

**DIAGNOSIS PENYAKIT TANAMAN TOMAT MENGGUNAKAN
ALGORITMA MODIFIED K-NEAREST NEIGHBOR (MKNN)**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Prasiwi Meilida Basuki
NIM: 115060801111032



PROGRAM STUDI INFORMATIKA/ILMU KOMPUTER
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2015

PENGESAHAN

DIAGNOSIS PENYAKIT TANAMAN TOMAT MENGGUNAKAN ALGORITMA
MODIFIED K-NEAREST NEIGHBOR (MKNN)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :

Prasiwi Meilida Basuki

NIM: 115060801111032

Skripsi ini telah diuji dan dinyatakan lulus pada
26 November 2015

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Drs. Marji., M.T.

NIP: 19670801 199203 1 001

Nurul Hidayat, S.Pd., M.Sc

NIP: 19680430 200212 1 001

Mengetahui

Ketua Program Studi Informatika

Drs. Marji., M.T.

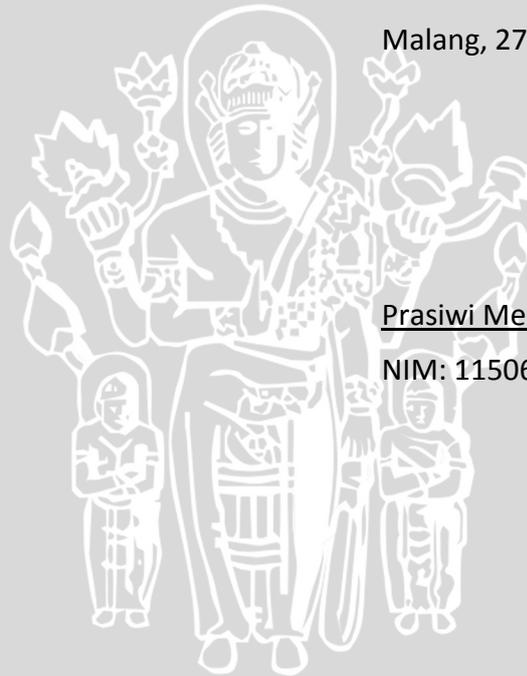
NIP: 19670801 199203 1 001

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, 27 November 2015



Prasiwi Meilida Basuki

NIM: 115060801111032

KATA PENGANTAR

Segala puji bagi Allah SWT yang telah melimpahkan rahmat dan karuniaNya kepada seluruh umat-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul **“Diagnosis Penyakit Tanaman Tomat Menggunakan Algoritma Modified K-Nearest Neighbor(MKNN)”** dengan baik. Shalawat serta salam kami haturkan kepada junjungan kami Nabi Muhammad SAW, semoga kami selalu istiqomah dalam meneladaninya. Skripsi ini merupakan serangkaian mata kuliah dan syarat kelulusan yang harus ditempuh oleh mahasiswa sebagai bukti penerapan dari beberapa mata kuliah yang telah dipelajari selama di bangku perkuliahan.

Tidak terlupakan penulis juga mengucapkan terimakasih yang sebesar-besarnya kepada pihak-pihak yang membantu terselesaikannya laporan Skripsi ini. Ucapan tersebut kamu berikan kepada :

1. Drs. Marji, M.T. dan Nurul Hidayat S.Pd., M.Sc., selaku dosen pembimbing yang telah banyak memberikan ilmu dan saran selama penyusunan skripsi ini.
2. Ir. Sutrisno, M.T, Ir. Heru Nurwasito, M.Kom., Himawat Aryadita, S.T, M.Sc., dan Edy Santoso, S.Si., M.Kom., selaku Ketua, Wakil Ketua I, Wakil Ketua II dan Wakil Ketua III Fakultas Ilmu Komputer, Universitas Brawijaya Malang.
3. Drs. Marji, M.T. dan Issa Arwani, S.Kom, M.Sc., selaku Ketua dan Sekretaris Program Studi Teknik Informatika, Universitas Brawijaya Malang.
4. Bapak dan Ibu Dosen yang telah memberikan ilmu pengetahuan serta wawasan baru kepada kami sebagai anak didik.
5. Seluruh Civitas Akademika Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya yang telah banyak memberikan bantuan dan dukungan selama menempuh studi dan menyelesaikan skripsi ini.
6. Ayahanda Suko Basuki dan Ibunda tercinta Wiwik Wahyuni, terima kasih atas curahan doa, kasih sayang yang tulus serta dukungan moril dan materil yang diberikan selama skripsi ini berlangsung.
7. Adik tercinta Sajulaila Wahyuning Basuki dan Akbar Lucky Basuki yang telah memberikan kasih sayang, semangat dan dukungan dari awal sampai akhir pengerjaan skripsi.
8. Endardy Satrio Wibowo Wasono, terima kasih atas doa, bantuan, dukungan serta kasih sayang yang diberikan selama ini.
9. Teman-teman seperjuangan Dian Malasari, Syela Ukmala, Ika Khoirun Nisak, Afny Wilujeng Setyorini dan Neny Silvia Nurhidayah, terima kasih atas bantuan, dukungan, motivasi dan waktu yang telah diberikan selama pengerjaan skripsi ini.

10. Seluruh teman-teman TIF-C dan informatika UB angkatan 2011 terimakasih atas bantuan, dukungan dan ilmu yang telah diberikan.
11. Teman-teman BEM Kabinet Bersatu II terimakasih atas bantuan, dukungan dan ilmu yang telah diberikan.
12. Semua pihak yang tidak dapat penulis sebutkan satu per satu yang terlibat baik secara langsung maupun tidak langsung demi terselesainya skripsi ini.

Penulis yakin bahwa skripsi ini masih sangat jauh dari sempurna. Saran dan kritik yang membangun dapat disampaikan melalui email penulis meilidapraswi@gmail.com. Harapannya skripsi ini dapat bermanfaat bagi penulis dan para pembaca.

Malang, 27 November 2015

Penulis
meilidapraswi@gmail.com



ABSTRAK

Prasiwi Meilida Basuki. 2015. *Diagnosis Penyakit Tanaman Tomat Menggunakan Algoritma Modified K-Nearest Neighbor (MKNN)*. Skripsi Program Studi Informatika / Ilmu Komputer, Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya, Malang. Dosen Pembimbing: Drs. Marji, M.T. dan Nurul Hidayat S.Pd., M.Sc.

Tanaman tomat adalah salah satu komoditas sayuran yang sangat potensial untuk dikembangkan. Di Indonesia tomat menjadi komoditas sayur unggulan, meskipun kenyataannya produktivitas tomat di Indonesia masih rendah. Selain untuk kebutuhan ekspor, kebutuhan tomat untuk konsumsi domestik pun cenderung mengalami peningkatan tiap tahun. Namun, produktivitas tomat setiap tahun mengalami penurunan yang disebabkan oleh penyakit-penyakit yang menyerang tanaman tomat. Untuk mengatasi penyakit yang menyerang tanaman tomat dibuatlah suatu sistem dengan menggunakan algoritma tertentu. Terdapat 15 gejala dan 6 jenis penyakit utama yang dapat menyerang tanaman tomat yang dapat digunakan sebagai parameter dalam sistem. Untuk mempermudah mendiagnosis penyakit tanaman tomat maka digunakan metode *Modified K-Nearest Neighbor* (MKNN). Metode ini merupakan metode modifikasi dari metode *K-Nearest Neighbor* (K-NN). *Output* dari sistem diagnosa penyakit tanaman tomat menggunakan metode *Modified K-Nearest Neighbor* berupa jenis penyakit dan cara pengendaliannya. Berdasarkan tiga skenario pengujian yang telah dilakukan didapatkan rata-rata nilai akurasi maksimum sebesar 98,92% dan rata-rata nilai akurasi minimum sebesar 89,04%.

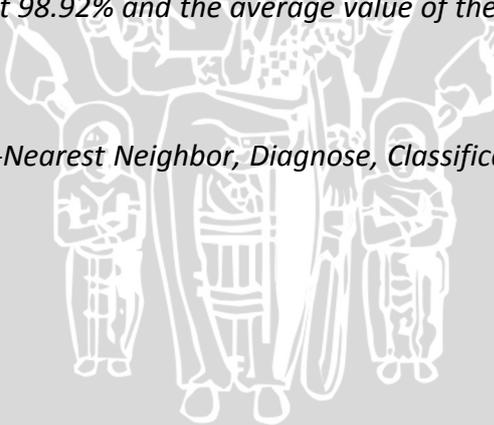
Kata Kunci: *Modified K-Nearest Neighbor*, Diagnosis, Klasifikasi, Penyakit Tanaman Tomat.

ABSTRACT

Prasiwi Meilida Basuki. 2015. Diagnose tomato plant disease using *Modified K-Nearest Neighbor (MKNN) Algorithm*. Undergraduate Thesis. Informatics / Computer Science Departement, Information Technology and Computer Science Program, Brawijaya University, Malang. Advisor: Drs. Marji, M.T. dan Nurul Hidayat S.Pd., M.Sc.

Tomato plants are one of vegetables potential to be developed. In Indonesia the vegetable seed of tomato becomes a commodity, despite the fact that the productivity of tomatoes in Indonesia is still low. In addition to requirements for export, the tomatoes need for domestic consumption also tend to has increased every year. However, the productivity of tomato every year decline caused by diseases that attack tomato plants. To overcome diseases that attack tomato plants invented a system by using a specific algorithm. There are 15 symptoms and 6 kinds of major diseases that can attack tomato plants that can be used as parameters in the system. This method is a modified method of K-Nearest Neighbor (K-NN). The output of the tomato plant disease diagnosis system using the Modified K-Nearest Neighbor such as the type of disease and how to control. Based on the three test scenarios that have been done, obtained an average value of maximum accuracy at 98.92% and the average value of the minimum accuracy of 89.04%..

Keywords : *Modified K-Nearest Neighbor, Diagnose, Classification, Tomato Plant Diseases.*



DAFTAR ISI

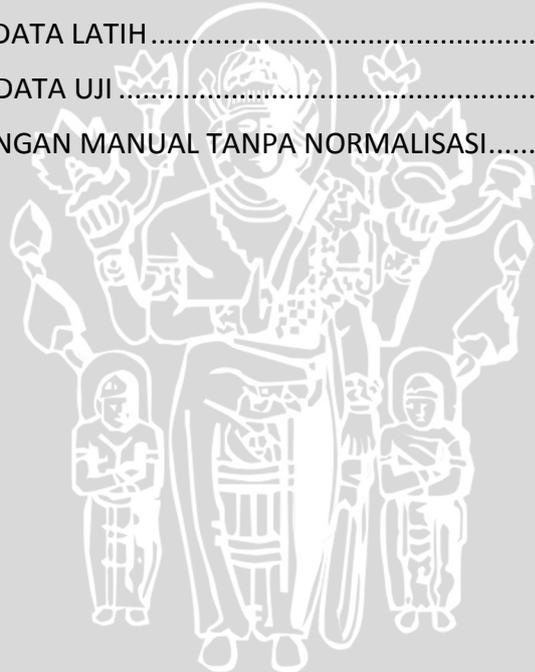
PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT	vii
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiii
DAFTAR PERSAMAAN.....	xv
DAFTAR SOURCE CODE	xvi
DAFTAR LAMPIRAN	xvii
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	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 Penelitian terkait.....	5
2.2 Data mining.....	7
2.2.1 Definisi data mining	7
2.2.2 Tahap-tahap data mining.....	8
2.2.3 Metode data mining.....	8
2.3 Klasifikasi.....	9
2.4 <i>K-Nearest Neighbor</i> (KNN).....	9
2.4.1 Definisi <i>K-Nearest Neighbor</i> (KNN)	10
2.4.2 Proses <i>K-Nearest Neighbor</i> (KNN).....	10
2.5 <i>Modified K-Nearest Neighbor</i> (MKNN)	11
2.5.1 Definisi <i>Modified K-Nearest Neighbor</i> (MKNN).....	11

2.5.2 Proses Modified <i>K-Nearest Neighbor</i> (MKNN)	12
2.6 Tanaman tomat.....	13
2.7 Penyakit tanaman tomat	14
2.7.1 Busuk daun.....	14
2.7.2 Bercak coklat	15
2.7.3 Layu fusarium.....	15
2.7.4 Layu bakteri.....	16
2.7.5 Gejala mosaik	17
2.7.6 Penyakit keriting	18
2.8 Akurasi sistem	19
BAB 3 METODOLOGI	20
3.1 Studi literatur	20
3.2 Pengumpulan data.....	21
3.3 Pengumpulan data.....	22
3.4 Analisis dan perancangan sistem.....	22
3.4.1 Diagram blok sistem.....	23
3.5 Implementasi	24
3.6 Pengujian sistem.....	24
3.7 Evaluasi dan analisis hasil	24
BAB 4 PERANCANGAN.....	26
4.1 Deskripsi sistem	26
4.2 <i>Preprocessing</i> data.....	27
4.3 Perancangan perangkat lunak.....	29
4.3.1 Perancangan <i>database</i>	29
4.3.2 Perancangan proses	32
4.4 Perhitungan manual	39
4.5 Perancangan desain antarmuka	46
4.5.1 Halaman Data Latih.....	46
4.5.2 Halaman Hasil Perhitungan.....	47
4.5.3 Halaman Diagnosis Penyakit Tanaman Tomat.....	48
4.5.4 Halaman Hasil Diagnosis Penyakit Tanaman Tomat.....	49
4.5.5 Halaman Pengujian	49

4.6 Perancangan pengujian	50
4.6.1 Perancangan Pengujian Pengaruh Nilai k Terhadap Akurasi	51
4.6.2 Perancangan Pengujian Pengaruh Jumlah Data Latih Berubah dan Data Uji Tetap Terhadap Akurasi	51
4.6.3 Perancangan Pengujian Pengaruh Data Latih Seimbang dan Data Latih Tidak Seimbang	52
BAB 5 IMPLEMENTASI	53
5.1 Implementasi sistem	53
5.1.1 Spesifikasi perangkat keras	54
5.1.2 Spesifikasi perangkat lunak	54
5.2 Batasan implementasi	54
5.3 Implementasi algoritma	55
5.3.1 Implementasi algoritma perhitungan normalisasi data	55
5.3.2 Implementasi algoritma perhitungan <i>euclidean distance</i>	56
5.3.3 Implementasi algoritma perhitungan validitas data latih	57
5.3.4 Implementasi algoritma perhitungan <i>weight voting</i>	58
5.4 Implementasi antarmuka	59
5.4.1 Implementasi antarmuka halaman data latih	59
5.4.2 Implementasi antarmuka halaman hasil perhitungan	60
5.4.3 Implementasi antarmuka halaman diagnosis penyakit	60
5.4.4 Implementasi antarmuka halaman hasil diagnosis penyakit	61
5.4.5 Implementasi antarmuka halaman pengujian	61
BAB 6 PENGUJIAN DAN ANALISIS	62
6.1 Pengujian pengaruh nilai k	62
6.1.1 Skenario Pengujian Pengaruh Nilai k	62
6.1.2 Analisis Hasil Skenario Pengujian Pengaruh Nilai k	62
6.2 Pengujian pada jumlah data latih berbeda dengan jumlah data uji tetap	64
6.2.1 Skenario Pengujian Pada Jumlah Data Latih Berbeda Dengan Jumlah Data Uji Tetap	64
6.2.2 Analisis Hasil Skenario Pengujian Pada Jumlah Data Latih Berbeda Dengan Jumlah Data Uji Tetap	64



6.3 Pengujian Pengaruh Data Latih Seimbang dan Data Latih Tidak Seimbang Terhadap Akurasi	66
6.3.1 Skenario Pengujian Pengaruh Data Latih Seimbang dan Data Latih Tidak Seimbang Terhadap Akurasi.....	66
6.3.2 Analisis Hasil Skenario Pengujian Pengaruh Data Latih Seimbang dan Tidak Seimbang Terhadap Akurasi.....	66
BAB 7 PENUTUP	68
7.1 Kesimpulan.....	68
7.2 Saran	69
DAFTAR PUSTAKA.....	70
LAMPIRAN A NILAI PEMBOBOTAN GEJALA PENYAKIT TOMAT.....	72
LAMPIRAN B HASIL WAWANCARA DENGAN PAKAR.....	74
LAMPIRAN C DAFTAR DATA LATIH.....	76
LAMPIRAN D DAFTAR DATA UJI.....	81
LAMPIRAN E PERHITUNGAN MANUAL TANPA NORMALISASI.....	84



DAFTAR TABEL

Tabel 2.1 Penelitian Terkait	6
Tabel 3.1 Penentuan Kebutuhan Data Penelitian.....	21
Tabel 4.1 Nilai Pembobotan 15 Gejala Penyakit Tanaman Tomat	27
Tabel 4.2 Pasangan gejala penyakit dan penyakit yang menyerang tanaman tomat	29
Tabel 4.3 Struktur Tabel data_training	30
Tabel 4.4 Struktur Tabel diagnosis.....	31
Tabel 4.5. Struktur Tabel nilai_k	31
Tabel 4.6 Struktur Tabel validitas.....	31
Tabel 4.7 Struktur Tabel weight_voting.....	31
Tabel 4.8 Dataset perhitungan manual.....	39
Tabel 4.9 Data Latih	40
Tabel 4.10 Data Uji	40
Tabel 4.11 Data Latih Hasil Normalisasi.....	41
Tabel 4.12 Data Uji Hasil Normalisasi	41
Tabel 4.14 Perhitungan Validitas	43
Tabel 4.15 Tabel Perhitungan Euclidean Data Uji 1.....	43
Tabel 4.13 Perhitungan Euclidean Data Latih	44
Tabel 4.16 Hasil Pencarian Nilai Weight voting Terbesar Sebanyak Nilai k.....	45
Tabel 4.17 Hasil Perhitungan 6 Data Uji	46
Tabel 4.18 Pengaruh Nilai k terhadap akurasi	51
Tabel 4.19 Pengaruh Jumlah Data Latih Berubah dan Data Uji Tetap.....	51
Tabel 4.20 Pengaruh Data Latih Seimbang dan Data Latih Tidak Seimbang	52
Tabel 5.1 Spesifikasi Perangkat Keras	54
Tabel 5.2 Spesifikasi Perangkat Lunak	54
Tabel 6.1 Hasil pengujian pengaruh nilai k	63
Tabel 6.2 Hasil Pengujian Pada Jumlah Data Latih Berbeda Dengan Jumlah Data Uji Tetap	65
Tabel 6.3 Hasil pengujian pengaruh data latih seimbang dan tidak seimbang	66

DAFTAR GAMBAR

Gambar 2.1 Penyakit Busuk Daun.....	14
Gambar 2.2 Penyakit Bercak Coklat.....	15
Gambar 2.3 Penyakit Layu Fusarium	16
Gambar 2.4 Penyakit Layu Bakteri	17
Gambar 2.5 Penyakit Mosaik	18
Gambar 2.6 Penyakit Keriting	19
Gambar 3.1 Diagram Blok Metodologi Penelitian	20
Gambar 3.2 Diagram Blok Proses Diagnosis Penyakit Tanaman Tomat Dengan Metode MKNN	23
Gambar 4.1 Diagram Alir Tahap Perancangan Sistem	26
Gambar 4.2 Rancangan Database Sistem	29
Gambar 4.3 Diagram Alir Sistem	32
Gambar 4.4 Diagram Alir Proses Perhitungan Normalisasi	33
Gambar 4.5 Diagram Alir Proses Algoritma MKNN.....	34
Gambar 4.6 Diagram Alir Proses Perhitungan Validitas	36
Gambar 4.7 Diagram Alir Proses Perhitungan Jarak Euclidean	38
Gambar 4.8 Proses Perhitungan Weight voting.....	38
Gambar 4.9 Perancangan Halaman Data Latih	46
Gambar 4.10 Perancangan Halaman Hasil Perhitungan.....	47
Gambar 4.11 Perancangan Halaman Diagnosis Penyakit Tanaman Tomat.....	48
.....	49
Gambar 4.12 Perancangan Halaman Hasil Diagnosis Penyakit Tanaman Tomat .	49
Gambar 4.13 Perancangan Halaman Pengujian	50
Gambar 5.1 Pohon Implementasi	53
Gambar 5.2 Implementasi Antarmuka Halaman Data Latih.....	59
Gambar 5.3 Implementasi Antarmuka Halaman Detail Perhitungan	60
Gambar 5.4 Implementasi Antarmuka Halaman Diagnosis Penyakit	60
Gambar 5.5 Implementasi Antarmuka Halaman Hasil Diagnosis Penyakit	61
Gambar 5.6 Implementasi Antarmuka Halaman Pengujian	61
Gambar 6.1 Grafik pengaruh nilai k terhadap rata-rata akurasi.....	63

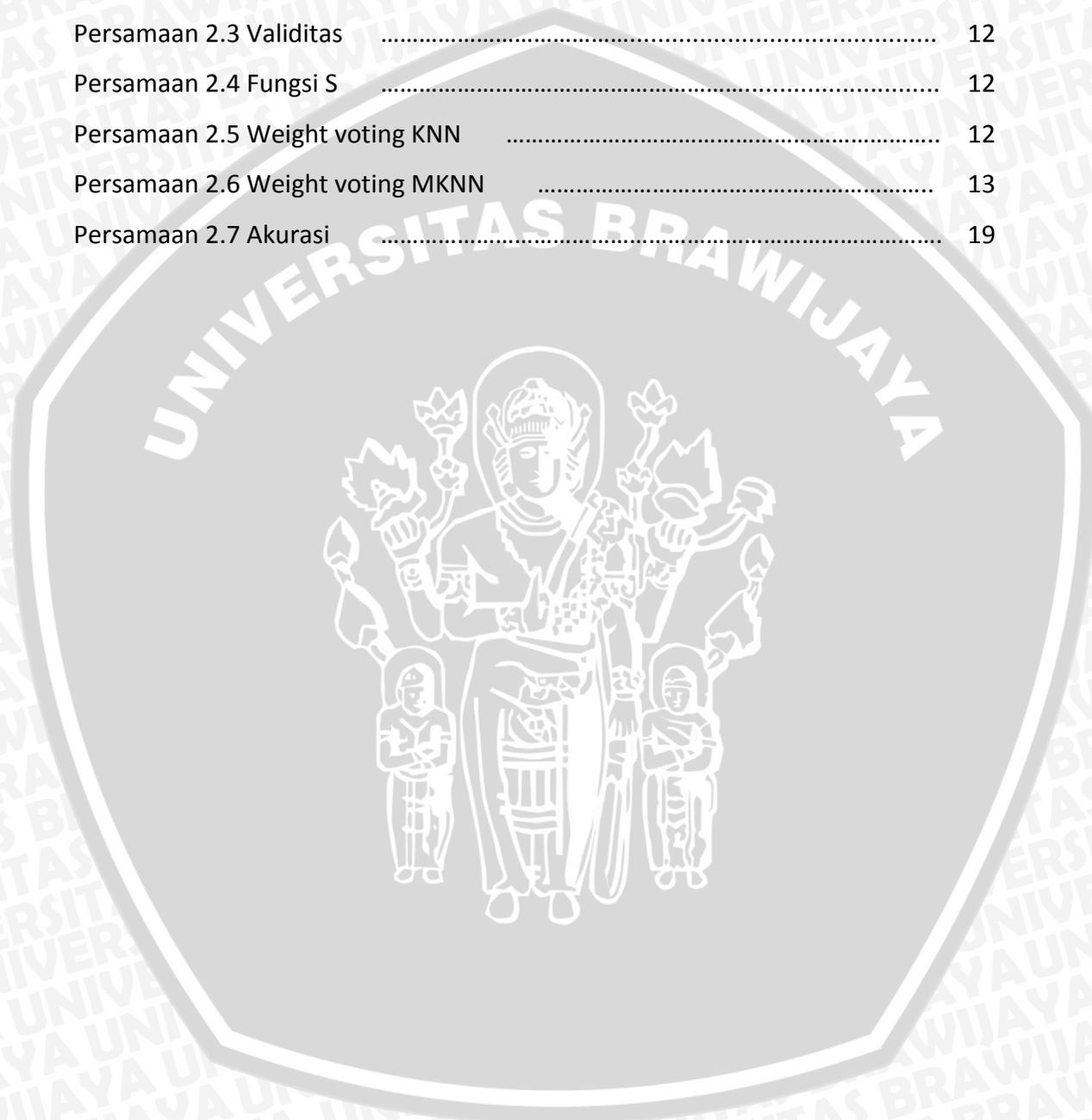
Gambar 6.2 Grafik penambahan jumlah data latih terhadap hasil akurasi..... 65

Gambar 6.3 Grafik pengaruh data latih seimbang dan tidak seimbang terhadap akurasi 67



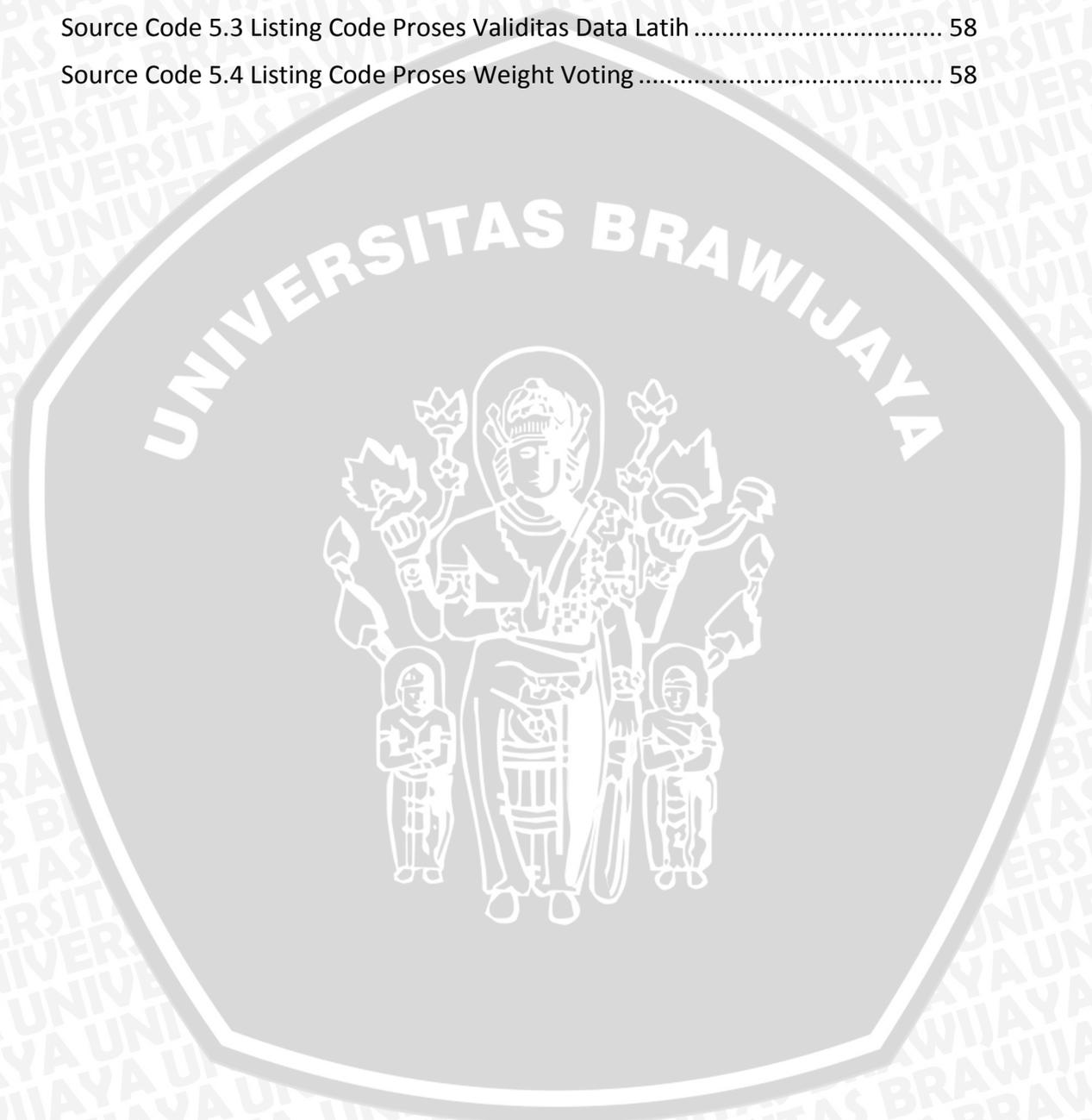
DAFTAR PERSAMAAN

Persamaan 2.1 Jarak Euclidean	10
Persamaan 2.2 Min-max normalisasi	11
Persamaan 2.3 Validitas	12
Persamaan 2.4 Fungsi S	12
Persamaan 2.5 Weight voting KNN	12
Persamaan 2.6 Weight voting MKNN	13
Persamaan 2.7 Akurasi	19



DAFTAR SOURCE CODE

Source Code 5.1 Listing Code Proses Normalisasi Data	55
Source Code 5.2 Listing Code Proses Euclidean Distance	57
Source Code 5.3 Listing Code Proses Validitas Data Latih	58
Source Code 5.4 Listing Code Proses Weight Voting	58



DAFTAR LAMPIRAN

LAMPIRAN A NILAI PEMBOBOTAN GEJALA PENYAKIT TOMAT.....	72
LAMPIRAN B HASIL WAWANCARA DENGAN PAKAR.....	74
LAMPIRAN C DAFTAR DATA LATIH.....	76
C.1 30 Data Latih.....	76
C.2 60 Data Latih.....	77
C.3 90 Data Latih.....	78
LAMPIRAN D DAFTAR DATA UJI.....	81
D.1 50 Data uji untuk pengujian data uji tetap dan data latih berbeda....	81
D.2 50 data uji untuk pengujian data latih seimbang dan tidak seimbang	82
LAMPIRAN E PERHITUNGAN MANUAL TANPA NORMALISASI.....	84



BAB 1 PENDAHULUAN

1.1 Latar belakang

Tanaman tomat adalah salah satu komoditas sayuran yang sangat potensial untuk dikembangkan. Tanaman ini dapat ditanam secara luas di dataran rendah sampai dataran tinggi pada lahan bekas sawah dan lahan kering (Setiawati, et al., 2001). Di Indonesia tomat menjadi komoditas sayur unggulan, meskipun kenyataannya produktivitas tomat di Indonesia masih rendah. Selama kurun waktu 1986-2006 Indonesia mengeksport tomat segar rata-rata tiap tahunnya sebesar 1.856.962 Kg ke pasar internasional dengan nilai sebesar US\$ 554.004. Selain untuk kebutuhan ekspor, kebutuhan tomat untuk konsumsi domestik pun cenderung mengalami peningkatan tiap tahun (Yasa, et al., 2012). Berdasarkan data dari Kementerian Pertanian produktivitas tomat pada tahun 2013-2014 mengalami penurunan. Pada tahun 2013 produktivitas tomat di Indonesia sebesar 16,61 Ton/Ha, sedangkan pada tahun 2014 produktivitas tomat di Indonesia menjadi 15,52 Ton/Ha (Kementerian Pertanian, 2015).

Tanaman tomat menjadi salah satu komoditas hortikultura yang bernilai ekonomi tinggi dan masih memerlukan penanganan serius, terutama dalam hal peningkatan hasilnya dan kualitas buahnya. Indonesia sebagai salah satu Negara beriklim tropis mempunyai potensi dan kesempatan yang cukup besar untuk memanfaatkan peluang usaha dibidang hortikultura, khususnya tomat (Yasa, et al., 2012).

Penyakit yang menyerang tanaman tomat merupakan salah satu penyebab rendahnya produktivitas tomat. Pengetahuan petani mengenai penyakit tomat pun masih sangat minim, sehingga penanganannya pun juga lambat dan menghasilkan tomat yang kurang berkualitas. Untuk mendiagnosis penyakit tanaman tomat para petani akan bertanya pada pakar atau ahli tanaman tomat untuk berkonsultasi mengenai jenis penyakit apa yang menyerang tanaman tomat mereka. Namun, pakar atau ahli tanaman tomat di Indonesia saat ini jumlahnya sangat terbatas sehingga banyak petani yang kesulitan untuk berkonsultasi untuk mendiagnosis penyakit tanaman tomat. Untuk mengetahui jenis penyakit tanaman tomat yang semula harus datang ke pakar atau ahli tanaman tomat yang membutuhkan banyak waktu dan relatif lama dapat diperbaiki dengan aplikasi komputer. Aplikasi komputer yang dapat digunakan yaitu membuat suatu sistem untuk mendiagnosis penyakit tanaman tomat dengan menggunakan metode tertentu. Dengan sistem ini informasi dapat disampaikan jauh lebih cepat dan mudah dimengerti.

Metode *K-Nearest Neighbor* merupakan salah satu metode klasifikasi yang sederhana dan banyak digunakan. Metode KNN (*K – Nearest Neighbor*) adalah suatu metode yang menggunakan klasifikasi ketetanggaan yang bertujuan untuk mengklasifikasikan objek baru berdasarkan atribut (*feature*) dan *training sample*. Kelebihan metode KNN ini yaitu lebih efektif di data latih dengan jumlah besar dan dapat menghasilkan data yang lebih akurat. Sedangkan kelemahan

metode KNN ini yaitu terletak pada rendahnya tingkat akurasi pada dataset multidimensional dapat diatasi dengan melakukan modifikasi terhadap metode KNN (Parvin, et al., 2010).

