

PENERAPAN METODE *NEIGHBOR WEIGHTED K-NEAREST NEIGHBOR* DALAM KLASIFIKASI *DIABETES MELLITUS*

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Dendry Zeta Maliha
NIM: 135150200111112



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

PENGESAHAN

PENERAPAN METODE *NEIGHBOR WEIGHTED K-NEAREST NEIGHBOR* DALAM
KLASIFIKASI *DIABETES MELLITUS*

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Dendry Zeta Maliha
NIM: 135150200111112

Skripsi ini telah diuji dan dinyatakan lulus pada
3 Januari 2019

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Edy Santoso, S.Si., M.Kom.
NIP: 19740414 200312 1 004

Muhammad Tanzil Furqon, S.Kom.,
M.CompSc.
NIP: 19820930 200801 1 004

Mengetahui
Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T, M.T, Ph.D
NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

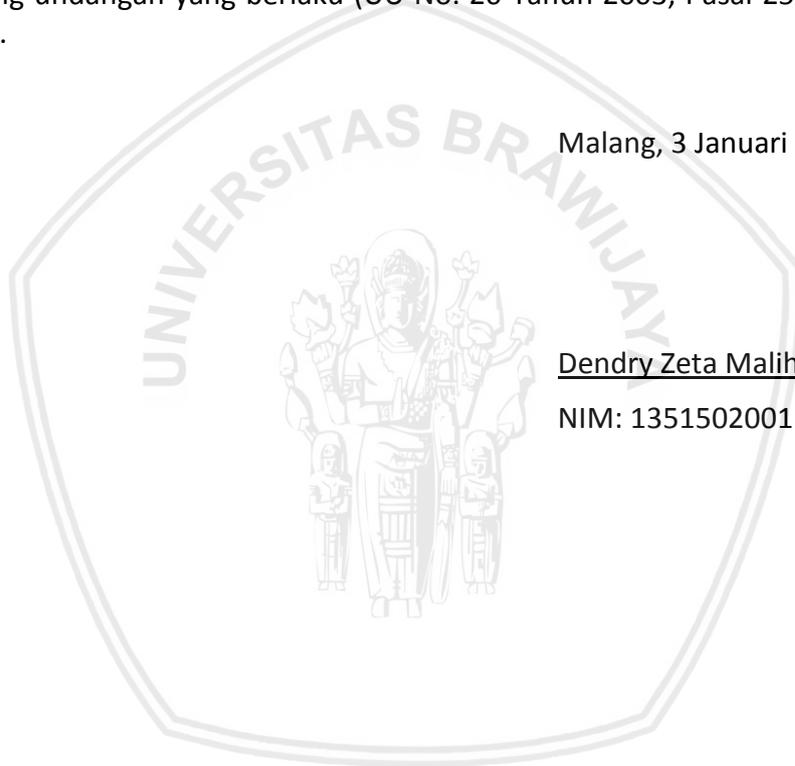
Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan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

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

Malang, 3 Januari 2019

Dendry Zeta Maliha

NIM: 13515020011112



PRAKATA

Segala puji bagi Allah SWT yang Maha Pengasih dan Maha Penyayang atas berkat dan limpahan nikmat-Nya, sehingga penulis dapat menyelesaikan pengerjaan skripsi yang berjudul “Penerapan Metode *Neighbor Weighted K-Nearest Neighbor* Dalam Klasifikasi *Diabetes Mellitus*” dengan baik. Pengerjaan skripsi ini ditujukan untuk memenuhi syarat ujian seminar skripsi dalam rangka memperoleh gelar Sarjana Komputer pada Program Studi Teknik Informatika, Jurusan Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya Malang. Dalam pengerjaan skripsi ini, penulis telah mendapat bantuan dari beberapa pihak, maka dari itu pada kesempatan ini penulis mengucapkan terimakasih yang sebesar-besarnya kepada:

1. Bapak Edy Santoso, S.Si., M.Kom., dan Bapak Muhammad Tanzil Furqon, S.Kom., M.CompSc., selaku dosen pembimbing yang telah meluangkan waktu dan memberi pengarahan dalam pengerjaan skripsi,
2. Bapak Wayan Firdaus Mahmudy, S.Si., M.T., Ph.D., selaku Dekan Fakultas Ilmu Komputer, Universitas Brawijaya,
3. Bapak Tri Astoto Kurniawan, S.T., M.T., Ph.D., selaku Ketua Jurusan Teknik Informatika,
4. Segenap Bapak dan Ibu Dosen yang telah mendidik dan mengajarkan ilmunya kepada penulis selama menempuh pendidikan di Program Studi Teknik Informatika, Fakultas Ilmu Komputer Universitas Brawijaya,
5. Kedua orang tua, Darius Claudius dan Suharnanik yang senantiasa memberi dukungan baik dalam bentuk materi maupun do’a,
6. Saudara penulis, Ike Rosmala Dewi, Ridwan Fajar Nuryudha, Gardini Puspasari, Denny Lukmanul Hakim, dan Rafika Epsilani Cyimbidiana atas do’a dan supportnya untuk penulis,
7. Teman dekat Agit Rachmawan, seluruh sahabat Annisa Puspitawuri, Dainty Resfulany Susanto, Maya Febrianita, Rakhmadina Noviyanti, Ratna Ayu Wijayanti, Selly Kurniasari, Syafitri Hidayatul Annur Aini yang selama ini telah banyak memberikan bantuan dan dukungan kepada penulis,
8. Semua pihak yang telah membantu dalam proses penyusunan skripsi ini.

Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna. Oleh karena itu penulis menerima saran dan kritik yang bersifat membangun demi perbaikan kearah kesempurnaan, baik dalam hal perbaikan penulisan maupun mutu isi skripsi. Penulis berharap semoga skripsi ini dapat bermanfaat bagi penulis dan pembacanya.

Malang, 3 Januari 2019

Penulis

ABSTRAK

Dendry Zeta Maliha, Penerapan Metode *Neighbor Weighted K-Nearest Neighbor* dalam klasifikasi Penyakit *Diabetes Mellitus*

Pembimbing: Edy Santoso, S.Si., M.Kom. dan Muhammad Tanzil Furqon, S.Kom., M.CompSc.

Diabetes mellitus adalah penyakit kronis yang ditimbulkan oleh kelainan sekresi insulin pada peningkatan glukosa yang tidak teratur. *Diabetes mellitus* dapat meningkatkan gula darah dalam tubuh, sehingga terjadi komplikasi penyakit yang dapat menyebabkan beberapa risiko seperti stroke, penyakit jantung, kebutaan, gagal ginjal dan kematian. Menurut *World Health Organization (WHO)*, sebanyak 300 juta populasi di dunia akan terpengaruh oleh *diabetes* pada tahun 2025. Selain itu terdapat beberapa penyakit yang memiliki gejala awal yang hampir mirip dengan penyakit *diabetes mellitus*, jika sampai melakukan kesalahan untuk menganalisa maka akan berakibat fatal pada penderita penyakit *diabetes mellitus*. Oleh sebab itu diperlukan aplikasi yang dapat mempermudah klasifikasi penyakit *diabetes mellitus*. Pada penelitian ini mengusulkan penerapan metode *Neighbor Weighted K-nearest Neighbor* dalam klasifikasi penyakit *diabetes mellitus*. Metode NWKNN menggunakan pembobotan pada kelas data. Hasil penelitian menunjukkan rata-rata akurasi dengan menggunakan nilai $K=15$ dan Nilai $E=2$ mendapatkan akurasi sebesar 92.3% pada data latih sebesar 130 data yang terbagi menjadi 10 *fold* dan data uji sebanyak 13 data pada setiap *fold*.

Kata kunci: *penyakit diabetes mellitus, Klasifikasi, K-Nearest Neighbor, Neighbor Weighted K-Nearest Neighbor*

ABSTRACT

Dendry Zeta Maliha, Penerapan Metode *Neighbor Weighted K-Nearest Neighbor* dalam klasifikasi Penyakit *Diabetes Mellitus*

Supervisors: Edy Santoso, S.Si., M.Kom. dan Muhammad Tanzil Furqon, S.Kom., M.CompSc.

Diabetes mellitus is a chronic disease caused by abnormalities in insulin secretion on irregular increases in glucose. Diabetes mellitus can increase blood sugar in the body, resulting in disease complications that can cause several risks such as stroke, heart disease, blindness, kidney failure and death According to the World Health Organization (WHO), as many as 300 million people in the world will be affected by diabetes in 2025. In addition there are some diseases that have early symptoms that are almost similar to diabetes mellitus, if you make a mistake to analyze it will be fatal in people with diabetes mellitus. Therefore an application is needed that can facilitate the classification of diabetes mellitus. In this study propose the application of the Neighbor Weighted K-nearest Neighbor method in the classification of diabetes mellitus. The NWKNN method uses weighting in the data class. The results showed the average accuracy using the value of $K = 15$ and the value of $E = 2$ obtained an accuracy of 92.3% in the training data of 130 data divided into 10 fold and test data as many as 13 data in each fold.

Keywords: *diabetes mellitus, classification, K-Nearest Neighbor, Neighbor Weighted K-Nearest Neighbor*

DAFTAR ISI

PENGESAHAN	i
PERNYATAAN ORISINALITAS	ii
PRAKATA	iii
ABSTRAK.....	iv
ABSTRACT.....	v
DAFTAR ISI.....	6
DAFTAR TABEL	9
DAFTAR GAMBAR.....	10
DAFTAR KODE PROGRAM	11
BAB 1 PENDAHULUAN	Error! Bookmark not defined.
1.1 Latar belakang	Error! Bookmark not defined.
1.2 Rumusan masalah.....	Error! Bookmark not defined.
1.3 Tujuan	Error! Bookmark not defined.
1.4 Manfaat	Error! Bookmark not defined.
1.5 Batasan masalah.....	Error! Bookmark not defined.
1.6 Sistematika pembahasan.....	Error! Bookmark not defined.
BAB 2 LANDASAN KEPUSTAKAAN	Error! Bookmark not defined.
2.1 Kajian Pustaka.....	Error! Bookmark not defined.
2.2 Penyakit <i>Diabetes Melitus</i>	Error! Bookmark not defined.
2.2.1 Diabetes Tipe 1	Error! Bookmark not defined.
2.2.2 Diabetes Tipe 2	Error! Bookmark not defined.
2.2.3 Gejala Penyakit <i>Diabetes Mellitus</i> ..	Error! Bookmark not defined.
2.3 K-Nearest Neighbor	Error! Bookmark not defined.
2.4 <i>Neighbor Weighted K-Nearest Neighbor</i> (NWKNN)	Error! Bookmark not defined.
BAB 3 METODOLOGI	Error! Bookmark not defined.
3.1 Kajian Pustaka.....	Error! Bookmark not defined.
3.2 Pengumpulan Data	Error! Bookmark not defined.
3.3 Lingkungan Implementasi.....	Error! Bookmark not defined.
3.4 Perancangan	Error! Bookmark not defined.
3.5 Implementasi Sistem	Error! Bookmark not defined.
3.6 Pengujian Sistem	Error! Bookmark not defined.
3.7 Kesimpulan dan Saran	Error! Bookmark not defined.



