

SISTEM DIAGNOSIS PENYAKIT TANAMAN KENTANG MENGUNAKAN METODE K-NEAREST NEIGHBOR

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Syndu Pramanda Galuh Widestra

NIM: 135150200111017



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2019**

PENGESAHAN

SISTEM DIAGNOSIS PENYAKIT TANAMAN KENTANG MENGGUNAKAN METODE
K-NEAREST NEIGHBOR

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Syndu Pramanda Galuh Widestra
NIM: 135150200111017

Skripsi ini telah diuji dan dinyatakan lulus pada
4 Januari 2019

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I



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

Dosen Pembimbing II



Ratih Kartika Dewi, S.T., M.Kom
NIK. 201503 890520 2 001

Mengetahui
Ketua Jurusan Teknik Informatika



Tri Astoto Kurniawan, S.T, M.T, Ph.D
NIP: 19710518 200312 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 di dalam naskah skripsi ini dapat dibuktikan terdapat unsurunsur 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,

Syndu Pramanda Galuh Widestra
NIM: 135150200111017

KATA PENGANTAR

Puji syukur atas kehadiran Tuhan Yang Maha Esa karena hanya dengan rahmat dan hidayah-Nya, penulis dapat menyelesaikan penyusunan skripsi dengan judul Sistem Diagnosis Penyakit Tanaman Kentang menggunakan Metode K-Nearest Neighbor dengan baik.

Dalam penyusunan skripsi ini penulis banyak terlibat dengan berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Nurul Hidayat, S.Pd., M.Sc. dan Ratih Kartika Dewi, S.T., M.Kom. selaku Dosen Pembimbing Skripsi yang telah dengan sabar membimbing dan mengarahkan penulis sehingga dapat menyelesaikan skripsi ini.
2. Agus Wahyu Widodo, S.T., M.Cs. selaku Ketua Program Studi Teknik Informatika.
3. Tri Astoto Kurniawan, S.T., M.T., Ph.D. selaku Ketua Jurusan Teknik Informatika.
4. Seluruh dosen Informatika/Illmu Komputer Universitas Brawijaya atas kesediaan membagi ilmu kepada penulis.
5. Seluruh civitas akademika informatika/ilmu komputer Universitas Brawijaya terutama yang telah banyak membantu dan memberi dukungan selama penulisan skripsi ini.
6. Ayah, Ibu, Adik Aldy dan seluruh keluarga besar atas segala nasihat, kasih sayang, perhatian dan kesabarannya di dalam membesarkan dan mendidik penulis, serta yang senantiasa tiada henti-hentinya memberikan doa dan semangat demi terselesaikannya skripsi ini.
7. Sofia, Iwan, Torik, Willi, Ari, Estu, Bagas, Mas Doni, dan Dimas yang selalu memberikan semangat.
8. Rekan-rekan kopma yang tidak bisa disebutkan satu persatu.
9. Semua pihak yang telah membantu penulis dalam pengerjaan skripsi yang tidak bisa disebutkan satu persatu.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih banyak kekurangan, sehingga saran dan kritik yang membangun sangat penulis harapkan. Akhir kata penulis berharap skripsi ini dapat membawa manfaat bagi semua pihak yang menggunakannya.

Malang,

Penulis

Syndu Pramanda Galuh Widestra

ABSTRAK

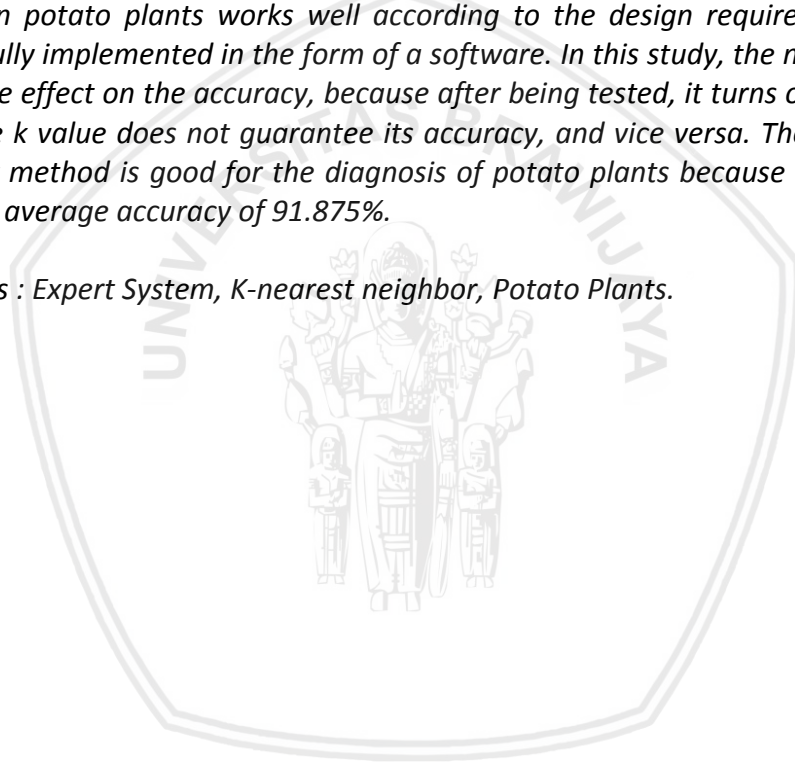
Kentang ialah salah satu tanaman yang potensial untuk dikembangkan. Produksi dan produktivitas kentang di Indonesia semakin menurun setiap tahunnya. Badan Pusat Statistik (BPS) mencatat produksi kentang di Indonesia mengalami penurunan 9,82%, dari 1.176.304 ton pada tahun 2009 menjadi 1.060.805 ton pada tahun 2010. Produktivitas kentang juga menurun dari 16,51 t/ha pada tahun 2009 menjadi 15,95 t/ha pada tahun 2010. Kendala menurunnya produktivitas kentang disebabkan oleh serangan hama dan penyakit, sehingga memerlukan suatu sistem yang dapat membantu mendiagnosis sejak dini terhadap serangan hama dan penyakit tanaman kentang. Metode yang dapat diimplementasikan untuk menyelesaikan permasalahan tersebut dalam mendiagnosis penyakit maupun melakukan prediksi yaitu *K-nearest neighbor* (*kNN*). Berdasarkan pengujian fungsional sistem diagnosis penyakit pada tanaman kentang berfungsi baik sesuai dengan rancangan kebutuhan dan berhasil diimplentasi dalam bentuk perangkat lunak. Pada penelitian ini, jumlah *K* tidak terlalu berpengaruh pada akurasi, karena setelah diuji ternyata semakin banyak nilai *k* tidak menjamin semakin kecil akurasinya, begitu pula sebaliknya. Metode *K-nearest neighbor* baik digunakan untuk diagnosis tanaman kentang karena menghasilkan rata-rata akurasi sebesar 91,875%.

Kata Kunci : Sistem Pakar, *K-nearest neighbor*, Tanaman Kentang.

ABSTRACT

Potato is one of the potential plant to be grown by people. The production and productivity of potato in Indonesia decreases each year. Badan Pusat Statistik (BPS) takes a note of the potato production in Indonesia and it is getting reduction of 9.82% from 1.176.304 ton in 2009 to 1.060.805 ton in 2010. The potato productivity also decreases from 16.51 t/ha in 2009 to 15.95 t/ha in 2010. The problem which causes its decreasing productivity is pest attacks and disease, so it needs a system which can help to diagnose it since early time from the pest attacks and the diseases of potato. Method which can be applied to solve the problem in diagnosing the disease as well as doing the prediction is by using K-nearest neighbor (kNN). Based on the functional testing, the disease diagnosis system in potato plants works well according to the design requirements and successfully implemented in the form of a software. In this study, the number of K have little effect on the accuracy, because after being tested, it turns out that the more the k value does not guarantee its accuracy, and vice versa. The K-nearest neighbor method is good for the diagnosis of potato plants because it produces result an average accuracy of 91.875%.

Keywords : Expert System, K-nearest neighbor, Potato Plants.



DAFTAR ISI

HALAMAN JUDUL	i
PENGESAHAN.....	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK	v
DAFTAR ISI	vii
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
DAFTAR SOURCE CODE.....	xii
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang	1
1.2 Rumusan masalah.....	3
1.3 Tujuan	3
1.4 Manfaat	3
1.5 Batasan masalah.....	3
1.6 Sistematika pembahasan.....	4
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Tinjauan Pustaka	5
2.2 Tanaman Kentang	7
2.2.1 Busuk Daun	8
2.2.2 Layu Fusarium	9
2.2.3 Bercak Daun	9
2.2.4 Virus Kompleks.....	10
2.2.5 Layu Bakteri.....	11
2.3 Data Mining.....	12
2.4 Klasifikasi	12
2.5 K-nearest neighbor.....	13
2.6 Pengujian Akurasi.....	15
BAB 3 METODOLOGI	16
3.1 Studi literatur	17
3.2 Analisis Kebutuhan.....	17
3.3 Pengumpulan Data.....	17
3.4 Perancangan Sistem	18
3.5 Implementasi.....	19
3.6 Pengujian Sistem	19

3.7 Analisis.....	20
3.8 Kesimpulan.....	20
BAB 4 PERANCANGAN.....	21
4.1 Tanaman Kentang	21
4.1.1 Busuk Daun	21
4.1.2 Layu Fusarium	22
4.1.3 Bercak Daun	22
4.1.4 Virus Kompleks.....	22
4.1.5 Layu Bakteri.....	23
4.2 Perancangan Proses	23
4.2.1 Akuisisi Pengetahuan	23
4.2.2 Basis Pengetahuan	24
4.2.3 Proses Klasifikasi K-Nearest Neighbor (K-NN).....	27
4.2.4 Perhitungan Euclidean Distance	28
4.2.5 Pengurutan Euclidean Distance	29
4.2.6 Ambil Data Sebanyak K	30
4.2.7 Hitung Kelas Terbanyak.....	30
4.3 Perhitungan Manual.....	31
4.4 Perancangan Antarmuka.....	38
4.4.1 Halaman Utama	38
4.4.2 Halaman Diagnosis.....	39
BAB 5 IMPLEMENTASI	41
5.1 Spesifikasi Sistem	41
5.1.1 Spesifikasi Perangkat Keras.....	41
5.1.2 Spesifikasi Perangkat Lunak	41
5.2 Batasan Implementasi.....	42
5.3 Implementasi Sistem.....	42
5.3.1 Implementasi Mesin Inferensi	42
5.3.2 Implementasi Antarmuka	50
BAB 6 PENGUJIAN DAN ANALISIS.....	52
6.1 Pengujian Akurasi.....	52
6.1.1 Pengujian dengan nilai K = 4	52
6.1.2 Pengujian dengan nilai K = 7	54
6.1.3 Pengujian dengan nilai K = 10	56
6.1.4 Pengujian dengan nilai K = 13	59
6.2 Analisis Pengujian Akurasi.....	61
6.2.1 Analisis Pengujian dengan nilai K = 4	61
6.2.2 Analisis Pengujian dengan nilai K = 7	61
6.2.3 Analisis Pengujian dengan nilai K = 10	61

6.2.4 Analisis Pengujian dengan nilai $K = 13$	61
BAB 7 KESIMPULAN DAN SARAN	62
7.1 Kesimpulan	62
7.2 Saran.....	62
DAFTAR PUSTAKA.....	63
LAMPIRAN 1 GEJALA DAN PENYAKIT TANAMAN KENTANG	64



DAFTAR TABEL

Tabel 2.1 Tinjauan Pustaka	Error! Bookmark not defined.
Tabel 4.1 Gejala Tanaman Kentang	Error! Bookmark not defined.
Tabel 4.2 Penyakit Tanaman Kentang.....	Error! Bookmark not defined.
Tabel 4.3 Data Latih Penyakit Tanaman Kentang	Error! Bookmark not defined.
Tabel 4.4 Data Latih	Error! Bookmark not defined.
Tabel 4.5 Data Uji	Error! Bookmark not defined.
Tabel 4.6 Euclidian Antar Data Latih	34
Tabel 4.7 Euclidian antar data latih yang telah diurutkan	35
Tabel 4.8 Euclidian antara data latih dan data uji sebanyak K	36
Tabel 4.9 Validitas Data Latih.....	37
Tabel 4.10 Jumlah setiap kelas yang telah terurut	37
Tabel 5.1 Spesifikasi Perangkat Keras	41
Tabel 5.2 Spesifikasi Perangkat Lunak	41
Tabel 6.1 Hasil Uji K=4.....	52
Tabel 6.2 Hasil Uji K=7.....	54
Tabel 6.3 Hasil Uji K=10.....	57
Tabel 6.4 Hasil Uji K=13.....	59

DAFTAR GAMBAR

Gambar 2.1 Busuk Daun.....	Error! Bookmark not defined.
Gambar 2.2 Layu Fusarium	Error! Bookmark not defined.
Gambar 2.3 Bercak Daun	Error! Bookmark not defined.
Gambar 2.4 Virus Kompleks	Error! Bookmark not defined.
Gambar 2.5 Layu Bakteri.....	Error! Bookmark not defined.
Gambar 2.6 Langkah – langkah proses data mining .	Error! Bookmark not defined.
Gambar 2.7 Ilustrasi algoritma K-NN	Error! Bookmark not defined.
Gambar 3.1 Blok metode penelitian	Error! Bookmark not defined.
Gambar 3.2 Desain Perancangan Sistem	Error! Bookmark not defined.
Gambar 3.3 Blok Diagram Proses Sistem	Error! Bookmark not defined.
Gambar 3.4 Diagram Blok Pengujian Akurasi	Error! Bookmark not defined.
Gambar 4.1 Diagram Alir KNN.....	Error! Bookmark not defined.
Gambar 4.2 Diagram Alir KNN.....	Error! Bookmark not defined.
Gambar 4.3 Diagram Alir Euclidian Distance	Error! Bookmark not defined.
Gambar 4.4 Diagram Alir Pengurutan Euclidian Distance	Error! Bookmark not defined.
Gambar 4.5 Antarmuka Halaman Utama.....	38
Gambar 5.1 Tampilan Antarmuka Splash Screen.....	50
Gambar 5.2 Tampilan Antarmuka Halaman Diagnosis	51
Gambar 5.3 Tampilan Antarmuka Halaman Hasil Diagnosis	51

DAFTAR SOURCE CODE

Source Code 5.1 Inisialisasi Variabel.....	47
Source Code 5.2 Penghitungan Euclidian Distance	47
Source Code 5.3 Pengurutan Euclidian distance	48
Source Code 5.4 Pengambilan Data Sebanyak K dan Penghitungan Jumlah Masing – Masing Kelas	49
Source Code 5.5 Pencarian jumlah kelas terbanyak dan penentuan kelas	49



BAB 1

PENDAHULUAN

1.1 Latar belakang

Kentang (*Solanum tuberosum* L.) merupakan salah satu umbi-umbian yang banyak digunakan sebagai sumber karbohidrat atau makanan pokok bagi masyarakat dunia setelah gandum, jagung dan beras. Sebagai umbi-umbian, kentang cukup menonjol dalam kandungan zat gizinya. Perbandingan protein terhadap karbohidrat yang terdapat di dalam umbi kentang lebih tinggi dari pada biji sereal dan umbi lainnya. Kandungan asam amino umbi kentang juga seimbang sehingga sangat baik bagi kesehatan (Niederhauser 1993). Umbi kentang mengandung sedikit lemak dan kolesterol, namun mengandung karbohidrat, sodium, serat diet, protein, vitamin C, kalsium, zat besi dan vitamin B6 yang cukup tinggi (Kolasa 1993).

Menurut Burlingame *et al.* (2009), selain sebagai sumber energi, kentang juga mengandung serat makanan (sampai 3,3%), asam askorbat (sampai 42 mg/100 g), kalium (sampai 693,8 mg/100 g), karotenoid total (sampai dengan 2700 mcg/100 g), dan fenol antioksidan seperti asam klorogenat (hingga 1570 mcg/100 g) dan polimer, dan anti-nutrisi seperti α -solanin (0,001- 47,2 mg/100 g), dan jumlah protein yang lebih rendah (0,85-4,2%), asam amino, mineral dan vitamin lain, dan komponen bioaktif. Komposisi tersebut mempengaruhi kualitas produk (Simek 1980).

Kentang merupakan salah satu tanaman sayuran yang potensial untuk dikembangkan. Produksi kentang di Indonesia mengalami penurunan 9,82%, dari 1.176.304 ton pada tahun 2009 menjadi 1.060.805 ton pada tahun 2010. Produktivitas kentang juga menurun dari 16,51 t/ha pada tahun 2009 menjadi 15,95 t/ha pada tahun 2010 (BPS 2010). Upaya untuk meningkatkan produktivitas kentang perlu didukung oleh varietas unggul dan bibit yang berkualitas tinggi.

Pengembangan komoditas kentang dihadapkan pada beberapa kendala, antara lain serangan hama dan penyakit. Penyakit hawar daun (late blight) yang disebabkan oleh cendawan *Phytophthora infestans*, merupakan salah satu penyakit utama pada tanaman kentang. Patogen ini memiliki keragaman genetik yang tinggi sehingga ketahanan varietas menjadi mudah patah (Cooke *et al.* 2003). Kerugian akibat penyakit ini dapat mencapai 100% pada tanaman kentang yang rentan, terutama pada musim hujan dengan kelembapan yang tinggi (Ojiambo *et al.* 2000).

Penggunaan varietas tahan merupakan cara yang paling tepat untuk mengendalikan penyakit hawar daun. Ketahanan yang bersifat durable dengan spektrum yang luas terhadap ras-ras *P. Infestans* dijumpai pada spesies kentang liar diploid *Solanum bulbocastanum* ($2n = 2x = 24$) (Helgeson *et al.* 1998). Namun, pemanfaatan spesies liar dalam pemuliaan tanaman kentang kerap kali sulit dilakukan. Hal ini karena spesies liar umumnya bersifat diploid yang secara genetik tidak kompatibel dengan varietas kentang komersial (tetraploid) atau yang dibudidayakan (During 1996).

Dengan perkembangan teknologi yang semakin canggih, adanya perangkat lunak yang dapat membantu mendiagnosis hama dan penyakit tanaman kentang akan sangat membantu petani agar dapat mendiagnosis dini dan melakukan penanganan dini terhadap serangan hama dan penyakit tanaman kentang, sehingga kegagalan panen akibat serangan hama dan penyakit dapat di minimalisir.

Suatu metode klasifikasi dapat diimplementasikan ke sebuah perangkat lunak untuk diagnosis hama dan penyakit tanaman kentang. Klasifikasi merupakan sebuah metode untuk mengelompokan data atau objek ke dalam kelas yang sebelumnya sudah didefinisikan. Data yang diinputkan dalam klasifikasi berupa sekumpulan atribut-atribut atau fitur-fitur yang merupakan ciri-ciri dari suatu data atau objek. Atribut atau fitur tersebut bisa berupa data diskrit maupun kontinyu.

Salah satu metode yang dapat digunakan untuk melakukan diagnosis hama dan penyakit pada tanaman kentang adalah *k-nearest neighbor*. *k-nearest neighbor (kNN)* termasuk kelompok instance-based learning. Algoritma ini juga merupakan salah satu teknik lazy learning. kNN dilakukan dengan mencari kelompok k objek dalam data training yang paling dekat (mirip) dengan objek pada data baru atau data testing (Wu, 2009).

