
14. $G15[$id] = $row[15]; $G16[$id] = $row[16];
15. $Diagnosa[$id] = $row[17]; $id++;
16. }
17. //mencari nilai minimum dan nilai maksimum dari tiap
   gejala
18. $minG1 = min($G1); $minG2 = min($G2);
19. $minG3 = min($G3); $minG4 = min($G4);

```

```

20. $minG5 = min($G5); $minG6 = min($G6);
21. $minG7 = min($G7); $minG8 = min($G8);
22. $minG9 = min($G9); $minG10 = min($G10);
23. $minG11 = min($G11); $minG12 = min($G12);
24. $minG13 = min($G13); $minG14 = min($G14);
25. $minG15 = min($G15); $minG16 = min($G16);
26. $maxG1 = max($G1); $maxG2 = max($G2);
27. $maxG3 = max($G3); $maxG4 = max($G4);
28. $maxG5 = max($G5); $maxG6 = max($G6);
29. $maxG7 = max($G7); $maxG8 = max($G8);
30. $maxG9 = max($G9); $maxG10 = max($G10);
31. $maxG11 = max($G11); $maxG12 = max($G12);
32. $maxG13 = max($G13); $maxG14 = max($G14);
33. $maxG15 = max($G15); $maxG16 = max($G16);
34. //rumus normalisasi data latih
35. for ($i=1;$i<$id;$i++) {
36. $G1normal[$i]=($G1[$i]-$minG1)/($maxG1-$minG1);
37. $G2normal[$i]=($G2[$i]-$minG2)/($maxG2-$minG2);
38. $G3normal[$i]=($G3[$i]-$minG3)/($maxG3-$minG3);
39. $G4normal[$i]=($G4[$i]-$minG4)/($maxG4-$minG4);
40. $G5normal[$i]=($G5[$i]-$minG5)/($maxG5-$minG5);
41. $G6normal[$i]=($G6[$i]-$minG6)/($maxG6-$minG6);
42. $G7normal[$i]=($G7[$i]-$minG7)/($maxG7-$minG7);
43. $G8normal[$i]=($G8[$i]-$minG8)/($maxG8-$minG8);
44. $G9normal[$i]=($G9[$i]-$minG9)/($maxG9-$minG9);
45. $G10normal[$i]=($G10[$i]-$minG10)/($maxG10-$minG10);
46. $G11normal[$i]=($G11[$i]-$minG11)/($maxG11-$minG11);
47. $G12normal[$i]=($G12[$i]-$minG12)/($maxG12-$minG12);
48. $G13normal[$i]=($G13[$i]-$minG13)/($maxG13-$minG13);
49. $G14normal[$i]=($G14[$i]-$minG14)/($maxG14-$minG14);
50. $G15normal[$i]=($G15[$i]-$minG15)/($maxG15-$minG15);
51. $G16normal[$i]=($G16[$i]-$minG16)/($maxG16-$minG16);
52. mysql_query("insert into normal_data_latih (G1, G2, G3,
    G4,G5,G6, G7, G8, G9, G10,G11,G12, G13, G14, G15, G16,
    Diagnosa) values ($G1normal[$i], $G2normal[$i],
    $G3normal[$i], $G4normal[$i], $G5normal[$i],
    $G6normal[$i], $G7normal[$i], $G8normal[$i],
    $G9normal[$i], $G10normal[$i], $G11normal[$i],
    $G12normal[$i], $G13normal[$i], $G14normal[$i],
    $G15normal[$i], $G16normal[$i], $Diagnosa[$i])");
53. }
54. ?>

```

Penjelasan Source code 5.1 adalah sebagai berikut:

- Baris 3 sampai 16 menampilkan isi dari tabel data_latih yang telah tersimpan dalam database.
- Baris 18 sampai 33 mencari nilai minimum dan nilai maksimum dari tiap data gejala
- Baris 36 sampai 51 proses perhitungan rumus normalisasi data latih
- Baris 52 memasukkan hasil data latih yang telah dinormalisasikan ke dalam tabel normal_data_latih.

Source code 5.2 Implementasi proses perhitungan normalisasi data uji

```

1.  <?php
2.  $truncate=mysql_query("TRUNCATE table
    normal_data_uji");
3.  $query = 'Select * from data_uji';
4.  $hasil = mysql_query($query, $conn);
5.  $id = 1;
6.  while ($row = mysql_fetch_array($hasil, MYSQL_NUM)) {
7.    $G1[$id] = $row[1];
8.    $G2[$id] = $row[2];
9.    $G3[$id] = $row[3]; $G4[$id] = $row[4];
10.   $G5[$id] = $row[5]; $G6[$id] = $row[6];
11.   $G7[$id] = $row[7]; $G8[$id] = $row[8];
12.   $G9[$id] = $row[9]; $G10[$id] = $row[10];
13.   $G11[$id] = $row[11]; $G12[$id] = $row[12];
14.   $G13[$id] = $row[13]; $G14[$id] = $row[14];
15.   $G15[$id] = $row[15]; $G16[$id] = $row[16];
16.   $Diagnosa[$id] = $row[17]; $id++;
17.  }
18.  //mencari nilai minimum dan nilai maksimum dari tiap
    gejala
19.  $minG1 = min($G1); $minG2 = min($G2);
20.  $minG3 = min($G3); $minG4 = min($G4);
21.  $minG5 = min($G5); $minG6 = min($G6);
22.  $minG7 = min($G7); $minG8 = min($G8);
23.  $minG9 = min($G9); $minG10 = min($G10);
24.  $minG11 = min($G11); $minG12 = min($G12);
25.  $minG13 = min($G13); $minG14 = min($G14);
26.  $minG15 = min($G15); $minG16 = min($G16);
27.  $maxG1 = max($G1); $maxG2 = max($G2);
28.  $maxG3 = max($G3); $maxG4 = max($G4);
29.  $maxG5 = max($G5); $maxG6 = max($G6);
30.  $maxG7 = max($G7); $maxG8 = max($G8);
31.  $maxG9 = max($G9); $maxG10 = max($G10);
32.  $maxG11 = max($G11); $maxG12 = max($G12);
33.  $maxG13 = max($G13); $maxG14 = max($G14);
34.  $maxG15 = max($G15); $maxG16 = max($G16);
35.  //rumus normalisasi data uji
36.  for ($i=1;$i<$id;$i++) {
37.    $G1normal[$i]=($G1[$i]-$minG1)/($maxG1-$minG1);
38.    $G2normal[$i]=($G2[$i]-$minG2)/($maxG2-$minG2);
39.    $G3normal[$i]=($G3[$i]-$minG3)/($maxG3-$minG3);
40.    $G4normal[$i]=($G4[$i]-$minG4)/($maxG4-$minG4);
41.    $G5normal[$i]=($G5[$i]-$minG5)/($maxG5-$minG5);
42.    $G6normal[$i]=($G6[$i]-$minG6)/($maxG6-$minG6);
43.    $G7normal[$i]=($G7[$i]-$minG7)/($maxG7-$minG7);
44.    $G8normal[$i]=($G8[$i]-$minG8)/($maxG8-$minG8);
45.    $G9normal[$i]=($G9[$i]-$minG9)/($maxG9-$minG9);
46.    $G10normal[$i]=($G10[$i]-$minG10)/($maxG10-$minG10);
47.    $G11normal[$i]=($G11[$i]-$minG11)/($maxG11-$minG11);
48.    $G12normal[$i]=($G12[$i]-$minG12)/($maxG12-$minG12);
49.    $G13normal[$i]=($G13[$i]-$minG13)/($maxG13-$minG13);
50.    $G14normal[$i]=($G14[$i]-$minG14)/($maxG14-$minG14);
51.    $G15normal[$i]=($G15[$i]-$minG15)/($maxG15-$minG15);
52.    $G16normal[$i]=($G16[$i]-$minG16)/($maxG16-$minG16);
53.    mysql_query("insert into normal_data_uji (G1, G2, G3,
        G4,G5,G6, G7, G8, G9, G10,G11,G12, G13, G14, G15, G16,
        Diagnosa) values ($G1normal[$i], $G2normal[$i],

```

```

    $G3normal[$i], $G4normal[$i], $G5normal[$i],
    $G6normal[$i], $G7normal[$i], $G8normal[$i],
    $G9normal[$i], $G10normal[$i], $G11normal[$i],
    $G12normal[$i], $G13normal[$i], $G14normal[$i],
    $G15normal[$i], $G16normal[$i], $Diagnosa[$i]);
54. }
55. ?>

```

Penjelasan Source code 5.2 adalah sebagai berikut:

- Baris 3 sampai 16 menampilkan isi dari tabel data_uji yang telah tersimpan dalam database.
- Baris 19 sampai 34 mencari nilai minimum dan nilai maksimum dari tiap data gejala.
- Baris 37 sampai 52 proses perhitungan rumus normalisasi data uji.
- Baris 53 memasukkan hasil data uji yang telah dinormalisasikan ke dalam tabel normal_data_uji.

5.3.1.2 Implementasi Proses Perhitungan Inisialisasi Fuzzy

Perhitungan inisialisasi fuzzy dilakukan untuk mengetahui nilai keanggotaan data latih pada setiap penyakit. Implementasi proses perhitungan inisialisasi fuzzy ditunjukkan pada Source code 5.3.

Source code 5.3 Implementasi proses perhitungan inisialisasi fuzzy

```

1. <?php
2. $truncate = mysql_query("TRUNCATE table data_fuzzy");
3. $query = 'Select * from normal_data_latih';
4. $hasil = mysql_query($query, $conn);
5. $id = 1;
6. //mengambil nilai dari tabel normal_data_latih
7. while ($row = mysql_fetch_array($hasil, MYSQL_NUM)) {
8. $G1[$id] = $row[1];
9. .
10. .
11. .
12. $G16[$id] = $row[16];
13. $Diagnosa[$id] = $row[17];
14. $id++;}
15. for($i=1;$i<161;$i++){
16. //rumus perhitungan jarak antar data latih
17. for($j=1;$j<81;$j++){
18. $jarak[$i][$j] = sqrt(pow($G1[$i] - $G1[$j], 2) +
    pow($G2[$i] - $G2[$j], 2) + pow($G3[$i] - $G3[$j], 2) +
    pow($G4[$i] - $G4[$j], 2) + pow($G5[$i] - $G5[$j], 2) +
    pow($G6[$i] - $G6[$j], 2) + pow($G7[$i] - $G7[$j], 2) +
    pow($G8[$i] - $G8[$j], 2) + pow($G9[$i] - $G9[$j], 2) +
    pow($G10[$i] - $G10[$j], 2) + pow($G11[$i] - $G11[$j], 2) +
    pow($G12[$i] - $G12[$j], 2) + pow($G13[$i] - $G13[$j], 2) +
    pow($G14[$i] - $G14[$j], 2) + pow($G15[$i] - $G15[$j], 2) +
    pow($G16[$i] - $G16[$j], 2));}
19. for ($i = 1; $i < 81; $i++) {
20. //inisialisasi hasil jarak
21. $d1 = $jarak[$i][1];

```

```
22. .
23. .
24. .
25. $d160 = $jarak[$i][160];
26. mysql_query ("INSERT INTO data_fuzzy(D1, ..., D160) values
($d1, ..., $d160)");}?>
27. <!-- proses pengurutan data fuzzy dari yang terkecil
berdasarkan id data-->
28. <?php
29. $query = 'Select * from data_fuzzy';
30. $hasil = mysql_query($query, $conn);
31. $id = 0;
32. while ($row = mysql_fetch_array($hasil, MYSQL_NUM)) {
33. $data0[$id] = $row[0];
34. .
35. .
36. .
37. $data80[$id] = $row[160]; $id++;
38. }
39. //pengurutan berdasarkan id
40. $data1 = insertionSort($data1);
41. .
42. .
43. .
44. $data80 = insertionSort($data80);
45. for ($i = 1; $i < 161; $i++) {
46. $d[$i] = 0;
47. }
48. for ($i = 0; $i < (count($data1) - 1); $i++) {
49. //for ($i = 0; $i < 160; $i++){
50. $data[1] = $data0[$data1[$i]];
51. .
52. .
53. .
54. $data[160] = $data0[$data160[$i]];
55. mysql_query("INSERT INTO data_fuzzy_id(D1, ..., D160)
values $data[1], ..., $data[160])");
56. }
57. ?>
58. <!--pengurutan data fuzzy dari yang terkecil berdasarkan
penyakit-->
59. <?php
60. //$truncate = mysql_query("TRUNCATE table
data_fuzzy_penyakit");
61. $query = 'Select * from data_fuzzy_id';
62. $hasil = mysql_query($query, $conn);
63. $id = 0;
64. while ($row = mysql_fetch_array($hasil, MYSQL_NUM)) {
65. $datap0[$id] = $row[0];
66. $datap1[$id] = $row[1];
67. .
68. .
69. .
70. $datap80[$id] = $row[160];
71. $id++;
72. }
73. for ($i = 0; $i < 160; $i++) {
74. //menambil nilai diagnosa penyakit pada masing masing data
```