BAB 4 PERANCANGAN	Error! Bookmark not defined.
4.1 Perancangan Tahapan Proses.....	Error! Bookmark not defined.
4.1.1 Proses Algoritme <i>Neighbor Weighted K-nearest Neighbor</i>	Error! Bookmark not defined.
4.1.2 Proses Pehitungan Ketetangaan dengan Cosine Similarity	Error! Bookmark not defined.
4.1.3 Proses Pembobotan Setiap Tipe Diabetes Mellitus	Error! Bookmark not defined.
4.1.4 Proses Perhitungan Nilai Skor	Error! Bookmark not defined.
4.2 Perhitungan Manual	Error! Bookmark not defined.
4.3 Perancangan Antarmuka	Error! Bookmark not defined.
4.3.1 Desain antarmuka halaman beranda	Error! Bookmark not defined.
4.3.2 Halaman Data.....	Error! Bookmark not defined.
4.3.3 Halaman Hasil Analisis	Error! Bookmark not defined.
4.4 Perancangan Pengujian	Error! Bookmark not defined.
4.4.1 Pengujian <i>K-Fold Cross Validation</i> ..	Error! Bookmark not defined.
4.4.2 Pengujian Pengaruh Nilai E	Error! Bookmark not defined.
BAB 5 IMPLEMENTASI.....	Error! Bookmark not defined.
5.1 Batasan Implementasi	Error! Bookmark not defined.
5.2 Implementasi Algoritme	Error! Bookmark not defined.
5.2.1 Implementasi Perhitungan <i>Cosine Similarity</i>	Error! Bookmark not defined.
5.2.2 Implementasi Algoritme Pengurutan <i>Cosine Similarity</i>	Error! Bookmark not defined.
5.2.3 Implementasi Algoritme Pembobotan Setiap Tipe	Error! Bookmark not defined.
5.2.4 Implementasi Algoritme Perhitungan Skor	Error! Bookmark not defined.
5.3 Implementasi Antarmuka	Error! Bookmark not defined.
5.3.1 Implementasi Antarmuka Beranda.	Error! Bookmark not defined.
5.3.2 Implementasi Antarmuka Perhitungan <i>Neighbor Weighted K-Nearest Neighbor</i>	Error! Bookmark not defined.
5.3.2.1 Implementasi Antarmuka Data Latih	Error! Bookmark not defined.
5.3.2.2 Implementasi Antarmuka Data Uji	Error! Bookmark not defined.
5.3.2.3 Implementasi Antarmuka Perhitungan <i>Cosine similiarity</i>	Error! Bookmark not defined.
5.3.2.4 Implementasi Antamuka Pengurutan Hasil <i>Cosine Similarity</i>	Error! Bookmark not defined.

[_5.3.2.5 Implementasi Antarmuka Hasil Perhitungan Bobot](#)**Error! Bookmark not defined.**

[_5.3.2.6 Implementasi Antarmuka Hasil Perhitungan Skor](#)**Error! Bookmark not defined.**

[_5.3.3 Implementasi Antarmuka Pengujian *K-fold Cross Validation*](#)**Error! Bookmark not defined.**

BAB 6 PENGUJIAN DAN ANALISIS**Error! Bookmark not defined.**

6.1 Pengujian dan Analisis *K-Fold Cross Validation***Error! Bookmark not defined.**

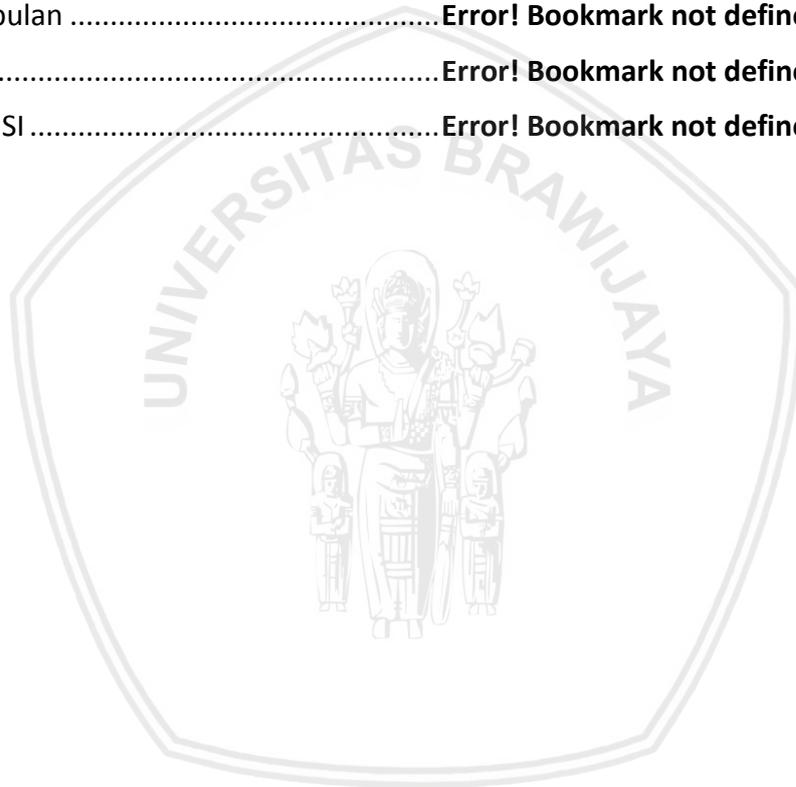
6.2 Pengujian dan Analisis Pengaruh Nilai E..**Error! Bookmark not defined.**

BAB 7 PENUTUP**Error! Bookmark not defined.**

7.1 Kesimpulan**Error! Bookmark not defined.**

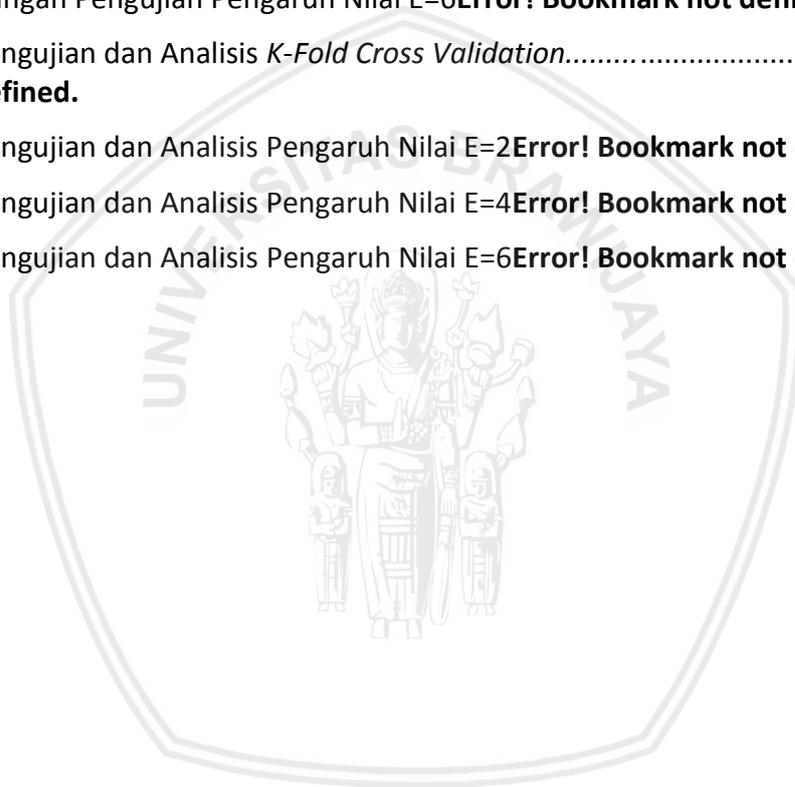
7.2 Saran**Error! Bookmark not defined.**

DAFTAR REFERENSI**Error! Bookmark not defined.**



DAFTAR TABEL

- Tabel 4.1 Dataset klasifikasi penyakit *diabetes mellitus* **Error! Bookmark not defined.**
- Tabel 4.2 Nilai kedekatan ketetangaan..... **Error! Bookmark not defined.**
- Tabel 4.3 Nilai bobot setiap jenis..... **Error! Bookmark not defined.**
- Tabel 4.4 Nilai *CosSim* yang telah diurutkan dengan $K=5$ **Error! Bookmark not defined.**
- Tabel 4.5 Perancangan Pengujian *K-Fold Cross Validation* **Error! Bookmark not defined.**
- Tabel 4.6 Perancangan Pengujian Pengaruh Nilai $E=2$ **Error! Bookmark not defined.**
- Tabel 4.7 Perancangan Pengujian Pengaruh Nilai $E=4$ **Error! Bookmark not defined.**
- Tabel 4.8 Perancangan Pengujian Pengaruh Nilai $E=6$ **Error! Bookmark not defined.**
- Tabel 6.1 Hasil Pengujian dan Analisis *K-Fold Cross Validation*..... **Error! Bookmark not defined.**
- Tabel 6.2 Hasil Pengujian dan Analisis Pengaruh Nilai $E=2$ **Error! Bookmark not defined.**
- Tabel 6.3 Hasil Pengujian dan Analisis Pengaruh Nilai $E=4$ **Error! Bookmark not defined.**
- Tabel 6.4 Hasil Pengujian dan Analisis Pengaruh Nilai $E=6$ **Error! Bookmark not defined.**



DAFTAR GAMBAR

- Gambar 4.1 Tahapan Proses Alir Sistem**Error! Bookmark not defined.**
- Gambar 4.2 Diagram Proses NWKNN**Error! Bookmark not defined.**
- Gambar 4.3 Diagram Perhitungan Ketetangaan.....**Error! Bookmark not defined.**
- Gambar 4.4 Proses Perhitungan Bobot Setiap Jenis.**Error! Bookmark not defined.**
- Gambar 4.5 Diagram Alir Perhitungan Skor.....**Error! Bookmark not defined.**
- Gambar 4.6 Perancangan Halaman Home**Error! Bookmark not defined.**
- Gambar 4.7 Perancangan Halaman Data.....**Error! Bookmark not defined.**
- Gambar 4.8 Perancangan Halaman Identifikasi**Error! Bookmark not defined.**
- Gambar 5.1 Implementasi Antarmuka Beranda.....**Error! Bookmark not defined.**
- Gambar 5.2 Implementasi Antarmuka Data Latih**Error! Bookmark not defined.**
- Gambar 5.3 Implementasi Antarmuka Data Uji**Error! Bookmark not defined.**
- Gambar 5.4 Implementasi Antarmuka Perhitungan *Cosine Similarity***Error! Bookmark not defined.**
- Gambar 5.5 Implementasi Antarmuka Perhitungan *Cosine Similarity***Error! Bookmark not defined.**
- Gambar 5.6 Implementasi Antarmuka Perhitungan *Cosine Similarity***Error! Bookmark not defined.**
- Gambar 5.7 Implementasi Antarmuka Pengurutan Hasil *Cosine Similarity***Error! Bookmark not defined.**
- Gambar 5.8 Implementasi Antarmuka Hasil Perhitungan Bobot**Error! Bookmark not defined.**
- Gambar 5.9 Implementasi Antarmuka Hasil Perhitungan Skor**Error! Bookmark not defined.**
- Gambar 5.10 Implementasi Antarmuka Hasil Perhitungan Skor**Error! Bookmark not defined.**
- Gambar 5.11 Implementasi Antarmuka Pengujian ..**Error! Bookmark not defined.**
- Gambar 5.12 Implementasi Antarmuka Hasil Pengujian**Error! Bookmark not defined.**
- Gambar 6.1 Grafik Pengujian *K-Fold Cross Validation*.....**Error! Bookmark not defined.**
- Gambar 6.2 Grafik Pengujian Pengaruh Nilai $E=2$ **Error! Bookmark not defined.**
- Gambar 6.3 Diagram Pengujian Pengaruh Nilai $E=4$ **Error! Bookmark not defined.**
- Gambar 6.4 Grafik Pengujian Pengaruh Nilai $E=6$ **Error! Bookmark not defined.**