. *K-nearest neighbor (kNN)* sering digunakan untuk mendiagnosis penyakit maupun melakukan prediksi, salah satu penelitian yang menggunakan metode . *k-nearest neighbor (kNN)* adalah penelitian yang dilakukan oleh Abdul Rohman dimana pada penelitian ini metode *k-nearest neighbor* digunakan untuk memprediksi kelulusan mahasiswa dan menghasilkan tingkat akurasi sebesar 85,15%.

Berdasarkan pemaparan dari penelitian-penelitian sebelumnya, maka penulis akan mengembangkan “sistem diagnosis penyakit pada tanaman kentang menggunakan metode *k-nearest neighbor*”. Sistem ini diharapkan dapat membantu petani untuk mengidentifikasi hama dan penyakit tanaman kentang. sistem menyediakan masukan berupa gejala yang dialami tanaman kentang untuk diproses, dan sistem dapat memberikan keluaran berupa informasi tentang penyakit yang menyerang tanaman kentang.

1.2 Rumusan masalah

Berdasarkan latar belakang diatas, maka diperoleh rumusan masalah yang mendasari penelitian ini adalah sebagai berikut :

1. Bagaimana implementasi suatu sistem untuk diagnosis penyakit tanaman kentang menggunakan metode *k-nearest neighbor*.
2. Bagaimana hasil pengujian sistem diagnosis penyakit tanaman kentang menggunakan metode *k-nearest neighbor*.

1.3 Tujuan

Tujuan pada penelitian ini adalah sebagai berikut :

1. Mengimplementasikan suatu sistem untuk diagnosis penyakit tanaman kentang dari gejala yang dialami tanaman menggunakan metode *k-nearest neighbor*.
2. Menguji akurasi sistem diagnosis penyakit tanaman kentang menggunakan metode *k-nearest neighbor*.

1.4 Manfaat

Manfaat yang diharapkan dari penelitian ini adalah dapat memberikan informasi tentang penyakit tanaman kentang sehingga petani dapat menanggulangi penyakit yang menyerang tanaman kentang.

1.5 Batasan masalah

Agar permasalahan yang dirumuskan dapat lebih berfokus dan tidak meluas, maka batasan-batasan yang ditentukan pada penelitian ini adalah sebagai berikut:

1. Sistem yang dikembangkan pada penelitian ini hanya mendiagnosis penyakit Layu fusarium, Layu Bakteri, Busuk Daun, Bercak Daun dan virus kompleks pada tanaman kentang.
2. Keluaran dari sistem ini berupa penyakit yang dialami oleh tanaman kentang dan cara penanganannya.
3. Menggunakan bahasa pemrograman JAVA.
4. Pengujian sistem dilakukan adalah pengujian akurasi.

1.6 Sistematika pembahasan

Adapun sistematika penulisan yang digunakan dalam menyusun laporan ini adalah sebagai berikut :

BAB I PENDAHULUAN

Pada BAB ini memuat latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, sistematika penulisan skripsi yang terkait dengan Sistem diagnosis penyakit pada tanaman kentang.

BAB II TINJAUAN PUSTAKA

Membahas tentang kajian pustaka yang berhubungan dengan penelitian ini dan teori dasar tentang tanaman kentang, Sistem, *K-nearest neighbor*, dan dasar teori lain yang mendukung penelitian ini.

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan metodologi dan proses analisa kebutuhan dan perancangan sistem yang digunakan dalam penelitian ini. Metode yang digunakan antara lain study literatur, pengumpulan data, analisis kebutuhan, perancangan sistem, implementasi, pengujian dan analisis sistem dan pengambilan kesimpulan.

BAB IV PERANCANGAN SISTEM

Membahas tentang kebutuhan dan perancangan pemodelan Sistem diagnosis Penyakit Tanaman Kentang dengan Metode *K-nearest neighbor*.

BAB V IMPLEMENTASI

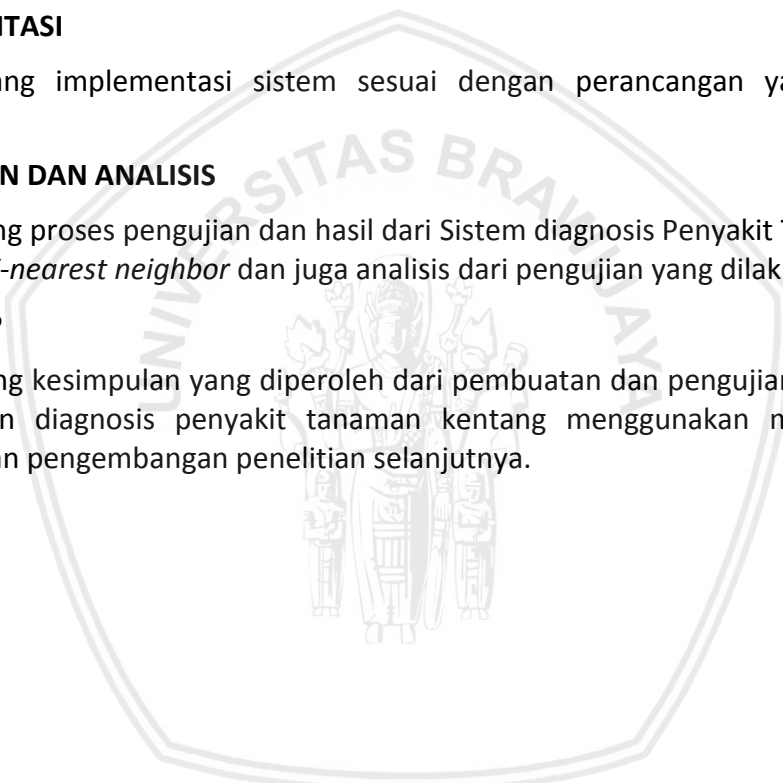
Membahas tentang implementasi sistem sesuai dengan perancangan yang telah dibuat sebelumnya.

BAB VI PENGUJIAN DAN ANALISIS

Membahas tentang proses pengujian dan hasil dari Sistem diagnosis Penyakit Tanaman Kentang dengan Metode *K-nearest neighbor* dan juga analisis dari pengujian yang dilakukan.

BAB VII PENUTUP

Membahas tentang kesimpulan yang diperoleh dari pembuatan dan pengujian perangkat lunak dalam pemodelan diagnosis penyakit tanaman kentang menggunakan metode *K-nearest neighbor* dan saran pengembangan penelitian selanjutnya.



BAB 2 LANDASAN KEPUSTAKAAN

2.1 Tinjauan Pustaka

Kajian pustaka pada penelitian ini akan membahas beberapa penelitian sistem yang telah dilakukan sebelumnya. Beberapa penelitian ini akan digunakan penulis untuk mendukung penelitian dalam skripsi ini. Penelitian penelitian tersebut ditunjukkan pada tabel 2.1.

Penelitian yang pertama dilakukan oleh Bayu Laksana Yudha, Lailil dan Randy pada tahun 2018 dengan judul “Klasifikasi Risiko Hipertensi Menggunakan Metode Neighbor Weighted K-Nearest Neighbor (NWKNN)”. Penelitian ini bertujuan untuk menaggulangi hipertensi secara dini yaitu dengan melakukan deteksi dini hipertensi dalam bentuk Sistem Kewaspadaan Dini (SKD). Data yang dibutuhkan pada penelitian ini dilakukan klasifikasi risiko hipertensi berdasarkan rekam medis dengan menggunakan metode klasifikasi *Neighbor Weighted K-Nearest Neighbor* (NWKNN). Metode ini merupakan perkembangan dari metode KNN. Pada NWKNN terdapat proses pembobotan di setiap kelas risiko hipertensi. Di dalam penelitian ini dilakukan klasifikasi hipertensi menjadi 4 risiko yaitu Normal, Pre Hipertensi, Stadium 1 dan Stadium 2. Hasil dari penelitian ini menunjukkan bahwa metode NWKNN dapat melakukan klasifikasi risiko hipertensi dengan baik saat data latih yang digunakan sebanyak 100 data dan data uji sebanyak 25 data, nilai $K=10$, dan nilai $E=4$ dengan hasil akurasi mencapai 88%.

Penelitian kedua dilakukan oleh Tri Halomoan Simanjuntak, Wayan Firdaus Mahmudy dan Sutrisno yang berjudul “Implementasi *Modified K-Nearest Neighbor* Dengan Otomatisasi Nilai K Pada Pengklasifikasian Penyakit Tanaman Kedelai”. Penelitian ini bertujuan untuk melakukan klasifikasi jenis penyakit yang menyerang tanaman kedelai. Variable yang dibutuhkan berupa gejala-gejala umum penyakit tanaman kedelai. Hasil akurasi tertinggi pada pengujian ini sebesar 100% dengan nilai $k=1$ dan rata-rata akurasi dari 5 percobaan sebesar 98,83%.

Penelitian ketiga dilakukan oleh Novita Mariana, Rara Sriartati Rejeki, Jeffri Alfa Razaq yang berjudul “Penerapan Algoritma k-NN (*Nearest Neighbor*) Untuk Deteksi Penyakit (Kanker Serviks)”. Penelitian ini bertujuan untuk memperoleh keputusan klinis penyakit kanker rahim berdasarkan tingkat stadium sebagai upaya meningkatkan pelayanan praktek kedokteran pada pasien untuk penanganan medis sedini mungkin. Variable yang dibutuhkan berupa gejala-gejala penyakit kanker serviks. Hasil dari penelitian ini berupa perangkat lunak untuk mencari informasi mengenai penyakit kanker serviks dan konsultasi keluhan gejala penyakit kanker serviks dengan mudah.

Penelitian kelima dilakukan oleh Eva Agustina Ompusunggu dan kawan-kawan yang berjudul “Identifikasi Penyakit Tanaman Jarak Pagar Menggunakan Metode *Fuzzy K-Nearest Neighbor* (FK-NN)”. Penelitian ini bertujuan untuk mengetahui dan memudahkan dalam mendiagnosis penyakit yang menyerang jarak pagar. Variable yang dibutuhkan berupa Gejala penyakit Tanaman Jarak. Hasil dari penelitian ini berupa perangkat lunak untuk diagnosis penyakit tanaman jarak dari sembilan penyakit yang ada. Hasil pengujian yang dilakukan pada penelitian ini didapatkan akurasi tertinggi sebesar 80%.

Tabel 0.1 Tinjauan Pustaka

No	Judul	Objek (Input)	Metode (Proses)	Hasil(output)
1.	Klasifikasi Risiko Hipertensi Menggunakan Metode Neighbor Weighted K-Nearest Neighbor (NWKNN)".	klasifikasi risiko hipertensi berdasarkan rekam medis. 4 risiko yaitu Normal, Pre Hipertensi, Stadium 1 dan Stadium 2	Metode Neighbor Weighted K-Nearest Neighbor.	Sistem ini menghasilkan klasifikasi deteksi secara dini risiko hipertensi
2.	Implementasi <i>Modified K-Nearest Neighbor</i> Dengan Otomatisasi Nilai K Pada Pengklasifikasian Penyakit Tanaman Kedelai	Input berupa gejala-gejala umum yang dialami tanaman kedelai	Metode K-nearest neighbor.	Sistem ini menghasilkan klasifikasi penyakit tanaman jenis apa yang menyerang tanaman kedelai.
1.	Penerapan Algoritma K-Nn (<i>Nearest Neighbor</i>) Untuk Deteksi Penyakit (Kanker Serviks)	Input berupa gejala-gejala penyakit kanker serviks.	Metode K-nearest neighbor.	Sistem ini menghasilkan perangkat lunak untuk mencari informasi mengenai penyakit kanker serviks dan konsultasi keluhan gejala penyakit kanker serviks dengan mudah.

2.2 Tanaman Kentang

Kentang (*Solanum tuberosum* L.) merupakan salah satu umbi-umbian yang banyak digunakan sebagai sumber karbohidrat atau makanan pokok bagi masyarakat dunia setelah gandum, jagung dan beras. Sebagai umbi-umbian, kentang cukup menonjol dalam kandungan zat gizinya. Perbandingan protein terhadap karbohidrat yang terdapat di dalam umbi kentang lebih tinggi dari pada biji sereal dan umbi lainnya. Kandungan asam amino umbi kentang juga seimbang sehingga sangat baik bagi kesehatan (Niederhauser 1993). Umbi kentang mengandung sedikit lemak dan kolesterol, namun mengandung karbohidrat, sodium, serat diet, protein, vitamin C, kalsium, zat besi dan vitamin B6 yang cukup tinggi (Kolasa 1993).

Menurut Burlingame et al.(2009), selain sebagai sumber energi, kentang juga mengandung serat makanan (sampai 3,3%), asam askorbat (sampai 42 mg/100 g), kalium (sampai 693,8 mg/100 g), karotenoid total (sampai dengan 2700 mcg/100 g), dan fenol antioksidan seperti asam klorogenat (hingga 1570 mcg/100 g) dan polimer, dan anti-nutrisi seperti α -solanin (0,001- 47,2 mg/100 g), dan jumlah protein yang lebih rendah (0,85-4,2%), asam amino, mineral dan vitamin lain, dan komponen bioaktif. Komposisi tersebut mempengaruhi kualitas produk (Simek 1980).

2.2.1 Busuk Daun

Penyakit busuk daun atau busuk phytophthora pada tanaman kentang disebabkan oleh cendawan *Phytophthora infestans*. Cendawan patogen ini ditularkan melalui udara dan air. Namun tidak tertutup kemungkinan penularannya melalui peralatan pertanian yang digunakan dan manusia yang kontak langsung dengan tanaman terinfeksi. Gejala awal penyakit busuk daun (busuk phytophthora) pada tanaman kentang yaitu berupa bercak-bercak kebasah-basahan pada bagian tengah daun dan tepi daun. Selanjutnya bercak-bercak tersebut melebar dan terbentuk daerah nekrotik yang berwarna coklat atau kehitam-hitaman. Bercak dikelilingi oleh masa aporangium yang berwarna putih dengan latar belakang berwarna hijau kelabu. Selanjutnya serangan dapat dengan cepat menyebar kebagian batang, tangkai dan umbi kentang. Serangan penyakit busuk daun pada kentang berkembang dan menyebar dengan cepat pada musim hujan dengan kelembaban tinggi dan pada suhu rendah. Cendawan patogen ini juga memiliki banyak tanaman inang, seperti gulma/rumput liar maupun tanaman budidaya lainnya. Tanaman budidaya yang menjadi inang cendawan ini antara lain terung, semangka, mentimun, melon, cabai, tomat, labu, oyong/gambas.

Sumber : <https://mitalom.com/wp-content/uploads/2017/03/Gambar-Gejala-Penyakit->



Gambar 0.1 Busuk Daun

Layu-Bakteri-pada-Kentang.jpg

2.2.2 Layu Fusarium

Penyakit layu fusarium pada tanaman kentang disebabkan oleh cendawan *Fusarium oxysporum*. Cendawan ini penularannya melalui air, udara, peralatan pertanian yang digunakan dan manusia. Gejala serangan layu fusarium pada tanaman kentang yaitu tanaman tiba-tiba menjadi layu, dimulai dari daun-daun tua (daun bagian bawah). Tulang-tulang daun berubah warna menjadi kuning. Apabila dibelah, jaringan batang dan akar berwarna coklat. Kemudian tanaman layu secara keseluruhan, mengering lalu mati. Cendawan *Fusarium oxysporum* bersifat polyfag, yaitu memiliki banyak sekali tanaman inang baik itu tanaman liar (gulma) maupun tanaman budidaya. Tanaman budidaya yang menjadi inang cendawan patogen ini antara lain cabai, tomat, labu, gambas/oyong, buncis, terung, kacang panjang, mentimun, pare/paria, seledri, semangka, melon dan tomat.

Sumber : <https://mitalom.com/wp-content/uploads/2017/03/Gambar-Gejala-Penyakit->



Gambar 0.2 Layu Fusarium

Layu-Bakteri-pada-Kentang.jpg

2.2.3 Bercak Daun

Penyakit bercak daun alternaria pada tanaman kentang dikalangan petani populer dengan nama penyakit bercak kering. Penyakit Bercak Kering / Bercak Daun Alternaria pada kentang disebabkan oleh cendawan *Alternaria sp.* Penyakit ini ditularkan melalui udara, air dan manusia.

Gejala awal serangan penyakit ini yaitu timbulnya bercak-bercak kecil pada daun, lama kelamaan bercak tersebut membesar. Bercak berwarna coklat dengan lingkaran-lingkaran sepusat. Masa konidia yang berwarna kelabu ampai hitam terlihat diatas bercak. Serangan dimulai dari daun paling bawah / daun-daun tua. Selanjutnya menyebar keatas kedaun yang lebih muda. Penyakit ini berkembang pesat pada kondisi kelembaban yang tinggi dengan suhu optimum berkisar antara 28 – 30 C.

Cendawan *Alternaria* sp. juga bersifat polyfag, yaitu memiliki banyak sekali tanaman inang. Tanaman budidaya yang menjadi inang patogen ini antara lain mentimun, melon, gambas/oyong, kentang, tomat, kacang panjang, labu, seledri, paria dan semangka.



Gambar 0.3 Bercak Daun

Sumber : <https://mitalom.com/wp-content/uploads/2017/03/Gambar-Gejala-Penyakit-Layu-Bakteri-pada-Kentang.jpg>

2.2.4 Virus Kompleks

Ada beberapa jenis virus penyebab penyakit pada tanaman kentang, yaitu *virus mosaik*, *virus Y*, *virus daun menggulung* dan lain sebagainya. Penyakit pada tanaman kentang yang disebabkan oleh virus ditularkan oleh serangga. Serangga/hama pembawa (vektor) penyakit virus antara lain kutu daun, thrips dan tungau. Penularan juga bisa melalui peralatan pertanian yang digunakan dan melalui manusia. Gejala serangan virus pada tanaman kentang bervariasi, tergantung jenis virus yang menyerang. Namun gejala umum penyakit yang disebabkan oleh virus antara lain terdapat gambaran mosaik pada daun-daun muda dengan beberapa corak. Bagian daun yang klorosis dapat berwarna hijau muda sampai kuning, bahkan mendekati putih. Seringkali permukaan daun menjadi tidak rata atau tampak mempunyai lekuk-lekuk hijau tua. Virus penyebab penyakit pada kentang memiliki banyak tanaman inang, antara lain kentang, tomat, kacang-kacangan, bawang, cabai, semangka, melon dan mentimun.



Gambar 0.4 Virus Kompleks

Sumber : <https://mitalom.com/wp-content/uploads/2017/03/Gambar-Gejala-Penyakit-Layu-Bakteri-pada-Kentang.jpg>

2.2.5 Layu Bakteri

Penyakit layu bakteri pada tanaman kentang disebabkan oleh bakteri *Ralstonia solanacearum*. Bakteri patogen ini ditularkan melalui air dan penyebarannya melalui bibit (umbi) apabila budidaya dilakukan menggunakan bibit yang terinfeksi. Gejala awal penyakit layu bakteri pada tanaman kentang yaitu adanya pucuk atau daun muda yang layu, selanjutnya menjalar kebawah (ke daun yang lebih tua) dan terus menjalar hingga ke pangkal batang dan akar hingga akhirnya tanaman layu secara keseluruhan dan mati. Penyakit layu bakteri pada kentang berkembang pesat pada musim hujan (curah hujan tinggi) dan pada lahan yang ber-pH rendah. Tanaman yang menjadi inang bakteri *Ralstonia solanacearum* antara lain terung, cabai, tomat dan kentang.