```
75. $query1 = "Select diagnosa from normal_data_latih where
ID='$datap1[$i]';
76. $hasil1 = mysql_query($query1, $conn);
77. $p1 = mysql_fetch_array($hasil1);
78. $pykt1 = $p1["diagnosa"];
79. .
80. .
81. .
82. $query160 = "Select diagnosa from normal_data_latih where
ID='$datap160[$i]';
83. $hasil160 = mysql_query($query160, $conn);
84. $p160 = mysql_fetch_array($hasil160);
85. $pykt160 = $p160["diagnosa"];
86. mysql_query("INSERT INTO data_fuzzy_penyakit(D1, ..., D160)
values ($pykt1, ..., $pykt160)");
87. }
88. ?>
89. <!--ambil k data fuzzy penyakit-->
90. <?php
91. $truncate = mysql_query("TRUNCATE table data_fuzzy_id_k");
92. $query = 'Select * from data_fuzzy_penyakit';
93. $hasil = mysql_query($query, $conn);
94. $id = 0;
95. while ($row = mysql_fetch_array($hasil, MYSQL_NUM)) {
96. $datapk0[$id] = $row[0];
97. .
98. .
99. .
100. $datapk160[$id] = $row[160];
101. $id++;
102. }
103. for ($i = 1; $i <= $k_inisialisasi; $i++) {
104. mysql_query("INSERT INTO data_fuzzy_id_k(D1, ..., D160)
values ($datapk1[$i], ..., $datapk80[$i])");
105. }
106. ?>
107. <!--data inisialisasi fuzzy berdasarkan k-->
108. <?php
109. $truncate = mysql_query("TRUNCATE table
inisialisasi_fuzzy");
110. $query1 = 'Select diagnosa from normal_data_latih';
111. $query2 = 'Select * from data_fuzzy_id_k';
112. $hasil1 = mysql_query($query1, $conn);
113. $hasil2 = mysql_query($query2, $conn);
114. $id = 1;
115. while ($row = mysql_fetch_array($hasil1, MYSQL_NUM)) {
116. //memanggil diagnosa dari tabel normal data latih
117. $diagnosa[$id] = $row[0];
118. $id++;
119. }
120. while ($row = mysql_fetch_array($hasil2, MYSQL_NUM)) {
121. $datafuzzyk0[$id] = $row[0];
122. $datafuzzyk1[$id] = $row[1];
123. .
124. .
125. .
126. $datafuzzyk160[$id] = $row[160];
127. $id++;
128. }
```

```

129. for ($i = 1; $i <= 5; $i++) { //menghitung jumlah id yang
    sesuai dengan penyakit
130. $query = "SELECT COUNT(D1) as num FROM data_fuzzy_id_k
    WHERE D1=$i";
131. $result = mysql_query($query);
132. $result = mysql_fetch_assoc($result);
133. $datafuzzyk[1][$i] = $result['num'];
134. .
135. .
136. .
137. $query = "SELECT COUNT(D80) as num FROM data_fuzzy_id_k
    WHERE D80=$i";
138. $result = mysql_query($query);
139. $result = mysql_fetch_assoc($result);
140. $datafuzzyk[160][$i] = $result['num'];
141. }
142. for ($j = 1; $j <= 5; $j++) { // perulangan data
143. for ($i = 1; $i <= 160; $i++) { // perulangan penyakit
144. if ($diagnosa[$i] == $j) {
145. //perhitungan jika penyakit pada data sesuai dengan
    penyakit yang diujikan
146. $inisial[$i][$j] = (0.51 + ($datafuzzyk[$i][$j] /
    $k_inisialisasi) * 0.49);
147. } else {
148. //rumus jika penyakit tidak sama
149. $inisial[$i][$j] = ((($datafuzzyk[$i][$j]) /
    $k_inisialisasi) * 0.49);
150. }
151. }
152. }
153. for ($j = 1; $j <= 5; $j++) {
154. $datafuzzyk1[$j] = $inisial[1][$j];
155. .
156. .
157. .
158. $datafuzzyk160[$j] = $inisial[160][$j];
159. mysql_query("INSERT INTO inisialisasi_fuzzy (D1, ..., D160)
    values ($datafuzzyk1[$j], ..., $datafuzzyk160[$j])");
160. }
161. ?>

```

Penjelasan Source code 5.3 adalah sebagai berikut:

- Baris 8 sampai 13 mengambil nilai dari tabel normal_data_latih.
- Baris 18 rumus perhitungan jarak antar data latih.
- Baris 21 sampai 25 inisialisasi hasil jarak.
- Baris 26 memasukkan hasil jarak ke dalam tabel data_fuzzy.
- Baris 33 sampai 37 mengambil nilai dari tabel data_fuzzy.
- Baris 40 sampai 54 proses pengurutan data dari yang terkecil ke yang terbesar berdasarkan id data.
- Baris 55 memasukkan hasil pengurutan data ke dalam tabel data_fuzzy_id.
- Baris 65 sampai 70 mengambil nilai dari tabel data_fuzzy.

- Baris 74 sampai 85 mengambil nilai diagnosa penyakit pada tiap data di tabel normal_data_latih.
- Baris 86 memasukkan hasil pengurutan data berdasarkan penyakit ke dalam tabel data_fuzzy_penyakit.
- Baris 96 sampai 100 mengambil nilai dari tabel data_fuzzy_id_k.
- Baris 103 sampai 104 mengambil k data fuzzy penyakit kemudian memasukkan ke dalam tabel data_fuzzy_id_k.
- Baris 117 mengambil diagnosa dari tabel normal_data latih.
- Baris 121 sampai 126 mengambil nilai dari tabel data_fuzzy_id_k.
- Baris 130 sampai 140 menghitung jumlah id yang sesuai dengan penyakit.
- Baris 145 proses perhitungan inialisasi fuzzy jika penyakit pada data sesuai dengan penyakit yang diujikan.
- Baris 149 proses perhitungan inialisasi fuzzy jika penyakit pada data tidak sesuai dengan penyakit yang diujikan.
- Baris 154 sampai 158 hasil inialisasi fuzzy disimpan ke dalam variabel-variabel.
- Baris 159 memasukkan hasil inialisasi fuzzy ke dalam tabel inialisasi_fuzzy.

5.3.2 Implementasi Algoritma KNN

Implementasi algoritma KNN pada pemodelan sistem pakar merupakan proses perhitungan jarak data uji terhadap tiap data latih kemudian jarak diurutkan dari yang terkecil dan diambil sebanyak k data. Implementasi proses algoritma KNN ditunjukkan pada Source code 5.4.

Source code 5.4 Implementasi proses algoritma KNN

```

1.  <!--proses perhitungan jarak data uji terhadap data latih-->
2.  <?php
3.  $truncate = mysql_query("TRUNCATE table jarak");
4.  $query1 = 'Select * from normal_data_latih';
5.  $query2 = 'Select * from normal_data_uji';
6.  $hasil1 = mysql_query($query1, $conn);
7.  $hasil2 = mysql_query($query2, $conn);
8.  $sid = 1;
9.  while ($row = mysql_fetch_array($hasil1, MYSQL_NUM)) {
10. $G1latih[$sid] = $row[1];
11. $G2latih[$sid] = $row[2];
12. $G3latih[$sid] = $row[3];
13. $G4latih[$sid] = $row[4];
14. $G5latih[$sid] = $row[5];
15. $G6latih[$sid] = $row[6];
16. $G7latih[$sid] = $row[7];
17. $G8latih[$sid] = $row[8];
18. $G9latih[$sid] = $row[9];
19. $G10latih[$sid] = $row[10];

```

```
20. $G11latih[$id] = $row[11];
21. $G12latih[$id] = $row[12];
22. $G13latih[$id] = $row[13];
23. $G14latih[$id] = $row[14];
24. $G15latih[$id] = $row[15];
25. $G16latih[$id] = $row[16];
26. $Diagnosalatih[$id] = $row[17];
27. $id++;
28. }
29. $id = 1;
30. while ($row = mysql_fetch_array($hasil2, MYSQL_NUM)) {
31. $G1uji[$id] = $row[1];
32. $G2uji[$id] = $row[2];
33. $G3uji[$id] = $row[3];
34. $G4uji[$id] = $row[4];
35. $G5uji[$id] = $row[5];
36. $G6uji[$id] = $row[6];
37. $G7uji[$id] = $row[7];
38. $G8uji[$id] = $row[8];
39. $G9uji[$id] = $row[9];
40. $G10uji[$id] = $row[10];
41. $G11uji[$id] = $row[11];
42. $G12uji[$id] = $row[12];
43. $G13uji[$id] = $row[13];
44. $G14uji[$id] = $row[14];
45. $G15uji[$id] = $row[15];
46. $G16uji[$id] = $row[16];
47. $Diagnosauji[$id] = $row[17];
48. $id++;
49. }
50. for ($i = 1; $i < 161; $i++) {
51. $jarak[$i] = sqrt(pow($G1uji[1] - $G1latih[$i], 2) +
    pow($G2uji[1] - $G2latih[$i], 2) + pow($G3uji[1] -
    $G3latih[$i], 2) + pow($G4uji[1] - $G4latih[$i], 2) +
    pow($G5uji[1] - $G5latih[$i], 2) + pow($G6uji[1] -
    $G6latih[$i], 2) + pow($G7uji[1] - $G7latih[$i], 2) +
    pow($G8uji[1] - $G8latih[$i], 2) + pow($G9uji[1] -
    $G9latih[$i], 2) + pow($G10uji[1] - $G10latih[$i], 2) +
    pow($G11uji[1] - $G11latih[$i], 2) + pow($G12uji[1] -
    $G12latih[$i], 2) + pow($G13uji[1] - $G13latih[$i], 2) +
    pow($G14uji[1] - $G14latih[$i], 2) + pow($G15uji[1] -
    $G15latih[$i], 2) + pow($G16uji[1] - $G16latih[$i], 2));
52. if ($i < 33) {
53. } else if ($i > 32 && $i < 79) {
54. } else if ($i > 78 && $i < 109) {
55. } else if ($i > 108 && $i < 132) {
56. } else {
57. }
58. mysql_query("INSERT INTO jarak (hasil_jarak,
    diagnosa) value ($jarak[$i], $diagnosa[$i])");
59. }
60. ?>
61. <!--pengurutan jarak uji terhadap latihan-->
62. <?php
```