Metode *Modified K-Nearest Neighbor*(MKNN) merupakan suatu metode yang mengklasifikasikan objek baru dengan cara menempatkan label kelas data sesuai dengan nilai k yang divalidasi. Nilai k dan besarnya data latih berpengaruh dalam penggunaan metode klasifikasi MKNN (Zainuddin, et al., 2013).

Pemecahan masalah untuk mendiagnosis penyakit pada tanaman tomat pernah diusulkan pada penelitian sebelumnya. Pada penelitian pertama membahas tentang sistem pakar untuk menDiagnosis dan menanggulangi hama dan penyakit pada tomat buah. Sistem pakar yang dikembangkan pada penelitian ini berbasis android dan menggunakan metode pengurutan untuk pencapaian solusi untuk identifikasi penyakit (Supriyanto, et al., 2013).

Penelitian lainnya membahas tentang penerapan metode M-KNN untuk mengklasifikasikan penyakit tanaman kedelai dengan menggunakan data morfologi tanaman kedelai. Pada penelitian ini menghasilkan tingkat akurasi rata-rata maksimum sebesar 92,74% dengan nilai k=3 (Zainuddin, et al., 2013).

Berdasarkan permasalahan tersebut maka penulis menggunakan metode *Modified K-Nearest Neighbor* untuk diagnosis penyakit tanaman tomat. Input dari sistem ini berupa gejala-gejala yang terdapat pada tanaman tomat, sedangkan outputnya berupa hasil diagnosis penyakit dan cara pengendaliannya. Diharapkan hasil dari implementasi Metode *Modified K-Nearest Neighbor* untuk diagnosis penyakit tanaman tomat dapat membantu petani tomat lebih cepat dalam mengidentifikasi jenis penyakit yang menyerang tanaman tomatnya.

1.2 Rumusan masalah

Berdasarkan uraian latar belakang tersebut, maka rumusan masalah adalah sebagai berikut :

1. Bagaimana mengimplementasikan metode *Modified K-Nearest Neighbor* untuk diagnosis penyakit pada tanaman tomat?
2. Bagaimana tingkat akurasi metode *Modified K-Nearest Neighbor* untuk diagnosis penyakit tanaman tomat?

1.3 Tujuan

Penelitian yang dilakukan ini memiliki tujuan :

1. Untuk mengimplementasikan metode *Modified K-Nearest Neighbor* pada aplikasi Diagnosis penyakit tanaman tomat.
2. Menghitung tingkat akurasi yang diperoleh dari hasil implementasi metode *Modified K-Nearest Neighbor* untuk pengklasifikasian penyakit tanaman tomat.

1.4 Manfaat

Manfaat yang bisa diperoleh dari penelitian ini yaitu diharapkan bisa memberikan informasi untuk memprediksi penyakit yang menyerang tanaman tomat, sehingga memudahkan petani tomat dalam mengambil tindakan dalam penanganan penyakit yang menyerang tanaman tomat selanjutnya.

1.5 Batasan masalah

Batasan masalah dalam penelitian ini yaitu:

1. Data yang digunakan untuk pengujian ini yaitu data yang diperoleh dari Balai Pengkajian Teknologi Pertanian (BPTP) Jawa Timur .
2. Data yang digunakan berjumlah 288 diambil pada tahun 2015 berdasarkan penelitian di Balai Pengkajian Teknologi Pertanian Malang dan dikonsultasikan dengan Prof. (Riset). Dr. Ir, Moh. Cholil Mahfud, M.S.
3. Data terdiri dari 6 jenis penyakit dan 15 gejala penyakit tanaman tomat.
4. Proses Diagnosis berdasarkan gejala yang tampak pada batang, daun dan buah tanaman tomat.
5. Sistem yang dirancang hanya untuk mendiagnosis penyakit tanaman tomat dan cara pengendaliannya.
6. Metode yang digunakan pada sistem ini yaitu metode *Modified K-Nearest Neighbor*.

1.6 Sistematika pembahasan

Untuk mencapai tujuan yang diharapkan, maka sistematika penulisan yang disusun dalam tugas akhir ini adalah sebagai berikut:

BAB I Pendahuluan

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

BAB II Tinjauan Pustaka

Tinjauan Pustaka menjelaskan tentang kajian pustaka terkait dengan penelitian yang telah ada seperti penelitian tentang "*Implementasi algoritma Modified K-Nearest Neighbor (MKNN) untuk menentukan tingkat resiko penyakit lemak darah (Profil Lipid)*", "*dan tentang "Penerapan Algoritma Modified K-Nearest Neighbor pada pengklasifikasian penyakit tanaman kedelai"*". Dasar teori yang diperlukan untuk mendukung penelitian adalah sistem pakar, Metode *K-Nearest Neighbor*, Metode *Modified K-Nearest Neighbor*, Pengertian tanaman tomat, jenis penyakit tanaman tomat, dan akurasi sistem.

BAB III Metodologi

Membahas tentang metode yang digunakan dalam penulisan yang terdiri dari studi literatur, pengambilan data sampel dari penyakit tanaman tomat, *Analisis*

kebutuhan, *Analisis* dan perancangan sistem, implementasi, uji coba sistem dan pengambilan kesimpulan dari aplikasi untuk Diagnosis penyakit pada tanaman tomat menggunakan metode *Modified K-Nearest Neighbor*.

BAB IV Analisis Kebutuhan dan Perancangan

Membahas tentang *Analisis* kebutuhan dari pemodelan sistem pakar untuk Diagnosis penyakit pada tanaman tomat menggunakan metode *Modified K-Nearest Neighbor*.

BAB V Implementasi

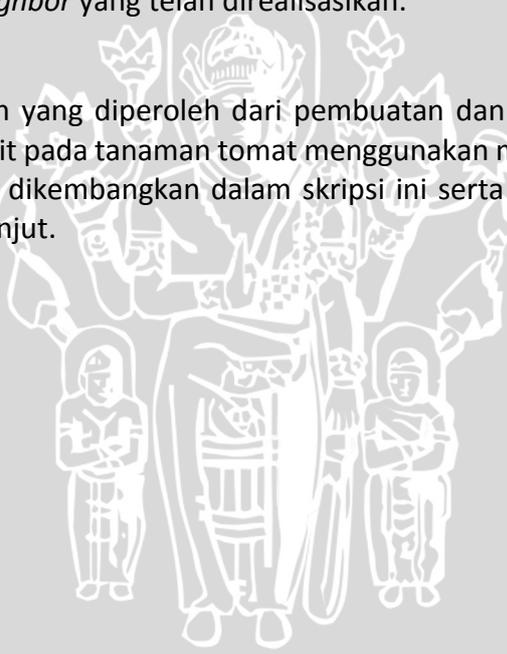
Membahas tentang hasil perancangan dari analisis kebutuhan dan implementasi dari aplikasi Diagnosis penyakit pada tanaman tomat menggunakan metode *Modified K-Nearest Neighbor*.

BAB VI Pengujian dan Analisis

Memuat tentang hasil pengujian dan analisis terhadap implementasi dari aplikasi untuk Diagnosis penyakit pada tanaman tomat menggunakan metode *Modified K-Nearest Neighbor* yang telah direalisasikan.

BAB VII Penutup

Memuat kesimpulan yang diperoleh dari pembuatan dan pengujian aplikasi untuk Diagnosis penyakit pada tanaman tomat menggunakan metode *Modified K-Nearest Neighbor* yang dikembangkan dalam skripsi ini serta saran-saran untuk pengembangan lebih lanjut.



BAB 2 LANDASAN KEPUSTAKAAN

Pada bab dua, terdiri dari penelitian terkait dan dasar teori. Penelitian terkait membahas penelitian yang telah ada dan penelitian yang diusulkan. Dasar teori membahas teori yang diperlukan untuk menyusun penelitian yang diusulkan. Kajian pustaka pada penelitian ini adalah membandingkan penelitian yang akan diusulkan dengan penelitian sebelumnya yang berjudul "*Sistem Pakar Diagnosis dan Penanggulangan Hama dan Penyakit Tomat Buah (Solanum lycopersicum) Dataran Tinggi Berbasis Android*" dan "*Penerapan Algoritma Modified K-Nearest Neighbor pada pengklasifikasian penyakit tanaman kedelai*". Dasar teori yang diperlukan berdasarkan latar belakang dan rumusan masalah adalah konsep dasar *Data mining*, metode K-NN, metode *Modified K-Nearest Neighbor* dan penjelasan tentang penyakit tanaman tomat.

2.1 Penelitian terkait

Pada sub bab ini dilakukan *Analisis* perbandingan terhadap beberapa penelitian sebelumnya terkait diagnosis penyakit tanaman tomat. Selain itu, juga akan dibahas penelitian terkait tentang penggunaan metode *Modified K-Nearest Neighbor* yang juga diusulkan pada penelitian penulis. Kajian penelitian sebelumnya yang telah dilakukan dan penelitian yang diusulkan ditunjukkan pada Tabel 2.1.

Penelitian pertama adalah penelitian yang dilakukan oleh Supriyanto dkk. Penelitian ini berjudul "*Sistem Pakar Diagnosis dan Penanggulangan Hama dan Penyakit Tomat Buah (Solanum lycopersicum) Dataran Tinggi Berbasis Android*", penyakit tanaman tomat dijadikan sebagai masalah utama dalam penelitian dan aplikasi dikembangkan berbasis android. Metode yang digunakan untuk menDiagnosis penyakit tanaman tomat yaitu dengan metode pengurutan untuk pencapaian solusi untuk identifikasi penyakit (Supriyanto, et al., 2013).

Penelitian kedua adalah penelitian yang dilakukan oleh Sofa Zainuddin dkk. Penelitian ini berjudul "*Penerapan Algoritma Modified K-Nearest Neighbor pada pengklasifikasian penyakit tanaman kedelai*", pada penelitian menjelaskan tentang penerapan metode *Modified K-Nerarest Neighbor* untuk mengklasifikasikan penyakit tanaman kedelai dengan menggunakan data morfologi tanaman kedelai. Untuk setiap jenis penyakit terdapat 34 variabel pada data morfologi tanaman kedelai. Pada penelitian ini tingkat akurasi tertinggi dari sistem klasifikasi penyakit tanaman kedelai dengan menggunakan metode MKNN dengan menggunakan 300 data latih adalah sebesar 92,74% dengan nilai $k=3$ (Zainuddin, et al., 2013) .

Penelitian ketiga adalah penelitian yang dilakukan oleh Noviana Ayu Kumalasari dkk. Penelitian ini berjudul "*Implementasi algoritma Modified K-Nearest Neighbor (MKNN) untuk menentukan tingkat resiko penyakit lemak darah (Profil Lipid)*", pada penelitian ini menjelaskan tentang implemntasi algoritma MKNN untuk memudahkan dalam penentuan dan pengklasifikasian tingkat resiko

penyakit lemak darah. Parameter yang digunakan ada 4 yaitu kolesterol total, kolesterol HDL, kolesterol LDL dan trigliserida. Berdasarkan hasil pengujian sistem, didapatkan rata-rata nilai akurasi maksimum yang dihasilkan sistem sebesar 85,81% pada jumlah data latih 140 dan rata-rata akurasi minimum sistem sebesar 73,55% yang dihasilkan pada jumlah data latih 60 (Kumalasari, et al., 2014).

Perbedaan pada penelitian sebelumnya dan penelitian ini adalah data masukan yang digunakan pada penelitian ini berupa gejala-gejala dari penyakit tanaman tomat yang terdiri dari 15 gejala penyakit. Selanjutnya proses perhitungan yang digunakan yaitu dengan metode MKNN. Sedangkan output dalam sistem ini berupa Diagnosis penyakit tanaman tomat dan cara pengendaliannya.

Tabel 2.1 Penelitian Terkait

No	Judul	Objek dan Input	Metode (Proses)	Hasil (Output)
1	Sistem Pakar Diagnosis dan Penanggulangan Hama dan Penyakit Tomat Buah (Solanum lycopersicum) Dataran Tinggi Berbasis Android	Inputan sistem pada penelitian ini berupa kerusakan-kerusakan yang muncul pada tanaman tomat yang ditanyakan secara berurut.	<ol style="list-style-type: none"> 1. Memasukkan jenis serangan atau gejala yang terjadi. 2. Pengguna akan diarahkan untuk mendapatkan jawaban dari permasalahan yang dihadapi di lapangan. 3. Pencapaian solusi untuk identifikasi penyakit dilakukan secara terurut dari parent sampai dengan child terakhir. 	- Hasil Diagnosis dan rekomendasi pengendalian hama dan penyakit tanaman tomat.
2	Penerapan Algoritma <i>Modified K-Nearest Neighbor</i> pada pengklasifikasian penyakit tanaman kedelai	Inputan sistem ini berupa Data morfologi tanaman kedelai	<ol style="list-style-type: none"> 1. Menghitung validitas data pada semua data di data latih 2. Menghitung <i>Euclidean</i>. 3. Mencari <i>weight voting</i> pada semua data uji menggunakan validitas data. 	<ul style="list-style-type: none"> - Kesimpulan penyakit beserta prosentase tingkat keyakinannya dan solusi untuk penanganan/pengendaliannya. - Tingkat akurasi rata-rata maksimum yang didapat adalah 92,74% dengan nilai k=3.

3	Implementasi Algoritma <i>Modified K-Nearest Neighbor</i> (MKNN) untuk menentukan tingkat resiko penyakit lemak darah (Profil Lipid)	Inputan sistem berupa kadar kolesterol total, kolesterol HDL, kolesterol LDL dan trigliserida.	<ol style="list-style-type: none"> 1. Memasukkan dataset penyakit lemak darah dan nilai k 2. Menghitung nilai validitas 3. Menghitung nilai <i>Euclidean</i> 4. Menghitung normalisasi 5. Hasil akurasi tingkat resiko penyakit lemak darah. 	<ul style="list-style-type: none"> - Kategori Tingkat resiko penyakit lemak darah yang terdiri dari rendah, waspada dan tinggi. - Rata-rata nilai akurasi maksimum yang dihasilkan sistem sebesar 85,81% pada jumlah data latih 140 dan rata-rata akurasi minimum sistem sebesar 73,55% yang dihasilkan pada jumlah data latih 60.
4	Diagnosis penyakit tanaman tomat menggunakan metode <i>Modified K-Nearest Neighbor</i> .	Inputan sistem berupa pertanyaan tentang gejala penyakit yang menyerang tanaman tomat.	<ol style="list-style-type: none"> 1. Menjawab pertanyaan dari sistem yang berupa gejala penyakit tanaman tomat. 2. Normalisasi data. 3. Menghitung nilai validitas. 4. Menghitung nilai <i>Euclidean</i>. 5. Menghitung <i>weighted voting</i> (pembobotan) dari data. 6. Menentukan kelas dari data uji tersebut. 	<ul style="list-style-type: none"> - Output dari sistem yaitu hasil Diagnosis berupa jenis penyakit dan cara pengendaliannya.

2.2 Data mining

Pada sub bab ini akan dijelaskan mengenai definisi dari *Data mining*, tahap-tahap dalam *Data mining* dan metode yang terdapat pada *Data mining*.

2.2.1 Definisi data mining

Data mining juga didefinisikan sebagai satu set teknik yang digunakan secara otomatis untuk mengeksplorasi secara menyeluruh dan membawa ke permukaan relasi-relasi yang kompleks pada suatu set data yang besar (Moertini, 2002).

Menurut Larose, *Data mining* adalah bidang yang digabung dari beberapa bidang keilmuan yang menyatukan teknik dari pembelajaran mesin, pengenalan pola, statistik, database dan visualisasi untuk pengenalan permasalahan pengambilan informasi dari *database* yang benar (Larose, 2005).



Ada beberapa definisi *Data mining* antara lain (Kusnawi, 2007):

1. *Data mining* adalah *Analisis* otomatis dari suatu data yang jumlahnya sangat besar atau kompleks dengan tujuan untuk menemukan pola atau kecenderungan penting yang biasanya tidak disadari keberadaannya.
2. *Data mining* adalah serangkaian proses untuk menggali nilai tambah dari suatu kumpulan data berupa pengetahuan yang selama ini tidak diketahui secara manual.
3. *Data mining* atau *Knowledge Discovery in Databases* (KDD) adalah pengambilan informasi yang tersembunyi, dimana informasi tersebut sebelumnya tidak dikenal dan berpotensi bermanfaat. Proses ini meliputi sejumlah pendekatan teknis yang berbeda, seperti *clustering*, *data summarization* dan *learning classification rules*.

2.2.2 Tahap-tahap data mining

Dalam *Data mining* terdapat beberapa tahap-tahap yang dilakukan untuk menemukan suatu model dari data, diantaranya (Kantardzic, 2003):

1. Merumuskan permasalahan
Dalam tahapan perumusan masalah ini ditetapkan rumusan masalah dan variabel-variabel yang terlibat.
2. Pengumpulan data
Pada tahapan ini dilakukan proses pembuatan data dan pengumpulan data yang akan digunakan.
3. *Preprocessing* data
Dalam tahapan ini dilakukan seleksi data yang akan digunakan dalam proses.
4. Estimasi model seleksi dan implementasi
Pada tahapan ini dilakukan estimasi model dan implementasi terhadap metode *Data mining* yang tepat.
5. Menafsirkan model dan menarik kesimpulan
Pada tahapan terakhir dilakukan penafsiran model dari *Data mining* dan ditarik kesimpulan dari hasil akhir yang didapat.

2.2.3 Metode data mining

Secara garis besar metode pada *Data mining* dapat dikelompokkan menjadi 2 kategori utama, yaitu:

1. *Descriptive mining*
Descriptive mining yaitu suatu proses untuk menemukan karakteristik penting dari data di suatu basis data. Teknik *Data mining* yang termasuk

dalam *descriptive mining* adalah *clustering*, *sequential mining* dan *association*.

2. Predictive mining

Predictive mining merupakan suatu metode untuk menemukan pola dari data dengan menggunakan beberapa variabel lain di masa depan. Salah satu teknik yang terdapat dalam *predictive mining* yaitu metode klasifikasi.

2.3 Klasifikasi

Klasifikasi adalah salah satu metode dari *Data mining* untuk menemukan model atau fungsi yang menjelaskan atau membedakan konsep atau kelas data, dengan tujuan untuk dapat memperkirakan kelas dari suatu objek yang labelnya tidak diketahui (Pramudiono, 2003). Klasifikasi bertujuan untuk memprediksi kelas dari suatu objek yang labelnya tidak diketahui.

Selain itu klasifikasi juga dapat berarti suatu teknik dengan melihat pada kelakuan dan atribut kelompok yang telah didefinisikan. Teknik tersebut dapat memberikan klasifikasi data baru dengan memanipulasi data yang telah diklasifikasikan sebelumnya dan hasil tersebut digunakan untuk memberikan sejumlah aturan-aturan pada data baru untuk diklasifikasikan. Teknik ini menggunakan *supervised induction*, yang memanfaatkan kumpulan pengujian dari record yang terklasifikasi untuk menentukan kelas-kelas tambahan (Kusnawi, 2007).

Secara umum, proses klasifikasi biasanya dibagi menjadi dua fase, yaitu (Pramudiono, 2003):

1. Learning

Pada fase ini, sebagian data yang telah diketahui kelas datanya diumpankan untuk membentuk model perkiraan.

2. Test

Pada fase test, model yang sudah terbentuk diuji dengan sebagian data lainnya untuk mengetahui akurasi dari model tersebut. Bila kurasinya mencukupi maka model ini dapat digunakan untuk memprediksi kelas data yang belum diketahui.

Pada proses klasifikasi terdapat beberapa metode, antara lain *decision tree*, *Bayesian*, *fuzzy*, *neural network*, *support vector machine (SVM)* dan *k-nearest neighbor* (Pramudiono, 2003).

2.4 K-Nearest Neighbor (KNN)

Pada sub bab ini akan dijelaskan mengenai definisi dari K-Nearest Neighbor (KNN), proses dari (KNN) dan tahapan dari proses KNN.

2.4.1 Definisi *K-Nearest Neighbor* (KNN)

K-Nearest Neighbor (KNN) merupakan suatu metode yang biasa digunakan untuk klasifikasi data. Algoritma KNN adalah sebuah metode untuk mengklasifikasikan objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. KNN adalah suatu metode yang menggunakan algoritma *supervised* dengan hasil dari *query instance* yang baru diklasifikasikan berdasarkan mayoritas dari kategori pada KNN. Tujuan dari algoritma ini yaitu mengklasifikasikan objek baru berdasarkan atribut dan *training sample* (Aktivia, 2012).

Prinsip umum dari algoritma ini adalah menemukan *k* data *training* untuk menentukan *K-Nearest Neighbor* berdasarkan ukuran jarak. Selanjutnya mayoritas dari *k* tetangga terdekat akan menjadi dasar untuk memutuskan kategori dari sample berikutnya (Yu, 2010).

Metode KNN merupakan metode yang sederhana, mudah diimplementasikan, dan efektif jika data *training* besar. Meskipun memiliki beberapa kelebihan metode KNN juga memiliki beberapa kelemahan seperti berikut (Parvin, et al., 2010):

1. Menggunakan perhitungan jarak yang belum diketahui pasti fungsi jarak yang digunakan.
2. Biaya komputasi yang tinggi karena perlu menghitung jarak setiap data *training*.
3. Perlu menentukan nilai *k* parameter, jumlah tetangga terdekat.
4. Membutuhkan memori yang besar.
5. Belum diketahui atribut yang lebih baik untuk menghasilkan hasil terbaik.
6. Rendahnya tingkat akurasi pada dataset multidimensi.

2.4.2 Proses *K-Nearest Neighbor* (KNN)

Prinsip kerja dari algoritma *K-Nearest Neighbor* adalah membandingkan data uji yang diberikan dengan data *training* yang mirip dan mencari jarak terdekat antara data yang dievaluasi dengan *k* dari tetangga terdekat dalam data *training*. Ukuran dekat atau jauhnya tetangga biasanya dihitung berdasarkan variabel (Han & Kamber, 2006).

1. Interval Scaled Variable

Interval scaled variable merupakan ukuran-ukuran kontinu dari skala linier. Ukuran-ukuran tersebut berupa ukuran jarak, yang umum digunakan adalah jarak *Euclidean*. Perhitungan untuk menghitung jarak *Euclidean* dengan persamaan berikut (Han & Kamber, 2006):

$$d(X_1, X_2) = \sqrt{\sum_{i=1}^n (X_{1i} - X_{2i})^2} \quad (2-1)$$

Dimana X_1 merupakan nilai variabel data testing dan X_2 merupakan nilai variabel dari data *training*. Data kontinu dapat digunakan rumusan normalisasi

atau standardisasi sebelum melakukan klasifikasi. Normalisasi bertujuan untuk mencegah atribut yang memiliki rentang terlalu besar dengan atribut yang bernilai kecil. Perhitungan min-max normalisasi dapat digunakan untuk mengubah atribut A dengan nilai v menjadi v' dalam range $[0,1]$. Perhitungan min-max normalisasi dengan persamaan (Han & Kamber, 2006) :

$$v' = \frac{v - \min_A}{\max_A - \min_A} \times (\text{new}_{\max A} - \text{new}_{\min A}) + \text{new}_{\min A} \quad (2-2)$$

Dimana :

V' = nilai normalisasi

V = nilai data yang akan dinormalisasi

\min_A = nilai terendah (minimal) data pada atribut A

\max_A = nilai tertinggi (maksimal) data pada atribut A

$\text{new}_{\min A}$ = Nilai minimum terbaru yang diinginkan

$\text{new}_{\max A}$ = Nilai maksimum terbaru yang diinginkan

2. Tahapan Proses K-Nearest Neighbor (KNN)

Tahapan proses algoritma *K-Nearest Neighbor* (KNN) diawali dari menentukan parameter nilai k yang akan digunakan. Kemudian menghitung jarak antar data yang akan diuji dengan semua data latih menggunakan perhitungan jarak euclidean. Hasil dari perhitungan jarak *euclidean* tersebut diurutkan dari jarak terkecil ke jarak terbesar dan menentukan jarak terdekat sesuai dengan k yang digunakan. Dari jarak terdekat sesuai dengan k yang digunakan dipasangkan kelasnya kemudian kelas dari tertangga yang terdekat dicari jumlahnya dan ditetapkan kelas tersebut sebagai kelas data uji.

2.5 Modified K-Nearest Neighbor (MKNN)

Pada sub bab ini akan dijelaskan mengenai definisi dari metode *Modified K-Nearest Neighbor* dan proses dari metode *Modified K-Nearest Neighbor*.

2.5.1 Definisi Modified K-Nearest Neighbor (MKNN)

Algoritma Modified K-Nearest Neighbor (MKNN) adalah suatu algoritma yang menempatkan label kelas data sesuai dengan k divalidasi poin data yang sudah ditetapkan dengan perhitungan K-Nearest Neighbor (KNN). Dengan kata lain, proses MKNN pertama dengan melakukan perhitungan validitas data pada semua data latih. Selanjutnya, dilakukan perhitungan untuk mencari weight voting pada semua data uji menggunakan validitas data (Parvin, et al., 2010).

2.5.2 Proses Modified K-Nearest Neighbor (MKNN)

Secara garis besar terdapat dua proses utama dalam algoritma MKNN, yaitu (Parvin, et al., 2010) :

1. Validitas data *training*

Validitas dari setiap data dihitung berdasarkan pada tetangganya. Validitas data dilakukan hanya sekali pada semua data latih. Setelah dilakukan validasi data selanjutnya data tersebut digunakan sebagai informasi tambahan (Parvin, et al., 2010).

Untuk menghitung validitas data pada data *training*, tetangga terdekatnya perlu dipertimbangkan. Diantara tetangga terdekat dengan data, validitas digunakan untuk menghitung jumlah titik dengan label yang sama untuk data tersebut. Persamaan yang digunakan untuk menghitung validitas data *training* seperti pada persamaan (2-3) (Parvin, et al., 2010).

$$\text{validitas}(x) = \frac{1}{k} \sum_{i=1}^k S(\text{label}(x), (\text{label}(N_i(x)))) \quad (2-3)$$

Dimana :

K : jumlah titik terdekat

Label (x) : kelas x

Label $N_i(x)$: label kelas titik terdekat x

Fungsi S digunakan untuk menghitung kesamaan antara titik x dan data ke-l dari tetangga terdekat. Fungsi S dijelaskan pada persamaan (2-4).

$$S(a, b) = \begin{cases} 1 & a = b \\ 0 & a \neq b \end{cases} \quad (2-4)$$

Dimana :

a : kelas a pada data *training*

b : kelas lain selain a pada data *training*

S akan bernilai 1, jika label kategori a sama dengan label kategori b. S akan bernilai 0, jika label kategori a tidak sama dengan label kategori b.

2. *Weight voting* KNN

Weight voting KNN adalah salah satu variasi metode KNN yang menggunakan k tetangga terdekat, terlepas dari kelas data, akan tetapi menggunakan *weight voting* dari masing-masing data pada data *training*. Masing-masing data diberikan *weight voting* yang biasanya sama dengan beberapa penurunan fungsi jarak dari data yang tidak diketahui. Persamaan *weight voting* dijelaskan pada persamaan (2-5) sebagai berikut (Parvin, et al., 2010):

$$W_{(i)} = \frac{1}{d+\alpha} \quad (2-5)$$

Dimana d adalah jarak dan α merupakan nilai *regulator smoothing*. Dalam penelitian ini nilai α yang digunakan adalah 0,5. Kemudian, *weight voting* ini

dijumlahkan setiap kelasnya dan kelas dengan jumlah terbesar yang akan dipilih menjadi sebuah keputusan. Selanjutnya, validitas dari tiap data pada data *training* dikalikan dengan weight berdasarkan pada jarak *Euclidean*. Dalam metode MKNN ini *weight voting* tiap tetangga dihitung menggunakan persamaan (2-6) (Parvin, et al., 2010).

$$W_{(i)} = \text{validitas}(i) \times \frac{1}{d_e + \alpha} \quad (2-6)$$

Dimana :

$W_{(i)}$: perhitungan *weight voting*

Validitas (i): Nilai validitas

d_e : Jarak *Euclidean*

α : Nilai *regulator smoothing* (pemulusan)

Teknik *weight voting* ini mempunyai pengaruh yang lebih penting terhadap data yang mempunyai nilai validitas lebih tinggi dan paling dekat dengan data. Selain itu, dengan mengalikan validitas data dengan jarak dapat mengatasi kelemahan dari setiap data yang mempunyai jarak dengan weight yang memiliki banyak masalah dalam outlier. Jadi, algoritma MKNN direkomendasikan secara signifikan lebih bagus daripada metode KNN saja yang didasarkan hanya pada jarak saja (Parvin, et al., 2010).

2.6 Tanaman tomat

Menurut ilmu tumbuh-tumbuhan (botani), tomat diklasifikasikan ke dalam taksonomi sebagai berikut (Wiryanta, 2002).

Kingdom : *Plantae*

Divisi : *Spermatophyta*

Subdivisi : *Angiospermae*

Kelas : *Dicotylodena*

Ordo : *Tubiflorae*

Famili : *Solanaceae*

Genus : *Lycopersicum*

Spesies : *Lycopersicum esculentum* Mill./Syn; *Licopersicon lycopersicum* Mill.

Tanaman tomat merupakan tanaman perdu semusim, berbatang lemah dan basah. Tomat umumnya dibudidayakan pada lahan kering atau pada lahan sawah. Tanaman ini tidak membutuhkan persyaratan khusus dalam budidayanya, namun tanaman tomat membutuhkan tanah yang gembur (Hanum, 2008).

Tanaman tomat terdiri dari akar, batang, daun, bunga dan biji. Tinggi tanaman tomat mencapai 2-3 meter. Sewaktu masih muda batang tanaman tomat

berbentuk bulat dan memiliki tekstur yang lunak, tetapi setelah tua batangnya berubah menjadi bersudut dan memiliki tekstur keras berkayu. Ciri khas dari batang tanaman tomat yaitu tumbuhnya bulu-bulu halus di seluruh permukaannya. Akar tanaman tomat berbentuk serabut yang menyebar ke segala arah. Kemampuan akar menembus lapisan tanah terbatas, yakni hanya pada kedalaman 30 – 70 cm (Wiryanta, 2002).

2.7 Penyakit tanaman tomat

Adapun jenis penyakit yang sering menyerang tanaman tomat yaitu sebagai berikut (Semangun, 1989):

2.7.1 Busuk daun

Penyakit busuk daun disebabkan oleh jamur *Phytophthora infestans* (Mont.) d By seperti yang menyebabkan penyakit busuk daun pada tanaman kentang. Gejala tanaman tomat yang terinfeksi penyakit busuk daun ini yaitu pada sisi bawah daun terdapat jamur berwarna putih dan kemudian pada daun terdapat bercak hitam kecoklatan atau keunguan. Pada bagian disekitar bercak daun berwarna kuning pucat yang beralih ke hijau biasa. Selain itu, pada buah tomat yang berwarna hijau muncul bercak berwarna coklat tua, agak keras dan berkerut. Bercak mempunyai batas yang cukup tegas dan batas ini tetap berwarna hijau pada bagian buah yang tidak sakit (Semangun, 1989). Penyakit Busuk Daun ditunjukkan pada Gambar 2.4.

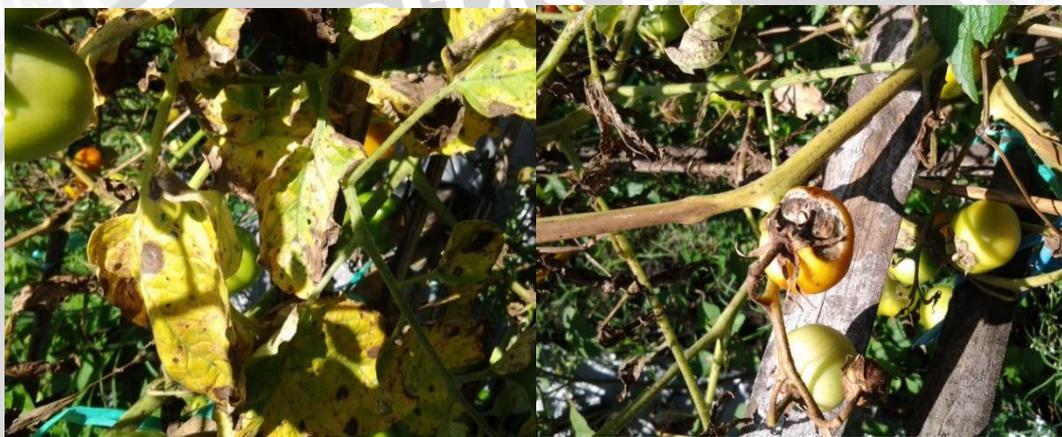


Gambar 2.1 Penyakit Busuk Daun

Pengendalian penyakit Busuk Daun pada tomat dapat dipakai Difolatan (kaptafol), Tomafol 80 WP (kaptafol), atau Dithane M-45 (mankozeb) dengan kadar 0,25-0,3 %. Cairan yang diperlukan adalah 600 l/ha. Untuk meningkatkan efektivitasnya, fungisida dapat ditambah pelekat, terutama pada musim hujan. Penyemprotan dilakukan 2 minggu sekali. Penyemprotan dimulai setelah tanam berumur 2 minggu, atau setelah intensitas penyakit mencapai 25% (Semangun, 1989).