Sumber : <https://mitalom.com/wp-content/uploads/2017/03/Gambar-Gejala-Penyakit->



Gambar 0.5 Layu Bakteri

2.3 Data Mining

Data Mining (Witten, 2011) didefinisikan sebagai proses penemuan pola dalam data. Berdasarkan tugasnya, data mining dikelompokkan menjadi deskripsi, estimasi, prediksi, klasifikasi, clustering dan asosiasi (Larose, 2005). Proses dalam tahap data mining (Gambar 1.) terdiri dari tiga langkah utama (Sumathi, 2006), yaitu :

a. Data Preparation

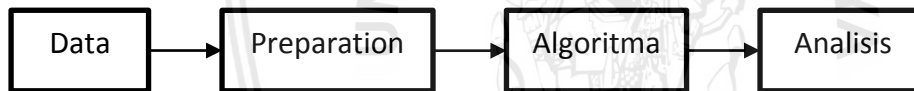
Pada langkah ini, data dipilih, dibersihkan, dan dilakukan preprocessed mengikuti pedoman dan knowledge dari ahli domain yang menangkap dan mengintegrasikan data internal dan eksternal ke dalam tinjauan organisasi secara menyeluruh.

b. Algoritma data mining

Penggunaan algoritma data mining dilakukan pada langkah ini untuk menggali data yang terintegrasi untuk memudahkan identifikasi informasi bernilai.

c. Fase analisa data

Keluaran dari data mining dievaluasi untuk melihat apakah knowledge domain ditemukan dalam bentuk rule yang telah diekstrak dari jaringan.



Gambar 0.6 Langkah – langkah proses data mining

2.4 Klasifikasi

Klasifikasi adalah proses penemuan model (atau fungsi) yang menggambarkan dan membedakan kelas data atau konsep yang bertujuan agar bisa digunakan untuk memprediksi kelas dari objek yang label kelasnya tidak diketahui (Han, 2006).

Algoritma klasifikasi yang banyak digunakan secara luas, yaitu Decision/classification trees, Bayesian classifiers/ Naïve Bayes classifiers, Neural networks, Analisa Statistik, Algoritma Genetika, Rough sets, k-nearest neighbor, Metode Rule Based, Memory based reasoning, dan Support vector machines (SVM).

Klasifikasi data terdiri dari 2 langkah proses. Pertama adalah learning (fase training), dimana algoritma klasifikasi dibuat untuk menganalisa data training lalu direpresentasikan dalam bentuk rule klasifikasi. Proses kedua adalah klasifikasi, dimana data tes digunakan untuk memperkirakan akurasi dari rule klasifikasi (Han, 2006). Proses klasifikasi didasarkan pada empat komponen (Gorunescu, 2011) :

a. Kelas

Variabel dependen yang berupa kategorikal yang merepresentasikan 'label' yang terdapat pada objek. Contohnya: resiko penyakit jantung, resiko kredit, customer loyalty, jenis gempa.

b. Predictor

Variabel independen yang direpresentasikan oleh karakteristik (atribut) data. Contohnya: merokok, minum alkohol, tekanan darah, tabungan, aset, gaji.

c. Training dataset

Satu set data yang berisi nilai dari kedua komponen di atas yang digunakan untuk menentukan kelas yang cocok berdasarkan predictor.

d. Testing dataset

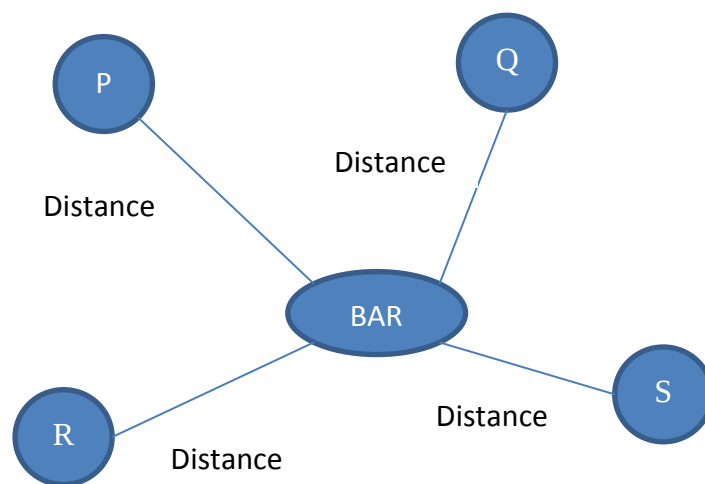
Berisi data baru yang akan diklasifikasikan oleh model yang telah dibuat dan akurasi klasifikasi dievaluasi.

2.5 K-nearest neighbor

k-nearest neighbor (kNN) termasuk kelompok instance-based learning. Algoritma ini juga merupakan salah satu teknik lazy learning. kNN dilakukan dengan mencari kelompok k objek dalam data training yang paling dekat (mirip) dengan objek pada data baru atau data testing (Wu, 2009).

Contoh kasus, misal diinginkan untuk mencari solusi terhadap masalah seorang pasien baru dengan menggunakan solusi dari pasien lama. Untuk mencari solusi dari pasien baru tersebut digunakan kedekatan dengan kasus pasien lama, solusi dari kasus lama yang memiliki kedekatan dengan kasus baru digunakan sebagai solusinya.

Terdapat pasien baru dan 4 pasien lama, yaitu P, Q, R, dan S (Gambar 2.2). Ketika ada pasien baru maka yang diambil solusi adalah solusi dari kasus pasien lama yang memiliki kedekatan terbesar.



Gambar 0.7 Ilustrasi algoritma K-NN

Misal D1 adalah jarak antara pasien baru dengan pasien P, D2 adalah jarak antara pasien baru dengan pasien Q, D3 adalah jarak antara pasien baru dengan pasien R, D4 adalah jarak antara pasien baru dengan pasien S. Dari ilustrasi gambar terlihat bahwa D2 yang paling terdekat dengan kasus baru. Dengan demikian maka solusi dari kasus pasien Q yang akan digunakan sebagai solusi dari pasien baru tersebut.

Ada banyak cara untuk mengukur jarak kedekatan antara data baru dengan data lama (data training), diantaranya euclidean distance dan manhattan distance (city block distance), yang paling sering digunakan adalah euclidean distance (Bramer, 2007), yaitu:

$$\sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \dots + (a_n - b_n)^2} \quad (2-1)$$

Dimana $a = a_1, a_2, \dots, a_n$, dan $b = b_1, b_2, \dots, b_n$ mewakili n nilai atribut dari dua record.

Untuk atribut dengan nilai kategori, pengukuran dengan euclidean distance tidak cocok. Sebagai penggantinya, digunakan fungsi sebagai berikut (Larose, 2006):

$$\text{different}(a_i, b_i) = \begin{cases} 0 & \text{jika } a_i = b_i \\ 1 & \text{selainnya} \end{cases}$$

Dimana a_i dan b_i adalah nilai kategori. Jika nilai atribut antara dua record yang dibandingkan sama maka nilai jaraknya 0, artinya mirip, sebaliknya, jika berbeda maka nilai kedekatannya 1, artinya tidak mirip sama sekali. Misalkan atribut warna dengan nilai merah dan merah, maka nilai kedekatannya 0, jika merah dan biru maka nilai kedekatannya 1.

Untuk mengukur jarak dari atribut yang mempunyai nilai besar, seperti atribut pendapatan, maka dilakukan normalisasi. Normalisasi bisa dilakukan dengan min-max normalization atau Z-score standardization (Larose, 2006). Jika data training terdiri dari

atribut campuran antara numerik dan kategori, lebih baik gunakan min-max normalization (Larose, 2006).

Untuk menghitung kemiripan kasus, digunakan rumus (Kusrini, 2009):

Keterangan :

P = Kasus baru

q = Kasus yang ada dalam penyimpanan

n = Jumlah atribut dalam tiap kasus

i = Atribut individu antara 1 sampai dengan n

f = Fungsi similarity atribut i antara kasus p dan kasus q

w = Bobot yang diberikan pada atribut ke-i

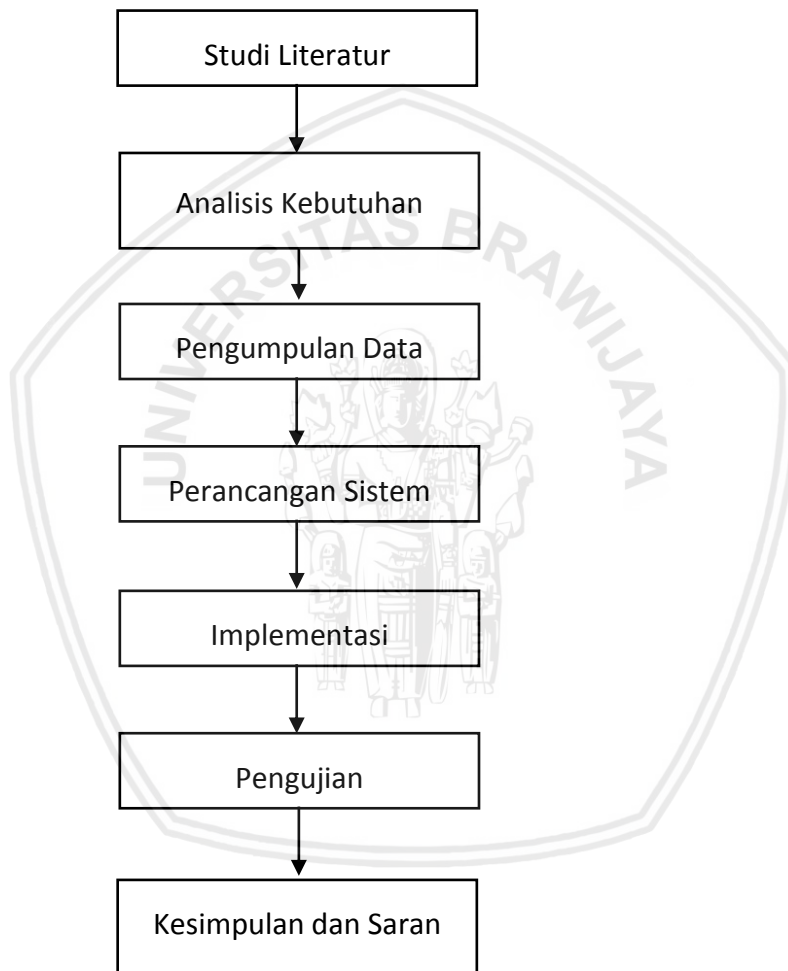
2.6 Pengujian Akurasi

Pengujian akurasi adalah suatu ukuran kedekatan hasil pengukuran terhadap angka sebenarnya (*true value / reference value*). Pada penelitian ini pengujian akurasi dilakukan untuk mengetahui kemampuan sistem dalam membuat keputusan. Akurasi dilakukan dengan menghitung jumlah diagnosis yang tepat dibagi dengan jumlah data. Tingkat akurasi ini dapat diperoleh dengan perhitungan pada persamaan .

$$\text{Tingkat Akurasi} = \frac{\sum \text{data uji benar}}{\sum \text{total data uji}} \times 100\%$$

BAB 3 METODOLOGI

Pada Bab ini akan dijelaskan langkah-langkah dalam penelitian “Sistem diagnosis Penyakit pada Tanaman Kentang menggunakan metode *K-nearest neighbor*”. Metodologi penelitian yang dilakukan dalam penelitian ini melalui beberapa tahapan yaitu studi literatur, analisis kebutuhan, pengumpulan data, perancangan sistem, implementasi, pengujian, analisis, pengambilan kesimpulan dan saran. Tahap pengerjaan ini diilustrasikan dengan blok metode penelitian seperti pada Gambar 3.1.



Gambar 0.1 Blok metode penelitian

3.1 Studi literatur

Studi literatur mempelajari literatur dari berbagai bidang ilmu yang menunjang penelitian tentang “Sistem diagnosis Penyakit pada Tanaman Kentang menggunakan metode *K-nearest neighbor*”. Diantaranya :

1. Android,
2. Metode K-nearest neighbor,
3. Tanaman Kentang,
4. Penyakit tanaman kentang.

3.2 Analisis Kebutuhan

Analisis kebutuhan merupakan proses identifikasi semua kebutuhan yang diperlukan dalam pengembangan “Sistem diagnosis Penyakit pada Tanaman Kentang menggunakan Metode *K-nearest neighbor*”. Analisis kebutuhan disesuaikan dengan variabel penelitian dan kebutuhan data yang akan digunakan.

Secara keseluruhan, kebutuhan yang digunakan dalam pengembangan sistem pada penelitian ini meliputi :

1. Data yang dibutuhkan, meliputi :
 - Data penyakit tanaman kentang
2. Variabel yang digunakan untuk melakukan diagnosis penyakit, meliputi :
Keadaan atau gejala yang terdapat pada umbi maupun daun tanaman.

3.3 Pengumpulan Data

Pada tahapan pengumpulan data, penelitian yang dibutuhkan adalah definisi penyakit tanaman kentang dan gejala-gejala setiap penyakit. Sumber data diperoleh dari hasil wawancara yang dilakukan dengan seorang pakar dari Dinas Pertanian Kabupaten Malang , penulis mendapatkan pengetahuan tentang gejala – gejala penyakit tanaman kentang serta cara penanggulangannya.

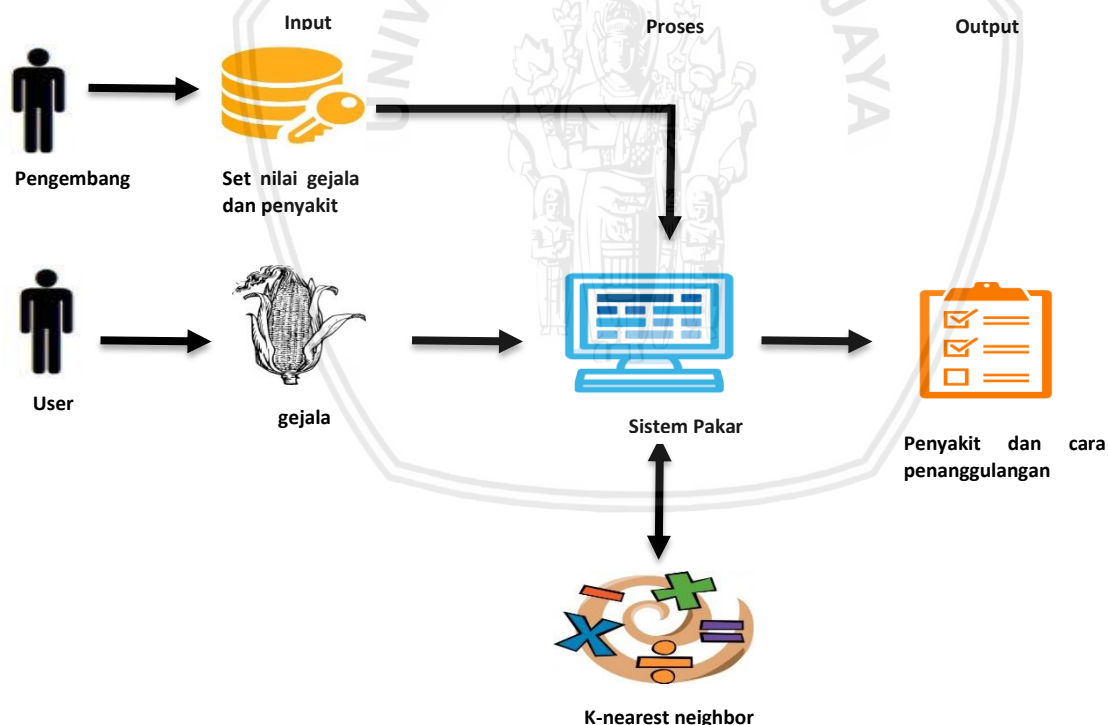
Berdasarkan cara pengumpulan data untuk penelitian terdapat dua jenis data yaitu sekunder dan primer, data sekunder adalah data yang berasal dari orang lain dan tidak digunakan untuk kegiatan penelitian tetapi digunakan untuk tujuan penelitian seperti melalui buku literatur. Data primer merupakan sebuah data yang didapatkan langsung dari objek penelitian. Metode pengumpulan data didapatkan langsung dari objek penelitian.

3.4 Perancangan Sistem

Perancangan sistem dibangun berdasarkan hasil pengambilan data dari lapangan dan analisis kebutuhan yang telah dilakukan. Tahapan ini akan dijelaskan secara lengkap pada bab perancangan yang memuat tentang analisis kebutuhan perangkat lunak dan perancangan arsitektur sistem.

Pemodelan “Sistem diagnosis Penyakit pada Tanaman Kentang” pada penelitian kali ini menggunakan metode *K-nearest neighbor* sebagai mesin inferensi. Metode *K-nearest neighbor* digunakan untuk mendapatkan nilai probabilitas berdasarkan data training. Nilai probabilitas yang sudah didapat akan digunakan sebagai proses pengambilan keputusan dalam melakukan diagnosis penyakit. Keluaran sistem akan berupa jenis penyakit yang menyerang tanaman kentang berdasarkan perhitungan metode *K-nearest neighbor* yang mempunyai nilai jarak terkecil.