DAFTAR KODE PROGRAM

Kode Program 5.1 Implementasi Algoritme *Cosine Similarity***Error! Bookmark not defined.**

Kode Program 5.2 Implementasi Algoritme Pengurutan *Cosine Similarity***Error! Bookmark not defined.**

Kode Program 5.3 Implementasi Algoritme Pembobotan Setiap Tipe**Error! Bookmark not defined.**

Kode Program 5.4 Implementasi Perhitungan Skor **Error! Bookmark not defined.**





BAB 1 PENDAHULUAN

1.1 Latar belakang

Diabetes mellitus adalah penyakit kronis yang ditimbulkan oleh kelainan sekresi insulin pada peningkatan glukosa yang tidak teratur. *Diabetes mellitus* dapat meningkatkan gula darah dalam tubuh, sehingga terjadi komplikasi penyakit yang dapat menyebabkan beberapa risiko seperti stroke, penyakit jantung, kebutaan, gagal ginjal dan kematian (V & Ravikumar, 2014). Saat ini penyakit *diabetes mellitus* telah menjadi masalah kesehatan global yang menyebabkan beberapa komplikasi kesehatan seperti penyakit *cardio vascular*, gagal ginjal, gangguan penglihatan dan sebagainya. Menurut *World Health Organization* (WHO), sebanyak 300 juta populasi di dunia akan terpengaruh oleh *diabetes* pada tahun 2025. Insulin adalah hormon alami yang disekresi oleh pankreas dalam tubuh manusia. Insulin mengubah gula menjadi molekul sederhana yang digunakan oleh sel-sel tubuh untuk menghasilkan energi. Perubahan cara kerja insulin pada tubuh dapat dipengaruhi oleh kurangnya insulin dan gula yang mulai terakumulasi dalam aliran darah. Akibatnya, glukosa darah mulai meningkat dan berkembang menjadi *diabetes mellitus*. Penyakit *diabetes mellitus* ini dibagi menjadi dua tipe yaitu tipe 1 dan tipe 2. *Diabetes mellitus* tipe 1, terjadi karena tubuh tidak memproduksi insulin dan biasanya didiagnosis pada anak-anak. Sedangkan pada *diabetes mellitus* tipe 2, terjadi karena tidak menghasilkan insulin yang cukup untuk tubuh dan *diabetes* tipe 2 umumnya didiagnosis pada orang dewasa (Saxena, Khan, & Singh, 2014).

Diabetes mellitus adalah salah satu jenis penyakit yang mengakibatkan kematian dan menjadi penyakit faktor keturunan. Sebagian besar masyarakat tidak mempedulikan pola hidup yang sehat. Kesehatan adalah salah satu hal yang terpenting dalam kehidupan sehari-hari, masyarakat kurang menyadari tentang permasalahan penanganan kesehatan yang dapat meningkatkan resiko kematian (Fernanda, Ratnawati, & Adikara, 2017). Sebagian besar masyarakat tidak mengetahui apakah penyakit diabetes yang diderita termasuk pada tipe 1 atau 2, karena minimnya pengetahuan tentang penyakit *diabetes mellitus* dan tidak mengetahui akan bahaya dari penyakit tersebut jika tidak segera tertangani dengan benar. Selain itu terdapat beberapa penyakit yang memiliki gejala awal yang hampir mirip dengan penyakit *diabetes mellitus*, jika sampai melakukan kesalahan untuk menganalisa maka akan berakibat fatal pada penderita penyakit *diabetes mellitus*.

Terdapat salah satu penelitian yang menerapkan metode *Neighbor Weighted K-Nearest Neighbor* pada permasalahan penyimpangan tumbuh kembang pada anak yang dilakukan oleh Rivaldi et al pada tahun 2017. Dalam penelitian tersebut membahas tentang proses tumbuh kembang fisik, mental, dan psikologi yang berlangsung dengan cepat sehingga membutuhkan perhatian yang lebih dari orang tua. Pada masa ini tumbuh kembang pada anak terdapat

penyimpangan dimana proses pertumbuhan dan perkembangan anak terhambat. Penyimpangan tumbuh kembang pada anak yang sering ditemukan adalah *Attention Deficit Hyperactivity Disorder* (ADHD), *autisme*, dan *down syndrome*. Pada penelitian ini mengklasifikasikan penyimpangan tumbuh kembang berdasarkan pada gejala yang muncul. Algoritme *Neighbor Weighted K-Nearest Neighbor* (NWKNN) merupakan algoritme pengembangan dari algoritme KNN. Hasil dari penelitian ini mendapatkan akurasi sebesar 95% dengan menggunakan data latih sebanyak 80 dan data uji sebanyak 30 dengan nilai $K=10$, dan $E=4$ (Rivaldi, Adikara, & Adinugroho, 2018).

Melihat permasalahan yang telah diuraikan di atas maka penulis akan membangun sistem cerdas yang dapat melakukan klasifikasi *diabetes mellitus*. Penulis mengajukan skripsi dengan judul "Penerapan Metode *Neighbor Weighted K-Nearest Neighbor* Dalam Klasifikasi Penyakit *Diabetes Mellitus*". Dengan adanya penelitian ini diharapkan dapat membantu untuk mengklasifikasikan *diabetes mellitus* berdasarkan gejala *diabetes mellitus*, yang dapat mempermudah pengguna untuk mengetahui tipe 1 dan tipe 2 dari penyakit *diabetes mellitus*.

1.2 Rumusan masalah

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, maka permasalahan yang dikaji adalah sebagai berikut :

1. Bagaimana penerapan metode *Neighbor Weighted K-Nearest Neighbor* dalam klasifikasi *diabetes mellitus*?
2. Bagaimana tingkat akurasi pengujian sistem pada metode *Neighbor Weighted K-Nearest Neighbor* dalam penyakit *diabetes mellitus*?

1.3 Tujuan

Tujuan dari penelitian ini adalah sebagai berikut :

1. Menerapkan metode *Neighbor Weighted K-Nearest Neighbor* untuk klasifikasi *diabetes mellitus*.
2. Menguji tingkat akurasi sistem pada penyakit *diabetes mellitus* dengan metode *Neighbor Weighted K-Nearest Neighbor*.

1.4 Manfaat

Manfaat dari penelitian ini diharapkan dapat memberikan informasi mengenai klasifikasi penyakit *diabetes mellitus* yang dapat membantu tenaga medis untuk mendiagnosis penyakit *diabetes mellitus*. Sistem ini juga akan membantu untuk mendeteksi gejala awal penyakit *diabetes mellitus*.

1.5 Batasan masalah

Batasan masalah pada penelitian ini adalah sebagai berikut:

1. Implementasi klasifikasi penyakit *diabetes mellitus* menggunakan 130 data dan 15 atribut yaitu nafsu makan meningkat, sering buang air kecil, peningkatan kehausan, turunnya berat badan, usia (15-20) tahun, faktor keturunan, mulut kering, mudah kelelahan/ kurangnya aktivitas fisik, sering mengantuk, mual/ muntah-muntah, timbulnya luka yang tak kunjung sembuh, gatal-gatal, mengkonsumsi makanan berkolesterol tinggi, obesitas, dan kadar glukosa meningkat. Data tersebut didapatkan dari RSUD Kota Mataram.
2. Bahasa pemrograman yang digunakan pada penelitian ini adalah PHP.

1.6 Sistematika pembahasan

Sistematika penyusunan laporan ditunjukkan untuk mencapai tujuan yang ada meliputi beberapa bab, sebagai berikut:

BAB I PENDAHULUAN

Bab ini berisi latar belakang klasifikasi penyakit *diabetes mellitus* menggunakan metode *Neighbor Weighted K-Nearest Neighbor*, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, dan sistematika pembahasan.

BAB II LANDASAN KEPUSTAKAAN

Pada bab ini untuk berisi tentang dasar teori yang digunakan untuk mendukung dalam penelitian klasifikasi penyakit *diabetes mellitus* menggunakan *Neighbor Weighted K-Nearest Neighbor*. Dasar teori meliputi pengertian penyakit *diabetes mellitus*, pengenalan pola, algoritme *Neighbor Weighted K-Nearest Neighbor*.

BAB III METODOLOGI

Metodologi terdiri dari studi literatur dalam pembuatan sistem untuk klasifikasi penyakit *diabetes mellitus* menggunakan metode *Neighbor Weighted K-Nearest Neighbor*, pengumpulan data, analisa kebutuhan untuk mengetahui kebutuhan dalam pembuatan sistem, perancangan sistem, implementasi sistem, pengujian sistem dan kesimpulan.

BAB IV PERANCANGAN

Pada bab ini berisi tentang hasil perancangan untuk implementasi dari klasifikasi penyakit *diabetes mellitus* menggunakan metode *Neighbor Weighted K-Nearest Neighbor*.

BAB V IMPLEMENTASI

Pada bagian ini menguraikan tentang implementasi klasifikasi penyakit *diabetes mellitus* menggunakan metode *Neighbor Weighted K-Nearest Neighbor*.

BAB VI PENGUJIAN DAN ANALISIS

Pada bab ini membahas tentang pengujian klasifikasi penyakit *diabetes mellitus* menggunakan metode *Neighbor Weighted K-Nearest Neighbor* dengan parameter pengujian.

BAB VII PENUTUP

Bab ini menguraikan kesimpulan yang diperoleh pada sistem klasifikasi penyakit *diabetes mellitus* menggunakan metode *Neighbor Weighted K-Nearest Neighbor* dan menyertakan saran yang dapat digunakan untuk pengembangan selanjutnya.



BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Kajian pustaka merupakan sebuah landasan teori penelitian yang telah dilakukan sebelumnya. Pada kajian pustaka ini akan membahas penelitian mengenai algoritme *Neighbor Weighted K-Nearest Neighbor* yang telah dilakukan sebelumnya.

Penelitian yang pada awalnya diusulkan oleh Tan pada tahun 2005 bertujuan untuk mengkategorikan teks dengan menggunakan algoritme *Neighbor Weighted K-Nearest Neighbor*. Dalam penelitian ini algoritme *Neighbor Weighted K-Nearest Neighbor* (NWKNN) digunakan untuk masalah pengkategorian teks yang tidak seimbang dan dibandingkan dengan kinerja dari algoritme *K-Nearest Neighbor* (KNN). Algoritme NWKNN menyeimbangkan data latih dengan memberikan bobot yang besar untuk tetangga dari kelas minor, sedangkan pada kelas mayoritas akan diberikan bobot yang kecil. Hasil dari penelitian tersebut menunjukkan bahwa algoritme NWKNN mencapai peningkatan kinerja klasifikasi yang signifikan pada data yang tidak seimbang dengan menggunakan eksponen=4 algoritme NWKNN mendapatkan rata-rata *precision* dan *recall* masing-masing sebesar 0.84% dan 0.91% , daripada kinerja dari algoritme KNN yang mendapatkan rata-rata *precision* 0.81% dan *recall* 0.76% (Tan, 2005).