```

63. $truncate = mysql_query("TRUNCATE table jarak_urut");
64. $query = 'Select * from jarak';
65. $hasil = mysql_query($query, $conn);
66. $id = 0;
67. while ($row = mysql_fetch_array($hasil, MYSQL_NUM)) {
68. $id_jarak[$id] = $row[0]; // memanggil id_jarak
69. $hasil_jarak[$id] = $row[1]; // memanggil hasil jarak
70. $target[$id] = $row[2]; // memanggil target jarak
71. $id++;
72. }
73. $id_j = insertionSort($hasil_jarak);
74. for ($i = 0; $i < 160; $i++) {
75. $id1 [$i] = $id_jarak[$id_j[$i]];
76. $jarak [$i] = $hasil_jarak[$id_j[$i]];
77. $diagnosa [$i] = $target[$id_j[$i]];
78. mysql_query("INSERT INTO jarak_urut (id_jarak, hasil_jarak,
diagnosa) values ($id1[$i], $jarak[$i], $diagnosa[$i])");
79. }
80. ?>
81. <!--ambil k jarak-->
82. <?php
83. $truncate = mysql_query("TRUNCATE table jarak_k");
84. $query = 'Select * from jarak_urut';
85. $hasil = mysql_query($query, $conn);
86. $id = 0;
87. while ($row = mysql_fetch_array($hasil, MYSQL_NUM)) {
88. $datajarak0[$id] = $row[0];
89. $datajarak1[$id] = $row[1];
90. $datajarak2[$id] = $row[2];
91. $datajarak3[$id] = $row[3];
92. $id++;
93. }
94. for ($i = 0; $i < $k_klasifikasi; $i++) {
95. $id_jarak = $datajarak1[$i];
96. $hasil_jarak = $datajarak2[$i];
97. $diagnosa = $datajarak3[$i];
98. mysql_query("INSERT INTO jarak_k (id_jarak, hasil_jarak,
diagnosa) values ($id_jarak, $hasil_jarak, $diagnosa)");
99. }
100. ?>

```

Penjelasan Source code 5.4 adalah sebagai berikut:

- Baris 10 sampai 26 mengambil isi tabel normal_data_latih.
- Baris 31 sampai 47 mengambil isi tabel normal_data_uji.
- Baris 51 rumus perhitungan jarak data uji terhadap tiap data latih.
- Baris 58 memasukkan hasil jarak ke dalam tabel jarak.
- Baris 62 sampai 80 mengurutkan jarak dari yang terkecil.
- Baris 82 sampai 97 mengambil nilai k data jarak.
- Baris 98 memasukkan hasil jarak sebanyak k kedalam tabel jarak_k.

5.3.3 Implementasi Algoritma Fuzzy KNN

Setelah data diambil sebanyak K data dari proses perhitungan KNN, maka langkah selanjutnya adalah proses fuzzy KNN untuk menghitung nilai keanggotaan tiap kelas penyakit. Proses implementasi algoritma fuzzy KNN ditunjukkan pada Source code 5.5.

Source code 5.5 Proses implementasi algoritma fuzzy knn

```

1.  <?php
2.  $truncate = mysql_query("TRUNCATE table keanggotaan");
3.  $query1 = 'Select * from jarak_k';
4.  $query2 = 'Select * from inisialisasi_fuzzy';
5.  $hasil1 = mysql_query($query1, $conn);
6.  $hasil2 = mysql_query($query2, $conn);
7.  $id = 1;
8.  //mengambil isi tabel jarak_k
9.  while ($row = mysql_fetch_array($hasil1, MYSQL_NUM)) {
10. $id_jarak_k[$id] = $row[1];
11. $hasil_jarak_k[$id] = $row[2];
12. $id++;
13. }
14. $id = 1;
15. //mengambil isi tabel inisialisasi fuzzy
16. while ($row = mysql_fetch_array($hasil2, MYSQL_NUM)) {
17. $dataf[1][$id] = $row[1];
18. .
19. .
20. .
21. $dataf[160][$id] = $row[160];
22. $id++;
23. }
24. // perhitungan keanggotaan
25. for ($i = 1; $i <= 5; $i++) {
26. $jumlah_keanggotaan1[$i] = 0;
27. $jumlah_keanggotaan2[$i] = 0;
28. $id_p[$i] = $i;
29. for ($j = 1; $j <= 3; $j++) {
30. $keanggotaan1 = ($dataf[$id_jarak_k[$j]][$i] * (pow((1 /
    $hasil_jarak_k[$j]), 2 / ($m - 1))));
31. $jumlah_keanggotaan1[$i] = $jumlah_keanggotaan1[$i] +
    $keanggotaan1;
32. $keanggotaan2 = (pow((1 / $hasil_jarak_k[$j]), 2 / ($m -
    1)));
33. $jumlah_keanggotaan2[$i] = $jumlah_keanggotaan2[$i] +
    $keanggotaan2;
34. }
35. $keanggotaan[$i] = $jumlah_keanggotaan1[$i] /
    $jumlah_keanggotaan2[$i];
36. //echo $keanggotaan[$i];
37. mysql_query("insert into keanggotaan (Penyakit,
    Nilai_keanggotaan) values ($id_p[$i], $keanggotaan[$i])");
38. }
39. }>
40. <!--mengurutkan dari nilai keanggotaan terbesar sebagai
    kelas target-->
41. <?php
42. $truncate = mysql_query("TRUNCATE table keanggotaanurut");

```