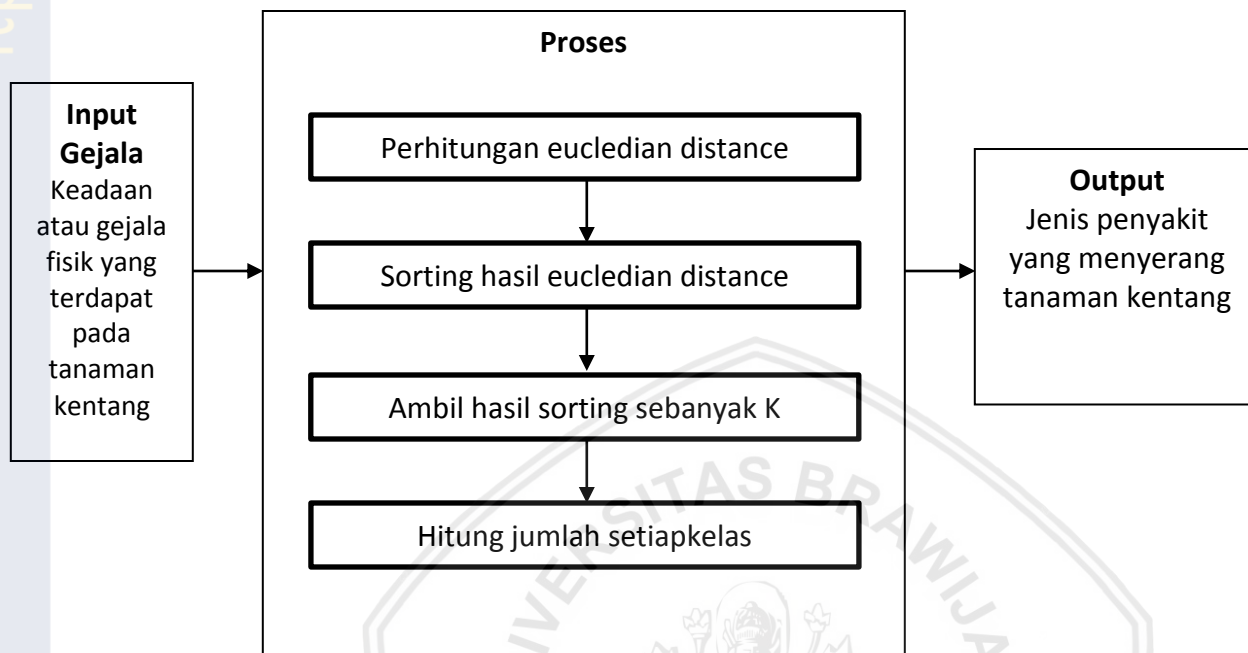
Gambar 3.2 menggambarkan Desain dari perancangan sistem. Pengembang adalah orang yang membuat sistem dan juga bertugas memasukkan nilai gejala dan penyakit sesuai data yang diberikan oleh pakar, Pengguna adalah aktor yang dapat mengakses sistem dimana pengguna nantinya akan memasukkan gejala ke sistem kemudian sistem akan memproses gejala yang telah dimasukkan oleh pengguna dengan metode *k-nearest neighbor* kemudian menampilkan hasil diagnosis sistem yang berupa penyakit dan cara penanggulangannya.



Gambar 0.2 Desain Perancangan Sistem

Sumber : Perancangan

Gambar 3.3 merupakan sebuah diagram blok yang menguraikan fungsi-fungsi sistem dan menggambarkan cara kerja sistem secara keseluruhan. Adapun diagram blok dapat dilihat sebagai berikut:



Gambar 0.3 Blok Diagram Proses Sistem

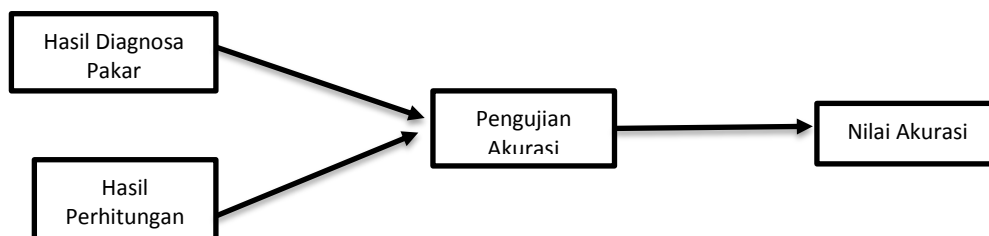
3.5 Implementasi

Implementasi sistem dilakukan dengan mengacu pada perancangan sistem yang telah ditentukan. Implementasi perangkat lunak dilakukan dengan menggunakan bahasa pemrograman JAVA, dan tools pendukung lainnya, implementasi sistem meliputi :

- Pembuatan *User Interface*.
- Penerapan metode K-nearest neighbor dalam pengembangan program dengan bahasa pemrograman JAVA.

3.6 Pengujian Sistem

Pengujian sistem dilakukan untuk mengetahui tingkat keberhasilan sistem yang telah dibangun. Pengujian sistem yang dilakukan adalah pengujian akurasi, pengujian akurasi dilakukan dengan membandingkan hasil diagnosis sistem dengan hasil diagnosis seorang pakar. Alur pengujian akurasi ditunjukkan pada gambar 3.4 berikut :



3.7 Analisis

Untuk analisis hasil dapat dilihat dari jawaban seorang pakar apakah sistem sudah berjalan dengan atau sistem sudah dapat mendiagnosis penyakit tanaman kentang secara benar dan tepat. Begitu juga dengan hasil dari penyebaran kuisioner yang disebarkan terhadap responden apakah sudah dapat digunakan atau belum.

3.8 Kesimpulan

Setelah proses yang dimulai dari studi literatur sampai pengujian dan analisis hasil selesai dilakukan, hal terakhir yang perlu dilakukan adalah penarikan kesimpulan. Penarikan kesimpulan dilakukan dengan cara apakah sistem sudah dapat berjalan dengan benar dan apakah sudah dapat mendiagnosis penyakit tanaman kentang sesuai dengan diagnosis dari pakar. Selain penarikan kesimpulan dilakukan evaluasi terhadap penelitian ini untuk mengetahui kekurangan dan kesalahannya agar untuk kedepannya penelitian tentang diagnosis penyakit tanaman kentang dapat dilakukan dengan lebih baik lagi dengan metode yang berbeda.

BAB 4 PERANCANGAN

Perancangan menjelaskan tentang kebutuhan yang diperlukan untuk membangun sistem diagnosis penyakit tanaman kentang menggunakan metode k-nearest .

4.1 Tanaman Kentang

Kentang (*Solanum tuberosum* L.) merupakan salah satu umbi-umbian yang banyak digunakan sebagai sumber karbohidrat atau makanan pokok bagi masyarakat dunia setelah gandum, jagung dan beras. Sebagai umbi-umbian, kentang cukup menonjol dalam kandungan zat gizinya. Perbandingan protein terhadap karbohidrat yang terdapat di dalam umbi kentang lebih tinggi dari pada biji sereal dan umbi lainnya. Kandungan asam amino umbi kentang juga seimbang sehingga sangat baik bagi kesehatan (Niederhauser 1993). Umbi kentang mengandung sedikit lemak dan kolesterol, namun mengandung karbohidrat, sodium, serat diet, protein, vitamin C, kalsium, zat besi dan vitamin B6 yang cukup tinggi (Kolasa 1993).

Menurut Burlingame et al.(2009), selain sebagai sumber energi, kentang juga mengandung serat makanan (sampai 3,3%), asam askorbat (sampai 42 mg/100 g), kalium (sampai 693,8 mg/100 g), karotenoid total (sampai dengan 2700 mcg/100 g), dan fenol antioksidan seperti asam klorogenat (hingga 1570 mcg/100 g) dan polimer, dan anti-nutrisi seperti α -solanin (0,001- 47,2 mg/100 g), dan jumlah protein yang lebih rendah (0,85-4,2%), asam amino, mineral dan vitamin lain, dan komponen bioaktif. Komposisi tersebut mempengaruhi kualitas produk (Simek 1980).

4.1.1 Busuk Daun

Penyakit busuk daun atau busuk phytophthora pada tanaman kentang disebabkan oleh cendawan *Phytophthora infestans*. Cendawan patogen ini ditularkan melalui udara dan air. Namun tidak tertutup kemungkinan penularannya melalui peralatan pertanian yang digunakan dan manusia yang kontak langsung dengan tanaman terinfeksi. Gejala awal penyakit busuk daun (busuk phytophthora) pada tanaman kentang yaitu berupa bercak-bercak kebasah-basahan pada bagian tengah daun dan tepi daun. Selanjutnya bercak-bercak tersebut melebar dan terbentuk daerah nekrotik yang berwarna coklat atau kehitam-hitaman. Bercak dikelilingi oleh masa aporangium yang berwarna putih dengan latar belakang berwarna hijau kelabu. Selanjutnya serangan dapat dengan cepat menyebar kebagian batang, tangkai dan umbi kentang. Serangan penyakit busuk daun pada kentang berkembang dan menyebar dengan cepat pada musim hujan dengan kelembaban tinggi dan pada suhu rendah. Cendawan patogen ini juga memiliki banyak tanaman inang, seperti gulma/rumput liar maupun tanaman budidaya lainnya. Tanaman budidaya yang menjadi inang

repository.ub.ac.id

cendawan ini antara lain terung, semangka, mentimun, melon, cabai, tomat, labu, oyong/gambas.

4.1.2 Layu Fusarium

Penyakit layu fusarium pada tanaman kentang disebabkan oleh cendawan *Fusarium oxysporum*. Cendawan ini penularannya melalui air, udara, peralatan pertanian yang digunakan dan manusia. Gejala serangan layu fusarium pada tanaman kentang yaitu tanaman tiba-tiba menjadi layu, dimulai dari daun-daun tua (daun bagian bawah). Tulang-tulang daun berubah warna menjadi kuning. Apabila dibelah, jaringan batang dan akar berwarna coklat. Kemudian tanaman layu secara keseluruhan, mengering lalu mati. Cendawan *Fusarium oxysporum* bersifat polyfag, yaitu memiliki banyak sekali tanaman inang baik itu tanaman liar (gulma) maupun tanaman budidaya. Tanaman budidaya yang menjadi inang cendawan patogen ini antara lain cabai, tomat, labu, gambas/oyong, buncis, terung, kacang panjang, mentimun, pare/paria, seledri, semangka, melon dan tomat.

4.1.3 Bercak Daun

Penyakit bercak daun alternaria pada tanaman kentang dikalangan petani populer dengan nama penyakit bercak kering. Penyakit Bercak Kering / Bercak Daun Alternaria pada kentang disebabkan oleh cendawan *Alternaria sp.* Penyakit ini ditularkan melalui udara, air dan manusia. Gejala awal serangan penyakit ini yaitu timbulnya bercak-bercak kecil pada daun, lama kelamaan bercak tersebut membesar. Bercak berwarna coklat dengan lingkaran-lingkaran sepusat. Masa konidia yang berwarna kelabu ampai hitam terlihat diatas bercak. Serangan dimulai dari daun paling bawah / daun-daun tua. Selanjutnya menyebar keatas kedaun yang lebih muda. Penyakit ini berkembang pesat pada kondisi kelembaban yang tinggi dengan suhu optimum berkisar antara 28 – 30 C.

Cendawan *Alternaria sp.* juga bersifat polyfag, yaitu memiliki banyak sekali tanaman inang. Tanaman budidaya yang menjadi inang patogen ini antara lain mentimun, melon, gambas/oyong, kentang, tomat, kacang panjang, labu, seledri, paria dan semangka.

4.1.4 Virus Kompleks

Ada beberapa jenis virus penyebab penyakit pada tanaman kentang, yaitu *virus mosaik*, *virus Y*, *virus daun menggulung* dan lain sebagainya. Penyakit pada tanaman kentang yang disebabkan oleh virus ditularkan oleh serangga. Serangga/hama pembawa (vektor) penyakit virus antara lain kutu daun, thrips dan tungau. Penularan juga bisa melalui peralatan pertanian yang digunakan dan melalui manusia. Gejala serangan virus pada tanaman kentang bervariasi, tergantung jenis virus yang menyerang. Namun gejala umum penyakit yang disebabkan oleh virus antara lain terdapat gambaran mosaik pada daun-daun muda dengan beberapa corak. Bagian daun yang klorosis dapat berwarna hijau

muda sampai kuning, bahkan mendekati putih. Seringkali permukaan daun menjadi tidak rata atau tampak mempunyai lekuk-lekuk hijau tua. Virus penyebab penyakit pada kentang memiliki banyak tanaman inang, antara lain kentang, tomat, kacang-kacangan, bawang, cabai, semangka, melon dan mentimun.

4.1.5 Layu Bakteri

Penyakit layu bakteri pada tanaman kentang disebabkan oleh bakteri *Ralstonia solanacearum*. Bakteri patogen ini ditularkan melalui air dan penyebarannya melalui bibit (umbi) apabila budidaya dilakukan menggunakan bibit yang terinfeksi. Gejala awal penyakit layu bakteri pada tanaman kentang yaitu adanya pucuk atau daun muda yang layu, selanjutnya menjalar kebawah (ke daun yang lebih tua) dan terus menjalar hingga ke pangkal batang dan akar hingga akhirnya tanaman layu secara keseluruhan dan mati. Penyakit layu bakteri pada kentang berkembang pesat pada musim hujan (curah hujan tinggi) dan pada lahan yang ber-pH rendah. Tanaman yang menjadi inang bakteri *Ralstonia solanacearum* antara lain terung, cabai, tomat dan kentang.

4.2 Perancangan Proses

Pada tahap ini, dilakukan perancangan sistem yang terdiri dari komponen-komponen penyusun sistem pakar, komponen – komponen dari sistem pakar itu sendiri terdiri dari akuisisi pengetahuan, mesin inferensi, basis pengetahuan, fasilitas penjelas dan rancangan antarmuka pengguna.

4.2.1 Akuisisi Pengetahuan

Akuisisi pengetahuan merupakan proses dan tahap pengumpulan data pengetahuan terhadap suatu masalah dari pakar atau dinas terkait. Ada beberapa cara yang dapat digunakan untuk memperoleh pengetahuan baik dari buku, observasi, internet dan terakhir dari pakar itu sendiri. Terdapat pula beberapa metode yang digunakan dalam akuisisi pengetahuan pada penelitian ini, yaitu :

1. Wawancara

Wawancara merupakan sebuah teknik pengumpulan data untuk memperoleh informasi yang didapat dari proses percakapan langsung atau tanya jawab kepada seorang narasumber. Pada penelitian ini wawancara bertujuan memperoleh pengetahuan pakar secara terperinci mengenai penyakit tanaman Kentang. Pengetahuan tentang penyakit tanaman Kentang tersebut meliputi informasi mengenai jenis penyakit tanaman Kentang, gejala penyakit tanaman Kentang, langkah-langkah pakar dalam mendiagnosis penyakit tanaman Kentang.

Narasumber atau pakar dalam penelitian ini adalah seorang pakar dari BPTP Karang Ploso Malang. Informasi mengenai jenis dan gejala penyakit tanaman Kentang didapatkan dari buku-buku referensi yang kemudian akan dikoreksi pakar pada saat wawancara apakah ada jenis atau gejala penyakit tanaman Kentang baru yang perlu ditambahkan ataupun dikurangi.

2. Analisa Protokol (Aturan)

Pada analisa protocol ini pakar akan diminta untuk memberikan proses pemikirannya. Peneliti akan menyediakan data kasus penyakit tanaman Kentang yang diperoleh dari hasil wawancara untuk disesuaikan dengan pemikiran pakar yang nantinya akan digunakan sebagai data *training* perhitungan *k-nearest neighbour*. Hasil dari proses ini digunakan sebagai acuan pembuatan basis pengetahuan untuk melakukan diagnosis penyakit tanaman Kentang.

4.2.2 Basis Pengetahuan

Basis pengetahuan berisi tentang pengetahuan yang diperlukan untuk memformulasikan, memahami dan memecahkan suatu permasalahan. Basis pengetahuan mempunyai dua elemen dasar yaitu fakta dan aturan khusus yang mengarahkan pengguna pengetahuan untuk memecahkan permasalahan dalam domain tertentu. Basis pengetahuan merupakan inti dari system dimana basis pengetahuan ini merupakan representasi dari seorang pakar.

Data hasil peneltian dan observasi lapangan yang telah dilakukan akan digunakan sebagai data *training* tersebut merupakan aturan-aturan yang nantinya akan digunakan sebagai basis pengetahuan pada sistem diagnosis tanaman Kentang. Kode dan gejala klinik yang terdapat pada tanaman Kentang dapat dilihat pada Tabel 4.1 dan Tabel 4.2.

Tabel 0.1 Gejala Tanaman Kentang

Kode	Gejala
G1	Timbul bercak kecil berwarna hijau kelabu dan agak basah kemudian berkembang
G2	warna bercak berubah menjadi coklat sampai hitam dengan bagian tepi berwarna putih yang merupakan sporangium
G3	Daun membusuk
G4	Tanaman mati
G5	Daun muda pada pucuk tanaman layu
G6	Daun tua dan daun bagian bawah menguning
G7	Daun menguning, daun menggulung, daun layu dan daun kering
G8	Pada bagian tanaman yang berada dalam tanah terdapat bercak-bercak berwarna coklat
G9	Infeksi akan menyebabkan akar dan umbi muda busuk
G10	Tanaman layu
G11	Umbi busuk
G12	Penyakit ini juga menyerang kentang digudang penyimpanan
G13	Infeksi masuk melalui luka-luka yang disebabkan nematoda/faktor mekanis
G14	Pada daun terdapat bercak kecil tersebar tidak teratur, berwarna coklat tua, lalu meluas ke daun muda
G15	Permukaan kulit umbi berbercak gelap tidak beraturan

G16	Umbi kering, berkerut dan keras
G17	Daun menggulung, daun menguning, dan jaringan mati
G18	Tanaman tumbuh kerdil, lurus dan pucat
G19	Umbi kecil-kecil / tidak menghasilkan sama sekali
G20	Mosaik laten pada daun
G21	Menyebabkan mosaik atau nekrosis lokal
G22	Pada permukaan umbi terdapat bercak-bercak berwarna kemerahan hingga kecoklatan
G23	Umbi kering, berkerut dan keras sampai umbi disimpan

Tabel 0.2 Penyakit Tanaman Kentang

Kode	Penyakit
P1	Busuk Daun
P2	Layu Bakteri
P3	Busuk Umbi
P4	Fusarium
P5	Bercak Kering
P6	Potato Virus
P7	Kudis

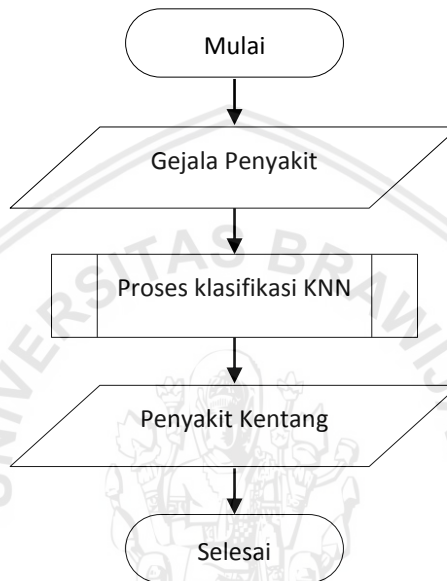
gejala dan penyakit tanaman Kentang, selanjutnya diubah menjadi data latih yang ditunjukkan oleh tabel 4.3.

