

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

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Syela Ukmala

NIM: 115060801111033



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

LEMBAR PERSETUJUAN

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

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :

Syela Ukmala

NIM: 115060801111033

Skripsi ini telah diuji dan dinyatakan lulus pada

7 Januari 2016

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Nurul Hidayat, S.Pd., M.Sc

NIP: 196804302002121001

Indriati, ST., M.Kom

NIP: 198310132015042002

Mengetahui

Ketua Program Studi Informatika

Drs. Marji, MT

NIP: 196708011992031001

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, Januari 2016

Syela Ukmala

NIM: 115060801111033

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa atas rahmat dan hidayahNya sehingga penulis dapat menyelesaikan Skripsi yang berjudul **“Pemodelan Sistem Pakar Untuk Identifikasi Penyakit Pada Tanaman Tomat Menggunakan Metode Fuzzy K-Nearest Neighbor”**. Skripsi ini disusun dan diajukan sebagai salah satu syarat untuk memperoleh gelar sarjana S-1 di Program Studi Informatika Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya. Penulis menyadari bahwa penyusunan skripsi ini tidak akan berjalan lancar tanpa berbagai masukan yang bermanfaat dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis ingin menyampaikan hormat dan rasa terima kasih kepada:

1. Nurul Hidayat, S.Pd., M.Sc, selaku dosen Pembimbing I yang telah banyak memberikan ilmu dan saran selama penyusunan skripsi ini.
2. Indriati, ST., S.Kom, selaku dosen Pembimbing II yang juga telah banyak memberikan ilmu dan saran selama penyusunan skripsi ini.
3. Ir. Sutrisno, M.T, selaku Ketua Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
4. Drs. Marji, M.T, selaku Ketua Program Studi Informatika/Illmu Komputer Universitas Brawijaya.
5. Prof. Dr. Ir. Moh. Cholil Mahfud, M.S, selaku pakar penyakit tanaman yang telah memberikan ilmu dan bimbingan dalam penyusunan skripsi ini.
6. Narsodin dan Kunainah, kedua orang tua penulis serta Zahro Lasarik dan Ahda Fahmiza, saudara penulis yang senantiasa mendukung dan memberi doa demi terselesaikannya skripsi ini.
7. Sahabat tercinta Vivi Rusiana, Indri Fitri, dan Putri Maya yang tak bosan mengingatkan dan mendukung penulis.
8. Sahabat seperjuangan dalam menempuh S.Kom Afni Wilujeng, Dian Malasari, Prasiwi Meilida, Ika Khoirunnisak, dan Neny Silvia yang selalu mendukung dan membantu penulis.
9. Seluruh pihak yang telah membantu penulisan yang tidak dapat penulis sebutkan satu persatu.

Penulis menyadari sepenuhnya bahwa skripsi ini masih jauh dari sempurna karena keterbatasan materi dan pengetahuan yang dimiliki penulis. Akhirnya, semoga skripsi ini dapat bermanfaat dan berguna bagi pembaca terutama mahasiswa PTIIK Universitas Brawijaya.

Malang, Januari 2016

Penulis

ABSTRAK

Syela Ukmala. 2015. Pemodelan Sistem Pakar Untuk Identifikasi Penyakit Pada Tanaman Tomat Menggunakan Metode *Fuzzy K-Nearest Neighbor*.

Fakultas Ilmu Komputer, Universitas Brawijaya, Malang.

Dosen Pembimbing: Nurul Hidayat S.Pd., M.Sc. dan Indriati, ST., M.Kom.

Tomat merupakan salah satu jenis buah yang dapat ditanam hampir di seluruh lahan pertanian Indonesia. Di Indonesia, lebih dari 5.000.000 ha luas lahan ditanami tomat dengan produksi mencapai 129 juta ton per tahun. Produksi tomat akan terus ditingkatkan untuk memenuhi permintaan pasar yang cenderung meningkat. Dalam meningkatkan produktifitas tomat, gangguan penyakit tanaman menjadi kendala petani karena keterbatasan pengetahuan tentang penyakit tanaman tomat dan jumlah pakar penyakit tanaman tomat sangat terbatas. Sehingga diperlukan sebuah sistem yang dapat mengidentifikasi penyakit tanaman tomat dengan mudah. Pemodelan sistem pakar ini menggunakan metode *fuzzy k-nearest neighbor* sebagai mesin inferensi untuk mengklasifikasikan jenis penyakit. Metode *fuzzy k-nearest neighbor* merupakan pengembangan dari metode *k-nearest neighbor*, yang mana akan dihitung nilai keanggotaan dari hasil *k-nearest neighbor*. Berdasarkan pengujian fungsional yang telah dilakukan, masing-masing kasus uji menghasilkan kesesuaian dengan kebutuhan sistem. Pengujian akurasi menggunakan 30 data uji. Data latih yang digunakan adalah variasi data seimbang yang berjumlah 48 dan variasi data tidak seimbang yang berjumlah 30, 50, dan 70. Serta menggunakan variasi $k=1$, $k=5$, $k=10$, $k=15$, $k=20$, $k=25$, dan $k=30$. Berdasarkan pengujian akurasi sistem dengan data latih seimbang, diperoleh hasil nilai akurasi tertinggi adalah 90% dan nilai akurasi terendah adalah 81,33333%. Sedangkan dengan data latih tidak seimbang diperoleh nilai akurasi tertinggi adalah 88,88889% dan nilai akurasi terendah adalah 13,33333%. Hal ini menunjukkan bahwa metode *fuzzy k-nearest neighbor* memiliki akurasi yang cukup baik untuk klasifikasi penyakit tanaman tomat menggunakan data latih seimbang.

Kata Kunci: *Fuzzy K-Nearest Neighbor*, Klasifikasi, Pemodelan sistem pakar, Penyakit tanaman tomat.

ABSTRACT

Syela Ukmala. 2015. Modeling Expert System To Identify Disease Of Tomato Plant Method Using Fuzzy K-Nearest Neighbor.

Faculty of Computer Science, University of Brawijaya, Malang.

Supervisor: Nurul Hidayat S.Pd., M.Sc. and Indriati, ST., M. Kom.

Tomatoes are one type of fruit that can be grown in almost all agricultural land in Indonesia. In Indonesia, more than 5,000,000 hectares of land planted with tomato production reached 129 million tons per year. Tomato production will continue to be improved to meet the market demand is likely to increase. In improving the productivity of tomato, nuisance plant disease becomes a constraint farmers due to limited knowledge about the disease and the number of tomato plants disease expert is very limited. So, we need a system that can identify diseases of tomato plants with ease. This expert system modeling using fuzzy k-nearest neighbor as an inference engine to classify the type of disease. Method fuzzy k-nearest neighbor is the development of k-nearest neighbor method, which will be calculated the value of the membership of the results of the k-nearest neighbor. Fungsionali based on testing that has been done, each test case generates conformity with the needs of the system. Accuracy testing using 30 test data. Training data is used by the data variation of 48 and unbalanced data variation which amounted to 30, 50, and 70. As well as using a variation $k = 1$, $k = 5$, $k = 10$, $k = 15$, $k = 20$, $k = 25$ and $k = 30$. By testing the accuracy of the system with a balanced training data, obtained by the results of the highest accuracy is 90% and the lowest accuracy is 81.33333%. While the unbalanced training data obtained by the value of the highest accuracy is 88.88889% and the lowest accuracy is 13.33333%. This shows that the method fuzzy k-nearest neighbor has a fairly good accuracy for the classification of the tomato plant diseases using training data balanced.

Keywords: Fuzzy K-Nearest Neighbor, classification, modeling expert system, diseases of tomato plants.

DAFTAR ISI

LEMBAR PERSETUJUAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xii
DAFTAR PERSAMAAN.....	xiii
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan	2
1.4 Manfaat.....	3
1.5 Batasan Masalah	3
1.6 Sistematika Pembahasan	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Pustaka	5
2.2 Tanaman Tomat.....	7
2.2.1 Sejarah Penyebaran Tanaman Tomat.....	7
2.2.2 Taksonomi Tanaman Tomat.....	7
2.2.3 Manfaat Tomat	8
2.2.4 Syarat Tumbuh Tanaman Tomat.....	10
2.2.5 Penyakit Tanaman Tomat	11
2.3 Pemodelan	14
2.3.1 Jenis-jenis Model.....	14
2.4 Sistem Pakar.....	15
2.4.1 Pengertian Sistem Pakar	15
2.4.2 Arsitektur Sistem Pakar.....	16
2.4.3 Basis Pengetahuan	17

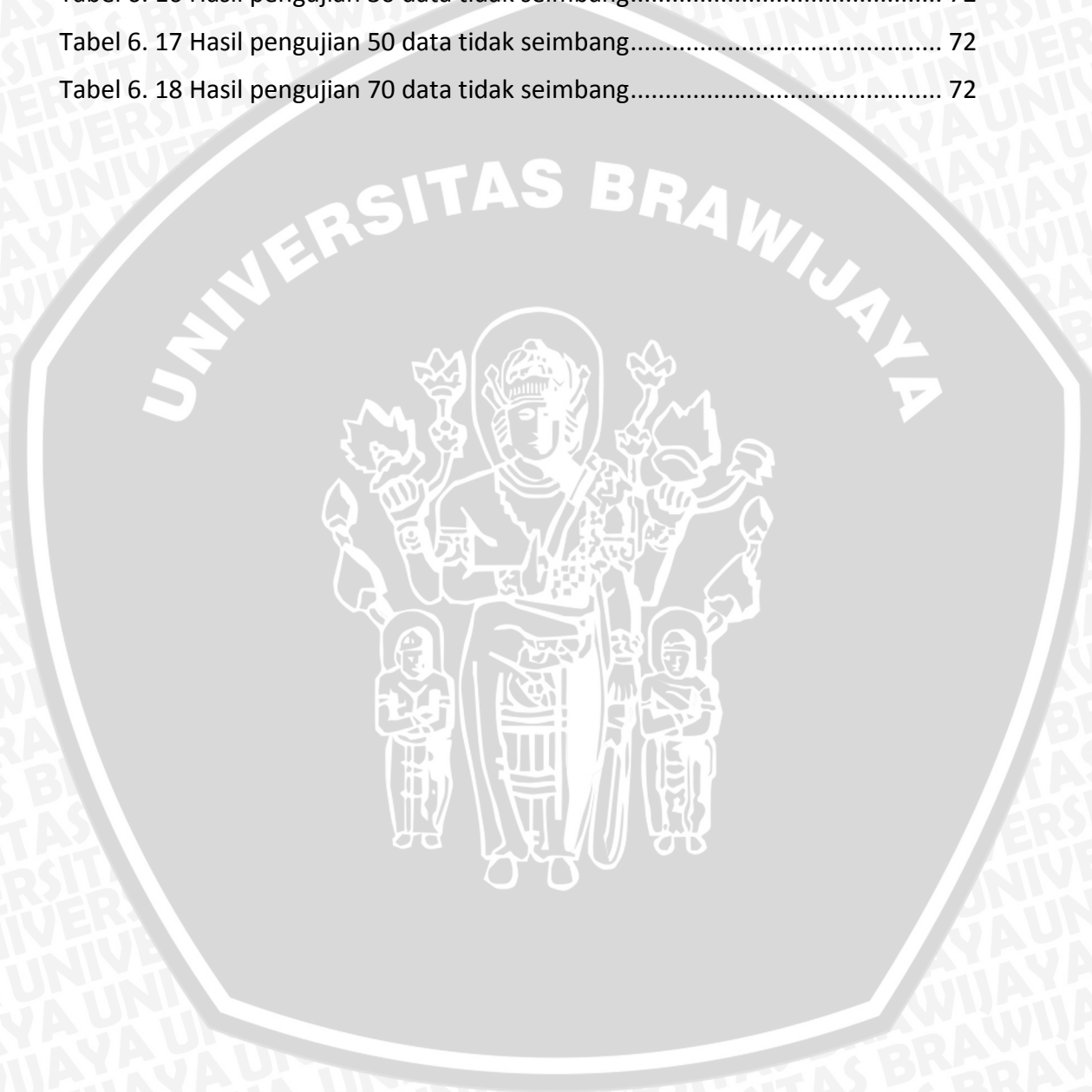
2.4.4 Representasi Pengetahuan	17
2.4.5 Mesin Inferensi.....	19
2.5 Fuzzy K-Nearest Neighbor.....	20
2.5.1 Proses <i>K-Nearest Neighbor</i>	20
2.5.2 Proses <i>Fuzzy K-Nearest Neighbor</i>	21
2.5.3 Kelebihan <i>Fuzzy K-Nearest Neighbor</i>	22
2.6 Pengujian	23
2.6.1 Pengujian Fungsional	23
2.6.2 Pengujian Akurasi.....	23
BAB 3 METODOLOGI	24
3.1 Studi Literatur	24
3.2 Pengumpulan Data	24
3.3 Analisa dan Perancangan.....	25
3.4 Implementasi Sistem	26
3.5 Pengujian Sistem.....	26
3.6 Kesimpulan.....	26
BAB 4 PERANCANGAN.....	27
4.1 Analisa Kebutuhan Perangkat Lunak	27
4.1.1 Identifikasi Pengguna	28
4.1.2 Analisa Kebutuhan Masukan.....	28
4.1.3 Analisa Kebutuhan Proses.....	28
4.1.4 Analisa Kebutuhan Keluaran	28
4.2 Perancangan Perangkat Lunak	29
4.2.1 Proses Klasifikasi	29
4.2.2 Normalisasi.....	30
4.2.3 Proses K-Nearest Neighbor	31
4.2.4 Proses Fuzzy K-Nearest Neighbor	32
4.3 Perancangan Sistem.....	34
4.3.1 Akuisisi Pengetahuan	35
4.3.2 Basis Pengetahuan	36
4.3.3 Mesin Inferensi.....	37
4.3.4 Perhitungan Manual.....	37

4.3.5 Perancangan Antarmuka.....	43
4.3.6 Perancangan Pengujian.....	46
BAB 5 PEMBAHASAN.....	49
5.1 Implementasi Spesifikasi Sistem.....	49
5.1.1 Spesifikasi Perangkat Keras.....	49
5.1.2 Spesifikasi Perangkat Lunak.....	50
5.2 Implementasi Algoritma.....	50
5.2.1 Implementasi Algoritma KNN.....	50
5.2.2 Implementasi Algoritma FKNN.....	54
5.3 Implementasi Program.....	56
5.3.1 Implementasi Antarmuka Halaman Identifikasi.....	56
5.3.2 Implementasi Antarmuka Halaman Hasil Identifikasi.....	57
5.3.3 Implementasi Antarmuka Halaman Login.....	58
5.3.4 Implementasi Antarmuka Halaman Admin.....	58
5.3.5 Implementasi Antarmuka Halaman Tambah Data.....	59
BAB 6 PENGUJIAN.....	60
6.1 Pengujian Fungsional.....	60
6.1.1 Prosedur Pengujian Fungsional.....	60
6.1.2 Hasil dan Analisis Pengujian Fungsional.....	63
6.2 Pengujian Akurasi Sistem.....	65
6.2.1 Prosedur Pengujian Akurasi Sistem.....	65
6.2.2 Hasil dan Analisis Pengujian Akurasi Sistem.....	71
BAB 7 PENUTUP.....	74
7.1 Kesimpulan.....	74
7.2 Saran.....	74
DAFTAR PUSTAKA.....	75
Lampiran Pembobotan Gejala.....	76
Lampiran Data Latih.....	77
Lampiran Pengujian.....	81

DAFTAR TABEL

Tabel 2. 1 Kajian pustaka.....	5
Tabel 2. 2 Kandungan nutrisi tomat.....	8
Tabel 4. 1 Identifikasi aktor	28
Tabel 4. 2 Kriteria gejala penyakit tanaman tomat.....	35
Tabel 4. 3 Aturan jenis penyakit tanaman tomat	37
Tabel 4. 4 Data latih penyakit tanaman tomat	38
Tabel 4. 5 Data uji penyakit tanaman tomat	38
Tabel 4. 6 Range nilai atribut	38
Tabel 4. 7 Normalisasi data latih.....	39
Tabel 4. 8 Normalisasi data uji	39
Tabel 4. 9 Jarak euclidean	39
Tabel 4. 10 Jarak euclidean diurutkan	40
Tabel 4. 11 Perancangan pengujian fungsional sistem.....	46
Tabel 4. 12 Perancangan pengujian 48 data seimbang	47
Tabel 4. 13 Perancangan pengujian 30 data tidak seimbang	47
Tabel 4. 14 Perancangan pengujian 50 data tidak seimbang	47
Tabel 4. 15 Perancangan pengujian 70 data tidak seimbang	48
Tabel 5. 1 Spesifikasi perangkat keras	49
Tabel 5. 2 Spesifikasi Perangkat Lunak	50
Tabel 6. 1 Penjelasan kasus uji login sukses	60
Tabel 6. 2 Penjelasan kasus uji login gagal.....	61
Tabel 6. 3 Penjelasan kasus uji logout	61
Tabel 6. 4 Penjelasan kasus uji menampilkan data latih	62
Tabel 6. 5 Penjelasan kasus uji menambahkan data latih	62
Tabel 6. 6 Penjelasan kasus uji menghapus data latih.....	62
Tabel 6. 7 Penjelasan uji menampilkan form identifikasi penyakit	63
Tabel 6. 8 Penjelasan kasus uji menampilkan hasil identifikasi penyakit.....	63
Tabel 6. 9 Hasil pengujian fungsional.....	63
Tabel 6. 10 Data uji.....	66
Tabel 6. 11 Variasi 48 data latih seimbang	67

Tabel 6. 12 Variasi 30 data latih tidak seimbang	68
Tabel 6. 13 Variasi 50 data latih tidak seimbang	68
Tabel 6. 14 Variasi 70 data latih tidak seimbang	69
Tabel 6. 15 Hasil pengujian 48 data seimbang.....	71
Tabel 6. 16 Hasil pengujian 30 data tidak seimbang.....	72
Tabel 6. 17 Hasil pengujian 50 data tidak seimbang.....	72
Tabel 6. 18 Hasil pengujian 70 data tidak seimbang.....	72

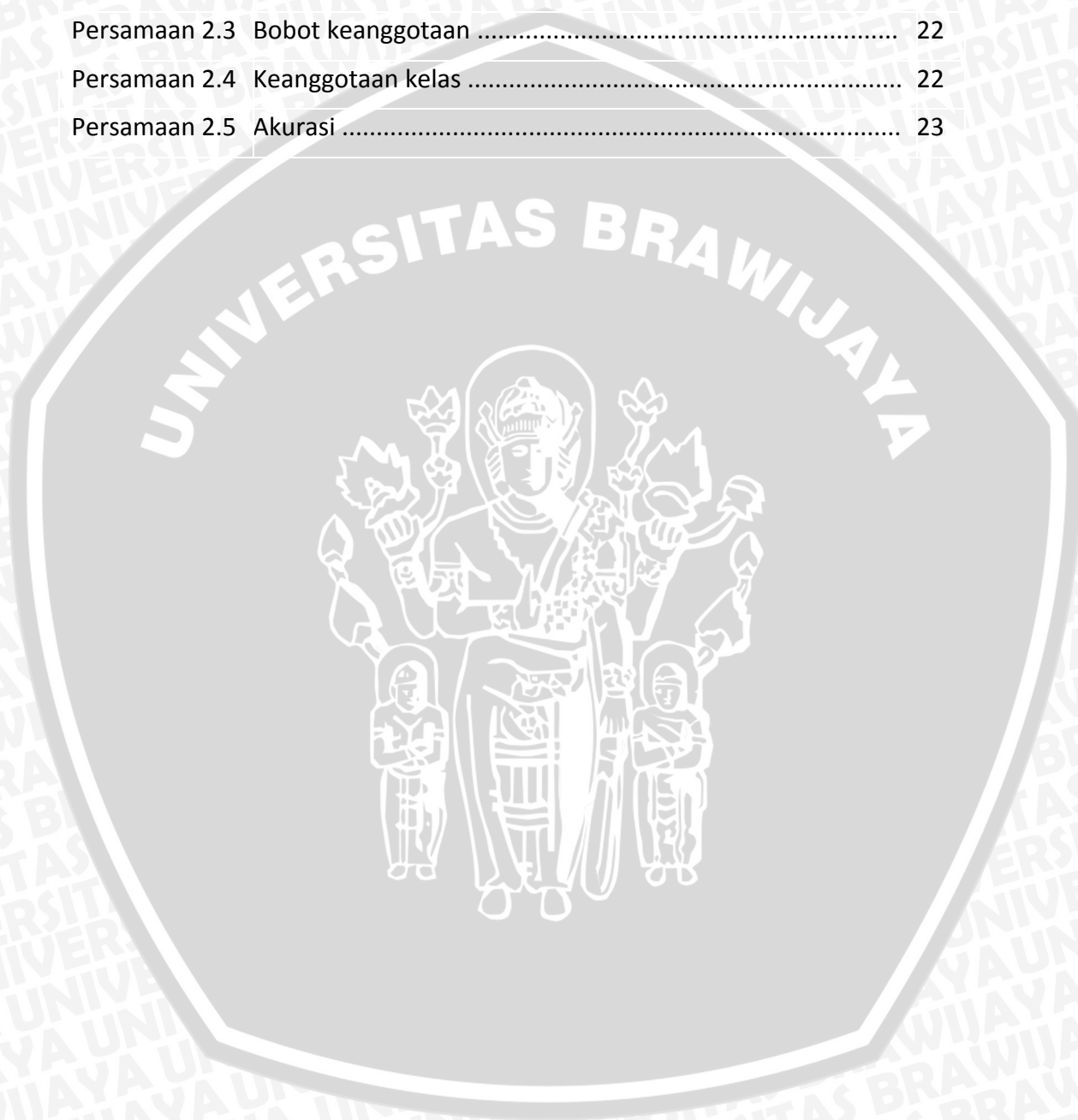


DAFTAR GAMBAR

Gambar 2. 1 <i>Solanum lycopersicum L</i>	8
Gambar 2. 2 Arsitektur sistem pakar	16
Gambar 2. 3 <i>Forward chaining</i>	19
Gambar 2. 4 <i>Backward Chaining</i>	20
Gambar 3. 1 Diagram blok metode penelitian	24
Gambar 3. 2 Arsitektur perancangan sistem	25
Gambar 4. 1 Pohon perancangan	27
Gambar 4. 2 Proses klasifikasi secara umum	29
Gambar 4. 3 Proses normalisasi	30
Gambar 4. 4 Proses KNN	31
Gambar 4. 5 Proses <i>euclidean</i>	32
Gambar 4. 6 Proses FKNN	33
Gambar 4. 7 Kerangka konsep sistem	34
Gambar 4. 8 Antarmuka halaman home	43
Gambar 4. 9 Antarmuka halaman identifikasi	44
Gambar 4. 10 Antarmuka halaman <i>contact</i>	44
Gambar 4. 11 Antarmuka halama <i>login</i>	45
Gambar 4. 12 Antarmuka halaman admin	45
Gambar 4. 13 Antarmuka halaman tambah data	46
Gambar 5. 1 Pohon implementasi	49
Gambar 5. 2 Implementasi Algoritma KNN	53
Gambar 5. 3 Implementasi Algoritma FKNN	56
Gambar 5. 4 Implementasi antarmuka halaman identifikasi	57
Gambar 5. 5 Implementasi antarmuka halaman hasil identifikasi	57
Gambar 5. 6 Implementasi antarmuka halaman login	58
Gambar 5. 7 Implementasi antarmuka halaman admin	59
Gambar 5. 8 Implementasi antarmuka halaman tambah data	59
Gambar 6. 1 Diagram hasil pengujian data seimbang	71
Gambar 6. 2 Diagram hasil pengujian data tidak seimbang	72

DAFTAR PERSAMAAN

Persamaan 2.1	Normalisasi	21
Persamaan 2.2	Jarak <i>euclidean</i>	21
Persamaan 2.3	Bobot keanggotaan	22
Persamaan 2.4	Keanggotaan kelas	22
Persamaan 2.5	Akurasi	23



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Indonesia sebagai salah satu negara beriklim tropis mempunyai potensi dan kesempatan yang cukup besar untuk memanfaatkan peluang usaha dibidang hortikultura. Banyak petani di Indonesia yang memanfaatkan peluang ini karena komoditas hortikultura bernilai ekonomi tinggi. Salah satu usaha dalam bidang hortikultura yaitu budidaya tanaman tomat. Tomat merupakan sayuran buah yang telah dikenal oleh masyarakat sebagai buah segar ataupun bahan olahan makanan.

Tomat merupakan salah satu sayuran yang dapat ditanam hampir di seluruh lahan pertanian Indonesia. Di Indonesia sendiri lebih dari 5.000.000 ha luas lahan ditanami tomat, dengan produksi mencapai 129 juta ton per tahun (Yasa, et al., 2012). Menurut Badan Pusat Statistika, produksi tomat di Indonesia dari tahun 2009 hingga tahun 2013 relatif mengalami peningkatan. Pada tahun 2012-2013 peningkatan produksi tomat mengalami kenaikan sebesar 11,12% atau 13.9179 ton (BPS, 2014). Produksi tomat akan terus ditingkatkan untuk memenuhi permintaan pasar yang cenderung meningkat.

Meningkatkan produksi tomat tidaklah mudah. Dalam melakukan budidaya tomat, petani sering mengalami kendala yang dapat menyebabkan hasil produksi tomat menurun secara kuantitas dan kualitas. Kendala tersebut dapat berupa kondisi alam, hama dan penyakit. Kendala yang paling sering dialami oleh petani adalah adanya gangguan penyakit. Jenis penyakit tanaman tomat yang beraneka ragam membutuhkan penanganan yang berbeda.

Kurangnya pemahaman petani tentang penyakit pada tanaman tomat menyebabkan petani kesulitan dalam menyelesaikan masalah penyakit pada tanaman tomat. Jumlah pakar penyakit tomat yang kurang memadai membuat kebanyakan petani tomat melakukan sendiri identifikasi penyakit tomat dengan pengetahuan yang terbatas. Penanganan penyakit tomat dari petani sendiri menjadi kurang optimal. Untuk membantu petani menangani masalah penyakit tanaman tomat dibutuhkan sebuah sistem yang mampu mengidentifikasi penyakit pada tanaman tomat.

Salah satu bentuk kemajuan teknologi komputer adalah adanya sistem pakar (*expert system*). Sistem pakar merupakan sebuah sistem yang meniru kemampuan pakar ahli dalam menyelesaikan suatu masalah berdasarkan bidang keahlian pakar. Dalam menyelesaikan permasalahan, sistem pakar memerlukan suatu metode untuk mengolah data masukan menjadi informasi yang berguna. Terdapat beberapa metode klasifikasi yang sering digunakan dalam teknik data mining, diantaranya adalah *metode k-nearest neighbor*, *naive bayes decision tree*, *fuzzy*, *neural network* dan *support vector machine (SVM)*.

Metode *Fuzzy K-Nearest Neighbor (FK-NN)* merupakan metode klasifikasi pada *data mining* dengan menggabungkan antara metode *fuzzy* dengan metode *K-Nearest Neighbor (K-NN)*. Metode *FK-NN* ditemukan oleh James M. Keller. Metode ini digunakan untuk mengatasi masalah dalam 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. (Wisdianto, 2013).

Metode *FK-NN* memiliki dua keunggulan utama dari pada algoritma *K-NN*. Pertama, algoritma ini mampu mempertimbangkan sifat yang tidak jelas kelasnya (ambigu) dari tetangga jika ada. Algoritma ini telah dirancang sedemikian rupa agar tetangga yang ambigu tidak memainkan peranan penting dalam klasifikasi saat ini. Keunggulan kedua yaitu sebuah instance akan memiliki derajat nilai keanggotaan pada setiap kelas sehingga akan lebih memberikan kekuatan atau kepercayaan suatu instance berada pada suatu kelas (Putri, et al., 2010).

Berdasarkan uraian latar belakang tersebut, peneliti membuat penelitian dengan judul “Pemodelan Sistem Pakar Untuk Identifikasi Penyakit Pada Tanaman Tomat Menggunakan Metode *Fuzzy K-Nearest Neighbor*”. Sistem yang akan dibuat nantinya *user* dapat memasukkan gejala penyakit yang tampak pada tanaman tomat. Dari gejala penyakit tersebut nilainya akan dikonversikan ke dalam bentuk angka agar bisa diproses dengan metode *fuzzy k-nearest neighbor*. Berdasarkan proses perhitungan akan menghasilkan nilai keanggotaan terbesar yang menunjukkan kelas target. Keluaran dari sistem ini yaitu jenis penyakit tanaman tomat. Sistem ini diharapkan mampu membantu petani mengidentifikasi penyakit tanaman tomat, sehingga dapat melakukan penanganan penyakit tomat secara tepat.

1.2 Rumusan Masalah

Permasalahan yang dapat dirumuskan dari uraian latar belakang tersebut yaitu:

1. Bagaimana merancang pemodelan sistem pakar untuk identifikasi penyakit pada tanaman tomat menggunakan metode *fuzzy k-nearest neighbor*?
2. Bagaimana hasil pengujian pemodelan sistem pakar untuk identifikasi penyakit pada tanaman tomat menggunakan metode *fuzzy k-nearest neighbor* terhadap hasil identifikasi pakar?

1.3 Tujuan

Tujuan yang ingin dicapai dalam penelitian ini yaitu:

1. Merancang pemodelan sistem pakar untuk identifikasi penyakit pada tanaman tomat menggunakan metode *fuzzy k-nearest neighbor*.
2. Menganalisa hasil pengujian pemodelan sistem pakar untuk identifikasi penyakit pada tanaman tomat menggunakan metode *fuzzy k-nearest neighbor* terhadap hasil identifikasi pakar.

1.4 Manfaat

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

1.5 Batasan Masalah

Batasan masalah dalam penelitian ini yaitu:

1. Proses identifikasi berdasarkan 15 gejala yang tampak pada daun, batang, dan buah.
2. Keluaran sistem adalah hasil identifikasi 6 jenis penyakit pada tanaman tomat, yaitu bercak coklat, busuk daun, layu fusarium, layu bakteri, mozaik, dan keriting.
3. Data latih berjumlah 188 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 tomat.

1.6 Sistematika Pembahasan

Untuk mencapai tujuan yang diharapkan, maka sistematika pembahasan dalam skripsi ini disusun sebagai berikut:

1. BAB I Pendahuluan
Bab ini berisi tentang latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, dan sistematika pembahasan.
2. 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 tomat, sistem pakar, metode *Fuzzy K-NN*, dan akurasi.
3. BAB III Metodologi
Bab ini membahas tentang tahapan metode penelitian yang dilakukan yaitu studi literatur, pengumpulan data, pengolahan data, perancangan sistem, implementasi sistem, serta analisis evaluasi dan kesimpulan.
4. BAB IV Perancangan
Bab ini membahas tentang analisa kebutuhan dari sistem untuk identifikasi penyakit tanaman tomat dan kemudian merancang hal-hal yang berhubungan dengan analisa tersebut.
5. BAB V Implementasi
Bab ini membahas tentang hasil perancangan dari analisis kebutuhan dan implementasi pemodelan sistem pakar identifikasi penyakit tanaman tomat menggunakan metode *fuzzy k-nearest neighbor*.

6. BAB VI Pengujian

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

7. BAB VII Penutup

Bab ini memuat kesimpulan yang diperoleh dari pembuatan dan pengujian pemodelan sistem pakar identifikasi penyakit tanaman tomat 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. Metode yang dibandingkan yaitu *Fuzzy K-Nearest Neighbor* dengan objek yang berbeda. 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 tomat, sistem pakar, metode *k-nearest neighbor*, metode *fuzzy k-nearest neighbor*, dan akurasi.

2.1 Kajian Pustaka

Kajian pustaka pada penelitian ini adalah membandingkan penelitian ini dengan penelitian sebelumnya yang berjudul “Sistem Pakar Tuberculosis Pada Anak Berdasarkan *Scoring System* Dengan Metode Fuzzy K-NN” dan “Perbandingan *K-Nearest Neighbor* dan *Fuzzy K-Nearest Neighbor* pada Diagnosis Penyakit Diabetes Melitus”. Tabel 2.1 merupakan tabel kajian pustaka yang membandingkan penelitian sebelumnya dengan penelitian yang diusulkan.

Tabel 2. 1 Kajian pustaka

No.	Judul	Objek	Metode	Hasil
		Input dan Parameter	Proses	Output dan Hasil
1.	Sistem Pakar Tuberculosis Pada Anak Berdasarkan <i>Scoring System</i> Dengan Metode Fuzzy K-NN	Input berupa gejala penyakit TB pada anak. Parameternya yaitu kontak TB, uji <i>tuberkulin</i> , keadaan gizi, demam, batuk, pembesaran kelenjar, pembengkakan tulang, foto <i>rontgen</i> dada, dan tes <i>mantoux</i> .	Membuat skor gejala TB, menghitung derajat keanggotaan, menentukan parameter k, menghitung jarak <i>euclid</i> , mengurutkan objek kedalam kelompok yang mempunyai jarak <i>euclid</i> terkecil, mengumpulkan kategori y. Dengan kategori <i>nearest neighbor</i> yang paling mayoritas maka dapat diprediksi nilai <i>query</i>	Output berupa (-) TB, TB laten atau (+) TB. Hasil penellitian yaitu memiliki akurasi tertinggi pada data latih ke 100 dan 150 dengan rata-rata 98,2% dan 97,8%.