2.7.2 Bercak coklat

Penyakit bercak coklat atau bercak kering merupakan penyakit daun yang umum dan tersebar luas di berbagai Negara penanam tomat. Penyakit bercak coklat ini disebabkan oleh jamur *Alternaria solani* Sor. Gejala penyakit bercak coklat ini yaitu pada daun yang terinfeksi terdapat bercak coklat dengan garis-garis melingkar berwarna lebih gelap. Selain itu, buah pada tanaman yang terinfeksi terdapat bercak coklat gelap atau hitam yang tampak mengendap atau berlekuk dan cabang mudah patah jika buah membesar. Meskipun bercak sangat terbatas, tampak bahwa penyakit mempunyai pengaruh fisiologi di luar bercak. Jika pada daun terdapat banyak bercak, daun akan cepat menjadi tua, layu atau gugur sebelum waktunya (Semangun, 1989). Penyakit Bercak Coklat ditunjukkan pada Gambar 2.5.



Gambar 2.2 Penyakit Bercak Coklat

Pengendalian penyakit Bercak Coklat yaitu agar menjadi lebih tahan, tanaman harus diberi pupuk yang seimbang. Untuk mencegah terbawanya jamur oleh biji dapat dilakukan disinfektasi biji. Agar tidak terjadi banyak infeksi pada bibit, pembibitan jangan terlalu lembab atau rapat. Pada penyakit Bercak Coklat ini lebih baik digunakan fungisida karbamat dan tidak terpengaruh oleh penyemprotan Difolatan (Semangun, 1989).

2.7.3 Layu fusarium

Di Indonesia penyakit layu sudah lama dikenal. Tetapi pada umumnya orang menduga bahwa penyakit ini hanya satu macam yaitu penyakit layu yang disebabkan oleh bakteri, sedangkan penyakit layu ini juga ada yang disebabkan oleh fusarium. Penyakit layu fusarium disebabkan oleh jamur *Fusarium oxysporum* (Schlecht.) f.sp. *lycopersici* (Sacc.) Snyder. Et Hans. Gejala tanaman yang terinfeksi oleh penyakit layu fusarium yaitu menjadi pucatnya tulang-tulang daun terutama daun-daun sebelah atas kemudian diikuti dengan merunduknya tangkai dan akhirnya tanaman menjadi layu secara keseluruhan. Selain itu, Tanaman tomat yang terinfeksi layu fusarium menjadi kerdil, buah sedikit dan kecil. Kadang-kadang kelayuan didahului dengan menguningnya daun, terutama daun-daun sebelah bawah. Jika tanaman yang terinfeksi itu dipotong dekat pangkal batang

atau dikelupas dengan kuku atau pisau akan terlihat suatu cincin coklat dari berkas pembuluh (Semangun, 1989). Penyakit Layu Fusarium ditunjukkan pada Gambar 2.6.



Gambar 2.3 Penyakit Layu Fusarium

Pengendalian penyakit Layu Fusarium dapat dilakukan dengan berbagai macam cara, salah satu caranya dengan pengendalian fungisida. Pengendalian dengan fungisida ini pencelupan akar dengan benomyl 1.000 ppm. Usaha lain untuk mengendalikan penyakit ini yaitu dengan meningkatkan suhu tanah menggunakan mulsa plastic, namun masih memerlukan banyak penelitian untuk dapat dilakukan dalam prakteknya (Semangun, 1989).

2.7.4 Layu bakteri

Penyakit layu bakteri tersebar secara luas di daerah tropika. Oleh karena itu penyakit ini mendapat banyak perhatian dari *Asian Vegetable Research and Development Center (AVRDC)* yang terletak di dekat Tainan, Taiwan. Penyebab penyakit ini yaitu bakteri *Pseudomonas solanacearum* (E.F. Sm.) E.F. Sm. Gejala awal tanaman yang terinfeksi penyakit layu bakteri yaitu layunya beberapa daun muda atau menguningnya daun-daun tua (daun-daun sebelah bawah). Selain itu, tanaman yang terinfeksi menjadi kerdil, tangkai merunduk, buah sedikit dan kecil. Jika pangkal batang dibelah terlihat warna pembuluh yang kecoklatan. Pada stadium penyakit yang lanjut, jika pangkal batang dibelah mengeluarkan lender berwarna putih susu (Semangun, 1989). Penyakit Layu Bakteri dapat dilihat pada Gambar 2.7.



Gambar 2.4 Penyakit Layu Bakteri

Pengendalian penyakit Layu Bakteri dapat dilakukan dengan beberapa cara, yaitu (Semangun, 1989) :

1. Jika tanah terinfeksi berat oleh bakteri ini sebaiknya diadakan pergiliran tanaman dengan menanam tanaman yang tidak dapat menjadi inang *P. solanacearum*. Misalnya tanaman familia kubis-kubisan (*Cruciferae*).
2. Untuk pertanaman skala kecil di tanah yang terjangkit, tanaman tomat dapat disambung di atas batang bawah yang tahan akan penyakit ini, antara lain takokak (*Solanum torvum Sw.*).
3. Dianjurkan agar para petani berusaha untuk menanam jenis-jenis tomat yang tahan terhadap penyakit ini, antara lain tomat Intan, Ratna, dan AV-15.

2.7.5 Gejala mosaik

Tanaman tomat sangat rentan terhadap penyakit mosaik. Di semua Negara penghasil tomat penyakit ini menimbulkan kerugian besar. Penyakit Gejala mosaik ini terdiri dari dua jenis penyakit yaitu mosaik tembakau dan mosaik ketimun, kedua penyakit ini memiliki kesamaan pada gejalanya sehingga digabung menjadi satu yaitu gejala mosaik. Perbedaannya yaitu mosaik tembakau disebabkan oleh virus *Marmor tabaci holmes* sedangkan mosaik ketimun disebabkan oleh virus *Marmor cucumeris var. vulgare Holmes*. Gejala tanaman yang terinfeksi penyakit ini yaitu pada daun terjadi bercak-bercak hijau muda atau kuning yang tidak teratur sehingga daun menjadi berkerut atau terpuntir. Selain itu, pembentukan buah pada puncak batang terhambat dan buah menjadi lebih kecil dari biasanya (Semangun, 1989). Penyakit Mosaik dapat dilihat pada Gambar 2.8.



Gambar 2.5 Penyakit Mosaik

Pengendalian penyakit Mosaik dapat dilakukan dengan beberapa cara, yaitu (Semangun, 1989) :

1. Tidak merokok selama bekerja di lahan tomat, khususnya pada waktu melakukan pekerjaan di persemaian (pembibitan) dan pada waktu memindah tanaman.
2. Persemaian diperiksa dengan teliti, bibit yang sakit dicabut agar tidak menjadi sumber infeksi. Sekitar persemaian dibersihkan dari gulma, terutama gulma yang dapat menjadi sumber infeksi.
3. Diusahakan pada saat tanaman masih muda tidak terlalu banyak dipegang dan tidak dipegang terlalu keras, misalnya pada saat memanjangkan tanaman dan pada waktu memangkas tanaman tomat.

2.7.6 Penyakit keriting

Pada tanaman tomat sering terdapat penyakit keriting (*leaf curl*) yang juga disebut penyakit “kerupuk” (*crinkle*) yang disebabkan oleh virus *Ruga tabaci Holmes* atau *Nicotiana virus 10 (Storey) Smith*. Gejala tanaman tomat yang terinfeksi penyakit keriting ini yaitu tanaman menjadi terhambat pertumbuhannya, daun-daun berkerut atau terpuntir dan daun-daun mengeriting (menggulung) ke atas. Selain itu, buah pada tanaman yang terinfeksi menjadi lebih sedikit dan bahkan tidak berbuah (Semangun, 1989). Penyakit Keriting ditunjukkan pada Gambar 2.9.



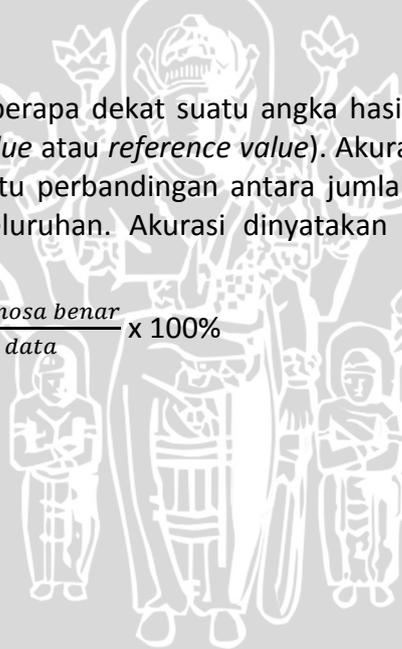
Gambar 2.6 Penyakit Keriting

Pengendalian penyakit Keriting dapat dilakukan dengan cara mengusahakan agar di dekat persemaian tomat tidak terdapat gulma atau tanaman lain yang dapat menjadi sumber virus (Semangun, 1989).

2.8 Akurasi sistem

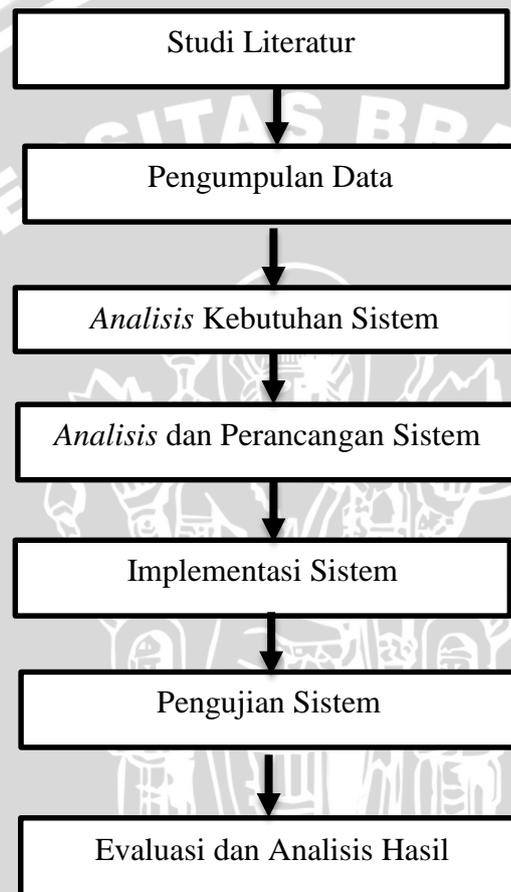
Akurasi merupakan seberapa dekat suatu angka hasil pengukuran terhadap angka sebenarnya (*true value* atau *reference value*). Akurasi dapat diperoleh dari persentase kebenaran, yaitu perbandingan antara jumlah Diagnosis yang tepat dengan jumlah data keseluruhan. Akurasi dinyatakan dalam persamaan 2-7 (Sulistiyandari, 2012).

$$\text{akurasi} = \frac{\text{jumlah diagnosa benar}}{\text{jumlah data}} \times 100\% \quad (2-7)$$



BAB 3 METODOLOGI

Bab ini menjelaskan tentang langkah-langkah yang akan dilakukan dalam penelitian. Penelitian dilakukan dengan melakukan beberapa tahapan yaitu studi literatur, pengumpulan data, *Analisis* kebutuhan sistem, *Analisis* dan perancangan sistem, implementasi sistem, pengujian sistem serta evaluasi dan analisis hasil dari sistem yang akan dibuat. Berikut ini merupakan langkah-langkah pengerjaan yang diilustrasikan dalam diagram blok metode penelitian pada Gambar 3.1.



Gambar 3.1 Diagram Blok Metodologi Penelitian

3.1 Studi literatur

Tahapan ini dilakukan untuk mendapatkan dasar-dasar teori dan sumber acuan untuk pembuatan sistem Diagnosis penyakit pada tanaman tomat menggunakan metode *Modified K-Nearest Neighbor*, diantaranya sebagai berikut :

1. Teori Metode *Modified K-Nearest Neighbor*
2. Berbagai jenis penyakit pada tanaman tomat dan pengendaliannya

Literatur-literatur tersebut diperoleh dari buku, jurnal, *e-book*, penelitian sebelumnya dan situs-situs ilmiah. Cara ini dilakukan untuk mendapatkan dasar-dasar referensi yang kuat bagi peneliti guna menyelesaikan penelitian.

3.2 Pengumpulan data

Lokasi penelitian ini di Balai Pengkajian Teknologi Pertanian (BPTP) Jawa Timur yang terletak di Karangploso, Malang, Jawa Timur. Penelitian ini menggunakan data latih yang digunakan sebagai acuan untuk pengembangan sistem. Data training yang digunakan terdiri dari 15 jenis gejala dari 6 jenis penyakit tanaman tomat. Data-data tersebut digunakan untuk menghitung tingkat keberhasilan dari sistem yang akan dibangun menggunakan metode *Modified K-Nearest Neighbor*.

Cara pengumpulan data untuk penelitian ini terdapat dua jenis data yaitu data primer dan data sekunder. Data primer adalah data yang didapatkan langsung dari responden penelitian. Sedangkan, data sekunder adalah data yang telah dikumpulkan oleh orang lain dan tidak dipersiapkan untuk kegiatan penelitian tetapi dapat digunakan untuk tujuan penelitian seperti melalui buku literatur atau yang lainnya. Metode pengumpulan data primer yang bersifat kuantitatif dapat menggunakan instrument kuisioner dan wawancara.

Tabel 3.1 Penentuan Kebutuhan Data Penelitian

No	Kebutuhan Data	Sumber Data	Metode	Kegunaan Data
1.	Data Mengenai penyakit tanaman tomat	Pakar Hama dan Penyakit	Wawancara	Sebagai data pengetahuan mengenai jenis penyakit tanaman tomat.
2.	Data kasus tanaman tomat yang terkena penyakit.	Balai Pengkajian Teknologi Pertanian (BPTP) Jawa Timur	Observasi	Data yang didapat akan digunakan untuk proses perhitungan menggunakan metode <i>Modified K-Nearest Neighbor</i> .
3.	Pengujian kasus secara manual untuk mendeteksi jenis penyakit yang menyerang pada tanaman tomat.	Data kasus tanaman tomat yang terkena penyakit dari pakar.	Kuisioner	Pengujian proses untuk menentukan jenis penyakit apa yang menyerang tanaman tomat.

3.3 Pengumpulan data

Analisis kebutuhan merupakan sebuah fase yang berfungsi untuk merumuskan kebutuhan-kebutuhan yang diperlukan dalam mengembangkan suatu sistem. Analisis kebutuhan harus sesuai dengan lokasi penelitian, variabel penelitian dan mempersiapkan kebutuhan penelitian.

Secara keseluruhan, kebutuhan yang digunakan dalam pembuatan sistem diagnosa penyakit tanaman tomat ini meliputi:

1. Spesifikasi kebutuhan *hardware*, meliputi :
 - Laptop
 - *Memory* 2GB
2. Spesifikasi kebutuhan *software*, meliputi :
 - Windows 8.1 64bit sebagai sistem operasi.
 - XAMPP sebagai *server localhost*, MySQL termasuk didalamnya sebagai *database management system* (DBMS).
3. Data yang dibutuhkan, meliputi :
 - Data gejala penyakit tanaman tomat
 - Data nilai pembobotan gejala penyakit tanaman tomat

3.4 Analisis dan perancangan sistem

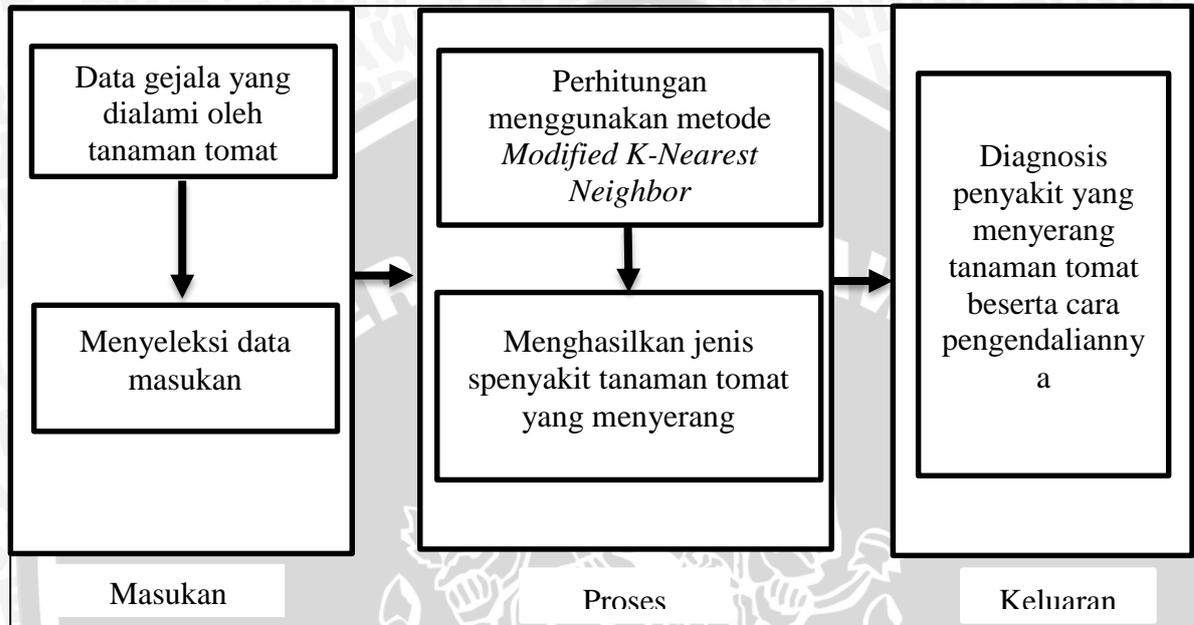
Analisis dan perancangan sistem membahas mengenai hal-hal yang terkait dengan apa saja yang diperlukan dalam pembuatan sistem. Hal-hal tersebut meliputi deskripsi sistem, perancangan program aplikasi, perhitungan manual, desain antarmuka dan perancangan pengujian. Hasil yang diharapkan pada perancangan sistem yaitu dapat membangun sebuah sistem berdasarkan hasil dan analisis yang ada. Secara umum, sistem yang dibangun adalah sebuah sistem untuk mendiagnosis penyakit tanaman tomat menggunakan metode *Modified K-Nearest Neighbor* (MKNN).

Pada tahap analisis, pertama menyiapkan data penyakit tanaman tomat yang telah berupa angka untuk digunakan dalam proses diagnosis penyakit. Data yang digunakan sebanyak 288 data. Kemudian membuat model metode yang akan digunakan dalam sebuah diagram alir agar permasalahan mudah untuk dipecahkan dan dilakukan perancangan sistem sesuai dengan analisis yang telah dilakukan.

Tahap perancangan sistem dilakukan untuk memecahkan permasalahan diagnosis penyakit tanaman tomat meliputi deskripsi umum sistem, perancangan model MKNN, perancangan proses MKNN dengan diagram alir, contoh perhitungan manual, perancangan antar muka dan perancangan pengujian sistem yang disajikan dalam bentuk tabel pengujian.

3.4.1 Diagram blok sistem

Diagram blok sistem merupakan diagram yang berbentuk blok-blok yang menggambarkan aliran proses dari komponen-komponen sistem yang memuat fungsi matematis. Diagram blok sistem menjelaskan cara kerja sistem yang dimulai dari masukan sampai keluaran yang dihasilkan. Diagram blok sistem yang akan digunakan dapat dilihat pada Gambar 3.2.



Gambar 3.2 Diagram Blok Proses Diagnosis Penyakit Tanaman Tomat Dengan Metode MKNN

Proses yang terjadi dalam Gambar 3.3 yaitu:

1. Masukan

Inputan atau masukan dari sistem ini yaitu berupa data gejala yang dialami oleh tanaman tomat dengan menjawab 15 pertanyaan mengenai gejala penyakit yang diberikan oleh sistem. Data tersebut kemudian diklasifikasikan sesuai dengan jenis penyakit yang menyerang tanaman tomat yang paling berpeluang besar untuk proses selanjutnya.

2. Proses

Proses perhitungan pada sistem ini menggunakan metode *Modified K-Nearest Neighbor*. Langkah-langkah perhitungan dengan menggunakan metode MKNN ini antara lain:

- Melakukan pengambilan nilai gejala dari jawaban yang diinputkan oleh pengguna
- Menentukan nilai k tetangga terdekat
- Menghitung validitas data latih
- Menghitung jarak *Euclidean*

- Menghitung *weighted voting* (pembobotan)
 - Menentukan kelas dari data uji tersebut
3. Keluaran

Hasil atau keluaran dari sistem ini adalah jenis penyakit yang menyerang tanaman tomat beserta cara pengendaliannya.

3.5 Implementasi

Implementasi perangkat lunak yang menerapkan algoritma *Modifed K-Nearest Neighbor* dilakukan dengan mengacu pada perancangan sistem yang dibuat. Implementasi sistem tersebut meliputi :

1. Implementasi *Interface*, menggunakan *Software Visual Studio C#*.
2. Implementasi basis data, dengan menggunakan sebuah DBMS MySQL pada *server localhost (XAMPP)* yang bertujuan untuk memudahkan manipulasi dan penyimpanan data.
3. Implementasi algoritma, melakukan perhitungan dengan metode *Modified K-Nearest Neighbor* ke dalam bahasa pemrograman C# dan menggunakan *software Visual Studio C#*.
4. Implementasi ini akan menghasilkan Diagnosis penyakit yang menyerang tanaman tomat melalui inputan yang berupa gejala penyakit.

3.6 Pengujian sistem

Pengujian sistem ini dilakukan agar dapat memastikan bahwa aplikasi yang telah dibangun dapat bekerja sesuai dengan tujuan yang diharapkan. Pengujian yang dilakukan ada 3 macam, yaitu:

1. Pengujian pengaruh nilai k
Pengujian ini dilakukan untuk mengetahui apakah nilai k berpengaruh terhadap akurasi yang dihasilkan sistem. Selain itu pengujian ini bertujuan untuk mencari nilai k terbaik dari sistem.
2. Pengujian pengaruh jumlah data latih
Pengujian pengaruh jumlah data latih ini dilakukan untuk mengetahui apakah jumlah data latih berpengaruh terhadap hasil akurasi yang dihasilkan sistem.
3. Pengujian pengaruh data latih seimbang dan data latih tidak seimbang
Pengujian ini dilakukan untuk melihat apakah jenis data latih seimbang dan tidak seimbang berpengaruh terhadap akurasi yang dihasilkan.

3.7 Evaluasi dan analisis hasil

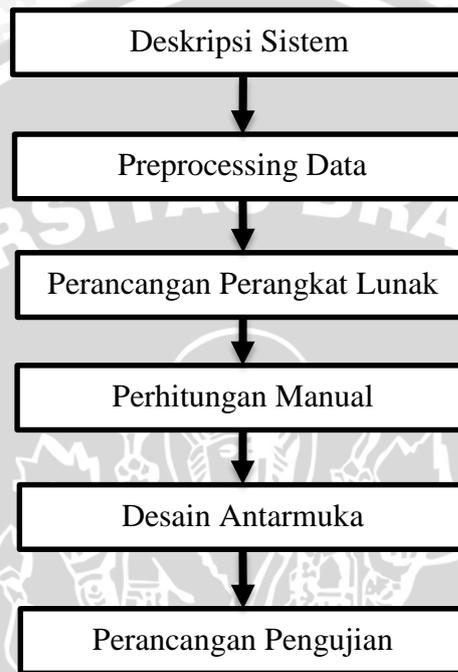
Tahap evaluasi dilakukan setelah proses pengujian sistem, evaluasi dilakukan untuk mempertimbangkan tingkat keberhasilan aplikasi dari

implementasi metode dalam menyelesaikan permasalahan Diagnosis penyakit tanaman tomat dan apakah kinerja metode MKNN lebih optimal sesuai dengan tujuan penelitian ini. Tahap akhir dari seluruh rangkaian proses yaitu proses penarikan kesimpulan terhadap hasil yang diperoleh dari proses pengujian dan analisis yang telah dievaluasi. Selanjutnya kesimpulan dan saran dapat digunakan oleh pembaca sebagai acuan untuk perbaikan serta pengembangan aplikasi ke depannya.



BAB 4 PERANCANGAN

Bab ini membahas tentang perancangan pada sistem untuk Diagnosis Penyakit Tanaman Tomat Menggunakan Metode *Modified K-Nearest Neighbor*. proses perancangan sistem terdiri dari deskripsi sistem, *preprocessing* data, perancangan perangkat lunak, perhitungan manual, desain antarmuka dan perancangan pengujian. Diagram alir proses perancangan sistem dapat dilihat pada Gambar 4.1.



Gambar 4.1 Diagram Alir Tahap Perancangan Sistem

4.1 Deskripsi sistem

Sistem yang akan dibuat dalam skripsi ini bertujuan untuk mengimplementasikan metode *Modified K-Nearest Neighbor* dan mengetahui kinerja metode tersebut dalam permasalahan Diagnosis penyakit tanaman tomat. Penyakit yang dapat didiagnosis dalam sistem ada 6 macam yaitu bercak coklat, busuk daun, layu bakteri, *layu fusarium*, gejala mosaik dan penyakit keriting. Input yang digunakan dalam sistem ini berupa 15 pertanyaan gejala penyakit yang menyerang tanaman tomat dan kemudian jawaban dari inputan tersebut akan diklasifikasikan menggunakan metode MKNN.

Pada sistem yang akan dibuat ini terdapat dua tahap utama yaitu pelatihan data dan pengujian data. Proses pelatihan data dilakukan untuk mendapatkan nilai *validitas* dari data latih yang kemudian digunakan untuk proses pengujian. Sedangkan, pada proses pengujian akan didapatkan hasil diagnosis penyakit menurut sistem dan kemudian hasil tersebut dibandingkan dengan hasil diagnosis pada data asli untuk mendapatkan nilai akurasi.

4.2 Preprocessing data

Preproses data merupakan suatu tahapan yang dilakukan untuk menyeleksi data dari semua alternatif yang telah terkumpul dalam tahap pengumpulan data. Alternatif data yang telah diseleksi kemudian dipisahkan sesuai dengan kriteria-kriteria yang telah ditentukan. Kriteria atau parameter yang digunakan pada data yang akan digunakan terdiri dari 15 gejala penyakit tanaman tomat. Setiap gejala penyakit ini diberikan nilai pembobotan dari pakar untuk mempermudah proses perhitungan pada sistem. Nilai pembobotan untuk setiap gejala berbeda-beda, setiap gejala memiliki tiga opsi nilai yaitu tidak ada, sedikit dan banyak. Nilai pembobotan dari 15 gejala penyakit tanaman tomat ditunjukkan pada Tabel 4.1.

Tabel 4.1 Nilai Pembobotan 15 Gejala Penyakit Tanaman Tomat

No	Kode	Gejala	Opsi	Nilai
1	G1	sisi bawah daun terdapat jamur putih	tidak ada	0
			sedikit	10
			banyak	20
2	G2	pada daun terdapat bercak coklat dengan bagian luar berwarna kuning ke hijau	tidak ada	0
			sedikit	45
			banyak	90
3	G3	daun berwarna belang antara hijau kekuningan dan berkerut/terpuntir	tidak ada	0
			sedikit	45
			banyak	90
4	G4	pada daun terdapat bercak coklat gelap, keras, dan melingkar sepusat	tidak ada	0
			sedikit	45
			banyak	90
5	G5	daun menguning, layu, dan gugur	tidak ada	0
			sedikit	10
			banyak	20
6	G6	daun mengeriting, dan menggulung ke atas	tidak ada	0
			sedikit	45
			banyak	90
7	G7	pada batang terdapat bercak coklat lonjong dan kosentris	tidak ada	0
			sedikit	10
			banyak	20

No	Kode	Gejala	Opsi	Nilai
8	G8	pada batang terdapat bercak kuning atau coklat	tidak ada	0
			sedikit	10
			banyak	20
9	G9	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
			sedikit	10
			banyak	20
14	G14	tanaman layu	tidak ada	0
			sedikit	10
			banyak	20
15	G15	tanaman kerdil	tidak ada	0
			sedikit	10
			banyak	20

Keterangan nilai bobot:

0 : tidak ada gejala

10 : gejala sedikit namun bukan gejala utama

20 : gejala banyak namun bukan gejala utama

45 : gejala sedikit dan merupakan gejala utama

90 : gejala banyak dan merupakan gejala utama

Lima belas gejala penyakit tomat tersebut mewakili dari 6 penyakit utama yang akan diDiagnosis pada sistem menggunakan metode MKNN. Pasangan gejala penyakit dan penyakit yang menyerang tanaman tomat dapat dilihat pada Tabel 4.2.

Tabel 4.2 Pasangan gejala penyakit dan penyakit yang menyerang tanaman tomat

G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	diagnosis
x	x						x			x					Busuk Daun
			x	x		x					x				Bercak Coklat
				x				x				x	x		Layu Fusarium
				x					x			x	x		Layu Bakteri
		x					x					x		x	Mosaik
					x							x		x	Keriting

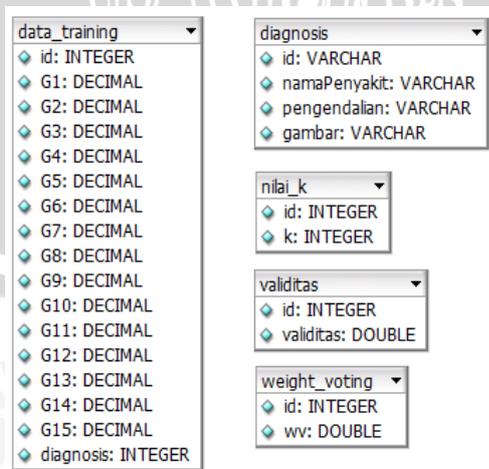
Keterangan: Gejala utama penyakit

4.3 Perancangan perangkat lunak

Perancangan perangkat lunak ini menjelaskan mengenai hubungan antar komponen-komponen sehingga mampu membentuk sebuah fungsi yang mampu memberikan pelayanan terhadap kebutuhan pengguna. Perancangan perangkat lunak pada aplikasi untuk Diagnosis penyakit tanaman tomat menggunakan metode MKNN terdiri dari perancangan database dan perancangan proses.

4.3.1 Perancangan *database*

Perancangan perangkat aplikasi Diagnosis penyakit tanaman tomat berupa basis data yang mengandung data pendukung dalam pembuatan aplikasi dan dapat digunakan dalam berbagai situasi kondisi serta dapat diatur oleh perangkat lunak yang disebut *Database Management Sistem* (DBMS) dan menggunakan DBMS MySQL. Rancangan database sistem dapat dilihat pada Gambar 4.2.



Gambar 4.2 Rancangan Database Sistem

Pada perancangan basis data aplikasi ini menggunakan lima tabel yaitu tabel data_training, tabel diagnosis, tabel nilai_k, tabel validitas dan tabel weight_voting. Entitas dalam basis data pada aplikasi ini adalah sebagai berikut:

1. Tabel data_training

Tabel data_training ini digunakan untuk menyimpan data latihan yang akan digunakan dalam proses perhitungan sistem. Tabel data_training dapat dilihat pada Tabel 4.3.

Tabel 4.3 Struktur Tabel data_training

No	Nama Field	Tipe	Lebar	Keterangan
1	Id	INTEGER	5	Kode penyakit
2	G1	DECIMAL	10,5	Nilai dari tiap gejala
3	G2	DECIMAL	10,5	Nilai dari tiap gejala
4	G3	DECIMAL	10,5	Nilai dari tiap gejala
5	G4	DECIMAL	10,5	Nilai dari tiap gejala
6	G5	DECIMAL	10,5	Nilai dari tiap gejala
7	G6	DECIMAL	10,5	Nilai dari tiap gejala
8	G7	DECIMAL	10,5	Nilai dari tiap gejala
9	G8	DECIMAL	10,5	Nilai dari tiap gejala
10	G9	DECIMAL	10,5	Nilai dari tiap gejala
11	G10	DECIMAL	10,5	Nilai dari tiap gejala
12	G11	DECIMAL	10,5	Nilai dari tiap gejala
13	G12	DECIMAL	10,5	Nilai dari tiap gejala
14	G13	DECIMAL	10,5	Nilai dari tiap gejala
15	G14	DECIMAL	10,5	Nilai dari tiap gejala
16	G15	DECIMAL	10,5	Nilai dari tiap gejala
17	diagnosis	INTEGER	3	Jenis penyakit

2. Tabel diagnosis

Tabel diagnosis ini digunakan untuk menyimpan nama penyakit dan cara pengendalian tiap penyakit yang menyerang tanaman tomat. Tabel diagnosis dapat dilihat pada Tabel 4.4.