Tabel 0.3 Data Latih Penyakit Tanaman Kentang

No	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	-	-	-	G23	Kelas
1	0	9	5	9	0	0	0	0	0	0	0	-	-	-	0	P1
2	9	0	5	9	0	0	0	0	0	0	0	-	-	-	0	P1
3	9	9	0	9	0	0	0	0	0	0	0	-	-	-	0	P1
4	9	9	5	0	0	0	0	0	0	0	0	-	-	-	0	P1
5	9	9	5	9	0	0	0	0	0	0	0	-	-	-	0	P1
6	0	0	3	0	0	9	0	0	0	0	0	-	-	-	0	P2
7	0	0	0	0	9	9	0	0	0	0	0	-	-	-	0	P2
8	0	0	3	0	9	9	0	0	0	0	0	-	-	-	0	P2
9	0	0	3	0	0	9	0	0	0	0	0	-	-	-	0	P2
10	0	0	0	0	9	9	0	0	0	0	0	-	-	-	0	P2
11	0	0	3	0	9	9	0	0	0	0	0	-	-	-	0	P2
12	0	0	0	0	0	0	0	9	9	0	0	-	-	-	0	P3
13	0	0	3	0	0	0	0	9	9	0	0	-	-	-	0	P3

14	0	0	0	0	0	0	9	9	0	0	0	-	-	-	0	P3
15	0	0	0	0	0	0	9	9	9	0	0	-	-	-	0	P3
16	0	0	3	0	0	0	9	9	9	0	0	-	-	-	0	P3
17	0	0	0	0	0	0	0	0	0	0	9	-	-	-	0	P4
18	0	0	0	0	0	0	0	0	0	9	0	-	-	-	0	P4
19	0	0	0	0	0	0	0	0	0	9	9	-	-	-	0	P4
20	0	0	0	0	0	0	0	0	0	9	9	-	-	-	0	P4
21	0	0	0	0	0	0	0	0	0	9	9	-	-	-	0	P4
22	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P5
23	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P5
24	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P5
25	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P5
26	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P5
27	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P5
28	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P5
29	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P5
30	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P5
31	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P5
32	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P5
33	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P5
34	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P5
35	0	0	0	0	0	0	0	0	0	0	0	-	-	-	9	P6
36	0	0	0	0	0	0	0	0	0	0	0	-	-	-	9	P6
37	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P6
38	0	0	0	0	0	0	0	0	0	0	0	-	-	-	9	P6
39	0	0	0	0	0	0	0	0	0	0	0	-	-	-	9	P6
40	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P6
41	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P7
42	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P7
43	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P7
44	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P7
45	0	0	0	0	0	0	0	0	0	0	0	-	-	-	0	P7

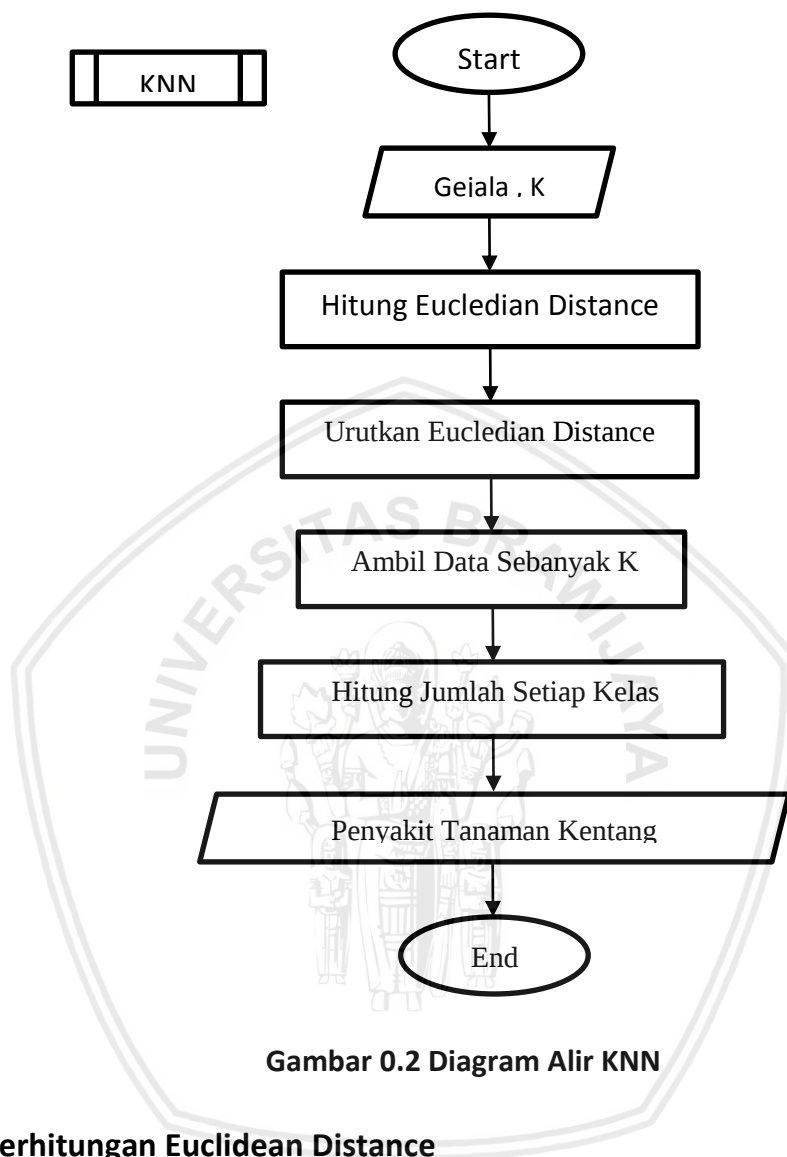
Untuk melakukan diagnosis penyakit tanaman Kentang, diberikan masukan-masukan berupa gejala-gejala penyakit tanaman dan nilai k pada sistem. Selanjutnya, dilakukan perhitungan jarak Euclidean antar data latih. Setelah jarak Euclidean didapatkan, dilakukan pengurutan mulai jarak euclidian terkecil hingga terbesar. Setelah didapatkan urutan jurak euclidian data uji terhadap setiap data latih, kemudian diambil data dengan jarak euclidian terdekat sebanyak K . Jika telah didapatkan data sebanyak K , kemudian kelas terbanyaklah yang dijadikan sebagai hasil akhir. Secara umum, diagram alir sistem ditunjukkan pada Gambar 4.1.



Gambar 0.1 Diagram Alir KNN

4.2.3 Proses Klasifikasi *K-Nearest Neighbor* (K-NN)

Dalam proses klasifikasi menggunakan K-NN, terdapat proses perhitungan jarak Euclidean antar data latih. Setelah jarak Euclidean didapatkan, dilakukan pengurutan mulai jarak euclidian terkecil hingga terbesar. Setelah didapatkan urutan jurak euclidian data uji terhadap setiap data latih, kemudian diambil data dengan jarak euclidian terdekat sebanyak K . Jika telah didapatkan data sebanyak K , kemudian kelas terbanyaklah yang dijadikan sebagai hasil akhir. Diagram alir klasifikasi MK-NN ditunjukkan pada Gambar 4.2.

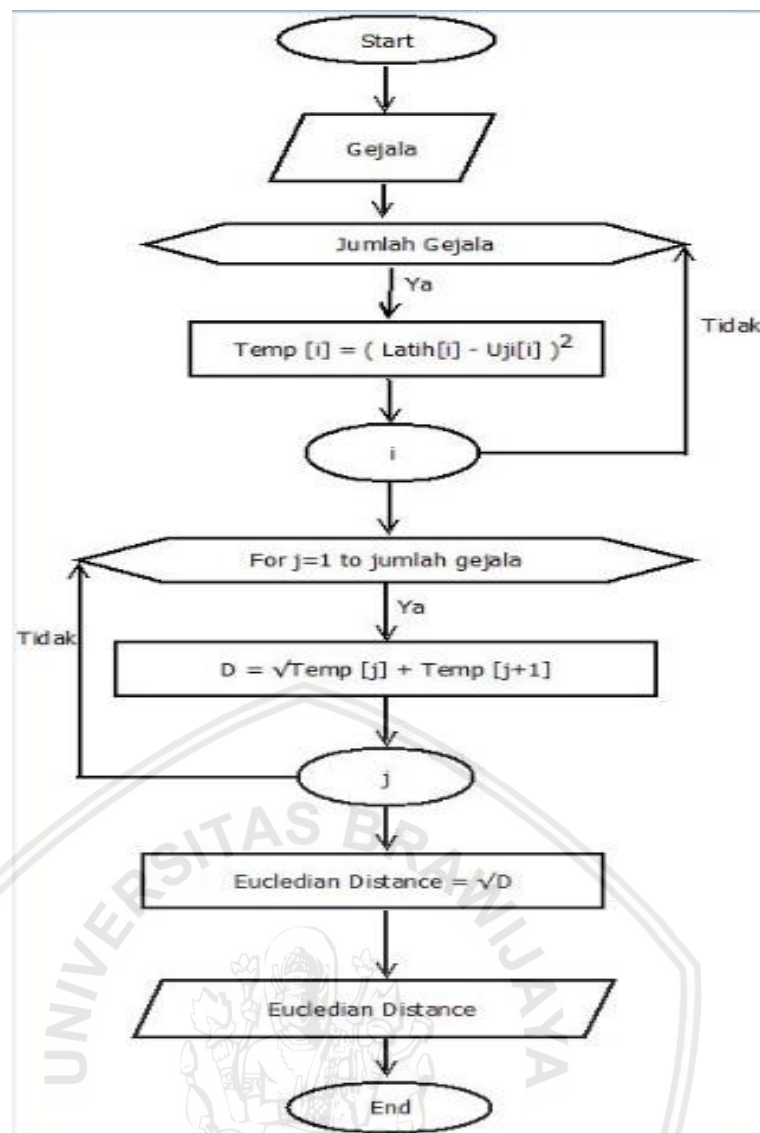


Gambar 0.2 Diagram Alir KNN

4.2.4 Perhitungan Euclidean Distance

Proses ini bertujuan untuk mendapatkan jarak antara data uji dengan setiap data latih. Tahapan-tahapan perhitungan jarak antara data latih dan data uji ditunjukkan pada Gambar 4.3.

	Eucledian Distance	
--	--------------------	--

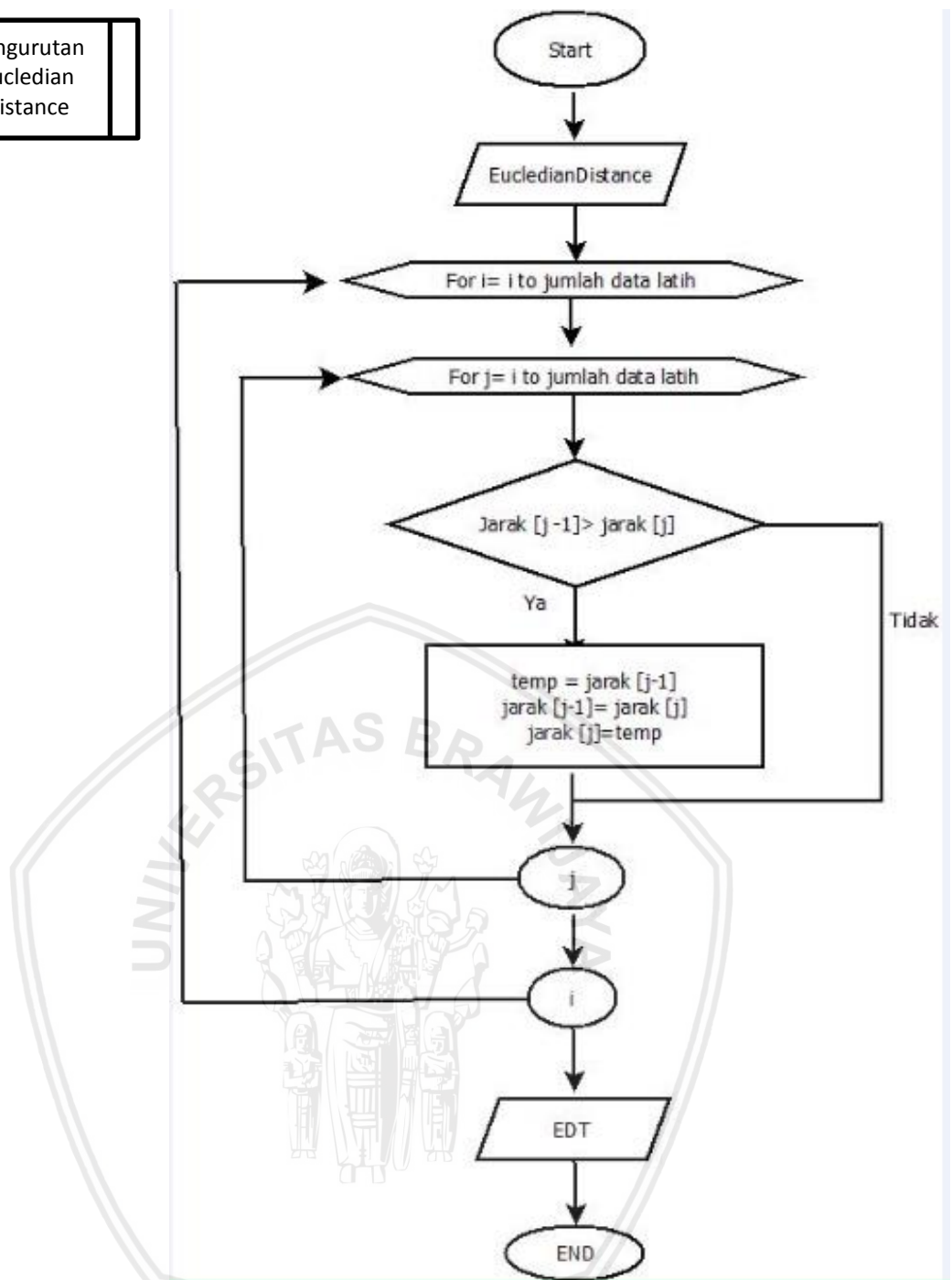


Gambar 0.3 Diagram Alir Eucledian Distance

4.2.5 Pengurutan Eucledian Distance

Proses ini bertujuan mendapatkan urutan nilai eucledian distance dari yang terkecil hingga terbesar dimana semakin kecil nilai eucledian dittance suau data latih terhadap data uji maka semakin mirip data latih tersebut terhadap data uji. Diagram alir perhitungan validitas ditunjukkan pada Gambar 4.4.

	Pengurutan Euclidian Distance	
--	-------------------------------------	--



Gambar 0.4 Diagram Alir Pengurutan Euclidian Distance

4.2.6 Ambil Data Sebanyak K

Proses ini bertujuan untuk data latih yang paling mirip dengan data uji dan data latih yang diambil sebanyak K.

4.2.7 Hitung Kelas Terbanyak

Setelah diambil data sebanyak K, kemudian dihitung jumlah setiap kelas dari data yang telah diambil, kelas dengan jumlah data terbanyak yang akan dijadikan sebagai kelas dari data uji.

[illegible]

12	0	0	0	0	0	0	0	9	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	P3
13	0	0	3	0	0	0	0	9	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	P3
14	0	0	0	0	0	0	9	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	P3
15	0	0	0	0	0	0	9	9	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	P3
16	0	0	3	0	0	0	9	9	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	P3
17	0	0	0	0	0	0	0	0	0	0	9	9	9	0	0	0	0	0	0	0	0	0	0	P4
18	0	0	0	0	0	0	0	0	0	9	0	9	9	0	0	0	0	0	0	0	0	0	0	P4
19	0	0	0	0	0	0	0	0	0	9	9	0	9	0	0	0	0	0	0	0	0	0	0	P4
20	0	0	0	0	0	0	0	0	0	9	9	9	0	0	0	0	0	0	0	0	0	0	0	P4
21	0	0	0	0	0	0	0	0	0	9	9	9	9	0	0	0	0	0	0	0	0	0	0	P4
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	9	9	0	0	P5
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	9	9	0	0	P5
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	9	0	9	0	0	P5
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	9	9	0	0	0	P5
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	9	9	9	0	0	P5
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0	9	9	0	0	P5
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	9	0	9	0	0	P5
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	9	9	9	0	0	P5
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	9	0	0	9	0	0	P5
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	9	0	9	9	0	0	P5
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	9	9	0	9	0	0	P5
33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	9	9	9	0	0	0	P5
34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	9	9	9	9	0	0	P5
35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	9	P6
36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	9	9	P6

37	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	P6
38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	9	P6
39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	9	9	P6
40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	P6
41	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	0	P7
42	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	3	0	0	0	0	0	0	0	P7
43	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	3	0	0	0	0	0	0	0	P7
44	0	0	0	0	0	0	0	0	0	0	0	0	0	9	9	0	0	0	0	0	0	0	0	P7
45	0	0	0	0	0	0	0	0	0	0	0	0	0	9	9	3	0	0	0	0	0	0	0	P7

Tabel 0.5 Data Uji

No	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	G16	G17	G18	G19	G20	G21	G22	G23
1	9	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	3	0	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	3	0	0	0	9	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	9	0	9	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	9	9	3	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	9	9	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	9	0

Langkah 1. Menentukan Euclidian Distance Data Latih Terhadap Data Uji

Digunakan persamaan (2-1) untuk menghitung jarak Euclidean antara data latih terhadap data uji. Hasilnya dapat dilihat pada Tabel 4.6.

Tabel 0.6 Euclidian Antar Data Latih

Data Latih Ke	Data Uji 1	Data Uji 2	Data Uji 3	Data Uji 4	Data Uji 5	Data Uji 6	Data Uji 7
1	13,67479	15,71623	18,11077	18,68154	18,92089	20,73644	16,8523
2	13,67479	15,71623	18,11077	18,68154	18,92089	20,73644	16,8523
3	9	18,24829	20,34699	20,12461	20,34699	22,04541	18,43909
4	5	15,71623	18,11077	18,68154	18,92089	20,73644	16,8523
5	10,29563	18,11077	20,22375	20,73644	20,95233	22,60531	19,10497
6	15,87451	12,72792	15,58846	15,87451	16,15549	18,24829	13,67479
7	18	9,486833	18,24829	18	18,24829	20,12461	16,09348
8	18,24829	9	18	18,24829	18,49324	20,34699	16,37071
9	15,87451	12,72792	15,58846	15,87451	16,15549	18,24829	13,67479
10	18	9,486833	18,24829	18	18,24829	20,12461	16,09348
11	18,24829	9	18	18,24829	18,49324	20,34699	16,37071
12	18	15,87451	13,0767	18	18,24829	20,12461	16,09348
13	18,24829	15,58846	12,72792	18,24829	18,49324	20,34699	16,37071
14	18	15,87451	3	18	18,24829	20,12461	16,09348
15	20,12461	18,24829	9,486833	20,12461	20,34699	22,04541	18,43909
16	20,34699	18	9	20,34699	20,56696	22,2486	18,68154
17	20,12461	18,24829	20,34699	15,58846	20,34699	22,04541	18,43909
18	20,12461	18,24829	20,34699	9	20,34699	22,04541	18,43909
19	20,12461	18,24829	20,34699	15,58846	20,34699	22,04541	18,43909
20	20,12461	18,24829	20,34699	9	20,34699	22,04541	18,43909
21	22,04541	20,34699	22,2486	12,72792	22,2486	23,81176	20,51828
22	20,12461	18,24829	20,34699	20,12461	20,34699	18	18,43909
23	20,12461	18,24829	20,34699	20,12461	20,34699	18	18,43909
24	20,12461	18,24829	20,34699	20,12461	20,34699	12,72792	18,43909
25	20,12461	18,24829	20,34699	20,12461	20,34699	12,72792	18,43909
26	22,04541	20,34699	22,2486	22,04541	22,2486	15,58846	20,51828
27	20,12461	18,24829	20,34699	20,12461	20,34699	18	18,43909
28	20,12461	18,24829	20,34699	20,12461	20,34699	12,72792	18,43909
29	22,04541	20,34699	22,2486	22,04541	22,2486	15,58846	20,51828
30	20,12461	18,24829	20,34699	20,12461	20,34699	12,72792	18,43909
31	22,04541	20,34699	22,2486	22,04541	22,2486	15,58846	20,51828
32	22,04541	20,34699	22,2486	22,04541	22,2486	9	20,51828
33	22,04541	20,34699	22,2486	22,04541	22,2486	9	20,51828
34	23,81176	22,2486	24	23,81176	24	12,72792	22,40536
35	18	15,87451	18,24829	18	18,24829	20,12461	9,848858
36	18,43909	16,37071	18,68154	18,43909	18,02776	20,51828	9
37	13,34166	10,29563	13,67479	13,34166	12,76715	16,09348	9
38	18	15,87451	18,24829	18	18,24829	20,12461	9,848858

39	18,43909	16,37071	18,68154	18,43909	18,02776	20,51828	9
40	13,34166	10,29563	13,67479	13,34166	12,76715	16,09348	9
41	15,58846	13,0767	15,87451	15,58846	9,486833	18	13,34166
42	15,87451	13,41641	16,15549	15,87451	9	18,24829	12,76715
43	15,87451	13,41641	16,15549	15,87451	9	18,24829	12,76715
44	18	15,87451	18,24829	18	3	20,12461	16,09348
45	18,24829	16,15549	18,49324	18,24829	0	20,34699	15,6205

Langkah 3. Melakukan pengurutan euclidian distance antara data latih terhadap data uji

Setelah jarak Euclidean didapatkan, selanjutnya nilai euclidian distance akan diurutkan dari yang terkecil hingga terbesar. Euclidian antara data latih terhadap data uji yang telah diurutkan berdasarkan nilai terkecil hingga terbesar ditunjukkan oleh tabel 4.7.