			<i>instance</i> yang telah dihitung.	
2.	Perbandingan <i>K-Nearest Neighbor</i> dan <i>Fuzzy K-Nearest Neighbor</i> pada Diagnosis Penyakit <i>Diabetes Melitus</i>	<i>Input</i> berupa kondisi pasien. Parameternya adalah 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.	Proses KNN: normalisasi data latih dan data uji, menghitung jarak <i>record</i> data uji terhadap data latih, mengurutkan data berdasarkan jarak terkecil, menentukan k <i>record</i> terdekat, dipilih kelas yang paling dominan. Proses FKNN: sama seperti langkah KNN, setelah penentuan k terdekat, mencari nilai keanggotaan untuk tiap kelas j, kemudian menghitung nilai keanggotaan data uji pada masing-masing kelas. Dipilih nilai keanggotaan terbesar	<i>Output</i> berupa positif DM (1) dan negatif DM (0). Hasil penelitian yaitu memiliki hasil akurasi tertinggi 98% untuk FKNN dan 96% untuk KNN dengan data latih 80, 130, 180, dan 230 serta 50 data uji.
3.	Pemodelan Sistem Pakar Untuk Identifikasi Penyakit Pada Tanaman Tomat Menggunakan Metode <i>Fuzzy K-Nearest Neighbor</i>	<i>Input</i> berupa kondisi fisik tanaman tomat. Parameternya adalah batang, daun, dan buah.	FKNN: normalisasi data latih dan data uji, menghitung jarak <i>record</i> data uji terhadap data latih, mengurutkan data berdasarkan jarak terkecil, menentukan k <i>record</i> terdekat, mencari nilai keanggotaan untuk tiap kelas j, kemudian	<i>Output</i> berupa diagnosis penyakit busuk daun, bercak coklat, layu fusarium, layu bakteri, mosaik, dan keriting.

			menghitung nilai keanggotaan data uji pada masing-masing kelas. Dipilih nilai keanggotaan terbesar	
--	--	--	--	--

Sumber: (Anggraeni, 2014), (Meristika, 2013)

Persamaan dari kedua penelitian sebelumnya adalah metode yang digunakan yaitu *fuzzy K-nearest neighbor*. Yang membedakan antara kedua penelitian adalah proses dan objek yang digunakan. Objek yang digunakan pada penelitian pertama adalah *Tuberculosis* pada anak dan pada penelitian kedua adalah *Diabetes Melitus*. Proses yang membedakan diantara kedua metode adalah proses *fuzzy*. Pada penelitian pertama, proses *fuzzy* digunakan dalam mencari keanggotaan dari data *input*. Sedangkan pada penelitian kedua, proses *fuzzy* untuk mencari keanggotaan dari *output*. Pada penelitian yang diajukan, peneliti menggunakan objek penyakit tanaman tomat dan menggunakan metode *fuzzy k-nearest neighbor* dengan proses *fuzzy* untuk mencari keanggotaan dari *output*.

2.2 Tanaman Tomat

2.2.1 Sejarah Penyebaran Tanaman Tomat

Beberapa sumber mengatakan tomat berasal dari Benua Amerika, terutama kawasan Amerika Tengah dan Amerika Selatan. Tanaman tomat ditemukan di sekitar pegunungan Andes dan Brazilia, kemudian menyebar ke Meksiko dan negara lainnya. Tomat menyebar di Amerika Selatan, dan juga beberapa negara di Eropa, Afrika, dan Asia. Pada tahun 1523, benih tanaman tomat dibawa oleh pedagang-pedagang Spanyol dari Benua Amerika ke Eropa. Pada tahun 1571, tanaman tomat mulai dikenalkan di Filipina. Tanaman tomat kemudian menyebar ke negara lainnya di Asia. Pada tahun 1600 tanaman tomat mulai menyebar di kawasan Eropa. Sedangkan tahun 1811 tanaman tomat baru masuk ke Indonesia, dan mulai tahun 1961 pengembangan budidaya tanaman tomat mendapat prioritas perhatian di Indonesia (Tim, 2013).

2.2.2 Taksonomi Tanaman Tomat

Dalam taksonomi tumbuhan, tanaman tomat diklasifikasikan sebagai berikut:

Kingdom : Plantae
 Devisi : Spermatophyta
 Subdivisi : Angiospermae
 Kelas : Dicotyledoneae
 Subkelas : Metachlamidae
 Ordo : Tubiflorae

Famili : Solanaceae
 Genus : Lycopersicon
 Spesies : *Solanum lycopersicum* L.

Karakteristik tanaman tomat memiliki akar tunggang, akar cabang, serta akar serabut yang berwarna keputih-putihan. Batang tanaman berbentuk bulat, bercabang banyak, mudah patah, dan cukup rimbun, sehingga secara keseluruhan berbentuk perdu. Daun tomat berwarna hijau, berbentuk oval, bergerigi, mempunyai celah yang menyirip, dan berbulu. Bunga tanaman tomat berwarna kuning dan tersusun dalam dompolan. Buah tomat matang berwarna merah dan berwarna hijau saat masih muda. Bentuk buah tomat beragam: lonjong, oval, pipih, meruncing, dan bulat (Sagala, 2009). Gambar 2.1 merupakan gambar tanaman tomat.



Gambar 2. 1 *Solanum lycopersicum* L

2.2.3 Manfaat Tomat

Pada mulanya tomat dikenal sebagai tanaman liar yang belum diketahui manfaatnya, tetapi di Peru sudah mulai dijadikan sebagai bahan makanan. Penggunaan tomat sebagai bahan makanan secara besar-besaran mulai dilakukan di Eropa, terutama dijadikan sebagai bumbu masak. Kini tomat digunakan untuk bumbu masakan sehari-hari, industri saus tomat, diawetkan dalam kaleng, dijadikan minuman, dimakan secara langsung, serta berbagai macam bahan makanan bergizi tinggi lainnya.

Tomat memiliki komposisi zat yang cukup lengkap dan baik untuk kesehatan tubuh. Yang paling menonjol diantara komposisi tersebut adalah vitamin A dan vitamin C. Tabel 2.2 menjelaskan kandungan gizi setiap 100 gr buah tomat segar.

Tabel 2. 2 Kandungan nutrisi tomat

Nutrien	Kandungan per 100 gr
Air (g)	93,76
Energi (kkal)	21
Protein (g)	0,85
Total lemak (g)	0,33

Karbohidrat (g)	4,64
Serat (g)	1,1
Abu (g)	0,42
Kalsium (mg)	5
Zat besi (mg)	0,45
Magnesium (mg)	11
Fosfor (mg)	24
Kalium (mg)	222
Natrium (mg)	9
Seng (mg)	0,09
Tembaga (mg)	0,074
Mangan (mg)	0,105
Selenium (mg)	0,4
Vitamin C (mg)	19,1
Tiamin (mg)	0,059
Riboflavin (mg)	0,048
Niasin (mg)	0,628
Asam pantotenat (mg)	0,247
Vitamin B6 (mg)	0,080
Vitamin A (IU)	623
Tokoferol (mg)	0,34
Triptofan (g)	0,006
Treonin (g)	0,021
Isoleusin (g)	0,020
Leusin (g)	0,031
Lisin (g)	0,031
Metionin (g)	0,007
Kistin (g)	0,011
Fenilalanin (g)	0,022
Tirosin (g)	0,015
Valin (g)	0,022
Arginin (g)	0,021
Histidin (g)	0,013

Alanin (g)	0,024
Asam aspartat (g)	0,118
Asam glutamat (g)	0,313
Glisin (g)	0,021
Prolin (g)	0,016
Serin (g)	0,023
Asam lemak jenuh (g)	0,045
Asam lemak tak jenuh tunggal (g)	0,050
Asam lemak tak jenuh ganda (g)	0,135

Sumber: (Kailaku, et al., 2007)

Vitamin-vitamin yang terkandung pada buah tomat sangat diperlukan tubuh untuk pertumbuhan dan kesehatan. Vitamin C berguna untuk mencegah sariawan, memelihara kesehatan gigi dan gusi. Berbagai penelitian menunjukkan bahwa likopen dalam tomat sebagai antioksidan yang bermanfaat untuk menurunkan resiko penyakit kanker prostat, kanker payudara, kanker serviks, kanker ovarium, penyakit kardiovaskuler, dan preeklamsia, serta mencegah osteoporosis.

2.2.4 Syarat Tumbuh Tanaman Tomat

Syarat tumbuh tanaman tomat adalah sebagai berikut:

1. Iklim

Tanaman tomat dapat tumbuh di daerah tropis maupun sub-tropis. Curah hujan yang cocok untuk tanaman ialah sekitar 750-1.250 mm/tahun. Keadaan tersebut berhubungan erat dengan ketersediaan air tanah bagi tanaman, terutama di daerah yang tidak terdapat irigasi teknis. Curah hujan yang tinggi (banyak hujan) juga dapat menghambat persarian.

Kekurangan sinar matahari dapat menyebabkan tanaman tomat mudah terserang penyakit, baik parasit maupun non-parasit. Sinar matahari berintensitas tinggi akan menghasilkan vitamin C dan karoten (provitamin A) yang lebih tinggi. Penyerapan unsur hara yang maksimal oleh tanaman tomat akan dicapai apabila pencahayaan selama 12-14 jam/hari, sedangkan intensitas cahaya yang cocok adalah 0,25 mj/m² per jam.

2. Suhu

Kisaran temperatur yang baik untuk pertumbuhan tomat ialah antara 20-27°C. Jika temperatur berada lebih dari 30°C atau kurang dari 10°C, maka akan mengakibatkan terhambatnya pembentukan buah tomat. Di negara-negara yang mempunyai empat musim, biasanya menggunakan pemanas (heater) untuk mengatur udara ketika musim dingin, udara panas dari heater disalurkan ke dalam green house melalui saluran fleksibel warna putih.

3. Kelembaban

Kelembaban relatif yang baik untuk pertumbuhan tanaman tomat ialah 25 %. Keadaan ini akan merangsang pertumbuhan untuk tanaman tomat yang masih muda karena asimilasi CO₂ menjadi lebih baik melalui stomata yang membuka lebih banyak. Akan tetapi, kelembaban relatif yang tinggi juga dapat merangsang mikroorganisme pengganggu tanaman.

4. Media Tanam

Secara umum, tanaman tomat dapat ditanam di segala jenis tanah, mulai dari tanah pasir sampai tanah lempung berpasir yang subur, gembur, berporus, banyak mengandung bahan organik dan unsur hara, serta mudah merembeskan air. Tingkat kemasaman tanah (pH) yang sesuai untuk budidaya tomat ialah berkisar 5,0-7,0. Akar tanaman tomat rentan terhadap kekurangan oksigen. Oleh karena itu, tanaman tomat tidak boleh tergenangi oleh air. Dalam pembudidayaan tanaman tomat, sebaiknya dipilih lokasi yang topografi tanahnya datar sehingga tidak perlu dibuat teras-teras dan tanggul.

5. Ketinggian Tempat

Tanaman tomat dapat tumbuh di berbagai ketinggian tempat, baik di dataran tinggi maupun di dataran rendah, tergantung varietasnya. Tanaman tomat yang sesuai untuk ditanam di dataran tinggi, misalnya varietas Kada, sedangkan varietas yang sesuai ditanam di dataran rendah, misalnya varietas Intan, varietas Ratna, varietas LV, dan varietas CLN. Selain itu, ada varietas tanaman tomat yang cocok ditanam di dataran rendah maupun di dataran tinggi, antara lain varietas tomat GH 2, varietas tomat GH 4, varietas Berlian, dan varietas Mutiara (Leovini, 2012).

2.2.5 Penyakit Tanaman Tomat

Penyakit tanaman merupakan gangguan pada tanaman yang disebabkan oleh mikroorganisme. Mikroorganisme tersebut adalah virus, bakteri, protozoa, jamur, dan cacing nematoda. Mikroorganisme tersebut dapat menyerang organ tumbuhan seperti pada akar, buah, batang, maupun daun. Tanaman yang terserang suatu penyakit akan terhambat pertumbuhannya, bahkan bisa mati. Beberapa penyakit yang sering menyerang tanaman tomat adalah sebagai berikut (Semangun, 1989):

1. Penyakit Busuk Daun

Penyakit busuk daun pada tanaman tomat disebabkan oleh jamur *Phytophthora infestans*. Jamur ini merupakan jamur yang sama dengan yang menyebabkan penyakit busuk daun pada tanaman kentang. Penyakit busuk daun biasanya berjangkit pada musim hujan dan di daerah dataran tinggi. Penyakit ini dapat menyerang pada semua fase pertumbuhan tanaman.

Gejala yang tampak pada tanaman tomat yang serang penyakit busuk daun yaitu pada daun terdapat bercak berwarna kuning pucat ke warna hijau. Pada sisi bawah daun terdapat fruktifikasi jamur yang berwarna putih seperti

beledu. Sedangkan pada buah terdapat bercak berwarna coklat tua, agak keras dan berkerut. Bercak mempunyai batas yang cukup tegas dan terkadang bercak berbentuk cincin.

Fungisida dapat dianjurkan sebagai alternatif untuk mengendalikan jamur *phytophthora infestans* pada tomat, antara lain: fungisida protektan Kocide 54^{WDG} dan fungisida sistemik Starmyl 25^{WP}. Dalam pemakaian fungisida Kocide 54^{WDG} dan Starmyl 25^{WP} dapat digunakan secara bergantian maupun bersama-sama, karena kedua fungisida ini teruji efektifitasnya dan tidak terjadi reaksi yang bersifat saling melemahkan.

2. Penyakit Bercak Coklat

Penyakit bercak coklat disebabkan oleh jamur *Alternaria solani* Sor. Jamur *A. solani* membentuk racun yang disebut *alternaric acid*. Jamur *A. solani* mempertahankan diri dari musim ke musim pada tanaman sakit, pada biji, atau pada sisa-sisa tanaman yang sakit. Jamur ini juga yang menyebabkan penyakit bercak kering pada tanaman kentang.

Gejala yang tampak pada tanaman tomat yang terserang penyakit bercak coklat antara lain: pada daun terdapat bercak berwarna coklat gelap yang melingkar berpusat. Jika bercak pada daun sudah banyak, daun akan cepat menjadi tua, layu, atau gugur sebelum waktunya. Terdapat bercak gelap pada batang. Jika infeksi terjadi di sekitar percabangan, cabang menjadi mudah patah. Pada buah terdapat bercak berwarna coklat gelap dan tampak mengendap berlekuk.

Untuk mencegah jamur yang terbawa oleh biji dapat dilakukan disinfestasi biji. Agar tidak terjadi banyak infeksi pada bibit, pembibitan jangan terlalu lembab atau rapat. Fungisida tembaga dan penyemprotan Difolatan kurang efektif dalam pengendalian penyakit bercak coklat. Lebih baik menggunakan fungisida Karbamat untuk mengendalikan penyakit bercak coklat.

3. Penyakit Layu Fusarium

Penyakit layu fusarium disebabkan oleh jamur *Fusarium oxysporum* (Schlecht.) f.sp. Jamur ini dapat bertahan lama dalam tanah dan sukar dihilangkan. Jamur *Fusarium oxysporum* menginfeksi pada akar, terutama melalui luka, kemudian menetap dan berkembang di berkas pembuluh. perkembangan Jamur dapat tersebar karena pengangkutan bibit, terbawa oleh alat pertanian, atau tanah yang terbawa oleh angin atau air.

Gejala yang tampak pada tanaman tomat yang terserang penyakit layu fusarium yaitu daun sebelah atas menjadi pucat dan daun sebelah bawah menguning. Kemudian diikuti dengan tangkai menjadi merunduk. Pada tanaman yang masih muda penyakit dapat menyebabkan tanaman mati secara mendadak. Sedangkan tanaman dewasa yang terinfeksi sering kali dapat bertahan dan membentuk buah, tetapi hasilnya buah sangat sedikit dan kecil-kecil. Tanaman juga menjadi tumbuh kerdil.

Untuk mengendalikan penyakit layu fusarium dapat dilakukan dengan mengusahkan agar tanaman jangan sampai terjadi luka. Benih tanaman didesinfeksi dengan air panas bersuhu 55° C selama 10-17 menit. Tanaman yang sudah terserang penyakit sebaiknya dicabut kemudian dimusnahkan dengan cara membakarnya.

4. Penyakit Layu Bakteri

Penyakit layu bakteri disebabkan oleh bakteri *Pseudomonas solanacearum* (E.F.Sm.). Bakteri ini mengadakan infeksi melalui luka-luka, termasuk luka yang disebabkan oleh nematoda. Penyakit layu bakteri pada tomat dibantu oleh suhu yang relatif tinggi, sehingga sebagian besar menyerang tanaman tomat didataran rendah. Intensitas penyakit sangat dipengaruhi oleh tanaman sebelumnya. Pupuk kandang yang baru (belum masak) dapat membawa bakteri layu. Bakteri akan berkembang lebih baik ditanah yang alkalis.

Gejala yang tampak pada tanaman tomat yang terserang penyakit layu bakteri yaitu beberapa daun muda yang layu dan daun-daun di sebelah bawah menguning. Batang tanaman cenderung membentuk lebih banyak akar adventif. Pada stadium penyakit yang lanjut, akan keluar lendir berwarna putih susu dari berkas pembuluh apabila batang dipotong. Lendir tersebut merupakan massa bakteri. Adanya massa bakteri yang dapat membedakan penyakit layu bakteri dengan penyakit layu fusarium.

Penanggulangan penyakit layu bakteri dapat dilakukan dengan cara pergiliran tanaman dengan tanaman yang tidak dapat menjadi inang *P. solanacearum*, misalnya tanaman familia kubis-kubisan (*Cruciferae*). Menggunakan persemaian bebas dari patogen dengan melakukan fungigasi persemaian dan persemaian disiram dengan air panas. Perlu diperhatikan bahwa ada juga tanaman lain yang mudah terserang oleh penyakit ini, misalnya terong, kentang, dan cabai. Untuk tanaman yang sudah terserang, segera dicabut dan dimusnahkan dengan cara dibakar agar tidak menular pada tanaman tomat.

5. Penyakit Mozaik

Penyakit mozaik disebabkan oleh virus. Virus yang sering menyerang tanaman tomat adalah virus *Marmor tabaci* Holmes dan virus *Marmor cucumeris* var. *Vulgare* Holmes. Kedua virus tersebut merupakan virus yang sering menyerang tanaman tembakau. virus menular dari tanaman ke tanaman lain secara mekanis. Virus mozaik tembakau meluas dengan cepat pada tanaman tomat, sehingga sering pada akhir musim tanam tidak ada tanaman yang tidak terinfeksi.

Gejala yang tampak pada tanaman yang terserang penyakit mozaik yaitu adanya bercak hijau muda atau kuning yang tidak teratur. Daun menjadi berkerut dan sukar berkembang. Daun cenderung menjadi terpuntir. Buah menjadi lebih kecil dari biasanya dan terkadang terdapat bercak-bercak nekrotik. Pada batang terdapat garis-garis hitam yang terdiri dari jaringan mati.

Pengendalian penyakit mozaik dapat dilakukan dengan mengindarkan persemaian tanaman tomat dari gulma dan kutu daun. Tanaman yang sakit di

persemaian dan tanaman muda dicabut agar tidak menjadi sumber infeksi. Mencuci tangan dengan sabun atau deterjen setelah memegang tanaman tomat atau gulma yang mungkin mengandung virus. Mengusahkan agar tidak menanam tanaman yang menjadi sumber virus di dekat lahan tanaman tomat, seperti tanaman mentimun, markisa, cabai, atau kacang panjang.

6. Penyakit Keriting

Penyakit keriting pada tanaman tomat disebabkan oleh virus *Ruga tabaci* Holmes. Virus ini merupakan virus yang sama dengan penyebab penyakit keriting pada tembakau. Virus penyebab penyakit ini tidak dapat ditularkan secara mekanis. Virus disebarkan oleh lalat putih, *Bemisia tabaci* Genn. yang termasuk familia *Aleyrodidae*.

Gejala yang tampak pada tanaman tomat yang terserang penyakit keriting yaitu daun menjadi berkerut, terpuntir, dan menggulung ke atas. Tulang daun membengkak dan berwarna hijau tua. Warna daun menjadi tidak merata. Tanaman memiliki cabang yang banyak dan menghasilkan sedikit buah. Bahkan terkadang tanaman tidak menghasilkan buah sama sekali.

Pengendalian penyakit keriting dapat dilakukan dengan mengusahkan agar di sekitar persemaian tanaman tomat tidak terdapat gulma atau tanaman lain yang dapat menjadi sumber virus.

2.3 Pemodelan

Model adalah representasi dari suatu objek, benda atau ide dalam bentuk yang disederhanakan. Model dapat merupakan tiruan dari suatu benda, sistem atau kejadian yang sesungguhnya yang hanya berisi informasi-informasi yang dianggap penting untuk ditelaah. Pemodelan sistem merupakan kumpulan aktivitas dalam pembuatan model dimana model merupakan perwakilan atau abstraksi dari sebuah obyek atau situasi aktual suatu penyederhanaan dari suatu realitas yang kompleks (Anshori, 2012).

2.3.1 Jenis-jenis Model

Jenis-jenis model dalam sebuah sistem adalah sebagai berikut (Anshori, 2012):

1. Model Ikonik

Model ikonik merupakan perwakilan fisik dari beberapa hal baik yang berdimensi dua (foto, petak, cetak biru) atau yang berdimensi tiga (*prototype* alat atau mesin).

2. Model Analog

Model yang mewakili situasi dinamik, yaitu keadaan berubah menurut waktu serta mempunyai kemampuan mengetengahkan karakteristik kejadian yang dikaji, banyak berkesesuaian dengan penjabaran hubungan kuantitatif antara sifat dan kelas-kelas yang berbeda. Contoh: kurva permintaan, diagram alir.

3. Model Simbolik

Suatu sistem lebih mengarah pada model simbolik yang merupakan perwakilan dari realitas yang sedang dikaji. Format model simbolik dapat berupa bentuk angka, simbol, dan rumus. Jenis model simbolik yang umum dipakai adalah suatu persamaan (*equation*). Model Simbolik ini terdiri atas model statik dan model dinamis. Model statik memberikan informasi tentang peubah-peubah model hanya pada titik tunggal dari waktu. Model dinamik mampu menelusuri jalur waktu dari peubah-peubah model, lebih sulit dan mahal pembuatannya namun memberikan kekuatan yang lebih tinggi pada analisis dunia nyata. Model yang menggunakan simbol-simbol untuk menggambarkan suatu sistem misalnya diagram alir proses, diagram tata letak fasilitas, diagram organisasi, dll. Model ini cepat pembuatannya dan mudah dimengerti orang lain.

4. Model Analitik

Merupakan model matematik yang menghasilkan solusi kuantitatif. Model analitik dapat memberikan hasil optimum tanpa proses coba-coba (*trial and error*). Model analitik umumnya bersifat statis, *prescriptive*, deterministik atau probabilistik.

5. Model Simulasi

Merupakan model yang menggambarkan hubungan sebab dan akibat (*cause and effect relationship*) dalam sebuah sistem pada model komputer yang mampu menggambarkan perilaku yang mungkin terjadi pada sistem nyatanya. Model simulasi dipakai sebagai alat untuk mengetahui apa yang terjadi jika (*what if*) satu atau lebih komponen atau variabelnya diubah. Seperti halnya model analitik, model simulasi pun memberikan output kuantitatif. Kelebihannya adalah dapat mengatasi kompleksitas fluktuasi statistis dan interdependensi. Model simulasi sering sangat sulit dibuat/dikonstruksi dan lebih bersifat deskriptif daripada preskriptif.

2.4 Sistem Pakar

2.4.1 Pengertian Sistem Pakar

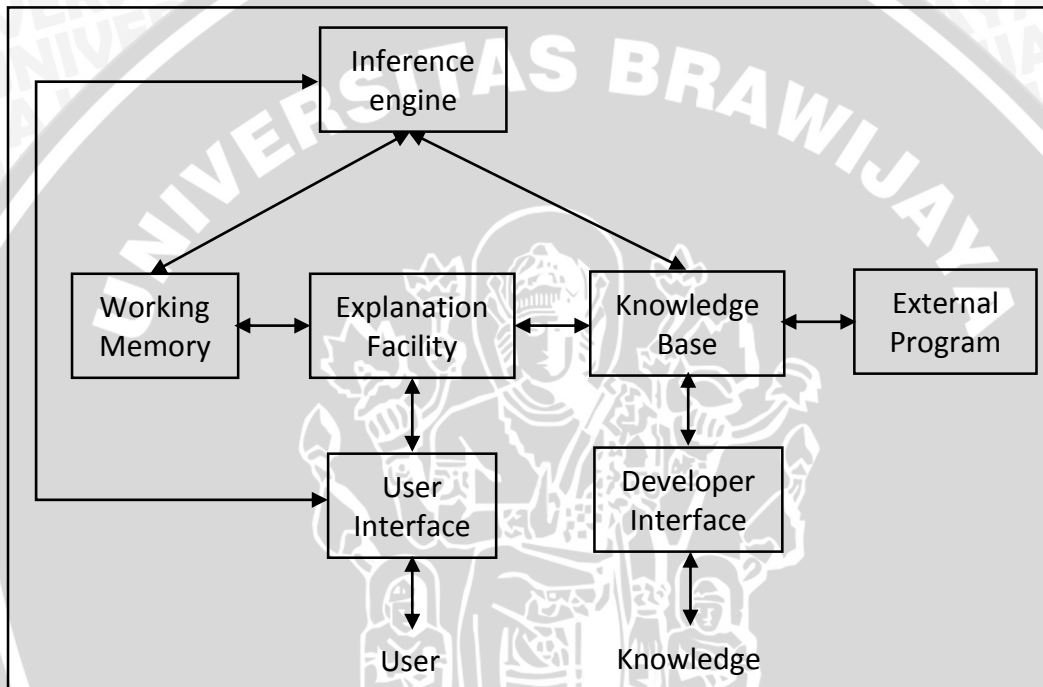
Sistem pakar (*expert system*) merupakan suatu sistem yang berusaha mengadopsi pengetahuan pakar ke dalam komputer, agar komputer tersebut dapat menyelesaikan masalah seperti yang biasa dilakukan oleh pakar. Ada beberapa pengertian sistem pakar menurut ahli, antara lain (Fadli, 2010):

- a. Menurut Durkin: Sistem pakar adalah suatu program komputer yang dirancang untuk memodelkan kemampuan penyelesaian masalah yang dilakukan seorang pakar.
- b. Menurut Ignizio: Sistem Pakar adalah suatu model dan prosedur yang berkaitan, dalam suatu domain tertentu, yang mana tingkat keahliannya dapat dibandingkan dengan keahlian seorang pakar.

- c. Menurut Giarratano dan Riley: Sistem pakar adalah suatu sistem komputer yang bisa menyamai atau meniru kemampuan seorang pakar. Arsitektur Sistem Pakar.

2.4.2 Arsitektur Sistem Pakar

Untuk membangun sistem pakar yang baik, suatu sistem pakar harus memiliki tiga komponen utama, yaitu: *knowledge base*, *working memory* dan *inference engine*. Selain ketiga komponen utama tersebut, ada beberapa komponen pendukung yaitu *user interface*, *developer interface*, *explanation facility*, dan *external programs*. Gambar arsitektur sistem pakar ditunjukkan pada Gambar 2.2 (Handojo & Irawan, 2004).



Gambar 2. 2 Arsitektur sistem pakar

Keterangan dari Gambar 2.2 adalah sebagai berikut:

1. *Knowledge base* adalah representasi pengetahuan dari pakar atau sumber-sumber pengetahuan lainnya yang diperlukan untuk memahami, memformulasi dan memecahkan masalah. *Knowledge base* ini terdiri dari dua elemen dasar, yaitu fakta dan *rules*.
2. *Inference engine* merupakan otak dari sistem pakar yang mengandung mekanisme fungsi berpikir dan pola-pola penalaran sistem yang digunakan oleh seorang pakar. Mekanisme ini yang menganalisis suatu masalah tertentu dan kemudian mencari solusi atau kesimpulan yang terbaik.
3. *Working memory* merupakan tempat penyimpanan fakta-fakta yang diketahui dari hasil menjawab pertanyaan. Fakta-fakta inilah yang nantinya akan diolah oleh mesin inferensi berdasarkan pengetahuan untuk menentukan suatu keputusan pemecahan masalah.

4. *User/developer interface*. Semua *software* pengembangan sistem pakar memberikan *interface* yang berbeda bagi *user* dan *developer*. *User* akan berhadapan dengan tampilan yang sederhana dan mudah sedangkan *developer* akan berhadapan dengan editor dan *source code* waktu mengembangkan program.
5. *Explanation facility* memberikan penjelasan kepada *user* untuk mengetahui alasan sehingga diberikan sebuah solusi.
6. *External programs*. Berbagai program seperti *database*, *spreadsheets*, algoritma, dan lainnya yang berfungsi untuk mendukung sistem.

2.4.3 Basis Pengetahuan

Basis pengetahuan berisi pengetahuan-pengetahuan dalam penyelesaian masalah, tentu saja di dalam domain tertentu. Ada 2 bentuk pendekatan basis pengetahuan yang sangat umum digunakan, yaitu (Fadli, 2010):

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

Pada penalaran berbasis aturan, pengetahuan direpresentasikan dengan menggunakan aturan berbentuk: IF-THEN. Bentuk ini digunakan apabila kita memiliki sejumlah pengetahuan pakar pada suatu permasalahan tertentu, dan si pakar dapat menyelesaikan masalah tersebut secara berurutan. Bentuk ini juga digunakan apabila dibutuhkan penjelasan tentang jejak (langkah-langkah) pencapaian solusi.

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

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

2.4.4 Representasi Pengetahuan

Representasi pengetahuan merupakan metode yang digunakan untuk mengkodekan pengetahuan dalam sebuah sistem pakar yang berbasis pengetahuan. Representasi dimaksudkan untuk menangkap sifat-sifat penting problema dan membuat informasi itu dapat diakses oleh prosedur pemecahan problema. Bahasa representasi harus dapat membuat seorang programmer mampu mengekspresikan pengetahuan yang diperlukan untuk mendapatkan solusi problema, dapat diterjemahkan ke dalam bahasa pemrograman dan dapat disimpan. Harus dirancang agar fakta-fakta dan pengetahuan lainnya yang terkandung didalamnya bisa digunakan untuk penalaran.

Pengetahuan dapat direpresentasikan dalam bentuk yang sederhana atau kompleks, tergantung dari masalahnya. Beberapa model representasi pengetahuan yang penting, adalah (Triyogo & Saputra, 2011):

1. Logika

Terdapat dua logika dalam model representasi pengetahuan, yaitu:

- a. Logika Proposisional (*Propositional Logic*) adalah suatu model untuk mendeklarasikan suatu fakta. Lambang-lambang proposisional menunjukkan proposisi atau pernyataan tentang segala sesuatu yang dapat benar atau salah. Lambang-lambang kalkulus proposisional antara lain: lambang pernyataan proposisional (P,Q,R,S,T,...atau disebut sebagai atom-atom), lambang kebenaran (*True/False*), lambang penghubung ($\wedge$, $\vee$, $\sim$, $\rightarrow$, $\leftrightarrow$, $\equiv$).
- b. Logika Predikat (Kalkulus Predikat) adalah suatu logika canggih yang menggunakan konsep dan kaidah proposional yang sama, serta memberi tambahan kemampuan untuk merepresentasikan pengetahuan dengan sangat cermat dan rinci. Kalkulus predikat memungkinkan kita bisa memecahkan statemen ke dalam bagian komponen, yang disebut objek, karakteristik objek, atau beberapa keterangan objek. Suatu proposisi atau premis dibagi menjadi dua bagian, yaitu argumen (atau objek) dan predikat (keterangan).

Contoh: PREDIKAT(individu[objek]1, individu[objek] 2)

Mobil berada dalam garasi. Dinyatakan menjadi:

Di dalam(mobil, garasi)

Di dalam = produk (keterangan)

Mobil = Argumen(objek)

Garasi = Argumen(objek).

2. Jaringan Semantik (*semantic nets*)

Jaringan semantik merupakan pengetahuan secara grafis yang menunjukkan hubungan antar berbagai objek. Komponen dasar untuk merepresentasikan pengetahuan dalam bentuk jaringan semantik adalah simpul (*node*) dan penghubung (*link*). Simpul merepresentasikan obyek, konsep, atau situasi. Penghubung menghubungkan antar simpul. Penghubung digambarkan dengan panah berarah dan diberi label untuk menyatakan hubungan yang direpresentasikan.

3. *Object-Attribute-Value triplets* (OAV)

Object dapat berupa bentuk fisik atau konsep. Atribut adalah karakteristik atau sifat dari *object* tersebut. *Values* (Nilai) adalah besaran/nilai/takaran spesifik dari atribut tersebut pada situasi tertentu. Dapat berupa numerik, string atau boolean. Sebuah *object* bisa memiliki beberapa atribut, bisa disebut OAV Multiatribut. Sebuah atribut dapat dianggap sebagai suatu *object* baru dan memiliki atribut sendiri.

2.4.5 Mesin Inferensi

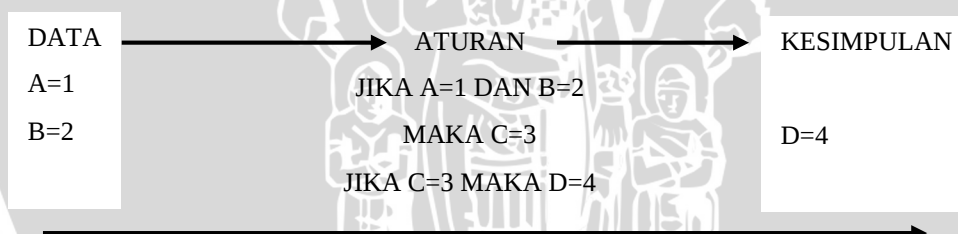
Mesin inferensi merupakan proses yang digunakan dalam sistem pakar untuk menghasilkan informasi baru dari informasi yang telah diketahui. Dalam sistem pakar, proses inferensi dilakukan dalam suatu modul yang disebut *Inference Engine* (Mesin inferensi). Ketika representasi pengetahuan (RP) pada bagian *knowledge base* telah lengkap, atau paling tidak telah berada pada level yang cukup akurat, maka RP tersebut telah siap digunakan. *Inference engine* merupakan modul yang berisi program tentang bagaimana mengendalikan proses *reasoning*.