```

43. $urut_keanggotaan = insertionSortMax($keanggotaan);
44. for ($urutmax = 1; $urutmax <= 5; $urutmax++) {
45. $id_p_urut[$urutmax] = $id_p [$urut_keanggotaan[$urutmax]];
46. $keanggotaan_urut[$urutmax] =
    $keanggotaan[$urut_keanggotaan[$urutmax]];
47. mysql_query("insert into keanggotaan_urut (Penyakit,
    Nilai_keanggotaan) values ($id_p_urut[$urutmax],
    $keanggotaan_urut[$urutmax])");
48. }
49. ?>

```

Penjelasan Source code 5.5 adalah sebagai berikut:

- Baris 9 sampai 12 mengambil isi tabel jarak_k.
- Baris 17 sampai 21 mengambil isi tabel inisialisasi_fuzzy.
- Baris 25 sampai 39 proses perhitungan keanggotaan dan memasukkan hasil ke dalam tabel keanggotaan.
- Baris 41 sampai 49 pengurutan nilai keanggotan terbesar dan memasukkan ke dalam tabel keanggotaan_urut.

5.4 Implementasi Antarmuka

Implementasi antarmuka pemodelan sistem pakar diagnosa penyakit tanaman kedelai ini meliputi antarmuka halaman beranda, antarmuka halaman diagnosa dan antarmuka halaman data.

5.4.1 Implementasi Antarmuka Halaman Beranda

Halaman beranda merupakan halaman awal yang akan diakses oleh pengguna, pada halaman terdapat *slide show* mengenai produk-produk olahan kedelai dan informasi mengenai penyakit-penyakit pada tanaman kedelai. Implementasi antarmuka halaman beranda ditunjukkan pada Gambar 5.2.

5.4.2 Implementasi Antarmuka Halaman Diagnosa

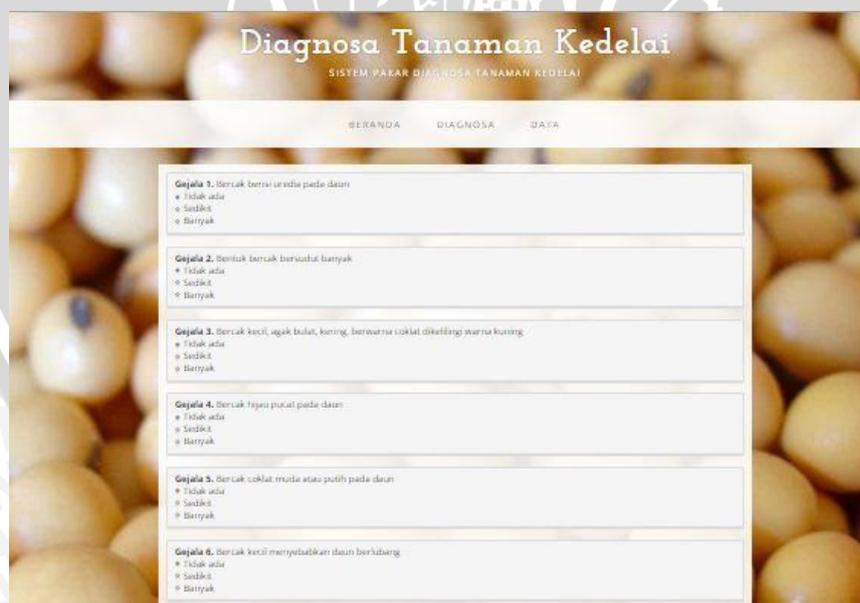
Halaman daignosa berisi kuisoner yang diajukan oleh sistem kepada pengguna berupa gejala-gejala penyakit tanaman kedelai dengan opsi jawaban berupa tidak ada, sedikit dan banyak. Tombol Lihat Diagnosa akan mengarahkan pada halaman hasil diagnosa dan pengguna dapat melihat hasil diagnosa sesuai dengan inputan gejala sebelumnya. Pada halaman hasil diagnosa, sistem akan menampilkan hasil diagnosa dan proses perhitungan sesuai dengan inputan gejala kuisoner oleh pengguna. Implementasi antarmuka halaman diagnosa ditunjukkan pada Gambar 5.3 dan antarmuka halaman hasil diagnosa ditunjukkan pada Gambar 5.4.

5.4.3 Implementasi Antarmuka Halaman Data

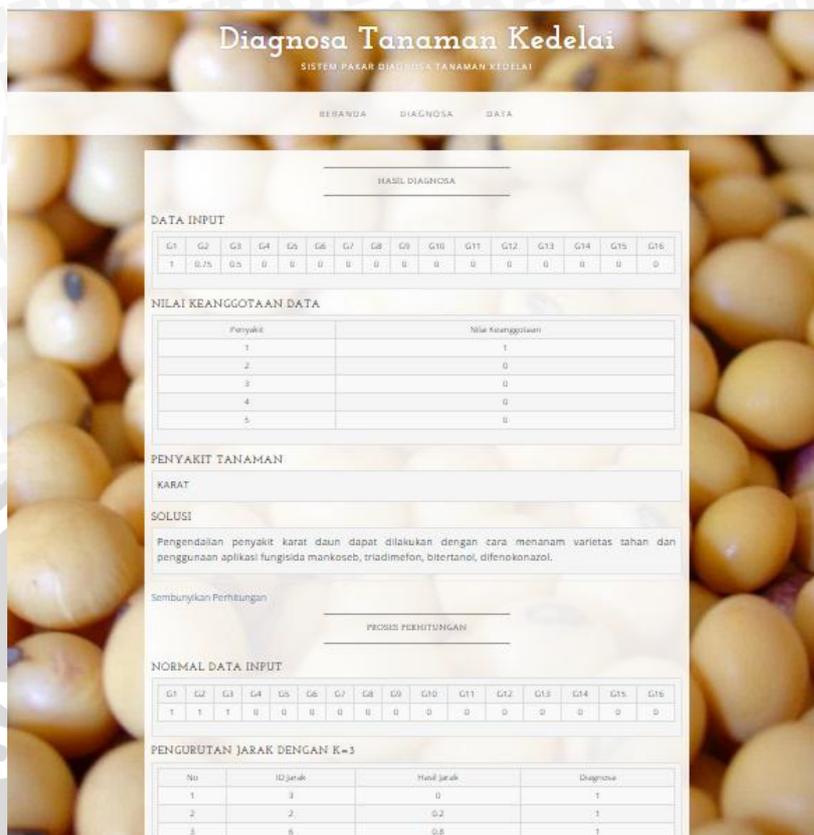
Halaman data menampilkan data latih dan data uji yang digunakan dalam sistem. Implementasi antarmuka halaman data ditunjukkan pada Gambar 5.5.



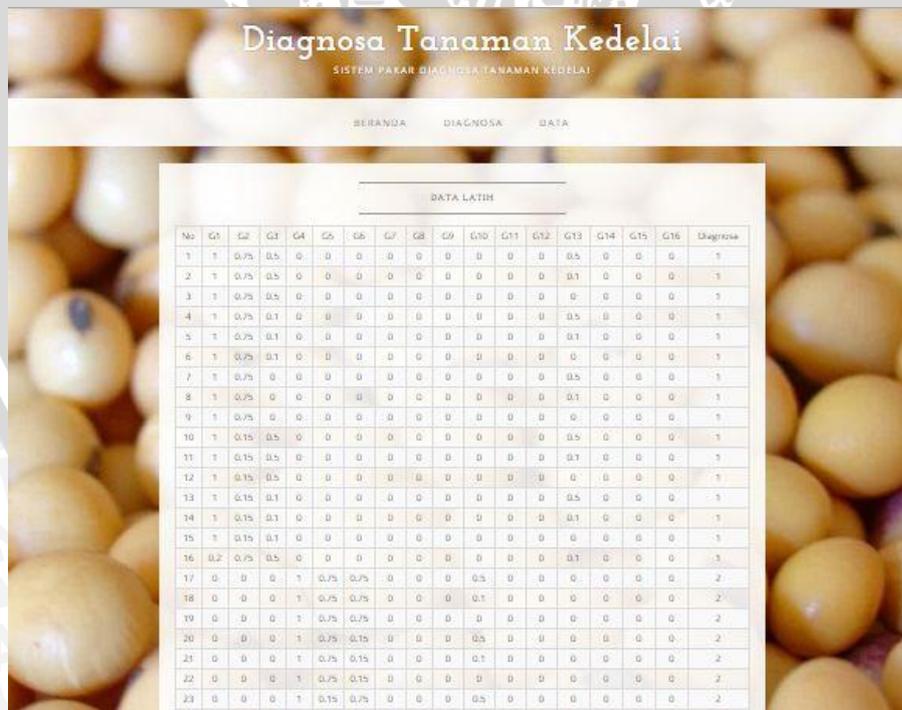
Gambar 5.2 Implementasi antarmuka halaman beranda



Gambar 5.3 Implementasi antarmuka halaman diagnosa



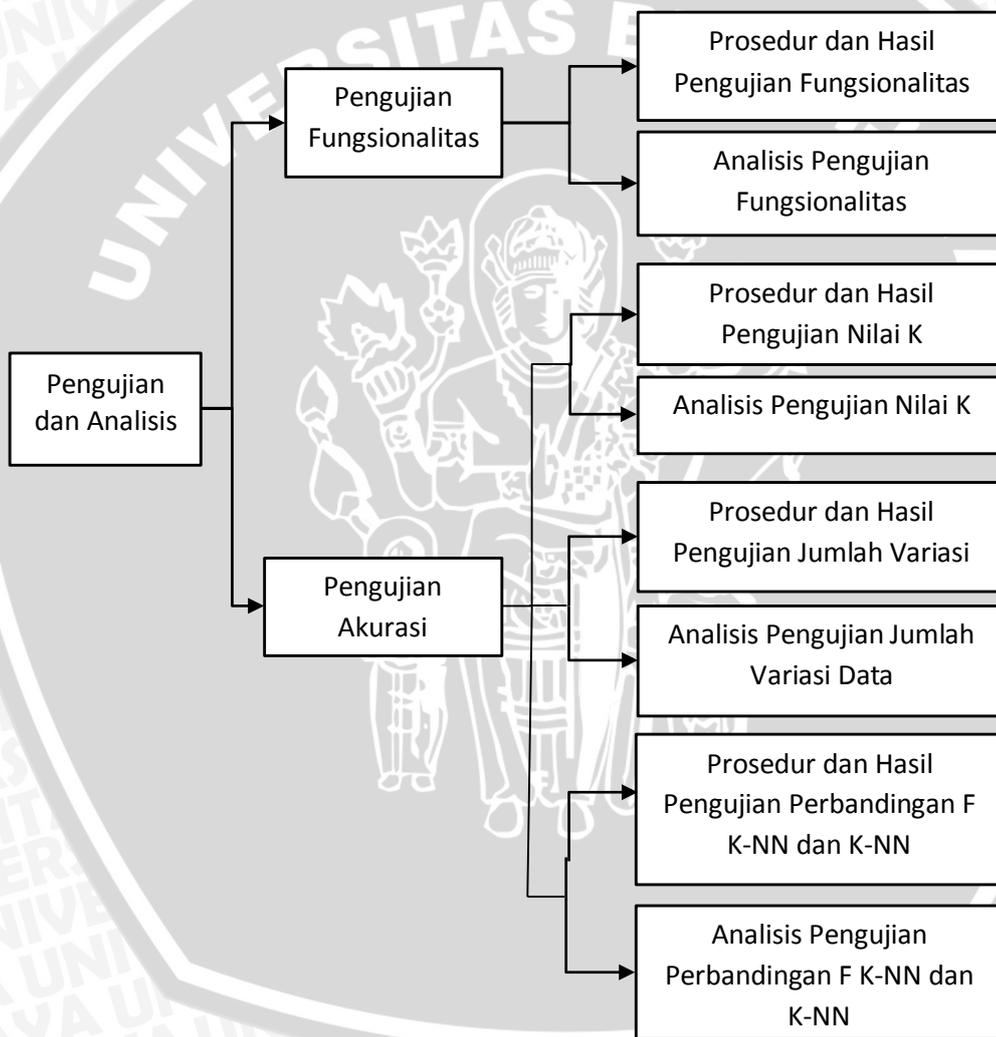
Gambar 5.4 Implementasi antarmuka halaman hasil diagnosa



Gambar 5.5 Implementasi antarmuka halaman data

BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini akan membahas mengenai prosedur dan hasil pengujian Pemodelan Sistem Pakar Diagnosa Penyakit Tanaman Kedelai dengan Metode Fuzzy K-Nearest Neighbor. Proses pengujian dilakukan melalui dua tahapan, yakni pengujian fungsionalitas dan pengujian akurasi. Pengujian fungsionalitas dilakukan untuk mengetahui apakah sistem yang dirancang telah sesuai dengan kebutuhan sistem yang diharapkan. Pengujian akurasi terdapat tiga tahapan yaitu pengujian nilai k, pengujian jumlah variasi data dengan menggunakan nilai k tertinggi pada pengujian nilai k dan pengujian perbandingan metode F K-NN dan K-NN. Pohon pengujian dan analisis ditunjukkan pada Gambar 6.1.



Gambar 6.1 Pohon pengujian dan analisis

6.1 Pengujian Fungsionalitas

Pengujian fungsionalitas adalah pengujian yang dilakukan terhadap sistem dengan tujuan mengetahui apakah sistem yang dirancang telah memenuhi daftar kebutuhan sistem yang diharapkan.

6.1.1 Prosedur dan Hasil Pengujian Fungsionalitas

Pengujian fungsionalitas dilakukan dengan membuat kasus uji untuk setiap daftar kebutuhan sistem yang telah dirancang pada Tabel 4.16. Hasil pengujian fungsionalitas ditunjukkan pada Tabel 6.1.

Tabel 6.1 Hasil pengujian fungsionalitas

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil yang didapat	Status Validasi
1.	Mengklik menu beranda	Sistem menampilkan halaman beranda	Sistem berhasil menampilkan halaman utama dengan benar	valid
2.	Mengklik menu diagnosa	Sistem menampilkan halaman diagnosa	Sistem berhasil menampilkan halaman diagnosa dengan benar	valid
3.	Mengklik tombol diagnosa	Sistem menampilkan halaman hasil diagnosa	Sistem berhasil menampilkan halaman hasil diagnosa dengan benar	valid
4.	Mengklik menu data	Sistem menampilkan halaman data	Sistem berhasil menampilkan halaman data dengan benar	valid

6.1.2 Analisis Pengujian Fungsionalitas

Analisa hasil pengujian fungsionalitas dilakukan dengan membandingkan kesesuaian antara hasil yang diharapkan dengan hasil yang didapat. Hasil pengujian yang ditunjukkan pada Tabel 6.1 memiliki tingkat kesesuaian 100%, sehingga dapat disimpulkan bahwa fungsionalitas dari Pemodelan Sistem Pakar Untuk Identifikasi Penyakit Pada Tanaman Kedelai Menggunakan Metode *Fuzzy K-Nearest Neighbor* berjalan dengan benar.

6.2 Pengujian Akurasi

Pengujian akurasi dilakukan dengan tujuan untuk mengetahui kecocokan hasil diagnosa Pemodelan Sistem Pakar Untuk Identifikasi Penyakit Pada Tanaman Kedelai Menggunakan Metode *Fuzzy K-Nearest Neighbor* dengan hasil diagnosa pakar. Terdapat dua pengujian akurasi yang digunakan yaitu pengujian nilai k dan

pengujian jumlah variasi data dengan menggunakan nilai K terbaik pada pengujian sebelumnya.

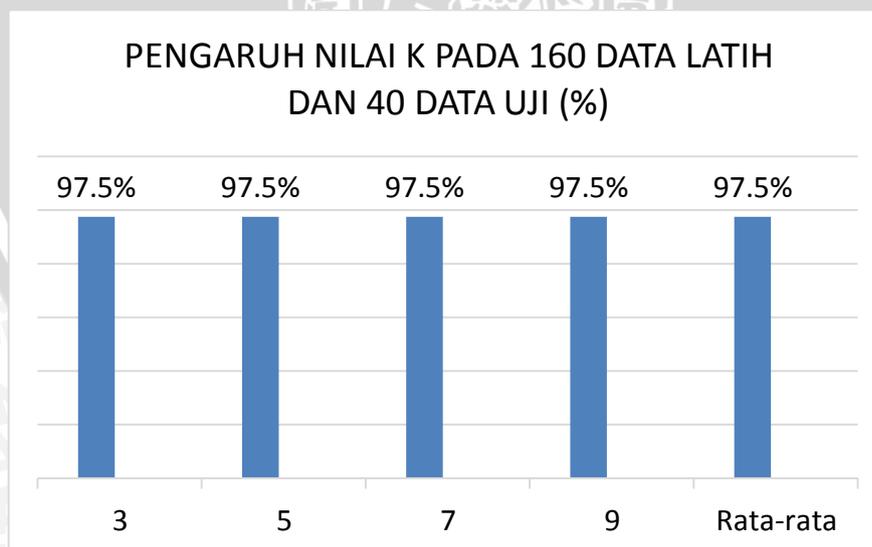
6.2.1 Prosedur dan Hasil Pengujian Nilai K

Pengujian nilai k dilakukan dengan tujuan untuk mengetahui tingkat akurasi antara hasil diagnosa sistem dengan hasil diagnosa yang dilakukan oleh pakar terhadap variabel k. Pengujian dilakukan untuk mencari nilai hasil akurasi tertinggi, nilai k dengan hasil tertinggi akan menjadi nilai tetap pada pengujian akurasi variasi jumlah data. Pengujian variasi nilai k dilakukan dengan menggunakan 160 data latih dan 40 data uji dengan variasi nilai k yaitu k=3, k=5, k=7 dan k=9. Hasil pengujian nilai k ditunjukkan pada Tabel 6.2.

Tabel 6.2 Hasil pengujian nilai k

Variasi Nilai K	Akurasi (%)
3	97.5%
5	97.5%
7	97.5%
9	97.5%
Rata-rata Akurasi	97.5%

Hasil pengujian pada Tabel 6.2 menunjukkan pengaruh nilai k terhadap tingkat akurasi. Pengujian menggunakan 160 data latih dan 40 data uji dengan nilai k=3, menghasilkan tingkat akurasi 97.5%. Pengujian dengan nilai k=5



Gambar 6.2 Grafik hasil pengujian pengaruh nilai k

menghasilkan tingkat akurasi 97.5%. Pengujian dengan nilai k=7 menghasilkan tingkat akurasi 97.5%. Pengujian dengan nilai k=9 menghasilkan tingkat akurasi 97.5%. Total rata-rata akurasi pada sistem pakar ini yaitu sebesar 97.5%. Grafik

hasil pengujian pengaruh nilai k pada 160 data latih dan 40 data uji ditunjukkan pada Gambar 6.2.