Dalam penelitian lainnya yang menerapkan metode *Neighbor Weighted K-Nearest Neighbor* yang digunakan untuk klasifikasi berdasarkan gejala yang muncul. Dalam penelitian tersebut membahas tentang proses tumbuh kembang fisik, mental, dan psikologi yang berlangsung dengan cepat sehingga membutuhkan perhatian yang lebih dari orang tua. Pada masa ini tumbuh kembang pada anak terdapat penyimpangan dimana proses pertumbuhan dan perkembangan anak terhambat. Penyimpangan tumbuh kembang pada anak yang sering ditemukan adalah *Attention Deficit Hyperactivity Disorder* (ADHD), autisme, *down syndrome*. Pada penelitian ini mengklasifikasikan penyimpangan tumbuh kembang berdasarkan pada gejala yang muncul. Hasil dari penelitian ini mendapatkan akurasi sebesar 95% dengan menggunakan data latih sebanyak 80 dan data uji sebanyak 30 dengan nilai $K = 10$, dan $E = 4$ (Rivaldi, Adikara, & Adinugroho, 2018).

Penelitian lainnya yang telah dilakukan oleh Fadila, et al pada tahun 2016 yang membahas tentang gangguan pertumbuhan dan perkembangan pada anak usia dini. Salah satu gangguan yang sering dialami oleh anak usia dini adalah ADHD (*Attention Deficit Hyperactivity Disorder*). Terdapat 3 jenis ADHD yaitu *Inattention*, *Impulsif*, dan *Hyperactivity*. Pada penelitian ini akan mengidentifikasi jenis ADHD berdasarkan gejala yang muncul menggunakan algoritme *Neighbor Weighted K-Nearest Neighbor* (NWKNN). Algoritme NWKNN digunakan untuk proses pembobotan terhadap jenis yang akan diklasifikasikan. Hasil dari penelitian ini menunjukkan bahwa algoritme NWKNN memiliki akurasi mencapai

95% dengan menggunakan 80 data training, 20 data uji, nilai K=10, dan nilai E=4 (Fadila, Indriati, & Ratnawati, 2016).

2.2 Penyakit *Diabetes Mellitus*

Diabetes mellitus merupakan suatu penyakit yang dapat ditandai dengan peningkatan kadar gula darah secara kontinu, terutama setelah makan. Sumber lain menyatakan bahwa *diabetes mellitus* adalah keadaan *hiperglikemia* kronis yang disertai berbagai kelainan pada metabolisme sebagai gangguan hormonal yang dapat menimbulkan berbagai komplikasi pada mata, ginjal dan pembuluh darah. *Hiperglikemia* adalah komplikasi akut dari berbagai macam gejala dengan *diabetes ketoasidosis* (DKA). Air dalam cairan sel ditarik keluar dari sel-sel yang dimasukkan ke dalam darah dan ginjal, nantinya akan membantu membuang glukosa ke dalam urine. Jika cairan dalam sel yang keluar tidak diganti maka akan muncul efek osmotik karena kadar glukosa tinggi dan hilangnya air kemudian akan mengarah pada dehidrasi. Sel-sel pada tubuh akan semakin dehidrasi karena kadar air didalamnya berkurang atau terkuras. Kondisi elektrolit yang tidak seimbang juga akan mengganggu dan berbahaya.

Diabetes merupakan sebuah kondisi dimana tubuh tidak memproduksi insulin dengan cukup atau tidak merespons zat insulin dengan benar. Insulin adalah hormon yang diproduksi pada pankreas, yang dapat memungkinkan sel untuk mendapatkan glukosa untuk diubah menjadi energi. Pada penyakit diabetes, tubuh tidak merespon insulin dengan baik atau kekurangan insulin. Kondisi tersebut menyebabkan glukosa didalam darah menumpuk dan sering menyebabkan komplikasi. Komplikasi jangka panjang meliputi penyakit *kardiovaskular*, gagal ginjal kronis, kerusakan retina yang dapat menyebabkan kebutaan, serta kerusakan saraf yang dapat menyebabkan impotensi dan risiko amputasi (Sutanto, 2010).

Diabetes mellitus (DM) adalah salah satu masalah kesehatan yang sangat besar. Sebesar 80% orang dengan DM tinggal pada negara yang berpenghasilan rendah dan menengah. Pada tahun 2006, terdapat lebih dari 50 juta orang yang menderita DM di Asia Tenggara. Jumlah penderita DM berusia sekitar 40-59 tahun. Ada beberapa jenis *diabetes mellitus* yaitu Diabetes Mellitus Tipe I dan Diabetes Mellitus Tipe II. Jenis Diabetes Mellitus yang paling banyak diderita adalah Diabetes Mellitus Tipe 2 (Trisnawati & Setyorogo, 2013).

2.2.1 Diabetes Tipe 1

Diabetes tipe 1 merupakan kegagalan tubuh dalam menghasilkan insulin. Diperkirakan ada sekitar 5 sampai 10% penderita diabetes yang didiagnosis menderita diabetes tipe 1. Sebagian besar penderita tipe 1 harus melakukan pengobatan dengan menyuntikkan insulin. Diabetes mellitus tipe 1 juga disebut sebagai *insulin-dependent diabetes mellitus* (IDDM), diabetes yang tergantung pada insulin. Diabetes tipe 1 mempunyai ciri khusus, yaitu hilangnya sel beta yang menghasilkan insulin pada *langerhans pancreas* sehingga terjadi kekurangan insulin pada tubuh. Diabetes tipe 1 ini dapat diderita oleh anak-anak

maupun dewasa. Hingga kini diabetes tipe 1 masih tergolong dalam kategori penyakit yang tidak dapat dicegah, maupun dengan cara diet atau olahraga. Pada fase awal penderita diabetes tipe 1 memiliki kesehatan dan berat badan yang cukup baik serta respon tubuh terhadap insulin masih normal. Penyebab kehilangan sel beta adalah kesalahan reaksi *autoimunitas* (kekebalan tubuh) yang menghancurkan sel beta pankreas. Kesalahan reaksi *autoimunitas* dipicu oleh adanya infeksi pada tubuh (Sutanto, 2010).

2.2.2 Diabetes Tipe 2

Diabetes mellitus tipe 2 adalah hasil dari penolakan tubuh terhadap zat insulin yaitu dimana sel gagal untuk menggunakan insulin dengan benar dan biasanya dikombinasikan dengan kekurangan insulin. *Diabetes mellitus* tipe 2 disebut sebagai *non-insulin-dependent diabetes mellitus* (NIDDM) atau diabetes yang tidak bergantung pada insulin. Diabetes ini terjadi akibat kombinasi kekurangan insulin dan resistensi terhadap insulin, atau berkurangnya kemampuan terhadap penggunaan insulin yang melibatkan reseptor insulin pada membran sel (Sutanto, 2010).

2.2.3 Gejala Penyakit *Diabetes Mellitus*

Gejala *diabetes mellitus* adalah sebuah kondisi kesehatan pada seseorang yang terkena *diabetes mellitus*. Apabila kondisi ini tidak mendapat penanganan yang khusus maka dapat memperburuk keadaan serta mengakibatkan terjadinya penyakit komplikasi atau kematian. Terdapat beberapa gejala pada penyakit *diabetes mellitus* yaitu (Devi & Shyla, 2016):

1. Nafsu Makan Meningkat

Ketika penyakit *diabetes mellitus* menyerang tubuh, maka nafsu makan juga akan meningkat. Gejala ini biasanya diawali ketika seseorang suka makan nasi terlalu banyak/berlebihan.

2. Sering Buang Air Kecil

Gejala ini terjadi karena ginjal ingin membersihkan kelebihan glukosa dalam sirkulasi darah. Seseorang menjadi lebih sering buang air kecil dan dalam jumlah yang besar.

3. Peningkatan Kehausan

Tingginya kadar gula dalam darah akan meningkatkan seringnya buang air kecil sebagai reaksi untuk mengurangi kadar gula. Saat kadar gula keluar bersama urine, tubuh juga akan kehilangan banyak air. Sehingga mengakibatkan dehidrasi atau mengalami peningkatan kehausan.

4. Turunnya Berat Badan

Gula dalam darah (glukosa) merupakan sumber energi bagi tubuh. Glukosa yang terbuang bersama urine juga mengandung banyak nutrisi dan kalori yang diperlukan tubuh manusia. Oleh karena itu penderita diabetes juga kehilangan berat badannya secara drastis.

5. Usia (15-40) tahun

Usia yang semakin bertambah biasanya mengalami kepekaan terhadap insulin. Penderita penyakit *diabetes mellitus* bisa dialami pada usia di bawah 30 tahun maupun di atas umur 40 tahun. Oleh karena itu dapat dicegah dengan menjaga pola makan dan kesehatan tubuh sejak dini.

6. Faktor Keturunan

Faktor genetik sangat berperan sehingga sulit dihilangkan. Gaya hidup dan pola suatu keluarga juga dapat mendukung risiko *diabetes mellitus*. Dengan memperbaiki pola makan dan pola hidup akan terhindar dari penyakit *diabetes mellitus*.

7. Mulut Kering

Penderita penyakit *diabetes mellitus* biasanya mengalami penurunan nafsu minum yang menyebabkan seseorang mengalami dehidrasi, sehingga mulut terlihat kering yang disebabkan oleh dehidrasi yang berlebihan.

8. Kurangnya Aktivitas Fisik

Penderita *diabetes mellitus* biasanya terjadi karena kurangnya gerak tubuh yang menyebabkan kegemukan dan melemahkan kerja organ-organ tubuh seperti jantung, ginjal, dan pankreas. Oleh sebab itu, untuk mencegah penyakit *diabetes mellitus* dengan melakukan olahraga yang cukup dan teratur.

9. Sering Mengantuk/Mudah Kelelahan

Penderita penyakit *diabetes mellitus* biasanya merasakan kelelahan yang berlebihan, hal tersebut terjadi karena tubuh tidak mampu memproses glukosa untuk energi.

10. Mual/Muntah-muntah

Penderita *diabetes mellitus* biasanya mengalami gangguan serabut saraf. Bila hal ini terjadi pada serabut saraf di lambung, maka terjadi gangguan pada gerak lambung dalam upaya mengkosongkan isinya ke dalam usus halus. Akibatnya, timbul sensasi perut terasa penuh, disertai mual hingga muntah. Mual dan muntah pada penderita *diabetes mellitus* juga dapat disebabkan oleh gangguan keseimbangan kadar gula dalam darah., baik kadar gula yang terlalu rendah maupun terlalu tinggi.

11. Timbulnya Luka yang Tak Kunjung Sembuh

Gejala ini merupakan efek dari kerusakan pembuluh darah. Gejala ini mengakibatkan penderita diabetes tidak merasakan sakit jika mengalami luka. Seseorang kadang tidak sadar mengalami luka pada tubuhnya. Gabungan kadar gula darah yang tinggi dan tidak adanya rasa nyeri menyebabkan luka yang semula kecil dapat membesar menjadi borok dan membusuk. Jika telah mengalami hal tersebut maka solusi untuk menyembuhkannya adalah amputasi.

12. Gatal-Gatal

Penderita diabetes biasanya mengalami kelainan kulit berupa gatal, biasanya terjadi pada daerah organ intim atau daerah lipatan kulit seperti ketiak. Sering juga pasien mengeluh timbulnya bisul dan luka yang lama sembuh. Luka ini dapat timbul akibat luka lecet atau tertusuk benda tajam.