Ada 3 metode inferensi dalam sistem pakar (Fadli, 2010), (Swanida & Bambang, 2011), yaitu :

1. Forward Chaining

Forward chaining menggunakan himpunan aturan kondisi-aksi, data digunakan untuk menentukan aturan mana yang akan dijalankan, kemudian aturan tersebut dijalankan. Mungkin proses menambahkan data ke memori kerja. Proses diulang sampai ditemukan suatu hasil (Wilson, 1998). Metode *forward chaining* cocok digunakan untuk menangani masalah pengendalian (*controlling*) dan peramalan (*prognosis*).

Pencocokan fakta atau pernyataan dimulai dari bagian sebelah kiri (IF dulu). Dengan kata lain, penalaran dimulai dari fakta terlebih dahulu untuk menguji kebenaran hipotesis. Bagaimana cara kerja metode inferensi *forward Chaining* ditunjukkan oleh Gambar 2.3.

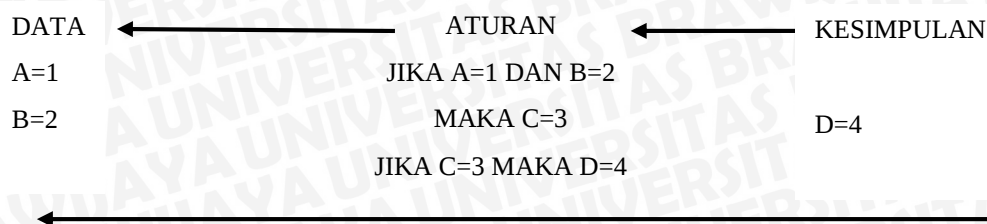


Gambar 2. 3 Forward chaining

2. Backward Chaining

Dalam *backward chaining*, penalaran dimulai dengan tujuan merunut balik ke jalur yang akan mengarahkan ketujuan tersebut (Giarattano dan Riley, 1994). Runut balik disebut juga sebagai *goal-driven reasoning*, merupakan cara yang efisien untuk memecahkan masalah yang dimodelkan sebagai masalah pemilihan terstruktur. Tujuan dari inferensi ini adalah mengambil pilihan terbaik dari banyak kemungkinan.

Pencocokan fakta atau pernyataan di mulai dari bagian sebelah kanan (THEN dulu). Dengan kata lain, penalaran dimulai dari hipotesis terlebih dahulu, dan untuk menguji kebenaran hipotesis tersebut dicari harus dicari fakta-fakta yang ada dalam basis pengetahuan. Bagaimana penalaran menggunakan metode *backward chaining* ditunjukkan oleh Gambar 2.4.



Gambar 2. 4 Backward Chaining

3. *Certainty Factor*

Dalam aplikasi sistem pakar terdapat suatu metode untuk menyelesaikan masalah ketidakpastian data, salah satu metode yang dapat digunakan adalah faktor kepastian (*certainty factor*). *Certainty factor* merupakan nilai parameter klinis yang diberikan untuk menunjukkan besarnya kepercayaan. Ada 2 macam faktor kepastian yang digunakan, yaitu faktor kepastian yang diisikan oleh pakar bersama dengan aturan dan faktor kepastian yang diberikan oleh pengguna. Faktor kepastian yang diisikan oleh pakar menggambarkan kepercayaan pakar terhadap hubungan antara *antecedent* dan konsekuensi. Sementara itu faktor kepastian dari pengguna menunjukkan besarnya kepercayaan terhadap keberadaan masing-masing elemen dalam *antecedent*.

2.5 Fuzzy K-Nearest Neighbor

Fuzzy k-nearest neighbor merupakan gabungan dari metode *k-nearest neighbor* dengan metode *fuzzy*. Metode *Fuzzy k-nearest neighbor* dikembangkan pertama kali oleh James M. Keller. Dasar dari algoritma ini adalah pemberian nilai membership sebagai fungsi pola jarak / kesamaan dari sejumlah himpunan *k-nearest neighbor* dan pemberian nilai keanggotaan *neighbor* pada kelas tertentu. Sehingga pada algoritma ini, data testing yang akan diklasifikasikan akan memiliki nilai keanggotaan pada semua kelas. Klasifikasi algoritma ini nantinya akan memilih nilai keanggotaan kelas pada data testing yang paling tinggi.

2.5.1 Proses K-Nearest Neighbor

K-Nearest Neighbor (K-NN) adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. K-NN termasuk algoritma *supervised learning* dimana hasil dari *query instance* yang baru diklasifikasikan berdasarkan mayoritas dari kategori pada KNN. Nanti kelas yang paling banyak akan menjadi kelas hasil klasifikasi (Anggraeni, 2014).

Langkah-langkah dalam metode K-NN yaitu (Meristika, 2013):

1. Normalisasi

Langkah pertama yang dilakukan yaitu melakukan normalisasi parameter terhadap nilai data uji dan data latih. Atribut yang telah dinormalisasi akan digunakan pada proses klasifikasi. Dalam penelitian ini menggunakan normalisasi

min-max. Persamaan 2.1 merupakan persamaan yang digunakan untuk menghitung normalisasi.

$$V' = \frac{V - \min_A}{\max_A - \min_A} \quad (2.1)$$

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

$\max_A$ = nilai maksimum dari suatu atribut, A.

2. Menghitung kuadrat jarak *euclidean*

Selanjutnya menghitung jarak terdekat antara data uji dengan data latih. Perhitungan ini dengan tujuan untuk menentukan perbedaan antara nilai-nilai atribut pada record x_1 dan x_2 . Persamaan 2.2 merupakan persamaan untuk menghitung jarak menggunakan *euclidean*.

$$\begin{aligned} x_1 &= (x_{11}, x_{12}, \dots, x_{1n}) \\ x_2 &= (x_{21}, x_{22}, \dots, x_{2n}) \\ d(x_1, x_2) &= \sqrt{\sum_{r=1}^n (a_r(x_1) - a_r(x_2))^2} \end{aligned} \quad (2.2)$$

Dimana:

x_1 dan x_2 = dua *record* dengan n atribut

n = banyaknya data

$d(x_1, x_2)$ = jarak *euclidean*

a_r = nilai atribut ke- r pada *record*.

3. Mengurutkan berdasarkan nilai jarak *euclidean*

Setelah diperoleh nilai jarak *euclidean*, data diurutkan berdasarkan jarak *euclidean* terkecil. Hal ini menunjukkan kemiripan data uji terhadap data latih yang paling dekat. Semakin kecil nilai jarak *euclidean* maka semakin dekat kemiripan kelas data uji.

4. Diambil sebanyak K tetangga

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

5. Target *output* kelas mayoritas

Dengan menggunakan kategori *nearest neighbor* yang paling mayoritas maka kelas target *output* data baru adalah kelas yang mayoritas.

2.5.2 Proses Fuzzy K-Nearest Neighbor

Proses *fuzzy k-nearest neighbor* tahap awalnya menggunakan proses KNN, namun penentuan hasil akhir menggunakan nilai keanggotaan kelas pada sejumlah k tetangga terdekat. Penentuan nilai keanggotaan kelas ke- i pada tetangga ke- j dihitung menggunakan Persamaan 2.3 (Keller, et al., 1985).

$$u_{ij} = \begin{cases} 0,51 + \left(\frac{n_j}{K}\right) * 0,49 & , \text{jika } j = i \\ \left(\frac{n_j}{K}\right) * 0,49 & , \text{jika } j \neq i \end{cases} \quad (2.3)$$

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.

Kemudian penentuan nilai keanggotaan kelas pada data uji x dihitung menggunakan persamaan 2.4.

$$u_i(x) = \frac{\sum_{j=1}^K u_{ij} (\|x - x_j\|^{-\frac{2}{m-1}})}{\sum_{j=1}^K (\|x - x_j\|^{-\frac{2}{m-1}})} \quad (2.4)$$

Dimana:

$u_i(x)$ = nilai keanggotaan data x ke kelas c_i

$x - x_j$ = selisih jarak data x ke data x_j dalam k tetangga terdekat

K = jumlah tetangga terdekat yang digunakan

m = bobot pangkat (*weight exponent*) yang besarnya $m > 1$.

Variabel m menentukan berapa banyak jarak bobot saat menghitung kontribusi masing-masing tetangga dengan nilai keanggotaan. Jika $m=2$, maka kontribusi masing-masing titik tetangga dibobot dengan kebalikan dari jaraknya titik yang diklasifikasikan. Jika m meningkat, tetangga lebih merata berbobot, dan jarak relatif mereka dari titik yang diklasifikasikan kurang berpengaruh. Jika m mendekati satu, tetangga dekat yang berbobot jauh lebih besar dari pada yang lebih jauh.

2.5.3 Kelebihan Fuzzy K-Nearest Neighbor

Fuzzy K-Nearest Neighbor (Fuzzy K-NN) memiliki dua keunggulan utama dari pada algoritma *K-NN* yaitu (Putri, et al., 2010):

1. Algoritma ini mampu mempertimbangkan sifat tidak jelas kelasnya (ambigu) dari tetangga jika ada. Algoritma ini telah dirancang sedemikian rupa agar tetangga yang ambigu tidak memainkan peranan penting dalam klasifikasi saat ini.
2. Sebuah *instance* akan memiliki derajat nilai keanggotaan pada setiap kelas sehingga akan lebih memberikan kekuatan atau kepercayaan suatu *instance* berada pada suatu kelas.

2.6 Pengujian

2.6.1 Pengujian Fungsional

Pengujian perangkat lunak dilakukan untuk mengetahui bahwa program dapat melakukan apa yang seharusnya dilakukan dan untuk menemukan sebuah kesalahan pada program sebelum program tersebut mulai digunakan. Metode dalam pengujian terdiri atas *white box testing* dan *black box testing*. *White box testing* merupakan pengujian yang didasarkan pada pengecekan terhadap detail perancangan dengan menggunakan struktur kontrol dari desain program secara procedural. *Black box testing* merupakan pengujian yang fokus pada persyaratan fungsional perangkat lunak. *Black box testing* menggunakan pendekatan komplementer yang memungkinkan untuk mengungkap kesalahan (Anggraeni, 2015).

Pengujian fungsional dilakukan untuk mengetahui apakah sistem yang dibuat telah menyediakan fungsi-fungsi sesuai dengan yang dibutuhkan. Pengujian fungsional merupakan pengujian dengan menggunakan metode *blackbox*, karena pengujian yang dilakukan tidak difokuskan pada jalannya algoritma program melainkan pada kesesuaian antara kinerja aplikasi dengan daftar kebutuhan.

2.6.2 Pengujian Akurasi

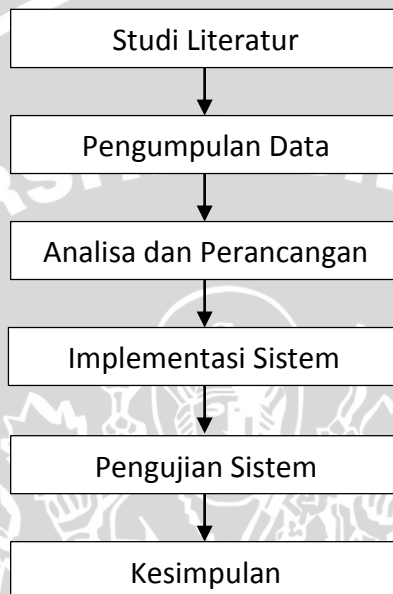
Akurasi merupakan seberapa dekat suatu angka hasil pengukuran terhadap angka sebenarnya (*true value* atau *reference value*). Dalam penelitian ini akurasi tingkat diagnosis penyakit tomat dihitung dari jumlah prediksi yang benar dibagi dengan jumlah data uji. Perhitungan tingkat akurasi dihitung menggunakan Persamaan 2.5.

$$akurasi = \frac{\sum \text{data uji benar}}{\sum \text{jumlah h total data uji}} \times 100\% \quad (2.5)$$

Jumlah prediksi benar merupakan jumlah record data uji yang diprediksi kelasnya menggunakan metode klasifikasi dan hasilnya sama dengan kelas sebenarnya. Sedangkan jumlah total prediksi adalah jumlah keseluruhan *record* yang diprediksi kelasnya (seluruh data uji).

BAB 3 METODOLOGI

Pada bab ini membahas tentang langkah-langkah dalam pembuatan pemodelan sistem pakar untuk identifikasi penyakit tanaman tomat menggunakan metode *fuzzy k-nearest neighbour*. Metode penelitian yang dilakukan dalam penelitian ini melalui beberapa tahapan yaitu studi literatur, pengumpulan data, analisa dan perancangan, implementasi sistem, uji coba sistem, dan kesimpulan. Tahapan-tahapan dalam penelitian tersebut diilustrasikan dengan Gambar 3.1.



Gambar 3. 1 Diagram blok metode penelitian

3.1 Studi Literatur

Tahapan penelitian ini membutuhkan studi literatur untuk merealisasikan pemodelan sistem pakar identifikasi penyakit tanaman tomat menggunakan metode *Fuzzy k-nearest neighbour*. Studi literatur bertujuan untuk mencari referensi yang relevan berkaitan dengan penelitian atau permasalahan agar peneliti dapat meningkatkan pemahaman serta pengetahuan mengenai permasalahan yang diangkat. Referensi berupa teori yang berkaitan dengan penyakit pada tanaman tomat, sistem pakar, dan metode *fuzzy k-nearest neighbour*. Informasi dan pustaka yang berkaitan dengan penelitian ini diperoleh dari buku, jurnal, laporan penelitian, situs internet, pakar, dosen pembimbing, dan rekan mahasiswa.

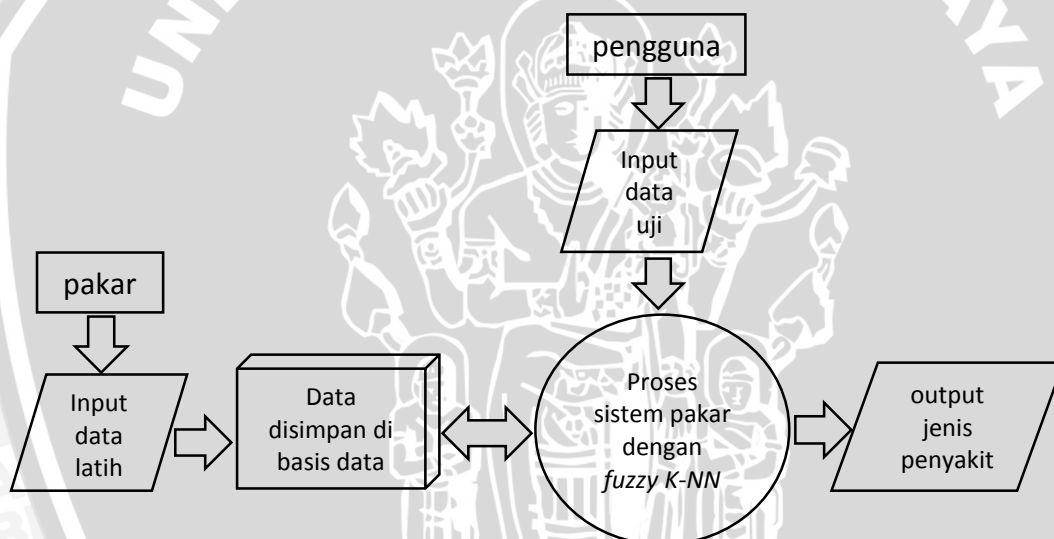
3.2 Pengumpulan Data

Pada tahapan ini dilakukan pengumpulan data dari objek yang akan diteliti. Data mengenai penyakit tanaman tomat diperoleh dari Balai Pengkajian Teknologi Pertanian Jawa Timur dengan alamat Jl. Raya Karangploso km 4 Malang. Data yang diperoleh berupa gejala yang tampak pada penyakit tanaman tomat dengan bobot tertentu. Gejala yang tampak pada daun, batang, dan buah dikelompokkan dalam

15 gejala. Untuk pembobotan gejala dikonsultasikan dengan Prof. (Riset). Dr. Ir. Moh. Cholil Mahfud, M.S. sebagai pakar penyakit tanaman tomat. Data yang diperoleh dijadikan sebagai data latih untuk pembelajaran identifikasi penyakit tanaman tomat dengan melakukan klasifikasi menggunakan metode *fuzzy k-nearest neighbor*.

3.3 Analisa dan Perancangan

Pada tahapan ini dilakukan analisis serta perancangan yang dijadikan sebagai acuan dalam implementasi sistem. Sistem perangkat lunak yang akan dibangun adalah sebuah aplikasi yang dapat mengidentifikasi penyakit tanaman tomat. Metode *fuzzy k-nearest neighbor* digunakan untuk mengklasifikasikan penyakit tanaman tomat berdasarkan gejala. Data latih dan data uji harus dikonversikan ke bentuk numerik sehingga dapat dilakukan perhitungan. Data latih akan disimpan didalam basis data dan dipanggil ketika dibutuhkan untuk dilakukan perhitungan klasifikasi. Perancangan aplikasi pemodelan sistem pakar penyakit tanaman tomat dapat dilihat pada gambar 3.2.



Gambar 3. 2 Arsitektur perancangan sistem

Gambar 3.2 tersebut menjelaskan bagaimana sistem ini bekerja. Pertama, admin (pakar) memasukkan data latih berupa gejala dan diagnosis dalam bentuk angka. Data akan disimpan dalam basis data. Pengguna kemudian dapat melakukan identifikasi penyakit dengan cara memilih gejala pada form dalam sistem sesuai dengan gejala yang tampak pada tanaman. Sistem akan memproses data yang sudah dimasukkan oleh pengguna dan data latih dari basis data menggunakan perhitungan *fuzzy k-nearest neighbor*. Hasil keluaran dari sistem ini berupa hasil identifikasi penyakit tanaman tomat. 6 jenis penyakit yang diklasifikasikan adalah penyakit busuk daun, bercak coklat, layu fusarium, layu bakteri, mozaik, dan keriting.

3.4 Implementasi Sistem

Pada tahap ini dilakukan proses implementasi sistem berdasarkan analisa dan perancangan yang telah dibuat pada tahap sebelumnya. Implementasi dari sistem meliputi:

1. Pembuatan antarmuka pengguna berupa halaman web.
2. Memasukkan data latih penelitian ke basisdata MySQL untuk diolah menjadi informasi yang berguna bagi sistem.
3. Penerapan metode *fuzzy k-nearest neighbor* dalam program yang dibangun menggunakan bahasa pemrograman PHP.

3.5 Pengujian Sistem

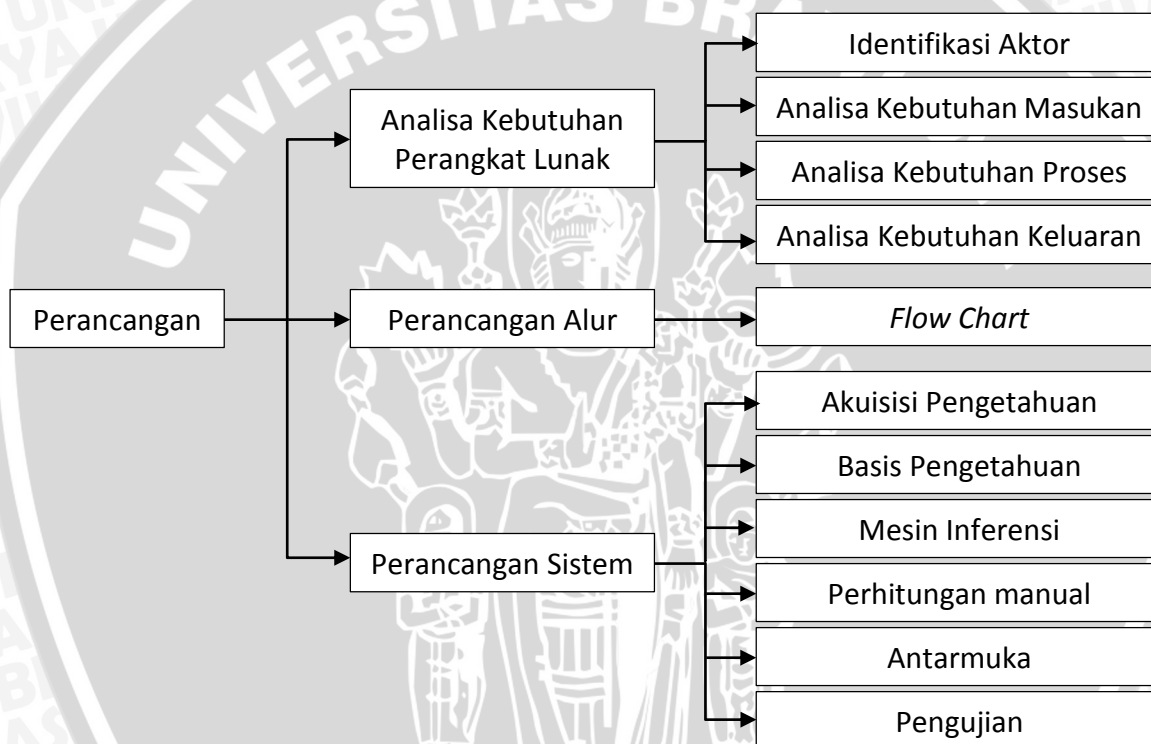
Pada tahap ini dilakukan pengujian untuk mengetahui akurasi dari sistem yang telah dibuat. Pengujian dilakukan dengan menggunakan variasi nilai K dan variasi data latih. Pengujian ini dilakukan dengan menyediakan sebanyak 30 data uji penyakit pada tomat. Keluaran sistem yang berupa hasil identifikasi kemudian dibandingkan dengan hasil identifikasi yang telah diberikan oleh pakar.

3.6 Kesimpulan

Pada tahap terakhir dari penelitian ini adalah pengambilan kesimpulan. Kesimpulan diambil dari hasil pengujian yang telah dilakukan dan analisis metode yang diterapkan. Pada tahap ini ditambahkan saran oleh peneliti yang ditujukan kepada pembaca maupun pengembang selanjutnya.

BAB 4 PERANCANGAN

Bab ini membahas tentang perancangan pada sistem “Pemodelan Sistem Pakar Untuk Identifikasi Penyakit Pada Tanaman Tomat Menggunakan Metode *Fuzzy K-Nearest Neighbor*”. Pohon perancangan pemodelan sistem pakar meliputi tiga tahapan yaitu analisa kebutuhan perangkat lunak, perancangan perangkat lunak, dan perancangan pemodelan sistem pakar. Analisa kebutuhan perangkat lunak terdiri dari identifikasi aktor, analisa kebutuhan masukan, analisa kebutuhan proses, dan analisa kebutuhan keluaran. Perancangan alur berupa *flowchart*. Pemodelan sistem pakar terdiri dari perancangan akuisisi pengetahuan, basis pengetahuan, mesin inferensi, *blackboard*, fasilitas penjelas, dan antarmuka. Pohon perancangan pemodelan sistem pakar dapat dilihat pada Gambar 4.1.



Gambar 4. 1 Pohon perancangan

4.1 Analisa Kebutuhan Perangkat Lunak

Analisa kebutuhan perangkat lunak terdiri dari identifikasi aktor yang berinteraksi dengan sistem, kebutuhan masukan, proses, dan keluaran sistem. Analisa kebutuhan ditujukan untuk menggambarkan kebutuhan-kebutuhan yang harus tersedia oleh sistem agar dapat memenuhi kebutuhan pengguna. Berikut ini adalah kebutuhan yang digunakan dalam pembuatan sistem pakar:

1. Kebutuhan perangkat keras, meliputi: komputer.
2. Kebutuhan perangkat lunak, meliputi: sistem operasi windows 8, browser, basis data Mysql, dan sublime text.
3. Kebutuhan data, meliputi: data skor tiap gejala penyakit tanaman tomat.

4.1.1 Identifikasi Pengguna

Sistem yang dibuat dapat diakses oleh 2 aktor yaitu pengguna umum dan pakar sebagai admin. Pengguna umum dan pakar dapat melihat data latih dan informasi dalam sistem dan melakukan identifikasi penyakit tanaman tomat menggunakan sistem tanpa harus melakukan *login*. Namun pakar memiliki kelebihan hak akses login untuk mengelola data latih. Deskripsi aktor dapat dilihat pada Tabel 4.1.

Tabel 4. 1 Identifikasi aktor

Aktor	Dekripsi Aktor
Pengguna Umum	Aktor yang dapat menggunakan sistem untuk informasi yang ada pada sistem dan dapat melakukan identifikasi penyakit tanaman tomat menggunakan sistem. aktor ini tidak dapat melakukan proses login.
Pakar Penyakit Tanaman Tomat	Aktor yang dapat menggunakan sistem untuk informasi yang ada pada sistem dan dapat melakukan identifikasi penyakit tanaman tomat menggunakan sistem. Aktor ini mendapat hak akses login untuk mengelola data latih dengan menambah, mengubah, atau menghapus data latih.

4.1.2 Analisa Kebutuhan Masukan

Kebutuhan masukan dari sistem ini diperoleh dari pakar dan pengguna. Data masukan dari pakar digunakan sebagai basis pengetahuan dari pemodelan sistem pakar dalam mengidentifikasi penyakit tanaman tomat. Data tersebut berupa data gejala penyakit tanaman tomat, data jenis penyakit tanaman tomat, dan bobot setiap gejala penyakit tanaman tomat. Sedangkan data masukan dari pengguna berupa gejala yang tampak pada tanaman tomat.

4.1.3 Analisa Kebutuhan Proses

Analisa kebutuhan proses merupakan suatu proses penalaran. Sistem akan melakukan penalaran untuk melakukan identifikasi penyakit tanaman tomat berdasarkan gejala yang dimasukkan oleh pengguna. Proses pada sistem berupa perhitungan dengan metode *fuzzy k-nearest neighbor*. Metode digunakan untuk mengidentifikasi jenis penyakit tanaman tomat. Perhitungan dimulai dengan menghitung jarak *euclidean* antara data uji dengan setiap data latih. Hasil perhitungan jarak *euclidean* akan diurutkan dan diambil k tetangga terdekat untuk proses *fuzzy k-nearest neighbor*. Proses berikutnya adalah menentukan jenis kelas yang terdapat pada data k ketetanggaa kemudian menentukan nilai keanggotaan setiap data terhadap setiap kelas. Kelas yang memiliki nilai keanggotaan terbesar akan digunakan sebagai kelas dari data uji.

4.1.4 Analisa Kebutuhan Keluaran

Keluaran dari sistem ini adalah hasil identifikasi penyakit tanaman tomat yang diperoleh dari perhitungan menggunakan perhitungan metode *fuzzy k-nearest*

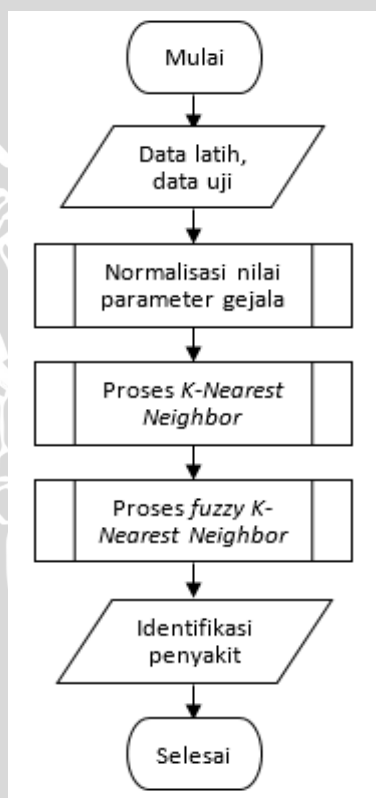
neighbor. Hasil identifikasi tersebut berdasarkan fakta gejala yang tampak pada tanaman tomat dan dimasukkan oleh pengguna ke dalam sistem. Hasil keluaran sistem terdiri dari tabel perhitungan *fuzzy k-nearest neighbor*, jenis penyakit, dan informasi tentang penanganan penyakit tersebut.

4.2 Perancangan Perangkat Lunak

Perancangan perangkat lunak menjelaskan tentang hubungan antar komponen dalam membangun sebuah sistem dengan metode yang telah ditentukan yaitu menggunakan metode *fuzzy k-nearest neighbor*. Sehingga dibuat *flowchart* sebagai perancangan untuk mengetahui alur jalannya program.

4.2.1 Proses Klasifikasi

Pada tahap ini dijelaskan mengenai alur sistem dalam pemrosesan data latih dan data uji gejala penyakit tanaman tomat menggunakan metode *fuzzy k-nearest neighbor*. Gambar 4.2 merupakan *flowchart* dari proses klasifikasi secara umum.



Gambar 4. 2 Proses klasifikasi secara umum

Proses klasifikasi menggunakan metode *fuzzy k-nearest neighbor* terdiri dari tiga proses utama yaitu:

1. Normalisasi nilai parameter gejala

Proses normalisasi dilakukan terhadap nilai parameter gejala pada data latih maupun data uji. Hal ini dilakukan agar parameter mempunyai skala nilai yang tidak berselisih jauh sehingga sistem dapat melakukan klasifikasi dengan baik.

2. Proses *K-Nearest Neighbor*

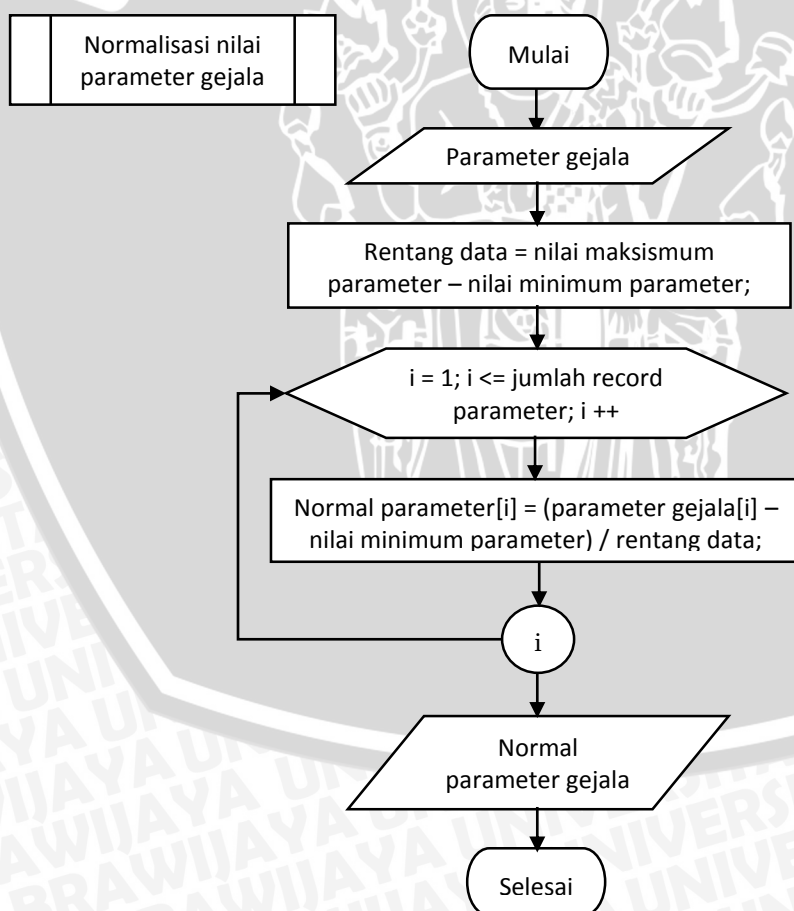
Proses *k-nearest neighbor* merupakan suatu proses klasifikasi terhadap data uji untuk dimasukkan kedalam kelas yang sudah ada yaitu kelas jenis penyakit. Proses pengklasifikasian menggunakan jarak *euclidean distance* antara data uji dengan data latih. Dan selanjutnya mayoritas kelas pada sejumlah k (tetangga terdekat) digunakan untuk menentukan data uji yang masuk dalam salah satu kelas prediksi.

3. Proses *Fuzzy K-Nearest Neighbor*

Pada proses *fuzzy k-nearest neighbor* dilakukan proses pemberian nilai keanggotaan pada data uji setelah melalui proses *K-NN*, sehingga setiap data uji mempunyai nilai relatif terhadap setiap kelas. Kelas dengan nilai keanggotaan terbesar yang menjadi kelas hasil *fuzzy k-nearest neighbor*.

4.2.2 Normalisasi

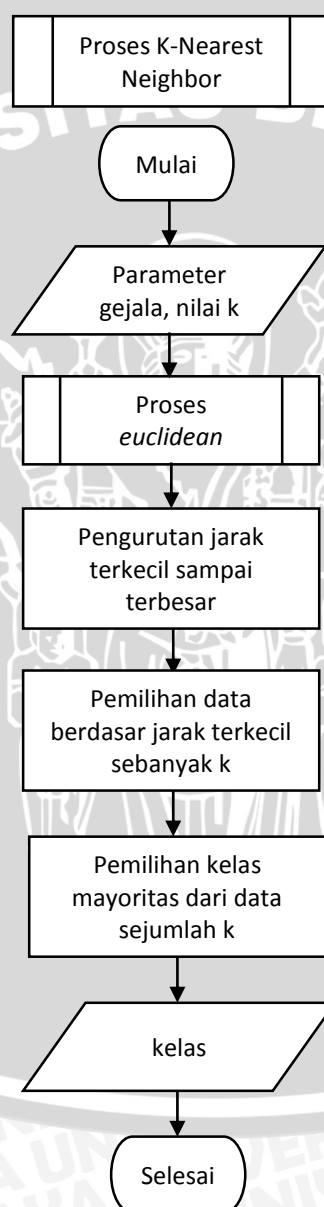
Proses normalisasi pada tahap ini yaitu perubahan nilai pada setiap parameter menjadi nilai yang baku dan memiliki standar skala nilai yang tidak berjauhan. Menormalisasi data menggunakan Persamaan 2.1. Gambar 4.3 merupakan alur dari proses normalisasi.