Tabel 4.4 Struktur Tabel diagnosis

No	Nama Field	Tipe	Lebar	Keterangan
1	Id	VARCHAR	15	Kode penyakit
2	namaPenyakit	VARCHAR	50	Nama penyakit
3	Pengendalian	VARCHAR	250	Pengendalian penyakit
4	Gambar	VARCHAR	100	Gambar penyakit

3. Tabel nilai_k

Tabel nilai_k ini digunakan untuk menyimpan nilai k yang digunakan untuk proses perhitungan. Tabel nilai_k dapat dilihat pada Tabel 4.5.

Tabel 4.5. Struktur Tabel nilai_k

No	Nama Field	Tipe	Lebar	Keterangan
1	Id	INTEGER	11	Kode nilai k
2	K	INTEGER	11	Nilai k

4. Tabel validitas

Tabel validitas ini digunakan untuk menyimpan nilai validitas yang telah dihitung kemudian nilai validitas ini digunakan untuk proses perhitungan selanjutnya. Tabel validitas dapat dilihat pada Tabel 4.6.

Tabel 4.6 Struktur Tabel validitas

No	Nama Field	Tipe	Lebar	Keterangan
1	Id	INTEGER	11	Kode validitas
2	validitas	DOUBLE	-	Nilai validitas

5. Tabel weight_voting

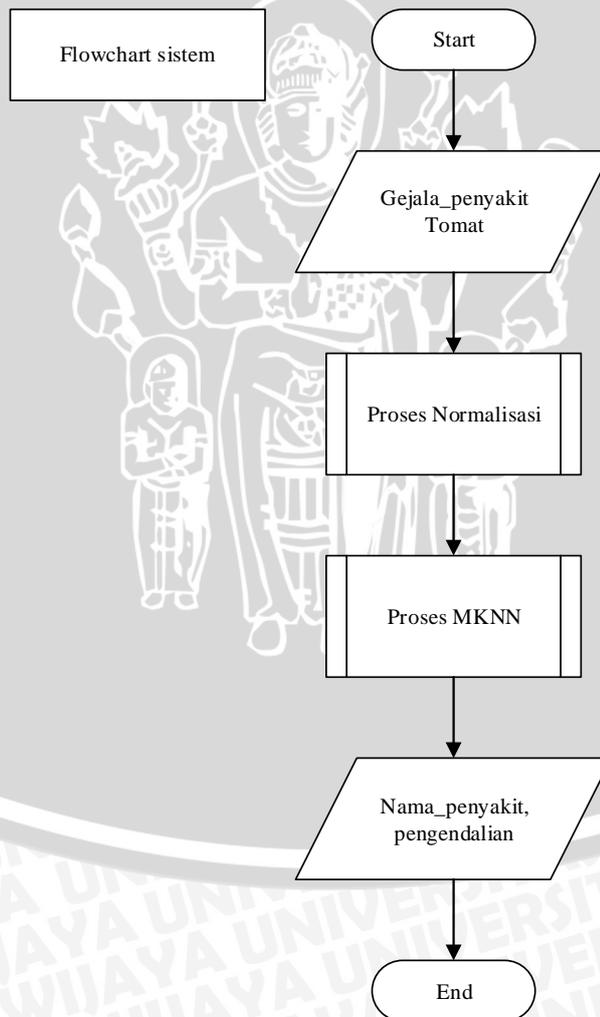
Tabel weight_voting ini digunakan untuk menyimpan setiap nilai weight voting dari hasil perhitungan weight voting yang digunakan untuk diagnosis penyakit tanaman tomat. Tabel weight_voting dapat dilihat pada Tabel 4.7.

Tabel 4.7 Struktur Tabel weight_voting

No	Nama Field	Tipe	Lebar	Keterangan
1	Id	INTEGER	15	Kode weight voting
2	wv	DOUBLE	-	Nilai weight voting

4.3.2 Perancangan proses

Proses Diagnosis penyakit tanaman tomat dilakukan dengan memasukkan data gejala-gejala penyakit tanaman tomat yang sudah tersimpan pada sistem. Proses memasukkan data gejala-gejala penyakit tersebut dilakukan dengan cara menjawab pertanyaan yang telah disediakan pada *interface* menu awal. Setelah selesai memasukkan gejala dan menekan tombol diagnosis, maka dapat dilakukan proses diagnosis perhitungan dengan menggunakan metode *Modified K-Nearest Neighbor* dan sistem akan menunjukkan hasil diagnosis berupa nama penyakit beserta pengendaliannya. Sedangkan pada menu data latih, pengguna memasukkan jumlah data latih dan nilai *k* yang akan digunakan untuk perhitungan. Setelah itu, pengguna dapat melihat hasil perhitungan dari nilai *validitas*, nilai *weight voting* dan hasil klasifikasi. Pada halaman pengujian pengguna memasukkan data uji yang akan diklasifikasikan, kemudian data tersebut akan dihitung dan hasil diagnosis dari sistem akan dicocokkan dengan hasil Diagnosis asli untuk dicari nilai akurasi. Diagram alir sistem dapat dilihat pada Gambar 4.3.

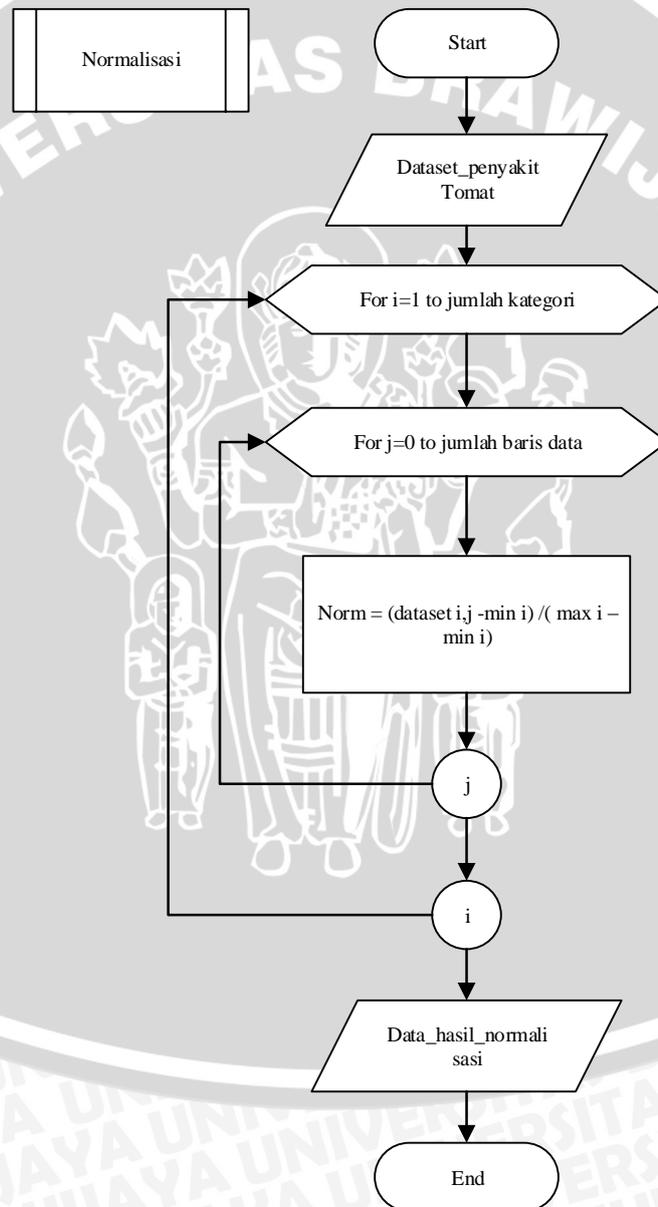


Gambar 4.3 Diagram Alir Sistem

4.3.2.1 Proses Perhitungan Normalisasi Data

Proses normalisasi data ini akan menormalisasikan data latih dan data uji yang akan digunakan dalam proses perhitungan. Hal ini bertujuan agar data berada pada range [0,1] sehingga sebaran data tidak terlalu jauh. Persamaan 2-2 digunakan pada proses normalisasi data latih dan data uji gejala penyakit tanaman tomat. Proses normalisasi data dapat dilihat pada Gambar 4.4.

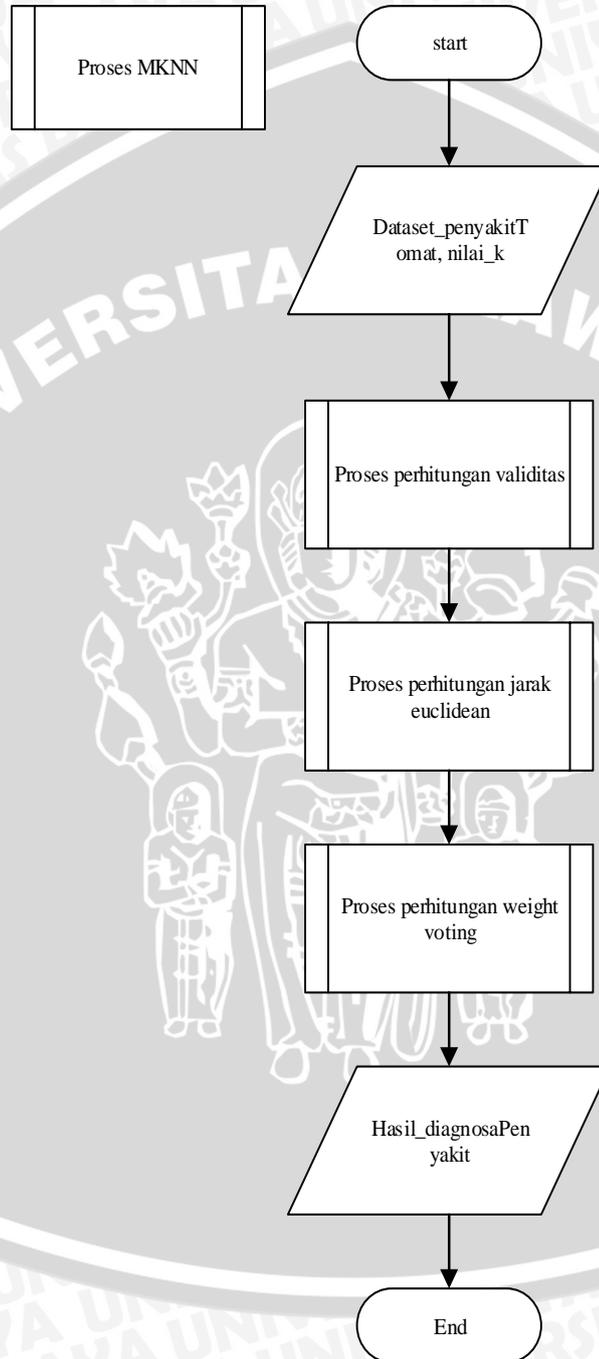
Pada diagram alir Gambar 4.4 proses normalisasi dimulai dengan menginputkan data gejala penyakit tanaman tomat. Kemudian data tersebut dihitung menggunakan persamaan 2-2 dan hasilnya normalisasi ini digunakan untuk proses perhitungan selanjutnya.



Gambar 4.4 Diagram Alir Proses Perhitungan Normalisasi

4.3.2.2 Proses Algoritma *Modified K-Nearest Neighbor*

Proses algoritma *Modified K-Nearest Neighbor* terdiri dari 3 proses utama, yaitu proses perhitungan *validitas*, proses perhitungan jarak *Euclidean*, dan perhitungan *weight voting*. Proses algoritma MKNN dapat dilihat pada Gambar 4.5.



Gambar 4.5 Diagram Alir Proses Algoritma MKNN

Pada algoritma MKNN tahapan awal dimulai dengan memasukkan jumlah data latih yang akan digunakan dan nilai k . selanjutnya, dilakukan proses *validitas*

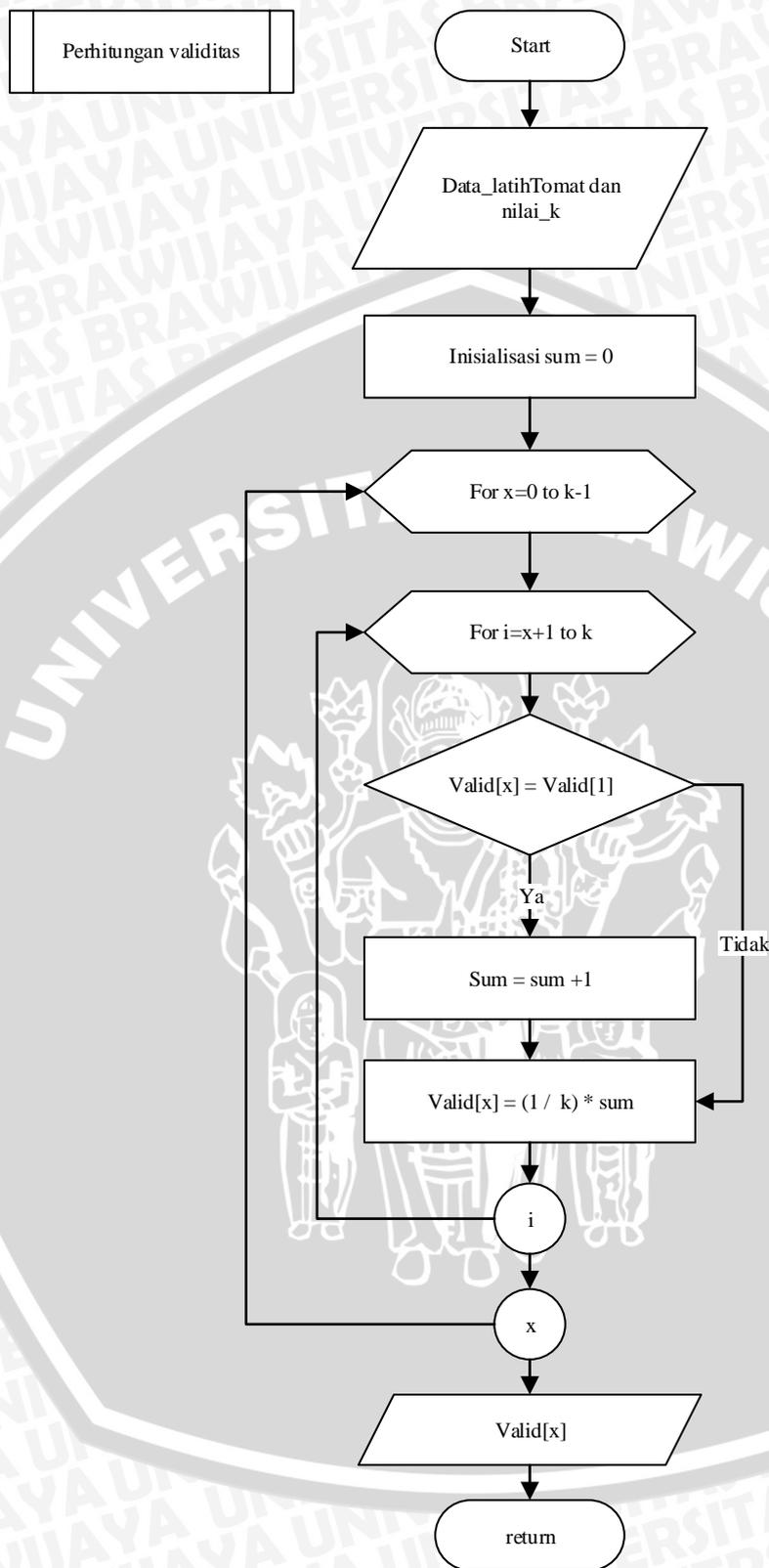
terhadap data latih. Setelah melakukan validitas data latih dilakukan proses perhitungan jarak *Euclidean* terlebih dahulu. Proses selanjutnya yaitu melakukan perhitungan jarak antar data uji dan data latih. Setelah proses tersebut selesai, maka dilakukan proses perhitungan *weight voting*. Tahapan terakhir didapatkan dengan menghitung nilai *weight voting* terbesar sesuai nilai k yang dimasukkan, maka akan dihasilkan *output* hasil diagnosis penyakit tanaman tomat.

4.3.2.3 Proses Perhitungan Validitas

Proses perhitungan validitas ini merupakan suatu proses melakukan validitas untuk tiap-tiap data latih. Validitas data latih ini dilakukan untuk memperoleh informasi lebih mengenai data yang akan digunakan untuk proses *weight voting*. Proses perhitungan validitas dapat dilihat pada Gambar 4.6.

Pada diagram alir perhitungan validitas terdiri dari beberapa tahapan. Tahapan awal yaitu memasukkan jumlah data latih penyakit tanaman tomat dan nilai k . Kemudian, dilakukan perhitungan validitas dengan membandingkan kelas kategori penyakit tanaman tomat pada data latihnya sesuai dengan nilai k yang diinputkan. Perhitungan ini dilakukan berdasarkan dengan persamaan 2-3. Proses yang terdapat pada perhitungan manual validitas data latih ini dapat dilihat pada Tabel 4.14.

Proses validitas yang dilakukan adalah dengan melakukan input data latih penyakit tanaman tomat kemudian membandingkan kelas-kelas pada data latihnya sesuai dengan ketentuan sebelumnya. Berdasarkan persamaan 2-4, pada proses validitas terdapat ketentuan, jika kelasnya sama maka nilainya adalah 1 dan jika kelasnya tidak sama maka nilainya 0 serta dilakukan perbandingan data sebanyak k . Kemudian $\text{valid}[x]$ akan dijumlah dan dibagi sebanyak k data yang telah diinputkan. Maka akan didapatkan output data nilai validitas tiap data uji pada proses klasifikasi algoritma MKNN.



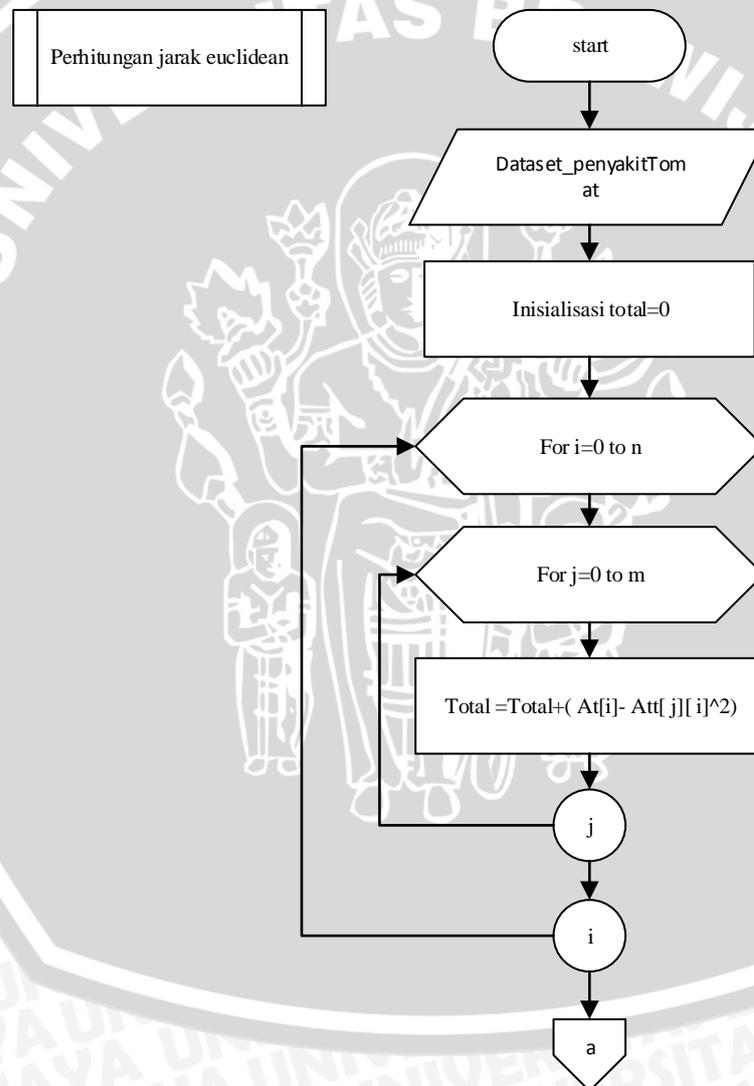
Gambar 4.6 Diagram Alir Proses Perhitungan Validitas

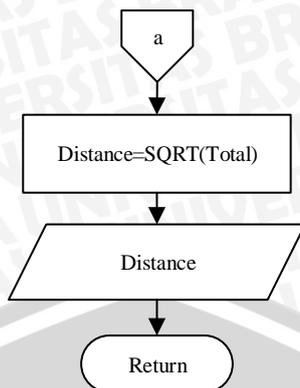


4.3.2.4 Proses Perhitungan Jarak *Euclidean*

Proses perhitungan jarak *Euclidean* ini dilakukan untuk menghitung jarak antar data latih serta jarak antar data latih dan data uji pada penyakit tanaman tomat. Perhitungan jarak tersebut berdasarkan atribut penyakit tanaman tomat. Diagram alir proses perhitungan jarak *Euclidean* dapat dilihat pada Gambar 4.7.

Pada diagram alir proses perhitungan jarak *euclidean* tahapan pertama yang dilakukan adalah melakukan input data gejala penyakit tanaman tomat. Inputan data tersebut akan dilakukan perhitungan *euclidean* sesuai dengan persamaan 2-1 sebanyak data inputan. Setelah itu, akan didapatkan output data nilai *euclidean* setiap data uji pada proses MKNN.

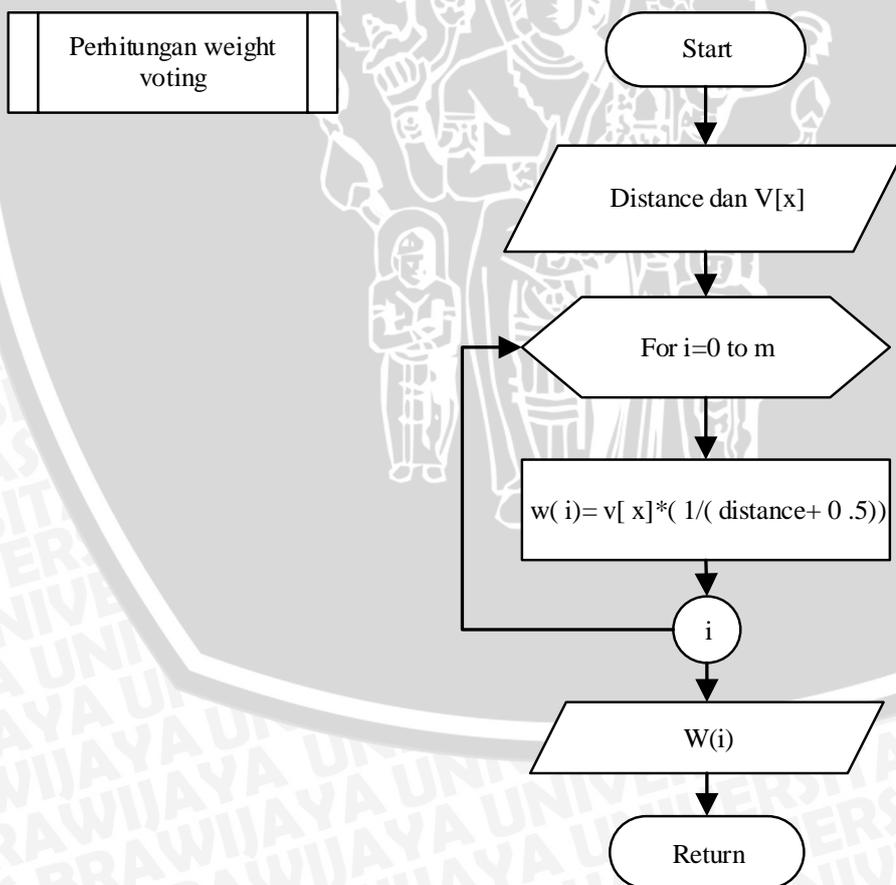




Gambar 4.7 Diagram Alir Proses Perhitungan Jarak Euclidean

4.3.2.5 Perancangan Proses Perhitungan *Weight voting*

Pada proses perhitungan *weight voting* terlebih dahulu memasukkan nilai hasil validitas masing-masing data latih yang diperoleh dari perhitungan jarak *euclidean*. Selanjutnya, akan dilakukan proses perhitungan *weight voting* dengan persamaan 2-6 sebanyak data uji.. Dari proses ini akan dihasilkan output nilai *weight voting* dari tiap data uji pada proses klasifikasi algoritma MKNN. Proses perhitungan *weight voting* dapat dilihat pada Gambar 4.8.



Gambar 4.8 Proses Perhitungan Weight voting

4.4 Perhitungan manual

Perhitungan manual berfungsi untuk memberikan gambaran umum perancangan sistem yang dibangun. Dari metode *Modified K-Nearest Neighbor*(MKNN) langkah-langkah dalam perhitungannya yaitu:

1. Menentukan nilai k atau tetangganya
2. Melakukan normalisasi data
3. Menghitung validitas data latih
4. Menghitung jarak *Euclidean*
5. Menghitung *weight voting* (pembobotan)
6. Menentukan kelas dari data uji

Pada perhitungan manual ini, digunakan 24 dataset dimana dibagi menjadi 2 bagian yaitu 18 data latih dan 6 data uji. Dataset yang digunakan dapat dilihat pada Tabel 4.8 berikut.

Tabel 4.8 Dataset perhitungan manual

No.	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	kelas
1	20	90	0	0	0	0	0	20	0	0	20	0	0	0	0	1
2	10	90	0	0	0	0	0	20	0	0	20	0	0	0	0	1
3	20	45	0	0	0	0	0	20	0	0	10	0	0	0	0	1
4	0	0	0	90	20	0	20	0	0	0	0	20	0	0	0	2
5	0	0	0	90	10	0	10	0	0	0	0	10	0	0	0	2
6	0	0	0	45	20	0	10	0	0	0	0	10	0	0	0	2
7	0	0	0	0	20	0	0	0	90	0	0	0	20	20	0	3
8	0	0	0	0	10	0	0	0	90	0	0	0	10	10	0	3
9	0	0	0	0	20	0	0	0	45	0	0	0	10	20	0	3
10	0	0	0	0	20	0	0	0	0	90	0	0	20	20	0	4
11	0	0	0	0	10	0	0	0	0	90	0	0	20	10	0	4
12	0	0	0	0	20	0	0	0	0	45	0	0	0	0	0	4
13	0	0	90	0	0	0	0	20	0	0	0	0	20	0	20	5
14	0	0	90	0	0	0	0	10	0	0	0	0	10	0	10	5
15	0	0	45	0	0	0	0	20	0	0	0	0	20	0	10	5
16	0	0	0	0	0	90	0	0	0	0	0	0	20	0	20	6
17	0	0	0	0	0	90	0	0	0	0	0	0	10	0	0	6
18	0	0	0	0	0	45	0	0	0	0	0	0	10	0	20	6
41	10	45	0	0	0	0	0	10	0	0	10	0	0	0	0	1
76	0	0	0	90	0	0	10	0	0	0	0	20	0	0	0	2
140	0	0	0	0	20	0	0	0	45	0	0	0	10	10	0	3
187	0	0	0	0	0	0	0	0	0	90	0	0	0	20	0	4
265	0	0	45	0	0	0	0	0	0	0	0	0	10	0	20	5
285	0	0	0	0	0	45	0	0	0	0	0	0	10	0	0	6

Dataset yang digunakan terdiri dari 15 gejala penyakit dan diagnosis dari tiap data. Pembagian data latih dan data uji dapat dilihat pada Tabel 4.9 dan Tabel 4.10 berikut.

Tabel 4.9 Data Latih

No.	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	kelas
1	20	90	0	0	0	0	0	20	0	0	20	0	0	0	0	1
2	10	90	0	0	0	0	0	20	0	0	20	0	0	0	0	1
3	20	45	0	0	0	0	0	20	0	0	10	0	0	0	0	1
4	0	0	0	90	20	0	20	0	0	0	0	20	0	0	0	2
5	0	0	0	90	10	0	10	0	0	0	0	10	0	0	0	2
6	0	0	0	45	20	0	10	0	0	0	0	10	0	0	0	2
7	0	0	0	0	20	0	0	0	90	0	0	0	20	20	0	3
8	0	0	0	0	10	0	0	0	90	0	0	0	10	10	0	3
9	0	0	0	0	20	0	0	0	45	0	0	0	10	20	0	3
10	0	0	0	0	20	0	0	0	0	90	0	0	20	20	0	4
11	0	0	0	0	10	0	0	0	0	90	0	0	20	10	0	4
12	0	0	0	0	20	0	0	0	0	45	0	0	0	0	0	4
13	0	0	90	0	0	0	0	20	0	0	0	0	20	0	20	5
14	0	0	90	0	0	0	0	10	0	0	0	0	10	0	10	5
15	0	0	45	0	0	0	0	20	0	0	0	0	20	0	10	5
16	0	0	0	0	0	90	0	0	0	0	0	0	20	0	20	6
17	0	0	0	0	0	90	0	0	0	0	0	0	10	0	0	6
18	0	0	0	0	0	45	0	0	0	0	0	0	10	0	20	6

Tabel 4.10 Data Uji

No.	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	kelas
41	10	45	0	0	0	0	0	10	0	0	10	0	0	0	0	1
76	0	0	0	90	0	0	10	0	0	0	0	20	0	0	0	2
140	0	0	0	0	20	0	0	0	45	0	0	0	10	10	0	3
187	0	0	0	0	0	0	0	0	0	90	0	0	0	20	0	4
265	0	0	45	0	0	0	0	0	0	0	0	0	10	0	20	5
285	0	0	0	0	0	45	0	0	0	0	0	0	10	0	0	6

Keterangan:

G1: Gejala penyakit ke-1

G2: Gejala penyakit ke-2

G3: Gejala penyakit ke-3

G4: Gejala penyakit ke-4

G5: Gejala penyakit ke-5

G6: Gejala penyakit ke-6

G7 : Gejala penyakit ke-7

G8 : Gejala penyakit ke-8

G9 : Gejala penyakit ke-9

G10: Gejala penyakit ke-10

G11: Gejala penyakit ke-11

G12: Gejala penyakit ke-12

G13: Gejala penyakit ke-13

G14: Gejala penyakit ke-14

G15: Gejala penyakit ke-15

Langkah 1. Menentukan nilai k atau tetangganya.

Pada perhitungan manual ini ditentukan nilai dari k adalah 3

Langkah 2. Melakukan proses normalisasi data.

Pada langkah ini dilakukan proses normalisasi data untuk menyamakan sebaran data latih maupun data uji. Data latih sebelum dinormalisasi dapat dilihat pada Tabel 4.9 dan data uji sebelum dinormalisasi dapat dilihat pada Tabel 4.10. Sedangkan data latih hasil normalisasi dapat dilihat pada Tabel 4.11 dan data uji hasil normalisasi dapat dilihat pada Tabel 4.12.

Tabel 4.11 Data Latih Hasil Normalisasi

no.	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	kelas
1	1	1	0	0	0	0	0	1	0	0	1	0	0	0	0	1
2	0.5	1	0	0	0	0	0	1	0	0	1	0	0	0	0	1
3	1	0.5	0	0	0	0	0	1	0	0	0.5	0	0	0	0	1
4	0	0	0	1	1	0	1	0	0	0	0	1	0	0	0	2
5	0	0	0	1	0.5	0	0.5	0	0	0	0	0.5	0	0	0	2
6	0	0	0	0.5	1	0	0.5	0	0	0	0	0.5	0	0	0	2
7	0	0	0	0	1	0	0	0	1	0	0	0	1	1	0	3
8	0	0	0	0	0.5	0	0	0	1	0	0	0	0.5	0.5	0	3
9	0	0	0	0	1	0	0	0	0.5	0	0	0	0.5	1	0	3
10	0	0	0	0	1	0	0	0	0	1	0	0	1	1	0	4
11	0	0	0	0	0.5	0	0	0	0	1	0	0	1	0.5	0	4
12	0	0	0	0	1	0	0	0	0	0.5	0	0	0	0	0	4
13	0	0	1	0	0	0	0	1	0	0	0	0	1	0	1	5
14	0	0	1	0	0	0	0	0.5	0	0	0	0	0.5	0	0.5	5
15	0	0	0.5	0	0	0	0	1	0	0	0	0	1	0	0.5	5
16	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	6
17	0	0	0	0	0	1	0	0	0	0	0	0	0.5	0	0	6
18	0	0	0	0	0	0.5	0	0	0	0	0	0	0.5	0	1	6

Tabel 4.12 Data Uji Hasil Normalisasi

no.	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	kelas
41	0.5	0.5	0	0	0	0	0	0.5	0	0	0.5	0	0	0	0	1
76	0	0	0	1	0	0	0.5	0	0	0	0	1	0	0	0	2
140	0	0	0	0	1	0	0	0	0.5	0	0	0	0.5	0.5	0	3
187	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	4
265	0	0	0.5	0	0	0	0	0	0	0	0	0	0.5	0	1	5
285	0	0	0	0	0	0.5	0	0	0	0	0	0	0.5	0	0	6



Langkah 3. Menghitung validitas data latih.