Tabel 0.7 Euclidian antar data latih yang telah diurutkan

Data Uji 1	Data Uji 2	Data Uji 3	Data Uji 4	Data Uji 5	Data Uji 6	Data Uji 7
5	9	3	9	0	9	9
9	9	9	9	3	9	9
10,29563	9,486833	9,486833	12,72792	9	12,72792	9
13,34166	9,486833	12,72792	13,34166	9	12,72792	9
13,34166	10,29563	13,0767	13,34166	9,486833	12,72792	9,848858
13,67479	10,29563	13,67479	15,58846	12,76715	12,72792	9,848858
13,67479	12,72792	13,67479	15,58846	12,76715	12,72792	12,76715
15,58846	12,72792	15,58846	15,58846	16,15549	15,58846	12,76715
15,87451	13,0767	15,58846	15,87451	16,15549	15,58846	13,34166
15,87451	13,41641	15,87451	15,87451	18,02776	15,58846	13,67479
15,87451	13,41641	16,15549	15,87451	18,02776	16,09348	13,67479
15,87451	15,58846	16,15549	15,87451	18,24829	16,09348	15,6205
18	15,71623	18	18	18,24829	18	16,09348
18	15,71623	18	18	18,24829	18	16,09348
18	15,71623	18,11077	18	18,24829	18	16,09348
18	15,87451	18,11077	18	18,24829	18	16,09348
18	15,87451	18,11077	18	18,24829	18,24829	16,09348
18	15,87451	18,24829	18	18,49324	18,24829	16,37071
18	15,87451	18,24829	18	18,49324	18,24829	16,37071
18,24829	15,87451	18,24829	18,24829	18,49324	18,24829	16,37071
18,24829	16,15549	18,24829	18,24829	18,92089	20,12461	16,8523
18,24829	16,37071	18,24829	18,24829	18,92089	20,12461	16,8523
18,24829	16,37071	18,49324	18,24829	18,92089	20,12461	16,8523
18,43909	18	18,68154	18,43909	20,34699	20,12461	18,43909
18,43909	18,11077	18,68154	18,43909	20,34699	20,12461	18,43909

20,12461	18,24829	20,22375	18,68154	20,34699	20,12461	18,43909
20,12461	18,24829	20,34699	18,68154	20,34699	20,12461	18,43909
20,12461	18,24829	20,34699	18,68154	20,34699	20,34699	18,43909
20,12461	18,24829	20,34699	20,12461	20,34699	20,34699	18,43909
20,12461	18,24829	20,34699	20,12461	20,34699	20,34699	18,43909
20,12461	18,24829	20,34699	20,12461	20,34699	20,34699	18,43909
20,12461	18,24829	20,34699	20,12461	20,34699	20,51828	18,43909
20,12461	18,24829	20,34699	20,12461	20,34699	20,51828	18,43909
20,12461	18,24829	20,34699	20,12461	20,34699	20,73644	18,43909
20,12461	18,24829	20,34699	20,12461	20,34699	20,73644	18,43909
20,12461	18,24829	20,34699	20,12461	20,34699	20,73644	18,43909
20,12461	18,24829	20,34699	20,12461	20,56696	22,04541	18,68154
20,34699	18,24829	20,34699	20,34699	20,95233	22,04541	19,10497
22,04541	20,34699	22,2486	20,73644	22,2486	22,04541	20,51828
22,04541	20,34699	22,2486	22,04541	22,2486	22,04541	20,51828
22,04541	20,34699	22,2486	22,04541	22,2486	22,04541	20,51828
22,04541	20,34699	22,2486	22,04541	22,2486	22,04541	20,51828
22,04541	20,34699	22,2486	22,04541	22,2486	22,2486	20,51828
22,04541	20,34699	22,2486	22,04541	22,2486	22,60531	20,51828
23,81176	22,2486	24	23,81176	24	23,81176	22,40536

Langkah 4. Mengambil data sebanyak K

Setelah melakukan pengurutan euclidian distance dari nilai terkecil hingga terbesar, selanjutnya data teratas diambil sebanyak K-data, nilai yang diambil sebanyak K ditunjukkan oleh tabel 4.8.

Tabel 0.8 Euclidian antara data latih dan data uji sebanyak K

Data Uji 1	Data Uji 2	Data Uji 3	Data Uji 4	Data Uji 5	Data Uji 6	Data Uji 7
5	9	3	9	0	9	9
9	9	9	9	3	9	9
10,29563	9,486833	9,486833	12,72792	9	12,72792	9
13,34166	9,486833	12,72792	13,34166	9	12,72792	9
13,34166	10,29563	13,07669	13,34166	9,486833	12,72792	9,848857
13,67479	10,29563	13,67479	15,58845	12,76714	12,72792	9,848857
13,67479	12,72792	13,67479	15,58845	12,76714	12,72792	12,76714
15,58845	12,72792	15,58845	15,58845	16,15549	15,58845	12,76714
15,87450	13,076697	15,588457	15,874508	16,155494	15,588457	13,341664

Langkah 5. Menghitung Jumlah setiap kelas berdasarkan data yang telah diambil sebanyak K

Setelah didapatkan data sebanyak K, selanjutnya menghitung jumlah setiap kelas berdasarkan data yang telah diambil sebanyak K. Jumlah setiap kelas hasil dari data yang diambil sebanyak K dapat dilihat pada tabel 4.9.

Tabel 0.9 Validitas Data Latih

Data Uji 1	Data Uji 2	Data Uji 3	Data Uji 4	Data Uji 5	Data Uji 6	Data Uji 7
5	0	0	0	0	0	0
1	6	2	1	2	0	0
0	0	5	0	0	0	0
0	0	0	5	0	0	0
1	1	0	1	5	0	3
0	0	0	0	0	9	0
2	2	2	2	2	0	6

Langkah 6. Penentuan Kelas

Setelah didapatkan Jumlah setiap kelas kemudian diambil kesimpulan berdasarkan jumlah kelas yang terbanyaklah yang dijadikan kelas data uji. Pengurutan jumlah kelas dari jumlah terbesar hingga terkecil dapat dilihat pada tabel 4.10 .

Tabel 0.10 Jumlah setiap kelas yang telah terurut

Data Uji 1	Data Uji 2	Data Uji 3	Data Uji 4	Data Uji 5	Data Uji 6	Data Uji 7
5	6	5	5	5	9	6
2	2	2	2	2	0	3
1	1	2	1	2	0	0
1	0	0	1	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

Dari tabel diatas didapatkan kesimpulan bahwa Data uji 1 termasuk kedalam Kelas penyakit 1, Data uji 2 termasuk kedalam Kelas penyakit 2, Data uji 3 termasuk kedalam Kelas penyakit 3, Data uji 4 termasuk kedalam Kelas penyakit 4, Data uji 5 termasuk kedalam Kelas penyakit 5, Data uji 6 termasuk kedalam Kelas penyakit 6, Data uji 7 termasuk kedalam Kelas penyakit 7.

4.4 Perancangan Antarmuka

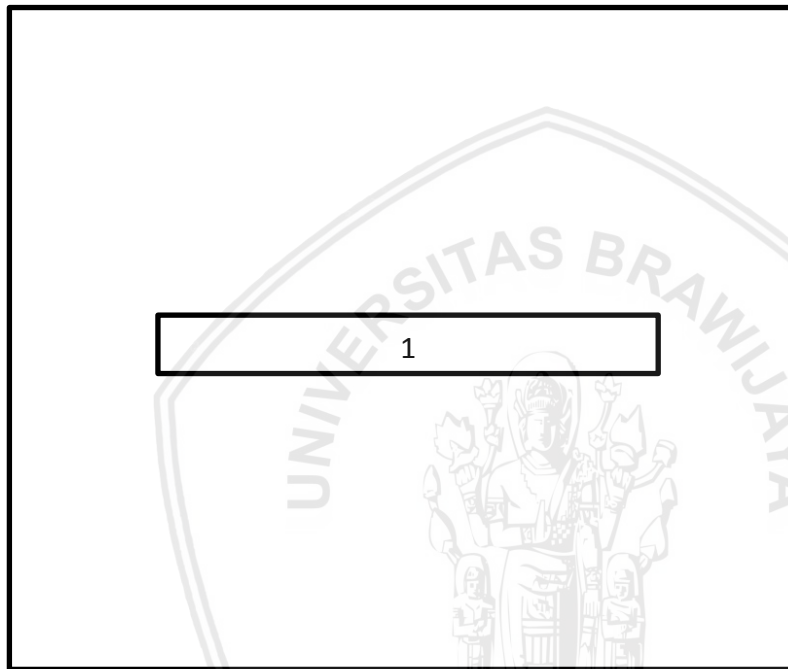
Antarmuka pengguna merupakan suatu mekanisme yang digunakan sebagai media komunikasi antara pengguna dan sistem. Antarmuka pengguna akan menerima informasi dari pengguna dan mengubahnya kedalam bentuk yang dapat diterima oleh sistem. Antarmuka menyediakan tampilan yang mudah digunakan dengan tujuan agar pengguna dapat memahami

repository.ub.ac.id

dan menggunakan sistem dengan mudah. Pada bagian ini akan dijelaskan mengenai spesifikasi rancangan antarmuka aplikasi sistem diagnosis penyakit kentang menggunakan metode K-Nearest Neighbor.

4.4.1 Halaman Utama

Halaman ini adalah halaman awal ketika pengguna membuka sistem, pada halaman ini terdapat menu yaitu Diagnosis yang ditunjukkan pada Gambar 4.6.



Gambar 0.5 Antarmuka Halaman Utama

Keterangan :

1. Tombol untuk masuk ke menu diagnosis

1.4.3 Halaman Diagnosis

Halaman ini adalah halaman yang muncul ketika pengguna mengklik tombol Diagnosis pada halaman utama, pada halaman ini pengguna dapat melakukan diagnosis penyakit tanaman kentang. Pada halaman ini terdapat daftar gejala – gejala yang ada pada penyakit tanaman kentang, pengguna dapat memasukkan gejala dengan cara memilih *checkbox* sesuai

repository.ub.ac.id

dengan gejala pada tanaman kentang. Antarmuka halaman diagnosis dapat dilihat pada Gambar

Diagnosa	
☐	Gejala 1
☐	Gejala 2
☐	Gejala 3
☐	Gejala 4
☐	Gejala 5
☐	Gejala 6
☐	Gejala 7

Diagram illustrating the interface of the diagnosis page. It shows a table with a header 'Diagnosa' and seven rows, each containing a checkbox and a symptom label (Gejala 1 through Gejala 7). A bracket labeled '1' groups the seven rows, indicating the list of symptoms. An arrow labeled '2' points to the 'Diagnosa' header, indicating the button to click for diagnosis.

4.8.

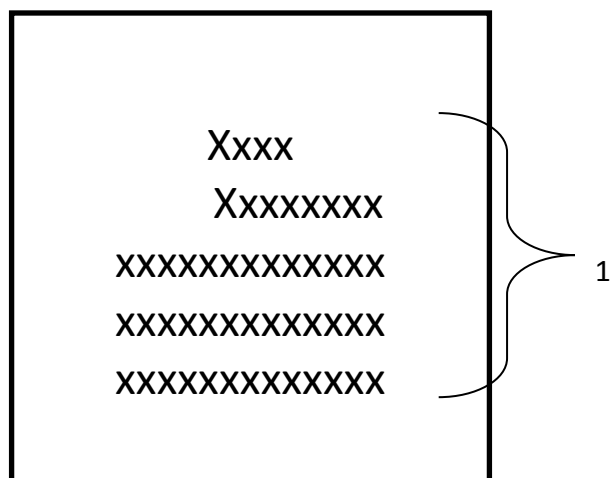
Keterangan :

Gambar 0.8 Antarmuka Halaman Diagnosis

1. Daftar gejala penyakit tanaman kentang
2. Tombol Diagnosis

1.4.4 Halaman Hasil Diagnosis

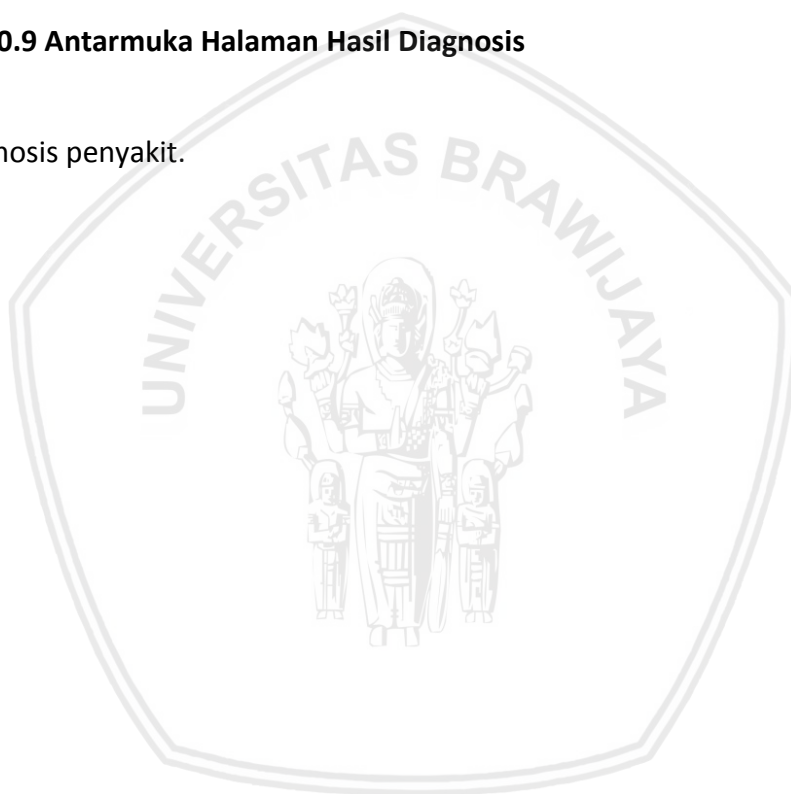
Halaman ini adalah halaman yang muncul ketika pengguna mengklik tombol diagnosis pada halaman diagnosis, pada halaman ini pengguna akan mendapatkan informasi hasil diagnosis penyakit tanaman kentang. Antarmuka halaman hasil diagnosis dapat dilihat pada Gambar 4.9.



Gambar 0.9 Antarmuka Halaman Hasil Diagnosis

Keterangan :

1. Hasil diagnosis penyakit.



BAB 5 IMPLEMENTASI

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

5.1 Spesifikasi Sistem

Hasil dari analisis kebutuhan dan perancangan sistem yang telah diuraikan pada bab perancangan menjadi acuan untuk melakukan implementasi menjadi sistem yang dapat berfungsi sesuai kebutuhan. Spesifikasi sistem diuraikan menjadi spesifikasi perangkat keras dan spesifikasi perangkat lunak.

5.1.1 Spesifikasi Perangkat Keras

Pengembangan Sistem Diagnosis Penyakit pada tanaman kentang menggunakan sebuah personal computer dengan spesifikasi perangkat keras yang ditunjukkan pada Tabel 5.1.

Tabel 0.1 Spesifikasi Perangkat Keras

Nama Komponen	Spesifikasi
Prosesor	Intel(R) Corei3(R) CPU N8820 @2.16GHz
Memori(RAM)	8,00 GB
Jenis Sistem	Sistem Operasi 64-bit, prosesor berbasis x64
Harddisk	500 GB

5.1.2 Spesifikasi Perangkat Lunak

Pengembangan Sistem Diagnosis Penyakit tanaman kentang menggunakan sebuah personal computer dengan spesifikasi perangkat lunak yang ditunjukkan pada Tabel 5.2.

Tabel 0.2 Spesifikasi Perangkat Lunak

Sistem Operasi	Microsoft Windows 8 Home 64-bit
Bahasa Pemrograman	Java
Tools Pemrograman	Android Studio
Emulator	HAXM

5.2 Batasan Implementasi

Batasan dalam implementasi Sistem diagnosis Penyakit tanaman kentang adalah sebagai berikut :

- 5.2.1.1.1.1 Sistem yang dibangun berdasarkan ruang lingkup aplikasi android dengan menggunakan bahasa pemrograman java.
- 5.2.1.1.1.2 Data yang digunakan dalam sistem disimpan dalam bentuk array.
- 5.2.1.1.1.3 Data yang digunakan berupa data-data gejala fisik pada tanaman kentang, data penyebab dari setiap gejala dan data penyakit yang menyerang tanaman kentang.
- 5.2.1.1.1.4 Masukan yang dilakukan oleh pengguna ke sistem berupa gejala fisik yang dialami.
- 5.2.1.1.1.5 Keluaran dari sistem adalah salah satu dari 7 penyakit pada tanaman kentang.
- 5.2.1.1.1.6 Metode yang digunakan adalah k-nearest neighbor.
- 5.2.1.1.1.7 Semua pengguna dapat mengakses sistem tanpa harus melakukan login.
- 5.2.1.1.1.8 Semua pengguna memiliki hak akses yang sama.
- 5.2.1.1.1.9 Semua pengguna berhak mengakses menu yang ada pada sistem.
- 5.2.1.1.1.10 Data yang digunakan pada sistem permanen dan tidak dapat diubah lagi.

5.3 Implementasi Sistem

Hasil perancangan sistem yang telah diuraikan pada bab perancangan dijadikan sebagai acuan untuk melakukan implementasi Sistem diagnosis Penyakit pada tanaman kentang. Bagian-bagian yang akan diimplementasikan pada sistem ini yaitu implementasi basis pengetahuan, implementasi mesin inferensi dan implementasi antarmuka.