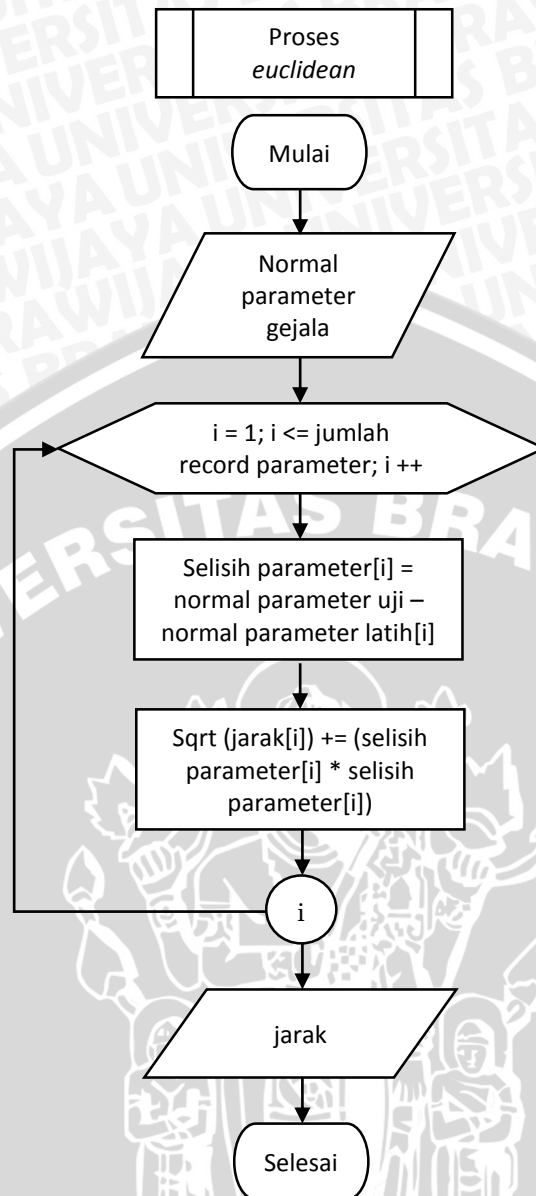
Gambar 4. 3 Proses normalisasi

4.2.3 Proses K-Nearest Neighbor

Proses klasifikasi menggunakan metode *K-NN* dengan menempatkan data uji pada salah satu kelas diantara 6 kelas jenis penyakit. Dalam menentukan data ke dalam sebuah kelas menggunakan Persamaan 2.2 untuk menentukan jarak antara data uji dengan data latih dan kemudian diurutkan mulai dari nilai jarak terkecil hingga terbesar. Selanjutnya ditentukan nilai *k* (tetangga terdekat) untuk mencari jumlah mayoritas kelas yang ada pada sejumlah *k* tersebut. Gambar 4.4 merupakan diagram alur proses *k-nearest neighbor* dan Gambar 4.5 merupakan alur proses *euclidean*.



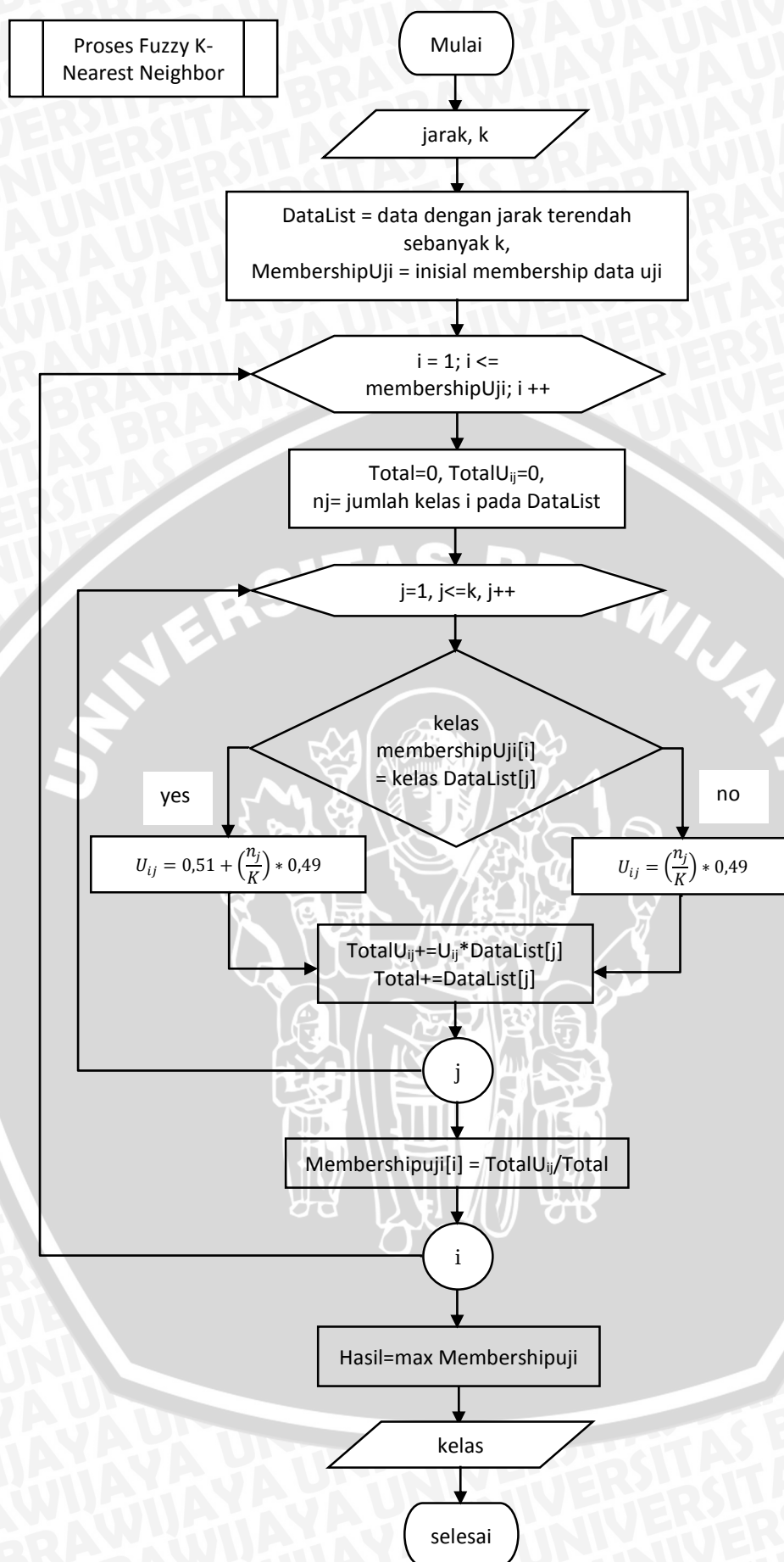
Gambar 4. 4 Proses KNN



Gambar 4. 5 Proses *euclidean*

4.2.4 Proses Fuzzy K-Nearest Neighbor

Pada proses *fuzzy k-nearest neighbor*, proses awalnya menggunakan proses yang sama dengan algoritma *k-nearest neighbor*, namun setelah proses penentuan nilai *k* adalah mencari nilai keanggotaan. Nilai derajat keanggotaan pada tiap kelas dalam *k* tetangga terdekat dihitung dengan algoritma *fuzzy* untuk menentukan kelas data uji. Pada algoritma ini nilai keanggotaan tiap kelas dihitung dan diambil keanggotaan terbesar yang digunakan untuk menentukan kelas prediksi. Gambar 4.6 merupakan alur proses *fuzzy k-nearest neighbor*.

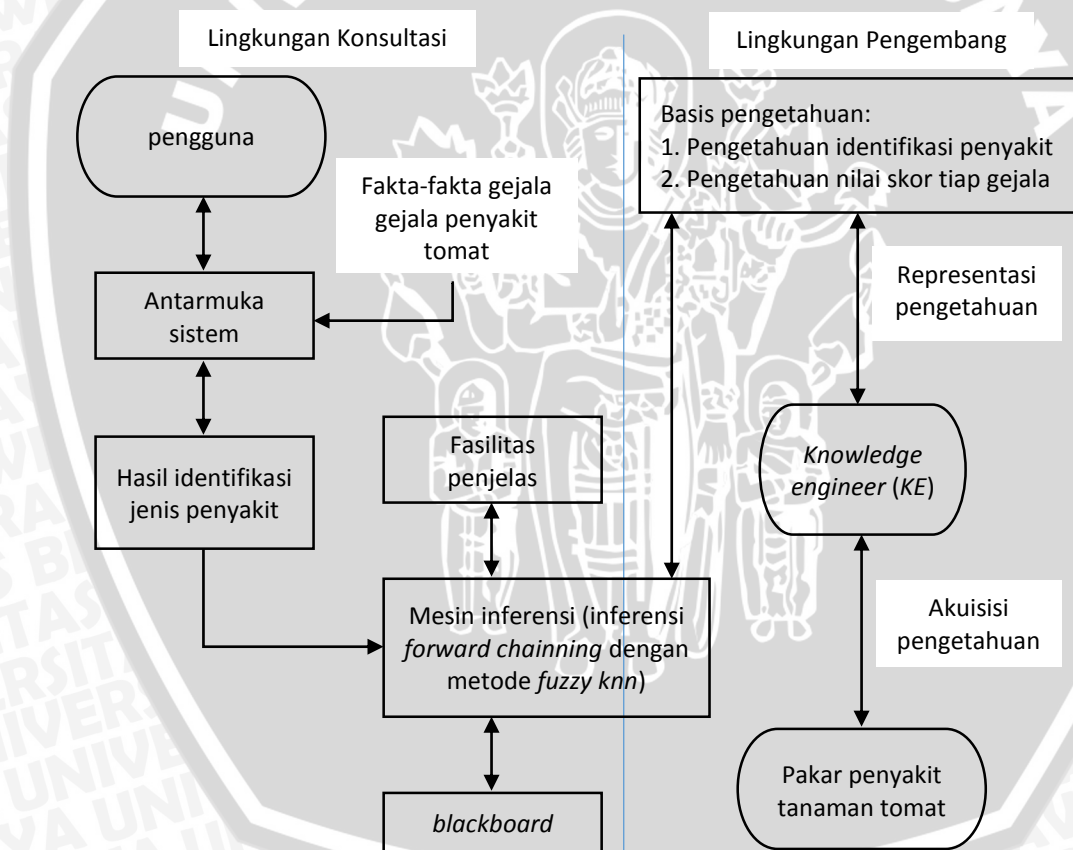


Gambar 4. 6 Proses FKNN

4.3 Perancangan Sistem

Sistem pada penelitian ini adalah pemodelan sistem pakar yang dapat mengidentifikasi penyakit tanaman tomat. Metode *fuzzy k-nearest neighbor* digunakan untuk proses pengambilan kesimpulan, sedangkan penelusuran jawaban dari perhitungan menggunakan metode *forward chaining*. Tahapan yang dilakukan pakar dalam melakukan identifikasi penyakit tanaman tomat adalah dengan cara melihat gejala yang tampak pada tanaman. Konsep sistem pakar yang dibangun dengan metode *fuzzy k-nearest neighbor* merupakan sistem yang pengambilan kesimpulannya berdasarkan data gejala yang diperoleh dari pakar.

Sistem menerima masukan dari pengguna berupa gejala yang tampak dari tanaman tomat. Masukan dari pengguna akan dilakukan proses perhitungan dengan metode *fuzzy k-nearest neighbor*. Dari hasil perhitungan akan diperoleh suatu kesimpulan jenis penyakit dari data uji pengguna. Gambar 4.7 menggambarkan kerangka konsep sistem.



Gambar 4. 7 Kerangka konsep sistem

Bagian atau komponen yang membangun pemodelan sistem pakar identifikasi penyakit tanaman tomat meliputi:

4.3.1 Akuisisi Pengetahuan

Akuisisi pengetahuan adalah proses akumulasi, transfer dan transformasi keahlian dalam pemecahan masalah oleh pakar atau sumber pengetahuan lain dari pakar ke sistem komputer untuk membangun basis pengetahuan. Pengetahuan dapat berasal dari pakar, buku, jurnal ilmiah, maupun internet. Metode yang digunakan dalam akuisisi pengetahuan pada penelitian ini adalah wawancara.

Wawancara merupakan suatu teknik pengumpulan data untuk mendapatkan informasi dari sumber data secara langsung melalui percakapan tanya jawab. Tujuan wawancara ini adalah memperoleh keterangan terperinci dan mendalam mengenai pandangan pakar terhadap permasalahan tertentu sehingga kesimpulan akhir dapat diperoleh.

Tujuan dari wawancara ini adalah memperoleh wawasan dari pakar untuk domain masalah tertentu. Pada wawancara ini *knowledge engineer* mengumpulkan semua informasi tentang gejala-gejala penyakit tanaman tomat. Setiap gejala akan diberikan nilai berdasarkan pembobotan yang diberikan oleh pakar. Berdasarkan wawancara dengan Prof. (Riset). Dr. Ir, Moh. Cholil Mahfud, M.S diperoleh nilai pembobotan tiap gejala penyakit tanaman tomat seperti pada Tabel 4.2.

Tabel 4. 2 Kriteria gejala penyakit tanaman tomat

No.	Kode	Gejala	Opsi	Nilai
1	G01	sisi bawah daun terdapat jamur putih	tidak ada	0
			Sedikit	10
			Banyak	20
2	G02	pada daun terdapat bercak coklat dengan bagian luar berwarna kuning ke hijau	tidak ada	0
			Sedikit	45
			banyak	90
3	G03	daun berwarna belang antara hijau kekuningan dan berkerut/terpuntir	tidak ada	0
			sedikit	45
			banyak	90
4	G04	pada daun terdapat bercak coklat gelap, keras, dan melingkar sepusat	tidak ada	0
			sedikit	45
			banyak	90
5	G05	daun menguning, layu, dan gugur	tidak ada	0
			sedikit	10
			banyak	20
6	G06	daun mengeriting, dan menggulung ke atas	tidak ada	0
			sedikit	45

			banyak	90
7	G07	pada batang terdapat bercak coklat lonjong dan kosentris	tidak ada	0
			sedikit	10
			banyak	20
8	G08	pada batang terdapat bercak kuning atau coklat	tidak ada	0
			sedikit	10
			banyak	20
9	G09	terdapat cincin coklat pada pembuluh yang dipotong	tidak ada	0
			sedikit	45
			banyak	90
10	G10	terdapat lendir pada pembuluh batang yang dipotong	tidak ada	0
			sedikit	45
			banyak	90
11	G11	pada buah terdapat bercak coklat tua, keras dan berkerut	tidak ada	0
			sedikit	10
			banyak	20
12	G12	pada buah terdapat bercak coklat gelap, mengendap, berlekuk, dan busuk kering	tidak ada	0
			sedikit	10
			banyak	20
13	G13	buah kecil dan sedikit	tidak ada	0
			Ada	20
14	G14	tanaman layu	tidak ada	0
			ada	20
15	G15	tanaman kerdil	tidak ada	0
			ada	20

Sumber: Pakar

Gejala pada tanaman tomat diringkaskan menjadi 15 gejala yang mewakili gejala dari 6 jenis penyakit tanaman tomat. Gejala yang khas pada penyakit tertentu diberi skor yang tinggi, yaitu 90 untuk banyak dan 45 untuk sedikit penampakan gejala yang muncul pada tanaman. Sedangkan untuk gejala yang tidak khas atau gejala umum akan diberi skor lebih rendah, yaitu 20 untuk banyak dan 10 untuk sedikit.

4.3.2 Basis Pengetahuan

Basis pengetahuan berisikan tentang pengetahuan yang relevan untuk memahami, merumuskan dan, dan memecahkan permasalahan. Basis pengetahuan mengarahkan pengguna pengguna untuk memecahkan permasalahan khusus dalam domain tertentu. Basis pengetahuan merupakan inti dari sistem dimana basis pengetahuan ini merupakan representasi pengetahuan dari pakar. Tabel 4.3 merupakan akuisisi representasi pengetahuan pakar sebelumnya yang menjadi acuan untuk basis pengetahuan pada penelitian ini.

Tabel 4. 3 Aturan jenis penyakit tanaman tomat

Rule	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Identifikasi
R1	x	x						x			x					Busuk Daun
R2				x	x		x					x				Bercak Coklat
R3					x				x				x	x		Layu Fusarium
R4					x					x			x	x		Layu Bakteri
R5			x					x					x		x	Mosaik
R6						x							x		x	Keriting

Sumber: Pakar

Setiap jenis penyakit memiliki satu gejala yang khas dan beberapa gejala umum. Yang termasuk dalam gejala khas adalah G02, G03, G04, G06, G09, dan G10. Pada basis pengetahuan dapat terjadi perubahan aturan sewaktu-waktu sesuai dengan keadaan yang berkembang saat ini.

4.3.3 Mesin Inferensi

Mesin inferensi adalah sebuah program yang berfungsi memandu proses penalaran terhadap suatu kondisi berdasarkan basis pengetahuan yang ada, manipulasi, dan mengarahkan suatu kaidah, model, dan fakta yang disimpan dalam basis pengetahuan untuk memperoleh kesimpulan. Dalam prosesnya, mesin inferensi menggunakan strategi pengendalian yang berfungsi sebagai panduan arah dalam melakukan suatu proses penalaran.

Metode penelusuran jawaban menggunakan metode *forward chaining*, yaitu dimulai dari sekumpulan fakta-fakta tentang suatu gejala yang diberikan oleh pengguna sebagai masukan pada sistem. Kemudian dilakukan proses pelacakan dari masukan nilai gejala dengan menggunakan perhitungan *fuzzy k-nearest neighbor*. Kesimpulan dari perhitungan akhir berupa identifikasi jenis penyakit tanaman tomat.

4.3.4 Perhitungan Manual

Pada proses ini dilakukan proses perhitungan manual mulai dari awal hingga akhir. Pada perhitungan manual ini hanya menggunakan 6 data latih agar perhitungan tidak terlalu banyak dan mudah dipahami. Satu data uji digunakan untuk menguji data tersebut masuk dalam kelas jenis penyakit apa. Tabel 4.4 merupakan tabel data latih penyakit tanaman tomat dan Tabel 4.5 merupakan tabel data uji.

Tabel 4. 4 Data latih penyakit tanaman tomat

No	Data	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Identifikasi
1	D029	0	0	0	0	0	90	0	0	0	0	0	0	20	0	20	P6
2	D038	0	0	90	0	0	0	0	20	0	0	0	0	20	0	20	P5
3	D113	0	0	0	0	20	0	0	0	0	90	0	0	20	20	0	P4
4	D131	0	0	0	90	20	0	20	0	0	0	0	20	0	0	0	P2
5	D142	20	90	0	0	0	0	0	20	0	0	20	0	0	0	0	P1
6	D162	0	0	0	0	20	0	0	0	90	0	0	0	20	20	0	P3

Tabel 4. 5 Data uji penyakit tanaman tomat

No	Data	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Identifikasi
1	D153	0	0	0	0	20	0	0	0	45	0	0	0	20	20	0	?

1. Normalisasi data latih dan data uji

Langkah pertama yang dilakukan dalam metode ini adalah normalisasi data setiap atribut. Normalisasi diperlukan agar nilai setiap atribut mempunyai jarak yang tidak berjauhan. Normalisasi yang digunakan adalah *min-max normalization* seperti pada Persamaan 2.1. Pada proses normalisasi secara manual kita gunakan satu sampel yaitu nilai pada *record* data pertama. Sehingga untuk atribut data selanjutnya menggunakan cara yang sama. *Record* yang dijadikan contoh normalisasi yaitu pada atribut G01 pada baris pertama yaitu bernilai 90. Seperti diketahui data nilai 90 pada atribut G01 memiliki komponen sebagai berikut:

$$V(x) = 20$$

Nilai *max* = 20

Nilai *min* = 0

Range nilai = nilai *max* - nilai *min* = 20 - 0 = 20.

Nilai maksimum diperoleh dari nilai terbesar pada atribut G01. Nilai minimum diperoleh dari nilai terkecil pada atribut G01. Untuk normalisasi pada atribut yang lainnya juga menggunakan cara yang sama. Untuk mempermudah perhitungan tiap *record* data maka nilai *min*, *max*, dan *range* nilai setiap atribut ditunjukkan pada Tabel 4.6.

Tabel 4. 6 Range nilai atribut

	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15
min	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
max	20	90	90	90	20	90	20	20	90	90	20	20	20	20	20
range	20	90	90	90	20	90	20	20	90	90	20	20	20	20	20

Setelah diketahui nilai *min*, *max* serta *range* setiap atribut, selanjutnya *record* pertama tinggal dihitung nilai normalisasinya menggunakan Persamaan 2.1. sehingga didapatkan nilai normalisasi untuk *record* pertama adalah sebagai berikut:

$$V' = (\text{nilai record} - \text{nilai min}) / \text{range}$$

$$V' = (0 - 0)/20 = 0.$$

Sehingga setiap *record* data pada data latih dan data uji dapat dilakukan normalisasi dengan cara yang sama sehingga mendapatkan hasil seperti Tabel 4.7 untuk normalisasi data latih dan Tabel 4.8 untuk normalisasi data uji.

Tabel 4. 7 Normalisasi data latih

No	Data	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Identifikasi
1	D029	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	P6
2	D038	0	0	1	0	0	0	0	1	0	0	0	0	1	0	1	P5
3	D113	0	0	0	0	1	0	0	0	0	1	0	0	1	1	0	P4
4	D131	0	0	0	1	1	0	1	0	0	0	0	1	0	0	0	P2
5	D142	1	1	0	0	0	0	0	1	0	0	1	0	0	0	0	P1
6	D162	0	0	0	0	1	0	0	0	1	0	0	0	1	1	0	P3

Tabel 4. 8 Normalisasi data uji

No	Data	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Identifikasi
1	D153	0	0	0	0	1	0	0	0	0,5	0	0	0	1	1	0	?

2. Proses *k-nearest neighbor*

Setelah melakukan proses normalisasi, proses selanjutnya adalah melakukan proses perhitungan *k-nearest neighbor*. Setiap baris data latih yang terdiri dari 12 atribut dihitung jaraknya dengan data uji menggunakan persamaan 2.3. Berikut ini merupakan contoh perhitungan jarak *euclidean* pada data *record* yang pertama:

$$d(x_1, x_2) = \sqrt{\sum_{r=1}^n (a_r(x_1) - a_r(x_2))^2}$$

$$d(1) = \sqrt{(0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (1-0)^2 + (0-1)^2 + (0-0)^2 + (0-0)^2 + (0,5-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (1-1)^2 + (1-0)^2 + (0-1)^2}$$

$$d(1) = 2,06155281$$

Dengan cara yang sama untuk menghitung jarak *euclidean* pada setiap data latih. Selanjutnya *record* diurutkan dari nilai terkecil hingga terbesar. Setelah *record* diurutkan. Tabel hasil perhitungan jarak *euclidean* untuk semua *record* data sebelum dan sesudah diurutkan dapat dilihat pada Tabel 4.9 dan Tabel 4.10.

Tabel 4. 9 Jarak euclidean

No	Data	jarak	target
1	D029	2,06155281	P6
2	D038	2,29128785	P5
3	D113	1,11803399	P4
4	D131	2,29128785	P2
5	D142	2,6925824	P1
6	D162	0,5	P3

Tabel 4. 10 Jarak euclidean diurutkan

Rank	Data	jarak	target
1	D029	0,5	P3
2	D038	1,11803399	P4
3	D113	2,06155281	P6
4	D131	2,29128785	P2
5	D142	2,29128785	P5
6	D162	2,6925824	P1

Setelah data diurutkan, diambil nilai data *record* sebanyak k tetangga terdekat yang pada perhitungan manual ini ditentukan nilai k adalah 5. Dari Tabel 4.10 dapat diketahui bahwa dari 5 tetangga terdekat diperoleh jumlah kelas P1=0, P2=1, P3=1, P4=1, P5=1, dan P6=1 karena kelas P1 tidak terdapat dalam k.

3. Proses fuzzy k-nearest neighbor

Setelah proses penentuan nilai k maka akan dilakukan proses fuzzy yang bertujuan untuk mencari nilai keanggotaan data uji pada masing-masing kelas menggunakan persamaan 2.4. Sehingga perlu diketahui nilai k (tetangga terdekat), n_1 (jumlah penyakit1 pada k), n_2 (jumlah penyakit2 pada k), n_3 (jumlah penyakit3 pada k), n_4 (jumlah penyakit4 pada k), n_5 (jumlah penyakit5 pada k), dan n_6 (jumlah penyakit6 pada k) untuk mengitung nilai *membership* tiap kelas.

Diketahui:

$$k = 5, n_1 = 0, n_2 = 1, n_3 = 1, n_4 = 1, n_5 = 1, n_6 = 1$$

$$\begin{aligned} U_{ya P1} &= 0,51 + \left(\frac{n_1}{k}\right) * 0,49 \\ &= 0,51 + \left(\frac{0}{5}\right) * 0,49 \\ &= 0,51 \end{aligned}$$

$$\begin{aligned} U_{tidak P1} &= \left(\frac{n_1}{k}\right) * 0,49 \\ &= \left(\frac{0}{5}\right) * 0,49 \\ &= 0 \end{aligned}$$

$$\begin{aligned} U_{ya P2} &= 0,51 + \left(\frac{n_2}{k}\right) * 0,49 \\ &= 0,51 + \left(\frac{1}{5}\right) * 0,49 \\ &= 0,608 \end{aligned}$$

$$U_{tidak P2} = \left(\frac{n_2}{k}\right) * 0,49$$

$$= \left(\frac{1}{5}\right) * 0,49$$

$$= 0,098$$

$$U_{ya P3} = 0,51 + \left(\frac{n_3}{k}\right) * 0,49$$

$$= 0,51 + \left(\frac{1}{5}\right) * 0,49$$

$$= 0,608$$

$$U_{tidak P3} = \left(\frac{n_3}{k}\right) * 0,49$$

$$= \left(\frac{1}{5}\right) * 0,49$$

$$= 0,098$$

$$U_{ya P4} = 0,51 + \left(\frac{n_4}{k}\right) * 0,49$$

$$= 0,51 + \left(\frac{1}{5}\right) * 0,49$$

$$= 0,608$$

$$U_{tidak P4} = \left(\frac{n_4}{k}\right) * 0,49$$

$$= \left(\frac{1}{5}\right) * 0,49$$

$$= 0,098$$

$$U_{ya P5} = 0,51 + \left(\frac{n_5}{k}\right) * 0,49$$

$$= 0,51 + \left(\frac{1}{5}\right) * 0,49$$

$$= 0,608$$

$$U_{tidak P5} = \left(\frac{n_5}{k}\right) * 0,49$$

$$= \left(\frac{1}{5}\right) * 0,49$$

$$= 0,098$$

$$U_{ya P6} = 0,51 + \left(\frac{n_6}{k}\right) * 0,49$$

$$= 0,51 + \left(\frac{1}{5}\right) * 0,49$$

$$= 0,608$$

$$U_{tidak P6} = \left(\frac{n_6}{k}\right) * 0,49$$

$$= \left(\frac{1}{5}\right) * 0,49$$

$$= 0,098$$

Setelah mendapatkan nilai *membership* untuk masing-masing kelas, dilakukan perhitungan nilai keanggotaan data uji pada tiap kelas menggunakan Persamaan 2.5 dengan nilai $m = 2$.

$$U_{P1} = \frac{(0 * 0,5^{\frac{-2}{2-1}}) + (0 * 1,11803^{\frac{-2}{2-1}}) + (0 * 2,06155^{\frac{-2}{2-1}}) + (0 * 2,29128^{\frac{-2}{2-1}}) + (0 * 2,29128^{\frac{-2}{2-1}})}{0,5^{\frac{-2}{2-1}} + 1,11803^{\frac{-2}{2-1}} + 2,06155^{\frac{-2}{2-1}} + 2,29128^{\frac{-2}{2-1}} + 2,29128^{\frac{-2}{2-1}}} = 0$$

$$U_{P2} = \frac{(0,098 * 0,5^{\frac{-2}{2-1}}) + (0,098 * 1,11803^{\frac{-2}{2-1}}) + (0,098 * 2,06155^{\frac{-2}{2-1}}) + (0,608 * 2,29128^{\frac{-2}{2-1}}) + (0,098 * 2,29128^{\frac{-2}{2-1}})}{0,5^{\frac{-2}{2-1}} + 1,11803^{\frac{-2}{2-1}} + 2,06155^{\frac{-2}{2-1}} + 2,29128^{\frac{-2}{2-1}} + 2,29128^{\frac{-2}{2-1}}} = 0,11593$$

$$U_{P3} = \frac{(0,608 * 0,5^{\frac{-2}{2-1}}) + (0,098 * 1,11803^{\frac{-2}{2-1}}) + (0,098 * 2,06155^{\frac{-2}{2-1}}) + (0,098 * 2,29128^{\frac{-2}{2-1}}) + (0,098 * 2,29128^{\frac{-2}{2-1}})}{0,5^{\frac{-2}{2-1}} + 1,11803^{\frac{-2}{2-1}} + 2,06155^{\frac{-2}{2-1}} + 2,29128^{\frac{-2}{2-1}} + 2,29128^{\frac{-2}{2-1}}} = 0,47464$$

$$U_{P4} = \frac{(0,098 * 0,5^{\frac{-2}{2-1}}) + (0,608 * 1,11803^{\frac{-2}{2-1}}) + (0,098 * 2,06155^{\frac{-2}{2-1}}) + (0,098 * 2,29128^{\frac{-2}{2-1}}) + (0,098 * 2,29128^{\frac{-2}{2-1}})}{0,5^{\frac{-2}{2-1}} + 1,11803^{\frac{-2}{2-1}} + 2,06155^{\frac{-2}{2-1}} + 2,29128^{\frac{-2}{2-1}} + 2,29128^{\frac{-2}{2-1}}} = 0,17332$$

$$U_{P5} = \frac{(0,098 * 0,5^{\frac{-2}{2-1}}) + (0,098 * 1,11803^{\frac{-2}{2-1}}) + (0,098 * 2,06155^{\frac{-2}{2-1}}) + (0,098 * 2,29128^{\frac{-2}{2-1}}) + (0,608 * 2,29128^{\frac{-2}{2-1}})}{0,5^{\frac{-2}{2-1}} + 1,11803^{\frac{-2}{2-1}} + 2,06155^{\frac{-2}{2-1}} + 2,29128^{\frac{-2}{2-1}} + 2,29128^{\frac{-2}{2-1}}} = 0,11593$$

$$U_{P6} = \frac{(0,098 * 0,5^{\frac{-2}{2-1}}) + (0,098 * 1,11803^{\frac{-2}{2-1}}) + (0,608 * 2,06155^{\frac{-2}{2-1}}) + (0,098 * 2,29128^{\frac{-2}{2-1}}) + (0,098 * 2,29128^{\frac{-2}{2-1}})}{0,5^{\frac{-2}{2-1}} + 1,11803^{\frac{-2}{2-1}} + 2,06155^{\frac{-2}{2-1}} + 2,29128^{\frac{-2}{2-1}} + 2,29128^{\frac{-2}{2-1}}} = 0,12015$$

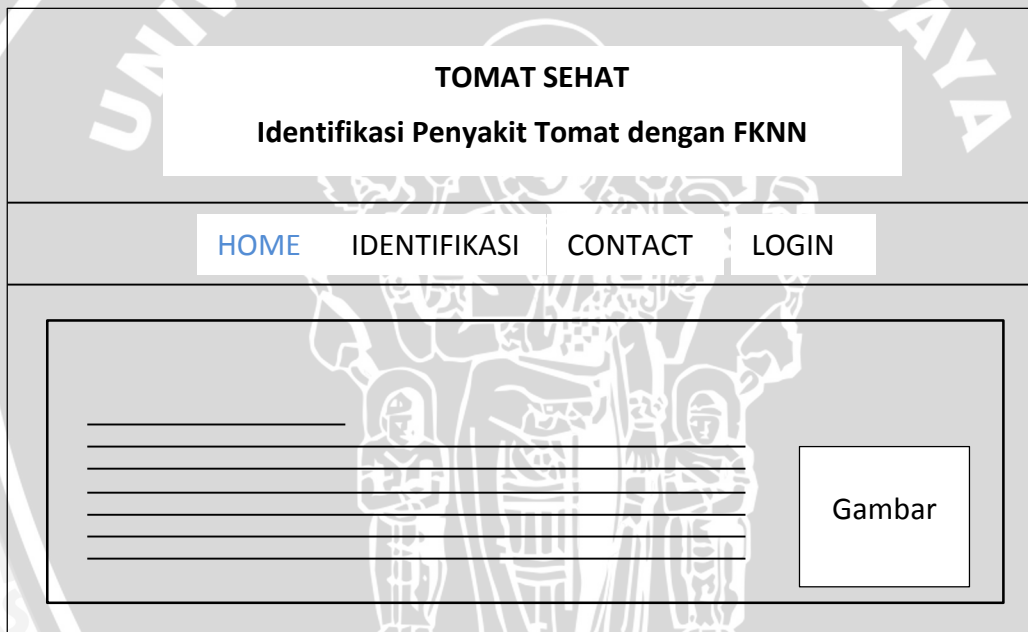
Dari perhitungan tersebut diperoleh nilai keanggotaan tertinggi pada P3, sehingga diperoleh kesimpulan bahwa data uji teridentifikasi penyakit 3 atau layu fusarium.