Langkah ketiga yaitu menghitung nilai validitas data latih dengan menggunakan persamaan 2-3. Sebelum menghitung nilai validitas, terlebih dahulu mencari jarak data antar data training menggunakan perhitungan *Euclidean* dengan memasukkan data pada persamaan 2-1.

$$\begin{aligned}
 d_{(1,2)} &= \sqrt{\sum_{i=1}^p (x_{2i} - x_{1i})^2} \\
 &= \sqrt{(0.5 - 1)^2 + (1 - 1)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (1 - 1)^2} \\
 &\quad + (0 - 0)^2 + (0 - 0)^2 + (1 - 1)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2} \\
 &= 0.5
 \end{aligned}$$

Melakukan perhitungan yang sama untuk semua data latih. Hasil perhitungan *Euclidean* data latih ini dapat dilihat pada Tabel 4.13. Setelah ditentukan nilai k nya maka dihitung nilai validitas dari data latih yang memiliki jarak terdekat dengan persamaan 2-3.

$$\begin{aligned}
 \text{validitas}(x) &= \frac{1}{k} \sum_{i=1}^k S(\text{label}(x), (\text{label}(N_i(x)))) \\
 &= \frac{1}{3} \sum_{i=1}^3 S(\text{label}(x = 1), (\text{label}(N_i(x = 2)))) \\
 &= \frac{1}{3} \times (1 + 1 + 0) \\
 &= \frac{2}{3} \\
 &= 0.6667
 \end{aligned}$$

Melakukan perhitungan yang sama untuk semua data latih. Hasil perhitungan validitas ini dapat dilihat pada Tabel 4.14.

Langkah 4. Menghitung jarak *Euclidean* data uji

Pada langkah selanjutnya adalah mencari nilai *Euclidean* dengan memasukkan data uji pada persamaan 2-1, dimana contoh perhitungan data uji pertama sebagai berikut.

$$\begin{aligned}
 d_{(1,2)} &= \sqrt{\sum_{i=1}^p (x_{2i} - x_{1i})^2} \\
 &= \sqrt{(0.5 - 1)^2 + (0.5 - 1)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 +} \\
 &\quad (0.5 - 1)^2 + (0 - 0)^2 + (0 - 0)^2 + (0.5 - 1)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (0 - 0)^2} \\
 &= 1
 \end{aligned}$$

Tabel 4.14 Perhitungan Validitas

No	k=1	k=2	k=3	Sum S(a,b)	Validitas
1	1	1	0	2	0.666667
2	1	1	0	2	0.666667
3	1	1	0	2	0.666667
4	1	1	0	2	0.666667
5	1	1	0	2	0.666667
6	1	1	0	2	0.666667
7	1	1	0	2	0.666667
8	0	1	1	2	0.666667
9	1	1	0	2	0.666667
10	0	0	1	1	0.333333
11	0	1	1	2	0.666667
12	0	0	1	1	0.333333
13	1	1	0	2	0.666667
14	1	1	0	2	0.666667
15	1	1	0	2	0.666667
16	0	1	1	2	0.666667
17	0	1	1	2	0.666667
18	0	1	1	2	0.666667

Hasil perhitungan *Euclidean* ini dapat dilihat pada Tabel 4.15.

Tabel 4.15 Tabel Perhitungan Euclidean Data Uji 1

data	SUM euclidean	euclidean
d(41,1)	1	1
d(41,2)	0.75	0.866025
d(41,3)	0.5	0.707107
d(41,4)	5	2.236068
d(41,5)	2.75	1.658312
d(41,6)	2.75	1.658312
d(41,7)	5	2.236068
d(41,8)	2.75	1.658312
d(41,9)	3.5	1.870829
d(41,10)	5	2.236068
d(41,11)	3.5	1.870829
d(41,12)	2.25	1.5
d(41,13)	4	2
d(41,14)	2.25	1.5
d(41,15)	2.5	1.581139
d(41,16)	4	2
d(41,17)	2.25	1.5
d(41,18)	2.5	1.581139



Tabel 4.13 Perhitungan Euclidean Data Latih

d(1)	d(2)	d(3)	d(4)	d(5)	d(6)	d(7)	d(8)	d(9)	d(10)	d(11)	d(12)	d(13)	d(14)	d(15)	d(16)	d(17)	d(18)	kelas
0	0.5	0.71	2.83	2.40	2.40	2.83	2.40	2.55	2.83	2.55	2.29	2.45	2.18	2.12	2.65	2.29	2.35	1
0.50	0	0.87	2.69	2.24	2.24	2.69	2.24	2.40	2.69	2.40	2.12	2.29	2	1.94	2.50	2.12	2.18	1
0.71	0.87	0	2.55	2.06	2.06	2.55	2.06	2.24	2.55	2.24	1.94	2.12	1.80	1.73	2.35	1.94	2	1
2.83	2.69	2.55	0	0.87	0.87	2.45	2.18	2.12	2.45	2.35	1.80	2.83	2.40	2.55	2.65	2.29	2.35	2
2.40	2.24	2.06	0.87	0	0.71	2.18	1.73	1.80	2.18	1.94	1.41	2.40	1.87	2.06	2.18	1.73	1.80	2
2.40	2.24	2.06	0.87	0.71	0	1.94	1.58	1.50	1.94	1.80	1	2.40	1.87	2.06	2.18	1.73	1.80	2
2.65	2.69	2.55	2.45	2.18	1.94	0	0.87	0.71	1.41	1.58	1.80	2.45	2.18	2.12	2.24	2.06	2.12	3
2.35	2.24	2.06	2.18	1.73	1.58	0.87	0	0.87	1.66	1.5	1.41	2.18	1.73	1.80	1.94	1.58	1.66	3
2.35	2.40	2.24	2.12	1.80	1.50	0.71	0.87	0	1.22	1.41	1.32	2.35	1.94	2	2.12	1.80	1.87	3
2.65	2.69	2.55	2.45	2.18	1.94	1.41	1.66	1.22	0	0.71	1.50	2.45	2.18	2.12	2.24	2.06	2.12	4
2.50	2.40	2.24	2.35	1.94	1.80	1.58	1.50	1.41	0.71	0	1.32	2.12	1.80	1.73	1.87	1.66	1.73	4
2.29	2.12	1.94	1.80	1.41	1	1.80	1.41	1.32	1.5	1.32	0	2.29	1.73	1.94	2.06	1.58	1.66	4
2.45	2.29	2.12	2.83	2.40	2.40	2.45	2.18	2.35	2.45	2.12	2.29	0	0.87	0.71	1.73	2.06	1.58	5
2.18	2	1.80	2.40	1.87	1.87	2.18	1.73	1.94	2.18	1.80	1.73	0.87	0	0.87	1.66	1.58	1.32	5
2.12	1.94	1.73	2.55	2.06	2.06	2.12	1.80	2	2.12	1.73	1.94	0.71	0.87	0	1.58	1.66	1.41	5
2.65	2.50	2.35	2.65	2.18	2.18	2.24	1.94	2.12	2.24	1.87	2.06	1.73	1.66	1.58	0	1.12	0.71	6
2.29	2.12	1.94	2.29	1.73	1.73	2.06	1.58	1.80	2.06	1.66	1.58	2.06	1.58	1.66	1.12	0	1.12	6
2.35	2.18	2	2.35	1.80	1.80	2.12	1.66	1.87	2.12	1.73	1.66	1.58	1.32	1.41	0.71	1.12	0	6

Keterangan:

 jarak *Euclidean* yang memiliki ketetanggaan terdekat dengan data

Langkah 6. Menentukan kelas dari data uji

Setelah didapatkan nilai *weight voting* dari semua data latih maka dilakukan pencarian nilai *weight voting* yang terbesar sebanyak nilai *k* yang telah ditentukan. Hasil pencarian nilai *weight voting* terbesar dapat dilihat pada Tabel 4.17.

Tabel 4.16 Hasil Pencarian Nilai Weight voting Terbesar Sebanyak Nilai *k*

data	weight voting	diagnosis
d(41,3)	0.55228475	1
d(41,2)	0.488033872	1
d(41,1)	0.444444444	1
d(41,14)	0.333333333	5
d(41,17)	0.333333333	6
d(41,15)	0.320337431	5
d(41,18)	0.320337431	6
d(41,5)	0.308883305	2
d(41,6)	0.308883305	2
d(41,8)	0.308883305	3
d(41,9)	0.281195629	3
d(41,11)	0.281195629	4
d(41,13)	0.266666667	5
d(41,16)	0.266666667	6
d(41,4)	0.243658664	2
d(41,7)	0.243658664	3
d(41,12)	0.166666667	4
d(41,10)	0.121829332	4

Keterangan:  Nilai *weight voting* terbesar

Dari Tabel 4.17 maka didapatkan 3 *weight voting* terbesar yaitu 0.55228475 dengan kelasnya 1, 0.488033872 dengan kelasnya 1, dan 0.444444444 dengan kelasnya 1. Setelah diambil nilai *weight voting* terbesar sebanyak 3, lalu selanjutnya dilakukan penjumlahan nilai *weight voting* antar kelas yang sama. Dari data uji 1, maka kelas kategori 1 jumlah *weight voting*nya adalah 1.48 sehingga data uji 1 kelas kategorinya adalah kelas 1 karena nilai *weight voting*nya lebih besar dari kelas kategori lain.

Perhitungan yang sama dilakukan untuk data uji yang lainnya yaitu data 2, data 3, data 4, data 5 dan data 6. Dari 6 data uji yang digunakan didapatkan hasil akurasi sebesar 100% dengan 6 nilai prediksi benar dan tidak ada prediksi salah. Hasil perhitungan dari 6 data uji dapat dilihat dalam Tabel 4.18.

Tabel 4.17 Hasil Perhitungan 6 Data Uji

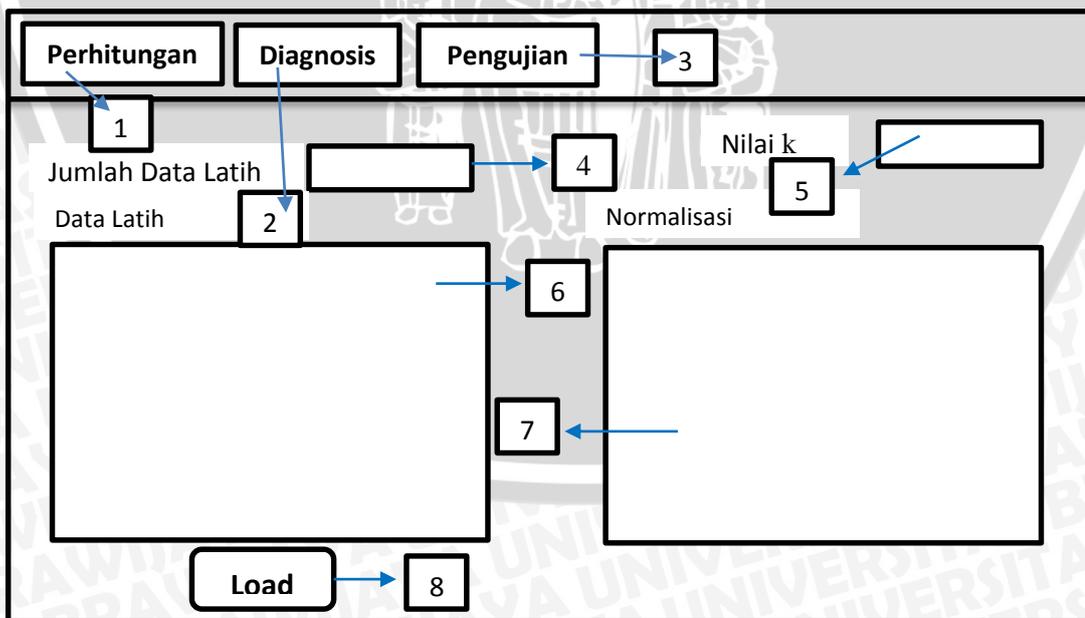
Nilai k	Data Uji	Hasil	
		Data Asli	Prediksi sistem
3	41	1	1
	76	2	2
	140	3	3
	187	4	4
	265	5	5
	285	6	6
Akurasi			100%

4.5 Perancangan desain antarmuka

Perancangan desain antarmuka pada aplikasi Diagnosis penyakit tanaman tomat menggunakan metode MKNN ini ada lima perancangan antarmuka. Perancangan antarmuka ini dibuat berdasarkan fungsi atau menu yang akan ditampilkan pada aplikasi. Kelima antarmuka tersebut yaitu halaman data latih, halaman Diagnosis, halaman hasil Diagnosis, halaman detail perhitungan dan halaman pengujian.

4.5.1 Halaman Data Latih

Halaman data latih merupakan halaman yang berfungsi untuk memasukkan jumlah data latih dan nilai k untuk proses perhitungan. Perancangan halaman data latih dapat dilihat pada Gambar 4.9.



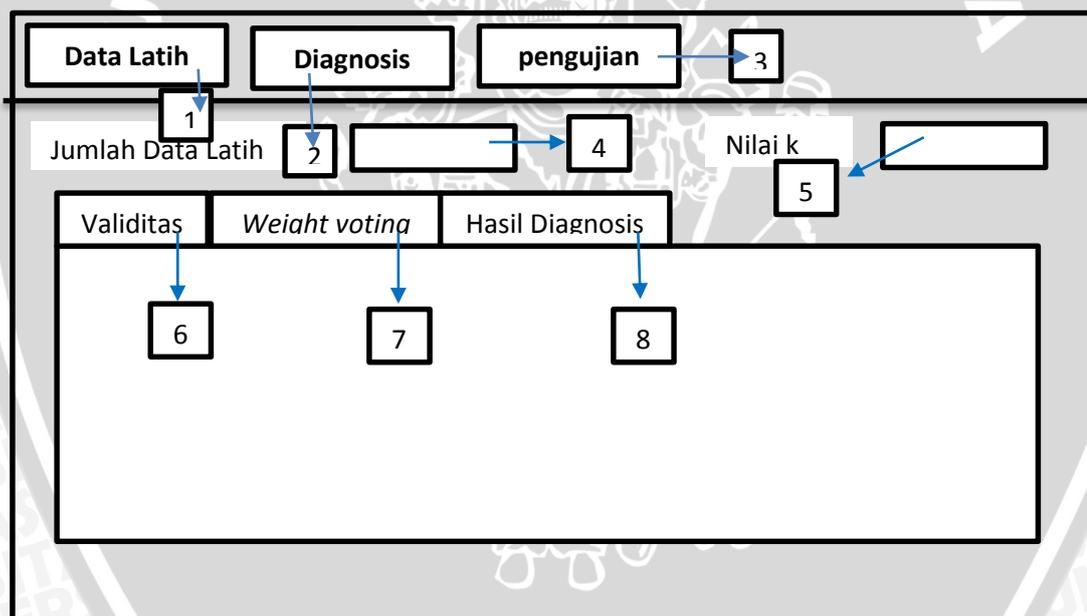
Gambar 4.9 Perancangan Halaman Data Latih

Keterangan Gambar:

1. Menu Perhitungan untuk melihat detail hasil perhitungan dari metode MKNN.
2. Menu Diagnosis untuk melihat halaman Diagnosis.
3. Menu pengujian untuk melihat halaman pengujian.
4. *Field* untuk memasukkan jumlah data latih
5. *Field* untuk memasukkan nilai k.
6. *Field* yang menampilkan data latih.
7. *Field* yang menampilkan hasil normalisasi data latih.
8. Tombol untuk load data latih dan memproses normalisasi

4.5.2 Halaman Hasil Perhitungan

Halaman hasil perhitungan merupakan halaman untuk melihat detail hasil perhitungan dari metode *Modified K-Nearest Neighbor*. Perancangan halaman detail perhitungan dapat dilihat pada Gambar 4.10



Gambar 4.10 Perancangan Halaman Hasil Perhitungan

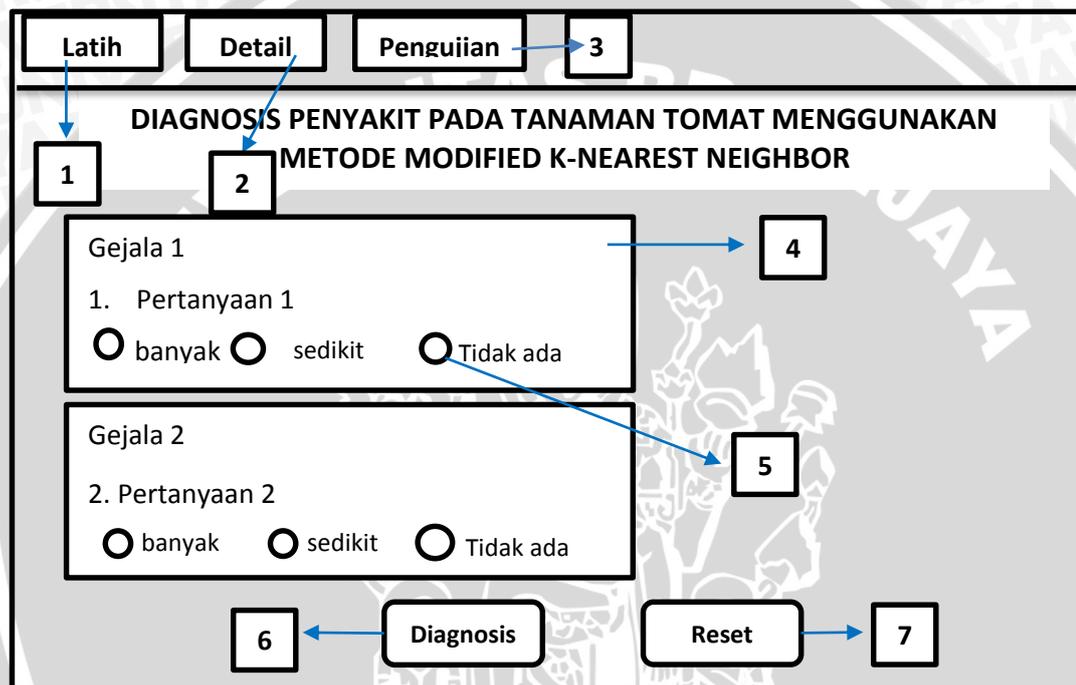
Keterangan Gambar:

1. Menu Latih untuk melihat halaman data latih.
2. Menu Diagnosis untuk melihat halaman Diagnosis.
3. Menu pengujian untuk melihat halaman pengujian.
4. *Field* untuk menampilkan jumlah data latih.
5. Filed untuk menampilkan nilai k.

6. Tab *Field* yang menampilkan hasil perhitungan validitas.
7. Tab *Field* yang menampilkan hasil perhitungan *weight voting*.
8. Tab *Field* yang menampilkan hasil diagnosis.

4.5.3 Halaman Diagnosis Penyakit Tanaman Tomat

Halaman Diagnosis penyakit tanaman tomat merupakan halaman yang digunakan untuk memasukkan gejala-gejala pada penyakit tanaman tomat. Perancangan halaman Diagnosis penyakit tanaman tomat dapat dilihat pada Gambar 4.11.



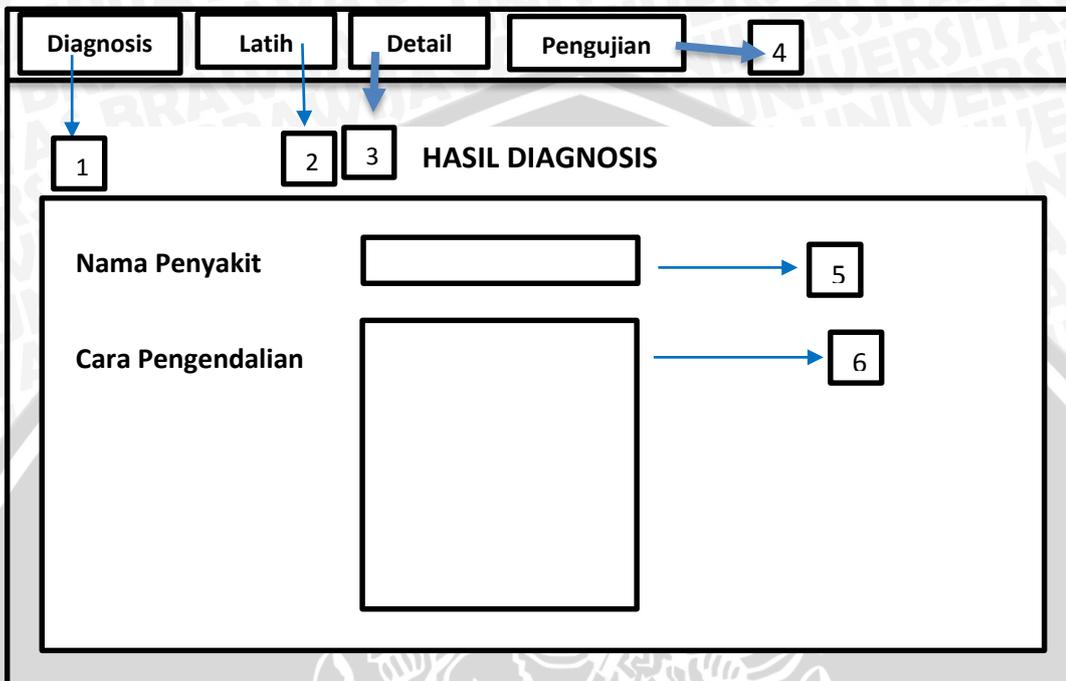
Gambar 4.11 Perancangan Halaman Diagnosis Penyakit Tanaman Tomat

Keterangan Gambar:

1. Menu data latih untuk membuka halaman data latih.
2. Menu Detail untuk membuka halaman detail perhitungan.
3. Menu pengujian untuk membuka halaman pengujian.
4. *Field* yang menampilkan pertanyaan tentang gejala penyakit tanaman tomat.
5. Radio button untuk menjawab pertanyaan tentang gejala penyakit tanaman tomat.
6. Tombol untuk menampilkan hasil Diagnosis.
7. Tombol untuk mengosongkan jawaban pada *Field* pertanyaan.

4.5.4 Halaman Hasil Diagnosis Penyakit Tanaman Tomat

Halaman hasil Diagnosis penyakit tanaman tomat merupakan halaman untuk melihat hasil Diagnosis penyakit dan cara pengendaliannya. Perancangan halaman hasil Diagnosis penyakit tanaman tomat ditunjukkan pada Gambar 4.12.



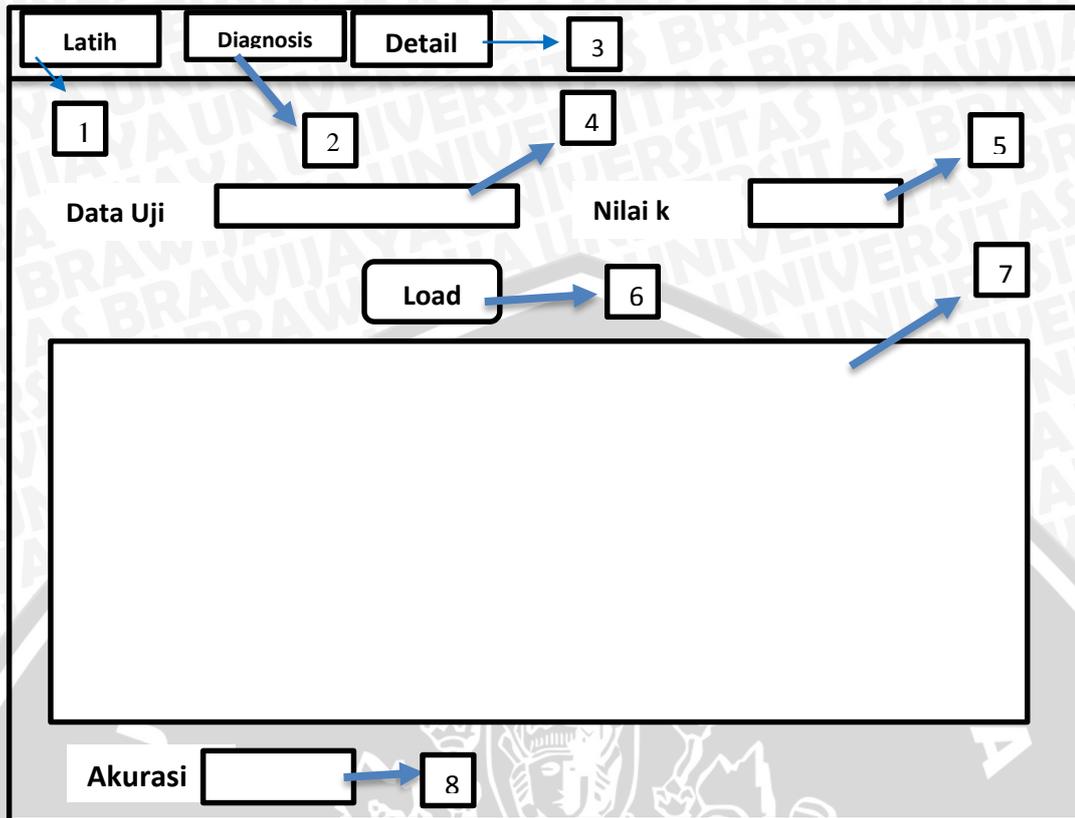
Gambar 4.12 Perancangan Halaman Hasil Diagnosis Penyakit Tanaman Tomat

Keterangan Gambar:

1. Menu Diagnosis untuk membuka halaman utama.
2. Menu Latih untuk membuka halaman data latih.
3. Menu detail untuk membuka halaman detail perhitungan.
4. Menu pengujian untuk melihat halaman pengujian.
5. *Field* yang menampilkan nama penyakit tanaman tomat.
6. *Field* yang menampilkan cara pengendalian penyakit yang menyerang tanaman tomat.

4.5.5 Halaman Pengujian

Halaman Pengujian merupakan halaman yang digunakan untuk melakukan pengujian terhadap sistem untuk memperoleh hasil akurasi. Perancangan halaman Pengujian dapat dilihat pada Gambar 4.13.



Gambar 4.13 Perancangan Halaman Pengujian

Keterangan Gambar:

1. Menu latihan untuk membuka halaman data latihan
2. Menu Diagnosis untuk membuka halaman Diagnosis penyakit.
3. Menu detail untuk membuka halaman detail perhitungan.
4. *Field* yang menampilkan path dari file data uji
5. *Field* yang menampilkan nilai k yang sudah diinputkan
6. Tombol untuk memproses dan menampilkan data uji.
7. *Field* yang menampilkan data uji beserta hasil prediksi sistem.
8. *Field* yang menampilkan hasil akurasi dari data uji.

4.6 Perancangan pengujian

Pada sub bab ini akan dijelaskan mengenai pengujian data yang akan dilakukan pada sistem. Pengujian yang akan dilakukan terdapat tiga macam yaitu pengujian pengaruh nilai k terhadap akurasi, pengujian pengaruh jumlah data latihan berubah dan data uji tetap terhadap akurasi dan pengujian pengaruh data latihan seimbang dan data latihan tidak seimbang terhadap akurasi. *Output* dari proses pengujian ini adalah hasil prediksi sistem yang kemudian dibandingkan dengan data yang sudah

ada. Kemudian dari perbandingan tersebut akan didapatkan nilai akurasi. Berikut adalah penjelasan mengenai deskripsi pengujian yang akan dilakukan.

4.6.1 Perancangan Pengujian Pengaruh Nilai k Terhadap Akurasi

Nilai k atau nilai ketetangaan merupakan suatu hal yang mempengaruhi kinerja pada metode *Modified K-Nearest Neighbor*. Percobaan akan dilakukan dengan nilai k 1 sampai 10 dengan jumlah data latih 25% (72 data), data latih 50% (144 data) dan data latih 75% (216 data). Hasil pengujian akan ditampilkan pada Tabel 4.19

Tabel 4.18 Pengaruh Nilai k terhadap akurasi

Nilai k	Akurasi Pengujian Untuk Data Latih (%)			Rata-Rata Akurasi (%)
	25%	50%	75%	
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

4.6.2 Perancangan Pengujian Pengaruh Jumlah Data Latih Berubah dan Data Uji Tetap Terhadap Akurasi

Tujuan dari pengujian ini untuk mengetahui apakah jumlah data latih berpengaruh terhadap akurasi yang dihasilkan oleh sistem. Data latih yang digunakan menggunakan data latih dengan kelas seimbang dan nilai k yang digunakan diambil nilai k terbaik dari pengujian sebelumnya. Hasil pengujian akan ditampilkan pada Tabel 4.20.

Tabel 4.19 Pengaruh Jumlah Data Latih Berubah dan Data Uji Tetap

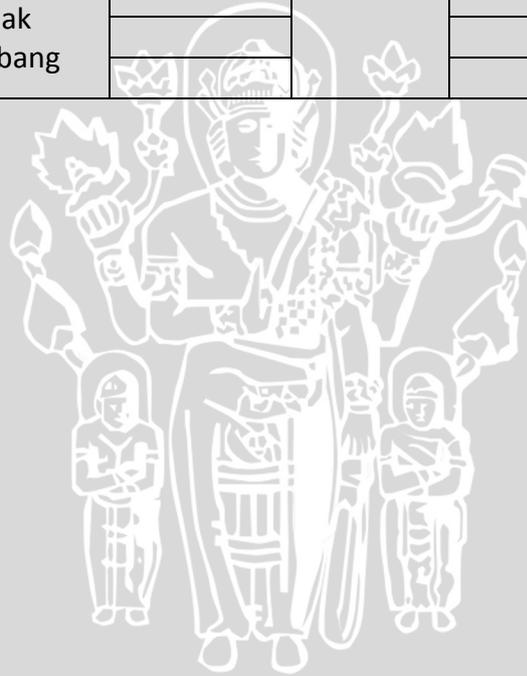
Nilai k	Jumlah Data Uji	Jumlah Data Latih	Akurasi (%)

4.6.3 Perancangan Pengujian Pengaruh Data Latih Seimbang dan Data Latih Tidak Seimbang

Pengujian ini dilakukan untuk mengetahui pengaruh data latih seimbang dan data latih tidak seimbang terhadap akurasi yang dihasilkan sistem. Jumlah data latih seimbang dan data latih tidak seimbang yang digunakan diambil dari jumlah data latih terbaik dari pengujian kedua. Nilai k yang digunakan diambil dari nilai k terbaik hasil pengujian pertama dan data uji yang digunakan sebanyak 50 data diambil secara acak. Hasil pengujian akan ditampilkan pada Tabel 4.21.

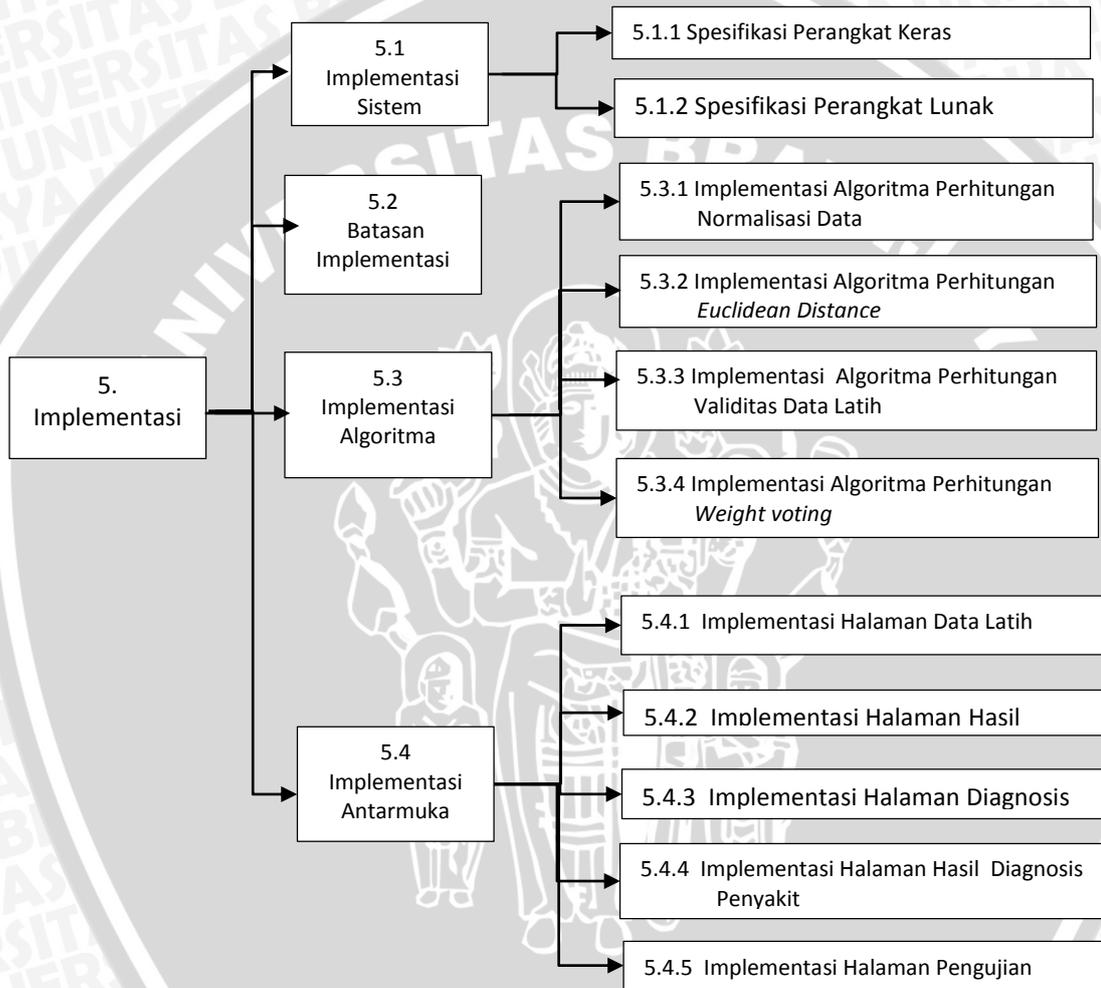
Tabel 4.20 Pengaruh Data Latih Seimbang dan Data Latih Tidak Seimbang

Jenis Data Latih	Jumlah Data Latih	Jumlah Data Uji	Akurasi (%)
Seimbang			
Tidak Seimbang			



BAB 5 IMPLEMENTASI

Bab ini membahas mengenai implementasi perangkat lunak berdasarkan hasil yang telah diperoleh dari analisis kebutuhan dan proses perancangan perangkat lunak yang dibuat. Pembahasan terdiri dari penjelasan tentang spesifikasi sistem, batasan-batasan dalam implementasi, implementasi algoritma pada program, dan implementasi antarmuka. Berikut merupakan tahapan-tahapan implementasi sistem ditunjukkan pada Gambar 5.1.