13. Mengonsumsi Makanan Berkolesterol Tinggi

Pola makan yang sehat akan mempengaruhi kesehatan tubuh manusia. Apabila mengonsumsi makanan berkolesterol cukup tinggi, seseorang akan lebih mudah terkena penyakit *diabetes mellitus*.

14. Obesitas

Faktor kelebihan berat badan pada tubuh seseorang akan mempengaruhi kadar hormon insulin. Sel-sel tubuh dan jaringan lemak akan bersaing untuk menyerap insulin. Pankreas akan memproduksi banyak insulin sehingga mengakibatkan kerusakan pankreas. Untuk menghindari hal tersebut maka dianjurkan tidak mengonsumsi makanan tinggi kalori.

15. Kadar Glukosa Darah Meningkat

Penderita penyakit *diabetes mellitus* biasanya memiliki tekanan darah di atas 140/90 mmHg. Penyebabnya adalah mengonsumsi makanan yang memiliki kadar garam yang tinggi, selain itu tubuh penderita memproduksi garam yang berlebih sehingga penderita mudah terkena risiko tekanan darah tinggi.

2.3 K-Nearest Neighbor

Algoritme *K-Nearest Neighbor* (K-NN) bertujuan untuk mengklasifikasikan sebuah objek baru berdasarkan atribut dan data *training sampel*. Dimana nantinya hasil dari *sampel* uji yang baru diklasifikasikan sesuai dengan mayoritas dari kategori pada K-NN. Algoritme K-NN menggunakan klasifikasi untuk ketetangaan yang digunakan sebagai nilai prediksi dari sampel uji yang baru (Krisandi, Helmi, & Prihandoso, 2013). Langkah pada algoritme KNN dapat diawali dengan menentukan nilai parameter K, kemudian menghitung nilai dari kedekatan ketetangaan antara data uji dengan data latih dengan menggunakan persamaan *Cosine Similarity* (*CosSim*) (Fadila, Indriati, & Ratnawati, 2016).

Perhitungan *CosSim* dapat diformulasikan sebagai Persamaan 2.1.

$$\text{CosSim}(q, d_j) = \frac{\vec{d}_j \cdot \vec{q}}{|\vec{d}_j| \cdot |q|} = \frac{\sum_{i=1}^m (w_{ij} \cdot w_{iq})}{\sqrt{\sum_{i=1}^m w_{ij}^2 \cdot \sum_{i=1}^m w_{iq}^2}} \quad (2.1)$$

Keterangan:

q = nilai data uji

d_j = nilai data latih

$|\vec{d}_j| \cdot |q|$ = hasil perkalian vektor antara data latih dan data uji

w_{ij} = nilai i pada data latih ke- j

w_{iq} = nilai i pada data uji ke- q

Setelah mendapatkan hasil kedekatan ketetanggaan, langkah selanjutnya adalah melakukan pengurutan hasil dari *CosSim* dari terbesar ke terkecil. Ketika telah mendapatkan hasil K tetangga terdekat dari hasil pengurutan yang telah dilakukan maka akan dihitung nilai pembobotannya.

2.4 Neighbor Weighted K-Nearest Neighbor (NWKNN)

Metode NWKNN adalah algoritme yang telah dikembangkan dari KNN. Metode NWKNN menggunakan pembobotan pada kelas data. Bobot yang akan diberikan lebih sedikit pada tetangga yang berasal dari kelas yang mayoritas, dan untuk kelas yang minoritas akan diberikan bobot yang lebih besar (Indriati & Ridok, 2016).

Perhitungan bobot dapat digunakan pada Persamaan 2.2

$$Weight = \frac{1}{\left(\frac{Num(c_i^d)}{\min\{Num(c_j^d) | j=1, \dots, K^*\}}\right)^{1/exp}} \quad (2.2)$$

Keterangan:

$Num(C_i^d)$ = banyaknya data latih d pada kelas i

$Num(C_j^d)$ = banyaknya data latih d pada kelas j , dimana j terdapat dalam himpunan k tetangga terdekat

Exp = eksponen (nilai exp lebih dari 1)

Pada setiap data yang telah mendapatkan nilai bobot akan digunakan untuk menghitung nilai skor data uji terhadap setiap kelas. Dimana hasil perhitungan nilai bobot akan dikalikan dengan persamaan hasil skor. Perhitungan tersebut akan diterapkan pada rumus hasil skor dengan Persamaan 2.3.

$$score(d, c_i) = \sum_{d_j \in KNN(d)} weight_i Sim(d, d_j) \delta(d_j, c_i) \quad (2.3)$$

Keterangan:

$Weight_i$ = bobot pada jenis atau kelas i

$d_j \in KNN(d)$ = data latih d_j pada kumpulan tetangga terdekat dari data uji d

$\delta(d_j, C_i)$ = akan bernilai 1 jika nilai jarak $\in C_i$ dan bernilai 0 jika jarak $\notin C_i$.



$Sim(d, d_j)$ = nilai *Cosine Similiarity* antara data uji dan data latihan

C_i = jenis atau kelas i



BAB 3 METODOLOGI

Pada bab ini, metodologi penelitian memuat langkah-langkah membangun sistem klasifikasi penyakit *diabetes mellitus* menggunakan algoritma *Neighbor Weighted K-Nearest Neighbor*. Metodologi penelitian meliputi studi literature, pengumpulan data, analisis kebutuhan sistem, perancangan sistem, implementasi sistem, pengujian sistem, pengujian dan analisis hasil, serta kesimpulan dan saran.

3.1 Kajian Pustaka

Penelitian ini menggunakan kajian pustaka dengan cara melakukan pencarian referensi relevan dan mengeksplorasi informasi yang berkaitan dengan implementasi sistem klasifikasi penyakit *diabetes mellitus* menggunakan algoritma *Neighbor Weighted K-Nearest Neighbor*. Kajian pustaka pada penelitian ini diperoleh dari jurnal, buku, internet dan laporan penelitian yang sudah pernah dilakukan. Adapun dasar teori yang digunakan adalah sebagai berikut:

1. Penyakit *Diabetes Mellitus*
2. *K-Nearest Neighbor*
3. *Neighbor Weighted K-Nearest Neighbor*

3.2 Pengumpulan Data

Penelitian ini menggunakan data dari penelitian sebelumnya yang dilakukan oleh Silvia Ikmalia Fernanda yang berjudul "Identifikasi Penyakit *Diabetes Mellitus* Menggunakan Metode *Modified K-Nearest Neighbor* (MKNN)". Dimana data tersebut memiliki 15 atribut dan memiliki 130 data. Variabel atribut kelas yang digunakan pada penelitian tersebut adalah nafsu makan meningkat, sering buang air kecil, penigkatan kehausan, turunnya berat badan, usia (15-20) tahun, faktor keturunan, mulut kering, mudah kelelahan/ kurangnya aktivitas fisik, sering mengantuk, mual/ muntah-muntah, timbulnya luka yang tak kunjung sembuh, gatal-gatal, mengkonsumsi makanan berkolestrol tinggi, obesitas, dan kadar glukosa meningkat.

3.3 Lingkungan Implementasi

1. Spesifikasi kebutuhan perangkat lunak adalah sebagai berikut:
 - Sistem operasi windows 7
 - PHP MySQL
2. Spesifikasi kebutuhan perangkat keras sebagai berikut:
 - Laptop
 - RAM 3.0 GB

- System type 32-bit *operating system*
3. Kebutuhan data antara lain yaitu :
 - Data spesifikasi *diabetes mellitus*

3.4 Perancangan

Perancangan sistem digunakan untuk membuat langkah-langkah pengerjaan yang diperlukan saat sistem akan dibangun. Tahapan perancangan sistem yang akan dibangun pada penelitian ini meliputi deskripsi sistem, perancangan sistem, perhitungan manual, desain antarmuka, serta perancangan untuk pengujian. Pada penelitian ini akan menerapkan sistem untuk klasifikasi *diabetes mellitus* dengan menggunakan algoritme *Neighbor Weighted K-Nearest Neighbor*.

3.5 Implementasi Sistem

Implementasi pada sistem adalah sebuah tahapan untuk membangun sistem yang mengarah pada perancangan sistem implementasi algoritme *Neighbor Weighted K-Nearest Neighbor* yang digunakan untuk klasifikasi penyakit *diabetes mellitus*. Selain itu berikut adalah langkah-langkah untuk implementasi :

4. Implementasi interface menggunakan PHP MySQL.
5. Implementasi algoritma dengan melakukan perhitungan klasifikasi penyakit *diabetes mellitus* menggunakan metode algoritme *Neighbor Weighted K-Nearest Neighbor* ke dalam bahasa pemrograman PHP MSQQL.
6. Implementasi menghasilkan klasifikasi penyakit *diabetes mellitus*.

3.6 Pengujian Sistem

Dalam hal ini pengujian dilakukan untuk mengetahui apakah sistem sudah berjalan sesuai dengan kebutuhan, serta apakah perancangan sistem dapat berjalan baik. Selain itu pada penelitian ini akan dilakukan analisis hasil setelah dilakukannya pengujian.

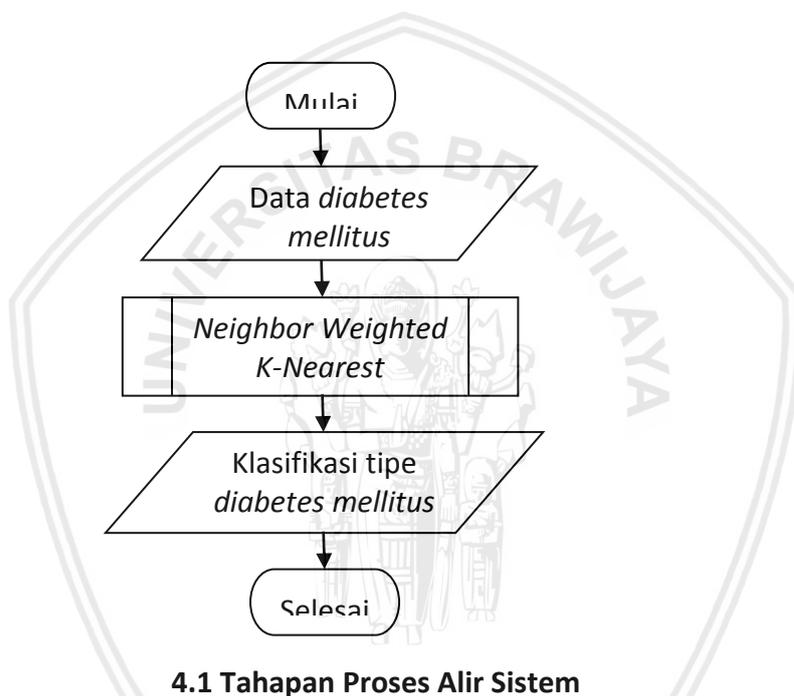
3.7 Kesimpulan dan Saran

kesimpulan dilakukan setelah semua tahapan mulai dari perancangan hingga pengujian telah dilakukan sepenuhnya, maka dapat menarik kesimpulan. Terdapat tahapan terakhir yaitu saran yang digunakan untuk perbaikan yang diharapkan kedepannya dapat memperbaiki kesalahan yang ada serta dapat mengembangkan sistem.