4.3.5 Perancangan Antarmuka

Perancangan antarmuka membahas tentang tampilan sistem yang akan dibangun. Antarmuka digunakan oleh pengguna untuk berkomunikasi seperti melihat informasi pada sistem, melakukan login, dan melakukan identifikasi. Pada perancangan antarmuka terdapat beberapa fitur untuk mempermudah pengguna dalam menggunakan sistem. Berikut ini merupakan perancangan antarmuka fitur pada sistem:

1. Antarmuka halaman *home*

Pada antarmuka halaman *home* terdapat fitur menu: *home*, *identifikasi*, *contact* dan *login*. Pada halaman *home* ini menampilkan informasi tentang sistem. Antarmuka halaman *home* digambarkan oleh Gambar 4.8.



Gambar 4. 8 Antarmuka halaman *home*

2. Antarmuka halaman *identifikasi*

Pada halaman *identifikasi* terdapat form *identifikasi* yang berisi gambar dan pertanyaan mengenai gejala penyakit. Setelah di-*submit*, akan ditampilkan form hasil perhitungan dan hasil dari *identifikasi* penyakit tanaman tomat beserta informasi mengenai penyakit tersebut. Antarmuka halaman *identifikasi* digambarkan oleh Gambar 4.9.

TOMAT SEHAT Identifikasi Penyakit Tomat dengan FKNN	
HOME	IDENTIFIKASI
CONTACT	LOGIN
Gambar Gambar Gambar	

Gambar 4. 9 Antarmuka halaman identifikasi

3. Antarmuka halaman *contact*

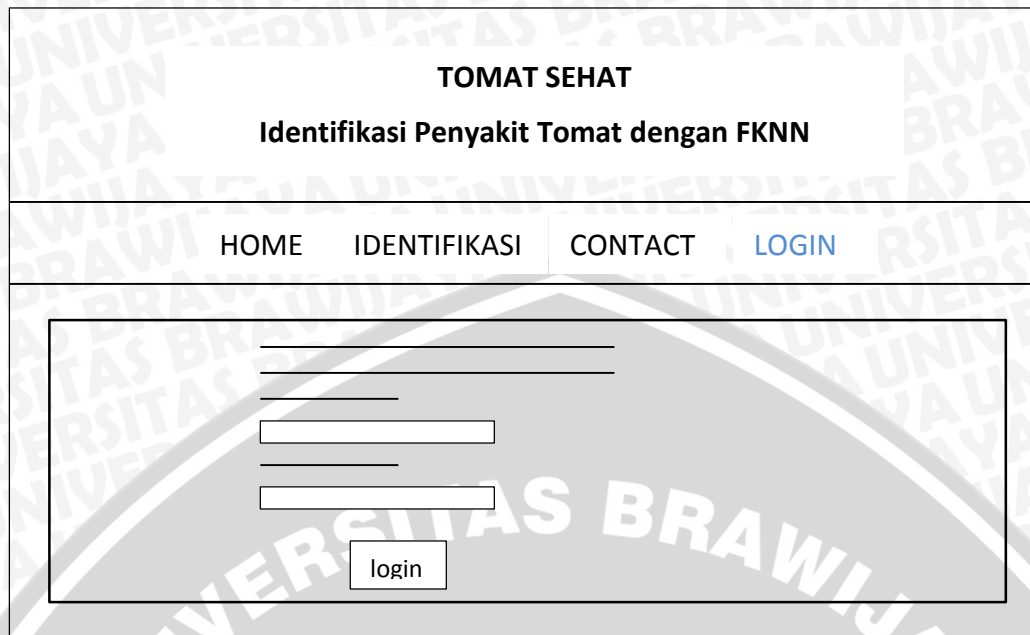
Pada halaman *contact* menampilkan informasi tentang pakar sebagai narasumber dari sistem ini. Antarmuka halaman *contact* digambarkan oleh Gambar 4.10.

TOMAT SEHAT Identifikasi Penyakit Tomat dengan FKNN	
HOME	IDENTIFIKASI
CONTACT	LOGIN
Foto	

Gambar 4. 10 Antarmuka halaman *contact*

4. Antarmuka halaman *login*

Pada halaman *login* terdapat form pengisian *username* dan *password* untuk admin. Hanya admin yang memiliki *username* dan *password* yang dapat masuk ke halaman admin. Antarmuka halaman *login* digambarkan oleh Gambar 4.11.



TOMAT SEHAT

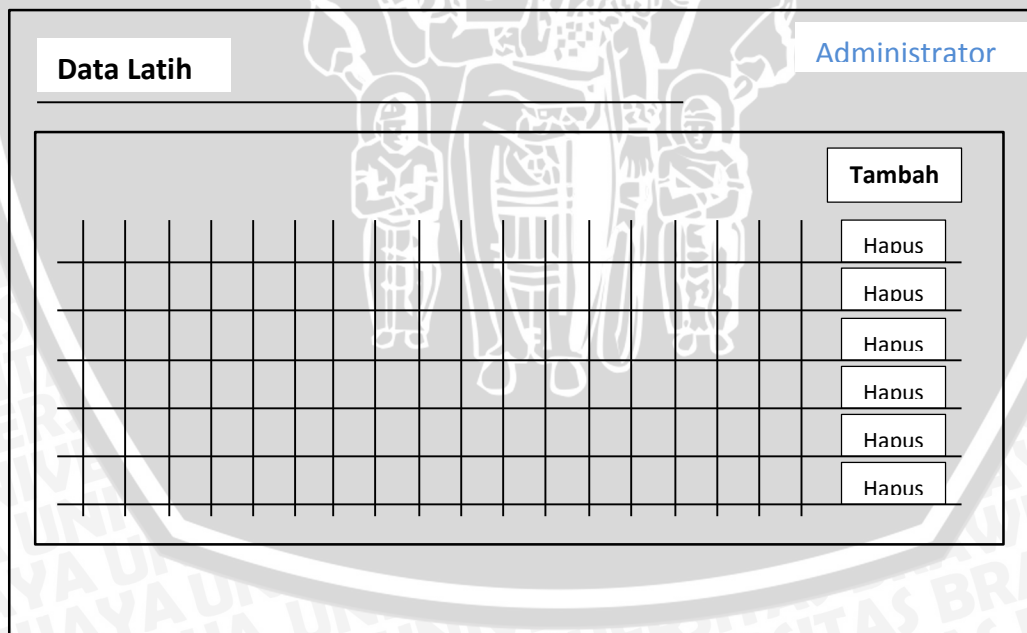
Identifikasi Penyakit Tomat dengan FKNN

HOME IDENTIFIKASI CONTACT [LOGIN](#)

Gambar 4. 11 Antarmuka halama login

5. Antarmuka halaman admin

Pada halaman admin menampilkan data latih dari basisdata, juga terdapat fitur hapus data dan tambah data. Pada pojok halaman terdapat status adminnistrators dan merupakan fitur untuk *logout*. Antarmuka halaman admin digambarkan oleh Gambar 4.12.



Data Latih
[Administrator](#)

Gambar 4. 12 Antarmuka halaman admin

6. Antarmuka halaman tambah data

Pada halaman tambah data menampilkan form untuk pengisian data yang ingin ditambah. Antarmuka halaman tambah data digambarkan oleh Gambar

Gambar 4. 13 Antarmuka halaman tambah data

4.3.6 Perancangan Pengujian

Pada tahap ini dilakukan perancangan pengujian terhadap sistem yang telah dibuat. Pengujian dilakukan dengan menggunakan pengujian fungsional sistem dan pengujian akurasi sistem. Pengujian fungsional sistem dilakukan dengan menggunakan metode *blackbox testing* untuk memeriksa apakah sistem telah berjalan sesuai dengan parameter yang telah ditentukan. Tabel 4.11 merupakan contoh tabel pengujian validasi sistem.

Tabel 4. 11 Perancangan pengujian fungsional sistem

No	Kasus uji	Yang Diharapkan	Yang didapatkan	Status
1.	login	User dapat masuk ke sistem	User dapat masuk ke sistem	valid
2.	...	...	...	(valid/tidak)
3.	...	...	...	(valid/tidak)

Pengujian akurasi diperoleh dari pengaruh nilai k dan data latih terhadap tingkat akurasi hasil klasifikasi. Pengujian dilakukan dengan nilai k yang bervariasi dari k=1, k=5, k=10, k=15, k=20, k=25, dan k=30. Data latih yang digunakan merupakan variasi data seimbang dan variasi data tidak seimbang. Variasi data seimbang yang digunakan berjumlah 48 data yang terdiri dari masing-masing jenis penyakit tanaman tomat dengan jumlah data yang sama. Variasi data tidak

seimbang berjumlah 30, 50, dan 70 data yang terdiri dari masing-masing jenis penyakit tanaman tomat namun dengan jumlah yang tidak sama. Sedangkan data uji yang digunakan berjumlah 30. Untuk mengukur tingkat akurasi dilakukan dengan cara membandingkan keluaran sistem dengan identifikasi pakar. Dari hasil membandingkan diperoleh berapa jumlah hasil keluaran sistem yang hasilnya sama dengan hasil identifikasi pakar. Sehingga akan diperoleh tingkat akurasi pada jumlah k tertentu dan pada variasi data latih tertentu. Tabel 4.12 merupakan tabel perancangan pengujian data seimbang. Tabel 4.13, Tabel 4.14, dan Tabel 4.15 merupakan tabel perancangan pengujian data tidak seimbang.

Tabel 4. 12 Perancangan pengujian 48 data seimbang

No	Data latih	Nilai akurasi sistem (%)						
		K=1	K=5	K=10	K=15	K=20	K=25	K=30
1	Variasi1							
2	Variasi2							
3	Variasi3							
4	Variasi4							
5	Variasi5							
Rata-rata								

Tabel 4. 13 Perancangan pengujian 30 data tidak seimbang

No	Data latih	Nilai akurasi sistem (%)						
		K=1	K=5	K=10	K=15	K=20	K=25	K=30
1	Variasi1							
2	Variasi2							
3	Variasi3							
Rata-rata								

Tabel 4. 14 Perancangan pengujian 50 data tidak seimbang

No	Data latih	Nilai akurasi sistem (%)						
		K=1	K=5	K=10	K=15	K=20	K=25	K=30
1	Variasi1							
2	Variasi2							
3	Variasi3							
Rata-rata								

Tabel 4. 15 Perancangan pengujian 70 data tidak seimbang

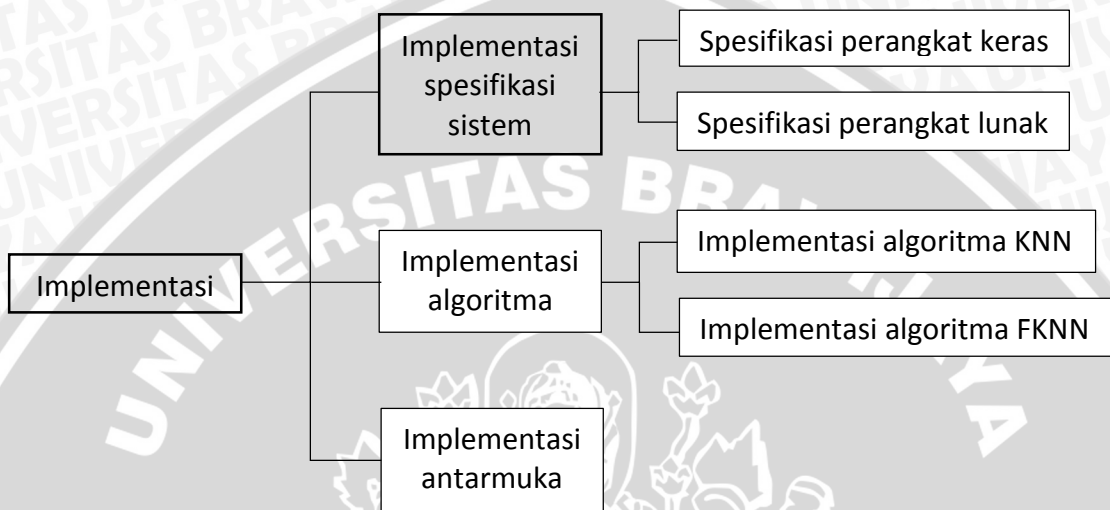
No	Data latih	Nilai akurasi sistem (%)						
		K=1	K=5	K=10	K=15	K=20	K=25	K=30
1	Variasi1							
2	Variasi2							
3	Variasi3							
Rata-rata								

Setelah diketahui hasil akurasi rata-rata dari masing-masing nilai k, maka akan dipilih nilai akurasi rata-rata terbesar sebagai akurasi terbaik dari sistem yang telah dibuat.



BAB 5 PEMBAHASAN

Bab ini membahas mengenai implementasi berdasarkan perancangan sistem yang telah dibuat pada tahapan sebelumnya. Pembahasan terdiri dari penjelasan tentang implementasi spesifikasi sistem, implementasi algoritma, dan implementasi antarmuka. Tahapan implementasi diilustrasikan pada pohon implementasi yang ditunjukkan pada gambar 5.1.



Gambar 5. 1 Pohon implementasi

5.1 Implementasi Spesifikasi Sistem

Hasil dari proses analisa kebutuhan dan perancangan sistem yang telah diuraikan pada bab sebelumnya menjadi acuan untuk proses implementasi sistem. Proses implementasi sistem membutuhkan spesifikasi perangkat yang sesuai agar sistem yang dibangun dapat berfungsi sesuai dengan kebutuhan. Spesifikasi perangkat yang digunakan terdiri dari spesifikasi perangkat keras dan perangkat lunak.

5.1.1 Spesifikasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan dalam implementasi sistem ditunjukkan pada tabel 5.1.

Tabel 5. 1 Spesifikasi perangkat keras

Nama Komponen	Spesifikasi
Komputer	Acer 4739Z
Prosesor	Intel Core i3-380M @ 2.53 GHz
Memori RAM	2 GB

5.1.2 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam implementasi sistem ditunjukkan pada tabel 5.2.

Tabel 5. 2 Spesifikasi Perangkat Lunak

Nama	Spesifikasi
<i>Tools Pemrograman</i>	<i>Sublime Text</i>
<i>Tools DBMS</i>	XAMPP v 3.1.0
<i>Internet Browser</i>	<i>Google Chrome</i>

5.2 Implementasi Algoritma

Implementasi sistem ini mempunyai dua proses utama yaitu proses perhitungan jarak dengan menggunakan metode KNN dan proses perhitungan nilai keanggotaan dengan menggunakan metode FKNN. Implementasi dari masing-masing metode tersebut dijelaskan sebagai berikut:

5.2.1 Implementasi Algoritma KNN

Pada proses ini akan dilakukan normalisasi data latih dan data uji. Setelah data dinormalisasi maka langkah selanjutnya adalah menghitung jarak euclien distance data uji terhadap semua data latih. Kemudian diambil data dengan jarak terkecil sebanyak K data. Tahapan implementasi metode KNN dijelaskan dengan Gambar 5.2.

```

1. //mengambil data latih dari database
2. $querylatih = mysql_query("SELECT * FROM data_latih order by
3. id_data");
4. $jumlahlatih = mysql_num_rows($querylatih);
5.
6. for ($i=1; $i<=$jumlahlatih; $i++) {
7.
8. $barislatih = mysql_fetch_array($querylatih);
9.
10. //proses menyimpan setiap nilai parameter data latih
11. ke variabel array
12.
13. $id[$i] = $barislatih['id_data'];
14. $G01latih[$i] = $barislatih['G01'];
15. $G02latih[$i] = $barislatih['G02'];
16. $G03latih[$i] = $barislatih['G03'];
17. $G04latih[$i] = $barislatih['G04'];
18. $G05latih[$i] = $barislatih['G05'];
19. $G06latih[$i] = $barislatih['G06'];
20. $G07latih[$i] = $barislatih['G07'];
21. $G08latih[$i] = $barislatih['G08'];
22. $G09latih[$i] = $barislatih['G09'];
23. $G10latih[$i] = $barislatih['G10'];
24. $G11latih[$i] = $barislatih['G11'];
25. $G12latih[$i] = $barislatih['G12'];
26. $G13latih[$i] = $barislatih['G13'];

```

```

27. $G14latih[$i] = $barislatih['G14'];
28. $G15latih[$i] = $barislatih['G15'];
29. $Identifikasi[$i] = $barislatih['Identifikasi'];
30.
31. }
32.
33. //mengambil data input user dari diagnosis.php sebagai data uji
34. $G01uji = $_POST['G01'];
35. $G02uji = $_POST['G02'];
36. $G03uji = $_POST['G03'];
37. $G04uji = $_POST['G04'];
38. $G05uji = $_POST['G05'];
39. $G06uji = $_POST['G06'];
40. $G07uji = $_POST['G07'];
41. $G08uji = $_POST['G08'];
42. $G09uji = $_POST['G09'];
43. $G10uji = $_POST['G10'];
44. $G11uji = $_POST['G11'];
45. $G12uji = $_POST['G12'];
46. $G13uji = $_POST['G13'];
47. $G14uji = $_POST['G14'];
48. $G15uji = $_POST['G15'];
49.
50. //menyimpan nilai parameter data uji ke variabel array
51. $G01latih[$jumlahlatih+1] = $G01uji;
52. $G02latih[$jumlahlatih+1] = $G02uji;
53. $G03latih[$jumlahlatih+1] = $G03uji;
54. $G04latih[$jumlahlatih+1] = $G04uji;
55. $G05latih[$jumlahlatih+1] = $G05uji;
56. $G06latih[$jumlahlatih+1] = $G06uji;
57. $G07latih[$jumlahlatih+1] = $G07uji;
58. $G08latih[$jumlahlatih+1] = $G08uji;
59. $G09latih[$jumlahlatih+1] = $G09uji;
60. $G10latih[$jumlahlatih+1] = $G10uji;
61. $G11latih[$jumlahlatih+1] = $G11uji;
62. $G12latih[$jumlahlatih+1] = $G12uji;
63. $G13latih[$jumlahlatih+1] = $G13uji;
64. $G14latih[$jumlahlatih+1] = $G14uji;
65. $G15latih[$jumlahlatih+1] = $G15uji;
66.
67. //mencari nilai max,min,range tiap parameter data
68. $minG01 = min($G01latih);
69. $maxG01 = max($G01latih);
70. $rangeG01 = $maxG01-$minG01;
71.
72. $minG02 = min($G02latih);
73. $maxG02 = max($G02latih);
74. $rangeG02 = $maxG02-$minG02;
75.
76. $minG03 = min($G03latih);
77. $maxG03 = max($G03latih);
78. $rangeG03 = $maxG03-$minG03;
79.
80. $minG04 = min($G04latih);
81. $maxG04 = max($G04latih);
82. $rangeG04 = $maxG04-$minG04;
83.
84. $minG05 = min($G05latih);
85. $maxG05 = max($G05latih);
86. $rangeG05 = $maxG05-$minG05;
87.
88. $minG06 = min($G06latih);
89. $maxG06 = max($G06latih);

```

```

90. $rangeG06 = $maxG06-$minG06;
91.
92. $minG07 = min($G07latih);
93. $maxG07 = max($G07latih);
94. $rangeG07 = $maxG07-$minG07;
95.
96. $minG08 = min($G08latih);
97. $maxG08 = max($G08latih);
98. $rangeG08 = $maxG08-$minG08;
99.
100. $minG09 = min($G09latih);
101. $maxG09 = max($G09latih);
102. $rangeG09 = $maxG09-$minG09;
103.
104. $minG10 = min($G10latih);
105. $maxG10 = max($G10latih);
106. $rangeG10 = $maxG10-$minG10;
107.
108. $minG11 = min($G11latih);
109. $maxG11 = max($G11latih);
110. $rangeG11 = $maxG11-$minG11;
111.
112. $minG12 = min($G12latih);
113. $maxG12 = max($G12latih);
114. $rangeG12 = $maxG12-$minG12;
115.
116. $minG13 = min($G13latih);
117. $maxG13 = max($G13latih);
118. $rangeG13 = $maxG13-$minG13;
119.
120. $minG14 = min($G14latih);
121. $maxG14 = max($G14latih);
122. $rangeG14 = $maxG14-$minG14;
123.
124. $minG15 = min($G15latih);
125. $maxG15 = max($G15latih);
126. $rangeG15 = $maxG15-$minG15;
127.
128. //normalisasi
129. for ($i=1; $i<=$jumlahlatih; $i++) {
130.
131. //normalisasi nilai setiap parameter data latih
132. $normalG01latih[$i] = ($G01latih[$i]-$minG01)/$rangeG01;
133. $normalG02latih[$i] = ($G02latih[$i]-$minG02)/$rangeG02;
134. $normalG03latih[$i] = ($G03latih[$i]-$minG03)/$rangeG03;
135. $normalG04latih[$i] = ($G04latih[$i]-$minG04)/$rangeG04;
136. $normalG05latih[$i] = ($G05latih[$i]-$minG05)/$rangeG05;
137. $normalG06latih[$i] = ($G06latih[$i]-$minG06)/$rangeG06;
138. $normalG07latih[$i] = ($G07latih[$i]-$minG07)/$rangeG07;
139. $normalG08latih[$i] = ($G08latih[$i]-$minG08)/$rangeG08;
140. $normalG09latih[$i] = ($G09latih[$i]-$minG09)/$rangeG09;
141. $normalG10latih[$i] = ($G10latih[$i]-$minG10)/$rangeG10;
142. $normalG11latih[$i] = ($G11latih[$i]-$minG11)/$rangeG11;
143. $normalG12latih[$i] = ($G12latih[$i]-$minG12)/$rangeG12;
144. $normalG13latih[$i] = ($G13latih[$i]-$minG13)/$rangeG13;
145. $normalG14latih[$i] = ($G14latih[$i]-$minG14)/$rangeG14;
146. $normalG15latih[$i] = ($G15latih[$i]-$minG15)/$rangeG15;
147. }
148.
149. //normalisasi nilai parameter data uji
150. $normalG01uji = ($G01uji-$minG01)/$rangeG01;
151. $normalG02uji = ($G02uji-$minG02)/$rangeG02;
152. $normalG03uji = ($G03uji-$minG03)/$rangeG03;

```

```

153. $normalG04uji = ($G04uji-$minG04)/$rangeG04;
154. $normalG05uji = ($G05uji-$minG05)/$rangeG05;
155. $normalG06uji = ($G06uji-$minG06)/$rangeG06;
156. $normalG07uji = ($G07uji-$minG07)/$rangeG07;
157. $normalG08uji = ($G08uji-$minG08)/$rangeG08;
158. $normalG09uji = ($G09uji-$minG09)/$rangeG09;
159. $normalG10uji = ($G10uji-$minG10)/$rangeG10;
160. $normalG11uji = ($G11uji-$minG11)/$rangeG11;
161. $normalG12uji = ($G12uji-$minG12)/$rangeG12;
162. $normalG13uji = ($G13uji-$minG13)/$rangeG13;
163. $normalG14uji = ($G14uji-$minG14)/$rangeG14;
164. $normalG15uji = ($G15uji-$minG15)/$rangeG15;
165.
166. //menghitung jarak euclidean distance
167. for ($i=1; $i<=$jumlahlatih; $i++) {
168.
169. //mencari selisih data latih dan data uji
170. $selisihG01[$i] = $normalG01uji-$normalG01latih[$i];
171. $selisihG02[$i] = $normalG02uji-$normalG02latih[$i];
172. $selisihG03[$i] = $normalG03uji-$normalG03latih[$i];
173. $selisihG04[$i] = $normalG04uji-$normalG04latih[$i];
174. $selisihG05[$i] = $normalG05uji-$normalG05latih[$i];
175. $selisihG06[$i] = $normalG06uji-$normalG06latih[$i];
176. $selisihG07[$i] = $normalG07uji-$normalG07latih[$i];
177. $selisihG08[$i] = $normalG08uji-$normalG08latih[$i];
178. $selisihG09[$i] = $normalG09uji-$normalG09latih[$i];
179. $selisihG10[$i] = $normalG10uji-$normalG10latih[$i];
180. $selisihG11[$i] = $normalG11uji-$normalG11latih[$i];
181. $selisihG12[$i] = $normalG12uji-$normalG12latih[$i];
182. $selisihG13[$i] = $normalG13uji-$normalG13latih[$i];
183. $selisihG14[$i] = $normalG14uji-$normalG14latih[$i];
184. $selisihG15[$i] = $normalG15uji-$normalG15latih[$i];
185.
186. //mencari jarak data latih dan data uji
187. $jarak[$i]=($selisihG01[$i]*$selisihG01[$i])+($selisihG02[$i]*
188. $selisihG02[$i])+($selisihG03[$i]*$selisihG03[$i])+($selisihG04[$i]*
189. $selisihG04[$i])+($selisihG05[$i]*$selisihG05[$i])+($selisihG06[$i]*
190. $selisihG06[$i])+($selisihG07[$i]*$selisihG07[$i])+($selisihG08[$i]*
191. $selisihG08[$i])+($selisihG09[$i]*$selisihG09[$i])+($selisihG10[$i]*
192. $selisihG10[$i])+($selisihG11[$i]*$selisihG11[$i])+($selisihG12[$i]*
193. $selisihG12[$i])+($selisihG13[$i]*$selisihG13[$i])+($selisihG14[$i]*
194. $selisihG14[$i])+($selisihG15[$i]*$selisihG15[$i]);
195.
196. $hasiljarak[$i] = sqrt($jarak[$i]);
197.
198. //Mengupdate jarak pada database
199. $update = mysql_query("UPDATE data_latih SET jarak = $hasiljarak[$i]
200. WHERE id_data = $i");
201. }

```

Gambar 5. 2 Implementasi Algoritma KNN

Penjelasan dari Gambar 5.2 adalah:

1. Baris ke-1 sampai baris ke-159 merupakan proses normalisasi data latih dan data uji.
2. Baris ke-159 sampai baris ke-192 merupakan proses perhitungan jarak eucliden.

5.2.2 Implementasi Algoritma FKNN

Setelah data diambil sebanyak K dari proses perhitungan KNN, maka langkah selanjutnya adalah proses FKNN dengan menghitung nilai keanggotaan tiap kelas seperti pada Gambar 5.3 berikut.

```

1.  $ka = 5;
2.
3.  //perhitungan nilai keanggotaan tiap kelas
4.  $querylatih1 = mysql_query("SELECT * FROM data_latih order by jarak ASC
5.  limit $ka");
6.
7.  while ($barislatih = mysql_fetch_assoc($querylatih1)) {
8.
9.  //menghitung jumlah kelas output pada k
10. if ($barislatih['Identifikasi']=='P1')$dataP01++;
11. elseif ($barislatih['Identifikasi']=='P2')$dataP02++;
12. elseif ($barislatih['Identifikasi']=='P3')$dataP03++;
13. elseif ($barislatih['Identifikasi']=='P4')$dataP04++;
14. elseif ($barislatih['Identifikasi']=='P5')$dataP05++;
15. else $dataP06++;
16. }
17.
18. $iyaP01 = 0.51+(($dataP01/$ka)*0.49);
19. $tidakP01 = ($dataP01/$ka)*0.49;
20. $iyaP02 = 0.51+(($dataP02/$ka)*0.49);
21. $tidakP02 = ($dataP02/$ka)*0.49;
22. $iyaP03 = 0.51+(($dataP03/$ka)*0.49);
23. $tidakP03 = ($dataP03/$ka)*0.49;
24. $iyaP04 = 0.51+(($dataP04/$ka)*0.49);
25. $tidakP04 = ($dataP04/$ka)*0.49;
26. $iyaP05 = 0.51+(($dataP05/$ka)*0.49);
27. $tidakP05 = ($dataP05/$ka)*0.49;
28. $iyaP06 = 0.51+(($dataP06/$ka)*0.49);
29. $tidakP06 = ($dataP06/$ka)*0.49;
30.
31. $querylatih2 = mysql_query("SELECT * FROM data_latih order by jarak ASC
32. limit $ka");
33.
34. while ($barislatih = mysql_fetch_assoc($querylatih2)) {
35.
36. //perhitungan nilai keanggotaan data uji pada kelas P01
37. if ($barislatih['Identifikasi']=='P1')$membershipP01=$iyaP01;
38. elseif ($barislatih['Identifikasi']!='P1')$membershipP01=$tidakP01;
39. else { $membershipP01=0; $hasiljarak=0; }
40. $pangkat = -2;
41. $resultjarak1=pow($hasiljarak,$pangkat);
42. if ($hasiljarak==0)continue;
43. $totaljarak1+= $resultjarak1;
44. $fuzzyP01+= $membershipP01*$resultjarak1;
45. $hasilfuzzyP01 = $fuzzyP01/$totaljarak1;
46.
47. //perhitungan nilai keanggotaan data uji pada kelas P02
48. if ($barislatih['Identifikasi']=='P2')$membershipP02=$iyaP02;
49. elseif ($barislatih['Identifikasi']!='P2')$membershipP02=$tidakP02;
50. else { $membershipP02=0; $hasiljarak=0; }
51. $pangkat = -2;
52. $resultjarak2=pow($hasiljarak,$pangkat);
53. if ($hasiljarak==0)continue;
54. $totaljarak2+= $resultjarak2;
55. $fuzzyP02+= $membershipP02*$resultjarak2;
56. $hasilfuzzyP02 = $fuzzyP02/$totaljarak2;

```

```

57.
58. //perhitungan nilai keanggotaan data uji pada kelas P03
59. if( $barislatih['Identifikasi']=='P3')$membershipP03=$iyaP03;
60. elseif ($barislatih['Identifikasi']!='P3')$membershipP03=$tidakP03;
61. else { $membershipP03=0; $hasiljarak=0; }
62. $pangkat = -2;
63. $resultjarak3=pow($hasiljarak,$pangkat);
64. if ($hasiljarak==0)continue;
65. $totaljarak3+= $resultjarak3;
66. $fuzzyP03+= $membershipP03*$resultjarak3;
67. $hasilfuzzyP03 = $fuzzyP03/$totaljarak3;
68.
69. //perhitungan nilai keanggotaan data uji pada kelas P04
70. if( $barislatih['Identifikasi']=='P4')$membershipP04=$iyaP04;
71. elseif ($barislatih['Identifikasi']!='P4')$membershipP04=$tidakP04;
72. else { $membershipP04=0; $hasiljarak=0; }
73. $pangkat = -2;
74. $resultjarak4=pow($hasiljarak,$pangkat);
75. if ($hasiljarak==0)continue;
76. $totaljarak4+= $resultjarak4;
77. $fuzzyP04+= $membershipP04*$resultjarak4;
78. $hasilfuzzyP04 = $fuzzyP04/$totaljarak4;
79.
80. //perhitungan nilai keanggotaan data uji pada kelas P05
81. if( $barislatih['Identifikasi']=='P5')$membershipP05=$iyaP05;
82. elseif ($barislatih['Identifikasi']!='P5')$membershipP05=$tidakP05;
83. else { $membershipP05=0; $hasiljarak=0; }
84. $pangkat = -2;
85. $resultjarak5=pow($hasiljarak,$pangkat);
86. if ($hasiljarak==0)continue;
87. $totaljarak5+= $resultjarak5;
88. $fuzzyP05+= $membershipP05*$resultjarak5;
89. $hasilfuzzyP05 = $fuzzyP05/$totaljarak5;
90.
91. //perhitungan nilai keanggotaan data uji pada kelas P06
92. if( $barislatih['Identifikasi']=='P6')$membershipP06=$iyaP06;
93. elseif ($barislatih['Identifikasi']!='P6')$membershipP06=$tidakP06;
94. else { $membershipP06=0; $hasiljarak=0; }
95. $pangkat = -2;
96. $resultjarak6=pow($hasiljarak,$pangkat);
97. if ($hasiljarak==0)continue;
98. $totaljarak6+= $resultjarak6;
99. $fuzzyP06+= $membershipP06*$resultjarak6;
100. $hasilfuzzyP06 = $fuzzyP06/$totaljarak6;
101. }
102.
103. //mencari nilai keanggotaan terbesar
104. if ($hasilfuzzyP01>=$hasilfuzzyP02 && $hasilfuzzyP01>=$hasilfuzzyP03 &&
105. $hasilfuzzyP01>=$hasilfuzzyP04 && $hasilfuzzyP01>=$hasilfuzzyP05 &&
106. $hasilfuzzyP01>=$hasilfuzzyP06) {
107. $prediksifuzzy = 'Busuk Daun';
108. }
109.
110.
111. Elseif($hasilfuzzyP02>=$hasilfuzzyP01 && $hasilfuzzyP02>=$hasilfuzzyP03
112. && $hasilfuzzyP02>=$hasilfuzzyP04 && $hasilfuzzyP02>=$hasilfuzzyP05 &&
113. $hasilfuzzyP02>=$hasilfuzzyP06) {
114. $prediksifuzzy = 'Bercak Coklat';
115. }
116.
117. elseif($hasilfuzzyP03>=$hasilfuzzyP01 && $hasilfuzzyP03>=$hasilfuzzyP02
118. && $hasilfuzzyP03>=$hasilfuzzyP04 && $hasilfuzzyP03>=$hasilfuzzyP05 &&
119. $hasilfuzzyP03>=$hasilfuzzyP06) {

```

120.	\$prediksifuzzy = 'Layu Fusarium';
121.	}
122.	
123.	elseif(\$hasilfuzzyP04>=\$hasilfuzzyP01 && \$hasilfuzzyP04>=\$hasilfuzzyP02
124.	&& \$hasilfuzzyP04>=\$hasilfuzzyP03 && \$hasilfuzzyP04>=\$hasilfuzzyP05 &&
125.	\$hasilfuzzyP04>=\$hasilfuzzyP06) {
126.	\$prediksifuzzy = 'Layu Bakteri';
127.	}
128.	
129.	elseif(\$hasilfuzzyP05>=\$hasilfuzzyP01 && \$hasilfuzzyP05>=\$hasilfuzzyP02
130.	&& \$hasilfuzzyP05>=\$hasilfuzzyP03 && \$hasilfuzzyP05>=\$hasilfuzzyP04 &&
131.	\$hasilfuzzyP05>=\$hasilfuzzyP06) {
132.	\$prediksifuzzy = 'Mozaik';
133.	}
134.	
135.	elseif(\$hasilfuzzyP06>=\$hasilfuzzyP01 && \$hasilfuzzyP06>=\$hasilfuzzyP02
136.	&& \$hasilfuzzyP06>=\$hasilfuzzyP03 && \$hasilfuzzyP06>=\$hasilfuzzyP04 &&
137.	\$hasilfuzzyP06>=\$hasilfuzzyP05) {
138.	\$prediksifuzzy = 'Keriting';
139.	}

Gambar 5. 3 Implementasi Algoritma FKNN

Penjelasan dari Gambar 5.3 adalah:

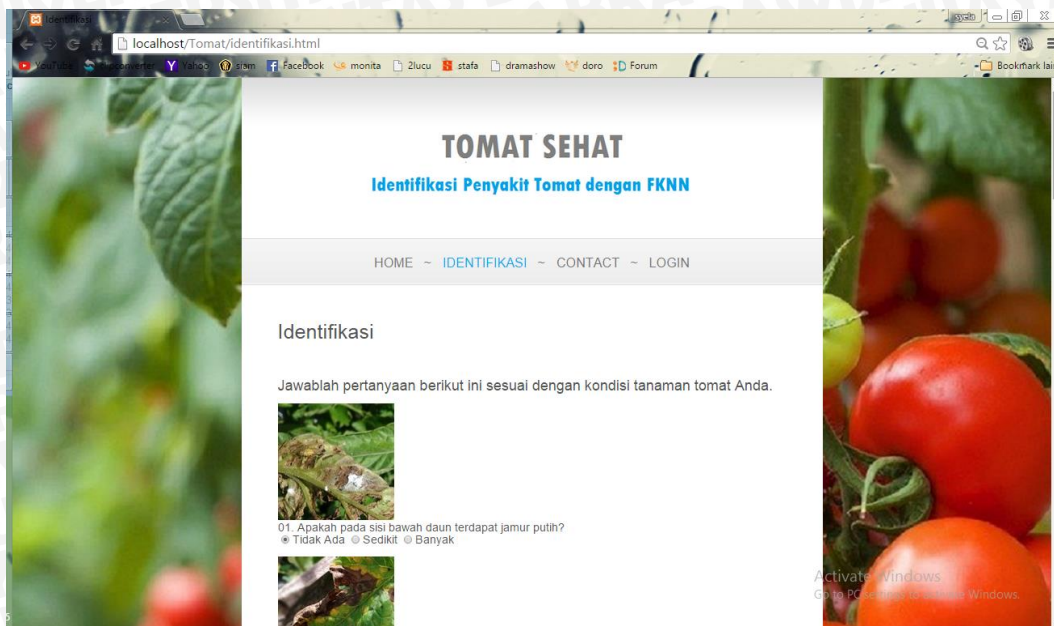
1. Baris ke-1 sampai baris ke-106 merupakan proses perhitungan nilai keanggotaan pada setiap kelas.
2. Baris ke-108 sampai baris ke-138 merupakan proses mencari nilai keanggotaan terbesar untuk dijadikan output sistem.