Gambar 5.1 Pohon Implementasi

5.1 Implementasi sistem

Hasil dari proses *Analisis* kebutuhan dan perancangan perangkat lunak yang telah diuraikan pada Bab 4 akan 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 dibutuhkan oleh sistem terdiri dari spesifikasi perangkat keras dan spesifikasi perangkat lunak.

5.1.1 Spesifikasi perangkat keras

Pengembangan Sistem untuk Diagnosis Penyakit pada Tanaman Tomat menggunakan perangkat komputer dengan spesifikasi perangkat keras seperti Tabel 5.1.

Tabel 5.1 Spesifikasi Perangkat Keras

Nama Komponen	Spesifikasi
Prosesor	Intel® Core™ CPU B940 @ 2.00 GHz
Memori	2.0 GB DDR3
Kartu Grafis	Intel® HD Graphic
Harddisk	320 GB

5.1.2 Spesifikasi perangkat lunak

Pengembangan sistem untuk Diagnosis penyakit tanaman tomat menggunakan perangkat komputer dengan spesifikasi perangkat lunak seperti pada Tabel 5.2.

Tabel 5.2 Spesifikasi Perangkat Lunak

Nama	Spesifikasi
Sistem operasi	Microsoft Windows 8.1 Profesional 64-bit
Bahasa pemrograman	C#
Tools Pemrograman	Microsoft Visual Studio Professional 2013
DBMS	Mysql
Tools DBMS	Mysql 1.8.2

5.2 Batasan implementasi

Beberapa batasan yang digunakan dalam mengimplementasikan Sistem Diagnosis Penyakit Tanaman Tomat adalah sebagai berikut :

1. Sistem dibangun berdasarkan ruang lingkup *Dekstop Application* dengan menggunakan bahasa pemrograman C#.
2. Data-data yang digunakan dalam sistem disimpan dalam *Database Management System* (DBMS) MySQL.

3. Metode yang digunakan dalam menyelesaikan masalah adalah Metode *Modified K-Nearest Neighbor*(MKNN).
4. *Input* yang digunakan dalam sistem merupakan gejala penyakit yang menyerang tanaman tomat.
5. *Output* yang diterima pengguna adalah Diagnosis penyakit yang menyerang tanaman tomat.
6. Sistem ini dapat mendiagnosis 6 jenis penyakit tanaman tomat dan 15 gejala penyakit tanaman tomat.

5.3 Implementasi algoritma

Implementasi algoritma ini mengacu pada bab perancangan sub bab perancangan perangkat lunak yang mempunyai beberapa proses utama yang terbagi dalam beberapa fungsi. Pada sub bab ini hanya dicantumkan algoritma dari sebagian proses saja, sehingga tidak semua proses algoritma dicantumkan. Proses algoritma yang dicantumkan yaitu implementasi algoritma normalisasi data, implementasi algoritma perhitungan *euclidean*, implementasi algoritma perhitungan validitas data latih dan implementasi algoritma perhitungan *weight voting*.

5.3.1 Implementasi algoritma perhitungan normalisasi data

Proses perhitungan normalisasi data pada metode MKNN berguna untuk menyetarakan sebaran data sehingga dapat memudahkan proses perhitungan yang menggunakan Persamaan 2-2. Proses normalisasi dilakukan pada data uji dan data latih. Implementasi algoritma perhitungan normalisasi data terdapat pada Gambar 5.2.

```

27 public Normalisasi(Tomat tom, double[] max)
28     {
29         G1 = (double) tom.getG1() / (double)max[0];
30         G2 = (double) tom.getG2() / (double)max[1];
31         G3 = (double) tom.getG3() / (double)max[2];
32         G4 = (double) tom.getG4() / (double)max[3];
33         G5 = (double) tom.getG5() / (double)max[4];
34         G6 = (double) tom.getG6() / (double)max[5];
35         G7 = (double) tom.getG7() / (double)max[6];
36         G8 = (double) tom.getG8() / (double)max[7];
37         G9 = (double) tom.getG9() / (double)max[8];
38         G10 = (double) tom.getG10() / (double)max[9];
39         G11 = (double) tom.getG11() / (double)max[10];
40         G12 = (double) tom.getG12() / (double)max[11];
41         G13 = (double) tom.getG13() / (double)max[12];
42         G14 = (double) tom.getG14() / (double)max[13];
43         G15 = (double) tom.getG15() / (double)max[14];
44     }

```

Source Code 5.1 Listing Code Proses Normalisasi Data

Penjelasan *source code* algoritma perhitungan normalisasi data pada Gambar 5.2 adalah sebagai berikut:

1. Melakukan normalisasi pada gejala 1 sampai gejala 15 di setiap data pada sistem menggunakan rumus yang telah ditentukan, ditunjukkan pada baris 29-43.

5.3.2 Implementasi algoritma perhitungan *euclidean distance*

Proses perhitungan *euclidean distance* ini menghitung jarak antar data latih maupun data uji yang nantinya akan digunakan dalam proses menghitung validitas data latih. Implementasi algoritma perhitungan *euclidean* terdapat pada Gambar 5.3.

```
144 public double hitungDistances(Tomat tom, double dist, int j) {
145
146     Tomat tmp = this.ntomat[j];
147
148     //jarak antara G1
149     dist = dist + getDistance(tom.getG1(), tmp.getG1());
150
151     // jarak antara G2
152     dist = dist + getDistance(tom.getG2(), tmp.getG2());
153
154     // jarak antara G3
155     dist = dist + getDistance(tom.getG3(), tmp.getG3());
156
157     // jarak antara G4
158     dist = dist + getDistance(tom.getG4(), tmp.getG4());
159
160     // jarak antara G5
161     dist = dist + getDistance(tom.getG5(), tmp.getG5());
162
163     // jarak antara G6
164     dist = dist + getDistance(tom.getG6(), tmp.getG6());
165
166     // jarak antara G7
167     dist = dist + getDistance(tom.getG7(), tmp.getG7());
168
169     // jarak antara G8
170     dist = dist + getDistance(tom.getG8(), tmp.getG8());
171
172     // jarak antara G9
173     dist = dist + getDistance(tom.getG9(), tmp.getG9());
174
175     // jarak antara G10
176     dist = dist + getDistance(tom.getG10(), tmp.getG10());
177
178     // jarak antara G11
179     dist = dist + getDistance(tom.getG11(), tmp.getG11());
180
181     // jarak antara G12
182     dist = dist + getDistance(tom.getG12(), tmp.getG12());
183
184     // jarak antara G13
185     dist = dist + getDistance(tom.getG13(), tmp.getG13());
186 }
```

```

187
188         // jarak antara G14
189         dist = dist + getDistance(tom.getG14(), tmp.getG14());
190
191         // jarak antara G15
192         dist = dist + getDistance(tom.getG15(), tmp.getG15());
193
194         dist = (double) Math.Sqrt(dist);
195
196         return dist;
197     }
198

```

Source Code 5.2 Listing Code Proses Euclidean Distance

Penjelasan *source code* algoritma perhitungan *euclidean distance* pada Gambar 5.3 adalah sebagai berikut:

1. Menjumlahkan jarak pada gejala 1 ditunjukkan pada baris 149-150.
2. Menjumlahkan jarak pada gejala 2 ditunjukkan pada baris 152-153.
3. Menjumlahkan jarak pada gejala 3 ditunjukkan pada baris 155-156.
4. Menjumlahkan jarak pada gejala 4 ditunjukkan pada baris 158-159.
5. Menjumlahkan jarak pada gejala 5 ditunjukkan pada baris 161-162.
6. Menjumlahkan jarak pada gejala 6 ditunjukkan pada baris 164-165.
7. Menjumlahkan jarak pada gejala 7 ditunjukkan pada baris 167-168.
8. Menjumlahkan jarak pada gejala 8 ditunjukkan pada baris 170-171.
9. Menjumlahkan jarak pada gejala 9 ditunjukkan pada baris 173-174.
10. Menjumlahkan jarak pada gejala 10 ditunjukkan pada baris 176-177.
11. Menjumlahkan jarak pada gejala 11 ditunjukkan pada baris 179-180.
12. Menjumlahkan jarak pada gejala 12 ditunjukkan pada baris 182-183.
13. Menjumlahkan jarak pada gejala 13 ditunjukkan pada baris 185-186.
14. Menjumlahkan jarak pada gejala 14 ditunjukkan pada baris 188-189.
15. Menjumlahkan jarak pada gejala 15 ditunjukkan pada baris 191-192.
16. Menghitung hasil akhir *euclidean distance* dengan mengakarkan hasil penjumlahan setiap gejala ditunjukkan pada baris 195.

5.3.3 Implementasi algoritma perhitungan validitas data latih

Pada proses perhitungan validitas data latih ini dilakukan dengan membandingkan kelas pada data latih sesuai dengan nilai k yang diinputkan. Jika kelas yang dibandingkan sama maka akan bernilai 1 dan jika kelas yang dibandingkan tidak sama maka akan bernilai 0 sesuai dengan Persamaan 2-4 . Jumlah dari hasil perbandingan tersebut dibagi dengan nilai k yang diinputkan dan

menghasilkan nilai validitas setiap data latih sesuai dengan Persamaan 2-3. Implementasi algoritma perhitungan validitas data latih terdapat pada Gambar 5.4

```

70 int sum = 0;
71     for (int i = 1; i < k + 1; i++)
72     {
73         Tomat tmp = ntomat[ distances[i].index ];
74
75         if (Convert.ToInt32(tmp.getDiagnosis()) ==
76 Convert.ToInt32(tom.getDiagnosis()))
77         {
78             sum = sum + 1;
79         }
80     }
81
82     // Hitung Validitas
83     double valid = (1 / (double) k) * (double) sum;
84
85     return valid;

```

Source Code 5.3 Listing Code Proses Validitas Data Latih

Penjelasan source code algoritma perhitungan validitas data latih pada Gambar 5.4 adalah sebagai berikut:

1. Memberikan nilai 0 jika kelas tidak sama ditunjukkan pada baris 70.
2. Memberikan nilai 1 jika kelas sama ditunjukkan pada baris 71-80.
3. Menghitung nilai validitas ditunjukkan pada baris 83.

5.3.4 Implementasi algoritma perhitungan *weight voting*

Pada proses perhitungan *weight voting* ini membahas perhitungan bobot pada data penyakit tanaman tomat, dengan cara membandingkan hasil perhitungan validitas dengan *euclidean*. Dari tahapan ini akan dilakukan pembobotan yang bertujuan untuk menentukan kelas pada data uji. Implementasi algoritma perhitungan *weight voting* terdapat pada Gambar 5.5.

```

131 double[] wvoting = new double[jumlahDt];
132 List<WeightVoting> wv = new List<WeightVoting>();
133 bdb.query("DELETE FROM weight_voting");
134     for (int i =0; i < jumlahDt; i++)
135     {
136         WeightVoting w = new WeightVoting();
137         w.value = (validitas[i] * (1 / (dd[i].distance +
138 0.5)));
139
140         w.index = dd[i].index;
141         w.diagnosis =
142 Convert.ToInt32(testset[i].getDiagnosis());
143         wv.Add(w);
144
145         bdb.query("INSERT INTO weight_voting(id, wv) value("+
146 (i+1) +", "+ w.value + ")");
147     }

```

Source Code 5.4 Listing Code Proses Weight Voting

Penjelasan *source code* algoritma perhitungan *weight voting* pada Gambar 5.5 adalah sebagai berikut:

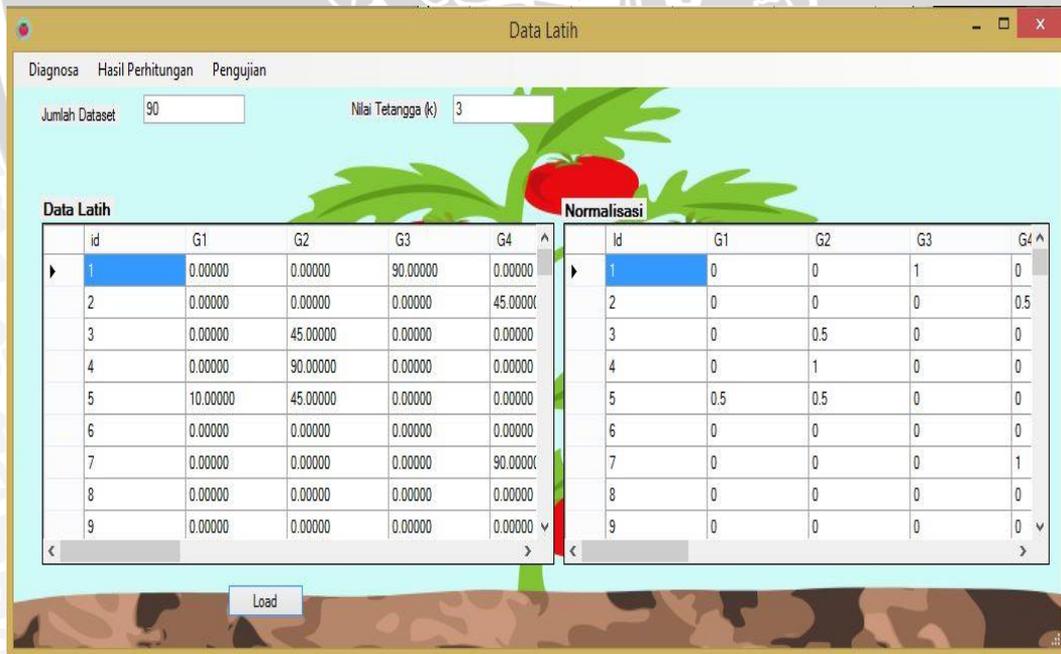
1. Menghapus nilai *weight voting* pada *database* ditunjukkan pada baris 133.
2. Menghitung nilai *weight voting* ditunjukkan pada baris 134-143.
3. Menambahkan nilai *weight voting* baru pada *database* ditunjukkan pada baris 145-147.

5.4 Implementasi antarmuka

Antarmuka Aplikasi untuk diagnosis penyakit tanaman tomat ini digunakan oleh pengguna untuk berinteraksi dengan sistem perangkat lunak. Pada implementasi antarmuka perangkat lunak ini tidak semua halaman pada sistem ditampilkan, tetapi hanya antarmuka halaman tertentu saja. Beberapa antarmuka yang ditampilkan yaitu implementasi halaman data latih, implementasi halaman detail perhitungan, implementasi halaman Diagnosis penyakit, implementasi halaman hasil Diagnosis penyakit dan implementasi halaman pengujian.

5.4.1 Implementasi antarmuka halaman data latih

Halaman data latih merupakan halaman untuk menginputkan jumlah data latih dan nilai k yang akan digunakan. Pada halaman ini menampilkan data latih asli dan data latih yang telah dinormalisasi. Data latih yang telah dinormalisasi ini kemudian digunakan untuk proses perhitungan berikutnya. Implementasi antarmuka halaman data latih terdapat pada Gambar 5.2.



Gambar 5.2 Implementasi Antarmuka Halaman Data Latih

5.4.2 Implementasi antarmuka halaman hasil perhitungan

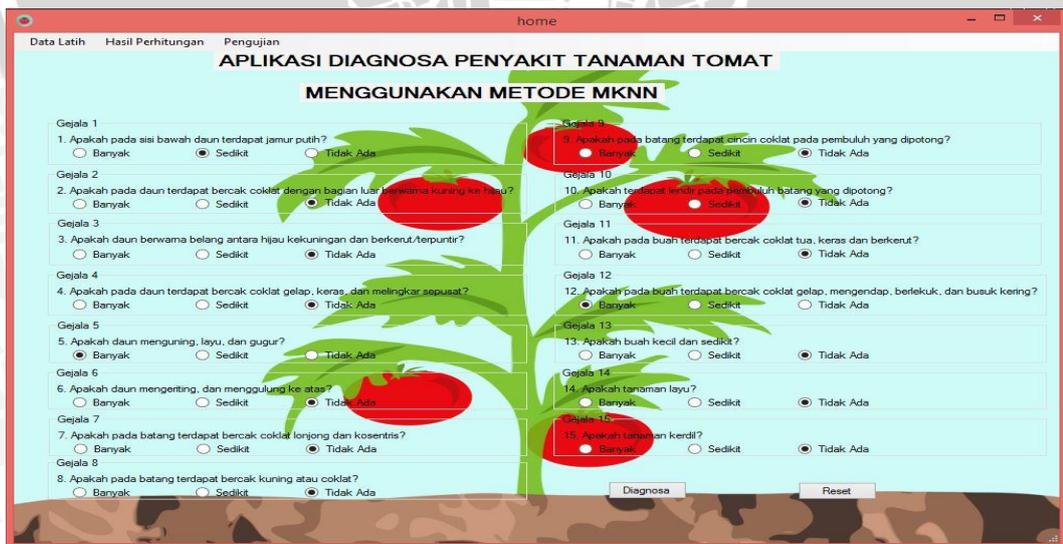
Halaman detail perhitungan merupakan halaman yang berfungsi untuk menampilkan hasil perhitungan dari metode MKNN. Nilai perhitungan yang ditampilkan yaitu nilai validitas dan nilai perhitungan *weight voting*. Implementasi antarmuka halaman detail perhitungan terdapat pada Gambar 5.3.



Gambar 5.3 Implementasi Antarmuka Halaman Detail Perhitungan

5.4.3 Implementasi antarmuka halaman diagnosis penyakit

Halaman Diagnosis penyakit merupakan halaman untuk menginputkan gejala penyakit tanaman tomat. *User* harus menjawab 15 pertanyaan yang terdapat pada sistem agar sistem bisa menDiagnosis penyakit tanaman tomat. Setelah menjawab semua pertanyaan *user* mengklik tombol Diagnosis untuk mendapatkan hasil. Implementasi antarmuka halaman Diagnosis penyakit terdapat pada Gambar 5.4.



Gambar 5.4 Implementasi Antarmuka Halaman Diagnosis Penyakit

5.4.4 Implementasi antarmuka halaman hasil diagnosis penyakit

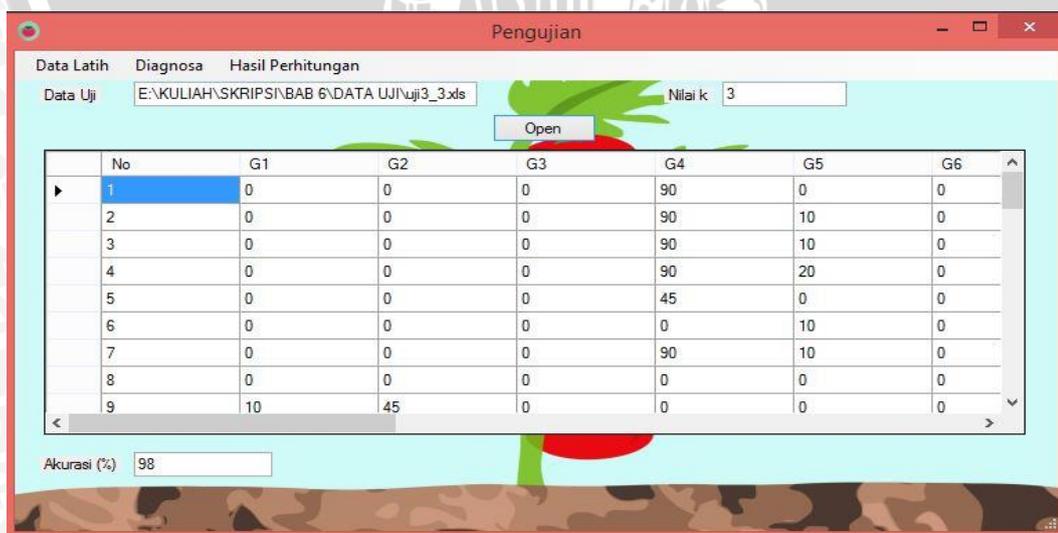
Halaman hasil Diagnosis penyakit merupakan halaman yang berfungsi untuk menampilkan hasil Diagnosis atau prediksi penyakit tanaman tomat. Pada halaman ini ditampilkan jenis penyakit yang menyerang tanaman tomat, cara pengendalian dan gambar dari penyakit yang diDiagnosis. Implementasi antarmuka halaman hasil Diagnosis penyakit terdapat pada Gambar 5.5.



Gambar 5.5 Implementasi Antarmuka Halaman Hasil Diagnosis Penyakit

5.4.5 Implementasi antarmuka halaman pengujian

Halaman pengujian merupakan halaman untuk melakukan pengujian pada data uji. Pada halaman ini data uji diinputkan kemudian sistem akan menampilkan kelas prediksi dan kelas dari sistem dicocokkan dengan kelas pada data asli. Selain itu pada halaman ini juga ditampilkan hasil akurasi sistem. Implementasi antarmuka halaman pengujian terdapat pada Gambar 5.6.



Gambar 5.6 Implementasi Antarmuka Halaman Pengujian

BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini membahas mengenai proses pengujian terhadap aplikasi yang dibuat dan metode yang digunakan. Proses pengujian dilakukan melalui tiga tahap pengujian, yaitu pengujian fungsionalitas, pengujian pengaruh nilai k dan pengujian pengaruh penambahan jumlah data latih. Pengujian tersebut dilakukan untuk mengetahui hasil akurasi dari implementasi yang telah dilakukan.

6.1 Pengujian pengaruh nilai k

Pengujian pengaruh nilai k dilakukan untuk mengetahui apakah nilai k berpengaruh terhadap hasil akurasi sistem jika nilai k tersebut diubah. Pengujian ini dilakukan dengan cara mengubah nilai k dari $k=1$ sampai $k=10$ kemudian masing-masing nilai k tersebut dicoba pada saat data latih 25%, data latih 50% dan data latih 75%. Hasil akurasi dari masing-masing nilai k pada data latih yang berbeda dirata-rata dan kemudian hasil tersebut dibandingkan untuk dicari nilai k yang menghasilkan akurasi sistem tertinggi.

6.1.1 Skenario Pengujian Pengaruh Nilai k

Prosedur pengujian pengaruh nilai k dilakukan dengan cara menghitung nilai akurasi pada $k=1$ sampai $k=10$. Setiap nilai k akan dilakukan uji coba sebanyak 3 kali dan didapatkan nilai akurasi rata-rata. Jumlah data latih yang digunakan terdapat 3 macam yaitu data latih 25%, data latih 50% dan data latih 75%. Data latih 25% terdiri dari 72 data latih sedangkan data uji yang digunakan berjumlah 216 data. Data latih 50% terdiri dari 144 data latih dan 144 data uji. Data latih 75% terdiri dari 216 data latih dan data uji yang digunakan berjumlah 72 data.

6.1.2 Analisis Hasil Skenario Pengujian Pengaruh Nilai k

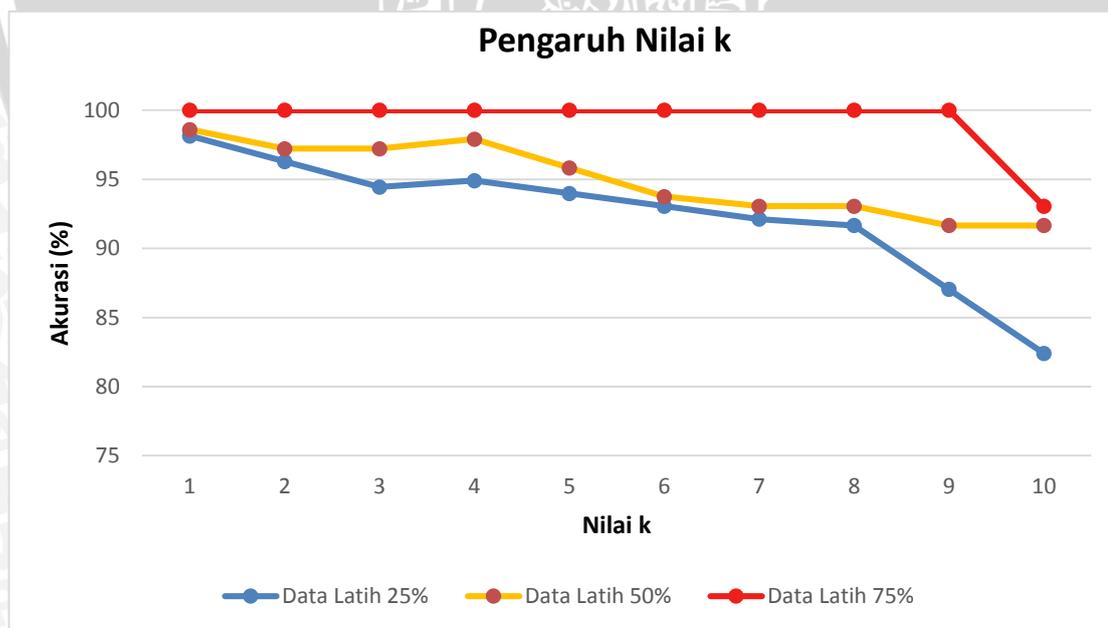
Pada pengujian pengaruh nilai k yang telah dilakukan, didapatkan rata-rata nilai akurasi yang berbeda-beda. Nilai akurasi rata-rata tertinggi sebesar 98,92% pada saat nilai $k=1$, sedangkan nilai akurasi rata-rata terendah yang dihasilkan sebesar 89,04% pada saat nilai $k=10$. Dapat dilihat pada Tabel 6.16 bahwa untuk setiap kenaikan nilai k maka hasil akurasi akan semakin menurun sampai nilai $k = 10$. Hasil pengujian pengaruh nilai k ditunjukkan pada Tabel 6.16.

Berdasarkan hasil uji coba dengan masing-masing data uji, perubahan nilai k memberikan pengaruh terhadap nilai akurasi rata-rata yang dihasilkan. Pengaruh tersebut dapat dilihat ketika terjadi penambahan nilai k maka nilai akurasi rata-rata cenderung menurun seiring dengan penambahan nilai k tersebut. Hal ini disebabkan karena semakin besar nilai k maka semakin banyak ketetapan yang digunakan untuk proses klasifikasi sehingga kemungkinan untuk terjadi *noise* semakin besar. Sebaliknya semakin kecil nilai k maka semakin sedikit ketetapan yang digunakan untuk proses klasifikasi data baru sehingga kemungkinan terjadi *noise* pun semakin kecil.

Tabel 6.1 Hasil pengujian pengaruh nilai k

Nilai k	Akurasi Pengujian Untuk Data Latih (%)			Rata-Rata Akurasi (%)
	25%	50%	75%	
1	98.15	98.61	100	98.92
2	96.30	97.22	100	97.84
3	94.44	97.22	100	97.22
4	94.91	97.92	100	97.61
5	93.98	95.83	100	96.60
6	93.06	93.75	100	95.60
7	92.13	93.06	100	95.06
8	91.67	93.06	100	94.91
9	87.04	91.67	100	92.90
10	82.41	91.67	93.06	89.04

Dengan adanya proses perhitungan jarak menggunakan *euclidean distance* pada metode ini, ketika nilai k kecil maka hanya tetangga yang memiliki kedekatan data terbaik saja yang digunakan. Hal ini dikarenakan metode *euclidean distance* digunakan untuk mencari jarak terdekat antar data dimana semakin kecil nilai *euclidean distance* maka jarak antar data semakin dekat. Pada uji coba dengan 288 dataset yang digunakan nilai akurasi rata-rata tertinggi dihasilkan pada saat k=1, karena nilai k yang kecil akan mengurangi *noise*. Namun, nilai akurasi tersebut tidak hanya dipengaruhi oleh besar nilai k tetapi juga dipengaruhi juga oleh sebaran data dari masing-masing kelas dan jumlah data latih yang digunakan. Grafik pengaruh nilai k ditunjukkan pada Gambar 6.1.



Gambar 6.1 Grafik pengaruh nilai k terhadap rata-rata akurasi

Pada pengujian yang dilakukan nilai akurasi maksimum cenderung terjadi saat nilai $k=1$. Hal ini disebabkan karena pada data latih rata-rata data antar kelas memiliki selisih yang sangat kecil, dengan demikian dapat diartikan jarak antar kelas pada dataset yang digunakan sangat berdekatan. Semakin dekat jarak suatu data dapat diartikan bahwa data tersebut hampir memiliki karakteristik yang sama. Sehingga ketika nilai $k=1$ berarti hanya ada 1 data tetangga terdekat dengan karakteristik data yang hampir sama dengan data uji bahkan tergolong dalam kelas yang sama yang digunakan dalam proses klasifikasi, hal ini memungkinkan untuk melakukan klasifikasi dengan benar. Sebaliknya, jika nilai k yang digunakan semakin besar berarti semakin banyak tetangga yang digunakan dalam proses klasifikasi termasuk dengan data yang berjauhan dengan data uji dimana hampir tidak memiliki karakteristik yang sama. Hal ini menyebabkan terjadinya *noise* yang semakin besar ditambah lagi dengan adanya dominasi atau frekuensi kelas data latih yang tidak seimbang dari suatu kelas tertentu sehingga hasilnya data cenderung diklasifikasikan pada data kelas yang mendominasi.

6.2 Pengujian pada jumlah data latih berbeda dengan jumlah data uji tetap

Pengujian pada jumlah data latih yang berbeda dengan jumlah data uji tetap dilakukan untuk mengetahui apakah penambahan pada jumlah data latih memiliki pengaruh terhadap tingkat akurasi yang dihasilkan sistem. Pengujian ini dilakukan dengan cara mengubah jumlah data latih sebanyak tiga kali dan nilai k yang digunakan adalah nilai k terbaik yang dihasilkan dari pengujian sebelumnya yaitu $k=1$. Untuk data latih yang digunakan merupakan data latih dengan kelas seimbang dengan jumlah 30 data, 60 data dan 90 data.

6.2.1 Skenario Pengujian Pada Jumlah Data Latih Berbeda Dengan Jumlah Data Uji Tetap

Prosedur pengujian pada penambahan data latih dengan data uji tetap dilakukan dengan cara menghitung nilai akurasi pada saat jumlah data latih sebanyak 30 data dan data uji sebanyak 50 data. Selanjutnya menghitung nilai akurasi pada saat jumlah data latih ditambah menjadi 60 data dan data uji tetap sebanyak 50 data. Data latih ditambahkan lagi menjadi 90 data dan data uji tetap sebanyak 50 data kemudian dihitung nilai akurasi sistem. Nilai k yang digunakan pada pengujian ini diambil nilai k terbaik pada hasil pengujian sebelumnya yaitu $k=1$.

6.2.2 Analisis Hasil Skenario Pengujian Pada Jumlah Data Latih Berbeda Dengan Jumlah Data Uji Tetap

Pada pengujian jumlah data latih berbeda dengan jumlah data uji tetap didapatkan nilai akurasi yang berbeda-beda. Hasil akurasi pada saat jumlah data latih sebanyak 30 data sebesar 92%. Sedangkan hasil akurasi pada saat jumlah data latih sebanyak 60 dan 90 data sebesar 96% terjadi peningkatan sebesar 4%

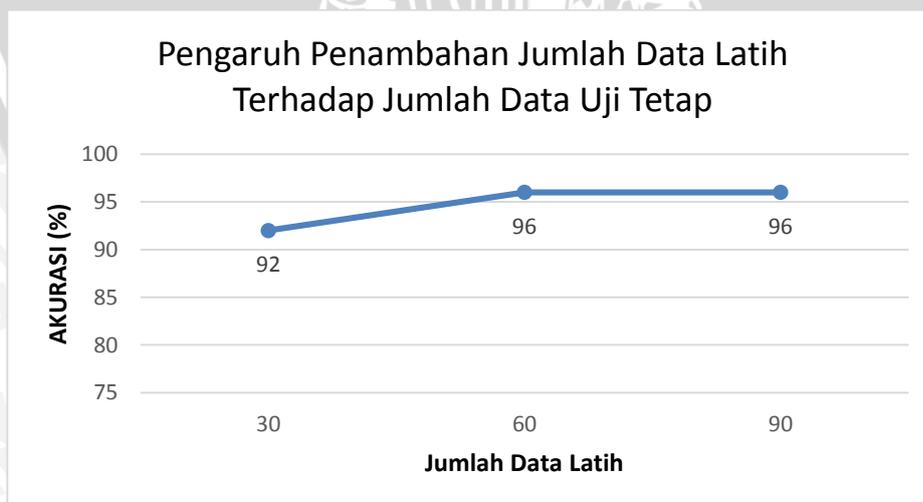
dari pengujian dengan 30 data latih. Hasil akurasi terbaik sebesar 96% pada saat jumlah data latih sebanyak 60 data dan 90 data. Hasil pengujian ditunjukkan pada Tabel 6.17.