BAB 4 PERANCANGAN

4.1 Perancangan Tahapan Proses

Data yang digunakan pada sistem klasifikasi penyakit diabetes mellitus ini adalah data rekam medis yang dimana parameter masukannya sudah tersimpan di dalam sistem. Parameter ini nantinya akan diseleksi oleh user atau pasien sesuai dengan gejala yang diderita. Kemudian sistem melakukan perhitungan menggunakan algoritme *Neighbor Weighted K-Nearest Neighbor*, sehingga dapat diketahui tipe penyakit diabetes mellitus yang diderita oleh pasien tersebut. Keluaran dari sistem ini merupakan klasifikasi tipe penyakit *diabetes mellitus*. Gambar 4.1 akan menunjukkan tahapan proses alir sistem yang akan dilakukan.

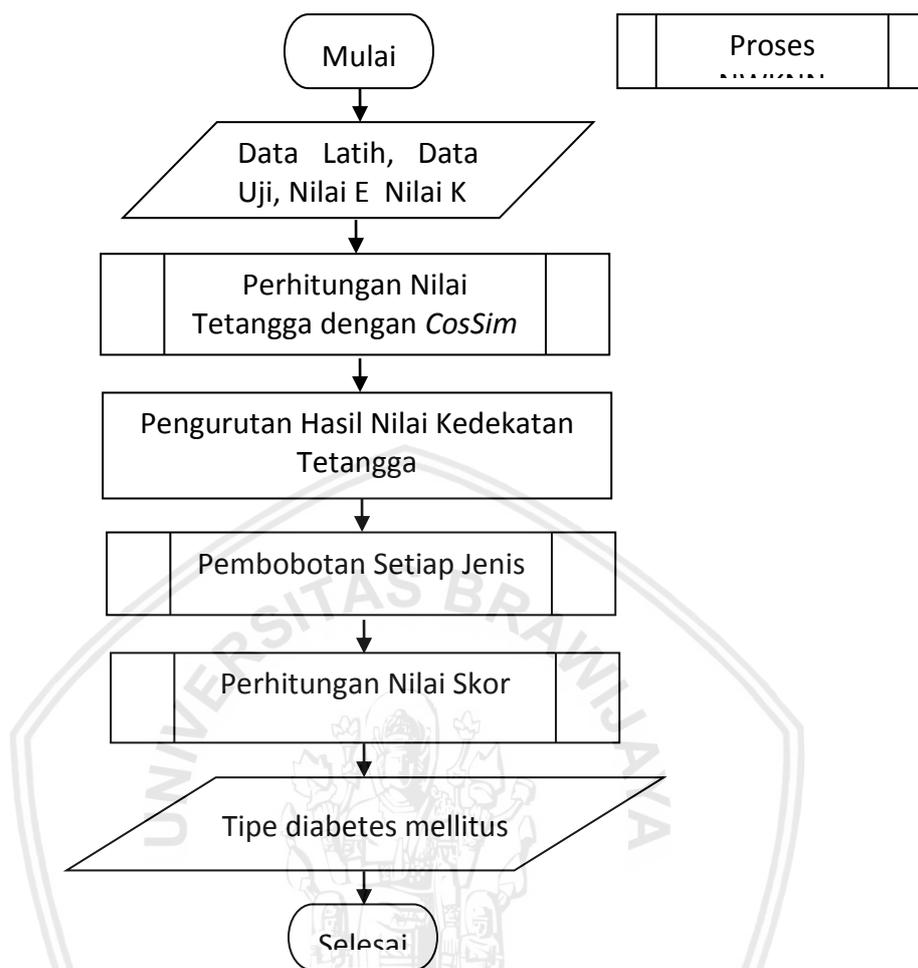


Gambar

4.1 Tahapan Proses Alir Sistem

4.1.1 Proses Algoritme *Neighbor Weighted K-nearest Neighbor*

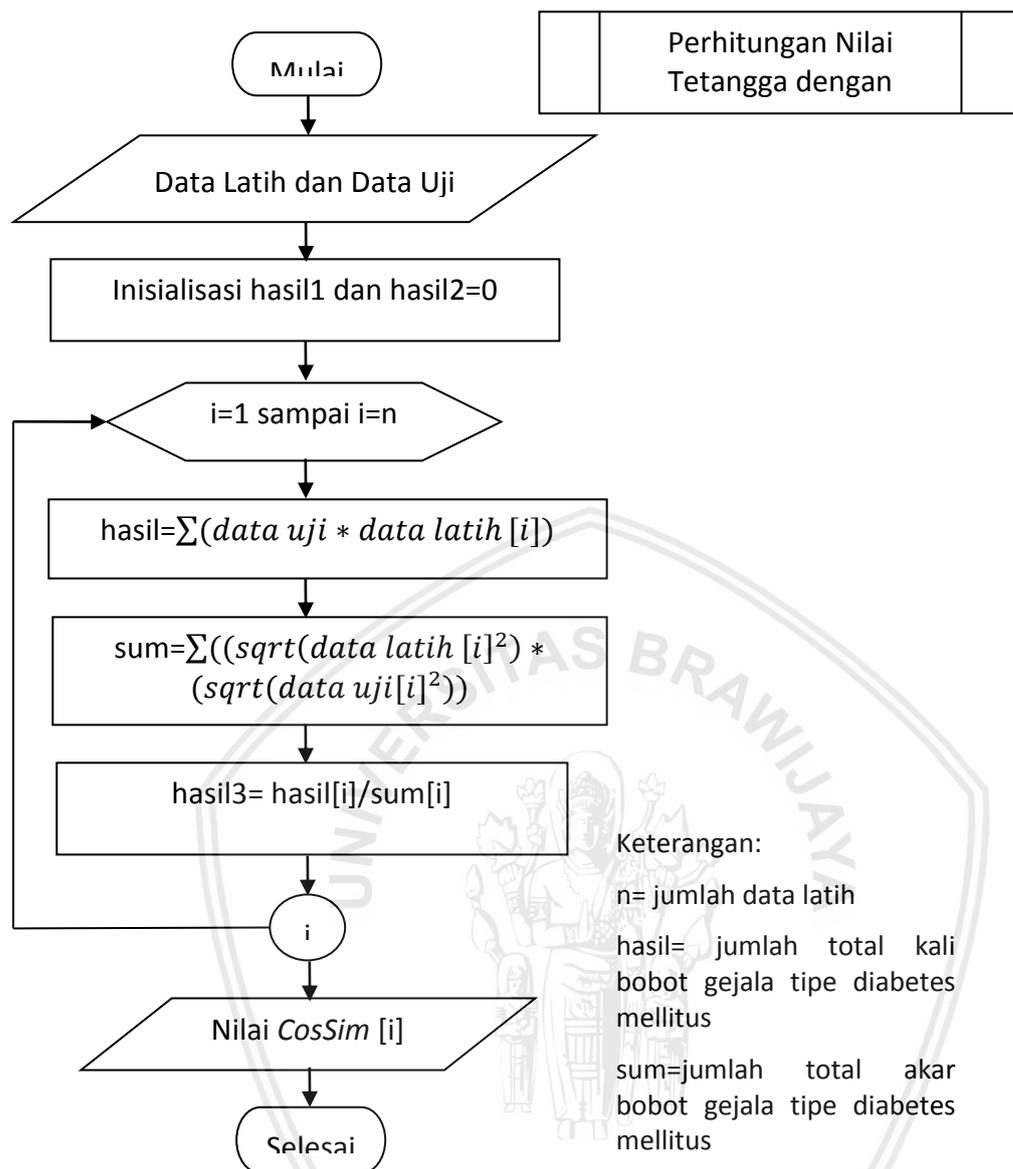
Terdapat beberapa proses pada algoritme *Neighbor Weighted K-Nearest Neighbor* (NWKNN), proses tersebut adalah menghitung kedekatan ketetangaan menggunakan *Cosine Similarity (CosSim)*, mengurutkan nilai *CosSim* dari terbesar hingga terkecil, pembobotan pada setiap jenis gejala dan menghitung skor jenis gejala yang sesuai dengan nilai K. Proses perhitungan NWKNN dapat dilihat pada diagram Gambar 4.2.



Gambar 4.2 Diagram Proses NWKNN

4.1.2 Proses Pehitungan Ketetangaan dengan Cosine Similarity

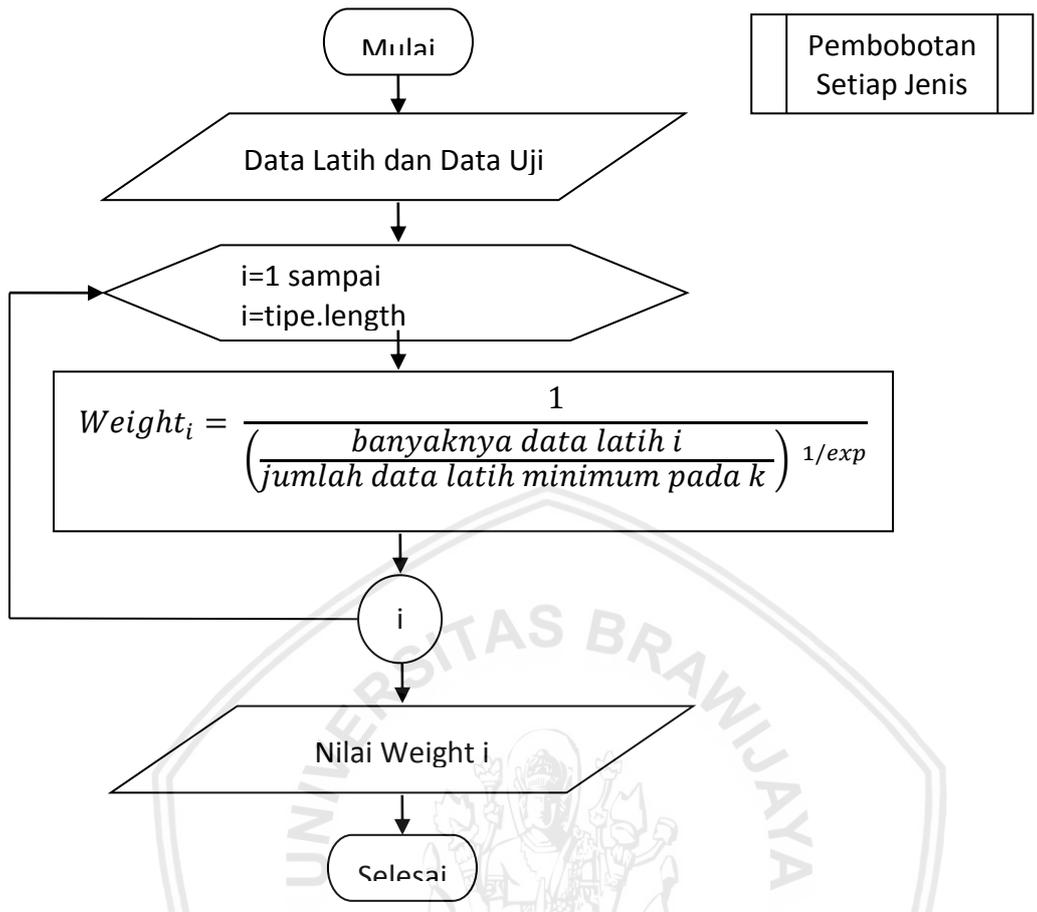
Perhitungan *Cosine Similarity* (*CosSim*) bertujuan untuk menghitung kedekatan kesamaan ketetangaan antara data latih dan data uji pada setiap masukan pasien terhadap parameter yang telah ada di sistem. Tahapan pertama dari proses ini yaitu memasukkan nilai setiap parameter yang telah diinputkan oleh pasien, kemudian masukkan tersebut akan dihitung kesamaannya menggunakan *Cosine Similarity*. Setelah proses perhitungan *CosSim* selesai maka akan didapatkan nilai ketetangaan antara data latih dan data uji. Proses perhitungan *CosSim* dapat dilihat pada persamaan 2.1 yang akan ditampilkan pada diagram Gambar 4.3.