6.2.2 Analisis Pengujian Nilai K

Berdasarkan pengujian akurasi variasi nilai k diperoleh nilai akurasi yang sama pada semua nilai k yaitu sebesar 97.5%. Hal tersebut karena nilai k yang digunakan dalam pengujian adalah k klasifikasi untuk menentukan jarak data uji pada tiap data latih. Sedangkan pada langkah sebelumnya telah diketahui nilai keanggotaan kelas pada data latih dengan menggunakan nilai k inisialisasi sama dengan 3.

6.2.3 Prosedur dan Hasil Pengujian Jumlah Variasi Data

Pengujian jumlah variasi data ini digunakan untuk mengetahui tingkat akurasi data dengan menggunakan nilai k terbaik yang didapat pada pengujian nilai k. Pada pengujian nilai k, pada semua nilai k tersebut memiliki akurasi yang sama yaitu 97.5%, sehingga dalam pengujian jumlah variasi data ini menggunakan salah satu nilai k yaitu k=3. Pengujian jumlah variasi data meliputi 160 data latih dengan 40 data uji, 130 data latih dengan 70 data uji, 100 data latih dengan 100 data uji dan 70 data latih dengan 130 data uji. Hasil pengujian variasi data ditunjukkan pada Tabel 6.3.

Tabel 6.3 Hasil pengujian variasi data

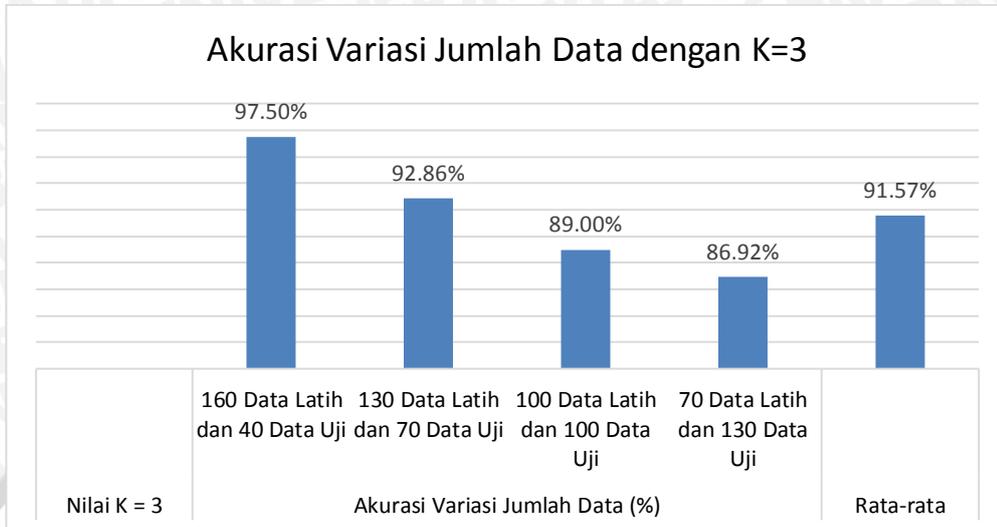
Nilai K	Akurasi Variasi Jumlah Data (%)				Rata-rata
	160 data latih dan 40 data uji	130 data latih dan 70 data uji	100 data latih dan 100 data uji	70 data latih dan 130 data uji	
3	97.50%	92.86%	89.00%	86.92%	91.57%

Hasil pengujian pada Tabel 6.3 menunjukkan pengaruh variasi jumlah data dengan nilai k yang sama terhadap tingkat akurasi. Pengujian dengan 160 data latih dan 40 data uji menghasilkan tingkat akurasi sebesar 97.5%. Pengujian dengan 130 data latih dan 70 data uji menghasilkan tingkat akurasi sebesar 92.86%. Pengujian dengan 100 data latih dan 100 data uji menghasilkan tingkat akurasi sebesar 89%. Pengujian dengan 70 data latih dan 130 data uji menghasilkan tingkat akurasi sebesar 86.92%. Nilai akurasi minimum sebesar 86.92%, nilai akurasi maksimum sebesar 97.5% dan total rata-rata akurasi sistem pakar diagnosa penyakit tanaman kedelai sebesar 91.57%. Grafik pengujian variasi variasi jumlah data dengan k=3 dapat dilihat pada Gambar 6.3.

6.2.4 Analisis Pengujian Jumlah Variasi Data

Berdasarkan pengujian akurasi variasi jumlah data maka diperoleh nilai rata-rata akurasi pada masing-masing jumlah data latih dan data uji. Pada grafik akurasi variasi jumlah data pada Gambar 6.3 dapat dilihat bahwa grafik dengan jumlah data latih lebih banyak memiliki tingkat akurasi yang lebih besar daripada jumlah

data latih yang lebih sedikit. Sehingga jumlah data latih sebanyak 160 data dapat digunakan sebagai ketetapan jumlah data latih.



Gambar 6.3 Grafik hasil pengujian variasi jumlah data

6.2.5 Prosedur dan Hasil Pengujian Perbandingan Akurasi Fuzzy K-NN dan K-NN

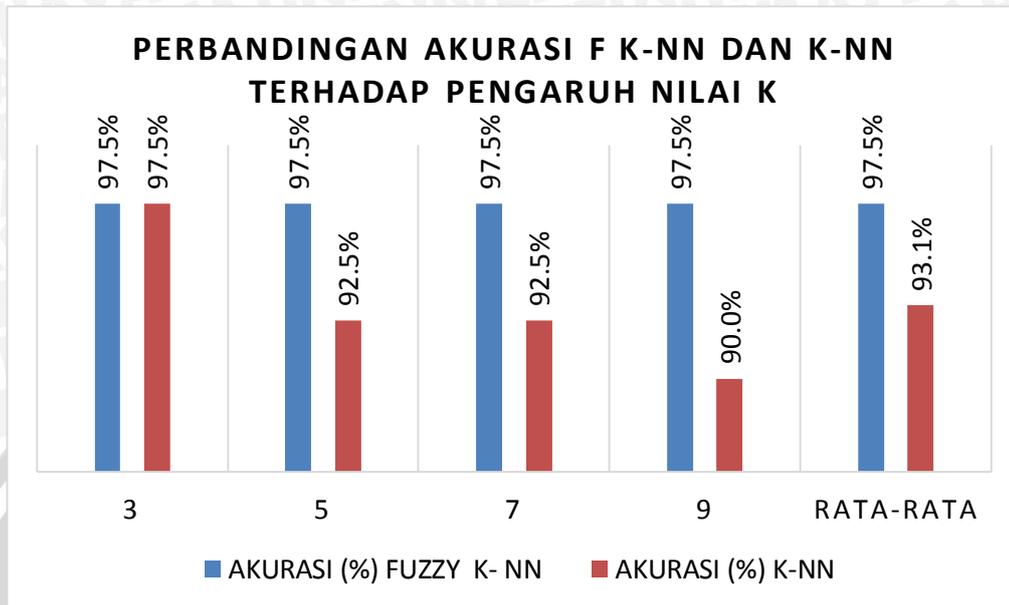
Pengujian perbandingan akurasi antara metode *Fuzzy K-NN* dan *K-NN* dilakukan untuk menghitung serta membandingkan tingkat akurasi pada metode *Fuzzy K-NN* dan *K-NN*. Pengujian pada metode *K-NN* dilakukan sama dengan pengujian pada metode *Fuzzy K-NN* yaitu pengujian variasi nilai *k* dan pengujian variasi data. Hasil pengujian perbandingan akurasi metode *Fuzzy K-NN* dan *K-NN* terhadap pengaruh nilai *k* ditunjukkan pada Tabel 6.4 dan hasil Pengujian perbandingan akurasi metode *Fuzzy K-NN* dan *K-NN* terhadap variasi jumlah data ditunjukkan pada Tabel 6.5.

Tabel 6.4 Hasil perbandingan *fuzzy k-nn* dan *k-nn* terhadap pengaruh nilai *k*

Variasi Nilai K	Akurasi (%)	
	Fuzzy K-NN	K-NN
3	97.5%	97.5%
5	97.5%	92.5%
7	97.5%	92.5%
9	97.5%	90.0%
Rata-rata Akurasi	97.5%	93.1%

Hasil pengujian perbandingan metode *fuzzy k-nn* dan *k-nn* terhadap pengaruh nilai *k* pada Tabel 6.4 menunjukkan nilai akurasi untuk *k=3* pada metode *fuzzy k-nn* sebesar 97.5% dan pada metode *k-nn* sebesar 97.5%, untuk *k=5* pada metode *fuzzy k-nn* sebesar 97.5% dan pada metode *k-nn* sebesar 92.5%, untuk *k=7* pada metode *fuzzy k-nn* sebesar 97.5% dan pada metode *k-nn* sebesar 92.5%, dan untuk *k=9* pada metode *fuzzy k-nn* sebesar 97.5% dan pada metode *k-nn* sebesar 90%.

Rata-rata akurasi pada metode *fuzzy k-nn* sebesar 97.5% dan pada metode *k-nn* sebesar 93.1%. Grafik hasil perbandingan metode *fuzzy k-nn* dan *k-nn* terhadap pengaruh nilai *k* ditunjukkan pada Gambar 6.4.

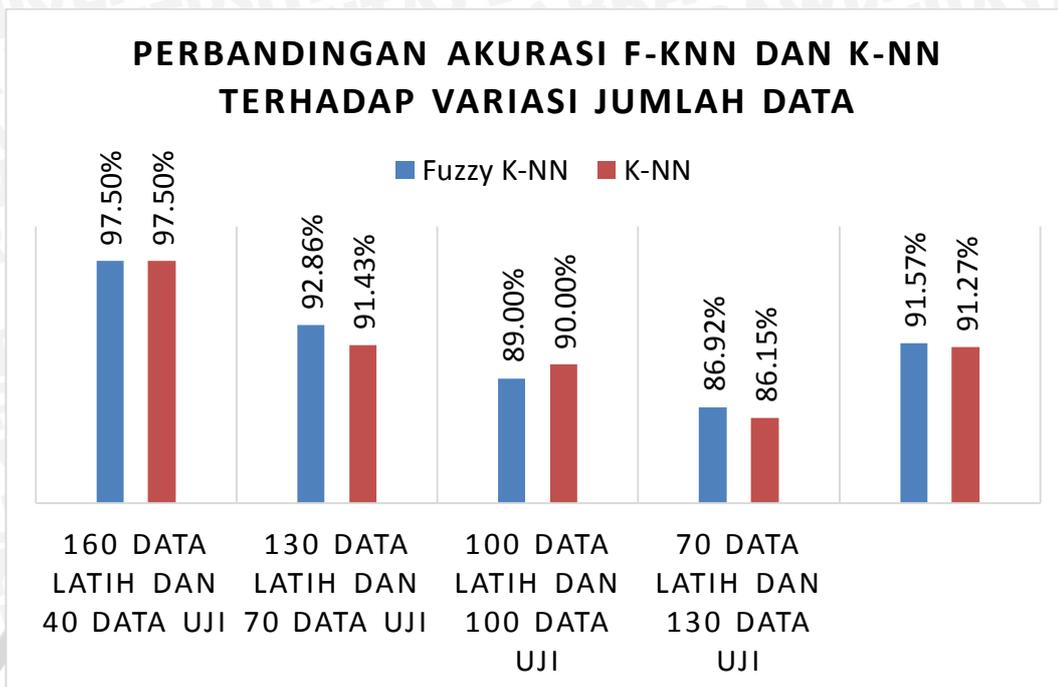


Gambar 6.4 Grafik hasil perbandingan F K-NN dan K-NN terhadap nilai *k*

Hasil pengujian perbandingan metode *fuzzy k-nn* dan *k-nn* terhadap variasi jumlah data pada Tabel 6.5 menunjukkan nilai akurasi untuk 160 data latih dan 40 data uji pada metode *fuzzy k-nn* sebesar 97.5% dan pada metode *k-nn* sebesar 97.5%, untuk 130 data latih dan 70 data uji pada metode *fuzzy k-nn* sebesar 92.86% dan pada metode *k-nn* sebesar 91.43%, untuk 100 data latih dan 100 data uji pada metode *fuzzy k-nn* sebesar 89.00% dan pada metode *k-nn* sebesar 90.00%, dan untuk 70 data latih dan 130 data uji pada metode *fuzzy k-nn* sebesar 86.92% dan pada metode *k-nn* sebesar 86.15%. Rata-rata akurasi pada metode *fuzzy k-nn* sebesar 91.57% dan pada metode *k-nn* sebesar 91.27%. Grafik hasil perbandingan metode *fuzzy k-nn* dan *k-nn* terhadap pengaruh nilai *k* ditunjukkan pada Gambar 6.5.

Tabel 6.5 Hasil perbandingan *fuzzy k-nn* dan *k-nn* terhadap variasi jumlah data

Nilai K = 3	Akurasi Variasi Jumlah Data (%)				Rata-rata
	160 Data Latih dan 40 Data Uji	130 Data Latih dan 70 Data Uji	100 Data Latih dan 100 Data Uji	70 Data Latih dan 130 Data Uji	
Fuzzy K-NN	97.50%	92.86%	89.00%	86.92%	91.57%
K-NN	97.50%	91.43%	90.00%	86.15%	91.27%



Gambar 6.5 Grafik hasil perbandingan F K-NN dan K-NN variasi jumlah data

6.2.6 Analisis Pengujian Perbandingan Akurasi Fuzzy K-NN dan K-NN

Pada Tabel 6.6 ditunjukkan data perbandingan antara pakar, metode f k-nn dan metode k-nn untuk mengetahui tingkat perbandingan akurasi terhadap pengaruh nilai $k=7$ dengan menggunakan 160 data latih dan 40 data uji. Pada data ke-7 hasil diagnosa pakar adalah penyakit 1, metode f k-nn menghasilkan diagnosa penyakit 1, namun metode k-nn menghasilkan diagnosa penyakit 3. Pada data ke-20 hasil diagnosa pakar adalah penyakit 3, metode *fuzzy* k-nn menghasilkan diagnosa penyakit 3, namun metode k-nn menunjukkan diagnosa penyakit 1. Data pada Tabel 6.6 dilampirkan pada Lampiran 5.

Pada Tabel 6.7 ditunjukkan data perbandingan antara pakar, metode f k-nn dan metode k-nn untuk mengetahui tingkat perbandingan akurasi terhadap pengaruh variasi jumlah data dengan menggunakan 130 data latih dan 70 data uji. Pada data ke-54 hasil diagnosa pakar adalah penyakit 4, metode f k-nn menghasilkan diagnosa penyakit 4, namun metode k-nn menghasilkan diagnosa penyakit 2. Data pada Tabel 6.6 dilampirkan pada Lampiran 6.