Tabel 6.2 Hasil Pengujian Pada Jumlah Data Latih Berbeda Dengan Jumlah Data Uji Tetap

Nilai k	Jumlah Data Uji	Jumlah Data Latih	Akurasi (%)
1	50	30	92%
		60	96%
		90	96%

Penyebab perbedaan akurasi yang dihasilkan pada Tabel 6.17 karena semakin banyak data latih yang digunakan maka semakin banyak data yang dibandingkan dan hal ini juga akan berpengaruh terhadap akurasi. Dalam pengujian ini data latih yang digunakan merupakan data latih dengan kelas seimbang (*balanced class*) sehingga tidak ada kelas yang mendominasi.

Berdasarkan hasil uji coba pada masing-masing kasus, penambahan jumlah data latih memiliki pengaruh terhadap hasil akurasi yang dihasilkan sistem terlebih jika data latih yang digunakan dengan kelas seimbang. Pengaruh penambahan data latih terlihat ketika jumlah data latih semakin besar maka nilai akurasi yang dihasilkan juga meningkat. Hal ini disebabkan oleh semakin banyak data latih yang digunakan maka semakin banyak data yang dibandingkan dan semakin banyak juga jarak yang mendekati kelas data uji yang akan diklasifikasi. Dalam pengujian ini data latih yang digunakan merupakan data latih dengan kelas seimbang artinya tidak ada kelas yang lebih mendominasi, karena pada data latih tidak seimbang akan menimbulkan *noise* dalam proses klasifikasi. Grafik hasil pengujian jumlah data latih berbeda dengan jumlah data uji tetap ditunjukkan pada Gambar 6.2.



Gambar 6.2 Grafik penambahan jumlah data latih terhadap hasil akurasi

6.3 Pengujian Pengaruh Data Latih Seimbang dan Data Latih Tidak Seimbang Terhadap Akurasi

Pengujian ketiga yang dilakukan yaitu pengujian pengaruh data latih seimbang dan data latih tidak seimbang terhadap akurasi. Pada pengujian ini data latih seimbang dan data latih tidak seimbang yang digunakan terdapat tiga macam yaitu 30 data, 60 data dan 90 data. Nilai k yang digunakan untuk pengujian ini yaitu k=1 diaman merupakan nilai k terbaik pada hasil pengujian pertama. Data uji yang digunakan berjumlah 50 data yang belum digunakan dan diambil secara random. Pada data latih seimbang memiliki jumlah data yang sama untuk setiap kelasnya, sedangkan untuk data latih tidak seimbang memiliki jumlah data yang tidak sama untuk setiap kelasnya karena diambil secara random.

6.3.1 Skenario Pengujian Pengaruh Data Latih Seimbang dan Data Latih Tidak Seimbang Terhadap Akurasi

Prosedur pengujian pengaruh data latih seimbang dan tidak seimbang dilakukan dengan cara menghitung akurasi dari data uji yang diinputkan pada sistem. Data uji yang diinputkan berjumlah 50 data yang diambil secara random dan diambil data yang belum digunakan. Uji coba ini dilakukan sebanyak tiga kali dengan jumlah data latih seimbang yang berbeda dan juga dilakukan sebanyak tiga kali uji coba dengan data latih tidak seimbang yang berbeda pula.

6.3.2 Analisis Hasil Skenario Pengujian Pengaruh Data Latih Seimbang dan Tidak Seimbang Terhadap Akurasi

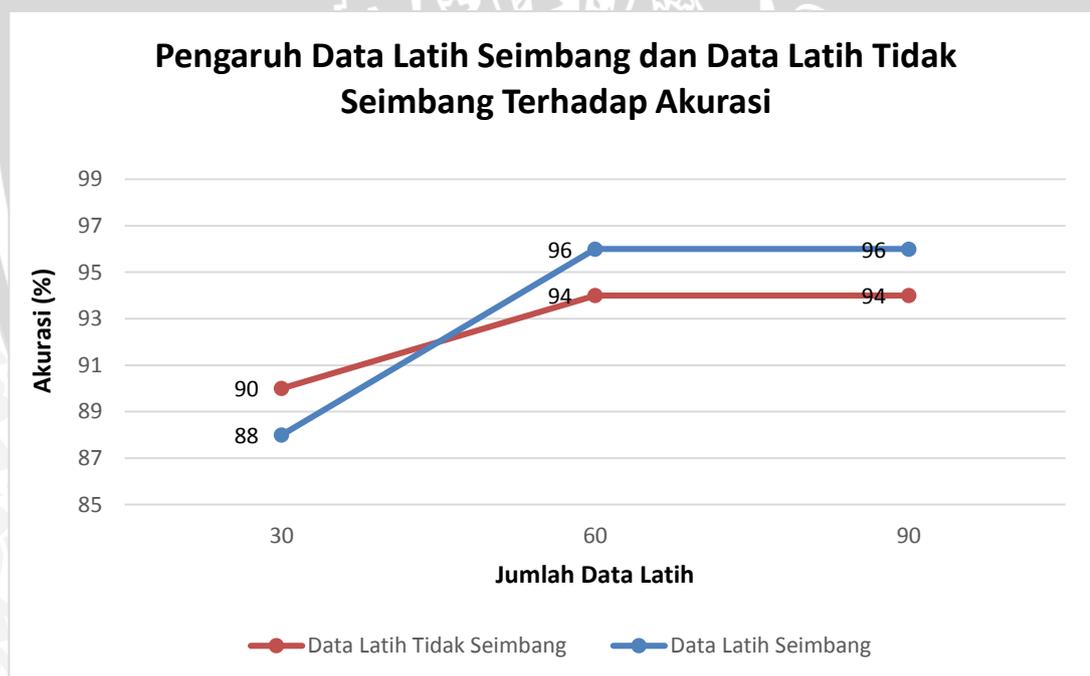
Pada pengujian pengaruh data latih seimbang dan data latih tidak seimbang terhadap akurasi yang telah dilakukan, didapatkan nilai akurasi yang berbeda-beda. Nilai akurasi tertinggi pada data latih seimbang sebesar 96% pada saat jumlah data latih 60 dan 90 data, sedangkan nilai akurasi rata-rata terendah yang dihasilkan sebesar 88% pada saat jumlah data latih 30 data. Pada data latih tidak seimbang akurasi tertinggi sebesar 94% dihasilkan pada saat jumlah data latih 60 dan 90 data, sedangkan akurasi terendah sebesar 90% pada saat jumlah data latih 30 data. Hasil pengujian pengaruh data latih seimbang dan data latih tidak seimbang ditunjukkan pada Tabel 6.18.

Tabel 6.3 Hasil pengujian pengaruh data latih seimbang dan tidak seimbang

Jenis Data Latih	Jumlah Data Latih	Jumlah Data Uji	Akurasi (%)
Seimbang	30	50	88
	60		96
	90		96
Tidak Seimbang	30	50	90
	60		94
	90		94

Pada Tabel 6.18 data latih yang digunakan pada jenis data latih seimbang diambil sejumlah data pada setiap kelas dengan jumlah yang sama, sehingga masing-masing kelas memiliki jumlah data latih yang sama. Misalnya, pada data latih seimbang dengan jumlah data 30 maka didalamnya terdapat 5 data kelas busuk daun, 5 data kelas bercak coklat, 5 data kelas layu fusarium, 5 data kelas layu bakteri, 5 data kelas gejala mosaik dan 5 data kelas penyakit keriting. Sedangkan untuk data latih tidak seimbang diambil sejumlah data secara acak, jadi setiap kelas memiliki jumlah data yang berbeda.

Berdasarkan hasil uji coba yang telah dilakukan, pada data latih kelas seimbang dan tidak seimbang mengalami peningkatan akurasi seiring dengan bertambahnya jumlah data latih yang digunakan. Namun, akurasi yang dihasilkan oleh data latih seimbang lebih tinggi daripada akurasi yang dihasilkan oleh data latih tidak seimbang. Hal ini disebabkan karena pada data latih tidak seimbang terjadi dominasi pada kelas tertentu sehingga menimbulkan banyak data yang tidak relevan (*noise*). *Noise* yang muncul pada data latih tidak seimbang menyebabkan terjadinya kesalahan dalam proses klasifikasi dan data yang diklasifikasi cenderung mengacu pada kelas yang mendominasi tersebut. Grafik hasil pengujian pengaruh data latih seimbang dan tidak seimbang terhadap akurasi ditunjukkan pada Gambar 6.3.



Gambar 6.3 Grafik pengaruh data latih seimbang dan tidak seimbang terhadap akurasi

BAB 7 PENUTUP

Bab ini membahas tentang kesimpulan dan saran dari penelitian yang telah dilakukan berdasarkan hasil perancangan, implementasi dan pengujian sistem pada permasalahan diagnosis penyakit tanaman tomat menggunakan metode *Modified K-Nearest Neighbor*.

7.1 Kesimpulan

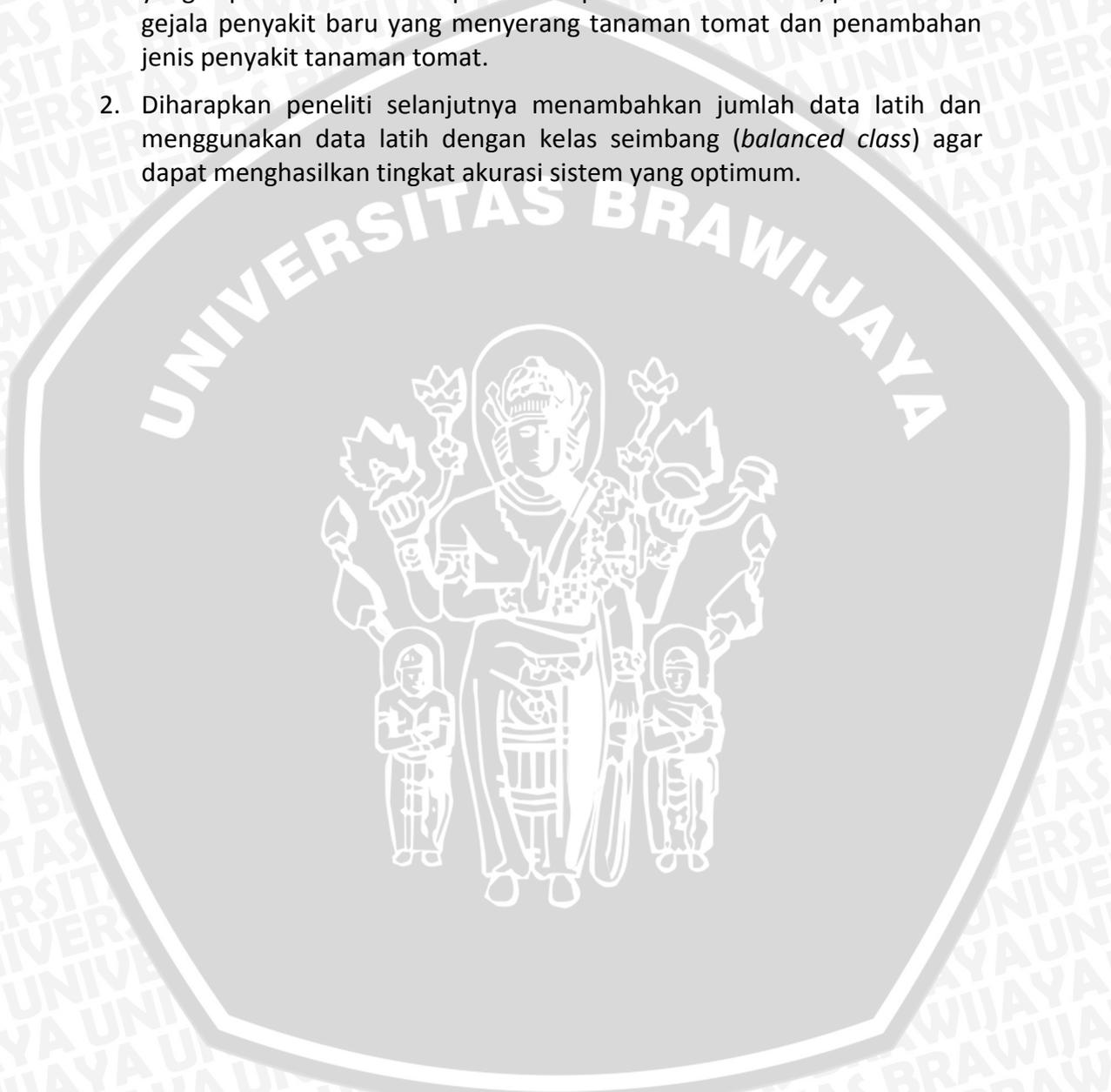
Berdasarkan pada hasil perancangan, implementasi dan pengujian yang telah dilakukan pada permasalahan diagnosis penyakit tanaman tomat menggunakan metode *Modified K-Nearest Neighbor*, terdapat beberapa kesimpulan yaitu:

1. Metode *Modified K-Nearest Neighbor* dapat diimplementasikan untuk mendiagnosis penyakit tanaman tomat dengan menggunakan 15 parameter yaitu gejala-gejala penyakit yang menyerang tanaman tomat dengan 6 jenis penyakit yaitu busuk daun, bercak coklat, layu fusarium, layu bakteri, gejala mosaik dan penyakit keriting.
2. Rata-rata akurasi maksimum yang dihasilkan pada penelitian ini sebesar 98,92%, sedangkan rata-rata akurasi minimum sebesar 89,04%. Tingkat akurasi yang dihasilkan oleh metode *Modified K-Nearest Neighbor* dipengaruhi oleh beberapa parameter sebagai berikut :
 - a. Penambahan atau pengurangan pada nilai k berpengaruh pada hasil akurasi yang dihasilkan sistem. Secara umum rata-rata nilai akurasi sistem semakin menurun sejalan dengan adanya penambahan nilai k , hal ini disebabkan semakin banyaknya data yang digunakan untuk proses klasifikasi terlebih jika terdapat kelas data tertentu yang mendominasi pada pemilihan data tetangga terdekat sejumlah k . Nilai k terbaik pada metode ini tergantung pada data yang digunakan.
 - b. Peningkatan jumlah data latih turut disertai dengan peningkatan nilai akurasi terutama pada kelas seimbang. Hal tersebut dikarenakan semakin banyak data latih, maka semakin banyak data referensi yang mendekati kelas data prediksi.
 - c. Pada data latih tidak seimbang dan data latih seimbang mengalami peningkatan akurasi seiring dengan penambahan jumlah data latih. Namun, pada data latih seimbang menghasilkan akurasi yang lebih besar daripada data latih tidak seimbang. Hal ini terjadi karena pada data latih kelas tidak seimbang menimbulkan *noise* sehingga klasifikasi cenderung mengacu pada kelas yang mendominasi dalam data tersebut.

7.2 Saran

Saran yang diberikan untuk pengembangan sistem dalam penelitian selanjutnya adalah :

1. Sistem dapat dikembangkan oleh peneliti selanjutnya dengan menambahkan parameter untuk meningkatkan tingkat akurasi. Parameter yang dapat ditambahkan seperti hama pada tanaman tomat, penambahan gejala penyakit baru yang menyerang tanaman tomat dan penambahan jenis penyakit tanaman tomat.
2. Diharapkan peneliti selanjutnya menambahkan jumlah data latih dan menggunakan data latih dengan kelas seimbang (*balanced class*) agar dapat menghasilkan tingkat akurasi sistem yang optimum.



DAFTAR PUSTAKA

- Aktivia, R., 2012. *Pengenalan Iris Mata Menggunakan Algoritma K-Nearest Neighbor Berbasis Histogram*. S1. Institut Pertanian Bogor.
- Han, J. & Kamber, M., 2006. *Data Mining : Concepts and Techniques*. 2nd ed. San Fransisco: Morgan Kauffman Publisher.
- Hanum, C., 2008. *Teknik Budidaya Tanaman, jilid 2*. Jakarta: Direktorat Pembinaan Sekolah Menengah Kejuruan, Departemen Pendidikan Nasional.
- Kantardzic, M., 2003. *Data Mining: Concepts, Models, Methods and Algorithm*. New York: John Wiley & Sons.
- Kementerian Pertanian, 2015. *Laporan Produktivitas Tomat tahun 2015*. [pdf] Kementerian Pertanian. Tersedia di: <
<http://www.pertanian.go.id/ATAP2014-HORTI-pdf/307-Prodvtv-Tomat.pdf>> [Diakses 1 September 2015].
- Kumalasari, N., Marji, & Dewi, C., 2014. *Implementasi Agoritma Modified K-Nearest Neighbor(MKNN) Untuk Menentukan Tingkat Resiko Penyakit Lemak Darah (Profil Lipid)*. S1. Universitas Brawijaya.
- Kusnawi. 2007. *Pengantar Solusi Data Mining*. Seminar Nasional Teknologi 2007 (SNT 2007), Issue ISSN : 1978 – 9777
- Larose, D. T., 2005. *Discovering Knowledge in Data*. New Jersey: John Wiley & Sons, Inc.
- Moertini, V. S., 2002. *Data mining Sebagai Solusi Bisnis. Integral, Volume 7*.
- Parvin, H., Alizadeh, H. & Minati, B., 2010. *A Modification on K-Nearest Neighbor Classifier*. Global Journal of Computer Science and Technology, Vol.10(Issue 14).
- Pramudiono, I., 2003. *Pengantar Data mining: Menambang Permata Pengetahuan di Gunung Data*. IlmuKomputer.com.
- Semangun, H., 1989. *Penyakit-penyakit Tanaman Hortikultura di Indonesia*. Yogyakarta: Gadjah Mada University Press.
- Setiawati, W., Sulastrini, I. & Gunaeni, N., 2001. *Penerapan Teknologi PHT Pada Tanaman Tomat*. Monografi, Volume 23.
- Sulistiyandari, 2012. *Penerapan Algoritma Modified K-Nearest Neighbor(MKNN) untuk Mengklasifikasikan Letak Protein pada Bakteri Pengaruh Pb2+ terhadap Aktivitas Pektinase dari Bacillus firmusri E-Coli*. S1. Universitas Brawijaya.

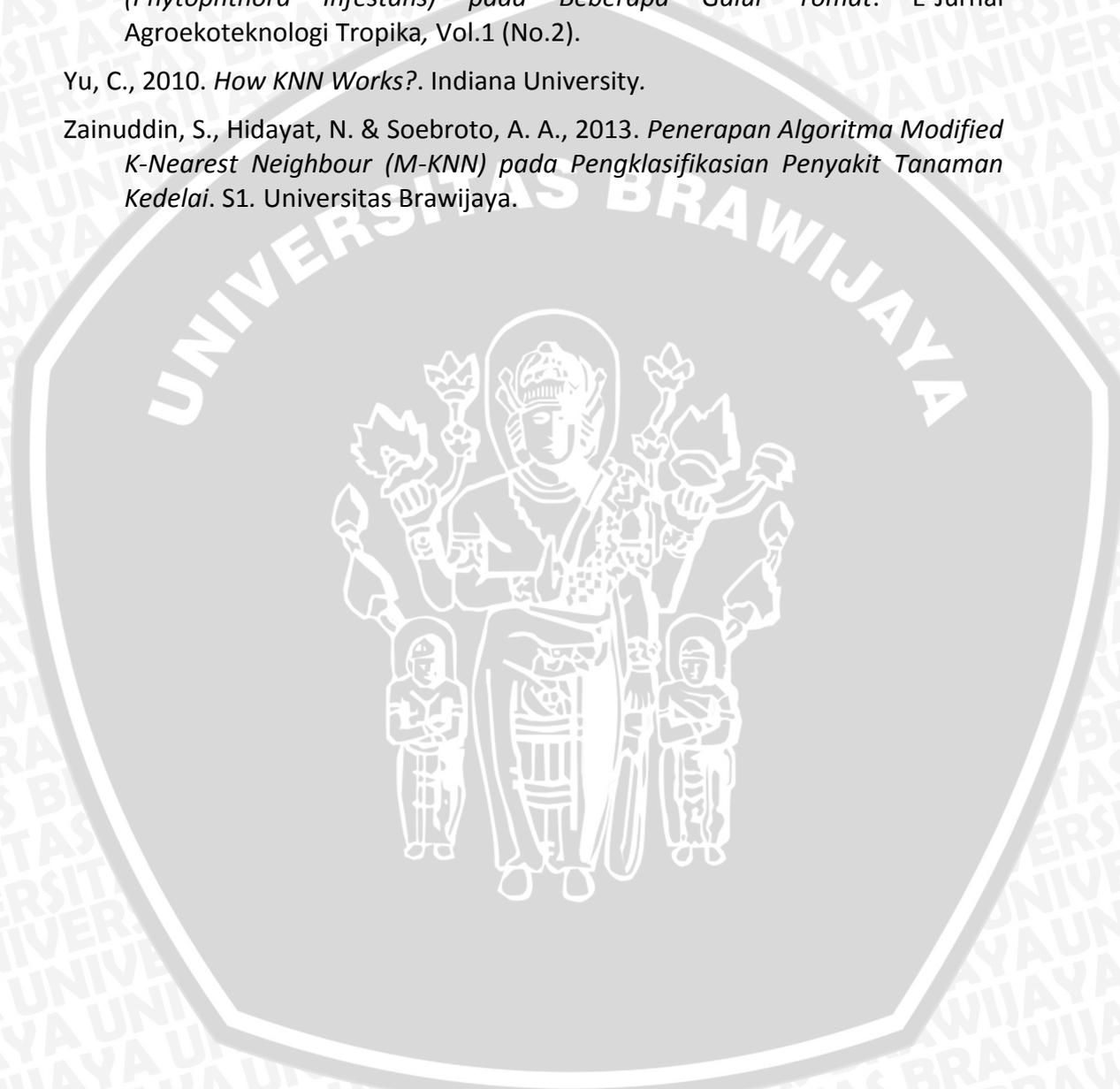
Supriyanto, Solahudin, M. & Rizal, S., 2013. *Sistem pakar Diagnosis dan penanggulangan hama dan penyakit tomat buah (Solanum lycopersicum) dataran tinggi berbasis android*. Seminar Nasional HIPI.

Wiriyanta, B. T. W., 2002. *Bertanam Tomat*. [e-book] Jakarta: AgroMedia Pustaka. Tersedia di: Google Books <<http://booksgoogle.com>> [Diakses 15 Maret 2015]

Yasa, I. N. D. et al., 2012. *Kajian Ketahanan Terhadap Penyakit Busuk Daun (Phytophthora Infestans) pada Beberapa Galur Tomat*. E-Jurnal Agroekoteknologi Tropika, Vol.1 (No.2).

Yu, C., 2010. *How KNN Works?*. Indiana University.

Zainuddin, S., Hidayat, N. & Soebroto, A. A., 2013. *Penerapan Algoritma Modified K-Nearest Neighbour (M-KNN) pada Pengklasifikasian Penyakit Tanaman Kedelai*. S1. Universitas Brawijaya.



LAMPIRAN A NILAI PEMBOBOTAN GEJALA PENYAKIT TOMAT

Lampiran Nilai Pembobotan Gejala Penyakit Tomat Dari Pakar

No.	Nama 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 kosentris	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

11	Pada buah terdapat bercak coklat tua, keras dan berkerut	tidak ada	0
		sedikit	10
		banyak	20
12	Pada buah terdapat bercak coklat gelap, mengendap, berlekuk dan terjadi busuk kering	tidak ada	0
		sedikit	10
		banyak	20
13	Buah sedikit dan kecil	tidak ada	0
		sedikit	10
		banyak	20
14	Tanaman layu	tidak ada	0
		sedikit	10
		banyak	20
15	Tanaman kerdil	tidak ada	0
		sedikit	10
		banyak	20

Keterangan Bobot:

- 0 = Tidak ada gejala
- 10 = Sedikit Gejala Kurang Khas
- 20 = Banyak Gejala Kurang Khas
- 45 = Sedikit Gejala Khas
- 90 = Banyak Gejala Khas

Malang,

Mengetahui,

Pakar Penyakit Tanaman


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

LAMPIRAN B HASIL WAWANCARA DENGAN PAKAR

Lampiran Hasil Wawancara

Tempat	:	Balai Pengembangan Teknologi Pertanian (BPTP) Jawa Timur
Tanggal	:	
Nama Narasumber	:	Prof. Dr. Ir. Moh. Cholil Mahfud, M.S.
Tujuan	:	Untuk mengetahui informasi mengenai penyakit pada tanaman tomat

Rangkuman Isi Wawancara:

No.	Pertanyaan	Jawaban
1.	Saya ingin mengambil topik mengenai penyakit pada tanaman sebagai obyek tugas akhir saya, dan mendapatkan saran untuk mencari informasi tersebut di BPTP Jawa Timur. Apakah disini menyediakan informasi tersebut?	Ya, informasi tersebut bisa didapatkan disini dan jika membutuhkan bantuan pakar dapat dikonsultasikan kepada saya.
2.	Untuk jenis tanamannya, kira-kira tanaman apa yang dapat saya gunakan sebagai bahan dalam tugas akhir saya?	Disini terdapat berbagai tanaman yang bisa anda pilih. Salah satu tanaman yang belum digunakan sebagai obyek penelitian sebelumnya adalah tomat. Anda bisa menggunakan tanaman tersebut sebagai obyeknya.
3.	Dalam pemeliharaan tanaman tomat, apakah terdapat permasalahan yang dihadapi oleh para petani?	Salah satu masalah yang dihadapi petani saat ini adalah kesulitan dalam mengidentifikasi dan menangani penyakit pada tanaman. Sebagian besar petani masih kurang memiliki pengetahuan mengenai penyakit pada tanaman.
4.	Saya berencana untuk membuat sebuah pemodelan sistem yang dapat mengidentifikasi penyakit pada tanaman tomat. Menurut anda, apakah hal tersebut dapat membantu para petani dalam mengatasi permasalahannya?	Hal tersebut tentunya sangat membantu para petani terutama bagi yang belum mendapatkan penyuluhan secara mendetail mengenai penyakit pada tanaman tomat.



No.	Pertanyaan	Jawaban
5.	Apakah disini saya bisa mendapatkan data mengenai gejala dan penanganan penyakit pada tanaman tomat?	Untuk gejala dan penanganan penyakit pada tanaman tomat bisa didapatkan melalui buku dan internet. Informasi tersebut kemudian anda rangkum dan dapat dikonsultasikan kepada saya. Jika membutuhkan bantuan dan informasi tambahan, bisa ditanyakan kepada saya dan akan saya bantu.

Malang

Mengetahui,

Pakar Penyakit Tanaman

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



LAMPIRAN C DAFTAR DATA LATIH

C.1 30 Data Latih

No.	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	diagnosis
1	0	90	0	0	0	0	0	0	0	0	20	0	0	0	0	1
2	0	0	0	45	0	0	10	0	0	0	0	20	0	0	0	2
3	0	0	0	0	10	0	0	0	45	0	0	0	0	20	0	3
4	0	0	0	0	20	0	0	0	0	45	0	0	20	10	0	4
5	0	0	90	0	0	0	0	20	0	0	0	0	20	0	20	5
6	0	0	0	0	0	90	0	0	0	0	0	0	20	0	20	6
7	0	45	0	0	0	0	0	20	0	0	0	0	0	0	0	1
8	0	0	0	90	20	0	0	0	0	0	0	10	0	0	0	2
9	0	0	0	0	10	0	0	0	90	0	0	0	20	0	0	3
10	0	0	0	0	0	0	0	0	0	45	0	0	10	10	0	4
11	0	0	90	0	0	0	0	0	0	0	0	0	10	0	0	5
12	0	0	0	0	0	45	0	0	0	0	0	0	0	0	0	6
13	10	45	0	0	0	0	0	10	0	0	10	0	0	0	0	1
14	0	0	0	90	10	0	20	0	0	0	0	10	0	0	0	2
15	0	0	0	0	20	0	0	0	45	0	0	0	10	0	0	3
16	0	0	0	0	10	0	0	0	0	90	0	0	10	20	0	4
17	0	0	45	0	0	0	0	0	0	0	0	0	0	0	10	5
18	0	0	0	0	0	45	0	0	0	0	0	0	20	0	20	6
19	0	90	0	0	0	0	0	10	0	0	10	0	0	0	0	1
20	0	0	0	45	10	0	20	0	0	0	0	0	0	0	0	2
21	0	0	0	0	10	0	0	0	45	0	0	0	0	20	0	3
22	0	0	0	0	20	0	0	0	0	45	0	0	0	0	0	4
23	0	0	90	0	0	0	0	10	0	0	0	0	20	0	0	5
24	0	0	0	0	0	45	0	0	0	0	0	0	20	0	10	6
25	20	45	0	0	0	0	0	10	0	0	20	0	0	0	0	1
26	0	0	0	45	10	0	20	0	0	0	0	20	0	0	0	2
27	0	0	0	0	20	0	0	0	45	0	0	0	20	20	0	3
28	0	0	0	0	0	0	0	0	0	45	0	0	0	0	0	4
29	0	0	45	0	0	0	0	0	0	0	0	0	20	0	10	5
30	0	0	0	0	0	45	0	0	0	0	0	0	10	0	0	6



C.2 60 Data Latih

No	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	diagnosis
1	0	90	0	0	0	0	0	0	0	0	20	0	0	0	0	1
2	0	0	0	45	0	0	10	0	0	0	0	20	0	0	0	2
3	0	0	0	0	10	0	0	0	45	0	0	0	0	20	0	3
4	0	0	0	0	20	0	0	0	0	45	0	0	20	10	0	4
5	0	0	90	0	0	0	0	20	0	0	0	0	20	0	20	5
6	0	0	0	0	0	90	0	0	0	0	0	0	20	0	20	6
7	0	45	0	0	0	0	0	20	0	0	0	0	0	0	0	1
8	0	0	0	90	20	0	0	0	0	0	0	10	0	0	0	2
9	0	0	0	0	10	0	0	0	90	0	0	0	20	0	0	3
10	0	0	0	0	0	0	0	0	0	45	0	0	10	10	0	4
11	0	0	90	0	0	0	0	0	0	0	0	0	10	0	0	5
12	0	0	0	0	0	45	0	0	0	0	0	0	0	0	0	6
13	10	45	0	0	0	0	0	10	0	0	10	0	0	0	0	1
14	0	0	0	90	10	0	20	0	0	0	0	10	0	0	0	2
15	0	0	0	0	20	0	0	0	45	0	0	0	10	0	0	3
16	0	0	0	0	10	0	0	0	0	90	0	0	10	20	0	4
17	0	0	45	0	0	0	0	0	0	0	0	0	0	0	10	5
18	0	0	0	0	0	45	0	0	0	0	0	0	20	0	20	6
19	0	90	0	0	0	0	0	10	0	0	10	0	0	0	0	1
20	0	0	0	45	10	0	20	0	0	0	0	0	0	0	0	2
21	0	0	0	0	10	0	0	0	45	0	0	0	0	20	0	3
22	0	0	0	0	20	0	0	0	0	45	0	0	0	0	0	4
23	0	0	90	0	0	0	0	10	0	0	0	0	20	0	0	5
24	0	0	0	0	0	45	0	0	0	0	0	0	20	0	10	6
25	20	45	0	0	0	0	0	10	0	0	20	0	0	0	0	1
26	0	0	0	45	10	0	20	0	0	0	0	20	0	0	0	2
27	0	0	0	0	20	0	0	0	45	0	0	0	20	20	0	3
28	0	0	0	0	0	0	0	0	0	45	0	0	0	0	0	4
29	0	0	45	0	0	0	0	0	0	0	0	0	20	0	10	5
30	0	0	0	0	0	45	0	0	0	0	0	0	10	0	0	6
31	10	90	0	0	0	0	0	0	0	0	20	0	0	0	0	1
32	0	0	0	45	20	0	10	0	0	0	0	10	0	0	0	2
33	0	0	0	0	20	0	0	0	45	0	0	0	0	0	0	3
34	0	0	0	0	20	0	0	0	0	90	0	0	20	20	0	4
35	0	0	90	0	0	0	0	20	0	0	0	0	20	0	0	5
36	0	0	0	0	0	90	0	0	0	0	0	0	0	0	10	6
37	0	45	0	0	0	0	0	0	0	0	10	0	0	0	0	1
38	0	0	0	90	10	0	10	0	0	0	0	20	0	0	0	2
39	0	0	0	0	10	0	0	0	45	0	0	0	20	0	0	3