5.3.1 Implementasi Mesin Inferensi

Sistem diagnosis Penyakit pada tanaman kentang ini memiliki empat tahap proses perhitungan yaitu yang pertama adalah Menghitung euclidean distance data latih terhadap data uji kemudian setelah mendapatkan euclidean distance setiap data latih terhadap data uji akan dilanjutkan ke proses kedua yaitu pengurutan euclidean distance, setelah jarak euclidean diurutkan kemudian diambil data sebanyak K yang memiliki jarak terkecil dan masuk ke proses akhir yaitu menghitung jumlah setiap kelas dimana kelas terbanyak lah yang akan dijadikan sebagai kelas dari data uji.

5.3.1.1 Implementasi Algoritma K-Nearest Neighbor

Implementasi algoritma proses diagnosis penyakit tanaman kentang mengacu pada proses perhitungan k-nearest neighbor. Proses diagnosis pertama kali dilakukan dengan melihat

kondisi tanaman kentang, kemudian pengguna memasukkan kondisi gejala fisik yang ada pada tanaman kentang untuk dilakukan proses diagnosis.

Implementasi Algoritma proses diagnosis dengan proses perhitungan k-nearest neighbor dapat dilihat pada Source Code 5.1 hingga 5.5.

	double latih[][]
1	={{0,9,5,9,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
2	{9,0,5,9,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
3	{9,9,0,9,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
4	{9,9,5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
5	{9,9,5,9,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
6	{0,0,3,0,0,9,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
7	{0,0,0,0,9,9,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
8	{0,0,3,0,9,9,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
9	{0,0,3,0,0,9,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
10	{0,0,0,0,9,9,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
11	{0,0,3,0,9,9,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
12	{0,0,0,0,0,0,0,9,9,0,0,0,0,0,0,0,0,0,0,0},
13	{0,0,3,0,0,0,0,9,9,0,0,0,0,0,0,0,0,0,0,0},
14	{0,0,0,0,0,0,9,9,0,0,0,0,0,0,0,0,0,0,0,0},
15	{0,0,0,0,0,0,9,9,9,0,0,0,0,0,0,0,0,0,0,0},
16	{0,0,3,0,0,0,9,9,9,0,0,0,0,0,0,0,0,0,0,0},
17	{0,0,0,0,0,0,0,0,0,9,9,9,0,0,0,0,0,0,0,0},
18	{0,0,0,0,0,0,0,0,0,9,9,9,0,0,0,0,0,0,0,0},
19	{0,0,0,0,0,0,0,0,0,9,9,9,0,0,0,0,0,0,0,0},
20	{0,0,0,0,0,0,0,0,0,9,9,9,0,0,0,0,0,0,0,0},
21	{0,0,0,0,0,0,0,0,0,9,9,9,9,0,0,0,0,0,0,0},
22	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,9,9,0},
23	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,0,9,9,0},
24	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,9,0,9,0},
25	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,9,9,0,0},
26	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,9,9,9,0},
27	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,0,0,9,9,0},
28	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,0,9,0,9,0},
29	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,0,9,9,9,0},
30	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,0,0,0,9,0},
31	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,0,9,9,9,0},
32	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,9,0,9,9,0},
33	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,9,9,0,0,0},
34	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,9,9,9,9,0},
35	{0,9,9},
36	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,0,0,0,0,0,9,9},

37	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,0,0,0,0,0,0},
38	{0,9,9},
39	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,0,0,0,0,9,9},
40	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,0,0,0,0,0,0},
41	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,0,0,0,0,0,0,0},
42	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,3,0,0,0,0,0,0},
43	{0,0,0,0,0,0,0,0,0,0,0,0,0,9,0,3,0,0,0,0,0,0},
44	{0,0,0,0,0,0,0,0,0,0,0,0,0,9,9,0,0,0,0,0,0,0},
45	{0,0,0,0,0,0,0,0,0,0,0,0,0,9,9,3,0,0,0,0,0,0}};
46	
47	String kelaslatih[] = {"Busuk Daun",
48	"Busuk Daun",
49	"Busuk Daun",
50	"Busuk Daun",
51	"Busuk Daun",
52	"Layu Bakteri",
53	"Layu Bakteri",
54	"Layu Bakteri",
55	"Layu Bakteri",
56	"Layu Bakteri",
57	"Layu Bakteri",
58	"Busuk Umbi",
59	"Busuk Umbi",
60	"Busuk Umbi",
61	"Busuk Umbi",
62	"Busuk Umbi",
63	"Fusarium",
64	"Fusarium",
65	"Fusarium",
66	"Fusarium",
67	"Fusarium",
68	"Potato Virus",
69	"Potato Virus",
70	"Potato Virus",
71	"Potato Virus",
72	"Potato Virus",
73	"Potato Virus",
74	"Potato Virus",
75	"Potato Virus",
76	"Potato Virus",
77	"Potato Virus",

78	"Potato Virus",
79	"Potato Virus",
80	"Potato Virus",
81	"Kudis",
82	"Kudis",
83	"Kudis",
84	"Kudis",
85	"Kudis",
86	"Kudis",
87	"Bercak Kering",
88	"Bercak Kering",
89	"Bercak Kering",
90	"Bercak Kering",
91	"Bercak Kering");
92	double k = Double.parseDouble(NilaiK.getText().toString());
93	double uji[][] = {{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}};
94	String Hasil = "Nilai K Tidak Boleh Lebih Dari " + latih.length;
95	if (k > latih.length) {
96	showMessage("Hasil", Hasil);
97	} else {
98	if (gejala1.isChecked()) {
99	uji[0][0] = 9;
100	}
101	if (gejala2.isChecked()) {
102	uji[0][1] = 9;
103	}
104	if (gejala3.isChecked()) {
105	uji[0][2] = 5;
106	}
107	if (gejala4.isChecked()) {
108	uji[0][3] = 9;
109	}
110	if (gejala5.isChecked()) {
111	uji[0][4] = 9;
112	}
113	if (gejala6.isChecked()) {
114	uji[0][5] = 9;
115	}
116	if (gejala7.isChecked()) {
117	uji[0][6] = 9;
118	}

119	if (gejala8.isChecked()) {
120	uji[0][7] = 9;
121	}
122	if (gejala9.isChecked()) {
123	uji[0][8] = 9;
124	}
125	if (gejala10.isChecked()) {
126	uji[0][9] = 9;
127	}
128	if (gejala11.isChecked()) {
129	uji[0][10] = 9;
130	}
131	if (gejala12.isChecked()) {
132	uji[0][11] = 9;
133	}
134	if (gejala13.isChecked()) {
135	uji[0][12] = 9;
136	}
137	if (gejala14.isChecked()) {
138	uji[0][13] = 9;
139	}
140	if (gejala15.isChecked()) {
141	uji[0][14] = 9;
142	}
143	if (gejala16.isChecked()) {
144	uji[0][15] = 3;
145	}
146	if (gejala17.isChecked()) {
147	uji[0][16] = 9;
148	}
149	if (gejala18.isChecked()) {
150	uji[0][17] = 9;
151	}
152	if (gejala19.isChecked()) {
153	uji[0][18] = 9;
154	}
155	if (gejala20.isChecked()) {
156	uji[0][19] = 9;
157	}
158	if (gejala21.isChecked()) {
159	uji[0][20] = 9;

160	}
161	if (gejala22.isChecked()) {
162	uji[0][21] = 9;
163	}
164	if (gejala23.isChecked()) {
165	uji[0][22] = 9;
166	}

Source Code 0.1 Inisialisasi Variabel

Implementasi algoritma yang pertama adalah inisialisasi variabel yang digunakan untuk menginisialisasikan data latih, data disimpan dalam bentuk array dimana variabel latih menyimpan nilai bobot setiap gejala pada setiap data latih, kemudian variabel kelaslatih berfungsi untuk menyimpan kelas setiap data latih, variable k akan menyimpan nilai K yang dimasukkan pengguna dan variable uji akan digunakan untuk menyimpan bobot gejala sesuai masukan pengguna.

1	double jarak[] = new double[latih.length];
2	double temp = 0;
3	for (int i = 0; i < uji.length; i++) {
4	for (int j = 0; j < latih.length; j++) {
5	for (int k = 0; k < uji[0].length; k++) {
6	temp = Math.pow(latih[j][k] - uji[i][k], 2);
7	jarak[j] += temp;
8	}
9	}
10	}
11	for (int i = 0; i < latih.length; i++) {
12	jarak[i] = Math.sqrt(jarak[i]);
13	}

Source Code 0.2 Penghitungan Euclidian Distance

Implementasi algoritma yang kedua adalah perhitungan nilai euclidian distance antara setiap data latih terhadap data uji, hasil perhitungan euclidian distanced setiap data latih terhadap data uji akan disimpan pada variable jarak.

1	double temp2 = 0;
2	String temp3 = "";
3	for (int i = 0; i < jarak.length; i++) {
4	for (int j = 1; j < (jarak.length - i); j++) {
5	if (jarak[j - 1] > jarak[j]) {
6	//swap elements
7	temp2 = jarak[j - 1];
8	jarak[j - 1] = jarak[j];

9	jarak[j] = temp2;
10	temp3 = kelaslatih[j - 1];
11	kelaslatih[j - 1] = kelaslatih[j];
12	kelaslatih[j] = temp3;
13	}
14	
15	}
16	}

Source Code 0.3 Pengurutan Euclidian distance

Implementasi Algoritma selanjutnya adalah pengurutan euclidian distance, setelah sebelumnya didapatkan nilai euclidian distance setiap data latih terhadap data uji, kemudian data-data tersebut diurutkan dari yang terkecil hingga terbesar .

1	double jumlahkelas[] = new double[7];
2	for (int i = 0; i < jumlahkelas.length; i++) {
3	jumlahkelas[i] = 0;
4	}
5	
6	
7	for (int i = 0; i < k; i++) {
8	if (kelaslatih[i].equals("Busuk Daun")) {
9	jumlahkelas[0] += 1;
10	}
11	if (kelaslatih[i].equals("Layu Bakteri")) {
12	jumlahkelas[1] += 1;
13	}
14	if (kelaslatih[i].equals("Busuk Umbi")) {
15	jumlahkelas[2] += 1;
16	}
17	if (kelaslatih[i].equals("Fusarium")) {
18	jumlahkelas[3] += 1;
19	}
20	if (kelaslatih[i].equals("Bercak Kering")) {
21	jumlahkelas[4] += 1;
22	}
23	if (kelaslatih[i].equals("Potato Virus")) {
24	jumlahkelas[5] += 1;
25	}
26	if (kelaslatih[i].equals("Kudis")) {
27	jumlahkelas[6] += 1;
28	}

29
30 }

Source Code 0.4 Pengambilan Data Sebanyak K dan Penghitungan Jumlah Masing – Masing Kelas

Implementasi Algoritma selanjutnya adalah implementasi algoritma untuk menghitung jumlah masing – masing kelas setelah dilakukannya pengurutan euclidian distance dan diambil data sebanyak K.

1	String namakelas[] = {"Busuk Daun","Layu Bakteri","Busuk Umi", "Fusarium", "Bercak Kering", "Potato Virus", "Kudis"};
2	double simpan4 = 0;
3	String simpan5 = "";
4	for (int i = 0; i < jumlahkelas.length; i++) {
5	for (int j = 1; j < (jumlahkelas.length - i); j++) {
6	if (jumlahkelas[j - 1] < jumlahkelas[j]) {
7	//swap elements
8	simpan4 = jumlahkelas[j - 1];
9	jumlahkelas[j - 1] = jumlahkelas[j];
10	jumlahkelas[j] = simpan4;
11	simpan5 = namakelas[j - 1];
12	namakelas[j - 1] = namakelas[j];
13	namakelas[j] = simpan5;
14	}
15	
16	}
17	}
18	Output = "Sebelum Sorting" + "\n" + Output1 + "\n" + "\n" + "Setelah Sorting" + "\n" + Output2 + "\n" + "\n" + "Tetangga Terdekat" + "\n" + Output3
19	+ "\n" + "\n" + "Jumlah Setiap Kelas" + "\n" + Output4 + "\n" + "\n" + "Kelas setelah disorting" + "\n" + Output5 + "\n" + "\n" +
20	"Kesimpulan" + "\n" + "\n" + "Tanaman Kentang Terserang "
21	+ namakelas[0];
22	showMessage("Hasil", Output);

Source Code 0.5 Pencarian jumlah kelas terbanyak dan penentuan kelas

Proses akhir dari algoritma ini adalah pencarian jumlah kelas terbanyak yang akan dijadikan sebagai kelas data uji, pada source code diatas ditunjukkan bahwa mula-mula jumlah setiap kelas dihitung kemudian diurutkan terlebih dahulu dari kelas dengan jumlah terbanyak hingga paling sedikit, setelah itu kelas yang menjadi urutan pertama lah yang menjadi kelas data uji.

5.3.2 Implementasi Antarmuka

Antarmuka sistem pakar diagnosis penyakit tanaman kentang ini digunakan oleh pengguna untuk berinteraksi dengan perangkat lunak. Antarmuka halaman terbagi menjadi 3 halaman yaitu antarmuka halaman splash screen, antarmuka halaman diagnosis dan antarmuka halaman hasil diagnosis.

5.3.2.1 Tampilan Antarmuka Splash Screen

Tampilan antarmuka halaman utama yang ditunjukkan pada Gambar 5.1 merupakan halaman awal ketika pengguna membuka sistem, pada halaman ini hanya menampilkan nama aplikasi kemudian selang beberapa detik akan masuk ke halaman diagnosis.



Gambar 0.1 Tampilan Antarmuka Splash Screen

5.3.2.2 Tampilan Antarmuka Diagnosis

Tampilan antarmuka diagnosis yang ditunjukkan pada Gambar 5.2 merupakan tampilan halaman yang akan muncul ketika pengguna memilih menu Diagnosis pada halaman utama dimana pada halaman ini akan ditampilkan textbox untuk nilai k, list checkbox gejala penyakit tanaman kentang dan juga sebuah tombol diagnosis.

Masukkan Nilai K

Nilai K

Pilih Gejala dan Kondisinya

- ☐ Timbul bercak kecil berwarna hijau kelabu dan agak basah kemudian berkembang warna bercak berubah menjadi coklat sampai hitam dengan bagian tepi berwarna putih yang merupakan sporangium
- ☐ Daun membusuk
- ☐ Tanaman mati
- ☐ Daun muda pada pokok tanaman layu
- ☐ Daun tua dan daun bagian bawah menguning
- ☐ Daun menguning, daun menggulung, daun layu dan daun kering
- ☐ Pada bagian tanaman yang berada dalam tanah terdapat bercak-bercak berwarna coklat
- ☐ Infeksi akan menyebabkan akar dan umbi muda busuk
- ☐ Tanaman layu
- ☐ Umbi busuk

Gambar 0.2 Tampilan Antarmuka Halaman Diagnosis

5.3.2.3 Tampilan Antarmuka Hasil Diagnosis

Tampilan antarmuka hasil diagnosis yang ditunjukkan pada Gambar 5.3 merupakan tampilan halaman yang muncul ketika pengguna mengklik tombol diagnosis pada halaman diagnosis, pada halaman ini pengguna akan mendapatkan informasi hasil perhitungan dan diagnosis tanaman kentang.

Hasil

2. Layu Bakteri
3. Busuk Umbi
4. Potato Virus
5. Potato Virus

Jumlah Setiap Kelas

Busuk Daun = 2.0
 Layu Bakteri = 2.0
 Busuk Umbi = 1.0
 Fusarium = 0.0
 Bercak Kering = 0.0
 Potato Virus = 0.0
 Kudis = 0.0

Kelas setelah disorting

1. Layu Bakteri
2. Potato Virus
3. Busuk Umbi
4. Busuk Daun
5. Fusarium
6. Bercak Kering
7. Kudis

Kesimpulan

Tanaman Kentang Terserang Layu Bakteri

Gambar 0.3 Tampilan Antarmuka Halaman Hasil Diagnosis

BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini akan dibahas mengenai pengujian algoritma yang terdiri dari skenario pengujian dan analisis hasil pengujian sistem hama dan penyakit tanaman kentang. Pengujian yang dilakukan adalah pengujian akurasi.

6.1 Pengujian Akurasi

Pengujian ini dilakukan dengan menguji tingkat keakuratan atau kesesuaian dari data uji yang didapatkan oleh pakar dengan hasil output dari sistem. Data testing yang didapatkan sebanyak 70 data. Kemudian akan dilakukan percobaan dengan masukan sesuai data uji dengan jumlah K yang berbeda-beda, nilai K yang digunakan adalah 4,7,10 dan 13, kemudian nilai akurasi akan dihitung berdasarkan jumlah data uji yang memiliki keluaran sama dengan diagnosis pakar.

Untuk mendapatkan nilai akurasi akan dilakukan uji kecocokan antara keluaran sistem dengan data uji yang didapatkan dari pakar, dimana hasil uji sistem ditunjukkan oleh Tabel 6.1 hingga 6.4.

6.1.1 Pengujian dengan nilai K = 4

Pengujian yang pertama adalah pengujian dengan nilai K = 4, data uji yang digunakan sebanyak 70 data uji. Hasil pengujian dengan nilai K = 4 ditunjukkan oleh tabel 6.1.

Tabel 0.1 Hasil Uji K = 4

No	Hasil Pakar	Hasil Sistem
1	Busuk Daun	Busuk Daun
2	Busuk Daun	Layu Bakteri
3	Busuk Daun	Busuk Daun
4	Busuk Daun	Busuk Daun
5	Busuk Daun	Busuk Daun
6	Busuk Daun	Busuk Daun
7	Busuk Daun	Busuk Daun
8	Busuk Daun	Busuk Daun
9	Busuk Daun	Busuk Daun
10	Busuk Daun	Busuk Daun
11	Layu Bakteri	Layu Bakteri
12	Layu Bakteri	Layu Bakteri
13	Layu Bakteri	Layu Bakteri
14	Layu Bakteri	Layu Bakteri
15	Layu Bakteri	Layu Bakteri
16	Layu Bakteri	Layu Bakteri
17	Layu Bakteri	Layu Bakteri

18	Layu Bakteri	Layu Bakteri
19	Layu Bakteri	Layu Bakteri
20	Busuk Umbi	Busuk Umbi
21	Busuk Umbi	Busuk Umbi
22	Busuk Umbi	Busuk Umbi
23	Busuk Umbi	Busuk Umbi
24	Busuk Umbi	Busuk Umbi
25	Busuk Umbi	Kudis
26	Busuk Umbi	Busuk Umbi
27	Busuk Umbi	Busuk Umbi
28	Busuk Umbi	Busuk Umbi
29	Fusarium	Fusarium
30	Fusarium	Fusarium
31	Fusarium	Fusarium
32	Fusarium	Fusarium
33	Fusarium	Fusarium
34	Fusarium	Fusarium
35	Fusarium	Fusarium
36	Fusarium	Fusarium
37	Fusarium	Fusarium
38	Fusarium	Fusarium
39	Bercak Kering	Bercak Kering
40	Bercak Kering	Bercak Kering
41	Bercak Kering	Bercak Kering
42	Bercak Kering	Bercak Kering
43	Bercak Kering	Bercak Kering
44	Bercak Kering	Bercak Kering
45	Bercak Kering	Bercak Kering
46	Potato Virus	Potato Virus
47	Potato Virus	Potato Virus
48	Potato Virus	Potato Virus
49	Potato Virus	Potato Virus
50	Potato Virus	Potato Virus
51	Potato Virus	Potato Virus
52	Potato Virus	Potato Virus
53	Potato Virus	Potato Virus
54	Potato Virus	Potato Virus
55	Potato Virus	Potato Virus
56	Potato Virus	Potato Virus
57	Potato Virus	Potato Virus
58	Potato Virus	Potato Virus

59	Potato Virus	Potato Virus
60	Potato Virus	Potato Virus
61	Potato Virus	Potato Virus
62	Potato Virus	Potato Virus
63	Kudis	Kudis
64	Kudis	Kudis
65	Kudis	Kudis
66	Kudis	Kudis
67	Kudis	Kudis
68	Kudis	Kudis
69	Kudis	Kudis
70	Kudis	Kudis

Dari hasil pengujian pada tabel 6.1 didapatkan sebanyak 2 data uji yang memiliki hasil klasifikasi berbeda dengan kelas sebenarnya, dari hasil tersebut kemudian akan dihitung akurasi menggunakan persamaan (2-2) sehingga dihasilkan akurasi sebagai berikut :

$$\begin{aligned}
 \text{Akurasi} &= \frac{68}{70} \times 100\% \\
 &= 97,14\%
 \end{aligned}$$

6.1.2 Pengujian dengan nilai K = 7

Pengujian yang pertama adalah pengujian dengan nilai K = 7, data uji yang digunakan sebanyak 70 data uji. Hasil pengujian dengan nilai K = 7 ditunjukkan oleh tabel 6.2.