Gambar 4.3 Diagram Perhitungan Ketetangaan

4.1.3 Proses Pembobotan Setiap Tipe Diabetes Mellitus

Tujuan pada proses ini adalah untuk mendapatkan nilai bobot setiap tipe penyakit diabetes mellitus. Proses perhitungan bobot terdapat pada persamaan 2.2 yang akan ditampilkan pada diagram alir Gambar 4.4.

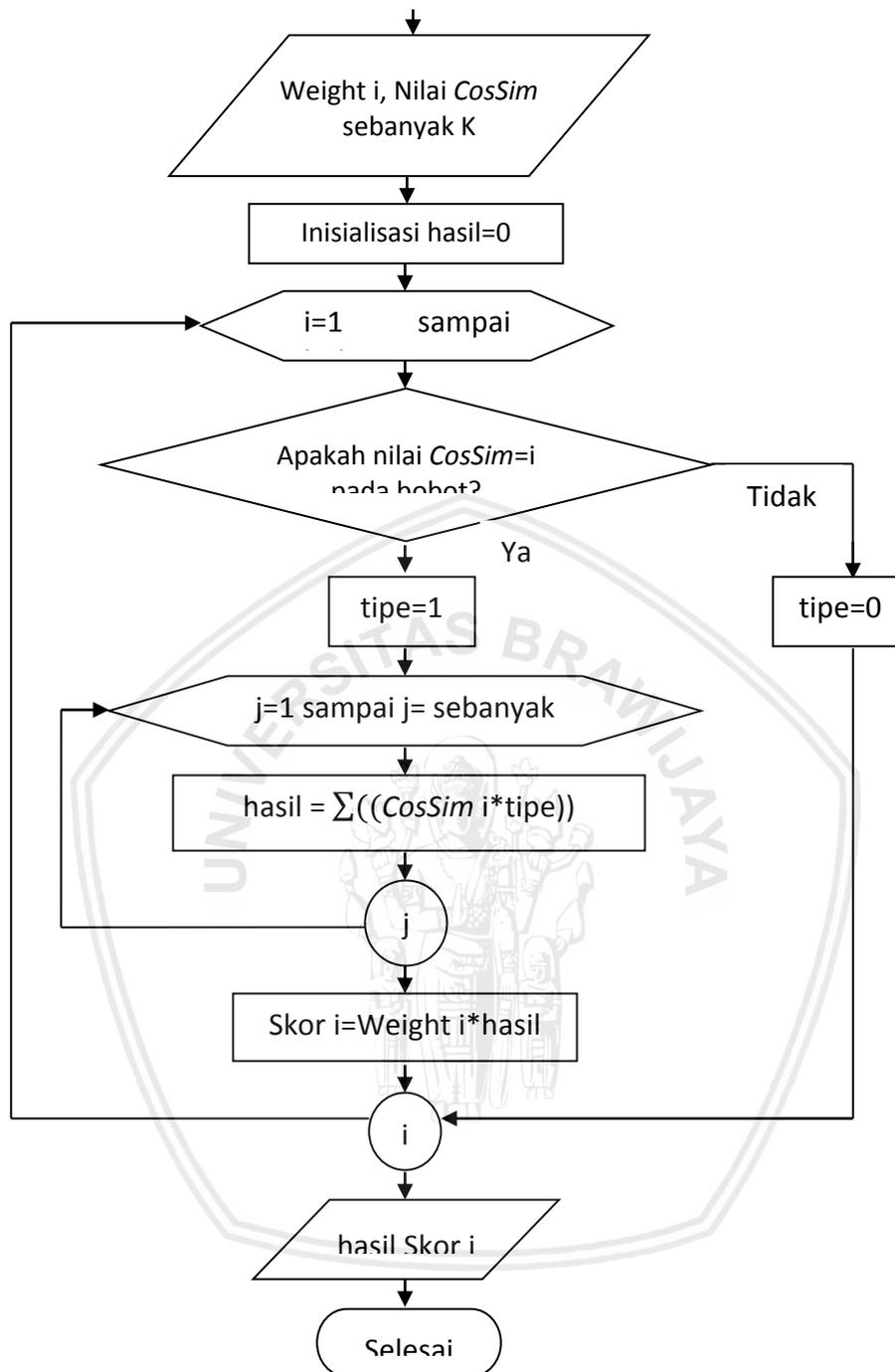


Gambar 4.4 Proses Perhitungan Bobot Setiap Jenis

4.1.4 Proses Perhitungan Nilai Skor

Pada proses perhitungan nilai skor ini digunakan untuk mendapatkan nilai skor pada setiap tipe penyakit *diabetes mellitus*. Proses perhitungan nilai skor terdapat pada persamaan 2.3 yang ditampilkan pada Gambar 4.5.





Gambar 4.5 Diagram Alir Perhitungan Skor

4.2 Perhitungan Manual

Perhitungan manual pada proses klasifikasi *diabetes mellitus* menggunakan algoritme NWKNN, berfungsi untuk menggambarkan perancangan tahapan perhitungan NWKNN pada sistem yang akan dibuat. Berikut tahapan proses manualisasi algoritma NWKNN:

Langkah 1. Menentukan dataset

Perhitungan NWKNN ini menggunakan data latih sebanyak 20 data dan data uji sebanyak 1, yang terdiri dari 15 gejala *diabetes mellitus*, tipe 1 dan tipe 2. Dataset klasifikasi penyakit *diabetes mellitus* dapat dilihat pada Tabel 4.1

Tabel 4.1 Dataset klasifikasi penyakit *diabetes mellitus*

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

Langkah 2. Menentukan nilai K tetangga

Pada perhitungan manual ini menggunakan nilai ketetanggaan sebanyak K=5

Langkah 3. Menghitung nilai kedekatan ketetanggaan

Pada langkah ini digunakan untuk menghitung nilai kedekatan ketetanggaan dengan menghitung nilai *Similarity* pada data uji terhadap data latih menggunakan persamaan *CosSim*. Berikut ini merupakan contoh perhitungan nilai kedekatan ketetanggaan yang menggunakan data uji ke 21 dengan data latih ke 1.

$$CosSim (q, d_j) = \frac{\vec{d}_j \cdot \vec{q}}{|\vec{d}_j| \cdot |q|} = \frac{\sum_{i=1}^m (w_{ij} \cdot w_{iq})}{\sqrt{\sum_{i=1}^m w_{ij}^2 \cdot \sum_{i=1}^m w_{iq}^2}}$$

Nilai w_{ij} merupakan bobot dari setiap parameter pada data latih sedangkan w_{iq} adalah bobot dari parameter pada data uji yang dihitung.

$$\bar{d}_j \cdot \bar{q} =$$

$$(0*0)+(1*0)+(0*1)+(1*1)+(1*1)+(1*1)+(0*1)+(0*0)+(1*1)+(0*0)+(0*1)+(1*0)+(1*0)+(0*1)+(1*1)= 5$$

$$|\bar{d}_j| \cdot |\bar{q}| =$$

$$\sqrt{(0^2) + (1^2) + (0^2) + (1^2) + (1^2) + (1^2) + (0^2) + (0^2) + (1^2) + (0^2) + (0^2) + (0^2) + (1^2) + (1^2) + (0^2) + (1^2)}$$

×

$$\sqrt{(0^2) + (0^2) + (1^2) + (1^2) + (1^2) + (1^2) + (1^2) + (1^2) + (0^2) + (1^2) + (0^2) + (1^2) + (0^2) + (1^2) + (0^2) + (1^2)}$$

$$= 8,485281374$$

$$(q, d_j) = \frac{\bar{d}_j \cdot \bar{q}}{|\bar{d}_j| \cdot |\bar{q}|} = \frac{\sum_{i=1}^m (w_{ij} \cdot w_{iq})}{\sqrt{\sum_{i=1}^m w_{ij}^2 \cdot \sum_{i=1}^m w_{iq}^2}} = 0,589255651$$

Contoh di atas merupakan perhitungan dari salah satu data latih terhadap data uji. Hasil perhitungan lainnya yang terkait dengan nilai kedekatan ketetangaan dappat dilihat pada Tabel 4.2

Tabel 4.2 Nilai kedekatan ketetangaan

No.	Hasil	Tipe
1	0,589256	Tipe 1
2	0,589256	Tipe 1
3	0,703526	Tipe 1
4	0,555556	Tipe 1
5	0,471405	Tipe 1
6	0,471405	Tipe 2
7	0,503953	Tipe 2
8	0,555556	Tipe 2
9	0,666667	Tipe 2
10	0,353553	Tipe 2
11	0,353553	Tipe 1
12	0,707107	Tipe 1
13	0,673575	Tipe 1



14	0,503953	Tipe 1
15	0,632456	Tipe 1
16	0,707107	Tipe 2
17	0,666667	Tipe 2
18	0,527046	Tipe 2
19	0,629941	Tipe 2
20	0,408248	Tipe 2

Langkah 4. Mengurutan nilai perhitungan dari kedekatan ketetangaan

Nilai kedekatan ketetangaan antara data uji dan data latih yang telah didapatkan dari perhitungan *CosSim*, akan diurutkan dari kelas yang memiliki kemiripan paling besar hingga paling kecil. Jika nilai *Similarity* semakin besar, maka semakin dekat nilai ketetanggaannya.

Langkah 5. Pembobotan

Proses perhitungan langkah kelima ini merupakan pembobotan pada setiap jenis *diabetes mellitus* dengan menggunakan nilai K dan nilai eksponen (*exp*) yang telah ditentukan sebelumnya. Kelas yang paling sedikit jumlahnya akan diberi bobot yang besar, sedangkan kelas dengan jumlah yang banyak akan diberikan nilai bobot yang kecil. Persamaan 2.5 digunakan pada proses ini berikut merupakan contoh perhitungan bobot *diabetes mellitus* dengan nilai K=5 dan E=2.

Jenis *diabetes mellitus* yang ada pada data latih yaitu:

- Tipe 1 sebanyak 10
- Tipe 2 sebanyak 10

Kedua kelas tersebut memiliki data yang seimbang yaitu sebesar 10 data, berikut adalah hasilnya:

$$Weight_i = \frac{1}{\left(\frac{\text{banyaknya data latih } i}{\text{jumlah data latih minimum pada } k} \right)^{1/exp}}$$

$$Weight (Tipe 1) = \frac{1}{\left(\frac{\text{JumlahDataTipe1}}{\text{JumlahDataMin}} \right)^{\frac{1}{2}}} = \frac{1}{\left(\frac{10}{10} \right)^{\frac{1}{2}}} = 1$$

$$Weight (Tipe 2) = \frac{1}{\left(\frac{\text{JumlahDataTipe2}}{\text{JumlahDataMin}} \right)^{\frac{1}{2}}} = \frac{1}{\left(\frac{10}{10} \right)^{\frac{1}{2}}} = 1$$

Nilai dari perhitungan bobot tersebut juga dihitung pada data uji dan data latih yang sudah ditentukan. Hasil pembobotan dapat dilihat pada Tabel 4.3.