5.3 Implementasi Program

Antarmuka merupakan media yang digunakan oleh pengguna untuk berinteraksi dengan sistem. Pada implementasi antarmuka ini, tidak semua antarmuka halaman pada sistem akan dijelaskan, tetapi hanya beberapa halaman tertentu saja. Antarmuka halaman yang akan dijelaskan pada implementasi ini antara lain halaman identifikasi, halaman hasil identifikasi, halaman *login*, halaman admin, dan halaman tambah data.

5.3.1 Implementasi Antarmuka Halaman Identifikasi

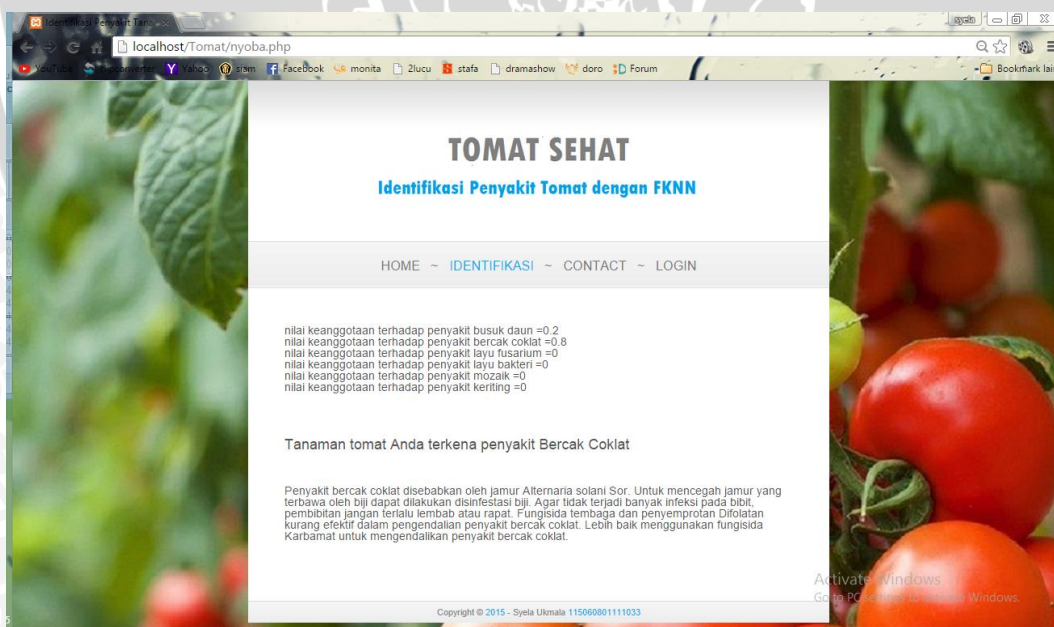
Halaman identifikasi merupakan halaman pertanyaan mengenai gejala tanaman tomat sebagai input sistem untuk proses perhitungan. Gambar ditampilkan sebagai contoh gejala yang tampak. Pengguna menjawab pertanyaan sesuai dengan kondisi tanaman tomat. Implementasi antarmuka halaman identifikasi ditunjukkan pada gambar 5.4.



Gambar 5. 4 Implementasi antarmuka halaman identifikasi

5.3.2 Implementasi Antarmuka Halaman Hasil Identifikasi

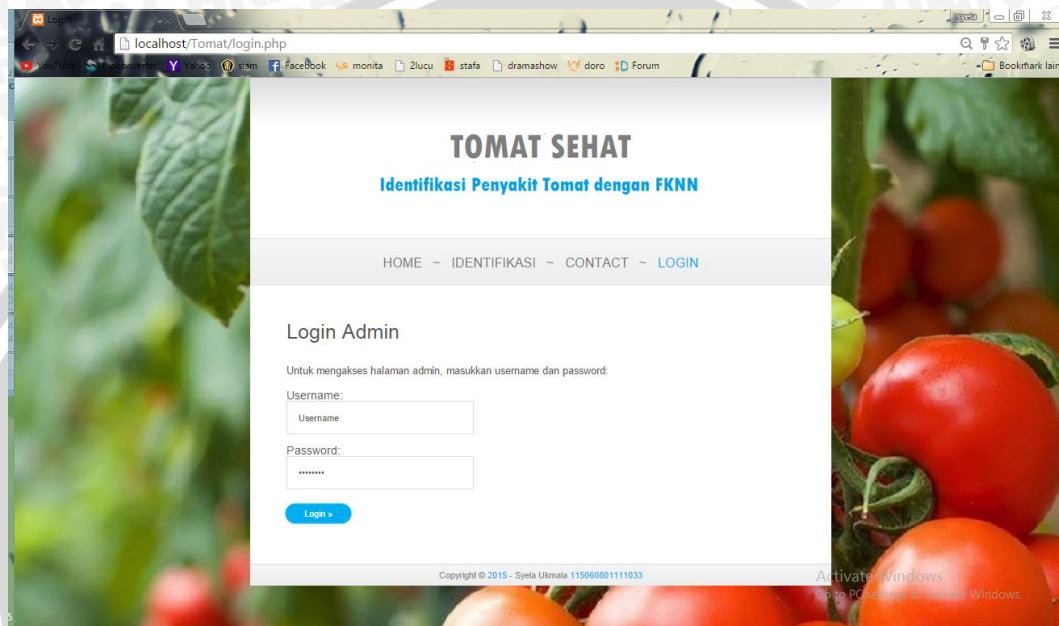
Halaman hasil identifikasi menampilkan nilai keanggotaan terhadap semua kelas penyakit, hasil identifikasi, dan penanggulangan penyakit. Halaman hasil identifikasi ditampilkan setelah pengguna mengirim jawaban pertanyaan gejala. Implementasi antarmuka halaman hasil identifikasi terlihat seperti pada Gambar 5.5.



Gambar 5. 5 Implementasi antarmuka halaman hasil identifikasi

5.3.3 Implementasi Antarmuka Halaman Login

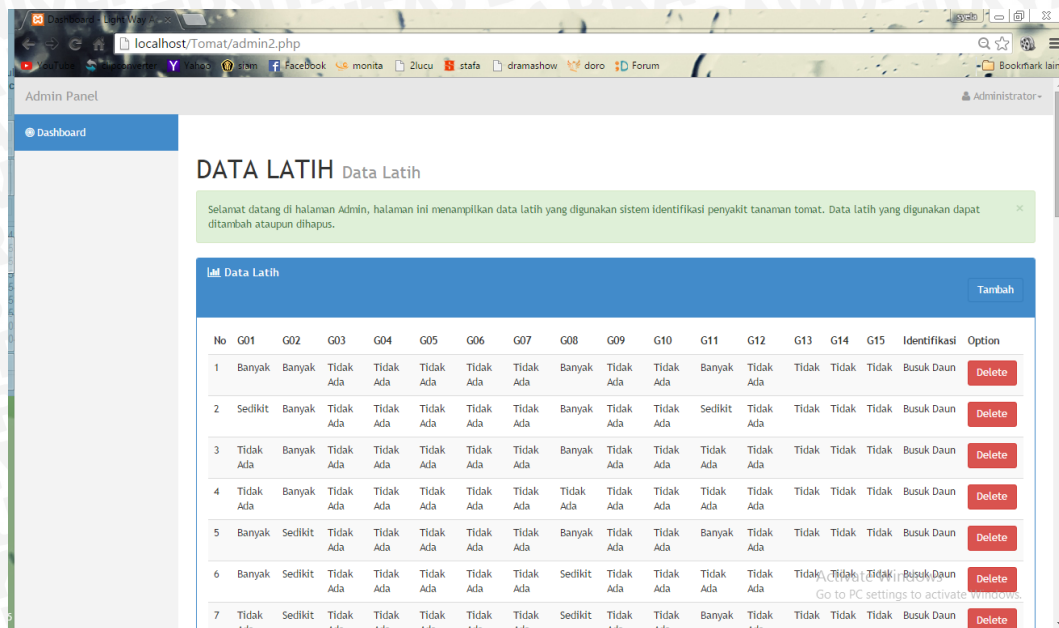
Halaman login merupakan halaman yang digunakan oleh pengguna untuk memasuki sistem agar bisa mengakses halaman sistem sesuai dengan hak aksesnya. Pengguna dapat memulai proses *login* dengan cara memasukkan *username* dan *password* ke dalam form yang tersedia pada halaman *login*. Setelah *username* dan *password* dimasukkan maka sistem akan mencocokkannya dengan *username* dan *password* yang telah tersimpan dalam database. Implementasi antarmuka halaman *login* ditunjukkan pada Gambar 5.6.



Gambar 5. 6 Implementasi antarmuka halaman login

5.3.4 Implementasi Antarmuka Halaman Admin

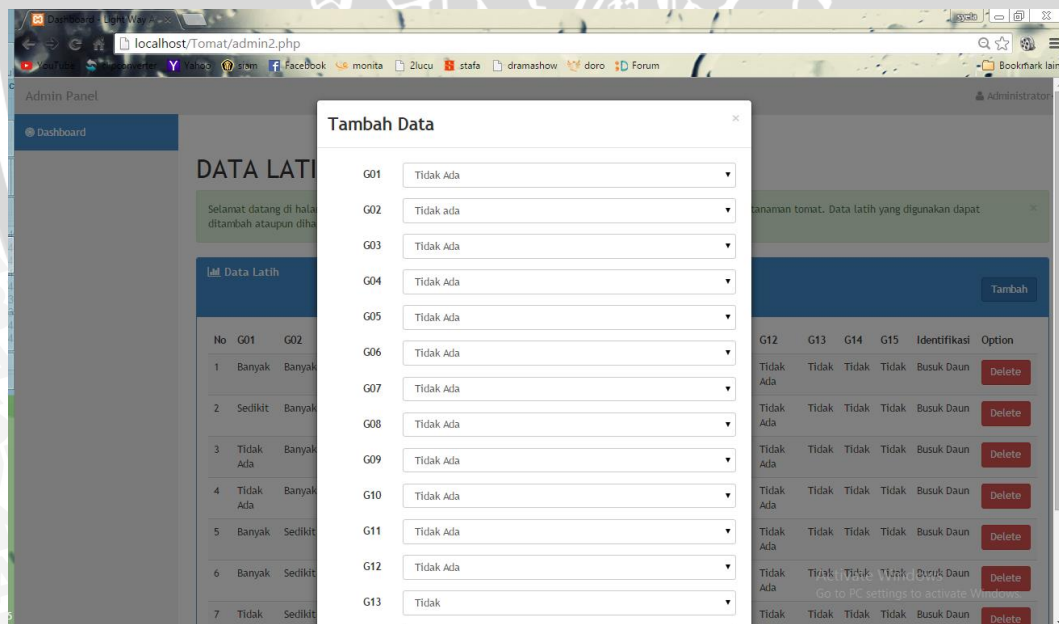
Antarmuka halaman admin menampilkan data latih yang digunakan pada sistem. didalam halaman sistem terdapat fitur tambah data dan hapus data. Implementasi antarmuka halaman admin terlihat seperti Gambar 5.7.



Gambar 5. 7 Implementasi antarmuka halaman admin

5.3.5 Implementasi Antarmuka Halaman Tambah Data

Fitur tambah data terdapat didalam halaman admin. Untuk menambahkan data baru, admin hanya perlu memasukkan gejala pada form yang tersedia. Implementasi antarmuka halaman tambah data seperti pada Gambar 5.8.



Gambar 5. 8 Implementasi antarmuka halaman tambah data

BAB 6 PENGUJIAN

Bab ini membahas mengenai tahap pengujian sistem. Tahap pengujian dilakukan melalui dua tahap pengujian, yaitu pengujian fungsional dan pengujian akurasi sistem. Pengujian fungsional dilakukan untuk mengetahui apakah sistem yang dibuat telah menyediakan fungsi-fungsi sesuai dengan yang dibutuhkan. Pengujian akurasi sistem dilakukan dengan mencocokkan hasil dari identifikasi sistem dengan hasil identifikasi dari pakar dengan nilai k tertentu.

6.1 Pengujian Fungsional

Pengujian fungsional dilakukan untuk mengetahui apakah sistem yang dibuat telah menyediakan fungsi-fungsi sesuai dengan yang dibutuhkan. Pengujian yang dilakukan tidak difokuskan pada jalannya algoritma program melainkan pada kesesuaian antara kinerja aplikasi dengan daftar kebutuhan. Pada setiap kebutuhan dilakukan proses pengujian dengan kasus uji masing-masing untuk mengetahui kesesuaian antara fungsi aplikasi dengan daftar kebutuhan.

6.1.1 Prosedur Pengujian Fungsional

Pengujian fungsional dilakukan dengan menggunakan kasus uji terhadap beberapa fitur utama. Beberapa kasus uji yang dilakukan yaitu:

1. Kasus uji *login*

Kasus uji *login* menjelaskan tentang pengujian fungsional proses *login*. Kasus uji *login* terdiri dari pengujian fungsional *login* sukses dan *login* gagal. Penjelasan kasus uji *login* ditunjukkan pada Tabel 6.1 dan Tabel 6.2.

Tabel 6. 1 Penjelasan kasus uji login sukses

Nama Kasus Uji	Kasus Uji <i>Login</i> sukses
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas <i>login</i> agar pengguna dapat masuk/menggunakan sistem sesuai dengan hak akses.
Prosedur Uji	1. Sistem dijalankan ketika program dieksekusi 2. Pengguna masuk ke halaman home 3. Pengguna memilih menu <i>login</i> 4. Pengguna mengisi <i>username</i> dan <i>password</i> ke dalam form yang telah disediakan 5. Pengguna menekan tombol <i>login</i>
Hasil yang diharapkan	Sistem dapat menampilkan halaman admin sesuai dengan hak akses pengguna jika data yang dimasukkan ketika <i>login</i> sesuai dengan data pada <i>database</i> .

Tabel 6. 2 Penjelasan kasus uji login gagal

Nama Kasus Uji	Kasus Uji <i>Login</i> Gagal
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas <i>login</i> agar pengguna dapat masuk/menggunakan sistem sesuai dengan hak akses.
Prosedur Uji	1. Sistem dijalankan ketika program dieksekusi 2. Pengguna masuk ke halaman home 3. Pengguna memilih menu <i>login</i> 4. Pengguna mengisi <i>username</i> dan <i>password</i> ke dalam form yang telah disediakan 5. Pengguna menekan tombol <i>login</i>
Hasil yang diharapkan	Sistem tidak menampilkan halaman admin dan tetap berada pada halaman login.

2. Kasus uji *logout*

Kasus uji *logout* menjelaskan tentang pengujian fungsional proses *logout* . Penjelasan kasus uji ditunjukkan pada Tabel 6.3.

Tabel 6. 3 Penjelasan kasus uji logout

Nama Kasus Uji	Kasus Uji <i>Logout</i>
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas <i>logout</i> bagi <i>admin</i> agar dapat keluar dari halaman admin pada sistem
Prosedur Uji	1. <i>Admin</i> berada pada halaman sesuai hak akses <i>admin</i> 2. <i>Admin</i> menekan tombol <i>logout</i>
Hasil yang diharapkan	<i>Admin</i> keluar dari halaman <i>admin</i> dan ditampilkan halaman utama sistem

3. Kasus uji menu data latih

Kasus uji menu data latih menjelaskan tentang pengujian yang terdiri dari pengujian fungsional menampilkan data latih yang ditunjukkan pada tabel 6.4, menambahkan data baru ditunjukkan pada Tabel 6.5 dan menghapus data ditunjukkan pada Tabel 6.6

Tabel 6. 4 Penjelasan kasus uji menampilkan data latih

Nama Kasus Uji	Kasus Uji Menampilkan Data Latih
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat menampilkan data latih yang telah disimpan dalam <i>database</i> .
Prosedur Uji	1. <i>Admin</i> telah melakukan <i>login</i> ke dalam sistem 2. Pada halaman admin ditampilkan tabel data latih
Hasil yang diharapkan	Sistem dapat menampilkan tabel data latih yang ada dalam basis data.

Tabel 6. 5 Penjelasan kasus uji menambahkan data latih

Nama Kasus Uji	Kasus Uji Menambahkan Data Latih
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat menambahkan data latih baru ke dalam <i>database</i>
Prosedur Uji	1. <i>Admin</i> telah melakukan <i>login</i> ke dalam sistem 2. <i>Admin</i> memilih menu Tambah data
Hasil yang diharapkan	Sistem dapat menambahkan data latih baru ke dalam <i>database</i> .

Tabel 6. 6 Penjelasan kasus uji menghapus data latih

Nama Kasus Uji	Kasus Uji Menghapus Data Latih
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat menghapus data latih dari <i>database</i>
Prosedur Uji	1. <i>Admin</i> telah melakukan <i>login</i> ke dalam sistem 2. <i>Admin</i> memilih hapus data
Hasil yang diharapkan	Sistem dapat menghapus data latih dari <i>database</i>

4. Kasus uji menu identifikasi

Kasus uji menu identifikasi menjelaskan tentang pengujian yang terdiri dari pengujian fungsional menampilkan *form* identifikasi penyakit, dan menampilkan hasil identifikasi penyakit. Penjelasan kasus uji tersebut ditunjukkan pada Tabel 6.7 dan Tabel 6.8.

Tabel 6. 7 Penjelasan uji menampilkan form identifikasi penyakit

Nama Kasus Uji	Kasus Uji Menampilkan Form Identifikasi Penyakit
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat menampilkan <i>form</i> identifikasi bagi pengguna untuk memasukkan gejala yang nampak pada tanaman.
Prosedur Uji	1. Pengguna memilih menu identifikasi
Hasil yang diharapkan	Sistem menampilkan <i>form</i> pertanyaan dan gambar gejala penyakit

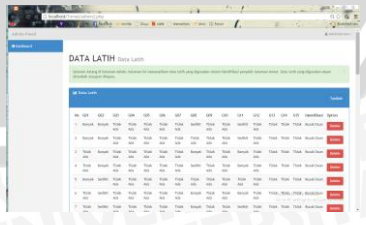
Tabel 6. 8 Penjelasan kasus uji menampilkan hasil identifikasi penyakit

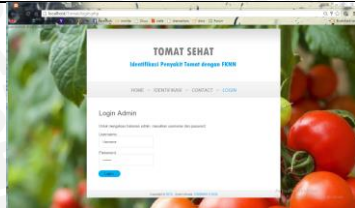
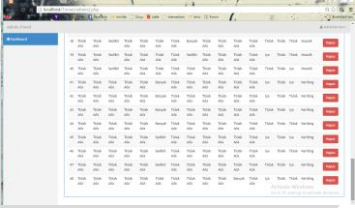
Nama Kasus Uji	Kasus Uji Menampilkan Hasil Identifikasi Penyakit
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat menampilkan hasil identifikasi penyakit sesuai dengan gejala yang telah dimasukkan oleh pengguna
Prosedur Uji	1. Pengguna telah menekan tombol pilihan di setiap pertanyaan pada form identifikasi 2. Pengguna menekan tombol identifikasi pada <i>form</i> identifikasi
Hasil yang diharapkan	Sistem dapat menampilkan hasil identifikasi penyakit dan penanganan dari penyakit tersebut

6.1.2 Hasil dan Analisis Pengujian Fungsional

Berdasarkan kasus uji yang telah dilaksanakan sesuai dengan prosedur pengujian, maka didapatkan hasil seperti yang ditunjukkan pada Tabel 6.9.

Tabel 6. 9 Hasil pengujian fungsional

No.	Nama Kasus Uji	Hasil yang Diharapkan	Hasil yang Didapatkan	Status
1.	<i>Login</i>	Sistem dapat menampilkan halaman admin sesuai dengan hak akses pengguna jika data yang dimasukkan ketika login sesuai dengan data pada <i>database</i> .	Sistem dapat menampilkan halaman admin. 	valid
2.	<i>Login gagal</i>	Sistem tidak menampilkan halaman admin dan tetap berada pada halaman login.	Sistem tetap berada pada halaman login.	valid

				
3.	Logout	Admin keluar dari halaman admin dan ditampilkan halaman utama sistem.	Sistem menampilkan halaman utama. 	valid
4.	Menampilkan data latih	Sistem dapat menampilkan data latih yang ada dalam basis data.	Sistem menampilkan tabel data latih. 	valid
5.	Menampilkan menghapus latih	Sistem dapat menghapus data latih dari basis data.	Sistem dapat menghapus data latih. 	valid
6.	Menambahkan data latih	Sistem dapat menambahkan data latih baru ke dalam basis data.	Sistem dapat menambahkan data latih ke dalam basis data. 	valid
7.	Menampilkan form identifikasi penyakit	Sistem dapat menampilkan form pertanyaan dan gambar gejala penyakit.	Sistem menampilkan form pertanyaan dan gambar gejala penyakit. 	valid

8.	Menampilkan hasil identifikasi penyakit	Sistem dapat menampilkan hasil identifikasi penyakit dan penanganan dari penyakit tersebut.	Sistem menampilkan hasil identifikasi penyakit dan penanganan dari penyakit tersebut.	valid
----	---	---	---	-------



Hasil pengujian fungsional dilakukan dengan melihat kesesuaian antara hasil yang diharapkan berdasarkan kebutuhan sistem dengan hasil yang didapatkan dari kinerja sistem. Berdasarkan hasil dari delapan kasus uji yang telah dilakukan, masing-masing kasus uji menghasilkan kesesuaian dengan kebutuhan sistem. Hal ini menunjukkan bahwa fungsional sistem telah berjalan dengan baik dan sesuai dengan kebutuhan yang diperlukan.

6.2 Pengujian Akurasi Sistem

Pengujian akurasi dilakukan untuk mengetahui seberapa besar keakuratan hasil *output* sistem dalam mengidentifikasi penyakit tanaman tomat. Pengujian akurasi sistem terbatas dilakukan terhadap variasi data latih dan nilai k. Sehingga nantinya akan dipilih variasi data latih dan nilai k dengan akurasi terbaik yang digunakan untuk sistem.

6.2.1 Prosedur Pengujian Akurasi Sistem

Pengujian dilakukan dengan menggunakan variasi data latih dan variasi nilai k. Jumlah data uji yang digunakan pada pengujian ini adalah 30 data. Jumlah data latih yang digunakan sama yaitu 48, namun dengan 5 variasi data latih berbeda yang diambil dari 188 data latih. Setiap variasi data latih akan diuji dengan nilai k, mulai dari k=1, k=5, k=10, k=15, k=20, k=25, hingga k=30. Penghitungan nilai akurasi menggunakan Persamaan 2.5 dengan menghitung jumlah keluaran sistem yang sama dengan identifikasi pakar dibagi dengan jumlah data uji. Tabel 6.10 merupakan tabel data uji yang telah diidentifikasi oleh pakar. Tabel 6.11 merupakan tabel variasi data latih seimbang. Sedangkan Tabel 6.12, Tabel 6.13, dan Tabel 6.14 merupakan tabel variasi data latih tidak seimbang.

Tabel 6. 10 Data uji

no	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	penyakit
1				B			B					B				P2
2			B					B					B			P5
3					B				B				B			P3
4					B							B		B		P3,P4
5										B			B	B		P4
6		B						B			B					P1
7						B								B	B	P6
8	B	B									B					P1
9					B				B					B		P3
10					B					B						P4
11			B												B	P5
12						B									B	P6
13				B	B			B					B			P2
14		B						B								P1
15									B					B		P3
16					B				B							P3
17			B										B			P5
18					B								B	B		P3,P4
19						B		B					B			P6
20										B			B			P4
21				B			B					B		B		P2
22	B	B											B		B	P1
23			B										B	B		P3
24				B										B		P3
25						B							B			P6
26			B										B		B	P5
27	B							B			B					P1
28	B			B									B			P4
29					B					B			B			P4
30		B								B			B	B		P3

Sumber: Pakar

Keterangan:

B = menunjukkan adanya gejala dengan jumlah yang banyak

P1 = Penyakit bususk daun

P2 = Penyakit bercak coklat

P3 = Penyakit layu fusarium

P4 = Penyakit layu bakteri

P5 = Penyakit mosaik

P6 = Penyakit keriting

Tabel 6. 11 Variasi 48 data latih seimbang

no	Data Latih				
	variasi1	variasi2	variasi3	variasi4	variasi5
1	2	3	19	1	3
2	14	4	19	3	5
3	17	6	21	8	7
4	18	12	22	14	9
5	19	19	24	15	14
6	23	27	26	16	19
7	25	28	29	17	28
8	29	29	32	19	29
9	38	33	37	29	33
10	40	34	40	33	34
11	41	39	41	37	37
12	44	40	42	39	40
13	46	41	53	40	41
14	50	48	56	41	46
15	53	49	59	54	47
16	55	52	61	57	48
17	70	64	63	65	51
18	71	68	64	67	55
19	78	69	68	69	59
20	80	70	70	70	62
21	85	74	71	78	70
22	95	77	77	80	72
23	97	78	78	81	76
24	100	80	82	86	78
25	101	83	86	91	79
26	104	88	87	96	80
27	109	90	89	98	83
28	113	94	104	103	92
29	114	96	105	109	93
30	118	102	109	113	104
31	119	107	112	121	106
32	124	109	116	123	107
33	130	117	124	124	109
34	131	135	126	134	110
35	133	136	129	141	120
36	137	138	133	143	125
37	142	147	137	144	128
38	146	148	141	153	138
39	157	148	155	160	143
40	159	156	159	164	145
41	160	160	160	169	154
42	162	167	164	171	158
43	163	168	164	172	160
44	166	171	168	173	161
45	169	177	170	176	168
46	171	179	171	182	176
47	173	183	181	185	178
48	182	187	188	186	183

Tabel 6. 12 Variasi 30 data latih tidak seimbang

no	Data Latih		
	variasi1	variasi2	variasi3
1	9	8	7
2	11	12	16
3	32	14	19
4	34	15	29
5	39	17	31
6	47	19	40
7	53	21	41
8	59	29	49
9	68	40	70
10	69	41	73
11	70	44	74
12	76	83	77
13	78	93	78
14	84	104	80
15	86	109	89
16	88	111	101
17	91	112	109
18	96	124	117
19	105	133	118
20	115	136	120
21	130	138	123
22	136	141	128
23	137	146	140
24	146	159	147
25	151	160	154
26	153	162	155
27	163	171	160
28	168	173	167
29	184	180	175
30	106	183	182

Tabel 6. 13 Variasi 50 data latih tidak seimbang

no	Data Latih		
	variasi1	variasi2	variasi3
1	8	7	7
2	9	8	9
3	11	12	11
4	12	14	16
5	14	15	19
6	17	16	29
7	21	17	31
8	32	19	32
9	34	21	34
10	39	29	40
11	47	31	41
12	53	40	47
13	59	41	49
14	68	44	53
15	69	49	59

16	70	73	70
17	76	74	73
18	78	77	74
19	83	80	76
20	84	83	77
21	86	93	78
22	88	101	80
23	91	104	84
24	93	109	88
25	96	111	89
26	104	112	91
27	105	117	101
28	111	118	105
29	112	120	109
30	115	123	115
31	124	124	117
32	130	128	118
33	133	133	120
34	136	136	123
35	136	138	128
36	137	140	136
37	138	141	140
38	146	146	147
39	146	154	151
40	151	155	153
41	153	159	154
42	159	160	155
43	162	162	160
44	163	167	163
45	168	171	167
46	171	173	168
47	180	175	175
48	183	180	182
49	184	182	184
50	106	183	106

Tabel 6. 14 Variasi 70 data latih tidak seimbang

no	Data Latih		
	variasi1	variasi2	variasi3
1	3	3	8
2	5	5	9
3	7	7	10
4	9	9	11
5	14	14	12
6	17	17	14
7	19	19	15
8	24	23	17
9	28	24	18
10	29	28	19
11	33	33	21
12	34	34	29
13	37	37	32
14	40	39	34
15	41	40	39

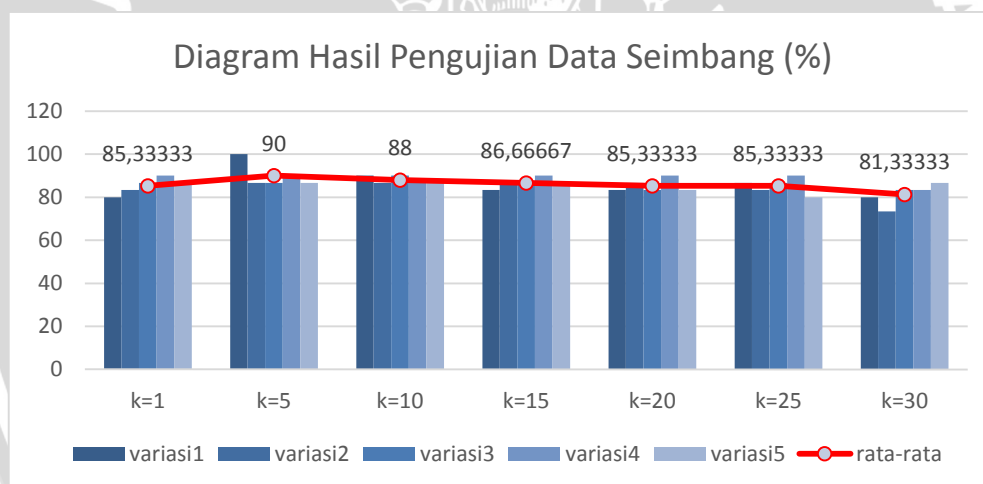
16	46	41	40
17	47	46	41
18	48	47	44
19	51	48	47
20	55	49	53
21	59	51	59
22	62	55	61
23	63	59	63
24	70	63	64
25	72	70	68
26	76	72	69
27	78	76	70
28	79	78	76
29	80	79	78
30	82	80	82
31	83	82	83
32	87	83	84
33	90	92	86
34	92	93	88
35	93	104	91
36	103	106	93
37	104	107	96
38	106	108	104
39	107	109	105
40	108	110	106
41	109	119	109
42	110	120	111
43	117	125	112
44	119	128	115
45	120	129	119
46	125	130	124
47	128	132	127
48	129	134	130
49	130	136	131
50	132	138	132
51	134	139	133
52	136	143	136
53	138	144	137
54	139	148	138
55	143	150	141
56	144	153	144
57	145	154	146
58	153	158	151
59	153	161	153
60	154	164	159
61	158	168	162
62	160	169	163
63	161	173	168
64	168	176	171
65	173	178	173
66	176	182	176
67	178	183	180
68	182	186	181
69	186	187	183
70	187	188	184

6.2.2 Hasil dan Analisis Pengujian Akurasi Sistem

Hasil pengujian akurasi pengaruh nilai k telah dilakukan terhadap masing-masing variasi data latih seimbang dan tidak seimbang dihasilkan nilai akurasi yang berbeda pada k yang sama. Sehingga dihitung nilai akurasi rata-rata dari setiap variasi data latih untuk mewakili nilai akurasi pada k tersebut. Tabel hasil pengujian akurasi sistem dengan data seimbang dapat dilihat pada Tabel 6.15 dan Gambar 6.1 merupakan gambar digram hasil pengujian data seimbang.