Dari beberapa contoh data tersebut dapat dilihat bahwa metode *fuzzy* k-nn menghasilkan diagnosa yang lebih baik daripada metode k-nn. Gambar merupakan hasil diagnosa data ke-7 pada nilai $k=7$ dengan menggunakan 160 data latih dan 40 data uji. Gambar menunjukkan bahwa hasil pengurutan jarak data ke-7 masuk ke dalam penyakit 3 dilihat dari kelas mayoritas, namun hasil diagnosa akhir data ke-7 masuk dalam penyakit 1. Hal tersebut karena metode *fuzzy* k-nn penentuan kelas tidak hanya dilihat dari jumlah data terbanyak yang mengikuti suatu kelas namun juga ditentukan berdasarkan nilai keanggotaan kelas terbesar.

Sedangkan pada metode k-nn penentuan kelas ditentukan berdasarkan jumlah data terbanyak yang mengikuti suatu kelas.

HASIL DIAGNOSA

DATA INPUT

G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	G16
1	0.15	0.1	0	0	0.15	0.75	0	0	0	0	0	0	0	0	0

PENGURUTAN JARAK DENGAN K=7

No.	ID Jarak	Hasil Jarak	Diagnosa
1	15	1.0198039027186	1
2	14	1.024695076596	1
3	28	1.0488088481702	1
4	81	1.0630145812735	3
5	89	1.0630145812735	3
6	94	1.0630145812735	3
7	85	1.0677078252031	3

NILAI KEANGGOTAAN DENGAN M= 2

Penyakit	Nilai Keanggotaan
1	1
2	0
3	0
4	0
5	0

Gambar 6.6 Hasil diagnosa data ke-7 dengan nilai k = 7

Tabel 6.6 Data perbandingan antara diagnosa pakar, f k-nn dan k-nn terhadap nilai k

No	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	G16	Pakar	Fuzzy K-NN	K-NN	
1	1	0.15	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	1	1	1	
.
7	1	0.15	0.1	0	0	0.15	0.75	0	0	0	0	0	0	0	0	0	1	1	3	
.
20	0	0.75	0	0	0	0	0.75	0	0	0	0	0	0	0	0.15	0	3	3	1	
.
.
.
34	0.2	0	0	0.2	0	0	0.15	0	0	0	0	0.15	0	0	0.15	0.1	5	4	2	
35	0	0	0	0	0	0	0	0	0	0.1	0.2	0	0.1	0.2	0.75	0	5	5	5	
.
.
.
39	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0.2	0.75	0.5	5	5	5	
40	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0.2	0.75	0.1	5	5	5	

Tabel 6.7 Data perbandingan antara diagnosa pakar, f k-nn dan k-nn terhadap variasi jumlah data

No	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	G16	Pakar	Fuzzy K-NN	K-NN
1	1	0.15	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	1	1	1
.
7	1	0.15	0.1	0	0	0.15	0.75	0	0	0	0	0	0	0	0	0	1	3	3
.
.
32	0	0.75	0	0	0	0	0.75	0	0	0	0	0	0	0	0.15	0	3	1	1
.
.
41	0	0.15	0.1	0	0	0	0.75	0.2	0	0	0	0.75	0.1	0	0	0.1	3	4	4
.
.
54	0	0	0	0.2	0	0	0	0	0	0.1	0	0.15	0.5	0	0	0.1	4	4	2
.
.
70	0	0.15	0	0.2	0	0.15	0	0	0.15	0.1	0	0.15	0.1	0.2	0.75	0.5	5	5	5

BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan perancangan, implementasi, dan hasil pengujian dari penelitian dengan judul Pemodelan Sistem Pakar Untuk Identifikasi Penyakit Pada Tanaman Kedelai Menggunakan Metode Fuzzy K-Nearest Neighbor ini, maka dapat diambil kesimpulan sebagai berikut:

1. Pemodelan sistem pakar ini mampu mengidentifikasi penyakit pada tanaman kedelai dengan menginputkan 16 parameter gejala ke dalam sistem yang kemudian diproses menggunakan metode *fuzzy k-nn* untuk mengetahui hasil diagnosa penyakit tanaman kedelai. Fitur pada pemodelan sistem pakar ini meliputi fitur beranda untuk menampilkan informasi umum dan penyakit tanaman kedelai, fitur diagnosa untuk menampilkan form input gejala dan fitur data untuk menampilkan data latih dan data uji yang digunakan dalam sistem.
2. Pengujian akurasi pada pemodelan sistem pakar ini meliputi:
 - a. Pengujian pengaruh nilai k klasifikasi dengan menggunakan 160 data latih dan 40 data uji memiliki nilai rata-rata akurasi sebesar 97.5%.
 - b. Pengujian pengaruh variasi jumlah data memiliki nilai rata-rata akurasi sebesar 91.57%.
 - c. Pengujian perbandingan metode *fuzzy k-nn* dan *k-nn* terhadap pengaruh nilai k memiliki nilai rata-rata akurasi pada metode *fuzzy k-nn* sebesar 97.5% dan pada metode *k-nn* sebesar 93.1%.
 - d. Pengujian perbandingan metode *fuzzy k-nn* dan *k-nn* terhadap pengaruh variasi jumlah data memiliki nilai rata-rata akurasi pada metode *fuzzy k-nn* sebesar 91.57% dan pada metode *k-nn* sebesar 91.27%.

Dari hasil pengujian tersebut menunjukkan bahwa metode *fuzzy k-nn* pada pemodelan sistem pakar ini memiliki tingkat akurasi yang baik untuk mengidentifikasi penyakit pada tanaman kedelai.

7.2 Saran

Pemodelan Sistem Pakar Untuk Identifikasi Penyakit Pada Tanaman Kedelai Menggunakan Metode Fuzzy K-Nearest Neighbor ini masih memiliki beberapa kekurangan. Saran yang dapat diberikan untuk pengembangan sistem pakar ini yaitu pemodelan sistem pakar ini dapat dikembangkan lebih lanjut menjadi sistem pakar yang dapat membantu dalam mengidentifikasi penyakit pada tanaman kedelai dengan menambahkan fitur-fitur lainnya seperti hak akses pakar, sehingga pakar dapat secara langsung menambahkan jika terdapat gejala dan penyakit baru ke dalam sistem.

DAFTAR PUSTAKA

- Arnanda, F., Kriswanto, Y., Izzatun, I., Nurlita, D., Fajriyani, A. & Utami, T.A., 2015. *Pemodelan Ketahanan Pangan Kedelai (Glycine Soya Max (Lenus&Merril)) Di Provinsi Jawa Tengah Dengan Pendekatan Spatial Regression*. Tersedia di: <<http://jurnal.unimus.ac.id/index.php/statistik/article/download/1429/1482>> [Diakses 26 September 2015]
- Badan Penelitian dan Pengembangan Pertanian Kementerian Pertanian, (2013)
- Badan Pusat Statistika (2015)
- Bertalya, 2009. *Konsep Data Mining*. Tersedia di: <<https://bertalya.staff.gunadarma.ac.id>> [Diakses 21 Oktober 2015]
- Fadli, A., 2010. *Sistem Pakar Dasar*. Tersedia di: <http://nyoman.staf.narotama.ac.id/files/2012/01/Ari_Fadli_Sistem_Pakar_Dasar.pdf> [Diakses 24 September 2015]
- Keller, J. M., Fray, M. R., Givens, J. A., 1985. A Fuzzy K-Nearest Neighbor Algorithm. IEEE.
- Kusumadewi, S., Purnomo, H., 2010. *Aplikasi Logika Fuzzy Untuk Pendukung Keputusan*. Yogyakarta: Graha Ilmu
- Meristika, Y.S., 2013. *Perbandingan K-Nearest Neighbor dan Fuzzy K-Nearest Neighbor pada Diagnosis Penyakit Diabetes Melitus*. S1. Universitas Brawijaya
- Rofika, 2015. *Sistem Pakar Diagnosa Penyakit Kulit Pada Anak Menggunakan Metode Fuzzy K-Nearest Neighbor*. S1. Universitas Brawijaya.
- Rukmana, R., Yudirachman, H. 2014. *Budi Daya dan Pengolahan Hasil Kacang Kedelai Unggul*. Bandung: Nuansa Aulia.
- Sutojo, T, dkk. 2011. *Kecerdasan Buatan*. Penerbit ANDI, Yogyakarta
- Ukmala, S., 2015. *Pemodelan Sistem Pakar Untuk Identifikasi Penyakit Pada Tanaman Tomat Menggunakan Metode Fuzzy K-Nearest Neighbor*. S1. Universitas Brawijaya
- Wijaya, M.T., 2014. *Implementasi Fuzzy K-Nearest Neighbor (FK-NN) untuk Prediksi Kelas Kebakaran Hutan*. S1. Universitas Brawijaya