40	0	0	0	0	0	0	0	0	0	90	0	0	10	0	0	4
41	0	0	45	0	0	0	0	0	0	0	0	0	0	0	20	5
42	0	0	0	0	0	45	0	0	0	0	0	0	20	0	0	6
43	0	45	0	0	0	0	0	20	0	0	10	0	0	0	0	1
44	0	0	0	45	0	0	0	0	0	0	0	20	0	0	0	2
45	0	0	0	0	0	0	0	0	45	0	0	0	10	20	0	3
46	0	0	0	0	10	0	0	0	0	90	0	0	10	10	0	4
47	0	0	90	0	0	0	0	0	0	0	0	0	0	0	10	5
48	0	0	0	0	0	90	0	0	0	0	0	0	20	0	0	6
49	20	90	0	0	0	0	0	10	0	0	10	0	0	0	0	1
50	0	0	0	45	10	0	10	0	0	0	0	0	0	0	0	2
51	0	0	0	0	0	0	0	0	90	0	0	0	10	0	0	3
52	0	0	0	0	10	0	0	0	0	45	0	0	0	20	0	4
53	0	0	45	0	0	0	0	10	0	0	0	0	20	0	0	5
54	0	0	0	0	0	45	0	0	0	0	0	0	10	0	20	6
55	20	90	0	0	0	0	0	10	0	0	10	0	0	0	0	1
56	0	0	0	45	10	0	10	0	0	0	0	20	0	0	0	2
57	0	0	0	0	20	0	0	0	90	0	0	0	20	0	0	3
58	0	0	0	0	10	0	0	0	0	90	0	0	20	10	0	4
59	0	0	90	0	0	0	0	20	0	0	0	0	0	0	0	5
60	0	0	0	0	0	45	0	0	0	0	0	0	10	0	10	6

C.3 90 Data Latih

No	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	diagnosis
1	0	90	0	0	0	0	0	0	0	0	20	0	0	0	0	1
2	0	0	0	45	0	0	10	0	0	0	0	20	0	0	0	2
3	0	0	0	0	10	0	0	0	45	0	0	0	0	20	0	3
4	0	0	0	0	20	0	0	0	0	45	0	0	20	10	0	4
5	0	0	90	0	0	0	0	20	0	0	0	0	20	0	20	5
6	0	0	0	0	0	90	0	0	0	0	0	0	20	0	20	6
7	0	45	0	0	0	0	0	20	0	0	0	0	0	0	0	1
8	0	0	0	90	20	0	0	0	0	0	0	10	0	0	0	2
9	0	0	0	0	10	0	0	0	90	0	0	0	20	0	0	3
10	0	0	0	0	0	0	0	0	0	45	0	0	10	10	0	4
11	0	0	90	0	0	0	0	0	0	0	0	0	10	0	0	5
12	0	0	0	0	0	45	0	0	0	0	0	0	0	0	0	6
13	10	45	0	0	0	0	0	10	0	0	10	0	0	0	0	1
14	0	0	0	90	10	0	20	0	0	0	0	10	0	0	0	2
15	0	0	0	0	20	0	0	0	45	0	0	0	10	0	0	3
16	0	0	0	0	10	0	0	0	0	90	0	0	10	20	0	4
17	0	0	45	0	0	0	0	0	0	0	0	0	0	0	10	5



18	0	0	0	0	0	45	0	0	0	0	0	0	20	0	20	6
19	0	90	0	0	0	0	0	10	0	0	10	0	0	0	0	1
20	0	0	0	45	10	0	20	0	0	0	0	0	0	0	0	2
21	0	0	0	0	10	0	0	0	45	0	0	0	0	20	0	3
22	0	0	0	0	20	0	0	0	0	45	0	0	0	0	0	4
23	0	0	90	0	0	0	0	10	0	0	0	0	20	0	0	5
24	0	0	0	0	0	45	0	0	0	0	0	0	20	0	10	6
25	20	45	0	0	0	0	0	10	0	0	20	0	0	0	0	1
26	0	0	0	45	10	0	20	0	0	0	0	20	0	0	0	2
27	0	0	0	0	20	0	0	0	45	0	0	0	20	20	0	3
28	0	0	0	0	0	0	0	0	0	45	0	0	0	0	0	4
29	0	0	45	0	0	0	0	0	0	0	0	0	20	0	10	5
30	0	0	0	0	0	45	0	0	0	0	0	0	10	0	0	6
31	10	90	0	0	0	0	0	0	0	0	20	0	0	0	0	1
32	0	0	0	45	20	0	10	0	0	0	0	10	0	0	0	2
33	0	0	0	0	20	0	0	0	45	0	0	0	0	0	0	3
34	0	0	0	0	20	0	0	0	0	90	0	0	20	20	0	4
35	0	0	90	0	0	0	0	20	0	0	0	0	20	0	0	5
36	0	0	0	0	0	90	0	0	0	0	0	0	0	0	10	6
37	0	45	0	0	0	0	0	0	0	0	10	0	0	0	0	1
38	0	0	0	90	10	0	10	0	0	0	0	20	0	0	0	2
39	0	0	0	0	10	0	0	0	45	0	0	0	20	0	0	3
40	0	0	0	0	0	0	0	0	0	90	0	0	10	0	0	4
41	0	0	45	0	0	0	0	0	0	0	0	0	0	0	20	5
42	0	0	0	0	0	45	0	0	0	0	0	0	20	0	0	6
43	0	45	0	0	0	0	0	20	0	0	10	0	0	0	0	1
44	0	0	0	45	0	0	0	0	0	0	0	20	0	0	0	2
45	0	0	0	0	0	0	0	0	45	0	0	0	10	20	0	3
46	0	0	0	0	10	0	0	0	0	90	0	0	10	10	0	4
47	0	0	90	0	0	0	0	0	0	0	0	0	0	0	10	5
48	0	0	0	0	0	90	0	0	0	0	0	0	20	0	0	6
49	20	90	0	0	0	0	0	10	0	0	10	0	0	0	0	1
50	0	0	0	45	10	0	10	0	0	0	0	0	0	0	0	2
51	0	0	0	0	0	0	0	0	90	0	0	0	10	0	0	3
52	0	0	0	0	10	0	0	0	0	45	0	0	0	20	0	4
53	0	0	45	0	0	0	0	10	0	0	0	0	20	0	0	5
54	0	0	0	0	0	45	0	0	0	0	0	0	10	0	20	6
55	20	90	0	0	0	0	0	10	0	0	10	0	0	0	0	1
56	0	0	0	45	10	0	10	0	0	0	0	20	0	0	0	2
57	0	0	0	0	20	0	0	0	90	0	0	0	20	0	0	3
58	0	0	0	0	10	0	0	0	0	90	0	0	20	10	0	4
59	0	0	90	0	0	0	0	20	0	0	0	0	0	0	0	5



60	0	0	0	0	0	45	0	0	0	0	0	0	10	0	10	6
61	0	90	0	0	0	0	0	20	0	0	20	0	0	0	0	1
62	0	0	0	45	10	0	20	0	0	0	0	10	0	0	0	2
63	0	0	0	0	20	0	0	0	90	0	0	0	0	10	0	3
64	0	0	0	0	0	0	0	0	0	45	0	0	20	20	0	4
65	0	0	90	0	0	0	0	20	0	0	0	0	0	0	0	5
66	0	0	0	0	0	45	0	0	0	0	0	0	0	0	10	6
67	10	90	0	0	0	0	0	10	0	0	20	0	0	0	0	1
68	0	0	0	90	20	0	10	0	0	0	0	0	0	0	0	2
69	0	0	0	0	0	0	0	0	45	0	0	0	20	20	0	3
70	0	0	0	0	20	0	0	0	0	45	0	0	10	10	0	4
71	0	0	90	0	0	0	0	20	0	0	0	0	0	0	0	5
72	0	0	0	0	0	90	0	0	0	0	0	0	10	0	20	6
73	10	45	0	0	0	0	0	10	0	0	20	0	0	0	0	1
74	0	0	0	90	0	0	10	0	0	0	0	20	0	0	0	2
75	0	0	0	0	10	0	0	0	90	0	0	0	0	10	0	3
76	0	0	0	0	0	0	0	0	0	90	0	0	20	10	0	4
77	0	0	90	0	0	0	0	10	0	0	0	0	0	0	0	5
78	0	0	0	0	0	90	0	0	0	0	0	0	0	0	20	6
79	20	45	0	0	0	0	0	10	0	0	0	0	0	0	0	1
80	0	0	0	90	10	0	0	0	0	0	0	10	0	0	0	2
81	0	0	0	0	20	0	0	0	45	0	0	0	20	10	0	3
82	0	0	0	0	0	0	0	0	0	45	0	0	0	10	0	4
83	0	0	45	0	0	0	0	0	0	0	0	0	20	0	20	5
84	0	0	0	0	0	90	0	0	0	0	0	0	0	0	0	6
85	0	45	0	0	0	0	0	10	0	0	10	0	0	0	0	1
86	0	0	0	90	10	0	10	0	0	0	0	0	0	0	0	2
87	0	0	0	0	0	0	0	0	90	0	0	0	10	20	0	3
88	0	0	0	0	10	0	0	0	0	90	0	0	0	0	0	4
89	0	0	45	0	0	0	0	0	0	0	0	0	10	0	10	5
90	0	0	0	0	0	90	0	0	0	0	0	0	10	0	0	6



LAMPIRAN D DAFTAR DATA UJI

D.1 50 Data uji untuk pengujian data uji tetap dan data latih berbeda

No	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	diagnosis
1	10	90	0	0	0	0	0	0	0	0	10	0	0	0	0	1
2	0	0	0	0	10	0	0	0	90	0	0	0	0	0	0	3
3	0	0	90	0	0	0	0	0	0	0	0	0	20	0	20	5
4	0	0	0	90	20	0	10	0	0	0	0	20	0	0	0	2
5	0	0	45	0	0	0	0	10	0	0	0	0	0	0	10	5
6	0	0	0	0	0	0	0	0	45	0	0	0	10	0	0	3
7	0	0	90	0	0	0	0	20	0	0	0	0	10	0	0	5
8	0	0	0	0	20	0	0	0	0	45	0	0	20	20	0	4
9	0	0	0	0	0	0	0	0	0	45	0	0	20	10	0	4
10	0	0	90	0	0	0	0	20	0	0	0	0	0	0	10	5
11	0	0	0	0	20	0	0	0	45	0	0	0	20	0	0	3
12	0	0	0	45	20	0	20	0	0	0	0	0	0	0	0	2
13	0	0	0	0	20	0	0	0	90	0	0	0	20	20	0	3
14	0	0	0	90	20	0	20	0	0	0	0	10	0	0	0	2
15	10	45	0	0	0	0	0	0	0	0	20	0	0	0	0	1
16	0	0	0	0	0	0	0	0	0	45	0	0	10	20	0	4
17	0	0	0	0	20	0	0	0	0	90	0	0	20	10	0	4
18	0	0	0	0	20	0	0	0	0	90	0	0	0	0	0	4
19	20	90	0	0	0	0	0	0	0	0	0	0	0	0	0	1
20	0	0	0	45	10	0	10	0	0	0	0	10	0	0	0	2
21	0	0	0	0	0	0	0	0	0	90	0	0	0	0	0	4
22	0	0	0	0	10	0	0	0	0	45	0	0	0	0	0	4
23	0	0	0	0	20	0	0	0	90	0	0	0	0	0	0	3
24	0	0	0	0	10	0	0	0	0	90	0	0	20	0	0	4
25	0	0	0	45	0	0	10	0	0	0	0	0	0	0	0	2
26	0	0	90	0	0	0	0	10	0	0	0	0	10	0	20	5
27	0	0	0	0	10	0	0	0	0	90	0	0	20	20	0	4
28	0	0	90	0	0	0	0	0	0	0	0	0	0	0	0	5
29	0	0	45	0	0	0	0	20	0	0	0	0	10	0	0	5
30	0	0	0	0	20	0	0	0	0	90	0	0	10	10	0	4
31	20	45	0	0	0	0	0	0	0	0	20	0	0	0	0	1
32	0	0	0	0	0	0	0	0	45	0	0	0	0	20	0	3
33	10	45	0	0	0	0	0	20	0	0	0	0	0	0	0	1
34	0	0	0	90	20	0	0	0	0	0	0	20	0	0	0	2
35	0	0	0	0	20	0	0	0	0	45	0	0	10	20	0	4
36	0	0	0	45	20	0	20	0	0	0	0	20	0	0	0	2

37	0	0	0	0	10	0	0	0	0	45	0	0	20	10	0	4
38	0	90	0	0	0	0	0	0	0	0	10	0	0	0	0	1
39	10	90	0	0	0	0	0	10	0	0	10	0	0	0	0	1
40	0	0	0	0	10	0	0	0	90	0	0	0	10	10	0	3
41	0	0	45	0	0	0	0	10	0	0	0	0	0	0	20	5
42	10	45	0	0	0	0	0	20	0	0	10	0	0	0	0	1
43	0	0	0	45	0	0	10	0	0	0	0	10	0	0	0	2
44	0	45	0	0	0	0	0	10	0	0	0	0	0	0	0	1
45	0	0	45	0	0	0	0	10	0	0	0	0	20	0	20	5
46	0	0	45	0	0	0	0	20	0	0	0	0	10	0	20	5
47	0	0	0	0	0	0	0	0	90	0	0	0	20	10	0	3
48	0	0	0	0	10	0	0	0	0	45	0	0	10	0	0	4
49	0	0	90	0	0	0	0	20	0	0	0	0	10	0	10	5
50	0	90	0	0	0	0	0	20	0	0	10	0	0	0	0	1

D.2 50 data uji untuk pengujian data latih seimbang dan tidak seimbang

No	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	diagnosis
1	0	0	0	90	0	0	20	0	0	0	0	20	0	0	0	2
2	0	0	0	90	10	0	20	0	0	0	0	0	0	0	0	2
3	0	0	0	90	10	0	0	0	0	0	0	0	0	0	0	2
4	0	0	0	90	20	0	20	0	0	0	0	0	0	0	0	2
5	0	0	0	45	0	0	20	0	0	0	0	20	0	0	0	2
6	0	0	0	0	10	0	0	0	45	0	0	0	20	10	0	3
7	0	0	0	90	10	0	20	0	0	0	0	20	0	0	0	2
8	0	0	0	0	0	0	0	0	90	0	0	0	0	0	0	3
9	10	45	0	0	0	0	0	20	0	0	20	0	0	0	0	1
10	0	0	0	0	20	0	0	0	0	90	0	0	10	20	0	4
11	0	0	0	0	20	0	0	0	0	90	0	0	0	20	0	4
12	0	0	0	0	0	0	0	0	45	0	0	0	0	10	0	3
13	0	0	45	0	0	0	0	0	0	0	0	0	20	0	0	5
14	0	0	0	0	0	90	0	0	0	0	0	0	20	0	10	6
15	0	0	0	0	0	0	0	0	0	90	0	0	20	0	0	4
16	20	90	0	0	0	0	0	20	0	0	10	0	0	0	0	1
17	20	45	0	0	0	0	0	20	0	0	10	0	0	0	0	1
18	0	0	0	0	10	0	0	0	90	0	0	0	20	20	0	3
19	20	45	0	0	0	0	0	20	0	0	20	0	0	0	0	1
20	0	0	0	45	0	0	20	0	0	0	0	10	0	0	0	2
21	20	90	0	0	0	0	0	0	0	0	10	0	0	0	0	1
22	10	45	0	0	0	0	0	0	0	0	10	0	0	0	0	1



23	0	0	0	0	0	45	0	0	0	0	0	0	0	0	20	6
24	0	0	0	90	10	0	10	0	0	0	0	10	0	0	0	2
25	0	0	0	0	0	0	0	0	0	90	0	0	10	20	0	4
26	0	45	0	0	0	0	0	10	0	0	20	0	0	0	0	1
27	0	90	0	0	0	0	0	20	0	0	0	0	0	0	0	1
28	0	0	0	0	20	0	0	0	0	45	0	0	0	20	0	4
29	0	0	0	0	20	0	0	0	45	0	0	0	0	20	0	3
30	0	0	0	0	10	0	0	0	45	0	0	0	10	0	0	3
31	0	0	0	0	10	0	0	0	0	45	0	0	0	10	0	4
32	0	0	0	0	10	0	0	0	0	90	0	0	0	20	0	4
33	0	0	0	0	20	0	0	0	0	45	0	0	0	10	0	4
34	0	0	45	0	0	0	0	10	0	0	0	0	10	0	10	5
35	0	0	0	0	10	0	0	0	0	90	0	0	10	0	0	4
36	0	0	0	0	10	0	0	0	45	0	0	0	10	10	0	3
37	0	0	0	90	0	0	20	0	0	0	0	0	0	0	0	2
38	0	0	45	0	0	0	0	20	0	0	0	0	20	0	20	5
39	0	0	90	0	0	0	0	10	0	0	0	0	10	0	10	5
40	10	90	0	0	0	0	0	0	0	0	0	0	0	0	0	1
41	0	0	45	0	0	0	0	10	0	0	0	0	20	0	10	5
42	0	45	0	0	0	0	0	0	0	0	0	0	0	0	0	1
43	0	0	45	0	0	0	0	10	0	0	0	0	0	0	0	5
44	0	0	0	0	0	0	0	0	45	0	0	0	0	0	0	3
45	0	0	0	90	0	0	0	0	0	0	0	10	0	0	0	2
46	0	0	0	0	10	0	0	0	45	0	0	0	10	20	0	3
47	0	0	0	45	0	0	0	0	0	0	0	0	0	0	0	2
48	0	0	0	0	0	0	0	0	0	90	0	0	0	20	0	4
49	0	0	0	0	10	0	0	0	0	45	0	0	10	20	0	4
50	0	0	0	90	0	0	10	0	0	0	0	10	0	0	0	2



LAMPIRAN E PERHITUNGAN MANUAL TANPA NORMALISASI

Perhitungan manual berfungsi untuk memberikan gambaran umum perancangan sistem yang dibangun. Perhitungan manual tanpa normalisasi ini digunakan untuk membandingkan hasil antara perhitungan yang menggunakan normalisasi pada perancangan.

Pada perhitungan manual ini, digunakan 24 dataset dimana dibagi menjadi 2 bagian yaitu 18 data latih dan 6 data uji. Dataset yang digunakan dapat dilihat pada Tabel 4.8 berikut.

Tabel 1.1 Dataset perhitungan manual

No.	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	kelas
1	20	90	0	0	0	0	0	20	0	0	20	0	0	0	0	1
2	10	90	0	0	0	0	0	20	0	0	20	0	0	0	0	1
3	20	45	0	0	0	0	0	20	0	0	10	0	0	0	0	1
4	0	0	0	90	20	0	20	0	0	0	0	20	0	0	0	2
5	0	0	0	90	10	0	10	0	0	0	0	10	0	0	0	2
6	0	0	0	45	20	0	10	0	0	0	0	10	0	0	0	2
7	0	0	0	0	20	0	0	0	90	0	0	0	20	20	0	3
8	0	0	0	0	10	0	0	0	90	0	0	0	10	10	0	3
9	0	0	0	0	20	0	0	0	45	0	0	0	10	20	0	3
10	0	0	0	0	20	0	0	0	0	90	0	0	20	20	0	4
11	0	0	0	0	10	0	0	0	0	90	0	0	20	10	0	4
12	0	0	0	0	20	0	0	0	0	45	0	0	0	0	0	4
13	0	0	90	0	0	0	0	20	0	0	0	0	20	0	20	5
14	0	0	90	0	0	0	0	10	0	0	0	0	10	0	10	5
15	0	0	45	0	0	0	0	20	0	0	0	0	20	0	10	5
16	0	0	0	0	0	90	0	0	0	0	0	0	20	0	20	6
17	0	0	0	0	0	90	0	0	0	0	0	0	10	0	0	6
18	0	0	0	0	0	45	0	0	0	0	0	0	10	0	20	6
41	10	45	0	0	0	0	0	10	0	0	10	0	0	0	0	1
76	0	0	0	90	0	0	10	0	0	0	0	20	0	0	0	2
140	0	0	0	0	20	0	0	0	45	0	0	0	10	10	0	3
187	0	0	0	0	0	0	0	0	0	90	0	0	0	20	0	4
265	0	0	45	0	0	0	0	0	0	0	0	0	10	0	20	5
285	0	0	0	0	0	45	0	0	0	0	0	0	10	0	0	6

Dataset yang digunakan terdiri dari 15 gejala penyakit dan diagnosis dari tiap data. Pembagian data latih dan data uji dapat dilihat pada Tabel 1.2 dan Tabel 1.3 berikut.

Tabel 1.2 Data Latih

No.	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	kelas
1	20	90	0	0	0	0	0	20	0	0	20	0	0	0	0	1
2	10	90	0	0	0	0	0	20	0	0	20	0	0	0	0	1
3	20	45	0	0	0	0	0	20	0	0	10	0	0	0	0	1
4	0	0	0	90	20	0	20	0	0	0	0	20	0	0	0	2
5	0	0	0	90	10	0	10	0	0	0	0	10	0	0	0	2
6	0	0	0	45	20	0	10	0	0	0	0	10	0	0	0	2
7	0	0	0	0	20	0	0	0	90	0	0	0	20	20	0	3
8	0	0	0	0	10	0	0	0	90	0	0	0	10	10	0	3
9	0	0	0	0	20	0	0	0	45	0	0	0	10	20	0	3
10	0	0	0	0	20	0	0	0	0	90	0	0	20	20	0	4
11	0	0	0	0	10	0	0	0	0	90	0	0	20	10	0	4
12	0	0	0	0	20	0	0	0	0	45	0	0	0	0	0	4
13	0	0	90	0	0	0	0	20	0	0	0	0	20	0	20	5
14	0	0	90	0	0	0	0	10	0	0	0	0	10	0	10	5
15	0	0	45	0	0	0	0	20	0	0	0	0	20	0	10	5
16	0	0	0	0	0	90	0	0	0	0	0	0	20	0	20	6
17	0	0	0	0	0	90	0	0	0	0	0	0	10	0	0	6
18	0	0	0	0	0	45	0	0	0	0	0	0	10	0	20	6

Tabel 1.3 Data Uji

No.	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	kelas
41	10	45	0	0	0	0	0	10	0	0	10	0	0	0	0	1
76	0	0	0	90	0	0	10	0	0	0	0	20	0	0	0	2
140	0	0	0	0	20	0	0	0	45	0	0	0	10	10	0	3
187	0	0	0	0	0	0	0	0	0	90	0	0	0	20	0	4
265	0	0	45	0	0	0	0	0	0	0	0	0	10	0	20	5
285	0	0	0	0	0	45	0	0	0	0	0	0	10	0	0	6

Langkah 1. Menentukan nilai k atau tetangganya.

Pada perhitungan manual ini ditentukan nilai dari k adalah 3.

Langkah 2. Menghitung validitas data latih.

Langkah kedua yaitu menghitung nilai validitas data latih dengan menggunakan persamaan 2-3. Sebelum menghitung nilai validitas, terlebih dahulu mencari jarak data antar data training menggunakan perhitungan *Euclidean* dengan memasukkan data pada persamaan 2-1.

$$d_{(1,2)} = \sqrt{\sum_{i=1}^p (x_{2i} - x_{1i})^2}$$

$$= \sqrt{(10-20)^2 + (90-90)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (20-20)^2 + (0-0)^2 + (0-0)^2 + (1-1)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2}$$

$$= 10$$

Melakukan perhitungan yang sama untuk semua data latih. Hasil perhitungan *Euclidean* data latih ini dapat dilihat pada Tabel 1.4. Setelah ditentukan nilai k nya maka dihitung nilai validitas dari data latih yang memiliki jarak terdekat dengan persamaan 2-3.

$$validitas(x) = \frac{1}{k} \sum_{i=1}^k S(\text{label}(x), (\text{label}(N_i(x))))$$

$$= \frac{1}{3} \sum_{i=1}^3 S(\text{label}(x=1), (\text{label}(N_i(x=2))))$$

$$= \frac{1}{3} \times (1 + 1 + 0)$$

$$= \frac{2}{3} = 0.6667$$

Melakukan perhitungan yang sama untuk semua data latih. Hasil perhitungan validitas ini dapat dilihat pada Tabel 1.5.

Tabel 1.5 Perhitungan Validitas

No	k=1	k=2	k=3	Sum S(a,b)	Validitas
1	1	1	0	2	0.666667
2	1	1	0	2	0.666667
3	1	1	0	2	0.666667
4	1	1	0	2	0.666667
5	1	1	0	2	0.666667
6	1	1	0	2	0.666667
7	1	1	0	2	0.666667
8	0	1	1	2	0.666667
9	1	1	0	2	0.666667
10	0	1	1	2	0.666667
11	0	1	1	2	0.666667
12	0	1	1	2	0.666667
13	1	1	0	2	0.666667
14	1	1	0	2	0.666667

Tabel 1.4 Perhitungan Euclidean Data Latih

d(1)	d(2)	d(3)	d(4)	d(5)	d(6)	d(7)	d(8)	d(9)	d(10)	d(11)	d(12)	d(13)	d(14)	d(15)	d(16)	d(17)	d(18)	kelas
0	10	46.10	136.38	133.04	109.20	136.38	133.04	110.57	136.38	134.16	108.28	133.42	131.91	106.89	134.91	132.29	108.74	1
10	0	47.17	135.28	131.91	107.82	135.28	131.91	109.20	135.28	133.04	106.89	132.29	130.77	105.48	133.79	131.15	107.35	1
46.10	47.17	0	110.57	106.42	74.50	110.57	106.42	76.49	110.57	107.82	73.14	106.89	105.00	71.06	108.74	105.48	73.82	1
136.38	135.28	110.57	0	17.32	47.17	133.42	131.53	106.89	133.42	132.66	104.52	136.38	134.91	110.57	134.91	132.29	108.74	2
133.04	131.91	106.42	17.32	0	46.10	131.53	128.84	104.52	131.53	130.00	102.10	133.04	130.38	106.42	131.53	128.84	104.52	2
109.20	107.82	74.50	47.17	46.10	0	105.48	103.08	68.92	105.48	104.52	65.19	109.20	105.95	74.50	107.35	104.04	71.76	2
136.38	135.28	110.57	133.42	131.53	105.48	0	17.32	46.10	127.28	128.06	104.52	133.42	159.69	106.89	131.91	130.77	106.89	3
133.04	131.91	106.42	131.53	128.84	103.08	17.32	0	47.17	128.45	127.67	102.10	131.53	157.48	104.52	130.00	128.06	103.56	3
110.57	109.20	76.49	106.89	104.52	68.92	46.10	47.17	0	101.12	102.10	67.45	108.74	115.11	73.82	106.89	104.52	72.46	3
136.38	135.28	110.57	133.42	131.53	105.48	127.28	128.45	101.12	0	14.14	53.15	133.42	131.91	106.89	131.91	130.77	106.89	4
134.16	133.04	107.82	132.66	130.00	104.52	128.06	127.67	102.10	14.14	0	51.23	131.15	129.61	104.04	129.61	128.45	104.04	4
108.28	106.89	73.14	104.52	102.10	65.19	104.52	102.10	67.45	53.15	51.23	0	108.28	104.52	73.14	106.42	103.08	70.36	4
133.42	132.29	106.89	136.38	133.04	109.20	133.42	131.53	108.74	133.42	131.15	106.32	0	20	46.10	128.84	130.77	103.08	5
131.53	130.38	104.52	133.04	129.61	105.00	131.53	128.84	105.48	131.53	129.23	103.51	17.32	0	47.17	128.45	128.06	101.61	5
106.89	105.48	71.06	110.57	106.42	74.50	106.89	104.52	73.82	106.89	104.04	72.39	46.10	48.22	0	103.08	103.56	68.19	5
134.91	133.79	108.74	134.91	131.53	107.35	131.91	130.00	106.89	131.91	129.61	104.43	128.84	128.84	103.08	0	22.36	46.10	6
132.29	131.15	105.48	132.29	128.84	104.04	130.77	128.06	104.52	130.77	128.45	103.08	130.77	128.45	103.56	22.36	0	49.24	6
108.74	107.35	73.82	108.74	104.52	71.76	106.89	103.56	72.46	106.89	104.04	67.31	103.08	102.10	68.19	46.10	49.24	0	6

Keterangan:

 jarak *Euclidean* yang memiliki ketetanggaan terdekat dengan data

Melakukan perhitungan yang sama untuk semua data uji. Hasil perhitungan *Euclidean* ini dapat dilihat pada Tabel 1.6.

Tabel 1.6 Tabel Perhitungan Euclidean Data Uji 1

data	SUM euclidean	euclidean
d(41,1)	2325	48.21825
d(41,2)	2225	47.16991
d(41,3)	200	14.14214
d(41,4)	11625	107.8193
d(41,5)	10725	103.5616
d(41,6)	4950	70.35624
d(41,7)	11625	107.8193
d(41,8)	10725	103.5616
d(41,9)	5250	72.45688
d(41,10)	11625	107.8193
d(41,11)	11025	105
d(41,12)	4750	68.92024
d(41,13)	11225	105.9481
d(41,14)	10525	102.5914
d(41,15)	4850	69.64194
d(41,16)	11225	105.9481
d(41,17)	10525	102.5914
d(41,18)	4850	69.64194

Langkah 5. Menghitung Pembobotan (*Weight voting*)

Pada tahapan menghitung nilai *weight voting* yang didapat dari memasukkan nilai validitas dan nilai *Euclidean* pada persamaan 2-6 dan dilakukan perhitungan untuk data uji 1.

$$\begin{aligned}
 W_1 &= \text{validitas (1)} \times \frac{1}{d_1 + 0.5} \\
 &= 0.6667 \times \frac{1}{48.21825 + 0.5} \\
 &= 0.013684125
 \end{aligned}$$

Melakukan perhitungan yang sama untuk semua data latihan. Hasil perhitungan *weight voting* pada data uji 1 dapat dilihat pada Tabel 1.7.

Tabel 1.7 Hasil Perhitungan Weight voting Data Uji 1

data	weight voting	diagnosis
d(41,1)	0.013684125	1
d(41,2)	0.013985064	1
d(41,3)	0.045530699	1
d(41,4)	0.006154644	2
d(41,5)	0.006406463	2
d(41,6)	0.009408723	2
d(41,7)	0.006154644	3
d(41,8)	0.006406463	3
d(41,9)	0.009137817	3
d(41,10)	0.006154644	4
d(41,11)	0.006319115	4
d(41,12)	0.009603347	4
d(41,13)	0.006262833	5
d(41,14)	0.006466752	5
d(41,15)	0.009504537	5
d(41,16)	0.006262833	6
d(41,17)	0.006466752	6
d(41,18)	0.009504537	6

Langkah 6. Menentukan kelas dari data uji

Setelah didapatkan nilai *weight voting* dari semua data latih maka dilakukan pencarian nilai *weight voting* yang terbesar sebanyak nilai k yang telah ditentukan. Hasil pencarian nilai *weight voting* terbesar dapat dilihat pada Tabel 1.8.

Tabel 1.8 Hasil Pencarian Nilai Weight voting Terbesar Sebanyak Nilai k

data	weight voting	diagnosis
d(41,3)	0.045530699	1
d(41,2)	0.013985064	1
d(41,1)	0.013684125	1
d(41,12)	0.009603347	4
d(41,15)	0.009504537	5
d(41,18)	0.009504537	6
d(41,6)	0.009408723	2
d(41,9)	0.009137817	3
d(41,14)	0.006466752	5
d(41,17)	0.006466752	6
d(41,5)	0.006406463	2
d(41,8)	0.006406463	3



d(41,11)	0.006319115	4
d(41,13)	0.006262833	5
d(41,16)	0.006262833	6
d(41,4)	0.006154644	2
d(41,7)	0.006154644	3
d(41,10)	0.006154644	4

Keterangan: Nilai weight voting terbesar

Dari Tabel 1.8 maka didapatkan 3 *weight voting* terbesar yaitu 0.045530699 dengan kelasnya 1, 0.013985064 dengan kelasnya 1, dan 0.013684125 dengan kelasnya 1. Setelah diambil nilai *weight voting* terbesar sebanyak 3, lalu selanjutnya dilakukan penjumlahan nilai *weight voting* antar kelas yang sama. Dari data uji 1, maka kelas kategori 1 jumlah *weight voting*nya adalah 0.07 sehingga data uji 1 kelas kategorinya adalah kelas 1 karena nilai *weight voting*nya lebih besar dari kelas kategori lain.

Perhitungan yang sama dilakukan untuk data uji yang lainnya yaitu data 2, data 3, data 4, data 5 dan data 6. Dari 6 data uji yang digunakan didapatkan hasil akurasi sebesar 100% dengan 6 nilai prediksi benar dan tidak ada prediksi salah. Hasil perhitungan dari 6 data uji dapat dilihat dalam Tabel 1.9.

Klasifikasi antara perhitungan manual dengan menggunakan normalisasi dengan perhitungan manual tanpa normalisasi menghasilkan hasil yang sama, meskipun angka yang didapatkan berbeda akibat proses normalisasi.

Tabel 1.9 Hasil Perhitungan 6 Data Uji

Nilai k	Data Uji	Hasil	
		Data Asli	Prediksi sistem
3	41	1	1
	76	2	2
	140	3	3
	187	4	4
	265	5	5
	285	6	6
Akurasi			100%