Tabel 6. 15 Hasil pengujian 48 data seimbang

No	Data latih	Nilai akurasi sistem (%)						
		k=1	k=5	k=10	k=15	k=20	k=25	k=30
1	variasi1	80	100	90	83,33333	83,33333	86,66667	80
2	variasi2	83,33333	86,66667	86,66667	86,66667	86,66667	83,33333	73,33333
3	variasi3	86,66667	86,66667	90	86,66667	83,33333	86,66667	83,33333
4	variasi4	90	90	86,66667	90	90	90	83,33333
5	variasi5	86,66667	86,66667	86,66667	86,66667	83,33333	80	86,66667
Rata-rata		85,33333	90	88	86,66667	85,33333	85,33333	81,33333



Gambar 6. 1 Diagram hasil pengujian data seimbang

Dari Gambar 6.1 dapat diketahui bahwa nilai rata-rata akurasi pada k=1 adalah 85,33333%. Pada k=5 nilai rata-rata akurasi mengalami kenaikan yaitu menjadi 90%. Namun mulai dari k=10, k=15, k=20, k=25, hingga k=30 nilai rata-rata akurasi terus mengalami penurunan. Rata-rata akurasi tertinggi pada k=5 dengan nilai 90%. Nilai akurasi terendah adalah 81,33333% pada k=30. Nilai akurasi sistem menggunakan data latih seimbang menghasilkan nilai akurasi yang cukup baik yaitu semua bernilai diatas 80%.

Untuk mengetahui hasil pengujian data latih tidak seimbang dapat dilihat pada Tabel 6.16, Tabel 6.17, dan Tabel 6.18. Gambar 6.2 merupakan gambar diagram hasil akurasi pengujian data tidak seimbang.

Tabel 6. 16 Hasil pengujian 30 data tidak seimbang

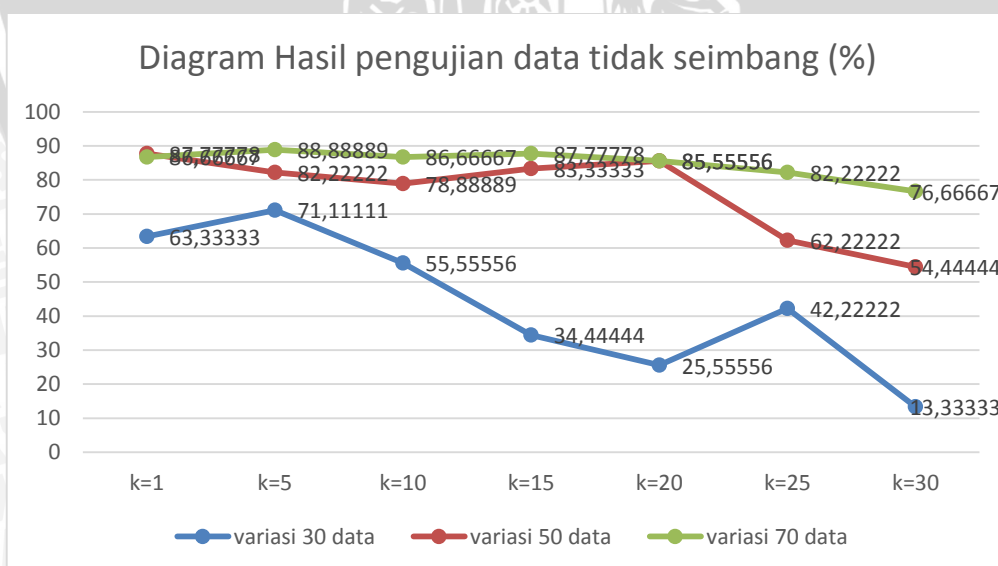
No	Data latih	Nilai akurasi sistem (%)						
		k=1	k=5	k=10	k=15	k=20	k=25	k=30
1	variasi 1	23,33333	70	56,66667	30	26,66667	83,33333	10
2	variasi 2	83,33333	76,66667	56,66667	36,66667	13,33333	13,33333	13,33333
3	variasi 3	83,33333	66,66667	53,33333	36,66667	36,66667	30	16,66667
Rata-rata		63,33333	71,11111	55,55556	34,44444	25,55556	42,22222	13,33333

Tabel 6. 17 Hasil pengujian 50 data tidak seimbang

No	Data latih	Nilai akurasi sistem (%)						
		k=1	k=5	k=10	k=15	k=20	k=25	k=30
1	variasi 1	83,33333	76,66667	76,66667	96,66667	93,33333	46,66667	50
2	variasi 2	90	86,66667	83,33333	83,33333	86,66667	76,66667	63,33333
3	variasi 3	90	83,33333	76,66667	70	76,66667	63,33333	50
Rata-rata		87,77778	82,22222	78,88889	83,33333	85,55556	62,22222	54,44444

Tabel 6. 18 Hasil pengujian 70 data tidak seimbang

No	Data latih	Nilai akurasi sistem (%)						
		k=1	k=5	k=10	k=15	k=20	k=25	k=30
1	variasi 1	86,66667	86,66667	86,66667	90	86,66667	86,66667	80
2	variasi 2	90	90	90	90	93,33333	83,33333	76,66667
3	variasi 3	83,33333	90	83,33333	83,33333	76,66667	76,66667	73,33333
Rata-rata		86,66667	88,88889	86,66667	87,77778	85,55556	82,22222	76,66667



Gambar 6. 2 Diagram hasil pengujian data tidak seimbang

Dari Gambar 6.2 dapat diketahui bahwa ketiga variasi data latih yang tidak seimbang memiliki nilai akurasi yang berbeda. Pada variasi 30 data latih memiliki akurasi tertinggi pada $k=5$ dengan nilai 71,11111% dan nilai akurasi terendah pada $k=30$ dengan nilai 13,33333%. Variasi 50 data latih memiliki nilai akurasi yang relatif lebih tinggi dari pada variasi 30 data. Pada variasi 50 data latih memiliki akurasi tertinggi pada $k=1$ dengan nilai 87,77778% dan nilai akurasi terendah pada $k=30$ dengan nilai 54,44444%. Variasi 70 data latih memiliki nilai akurasi yang relatif tinggi dan perubahan nilai akurasi tidak terlalu signifikan dibandingkan dengan variasi 30 data dan 50 data. Pada variasi 70 data latih memiliki akurasi tertinggi pada $k=5$ dengan nilai 88,88889% dan nilai akurasi terendah pada $k=30$ dengan nilai 76,66667%.

Nilai k yang terlalu kecil menyebabkan klasifikasi terlalu terpengaruh kepada kelas keanggotaan terdekat saja. Sedangkan nilai k yang terlalu tinggi juga akan menyebabkan akurasi rendah karena semakin banyak data yang tidak relevan. Data latih yang digunakan akan mempengaruhi jarak *euclidean* yang dihasilkan. Data latih yang jumlahnya seimbang akan menghasilkan akurasi yang lebih baik dari pada data latih tidak seimbang. Hal ini dikarenakan pada data seimbang sebaran kelas data pada masing-masing penyakit merata sedangkan pada data tidak seimbang terdapat dominasi pada kelas data tertentu.



BAB 7 PENUTUP

Bab ini membahas tentang kesimpulan dan saran dari peneliti berdasarkan hasil perancangan, implementasi dan pengujian system yang telah dilakukan.

7.1 Kesimpulan

Berdasarkan penelitian dari pemodelan sistem pakar untuk identifikasi penyakit pada tanaman tomat menggunakan metode *fuzzy k-nearest neighbor* dapat diambil kesimpulan sebagai berikut:

1. Pemodelan sistem pakar untuk identifikasi penyakit pada tanaman tomat menggunakan metode *fuzzy k-nearest neighbor* telah dirancang menggunakan bahasa pemrograman PHP dan basis data MySQL. Sistem ini mampu mengidentifikasi penyakit tomat beserta penanganannya. Terdapat 6 jenis penyakit tanaman yang dapat diidentifikasi oleh sistem ini yaitu penyakit busuk daun, bercak coklat, layu fusarium, layu bakteri, mozaik, dan keriting.
2. Berdasarkan pengujian akurasi sistem dengan data latih seimbang, diperoleh hasil nilai akurasi tertinggi adalah 90% pada $k=5$ dan nilai akurasi terendah adalah 81,33333% pada $k=30$. Sedangkan dengan data latih tidak seimbang diperoleh nilai akurasi tertinggi adalah 88,88889% pada $k=5$ dan nilai akurasi terendah adalah 13,33333% pada $k=30$. Data latih yang jumlahnya seimbang akan menghasilkan akurasi yang lebih baik dari pada data latih tidak seimbang. Hal ini dikarenakan pada data seimbang sebaran kelas data pada masing-masing penyakit merata sedangkan pada data tidak seimbang terdapat dominasi pada kelas data tertentu.

7.2 Saran

Saran yang dapat diberikan untuk pengembangan sistem agar sistem menjadi lebih baik antara lain:

1. Sistem dapat dikembangkan dengan penambahan fitur berupa pakar dapat menambahkan gejala maupun jenis penyakit. Sehingga tidak hanya terbatas pada 15 gejala dan 6 jenis penyakit tanaman tomat.
2. Sistem dapat menggunakan nilai $k=5$ dan jumlah data latih yang seimbang untuk memperoleh nilai akurasi yang tinggi.

DAFTAR PUSTAKA

- Anggraeni, D., 2015. Sistem Pakar untuk Identifikasi Hama dan Penyakit Tanaman Tebu dengan Metode Fuzzy-AHP. *Skripsi Mahasiswa PTIK Universitas Brawijaya*.
- Anshori, Y., 2012. Pendekatan Triangular Fuzzy Number dalam Metode Analytic Hierarchy Process. *Jurnal Ilmiah Foristek*, 2(1), pp. 126-135.
- BPS, B. P. S., 2014. Produksi Tomat Menurut provinsi, 2009-2013.
- Fadli, A., 2010. Sistem Pakar Dasar.
- Handojo, A. & Irawan, M. I., 2004. Perancangan Dan Pembuatan Aplikasi Sistem Pakar Untuk permasalahan Tindak Pidana Terhadap Harta Kekayaan. *Jurnal Informatika*, Volume 5, pp. 32-38.
- Kailaku, S. I., Dewandari, K. T. & Sunarmani, 2007. Potensi Likopen Dalam Tomat Untuk Kesehatan. *Buletin Teknologi Pascapanen Pertanian*, Volume 3.
- Keller, J. M., Gray, M. R. & Givens, J. A., 1985. A Fuzzy K-Nearest Neighbor Algorithm. *IEEE Transactions On System, Man, and Cybernetics*, Volume SMC-15, No.4.
- Leovini, H., 2012. Pemanfaatan Pupuk Organik Cair Pada Budidaya Tanaman Tomat (*Solanum lycopersicum* L.).
- Meristika, Y. S., 2013. Perbandingan K-Nearest Neighbor dan Fuzzy K-Nearest Neighbor pada Diagnosis Penyakit Diabetes Melitus.
- Putri, Y. C. M., Atastina, I. & Yulita, I. N., 2010. Analisis dan Implementasi Teknik Fuzzy k-Nearest Neighbor pada Kasus Imbalance Class.
- Sagala, A., 2009. respon Pertumbuhan Dan Produksi Tomat (*Solanum lycopersicum* Mill.) Dengan Pemberian Unsur Hara Makro-Mikro Dan Blotong.
- Semangun, H., 1989. *Penyakit-Penyakit Tanaman Hortikultura Di Indonesia*. Yogyakarta: KANISIUS.
- Shofa, R. A., 2014. Penerapan Metode Fuzzy K-Nearest Neighbor (FK-NN) Untuk Menentukan Kualitas Hasil Rendemen Tanaman Tebu.
- Tim, B. k. T., 2013. Pedoman Bertanam Tomat. *Bandung: Yrama Widya*.
- Triyogo, S. & Saputra, B., 2011. Sistem Pakar Untuk Pembelajaran Mengidentifikasi Jenis Bebatuan Berbasis Web.
- Wisdarianto, A., 2013. Penerapan Metode Fuzzy K-Nearest Neighbor (FK-NN) Untuk Pengklasifikasian Spam Email. Volume 1 no. 6.
- Yasa, I. N. D. et al., 2012. Kajian Ketahanan Terhadap Penyakit Busuk Daun (*Phytophthora Infestans*) pada Beberapa Galur Tomat. Volume 1.

Lampiran Pembobotan Gejala

No.	Gejala	Opsi	Nilai
1	Sisi bawah daun terdapat jamur putih	tidak ada	0
		sedikit	10
		banyak	20
2	Pada daun terdapat bercak coklat dengan bagian luar berwarna kuning ke hijau	tidak ada	0
		sedikit	45
		banyak	90
3	Daun berwarna belang antara hijau kekuningan dan berkerut/terpuntir	tidak ada	0
		sedikit	45
		banyak	90
4	Pada daun terdapat bercak coklat gelap, keras, dan melingkar sepusat	tidak ada	0
		sedikit	45
		banyak	90
5	Daun menguning, layu, dan gugur	tidak ada	0
		sedikit	10
		banyak	20
6	Daun mengeriting, dan menggulung ke atas	tidak ada	0
		sedikit	45
		banyak	90
7	Pada batang terdapat bercak coklat lonjong dan konsentris	tidak ada	0
		sedikit	10
		banyak	20
8	Pada batang terdapat bercak kuning atau coklat	tidak ada	0
		sedikit	10
		banyak	20
9	Terdapat cincin coklat pada pembuluh yang dipotong	tidak ada	0
		sedikit	45
		banyak	90
10	Terdapat lendir pada pembuluh batang yang dipotong	tidak ada	0
		sedikit	45
		banyak	90

Lampiran Data Latih

No.	Data	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Identifikasi
1.	1	0	0	0	90	20	0	10	0	0	0	0	0	0	0	0	P2
2.	2	0	45	0	0	0	0	0	10	0	0	0	0	0	0	0	P1
3.	3	0	0	0	0	10	0	0	0	45	0	0	0	20	0	0	P3
4.	4	0	0	0	45	0	0	0	0	0	0	0	10	0	0	0	P2
5.	5	0	0	0	45	0	0	10	0	0	0	0	10	0	0	0	P2
6.	6	0	0	0	90	10	0	10	0	0	0	0	10	0	0	0	P2
7.	7	0	0	0	90	10	0	0	0	0	0	0	0	0	0	0	P2
8.	8	0	45	0	0	0	0	0	0	0	0	10	0	0	0	0	P1
9.	9	0	0	0	0	10	0	0	0	90	0	0	0	0	0	0	P3
10.	10	10	45	0	0	0	0	0	20	0	0	10	0	0	0	0	P1
11.	11	0	0	0	45	20	0	10	0	0	0	0	0	0	0	0	P2
12.	12	0	0	90	0	0	0	0	10	0	0	0	0	0	0	20	P5
13.	13	0	0	0	0	0	90	0	0	0	0	0	0	0	0	20	P6
14.	14	0	0	0	0	10	0	0	0	0	45	0	0	20	0	0	P4
15.	15	0	0	45	0	0	0	0	0	0	0	0	0	20	0	0	P5
16.	16	10	90	0	0	0	0	0	20	0	0	0	0	0	0	0	P1
17.	17	0	0	45	0	0	0	0	20	0	0	0	0	20	0	20	P5
18.	18	0	0	0	45	0	0	20	0	0	0	0	10	0	0	0	P2
19.	19	0	0	0	0	10	0	0	0	0	90	0	0	20	0	0	P4
20.	20	10	45	0	0	0	0	0	10	0	0	20	0	0	0	0	P1
21.	21	0	0	0	0	0	0	0	0	90	0	0	0	20	20	0	P3
22.	22	0	45	0	0	0	0	0	0	0	0	20	0	0	0	0	P1
23.	23	0	0	0	0	10	0	0	0	90	0	0	0	0	20	0	P3
24.	24	0	0	0	0	0	0	0	0	45	0	0	0	0	0	0	P3
25.	25	20	45	0	0	0	0	0	20	0	0	20	0	0	0	0	P1
26.	26	0	0	0	90	0	0	0	0	0	0	0	10	0	0	0	P2
27.	27	0	0	0	45	10	0	10	0	0	0	0	10	0	0	0	P2
28.	28	0	0	45	0	0	0	0	10	0	0	0	0	20	0	0	P5
29.	29	0	0	0	0	0	90	0	0	0	0	0	0	20	0	20	P6
30.	30	0	0	0	90	0	0	10	0	0	0	0	10	0	0	0	P2
31.	31	10	45	0	0	0	0	0	0	0	0	10	0	0	0	0	P1
32.	32	20	45	0	0	0	0	0	10	0	0	20	0	0	0	0	P1
33.	33	0	0	0	0	0	0	0	0	90	0	0	0	0	0	0	P3
34.	34	0	0	0	0	10	0	0	0	45	0	0	0	0	20	0	P3
35.	35	0	0	0	45	20	0	0	0	0	0	0	10	0	0	0	P2
36.	36	0	0	0	45	10	0	20	0	0	0	0	0	0	0	0	P2
37.	37	0	0	90	0	0	0	0	0	0	0	0	0	20	0	20	P5
38.	38	0	0	90	0	0	0	0	20	0	0	0	0	20	0	20	P5
39.	39	0	0	0	0	20	0	0	0	0	45	0	0	20	0	0	P4
40.	40	0	0	0	0	0	45	0	0	0	0	0	0	20	0	0	P6
41.	41	0	0	0	0	0	45	0	0	0	0	0	0	0	0	20	P6
42.	42	0	0	0	45	0	0	20	0	0	0	0	20	0	0	0	P2
43.	43	10	90	0	0	0	0	0	10	0	0	20	0	0	0	0	P1
44.	44	0	0	45	0	0	0	0	10	0	0	0	0	20	0	20	P5
45.	45	20	90	0	0	0	0	0	0	0	0	0	0	0	0	0	P1
46.	46	0	0	0	0	20	0	0	0	0	45	0	0	20	20	0	P4
47.	47	0	0	0	0	20	0	0	0	0	90	0	0	0	0	0	P4
48.	48	0	0	0	0	10	0	0	0	0	45	0	0	0	0	0	P4
49.	49	20	90	0	0	0	0	0	20	0	0	10	0	0	0	0	P1
50.	50	0	45	0	0	0	0	0	10	0	0	20	0	0	0	0	P1

56.	56	20	45	0	0	0	0	0	20	0	0	0	0	0	0	P1
57.	57	10	45	0	0	0	0	0	0	0	0	0	0	0	0	P1
58.	58	0	0	0	90	10	0	0	0	0	0	10	0	0	0	P2
59.	59	0	0	0	0	10	0	0	0	45	0	0	0	0	0	P3
60.	60	0	0	0	0	20	0	0	0	0	45	0	0	0	20	P4
61.	61	0	0	0	45	0	0	20	0	0	0	0	0	0	0	P2
62.	62	20	90	0	0	0	0	0	10	0	0	10	0	0	0	P1
63.	63	10	45	0	0	0	0	0	20	0	0	0	0	0	0	P1
64.	64	0	0	0	0	0	0	0	0	90	0	0	0	0	20	P3
65.	65	0	0	0	0	0	0	0	0	45	0	0	0	0	20	P3
66.	66	10	45	0	0	0	0	0	10	0	0	10	0	0	0	P1
67.	67	0	0	0	90	10	0	20	0	0	0	0	10	0	0	P2
68.	68	0	0	0	0	0	0	0	0	0	45	0	0	0	20	P4
69.	69	0	0	90	0	0	0	0	10	0	0	0	0	0	0	P5
70.	70	0	0	0	0	0	90	0	0	0	0	0	0	0	20	P6
71.	71	0	0	45	0	0	0	0	0	0	0	0	0	0	0	P5
72.	72	0	0	0	45	20	0	0	0	0	0	0	20	0	0	P2
73.	73	0	90	0	0	0	0	0	0	0	0	10	0	0	0	P1
74.	74	0	90	0	0	0	0	0	20	0	0	10	0	0	0	P1
75.	75	10	45	0	0	0	0	0	20	0	0	20	0	0	0	P1
76.	76	0	0	0	90	0	0	10	0	0	0	0	20	0	0	P2
77.	77	0	0	0	0	20	0	0	0	45	0	0	0	0	20	P3
78.	78	0	0	0	0	0	90	0	0	0	0	0	0	0	0	P6
79.	79	0	90	0	0	0	0	0	20	0	0	20	0	0	0	P1
80.	80	0	0	0	0	10	0	0	0	45	0	0	0	20	20	P3
81.	81	0	0	0	45	20	0	10	0	0	0	0	20	0	0	P2
82.	82	20	45	0	0	0	0	0	20	0	0	10	0	0	0	P1
83.	83	0	0	0	0	20	0	0	0	0	90	0	0	20	0	P4
84.	84	0	0	0	90	0	0	20	0	0	0	0	10	0	0	P2
85.	85	0	90	0	0	0	0	0	0	0	0	0	0	0	0	P1
86.	86	0	0	0	0	10	0	0	0	0	45	0	0	0	20	P4
87.	87	0	0	0	90	10	0	20	0	0	0	0	20	0	0	P2
88.	88	0	0	0	90	20	0	0	0	0	0	0	10	0	0	P2
89.	89	0	0	0	0	0	0	0	0	0	45	0	0	0	0	P4
90.	90	10	45	0	0	0	0	0	0	0	0	0	20	0	0	P1
91.	91	0	0	0	45	20	0	0	0	0	0	0	0	0	0	P2
92.	92	0	45	0	0	0	0	0	20	0	0	10	0	0	0	P1
93.	93	0	0	90	0	0	0	0	0	0	0	0	0	0	0	P5
94.	94	0	0	0	90	20	0	20	0	0	0	0	10	0	0	P2
95.	95	0	0	0	45	10	0	20	0	0	0	0	10	0	0	P2
96.	96	0	0	45	0	0	0	0	20	0	0	0	0	20	0	P5
97.	97	0	0	0	45	0	0	0	0	0	0	0	20	0	0	P2
98.	98	20	45	0	0	0	0	0	0	0	0	10	0	0	0	P1
99.	99	10	90	0	0	0	0	0	20	0	0	20	0	0	0	P1
100.	100	0	0	0	90	20	0	10	0	0	0	0	10	0	0	P2
101.	101	0	0	0	90	0	0	0	0	0	0	0	0	0	0	P2
102.	102	0	45	0	0	0	0	0	10	0	0	10	0	0	0	P1
103.	103	10	45	0	0	0	0	0	10	0	0	0	0	0	0	P1
104.	104	0	0	90	0	0	0	0	0	0	0	0	0	0	0	P5
105.	105	0	0	0	90	10	0	10	0	0	0	0	20	0	0	P2

106.	106	0	45	0	0	0	0	0	20	0	0	20	0	0	0	0	P1
107.	107	0	0	45	0	0	0	0	0	0	0	0	0	0	0	20	P5
108.	108	0	0	0	90	0	0	10	0	0	0	0	0	0	0	0	P2
109.	109	0	0	0	0	0	45	0	0	0	0	0	0	20	0	20	P6
110.	110	0	0	0	90	10	0	20	0	0	0	0	0	0	0	0	P2
111.	111	0	90	0	0	0	0	0	10	0	0	10	0	0	0	0	P1
112.	112	0	0	90	0	0	0	0	20	0	0	0	0	0	0	20	P5
113.	113	0	0	0	0	20	0	0	0	0	90	0	0	20	20	0	P4
114.	114	0	90	0	0	0	0	0	20	0	0	0	0	0	0	0	P1
115.	115	0	0	0	90	10	0	10	0	0	0	0	0	0	0	0	P2
116.	116	10	90	0	0	0	0	0	0	0	0	10	0	0	0	0	P1
117.	117	0	0	0	90	20	0	0	0	0	0	0	0	0	0	0	P2
118.	118	0	0	0	0	0	0	0	0	0	45	0	0	20	0	0	P4
119.	119	0	0	0	0	20	0	0	0	90	0	0	0	20	0	0	P3
120.	120	0	0	0	0	10	0	0	0	90	0	0	0	20	0	0	P3
121.	121	0	0	0	90	0	0	0	0	0	0	0	20	0	0	0	P2
122.	122	0	0	0	45	10	0	0	0	0	0	0	20	0	0	0	P2
123.	123	0	0	0	45	0	0	10	0	0	0	0	0	0	0	0	P2
124.	124	0	0	0	0	0	0	0	0	0	90	0	0	0	0	0	P4
125.	125	0	90	0	0	0	0	0	10	0	0	20	0	0	0	0	P1
126.	126	0	0	90	0	0	0	0	0	0	0	0	0	20	0	0	P5
127.	127	20	90	0	0	0	0	0	10	0	0	0	0	0	0	0	P1
128.	128	0	45	0	0	0	0	0	20	0	0	0	0	0	0	0	P1
129.	129	0	0	0	45	20	0	20	0	0	0	0	0	0	0	0	P2
130.	130	0	0	90	0	0	0	0	10	0	0	0	0	20	0	20	P5
131.	131	0	0	0	90	20	0	20	0	0	0	0	20	0	0	0	P2
132.	132	0	0	0	0	10	0	0	0	0	90	0	0	0	0	0	P4
133.	133	0	0	90	0	0	0	0	10	0	0	0	0	20	0	0	P5
134.	134	10	90	0	0	0	0	0	10	0	0	10	0	0	0	0	P1
135.	135	0	0	0	0	20	0	0	0	45	0	0	0	20	0	0	P3
136.	136	0	0	0	45	10	0	10	0	0	0	0	20	0	0	0	P2
137.	137	0	0	0	0	0	0	0	0	0	90	0	0	20	0	0	P4
138.	138	0	0	0	0	0	0	0	0	0	90	0	0	0	20	0	P4
139.	139	20	45	0	0	0	0	0	10	0	0	10	0	0	0	0	P1
140.	140	0	0	0	0	0	0	0	0	0	90	0	0	20	20	0	P4
141.	141	0	0	45	0	0	0	0	10	0	0	0	0	0	0	0	P5
142.	142	20	90	0	0	0	0	0	20	0	0	20	0	0	0	0	P1
143.	143	0	0	90	0	0	0	0	20	0	0	0	0	0	0	0	P5
144.	144	0	0	0	0	10	0	0	0	90	0	0	0	20	20	0	P3
145.	145	20	90	0	0	0	0	0	0	0	0	10	0	0	0	0	P1
146.	146	10	90	0	0	0	0	0	20	0	0	10	0	0	0	0	P1
147.	147	0	0	45	0	0	0	0	0	0	0	0	0	20	0	20	P5
148.	148	0	0	0	0	20	0	0	0	90	0	0	0	0	20	0	P3
149.	149	0	0	0	45	10	0	20	0	0	0	0	20	0	0	0	P2
150.	150	0	0	0	0	10	0	0	0	0	90	0	0	20	20	0	P4
151.	151	0	0	0	90	0	0	20	0	0	0	0	0	0	0	0	P2
152.	152	0	0	0	45	10	0	0	0	0	0	0	0	0	0	0	P2
153.	153	0	0	0	0	20	0	0	0	45	0	0	0	20	20	0	P3
154.	154	0	0	0	45	10	0	0	0	0	0	0	10	0	0	0	P2
155.	155	0	0	0	90	20	0	10	0	0	0	0	20	0	0	0	P2
156.	156	10	90	0	0	0	0	0	0	0	0	0	0	0	0	0	P1
157.	157	0	0	0	90	0	0	20	0	0	0	0	20	0	0	0	P2
158.	158	0	0	0	45	10	0	10	0	0	0	0	0	0	0	0	P2
159.	159	0	0	0	0	0	0	0	0	45	0	0	0	20	0	0	P3
160.	160	0	0	0	0	0	45	0	0	0	0	0	0	0	0	0	P6

161.	161	20	45	0	0	0	0	0	0	0	0	0	0	0	0	0	P1
162.	162	0	0	0	0	20	0	0	0	90	0	0	0	20	20	0	P3
163.	163	0	0	0	45	20	0	20	0	0	0	0	20	0	0	0	P2
164.	164	0	0	0	0	20	0	0	0	45	0	0	0	0	0	0	P3
165.	165	0	0	45	0	0	0	0	20	0	0	0	0	0	0	20	P5
166.	166	20	45	0	0	0	0	0	10	0	0	0	0	0	0	0	P1
167.	167	20	90	0	0	0	0	0	0	0	0	20	0	0	0	0	P1
168.	168	0	0	0	0	20	0	0	0	0	90	0	0	0	20	0	P4
169.	169	0	0	0	0	0	0	0	0	45	0	0	0	20	20	0	P3
170.	170	0	0	0	45	0	0	0	0	0	0	0	0	0	0	0	P2
171.	171	0	0	0	0	10	0	0	0	0	90	0	0	0	20	0	P4
172.	172	0	0	0	45	0	0	10	0	0	0	0	20	0	0	0	P2
173.	173	0	0	45	0	0	0	0	10	0	0	0	0	0	0	20	P5
174.	174	0	0	0	45	20	0	10	0	0	0	0	10	0	0	0	P2
175.	175	20	45	0	0	0	0	0	0	0	0	20	0	0	0	0	P1
176.	176	0	0	0	0	10	0	0	0	0	45	0	0	20	20	0	P4
177.	177	0	0	0	0	20	0	0	0	0	45	0	0	0	0	0	P4
178.	178	0	0	45	0	0	0	0	20	0	0	0	0	0	0	0	P5
179.	179	0	90	0	0	0	0	0	0	0	0	20	0	0	0	0	P1
180.	180	10	90	0	0	0	0	0	0	0	0	20	0	0	0	0	P1
181.	181	10	90	0	0	0	0	0	10	0	0	0	0	0	0	0	P1
182.	182	0	0	0	0	0	0	0	0	0	45	0	0	20	20	0	P4
183.	183	0	0	90	0	0	0	0	20	0	0	0	0	20	0	0	P5
184.	184	0	0	0	90	20	0	20	0	0	0	0	0	0	0	0	P2
185.	185	0	90	0	0	0	0	0	10	0	0	0	0	0	0	0	P1
186.	186	0	0	0	90	10	0	0	0	0	0	0	20	0	0	0	P2
187.	187	0	45	0	0	0	0	0	0	0	0	0	0	0	0	0	P1
188.	188	20	90	0	0	0	0	0	20	0	0	0	0	0	0	0	P1

Keterangan:

P1 = Penyakit Busuk Daun
 P2 = Penyakit Bercak Coklat
 P3 = Penyakit Layu Fusarium
 P4 = Penyakit Layu Bakteri
 P5 = Penyakit Mozaik
 P6 = Penyakit Keriting

Lampiran Pengujian

Pengujian 48 data seimbang variasi 1

G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Pakar	k=1	k=5	k=10	k=15	k=20	k=25	k=30
	B			B					B				P2	P2	P2	P2	P2	P2	P2	P2
B					B					B			P5	P5	P5	P5	P5	P5	P5	P5
		B				B				B			P3	P3	P3	P3	P3	P3	P3	P3
		B							B		B		P3/p4	P2	P3	P2	P2	P2	P2	P2
							B			B	B		P4	P4	P4	P4	P4	P4	P4	P3
					B			B					P1	P1	P1	P1	P1	P1	P1	P1
			B								B	B	P6	P6	P6	P6	P6	P6	P6	P6
								B					P1	P1	P1	P1	P1	P1	P1	P1
		B				B					B		P3	P3	P3	P3	P3	P3	P3	P3
		B					B						P4	P4	P4	P4	P4	P4	P4	P4
B												B	P5	P5	P5	P5	P5	P5	P5	P5
			B									B	P6	P6	P6	P6	P6	P6	P6	P6
	B	B			B					B			P2	P4	P2	P3	P3	P3	P3	P3
					B								P1	P1	P1	P1	P1	P1	P1	P1
						B					B		P3	P3	P3	P3	P3	P3	P3	P3
		B				B							P3	P3	P3	P3	P3	P3	P3	P3
B										B			P5	P5	P5	P5	P5	P5	P5	P5
		B								B	B		P3/p4	P3	P3	P3	P3	P3	P3	P3
			B		B					B			P6	P6	P6	P6	P6	P6	P6	P6
							B			B			P4	P4	P4	P4	P4	P4	P4	P4
	B			B					B		B		P2	P2	P2	P2	P2	P2	P2	P2
										B		B	P1	P6	P1	P6	P1	P6	P1	P1
B										B	B		P3	P4	P3	P3	P4	P3	P3	P5
	B										B		P3	P2	P3	P3	P2	P2	P2	P4
			B							B			P6	P6	P6	P6	P6	P6	P6	P6
B										B		B	P5	P5	P5	P5	P5	P5	P5	P5
					B			B					P1	P1	P1	P1	P1	P1	P1	P1
	B									B			P4	P2	P4	P4	P2	P3	P6	P2
		B					B			B			P4	P4	P4	P4	P4	P4	P4	P4
						B				B	B		P3	P3	P3	P3	P3	P3	P3	P3
salah														6	0	3	5	5	4	6
benar														24	30	27	25	25	26	24
akurasi														80	100	90	83,3	83,3	86,7	80