- Wisdariato, A., Ridok, A., Rahman, M.A., 2013. *Penerapan Metode Fuzzy K-Nearest Neighbor (FK-NN) Untuk Pengklasifikasian Spam Email*. Program Studi Ilmu Komputer Universitas Brawijaya
- Yan, C., 2012, *Feature extraction using fuzzy complete linear discriminant analysis*. Nanjing University of Science and Technology
- Zainudin, S., Hidayat, N., Soebroto, A.A., 2013. *Penerapan Algoritma Modified K-Nearest Neighbor (M-KNN) Pada Pengklasifikasian Penyakit Tanaman Kedelai*. Program Studi Ilmu Komputer Universitas Brawijaya

LAMPIRAN 1

HASIL WAWANCARA

Tempat : Balai Pengkajian Teknologi Pertanian (BPTP) Jawa Timur

Tanggal : 29 Februari 2016

Nama Pakar : Prof. Dr. Ir. Moh. Cholil Mahfud, M.S.

No	Pertanyaan	Jawaban Pakar
1	Apakah di Balai Pengkajian Teknologi Pertanian (BPTP) Jawa Timur ini terdapat budidaya tanaman kedelai?	Ya, di Balai Pengkajian Teknologi Pertanian (BPTP) Jawa Timur ini ada budidaya tanaman kedelai.
2	Apakah tanaman kedelai di BPTP memiliki beberapa kendala seperti serangan penyakit?	Ya, terdapat beberapa tanaman kedelai yang terserang penyakit.
3	Penyakit apa sajakah yang menyerang pada tanaman kedelai?	Banyak, ada karat, pustule bakteri, virus mozaik dan lain lain
4	Saya ingin meneliti tentang penyakit tanaman kedelai untuk di aplikasikan ke dalam sistem pakar, penyakit-penyakit apa sajakah yang terdapat pada tanaman kedelai?	Banyak, tetapi terdapat 5 penyakit utama pada tanaman kedelai yang dapat menyebabkan kegagalan panen.
5	Apa sajakah parameter yang digunakan untuk mendiagnosa penyakit pada tanaman kedelai?	Parameternya yaitu berdasarkan gejala setiap penyakit yang ada pada tanaman kedelai tersebut.

Mengetahui,

Prof. Dr. Ir. Moh. Cholil Mahfud, M.S.

LAMPIRAN 2

DATA ATURAN PENYAKIT TANAMAN KEDELAI

Tempat : Balai Pengkajian Teknologi Pertanian (BPTP) Jawa Timur

Tanggal : 29 Februari 2016

Nama Pakar : Prof. Dr. Ir. Moh. Cholil Mahfud, M.S.

Kode	Gejala	Penyakit
G1	Bercak berisi uredia pada daun	Karat
G2	Bentuk bercak bersudut banyak	Karat
G3	Bercak kecil, agak bulat, kering, berwarna coklat dikelilingi warna kuning	Karat Target Spot Hawar Batang
G4	Bercak hijau pucat pada daun	Pustul Bakteri
G5	Bercak coklat muda atau putih pada daun	Pustul Bakteri
G6	Bercak kecil menyebabkan daun berlubang	Pustul Bakteri
G7	Daun berkerut	Virus Mosaik
G8	Daun memiliki gambaran mosaik	Virus Mosaik
G9	Tepi daun mengalami klorosis	Virus Mosaik
G10	Biji Mengecil	Virus Mosaik Pustul Bakteri
G11	Bentuk lingkaran seperti papan tembak pada daun	Target Spot
G12	Biji bercak coklat kemerahan	Target Spot
G13	Bercak coklat kemerahan pada batang	Target Spot Karat
G14	Miselium di pangkal batang	Hawar Batang
G15	Bercak coklat pada pangkal dan berkembang cepat	Hawar Batang
G16	Tanaman layu	Hawar Batang

Mengetahui,

Prof. Dr. Ir. Moh. Cholil Mahfud, M.S.

LAMPIRAN 3

DATA NILAI BOBOT GEJALA PENYAKIT TANAMAN KEDELAI

Tempat : Balai Pengkajian Teknologi Pertanian (BPTP) Jawa Timur

Tanggal : 29 Februari 2016

Nama Pakar : Prof. Dr. Ir. Moh. Cholil Mahfud, M.S.

No	Gejala	Penyakit	Opsi	Bobot
1	Bercak berisi uredia pada daun	1. Karat	Tidak ada	0
			Sedikit	0.2
			Banyak	1
2	Bentuk bercak bersudut banyak	1. Karat	Tidak ada	0
			Sedikit	0.15
			Banyak	0.75
3	Bercak kecil, agak bulat, kering, berwarna coklat dikelilingi warna kuning	1. Karat 2. Target Spot 3. Hawar Batang	Tidak ada	0
			Sedikit	0.1
			Banyak	0.5
4	Bercak hijau pucat pada daun	1. Pustul Bakteri	Tidak ada	0
			Sedikit	0.2
			Banyak	1
5	Bercak coklat muda atau putih pada daun	1. Pustul Bakteri	Tidak ada	0
			Sedikit	0.15
			Banyak	0.75
6	Bercak kecil menyebabkan daun berlubang	1. Pustul Bakteri	Tidak ada	0
			Sedikit	0.15
			Banyak	0.75
7	Daun berkerut	1. Virus Mosaik	Tidak ada	0
			Sedikit	0.15
			Banyak	0.75
8	Daun memiliki gambaran mosaik	1. Virus Mosaik	Tidak ada	0
			Sedikit	0.2
			Banyak	1

9	Tepi daun mengalami klorosis	1. Virus Mosaik	Tidak ada	0
			Sedikit	0.15
			Banyak	0.75
10	Biji Mengecil	1. Virus Mosaik	Tidak ada	0
		2. Pustul Bakteri	Sedikit	0.1
			Banyak	0.5
11	Bentuk lingkaran seperti papan tembak pada daun	1. Target Spot	Tidak ada	0
			Sedikit	0.2
			Banyak	1
12	Biji bercak coklat kemerahan	1. Target Spot	Tidak ada	0
			Sedikit	0.15
			Banyak	0.75
13	Bercak coklat kemerahan pada batang	1. Target Spot	Tidak ada	0
		2. Karat	Sedikit	0.1
			Banyak	0.5
14	Miselium di pangkal batang	1. Hawar Batang	Tidak ada	0
			Sedikit	0.2
			Banyak	0.1
15	Bercak coklat pada pangkal dan berkembangcepat	1. Hawar Batang	Tidak ada	0
			Sedikit	0.15
			Banyak	0.75
16	Tanaman layu	1. Hawar Batang	Tidak ada	0
			Sedikit	0.1
			Banyak	0.5

Mengetahui,

Prof. Dr. Ir. Moh. Cholil Mahfud, M.S.

LAMPIRAN 4

DATA UJI

Tempat : Balai Pengkajian Teknologi Pertanian (BPTP) Jawa Timur

Tanggal : 29 Februari 2016

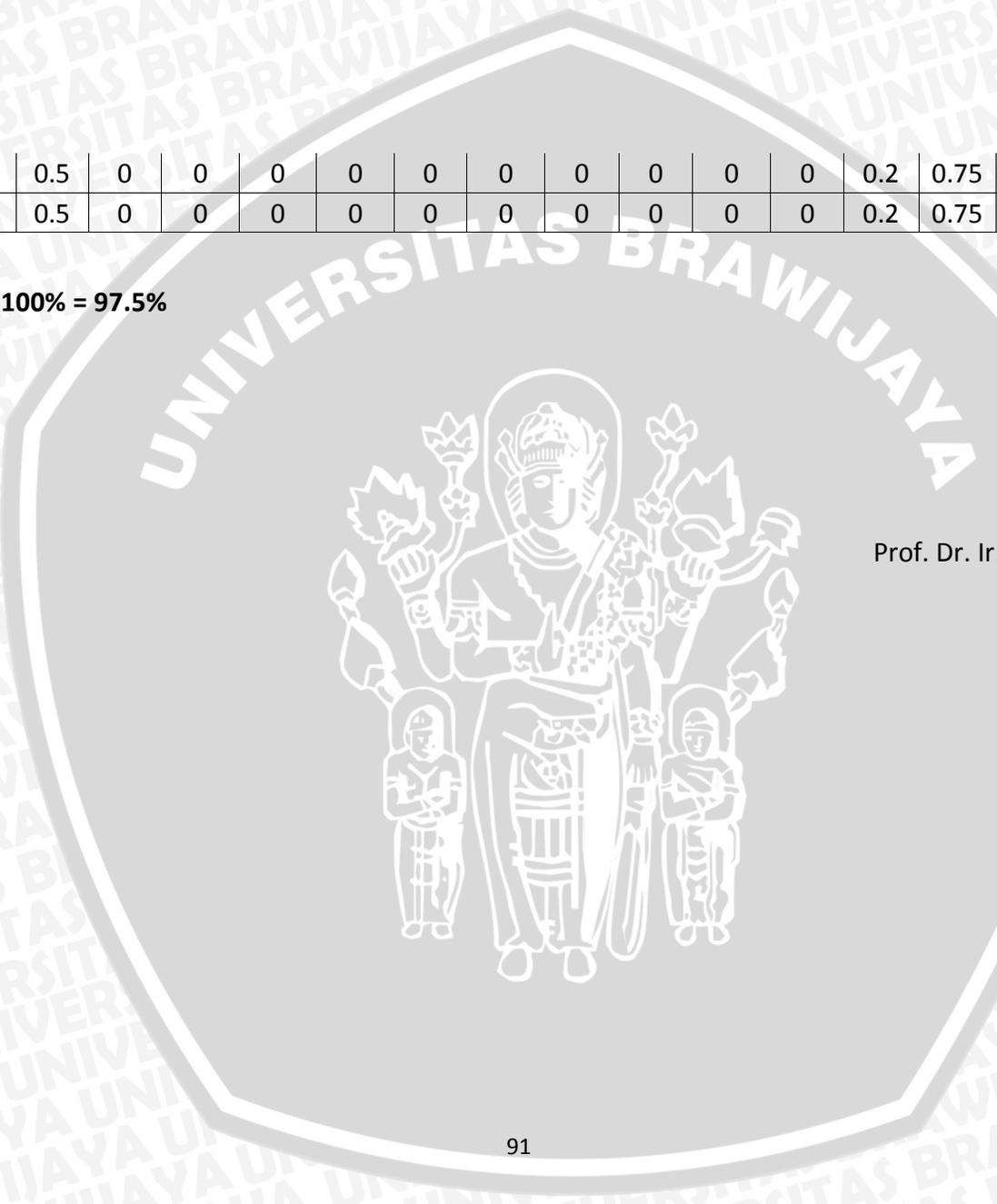
Nama Pakar : Prof. Dr. Ir. Moh. Cholil Mahfud, M.S.

No\Gejala	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	G16	Pakar	Sistem
1	1	0.15	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	1	1
2	1	0.15	0	0	0	0	0	0	0	0	0	0	0.1	0	0	0	1	1
3	1	0.15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
4	0.2	0.75	0.5	0	0	0	0	0	0	0	0	0	0.5	0	0	0	1	1
5	1	0	0.5	0.2	0.15	0.15	0	0	0.15	0	0.2	0	0.5	0.2	0	0.1	1	1
6	0.2	0.75	0.1	0	0	0.15	0	0	0	0.1	0	0	0	0	0	0	1	1
7	1	0.15	0.1	0	0	0.15	0.75	0	0	0	0	0	0	0	0	0	1	1
8	1	0.75	0.5	0	0	0.75	0.75	0.2	0	0.5	0	0	0.5	1	0.75	0.5	1	1
9	0	0	0	1	0.15	0.15	0	0	0	0.5	0	0	0	0	0	0	2	2
10	0	0	0	1	0.15	0.15	0	0	0	0.1	0	0	0	0	0	0	2	2
11	0	0	0	0.2	0.75	0.75	0	0	0	0	0	0	0	0	0	0	2	2
12	0	0	0	0.2	0.75	0.75	0	0	0	0.1	0	0	0	0	0	0	2	2
13	0	0.15	0	0	0.75	0.15	0	1	0.15	0	0	0.15	0	0	0	0.1	2	2
14	0	0	0.1	0	0.15	0.15	0	0	0.15	0	0	0	0	0	0.15	0	2	2

15	0	0	0	0	0.75	0	0	0	0	0.5	0.2	0	0.5	0.2	0	0	2	2
16	0	0.15	0	0	0.15	0.15	0	0.2	0.15	0.1	0	0	0	0	0	0.1	2	2
17	0	0	0	0	0	0	0.15	1	0.15	0	0	0	0	0	0	0	3	3
18	0	0	0	0	0	0	0.75	0.2	0.75	0.5	0	0	0	0	0	0	3	3
19	0	0	0	0	0	0	0.75	0.2	0.75	0.1	0	0	0	0	0	0	3	3
20	0	0.75	0	0	0	0	0.75	0	0	0	0	0	0	0	0.15	0	3	3
21	0	0	0	0	0	0	0	0	0.75	0	0	0.15	0	0	0	0	3	3
22	0	0	0	0	0.15	0	0.75	0	0	0	0	0	0	0	0	0.1	3	3
23	0	0	0	0	0	0	0.75	0	0	0.5	0	0.15	0	0	0	0	3	3
24	0	0	0	0	0	0	0.15	1	0.15	0	0	0	0.5	0	0	0.1	3	3
25	0	0	0.1	0	0	0	0	0	0	0	1	0.15	0.1	0	0	0	4	4
26	0	0	0.1	0	0	0	0	0	0	0	1	0.15	0	0	0	0	4	4
27	0	0	0.5	0	0	0	0	0	0	0	0.2	0.75	0.5	0	0	0	4	4
28	0	0	0.5	0	0	0	0	0	0	0	0.2	0.75	0.1	0	0	0	4	4
29	0	0.15	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4
30	0	0	0.1	0	0	0	0.15	0.2	0	0	0.2	0.15	1	0	0	0	4	4
31	0	0.75	0	0	0	0	0	0	0	0	0.2	0.75	0.5	0	0	0	4	4
32	0	0	0	0	0	0	0	0	0	0	1	0	0.1	0	0.15	0.1	4	4
33	0	0	0.5	0	0	0.15	0	0	0	0.1	0	0	0	0	0	0.1	5	5
34	0.2	0	0	0.2	0	0	0.15	0	0	0	0	0.15	0	0	0.15	0.1	5	4
35	0	0	0	0	0	0	0	0	0	0.1	0.2	0	0.1	0.2	0.75	0	5	5