Tabel 0.2 Hasil Uji K = 7

No	Hasil Pakar	Hasil Sistem
1	Busuk Daun	Busuk Daun
2	Busuk Daun	Bercak Kering
3	Busuk Daun	Busuk Daun
4	Busuk Daun	Busuk Daun
5	Busuk Daun	Busuk Daun
6	Busuk Daun	Busuk Daun
7	Busuk Daun	Busuk Daun
8	Busuk Daun	Busuk Daun
9	Busuk Daun	Busuk Daun
10	Busuk Daun	Busuk Daun
11	Layu Bakteri	Bercak Kering
12	Layu Bakteri	Layu Bakteri
13	Layu Bakteri	Layu Bakteri

14	Layu Bakteri	Layu Bakteri
15	Layu Bakteri	Bercak Kering
16	Layu Bakteri	Layu Bakteri
17	Layu Bakteri	Layu Bakteri
18	Layu Bakteri	Layu Bakteri
19	Layu Bakteri	Bercak Kering
20	Busuk Umbi	Busuk Umbi
21	Busuk Umbi	Busuk Umbi
22	Busuk Umbi	Busuk Umbi
23	Busuk Umbi	Busuk Umbi
24	Busuk Umbi	Layu Bakteri
25	Busuk Umbi	Busuk Umbi
26	Busuk Umbi	Busuk Umbi
27	Busuk Umbi	Busuk Umbi
28	Busuk Umbi	Busuk Umbi
29	Fusarium	Fusarium
30	Fusarium	Fusarium
31	Fusarium	Fusarium
32	Fusarium	Fusarium
33	Fusarium	Fusarium
34	Fusarium	Fusarium
35	Fusarium	Fusarium
36	Fusarium	Fusarium
37	Fusarium	Fusarium
38	Fusarium	Fusarium
39	Bercak Kering	Bercak Kering
40	Bercak Kering	Bercak Kering
41	Bercak Kering	Bercak Kering
42	Bercak Kering	Bercak Kering
43	Bercak Kering	Bercak Kering
44	Bercak Kering	Bercak Kering
45	Bercak Kering	Bercak Kering
46	Potato Virus	Potato Virus
47	Potato Virus	Potato Virus
48	Potato Virus	Potato Virus
49	Potato Virus	Potato Virus
50	Potato Virus	Potato Virus
51	Potato Virus	Potato Virus
52	Potato Virus	Potato Virus
53	Potato Virus	Potato Virus
54	Potato Virus	Potato Virus

55	Potato Virus	Potato Virus
56	Potato Virus	Potato Virus
57	Potato Virus	Potato Virus
58	Potato Virus	Potato Virus
59	Potato Virus	Potato Virus
60	Potato Virus	Potato Virus
61	Potato Virus	Potato Virus
62	Potato Virus	Potato Virus
63	Kudis	Kudis
64	Kudis	Kudis
65	Kudis	Kudis
66	Kudis	Kudis
67	Kudis	Kudis
68	Kudis	Kudis
69	Kudis	Kudis
70	Kudis	Kudis

Dari hasil pengujian pada tabel 6.2 didapatkan sebanyak 5 data uji yang memiliki hasil klasifikasi berbeda dengan kelas sebenarnya, dari hasil tersebut kemudian akan dihitung akurasinya menggunakan persamaan (2-2) sehingga dihasilkan akurasi sebagai berikut :

$$\begin{aligned}
 \text{Akurasi} &= \frac{65}{70} \times 100\% \\
 &= 92,86\%
 \end{aligned}$$

6.1.3 Pengujian dengan nilai K = 10

Pengujian yang pertama adalah pengujian dengan nilai K = 10, data uji yang digunakan sebanyak 70 data uji. Hasil pengujian dengan nilai K = 10 ditunjukkan oleh tabel 6.3.

Tabel 0.3 Data Uji K = 10

No	Hasil Pakar	Hasil Sistem
1	Busuk Daun	Busuk Daun
2	Busuk Daun	Layu Bakteri
3	Busuk Daun	Busuk Daun
4	Busuk Daun	Busuk Daun
5	Busuk Daun	Busuk Daun
6	Busuk Daun	Busuk Daun
7	Busuk Daun	Busuk Daun
8	Busuk Daun	Busuk Daun
9	Busuk Daun	Busuk Daun
10	Busuk Daun	Busuk Daun
11	Layu Bakteri	Layu Bakteri
12	Layu Bakteri	Layu Bakteri
13	Layu Bakteri	Layu Bakteri
14	Layu Bakteri	Layu Bakteri
15	Layu Bakteri	Layu Bakteri
16	Layu Bakteri	Layu Bakteri
17	Layu Bakteri	Layu Bakteri
18	Layu Bakteri	Layu Bakteri
19	Layu Bakteri	Layu Bakteri
20	Busuk Umbi	Busuk Umbi
21	Busuk Umbi	Busuk Umbi
22	Busuk Umbi	Busuk Umbi
23	Busuk Umbi	Busuk Umbi
24	Busuk Umbi	Busuk Umbi
25	Busuk Umbi	Busuk Umbi
26	Busuk Umbi	Busuk Umbi
27	Busuk Umbi	Busuk Umbi
28	Busuk Umbi	Busuk Umbi
29	Fusarium	Fusarium
30	Fusarium	Fusarium
31	Fusarium	Fusarium
32	Fusarium	Fusarium
33	Fusarium	Fusarium
34	Fusarium	Fusarium
35	Fusarium	Fusarium
36	Fusarium	Fusarium
37	Fusarium	Fusarium
38	Fusarium	Fusarium
39	Bercak Kering	Bercak Kering

40	Bercak Kering	Bercak Kering
41	Bercak Kering	Bercak Kering
42	Bercak Kering	Bercak Kering
43	Bercak Kering	Bercak Kering
44	Bercak Kering	Bercak Kering
45	Bercak Kering	Bercak Kering
46	Potato Virus	Potato Virus
47	Potato Virus	Potato Virus
48	Potato Virus	Potato Virus
49	Potato Virus	Potato Virus
50	Potato Virus	Potato Virus
51	Potato Virus	Potato Virus
52	Potato Virus	Potato Virus
53	Potato Virus	Potato Virus
54	Potato Virus	Potato Virus
55	Potato Virus	Bercak Kering
56	Potato Virus	Potato Virus
57	Potato Virus	Potato Virus
58	Potato Virus	Potato Virus
59	Potato Virus	Potato Virus
60	Potato Virus	Potato Virus
61	Potato Virus	Potato Virus
62	Potato Virus	Potato Virus
63	Kudis	Kudis
64	Kudis	Kudis
65	Kudis	Kudis
66	Kudis	Kudis
67	Kudis	Kudis
68	Kudis	Kudis
69	Kudis	Kudis
70	Kudis	Kudis

Dari hasil pengujian pada tabel 6.3 didapatkan sebanyak 2 data uji yang memiliki hasil klasifikasi berbeda dengan kelas sebenarnya, dari hasil tersebut kemudian akan dihitung akurasi menggunakan persamaan (2-2) sehingga dihasilkan akurasi sebagai berikut :

$$\begin{aligned}
 \text{Akurasi} &= \frac{68}{70} \times 100\% \\
 &= 97,14\%
 \end{aligned}$$

6.1.4 Pengujian dengan nilai K = 13

Pengujian yang pertama adalah pengujian dengan nilai K = 13, data uji yang digunakan sebanyak 70 data uji. Hasil pengujian dengan nilai K = 13 ditunjukkan oleh tabel 6.4.

Tabel 0.4 Hasil Uji K = 13

No	Hasil Pakar	Hasil Sistem
1	Busuk Daun	Busuk Daun
2	Busuk Daun	Layu Bakteri
3	Busuk Daun	Busuk Daun
4	Busuk Daun	Busuk Daun
5	Busuk Daun	Busuk Daun
6	Busuk Daun	Busuk Daun
7	Busuk Daun	Busuk Daun
8	Busuk Daun	Busuk Daun
9	Busuk Daun	Busuk Daun
10	Busuk Daun	Busuk Daun
11	Layu Bakteri	Layu Bakteri
12	Layu Bakteri	Layu Bakteri
13	Layu Bakteri	Layu Bakteri
14	Layu Bakteri	Layu Bakteri
15	Layu Bakteri	Layu Bakteri
16	Layu Bakteri	Layu Bakteri
17	Layu Bakteri	Layu Bakteri
18	Layu Bakteri	Layu Bakteri
19	Layu Bakteri	Layu Bakteri
20	Busuk Umbi	Layu Bakteri
21	Busuk Umbi	Layu Bakteri
22	Busuk Umbi	Busuk Umbi
23	Busuk Umbi	Busuk Umbi
24	Busuk Umbi	Layu Bakteri
25	Busuk Umbi	Layu Bakteri
26	Busuk Umbi	Busuk Umbi
27	Busuk Umbi	Busuk Umbi
28	Busuk Umbi	Busuk Umbi
29	Fusarium	Layu Bakteri
30	Fusarium	Layu Bakteri
31	Fusarium	Fusarium
32	Fusarium	Layu Bakteri
33	Fusarium	Fusarium
34	Fusarium	Fusarium
35	Fusarium	Layu Bakteri

36	Fusarium	Fusarium
37	Fusarium	Fusarium
38	Fusarium	Fusarium
39	Bercak Kering	Layu Bakteri
40	Bercak Kering	Bercak Kering
41	Bercak Kering	Bercak Kering
42	Bercak Kering	Bercak Kering
43	Bercak Kering	Bercak Kering
44	Bercak Kering	Bercak Kering
45	Bercak Kering	Bercak Kering
46	Potato Virus	Layu Bakteri
47	Potato Virus	Potato Virus
48	Potato Virus	Layu Bakteri
49	Potato Virus	Potato Virus
50	Potato Virus	Potato Virus
51	Potato Virus	Layu Bakteri
52	Potato Virus	Potato Virus
53	Potato Virus	Potato Virus
54	Potato Virus	Potato Virus
55	Potato Virus	Layu Bakteri
56	Potato Virus	Potato Virus
57	Potato Virus	Potato Virus
58	Potato Virus	Potato Virus
59	Potato Virus	Potato Virus
60	Potato Virus	Potato Virus
61	Potato Virus	Potato Virus
62	Potato Virus	Potato Virus
63	Kudis	Kudis
64	Kudis	Kudis
65	Kudis	Kudis
66	Kudis	Kudis
67	Kudis	Kudis
68	Kudis	Kudis
69	Kudis	Kudis
70	Kudis	Kudis

Dari hasil pengujian 6.4 didapatkan sebanyak 14 data uji yang memiliki hasil klasifikasi berbeda dengan kelas sebenarnya, dari hasil tersebut kemudian akan dihitung akurasi menggunakan persamaan (2-2) sehingga dihasilkan akurasi sebagai berikut :

$$\begin{aligned}
 \text{Akurasi} &= \frac{56}{70} \times 100\% \\
 &= 80\%
 \end{aligned}$$

6.2 Analisis Pengujian Akurasi

Menurut hasil pengujian dengan nilai K yang digunakan adalah 4, 7, 10, dan 13 dari Tabel 6.1 sampai 6.4 dapat dianalisis bahwa terjadi beberapa ketidaksesuaian data antara output hasil pakar dan hasil sistem.

6.2.1 Analisis pengujian dengan nilai K=4

Presentase pengujian akurasi dengan nilai K=4 dihasilkan presentase sebesar 97,14%. Dari hasil diagnosis sistem didapat 2 kesalahan yang disebabkan oleh gejala pada dua penyakit sedangkan satu output penyakit yang dihasilkan oleh sistem. Sehingga hal ini dapat disimpulkan bahwasanya semakin banyak gejala spesifik yang digunakan maka akurasi semakin tinggi, semakin banyak gejala umum yang digunakan maka akurasi semakin rendah.

6.2.2 Analisis pengujian dengan nilai K=7

Dari hasil perhitungan akurasi didapatkan presentase sebesar 92,86%. Terdapat 5 kesalahan hasil diagnosis sistem, kesalahan terjadi disebabkan karena gejala dimiliki oleh dua penyakit sedangkan sistem hanya dapat menghasilkan satu output penyakit. Dapat dikatakan semakin banyak gejala spesifik yang digunakan maka akurasi semakin tinggi, semakin banyak gejala umum yang digunakan maka akurasi semakin rendah.

6.2.3 Analisis pengujian dengan nilai K=10

Perhitungan persentase akurasi nilai K=10 adalah 97,14%. Hal ini menunjukkan adanya 2 kesalahan sistem diagnosis yang terjadi karena gejala dimiliki oleh dua penyakit namun satu output penyakit yang dihasilkan. Dapat dikatakan semakin banyak gejala spesifik yang digunakan maka akurasi semakin tinggi, semakin banyak gejala umum yang digunakan maka akurasi semakin rendah.

6.2.4 Analisis pengujian dengan nilai K=13

Perhitungan akurasi didapatkan presentase sebesar 80%. Terdapat 14 kesalahan hasil diagnosis sistem, penyebab kesalahan yang terjadi sama yaitu karena gejala dimiliki oleh dua penyakit sedangkan sistem hanya dapat menghasilkan satu output penyakit. Dapat dikatakan semakin banyak gejala spesifik yang digunakan maka akurasi semakin tinggi, semakin banyak gejala umum yang digunakan maka akurasi semakin rendah.

Rata-rata hasil analisis perhitungan pada masing-masing nilai didapatkan presentase sebesar 91,785%. Hal ini menunjukkan bahwa metode *K-nearest neighbor* ini sangat baik digunakan karena akurasi yang didapat dalam kategori tinggi.

BAB 7 KESIMPULAN DAN SARAN

7.1 Kesimpulan

Berdasarkan hasil pengujian dan analisis hasil penelitian ini, maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Sistem diagnosis penyakit pada tanaman kentang berhasil diimplentasi dalam bentuk perangkat lunak dengan fungsi melakukan diagnosis penyakit pada tanaman kentang.
2. Metode *K-nearest neighbor* baik digunakan untuk diagnosis tanaman kentang karena menghasilkan rata-rata akurasi sebesar 91,785% dan jumlah K tidak terlalu berpengaruh pada akurasi, karena setelah diuji ternyata semakin banyak nilai K tidak menjamin semakin kecil akurasi, begitu pula sebaliknya.

7.2 Saran

Berdasarkan penelitian yang telah dilakukan, maka saran untuk penelitian selanjutnya adalah sebagai berikut.

- 1 Sistem diagnosis penyakit tanaman kentang ini menghasilkan akurasi yang tinggi tetapi memiliki kelemahan yaitu jika gejala yang dimasukkan hanya terdiri dari satu gejala dan gejanya bukan gejala spesifik suatu penyakit tanaman kentang atau bisa dibilang satu gejala tersebut dapat mewakili lebih dari satu penyakit maka akan terjadi kemungkinan salah diagnosis karena keluaran dari sistem hanya berdasarkan peluang pada data latih, oleh karena itu akan lebih baik jika k-nearest neighbor digabung atau dioptimasi dengan suatu algoritma untuk menanggulangi kekurangan tersebut.
- 2 Sistem diagnosis penyakit tanaman kentang ini menghasilkan akurasi yang tinggi tetapi data latih yang digunakan masih dalam bentuk array sehingga tidak dapat dirubah.

DAFTAR PUSTAKA

- Ariadi, Dio, Kartika Fithriasari. 2017. Klasifikasi Berita Indonesia Menggunakan Metode K-nearest neighborian Classification dan Support Vector Machine dengan Confix Stripping Stemmer. Vol. 4, No.2
- Badan Pusat Staistik. 2015. Buletin Statistik Perdagangan Luar Negeri Impor April 2017. CV. Josevindo
- Badan Pusat Statistik. 2015. Produksi Tanaman Pangan 2015. CV. Tapasuma Ratu Agung
- Gorunescu, F. (2011). *Data Mining Concepts Models and Techniques*. Craiova: Springer.
- Kartika, Santoso, Sutrisno. 2017. Penentuan Siswa Berprestasi Menggunakan Metode *K-Nearest Neighbor* dan *Weighted Product* (Studi Kasus : SMP Negeri 3 Mejayan)
- Leidiyana, Lenny. 2013. Penerapan Algoritma *K-Nearest Neighbor* Untuk Penentuan Resiko Kredit Kepemilikan Kendaraan Bermotor
- Mariana, Novita. Rejeki, R. S. dan Jeffri. 2015. Penerapan Algoritma K-Nn (*Nearest Neighbor*) Untuk Deteksi Penyakit (Kanker Serviks).
- Mustakim. 2016. Algoritma *K-Nearest Neighbor Classification* Sebagai Sistem Prediksi Predikat Prestasi Mahasiswa
- Ompusunggu, E. A. 2017. Identifikasi Penyakit Tanaman Jarak Pagar Menggunakan Metode *Fuzzy K-Nearest Neighbor* (FK-NN)
- Rohman, Abdul. 2014. Model Algoritma *K-Nearest Neighbor* (k-NN) Untuk Prediksi Kelulusan Mahasiswa
- Semangun, Haryono. 1993. Penyakit-Penyakit Tanaman Pangan di Indonesia. Yogyakarta : Gadjah Mada University press
- Simanjuntak, Tri Halomoan, Mahmudy dan Sutrisno. 2017. Implementasi *Modified K-Nearest Neighbor* Dengan Otomatisasi Nilai K Pada Pengklasifikasian Penyakit Tanaman Kedelai.
- Sutojo. T., Mulyanto. E, Suhartono V., 2011. Kecerdasan Buatan. C.V Andi Offset.
- Yudha, Bayu Laksana. Lailil dan Randy. 2018. Klasifikasi Risiko Hipertensi Menggunakan Metode Neighbor Weighted K-Nearest Neighbor (NWKNN).