Pengujian 48 data seimbang variasi 2

no	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Pakar	k=1	k=5	k=10	k=15	k=20	k=25	k=30
1				B			B					B				P2	P2	P2	P2	P2	P2	P2	P2
2			B					B					B			P5	P4	P5	P5	P5	P5	P5	P5
3					B				B				B			P3	P3	P3	P3	P3	P3	P3	P3
4					B							B		B		P3/p4	P3	P3	P4	P2	P2	P2	P2
5										B			B	B		P4	P4	P4	P4	P4	P4	P4	P3
6		B						B			B					P1	P1	P1	P1	P1	P1	P1	P1
7						B								B	B	P6	P6	P6	P6	P6	P6	P6	P6
8	B	B									B					P1	P1	P1	P1	P1	P1	P1	P1
9					B				B					B		P3	P3	P3	P3	P3	P3	P3	P2
10					B					B						P4	P4	P4	P4	P4	P4	P4	P2
11			B												B	P5	P5	P5	P5	P5	P5	P5	P5
12						B									B	P6	P6	P6	P6	P6	P6	P6	P6
13				B	B			B					B			P2	P2	P2	P2	P2	P2	P2	P2
14		B						B								P1	P1	P1	P1	P1	P1	P1	P1
15									B					B		P3	P3	P3	P3	P3	P3	P3	P3
16					B				B							P3	P3	P3	P3	P3	P3	P3	P3
17			B										B			P5	P5	P5	P5	P5	P5	P5	P5
18					B								B	B		P3/p4	P3	P3	P3	P4	P4	P4	P4
19						B		B					B			P6	P6	P5	P6	P6	P6	P6	P5
20										B			B			P4	P4	P4	P4	P4	P4	P4	P4
21				B			B					B		B		P2	P2	P2	P2	P2	P2	P2	P2
22	B	B											B		B	P1	P5	P1	P6	P1	P1	P1	P1
23			B										B	B		P3	P3	P5	P5	P5	P5	P5	P5
24				B										B		P3	P4	P2	P2	P2	P2	P2	P2
25						B							B			P6	P6	P6	P6	P6	P6	P6	P6
26			B										B		B	P5	P5	P5	P5	P5	P5	P6	P5
27	B							B			B					P1	P6	P1	P1	P1	P1	P1	P1
28	B			B									B			P4	P6	P2	P2	P2	P2	P2	P2
29					B					B			B			P4	P4	P4	P4	P4	P4	P4	P4
30		B							B				B	B		P3	P3	P3	P3	P3	P3	P3	P3
																salah	5	4	4	4	4	5	8
																benar	25	26	26	26	26	25	22
																akurasi	83,3	86,7	86,7	86,7	86,7	83,3	73,3

Pengujian 48 data seimbang variasi 3

no	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Pakar	k=1	k=5	k=10	k=15	k=20	k=25	k=30
1				B			B					B				P2	P2	P2	P2	P2	P2	P2	P2
2			B					B					B			P5	P5	P5	P5	P5	P5	P5	P5
3					B				B				B			P3	P3	P3	P3	P3	P3	P3	P3
4				B								B		B		P3/p4	P4	P4	P4	P3	P2	P2	P2
5										B			B	B		P4	P4	P4	P4	P4	P4	P4	P4
6		B						B			B					P1	P1	P1	P1	P1	P1	P1	P1
7						B								B	B	P6	P6	P6	P6	P6	P6	P6	P6
8	B	B									B					P1	P1	P1	P1	P1	P1	P1	P1
9					B				B					B		P3	P3	P3	P3	P3	P3	P3	P3
10				B						B						P4	P4	P4	P4	P4	P4	P4	P4
11			B												B	P5	P5	P4	P5	P5	P5	P5	P5
12						B									B	P6	P6	P6	P6	P6	P6	P6	P6
13				B	B			B					B			P2	P2	P2	P2	P2	P2	P2	P2
14		B						B								P1	P1	P1	P1	P1	P1	P1	P1
15									B					B		P3	P3	P3	P3	P3	P3	P3	P3
16				B				B								P3	P3	P3	P3	P6	p3	P3	P3
17			B										B			P5	P5	P5	P5	P5	p6	P5	P5
18				B									B	B		P3/p4	P4	P4	P4	P3	p3	P3	P3
19						B		B					B			P6	P6	P6	P6	P6	p6	P6	P5
20										B			B			P4	P4	P4	P4	P4	p4	P4	P4
21				B			B					B		B		P2	P2	P2	P2	P2	p2	P2	P2
22	B	B											B		B	P1	P6	P1	P1	P1	p1	P1	P1
23			B										B	B		P3	P5	P5	P5	P5	p4	P4	P4
24				B										B		P3	P2	P2	P4	P2	p4	P3	P2
25						B							B			P6	P6	P6	P6	P6	p6	P6	P6
26			B										B		B	P5	P5	P5	P5	P5	p5	P6	P5
27	B							B			B					P1	P1	P1	P1	P1	p1	P1	P1
28	B			B									B			P4	P2	P2	P5	P6	p1	P1	P1
29					B					B			B			P4	P4	P4	P4	P4	p4	P4	P4
30		B							B				B	B		P3	P3	P3	P3	P3	p3	P3	P3
salah																	4	4	3	4	5	4	5
benar																	26	26	27	26	25	26	25
akurasi																	86,7	86,7	90	86,7	83,3	86,7	83,3

Pengujian 48 data seimbang variasi 4

no	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Pakar	k=1	k=5	k=10	k=15	k=20	k=25	k=30
1				B			B					B				P2	P2	P2	P2	P2	P2	P2	P2
2			B					B					B			P5	P5	P5	P5	P5	P5	P5	P5
3					B				B				B			P3	P3	P3	P3	P3	P3	P3	P3
4					B							B		B		P3/p4	P2	P2	P2	P3	P3	P2	P2
5										B			B	B		P4	P4	P4	P4	P4	P4	P4	P4
6		B						B			B					P1	P1	P1	P1	P1	P1	P1	P1
7						B								B	B	P6	P6	P6	P6	P6	P6	P6	P6
8	B	B									B					P1	P1	P1	P1	P1	P1	P1	P1
9					B				B					B		P3	P3	P3	P3	P3	P3	P3	P3
10					B					B						P4	P4	P4	P4	P4	P4	P4	P4
11			B												B	P5	P5	P5	P5	P5	P5	P5	P4
12						B									B	P6	P6	P6	P6	P6	P6	P6	P6
13				B	B			B					B			P2	P2	P2	P5	P2	P2	P2	P2
14		B						B								P1	P1	P1	P1	P1	P1	P1	P1
15									B					B		P3	P3	P3	P3	P3	P3	P3	P3
16					B				B							P3	P3	P3	P3	P3	P3	P3	P3
17			B										B			P5	P5	P5	P5	P5	P5	P5	P5
18					B								B	B		P3/p4	P3	P3	P4	P4	P3	P3	P4
19						B		B					B			P6	P6	P6	P6	P6	P6	P6	P6
20										B			B			P4	P4	P4	P4	P4	P4	P4	P4
21				B			B					B		B		P2	P2	P2	P2	P2	P2	P2	P2
22	B	B											B		B	P1	P6	P1	P1	P1	P1	P1	P1
23			B										B	B		P3	P3	P3	P3	P5	P5	P5	P5
24				B										B		P3	P3	P2	P2	P2	P2	P2	P2
25						B							B			P6	P6	P6	P6	P6	P6	P6	P6
26			B										B		B	P5	P5	P5	P5	P5	P5	P5	P5
27	B							B			B					P1	P1	P1	P1	P1	P1	P1	P1
28	B			B									B			P4	P6	P1	P1	P2	P2	P4	P2
29					B					B			B			P4	P4	P4	P4	P4	P4	P4	P4
30		B							B				B	B		P3	P3	P3	P3	P3	P3	P3	P3
																salah	3	3	4	3	3	3	5
																benar	27	27	26	27	27	27	25
																akurasi	90	90	86,7	90	90	90	83,3

Pengujian 48 data seimbang variasi 5

no	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Pakar	k=1	k=5	k=10	k=15	k=20	k=25	k=30
1				B			B					B				P2	P2	P2	P2	P2	P2	P2	P2
2			B					B					B			P5	P5	P5	P5	P5	P5	P5	P5
3					B				B				B			P3	P3	P3	P3	P3	P3	P3	P3
4					B							B		B		P3/p4	P2	P2	P4	P4	P2	P4	P4
5										B			B	B		P4	P4	P4	P4	P4	P4	P4	P4
6		B						B			B					P1	P1	P1	P1	P1	P1	P1	P1
7						B								B	B	P6	P6	P6	P6	P6	P6	P6	P6
8	B	B									B					P1	P1	P1	P1	P1	P1	P1	P1
9					B				B					B		P3	P3	P3	P3	P3	P3	P4	P3
10					B					B						P4	P4	P4	P4	P4	P4	p4	P4
11			B												B	P5	P5	P5	P5	P5	P5	P5	P5
12						B									B	P6	P6	P6	P6	P6	P6	P6	P6
13				B	B			B					B			P2	P2	P2	P2	P2	P2	P2	P2
14		B						B								P1	P1	P1	P1	P1	P1	P1	P1
15									B					B		P3	P3	P3	P3	P3	P3	P3	P3
16					B				B							P3	P3	P3	P3	P3	P3	P3	P3
17			B										B			P5	P5	P5	P5	P5	P5	P5	P5
18					B								B	B		P3/p4	P4	P4	P4	P4	P4	P4	P3
19						B		B					B			P6	P6	P6	P6	P6	P6	P6	P6
20										B			B			P4	P4	P4	P4	P4	P4	P4	P4
21				B			B					B		B		P2	P2	P2	P2	P2	P2	P2	P2
22	B	B											B		B	P1	P1	P1	P6	P6	P6	P6	P6
23			B										B	B		P3	P5	P5	P5	P5	P5	P5	P5
24				B										B		P3	P2	P2	P2	P2	P2	P2	P2
25						B							B			P6	P6	P6	P6	P6	P6	P6	P6
26			B										B		B	P5	P5	P5	P5	P5	P5	P5	P5
27	B							B			B					P1	P1	P1	P1	P1	P1	P6	P1
28	B			B									B			P4	P6	P2	P2	P6	P2	P2	P2
29					B					B			B			P4	P4	P4	P4	P4	P4	P4	P3
30		B							B				B	B		P3	P3	P3	P3	P3	P3	P3	P3
																salah	4	4	4	4	5	6	4
																benar	26	26	26	26	25	24	26
																akurasi	86,7	86,7	86,7	86,7	83,3	80	86,7

Pengujian 30 data tidak seimbang variasi 1

no	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Pakar	k=1	k=5	k=10	k=15	k=20	k=25	k=30
1				B			B					B				P2	p2	p2	p2	p2	p2	p2	p2
2			B					B					B			P5	p4	p5	p4	p4	p4	p5	p2
3					B				B				B			P3	p4	p3	p3	p4	p4	p3	p2
4					B							B		B		P3/p4	p2	p3	p4	p2	p2	p3	p2
5										B			B	B		P4	p4	p4	p4	p4	p4	p4	p2
6		B						B			B					P1	p2	p1	p1	p4	p2	p1	p2
7						B								B	B	P6	p4	p4	p4	p4	p4	p6	p2
8	B	B									B					P1	p2	p1	p3	p4	p2	p1	p2
9					B				B					B		P3	p2	p3	p3	p4	p4	p3	p2
10					B					B						P4	p2	p4	p4	p4	p4	p4	p2
11			B												B	P5	p2	p3	p3	p4	p2	p5	p2
12						B									B	P6	p2	p3	p3	p4	p2	p6	p2
13				B	B			B					B			P2	p2	p2	p2	p2	p2	p4	p2
14		B						B								P1	p2	p1	p1	p4	p2	p1	p2
15									B					B		P3	p2	p3	p3	p3	p4	p3	p2
16					B				B							P3	p2	p3	p3	p4	p4	p3	p2
17			B										B			P5	p2	p5	p4	p4	p4	p5	p2
18					B								B	B		P3/p4	p4	p4	p4	p4	p4	p3	p2
19						B		B					B			P6	p4	p5	p4	p4	p4	p6	p2
20										B			B			P4	p4	p4	p4	p4	p4	p4	p2
21				B			B					B		B		P2	p2	p2	p2	p2	p2	p2	p2
22	B	B											B		B	P1	p4	p5	p4	p4	p4	p5	p2
23			B										B	B		P3	p4	p5	p4	p4	p4	p4	p2
24				B										B		P3	p2	p2	p2	p2	p2	p4	p2
25						B							B			P6	p2	p4	p4	p4	p4	p6	p2
26			B										B		B	P5	p2	p5	p4	p4	p4	p5	p2
27	B							B			B					P1	p2	p1	p1	p4	p2	p1	p2
28	B			B									B			P4	p2	p2	p2	p2	p2	p2	p2
29					B					B			B			P4	p4	p4	p4	p4	p4	p4	p2
30		B							B				B	B		P3	p4	p3	p3	p4	p4	p3	p2
																salah	23	9	13	21	22	5	27
																benar	7	21	17	9	8	25	3
																akurasi	23,33	70	56,67	30	26,67	83,33	10

Pengujian 30 data tidak seimbang variasi 2

no	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Pakar	k=1	k=5	k=10	k=15	k=20	k=25	k=30
1				B			B					B				P2	p2	p1	p6	p6	p5	p5	p5
2			B					B					B			P5	p5	p5	p5	p5	p5	p5	p5
3					B				B				B			P3	p3	p3	p6	p5	p5	p5	p5
4					B							B		B		P3/p4	p4	p4	p4	p4	p5	p5	p5
5										B			B	B		P4	p4	p4	p4	p4	p5	p5	p5
6		B						B			B					P1	p1	p1	p5	p5	p5	p5	p5
7						B								B	B	P6	p6	p6	p6	p6	p5	p5	p5
8	B	B									B					P1	p1	p1	p1	p6	p5	p5	p5
9					B				B					B		P3	p3	p3	p4	p4	p6	p5	p5
10					B					B						P4	p4	p4	p4	p4	p5	p5	p5
11			B												B	P5	p5	p5	p5	p5	p5	p5	p5
12						B										P6	p6	p6	p6	p5	p5	p5	p5
13				B	B			B					B			P2	p4	p4	p5	p5	p5	p5	p5
14		B						B								P1	p1	p1	p1	p5	p5	p5	p5
15									B					B		P3	p3	p3	p3	p5	p5	p5	p5
16					B				B							P3	p3	p3	p4	p5	p4	p5	p5
17			B										B			P5	p5	p5	p5	p5	p5	p5	p5
18					B								B	B		P3/p4	p3	p3	p4	p4	p5	p5	p5
19						B		B					B			P6	p6	p6	p5	p5	p5	p5	p5
20										B			B			P4	p4	p4	p4	p5	p5	p5	p5
21				B			B					B		B		P2	p2	p4	p4	p4	p6	p5	p5
22	B	B											B		B	P1	p6	p5	p6	p6	p5	p5	p5
23			B										B	B		P3	p5	p5	p5	p5	p5	p5	p5
24				B										B		P3	p4	p4	p4	p4	p6	p5	p5
25						B							B			P6	p6	p6	p6	p6	p5	p5	p5
26			B										B		B	P5	p5	p5	p5	p5	p5	p5	p5
27	B							B			B					P1	p1	p1	p5	p5	p5	p5	p5
28	B			B									B			P4	p6	p6	p5	p5	p5	p5	p5
29					B					B			B			P4	p4	p4	p4	p4	p5	p5	p5
30		B							B				B	B		P3	p3	p3	p3	p5	p5	p5	p5
																salah	5	7	13	19	26	26	26
																benar	25	23	17	11	4	4	4
																akurat	83,33	76,67	56,67	36,67	13,33	13,33	13,33

Pengujian 30 data tidak seimbang variasi 3

no	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Pakar	k=1	k=5	k=10	k=15	k=20	k=25	k=30
1				B			B					B				P2	p2	p2	p2	p2	p6	p6	p1
2			B					B					B			P5	p5	p5	p6	p1	p6	p6	p1
3					B				B				B			P3	p3	p3	p6	p2	p6	p6	p1
4					B							B		B		P3/p4	p3	p2	p2	p2	p2	p6	p1
5										B			B	B		P4	p4	p4	p6	p6	p6	p6	p1
6		B						B			B					P1	p1	p1	p1	p1	p1	p1	p1
7						B								B	B	P6	p6	p6	p6	p6	p6	p6	p1
8	B	B									B					P1	p1	p1	p1	p1	p1	p1	p1
9					B				B					B		P3	p3	p3	p2	p2	p6	p2	p1
10					B					B						P4	p2	p2	p2	p2	p6	p6	p1
11			B												B	P5	p5	p6	p6	p6	p6	p6	p1
12						B									B	P6	p6	p6	p6	p6	p6	p6	p1
13				B	B			B					B			P2	p2	p2	p2	p2	p2	p2	p1
14		B						B								P1	p1	p1	p1	p1	p1	p1	p1
15									B					B		P3	p3	p3	p3	p2	p6	p6	p1
16					B				B							P3	p3	p2	p2	p2	p6	p6	p1
17			B										B			P5	p5	p6	p6	p6	p6	p6	p1
18					B								B	B		P3/p4	p3	p3	p3	p6	p6	p6	p1
19						B		B					B			P6	p6	p6	p6	p6	p6	p6	p1
20										B			B			P4	p4	p4	p6	p6	p6	p6	p1
21				B			B					B		B		P2	p2	p2	p2	p2	p2	p6	p1
22	B	B											B		B	P1	p6	p1	p1	p6	p1	p6	p1
23			B										B	B		P3	p4	p4	p6	p6	p6	p6	p1
24				B										B		P3	p2	p2	p2	p2	p2	p2	p1
25						B							B			P6	p6	p6	p6	p6	p6	p6	p1
26			B										B		B	P5	p5	p6	p6	p6	p6	p6	p1
27	B							B			B					P1	p1	p1	p1	p1	p1	p1	p1
28	B			B									B			P4	p2	p2	p2	p2	p6	p6	p1
29					B					B			B			P4	p4	p4	p4	p6	p6	p6	p1
30		B							B				B	B		P3	p3	p4	p3	p1	p1	p1	p1
																salah	5	10	14	19	19	21	25
																benar	25	20	16	11	11	9	5
																akurasi	83,33	66,67	53,33	36,67	36,67	30	16,67

Pengujian 50 data tidak seimbang variasi 1

no	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Pakar	k=1	k=5	k=10	k=15	k=20	k=25	k=30
1				B			B					B				P2	p2	p2	p2	p2	p2	p2	p2
2			B					B					B			P5	p5	p5	p5	p5	p5	p5	p5
3					B				B				B			P3	p3	p3	p3	p3	p3	p4	p4
4					B							B		B		P3/p4	p3	p2	p4	p2	p4	p4	p4
5										B			B	B		P4	p4	p4	p4	p4	p4	p4	p4
6		B						B			B					P1	p1	p1	p1	p5	p5	p5	p5
7						B								B	B	P6	p6	p4	p4	p4	p4	p4	p4
8	B	B									B					P1	p1	p1	p1	p1	p1	p1	p4
9					B				B					B		P3	p3	p3	p3	p3	p4	p4	p4
10					B					B						P4	p4	p4	p4	p4	p4	p4	p4
11			B												B	P5	p5	p5	p5	p5	p5	p5	p5
12						B									B	P6	p6	p5	p5	p5	p5	p5	p6
13				B	B			B					B			P2	p4	p2	p2	p2	p2	p2	p2
14		B						B								P1	p1	p1	p1	p5	p5	p5	p5
15									B					B		P3	p3	p3	p3	p3	p3	p4	p4
16					B				B							P3	p3	p3	p3	p3	p3	p4	p4
17			B										B			P5	p5	p5	p5	p5	p5	p5	p5
18					B								B	B		P3/p4	p3	p4	p4	p4	p4	p4	p4
19						B		B					B			P6	p6	p5	p5	p5	p5	p5	p5
20										B			B			P4	p4	p4	p4	p4	p4	p4	p4
21				B			B					B		B		P2	p2	p2	p2	p2	p2	p2	p2
22	B	B											B		B	P1	p3	p1	p1	p5	p5	p5	p5
23			B										B	B		P3	p5	p5	p5	p5	p4	p5	p4
24				B										B		P3	p4	p4	p2	p2	p2	p2	p2
25						B							B			P6	p6	p4	p4	p4	p4	p4	p4
26			B										B		B	P5	p5	p5	p5	p5	p5	p5	p5
27	B							B			B					P1	p3	p1	p1	p1	p5	p5	p5
28	B			B									B			P4	p4	p4	p2	p2	p2	p2	p2
29					B					B			B			P4	p4	p4	p4	p4	p4	p4	p4
30		B							B				B	B		P3	p3	p3	p3	p3	p3	p4	p4
salah																	5	7	7	11	12	16	15
benar																	25	23	23	29	28	14	15
akuras																	83,33	76,67	76,67	96,67	93,33	46,67	50

Pengujian 50 data tidak seimbang variasi 2

no	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Pakar	k=1	k=5	k=10	k=15	k=20	k=25	k=30
1				B			B					B				P2	p2	p2	p2	p2	p2	p2	p1
2			B					B					B			P5	p5	p5	p5	p5	p5	p5	p5
3					B				B				B			P3	p3	p3	p3	p3	p3	p3	p5
4					B							B		B		P3/p4	p3	p2	p2	p2	p2	p4	p4
5										B			B	B		P4	p4	p4	p4	p4	p4	p4	p4
6		B						B			B					P1	p1	p1	p1	p1	p1	p1	p1
7						B								B	B	P6	p6	p6	p6	p6	p6	p5	p5
8	B	B									B					P1	p1	p1	p1	p1	p1	p1	p1
9					B				B					B		P3	p3	p3	p3	p3	p3	p4	p4
10					B					B						P4	p4	p4	p4	p4	p4	p4	p4
11			B												B	P5	p5	p5	p5	p5	p5	p5	p5
12						B									B	P6	p6	p6	p6	p5	p6	p5	p5
13				B	B			B					B			P2	p2	p2	p2	p5	p5	p5	p1
14		B						B								P1	p1	p1	p1	p1	p1	p1	p1
15									B					B		P3	p3	p3	p3	p3	p3	p3	p4
16					B				B							P3	p3	p3	p3	p3	p3	p3	p3
17			B										B			P5	p5	p5	p5	p5	p5	p5	p5
18					B								B	B		P3/p4	p3	p3	p3	p3	p4	p4	p4
19						B		B					B			P6	p6	p5	p5	p5	p5	p5	p5
20										B			B			P4	p4	p4	p4	p4	p4	p4	p4
21				B			B					B		B		P2	p2	p2	p2	p2	p2	p2	p2
22	B	B											B		B	P1	p6	p1	p6	p1	p1	p1	p1
23			B										B	B		P3	p5	p5	p5	p5	p5	p5	p5
24				B										B		P3	p2	p2	p2	p2	p4	p4	p2
25						B							B			P6	p6	p6	p6	p6	p6	p6	p5
26			B										B		B	P5	p5	p5	p5	p5	p5	p5	p5
27	B							B			B					P1	p1	p1	p1	p1	p1	p1	p1
28	B			B									B			P4	p2	p2	p2	p2	p2	p2	p5
29					B					B			B			P4	p4	p4	p4	p4	p4	p4	p4
30		B							B				B	B		P3	p3	p3	p3	p3	p3	p3	p1
																salah	3	4	5	5	4	7	11
																benar	27	26	25	25	26	23	19
																akuras	90	86,67	83,33	83,33	86,67	76,67	63,33

Pengujian 50 data tidak seimbang variasi 3

no	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Pakar	k=1	k=5	k=10	k=15	k=20	k=25	k=30
1				B			B					B				P2	p2	p2	p2	p2	p2	p2	p2
2			B					B					B			P5	p5	p5	p6	p1	p1	p6	p1
3					B				B				B			P3	p3	p3	p3	p3	p3	p2	p2
4					B							B		B		P3/p4	p3	p2	p2	p2	p2	p2	p2
5									B				B	B		P4	p4	p4	p4	p4	p4	p4	p3
6		B						B			B					P1	p1	p1	p1	p1	p1	p1	p1
7						B								B	B	P6	p6	p6	p6	p6	p6	p6	p6
8	B	B									B					P1	p1	p1	p1	p1	p1	p1	p1
9					B				B					B		P3	p3	p3	p3	p3	p3	p2	p2
10					B					B						P4	p4	p4	p2	p2	p2	p2	p2
11			B												B	P5	p5	p6	p6	p6	p6	p2	p2
12						B									B	P6	p6	p6	p6	p6	p6	p6	p2
13				B	B			B					B			P2	p2	p2	p2	p2	p2	p2	p2
14		B						B								P1	p1	p1	p1	p1	p1	p1	p1
15									B					B		P3	p3	p3	p3	p3	p3	p3	p3
16					B				B							P3	p3	p3	p3	p2	p3	p2	p2
17			B										B			P5	p5	p5	p6	p6	p6	p6	p6
18					B								B	B		P3/p4	p3	p3	p3	p3	p3	p3	p2
19						B		B					B			P6	p6	p6	p6	p6	p6	p6	p6
20										B			B			P4	p4	p4	p4	p4	p4	p4	p6
21				B			B					B		B		P2	p2	p2	p2	p2	p2	p2	p2
22	B	B											B		B	P1	p6	p5	p1	p1	p1	p1	p1
23			B										B	B		P3	p4	p3	p3	p3	p3	p3	p3
24				B										B		P3	p2	p2	p2	p2	p2	p2	p2
25						B							B			P6	p6	p6	p6	p6	p6	p6	p6
26			B										B		B	P5	p5	p6	p6	p6	p6	p6	p6
27	B							B			B					P1	p1	p1	p1	p1	p1	p1	p1
28	B			B									B			P4	p2	p2	p2	p2	p2	p2	p2
29					B					B			B			P4	p4	p4	p4	p4	p4	p2	p2
30		B							B				B	B		P3	p3	p3	p3	p6	p3	p3	p3
salah																	3	5	7	9	7	11	15
benar																	27	25	23	21	23	19	15
akurasi																	90	83,33	76,67	70	76,67	63,33	50

Pengujian 70 data tidak seimbang variasi 1

no	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Pakar	k=1	k=5	k=10	k=15	k=20	k=25	k=30
1				B			B					B				P2	p2	p2	p2	p2	p2	p2	p2
2			B					B					B			P5	p5	p5	p5	p5	p5	p5	p5
3					B				B				B			P3	p3	p3	p3	p3	p3	p3	p3
4					B							B		B		P3/p4	p2	p2	p2	p2	p2	p2	p2
5										B			B	B		P4	p4	p4	p4	p4	p4	p4	p4
6		B						B			B					P1	p1	p1	p1	p1	p1	p1	p1
7						B								B	B	P6	p6	p6	p6	p6	p6	p6	p6
8	B	B									B					P1	p1	p1	p1	p1	p1	p1	p1
9					B				B					B		P3	p3	p3	p3	p3	p3	p3	p3
10					B					B						P4	p4	p4	p4	p4	p4	p4	p4
11			B												B	P5	p5	p5	p5	p5	p5	p5	p5
12						B									B	P6	p6	p6	p6	p6	p6	p6	p6
13				B	B			B					B			P2	p2	p2	p2	p2	p2	p2	p2
14		B						B								P1	p1	p1	p1	p1	p1	p1	p1
15									B					B		P3	p3	p3	p3	p3	p3	p3	p3
16					B				B							P3	p3	p3	p3	p3	p3	p3	p3
17			B										B			P5	p5	p5	p5	p5	p5	p5	p5
18					B								B	B		P3/p4	p4	p4	p3	p3	p3	p3	p3
19						B		B					B			P6	p6	p6	p6	p6	p6	p6	p5
20										B			B			P4	p4	p4	p4	p4	p4	p4	p4
21				B			B					B		B		P2	p2	p2	p2	p2	p2	p2	p2
22	B	B											B		B	P1	p1	p1	p1	p1	p1	p1	p1
23			B										B	B		P3	p5	p4	p5	p3	p5	p5	p5
24				B										B		P3	p2	p2	p2	p2	p2	p2	p2
25						B							B			P6	p6	p6	p6	p6	p6	p6	p3
26			B										B		B	P5	p5	p5	p5	p5	p5	p5	p5
27	B							B			B					P1	p1	p1	p1	p1	p1	p1	p1
28	B			B									B			P4	p1	p2	p2	p2	p2	p2	p2
29					B					B			B			P4	p4	p4	p4	p4	p4	p4	p4
30		B							B				B	B		P3	p3	p3	p3	p3	p3	p3	p3
salah																	4	4	4	3	4	4	6
benar																	26	26	26	27	26	26	24
akurasi																	86,67	86,67	86,67	90	86,67	86,67	80

Pengujian 70 data tidak seimbang variasi 2

no	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Pakar	k=1	k=5	k=10	k=15	k=20	k=25	k=30
1				B			B					B				P2	p2	p2	p2	p2	p2	p2	p2
2			B					B					B			P5	p5	p5	p5	p5	p5	p5	p5
3					B				B				B			P3	p3	p3	p3	p3	p3	p3	p3
4					B							B		B		P3/p4	p2	p2	p2	p3	p3	p3	p2
5										B			B	B		P4	p4	p4	p4	p4	p4	p4	p4
6		B						B			B					P1	p1	p1	p1	p1	p1	p1	p1
7						B								B	B	P6	p6	p6	p6	p6	p6	p3	p3
8	B	B									B					P1	p1	p1	p1	p1	p1	p1	p1
9					B				B					B		P3	p3	p3	p3	p3	p3	p3	p3
10					B					B						P4	p4	p4	p4	p4	p4	p4	p4
11			B												B	P5	p5	p5	p5	p5	p5	p5	p5
12						B									B	P6	p6	p6	p6	p6	p6	p5	p3
13				B	B			B					B			P2	p2	p2	p2	p2	p2	p2	p2
14		B						B								P1	p1	p1	p1	p1	p1	p1	p1
15									B					B		P3	p3	p3	p3	p3	p3	p3	p3
16					B				B							P3	p3	p3	p3	p3	p3	p3	p3
17			B										B			P5	p5	p5	p5	p5	p5	p5	p5
18					B								B	B		P3/p4	p4	p4	p3	p4	p3	p3	p3
19						B		B					B			P6	p6	p6	p6	p5	p5	p5	p5
20										B			B			P4	p4	p4	p4	p4	p4	p4	p4
21				B			B					B		B		P2	p2	p2	p2	p2	p2	p2	p2
22	B	B											B		B	P1	p1	p1	p1	p1	p1	p1	p1
23			B										B	B		P3	p5	p5	p5	p5	p3	p3	p5
24				B										B		P3	p2	p2	p2	p2	p2	p2	p2
25						B							B			P6	p6	p6	p6	p6	p6	p3	p3
26			B										B		B	P5	p5	p5	p5	p5	p5	p5	p5
27	B							B			B					P1	p1	p1	p1	p1	p1	p1	p1
28	B			B									B			P4	p6	p2	p2	p2	p2	p2	p2
29					B					B			B			P4	p4	p4	p4	p4	p4	p4	p4
30		B							B				B	B		P3	p3	p3	p3	p3	p3	p3	p3
salah																	3	3	3	3	2	5	7
benar																	27	27	27	27	28	25	23
akuras																	90	90	90	90	93,33	83,33	76,67

Pengujian 70 data tidak seimbang variasi 3

no	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10	G11	G12	G13	G14	G15	Pakar	k=1	k=5	k=10	k=15	k=20	k=25	k=30
1				B			B					B				P2	p2	p2	p2	p2	p2	p2	p2
2			B					B					B			P5	p5	p5	p5	p5	p5	p5	p5
3					B				B				B			P3	p3	p3	p3	p3	p3	p3	p3
4					B							B		B		P3/p4	p4	p4	p4	p2	p2	p4	p4
5										B			B	B		P4	p4	p4	p4	p4	p4	p4	p4
6		B						B			B					P1	p1	p1	p1	p1	p1	p1	p1
7						B							B	B		P6	p6	p6	p6	p6	p6	p5	p5
8	B	B									B					P1	p1	p1	p1	p1	p1	p1	p1
9					B				B					B		P3	p3	p3	p3	p3	p3	p3	p4
10					B					B						P4	p4	p4	p4	p4	p4	p4	p4
11			B												B	P5	p5	p5	p5	p5	p5	p5	p5
12						B									B	P6	p6	p6	p6	p6	p5	p5	p5
13				B	B			B					B			P2	p4	p2	p2	p2	p5	p2	p2
14		B						B								P1	p1	p1	p1	p1	p1	p1	p1
15									B					B		P3	p3	p3	p3	p3	p3	p3	p3
16					B				B							P3	p3	p3	p3	p3	p3	p3	p3
17			B										B			P5	p5	p5	p5	p5	p5	p5	p5
18					B								B	B		P3/p4	p3	p3	p3	p3	p3	p4	p3
19						B		B					B			P6	p6	p6	p5	p5	p5	p5	p5
20										B			B			P4	p4	p4	p4	p4	p4	p4	p4
21				B			B					B		B		P2	p2	p2	p2	p2	p2	p2	p2
22	B	B											B		B	P1	p6	p1	p6	p1	p1	p1	p1
23			B										B	B		P3	p5	p5	p5	p5	p5	p5	p5
24				B										B		P3	p4	p2	p2	p2	p2	p2	p2
25						B							B			P6	p6	p6	p6	p6	p6	p5	p5
26			B										B		B	P5	p5	p5	p5	p5	p5	p5	p5
27	B							B			B					P1	p1	p1	p1	p1	p1	p1	p1
28	B			B									B			P4	p6	p2	p2	p2	p2	p2	p2
29					B					B			B			P4	p4	p4	p4	p4	p4	p4	p4
30		B							B				B	B		P3	p3	p3	p3	p3	p3	p3	p3
																salah	5	3	5	5	7	7	8
																benar	25	27	25	25	23	23	22
																akurasi	83,33	90	83,33	83,33	76,67	76,67	73,33

Keterangan:

Warna kuning menunjukkan keluaran sistem yang tidak sama dengan identifikasi pakar.