36	0	0	0.5	0	0	0	0	0	0	0	0.2	0	0.1	0.2	0.15	0.1	5	5
37	0	0	0.5	0	0	0	0	0	0	0	0	0	0	1	0.15	0.1	5	5
38	0	0	0.1	0	0	0	0	0	0	0	0	0	0	1	0.15	0	5	5

39	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0.2	0.75	0.5	5	5
40	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0.2	0.75	0.1	5	5

Hasil Akurasi : $39/40 \times 100\% = 97.5\%$



Mengetahui,

Prof. Dr. Ir. Moh. Cholil Mahfud, M.S.

LAMPIRAN 5

HASIL PENGUJIAN PERBANDINGAN 40 DATA UJI DENGAN NILAI K=7

No	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	G16	Pakar	Fuzzy K-NN	K-NN
1	1	0.15	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	1	1	1
2	1	0.15	0	0	0	0	0	0	0	0	0	0	0.1	0	0	0	1	1	1
3	1	0.15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
4	0.2	0.75	0.5	0	0	0	0	0	0	0	0	0	0.5	0	0	0	1	1	1
5	1	0	0.5	0.2	0.15	0.15	0	0	0.15	0	0.2	0	0.5	0.2	0	0.1	1	1	1
6	0.2	0.75	0.1	0	0	0.15	0	0	0	0.1	0	0	0	0	0	0	1	1	1
7	1	0.15	0.1	0	0	0.15	0.75	0	0	0	0	0	0	0	0	0	1	1	3
8	1	0.75	0.5	0	0	0.75	0.75	0.2	0	0.5	0	0	0.5	1	0.75	0.5	1	1	1
9	0	0	0	1	0.15	0.15	0	0	0	0.5	0	0	0	0	0	0	2	2	2
10	0	0	0	1	0.15	0.15	0	0	0	0.1	0	0	0	0	0	0	2	2	2
11	0	0	0	0.2	0.75	0.75	0	0	0	0	0	0	0	0	0	0	2	2	2
12	0	0	0	0.2	0.75	0.75	0	0	0	0.1	0	0	0	0	0	0	2	2	2
13	0	0.15	0	0	0.75	0.15	0	1	0.15	0	0	0.15	0	0	0	0.1	2	2	2
14	0	0	0.1	0	0.15	0.15	0	0	0.15	0	0	0	0	0	0.15	0	2	2	2
15	0	0	0	0	0.75	0	0	0	0	0.5	0.2	0	0.5	0.2	0	0	2	2	2
16	0	0.15	0	0	0.15	0.15	0	0.2	0.15	0.1	0	0	0	0	0	0.1	2	2	2
17	0	0	0	0	0	0	0.15	1	0.15	0	0	0	0	0	0	0	3	3	3
18	0	0	0	0	0	0	0.75	0.2	0.75	0.5	0	0	0	0	0	0	3	3	3
19	0	0	0	0	0	0	0.75	0.2	0.75	0.1	0	0	0	0	0	0	3	3	3

20	0	0.75	0	0	0	0	0.75	0	0	0	0	0	0	0	0.15	0	3	3	1
21	0	0	0	0	0	0	0	0	0.75	0	0	0.15	0	0	0	0	3	3	3
22	0	0	0	0	0.15	0	0.75	0	0	0	0	0	0	0	0	0.1	3	3	3
23	0	0	0	0	0	0	0.75	0	0	0.5	0	0.15	0	0	0	0	3	3	3
24	0	0	0	0	0	0	0.15	1	0.15	0	0	0	0.5	0	0	0.1	3	3	3
25	0	0	0.1	0	0	0	0	0	0	0	1	0.15	0.1	0	0	0	4	4	4
26	0	0	0.1	0	0	0	0	0	0	0	1	0.15	0	0	0	0	4	4	4
27	0	0	0.5	0	0	0	0	0	0	0	0.2	0.75	0.5	0	0	0	4	4	4
28	0	0	0.5	0	0	0	0	0	0	0	0.2	0.75	0.1	0	0	0	4	4	4
29	0	0.15	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4,5
30	0	0	0.1	0	0	0	0.15	0.2	0	0	0.2	0.15	1	0	0	0	4	4	4
31	0	0.75	0	0	0	0	0	0	0	0	0.2	0.75	0.5	0	0	0	4	4	4
32	0	0	0	0	0	0	0	0	0	0	1	0	0.1	0	0.15	0.1	4	4	4
33	0	0	0.5	0	0	0.15	0	0	0	0.1	0	0	0	0	0	0.1	5	5	5
34	0.2	0	0	0.2	0	0	0.15	0	0	0	0	0.15	0	0	0.15	0.1	5	4	2
35	0	0	0	0	0	0	0	0	0	0.1	0.2	0	0.1	0.2	0.75	0	5	5	5
36	0	0	0.5	0	0	0	0	0	0	0	0.2	0	0.1	0.2	0.15	0.1	5	5	5
37	0	0	0.5	0	0	0	0	0	0	0	0	0	0	1	0.15	0.1	5	5	5
38	0	0	0.1	0	0	0	0	0	0	0	0	0	0	1	0.15	0	5	5	5
39	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0.2	0.75	0.5	5	5	5
40	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0.2	0.75	0.1	5	5	5

LAMPIRAN 6

HASIL PENGUJIAN PERBANDINGAN 130 DATA LATIH DAN 70 DATA UJI

No	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	G16	Pakar	Fuzzy K-NN	K-NN
1	1	0.15	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	1	1	1
2	1	0.15	0	0	0	0	0	0	0	0	0	0	0.1	0	0	0	1	1	1
3	1	0.15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
4	0.2	0.75	0.5	0	0	0	0	0	0	0	0	0	0.5	0	0	0	1	1	1
5	1	0	0.5	0.2	0.15	0.15	0	0	0.15	0	0.2	0	0.5	0.2	0	0.1	1	1	1
6	0.2	0.75	0.1	0	0	0.15	0	0	0	0.1	0	0	0	0	0	0	1	1	1
7	1	0.15	0.1	0	0	0.15	0.75	0	0	0	0	0	0	0	0	0	1	3	3
8	1	0.75	0.5	0	0	0.75	0.75	0.2	0	0.5	0	0	0.5	1	0.75	0.5	1	1	1
9	1	0.15	0.1	0	0	0	0	0	0	0	0	0	0.1	0	0	0	1	1	1
10	1	0.15	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
11	0.2	0.75	0.5	0	0	0	0	0	0	0	0	0	0.1	0	0	0	1	1	1
12	1	0.75	0.5	0	0	0	0.75	0	0	0	0	0.75	0.5	0	0.75	0.5	1	1	1
13	1	0.15	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	1	1	1
14	1	0.15	0.5	0.2	0	0	0.15	0	0	0	0	0.75	0.1	0	0	0.1	1	1	1
15	0	0	0	1	0.15	0.15	0	0	0	0.5	0	0	0	0	0	0	2	2	2
16	0	0	0	1	0.15	0.15	0	0	0	0.1	0	0	0	0	0	0	2	2	2
17	0	0	0	0.2	0.75	0.75	0	0	0	0	0	0	0	0	0	0	2	2	2
18	0	0	0	0.2	0.75	0.75	0	0	0	0.1	0	0	0	0	0	0	2	2	2

19	0	0.15	0	0	0.75	0.15	0	1	0.15	0	0	0.15	0	0	0	0.1	2	2	2
20	0	0	0.1	0	0.15	0.15	0	0	0.15	0	0	0	0	0	0.15	0	2	2	2
21	0	0	0	0	0.75	0	0	0	0	0.5	0.2	0	0.5	0.2	0	0	2	2	2
22	0	0.15	0	0	0.15	0.15	0	0.2	0.15	0.1	0	0	0	0	0	0.1	2	2	2
23	0	0	0	0.2	0.15	0.75	0	0	0	0.5	0	0	0	0	0	0	2	2	2
24	0	0	0	0.2	0.15	0.75	0	0	0	0.1	0	0	0	0	0	0	2	2	2
25	0	0	0	0	0.75	0	0.15	0	0	0.1	0	0	0	0	0.15	0	2	2	2
26	0	0	0	0.2	0	0.15	0	0	0	0.1	0	0	0	0	0	0.1	2	2	2
27	0	0	0	0.2	0.15	0.75	0.15	0	0.15	0.5	0.2	0	0	0	0	0	2	2	2
28	0	0.15	0	0.2	0.75	0	0	0	0	0	0.2	0	0.1	0	0	0.5	2	2	2
29	0	0	0	0	0	0	0.15	1	0.15	0	0	0	0	0	0	0	3	3	3
30	0	0	0	0	0	0	0.75	0.2	0.75	0.5	0	0	0	0	0	0	3	3	3
31	0	0	0	0	0	0	0.75	0.2	0.75	0.1	0	0	0	0	0	0	3	3	3
32	0	0.75	0	0	0	0	0.75	0	0	0	0	0	0	0	0.15	0	3	1	1
33	0	0	0	0	0	0	0	0	0.75	0	0	0.15	0	0	0	0	3	3	3
34	0	0	0	0	0.15	0	0.75	0	0	0	0	0	0	0	0	0.1	3	3	3
35	0	0	0	0	0	0	0.75	0	0	0.5	0	0.15	0	0	0	0	3	3	3
36	0	0	0	0	0	0	0.15	1	0.15	0	0	0	0.5	0	0	0.1	3	3	3
37	0	0	0	0	0	0	0.75	0.2	0.15	0	0	0	0	0	0	0	3	3	3
38	0	0	0	0	0	0	0.15	0.2	0.75	0.5	0	0	0	0	0	0	3	3	3
39	0	0	0	0	0	0	0.15	0.2	0.75	0.1	0	0	0	0	0	0	3	3	3
40	0	0.15	0	0	0	0	0.75	0	0	0.1	0	0	0	0.2	0	0	3	3	3
41	0	0.15	0.1	0	0	0	0.75	0.2	0	0	0	0.75	0.1	0	0	0.1	3	4	4
42	0	0.15	0	0	0	0	0	1	0.75	0	0	0	0	0.2	0	0.1	3	3	3

43	0	0	0.1	0	0	0	0	0	0	0	1	0.15	0.1	0	0	0	4	4	4
44	0	0	0.1	0	0	0	0	0	0	0	1	0.15	0	0	0	0	4	4	4
45	0	0	0.5	0	0	0	0	0	0	0	0.2	0.75	0.5	0	0	0	4	4	4
46	0	0	0.5	0	0	0	0	0	0	0	0.2	0.75	0.1	0	0	0	4	4	4
47	0	0.15	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4
48	0	0	0.1	0	0	0	0.15	0.2	0	0	0.2	0.15	1	0	0	0	4	4	4
49	0	0.75	0	0	0	0	0	0	0	0	0.2	0.75	0.5	0	0	0	4	4	4
50	0	0	0	0	0	0	0	0	0	0	1	0	0.1	0	0.15	0.1	4	4	4
51	0	0	0.1	0	0	0	0	0	0	0	0.2	0.75	0.5	0	0	0	4	4	4
52	0	0	0.1	0	0	0	0	0	0	0	0.2	0.75	0.1	0	0	0	4	4	4
53	0	0	0.5	0.2	0	0	0	0.2	0	0	1	0	0.5	0	0	0.1	4	4	4
54	0	0	0	0.2	0	0	0	0	0	0.1	0	0.15	0.5	0	0	0.1	4	4	2
55	0	0.15	0.5	0	0	0.15	0.15	0	0	0	0.2	0.75	0.5	0	0	0	4	4	4
56	0	0	0.1	0	0	0	0.15	0	0	0	0	0	0.1	0	0	0	4	2	2
57	0	0	0.5	0	0	0.15	0	0	0	0.1	0	0	0	0	0.1	5	5	5	
58	0.2	0	0	0.2	0	0	0.15	0	0	0	0	0.15	0	0	0.15	0.1	5	2	2
59	0	0	0	0	0	0	0	0	0	0.1	0.2	0	0.1	0.2	0.75	0	5	5	5
60	0	0	0.5	0	0	0	0	0	0	0	0.2	0	0.1	0.2	0.15	0.1	5	5	5
61	0	0	0.5	0	0	0	0	0	0	0	0	0	0	1	0.15	0.1	5	5	5
62	0	0	0.1	0	0	0	0	0	0	0	0	0	0	1	0.15	0	5	5	5
63	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0.2	0.75	0.5	5	5	5
64	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0.2	0.75	0.1	5	5	5
65	0	0	0.1	0	0	0	0	0	0	0	0	0	0	0.2	0.75	0.5	5	5	5
66	0	0	0.1	0	0	0	0	0	0	0	0	0	0	0.2	0.75	0.1	5	5	5

67	0	0	0.5	0	0	0	0	0	0.15	0	0	0.75	0	1	0.75	0.5	5	5	5
68	0.2	0	0.5	0	0.15	0	0	0.2	0	0.1	0	0.15	0.1	1	0.15	0.1	5	5	5
69	0	0	0.5	0	0	0	0.15	0	0.15	0	0	0	0.1	0	0.75	0.1	5	5	5
70	0	0.15	0	0.2	0	0.15	0	0	0.15	0.1	0	0.15	0.1	0.2	0.75	0.5	5	5	5