Tabel 4.3 Nilai bobot setiap jenis

No.	Jenis	Nilai Bobot
1	Tipe 1	1
2	Tipe 2	1

Langkah 6. Perhitungan skor

Setelah melakukan perhitungan nilai bobot pada setiap jenis *diabetes mellitus*, langkah selanjutnya adalah melakukan perhitungan skor terhadap nilai K pada setiap jenis *diabetes mellitus* untuk mengetahui hasil klasifikasi. Hasil skor yang memiliki nilai terbesar menjadi hasil klasifikasi namun sebelum itu data yang sesuai dengan nilai K yang telah diurutkan kesamaannya, diambil berdasarkan nilai K=5. Berikut adalah contoh perhitungan skor berdasarkan persamaan 2.7

- Mengambil data yang telah diurutkan kesamaannya pada K=5. Pada Tabel 4.4 merepresentasikan nilai *CosSim* yang sudah diurutkan dan diambil sesuai dengan nilai K.

Tabel 4.4 Nilai *CosSim* yang telah diurutkan dengan K=5

No	Hasil	Tipe
12	0,707107	Tipe 1
16	0,707107	Tipe 2
3	0,703526	Tipe 1
13	0,673575	Tipe 1
9	0,666667	Tipe 2

- Perhitungan skor pada setiap jenis *diabetes mellitus* yang masuk dalam K tetangga

Setelah nilai *CosSim* diurutkan dan diambil dari terbesar pada K=5, dan klasifikasi yang masuk adalah Tipe 1 dan Tipe 2 pada data uji ke 21. Maka skor yang dihitung adalah skor dari hasil klasifikasi tersebut.

Skor (Tipe 1)

$$1 \times (0,707106781 \times 1) + (0,707106781 \times 0) + (0,703526471 \times 1) + (0,673575314 \times 1) + (0,666666667 \times 0) = \mathbf{2,084208566}$$

Skor (Tipe 2)

$$1 \times (0,707106781 \times 0) + (0,707106781 \times 1) + (0,703526471 \times 0) + (0,673575314 \times 0) + (0,666666667 \times 1) = 1,373773448$$

Berdasarkan perhitungan di atas, maka hasil klasifikasi dengan data uji ke 21 adalah jenis Tipe 1 dikarenakan nilai terbesar dari perhitungan skor dengan K= 5.

4.3 Perancangan Antarmuka

Antarmuka adalah sebuah penghubung sistem dengan pengguna, antarmuka ini dapat memberikan kemudahan bagi pengguna untuk menjalankan sebuah sistem. Perancangan perangkat lunak atau desain antarmuka Klasifikasi penyakit *diabetes mellitus* dengan menggunakan metode *Neighbor Weighted K-Nearest Neighbor* akan menampilkan halaman beranda, halaman data, dan halaman hasil analisis.

4.3.1 Desain antarmuka halaman beranda

Pada tampilan awal, sistem akan menampilkan tab data yang berisi tombol beranda, data, dan hasil analisis NWKNN. Pada halaman beranda akan menampilkan tentang pengertian penyakit *diabetes mellitus* dan gejala penyakit *diabetes mellitus* serta dapat mengubah, menghapus dan menambahkan gejala sesuai dengan kebutuhan. Perancangan halaman ini terdapat pada Gambar 4.6.

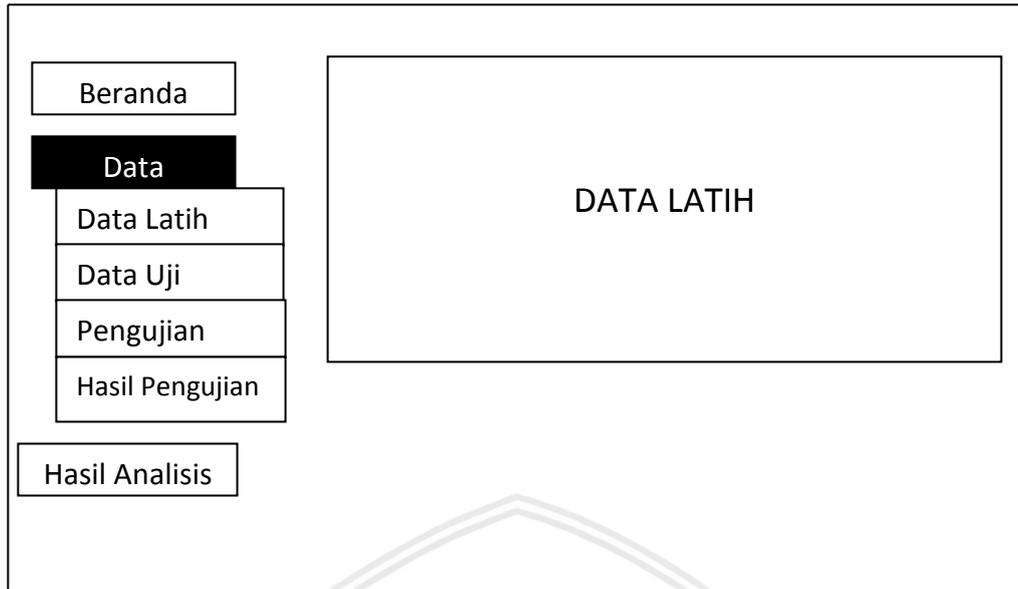


Gambar

4.6 Perancangan Halaman Home

4.3.2 Halaman Data

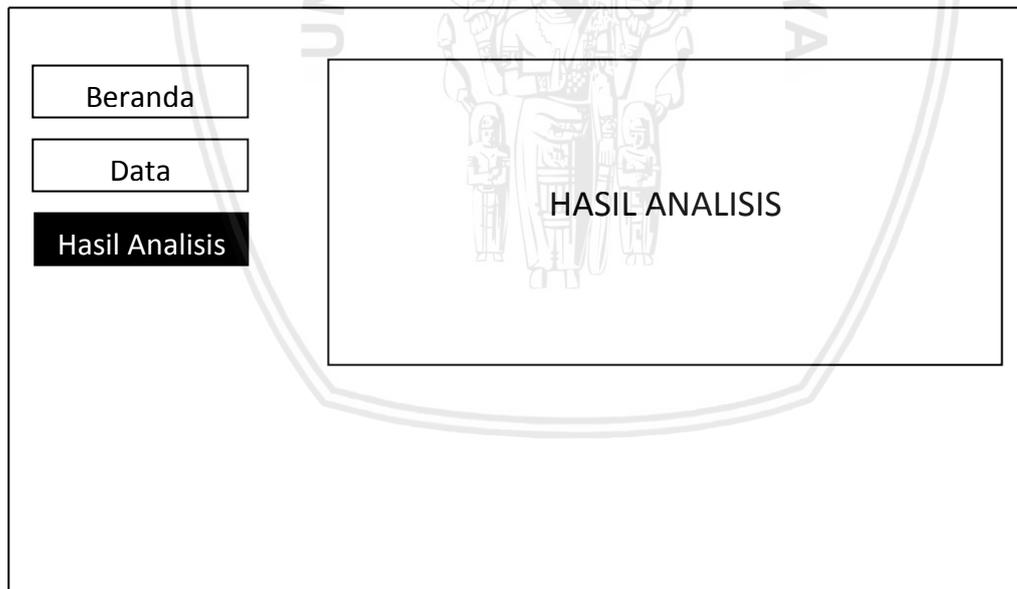
Pada halaman ini terdapat daftar data latih, data uji, pengujian dan hasil pengujian untuk klasifikasi tipe diabetes mellitus. Pada data uji pengguna bisa menginputkan gejala sesuai dengan yang diderita serta dapat menghapus inputan. Sedangkan pada halaman pengujian dapat memproses data yang akan di uji, selain itu untuk mengetahui hasil dari pengujian dapat dilihat pada halaman hasil pengujian. Perancangan halaman data terdapat pada Gambar 4.7.



Gambar 4.7 Perancangan Halaman Data

4.3.3 Halaman Hasil Analisis

Pada halaman ini pengguna dapat melihat hasil sproses perhitungan NWKNN pada halaman hasil analisis. Perancangan halaman identifikasi akan ditampilkan pada Gambar 4.8.



Gambar 4.8 Perancangan Halaman Identifikasi

4.4 Perancangan Pengujian

Pada sub bab ini akan menjelaskan tentang perancangan pengujian data terhadap sistem. Pengujian yang digunakan adalah *K-Fold Cross Validation* dan pengujian pengaruh nilai E, yang bertujuan untuk menguji akurasi sistem berdasarkan data uji terhadap data latih.

4.4.1 Pengujian *K-Fold Cross Validation*

Pengujian dilakukan dengan menggunakan *K-Fold Cross Validation*. Perancangan pengujian *K-Fold Cross Validation* dapat dilihat pada tabel 4.5

Tabel 4.5 Perancangan Pengujian *K-Fold Cross Validation*

Jumlah data uji	Nilai E	Nilai K	Jumlah Data Latih	Akurasi Data Latih (%)
13	2	15		

4.4.2 Pengujian Pengaruh Nilai E

Pengujian dilakukan dengan menggunakan perubahan nilai E pada sistem. Pengujian pengaruh nilai E dapat dilihat pada tabel 4.6, tabel 4.7 dan tabel 4.8.

Tabel 4.6 Perancangan Pengujian Pengaruh Nilai E=2

Jumlah data uji	Nilai K	Nilai E	Jumlah Data Latih	Akurasi Data Latih (%)
13	15	2		

Tabel 4.7 Perancangan Pengujian Pengaruh Nilai E=4

Jumlah data uji	Nilai K	Nilai E	Jumlah Data Latih	Akurasi Data Latih (%)
13	15	4		

Tabel 4.8 Perancangan Pengujian Pengaruh Nilai E=6

Jumlah data uji	Nilai K	Nilai E	Jumlah Data Latih	Akurasi Data Latih (%)
13	15	6		

BAB 5 IMPLEMENTASI

Pada bab ini akan membahas tentang implementasi sistem yang dibuat pada perancangan yang telah dilakukan. Implementasi yang dijelaskan meliputi batasan implementasi implementasi algoritme dan implementasi antarmuka.

5.1 Batasan Implementasi

Pada sistem ini terdapat beberapa batasan yang digunakan dalam penerapan metode *Neighbor Weighted K-Nearest Neighbor* dalam klasifikasi *diabetes mellitus* adalah sebagai berikut:

1. Sistem yang dibangun menggunakan bahasa pemrograman PHP.
2. Metode yang digunakan dalam pembuatan sistem sistem klasifikasi *diabetes mellitus* adalah NWKNN.
3. Data yng digunakan berasal dari skripsi.
4. *Input* dalam sistem merupakan data rekam medis pasien *diabetes mellitus*.
5. *Output* yang dihasilkan sistem adalah klasifikasi tipe *diabetes mellitus*, berdasarkan masukan data rekam medis.

5.2 Implementasi Algoritme

Implementasi algoritme ini mengacu pada perancangan proses yang telah dijelaskan pada bab perancangan sebelumnya, yaitu langkah-langkah proses perhitungan dalam algoritme NWKNN.

5.2.1 Implementasi Perhitungan *Cosine Similarity*

Persamaan *cosine similarity* digunakan untuk menentukan kedekatan ketetangaan antara data uji dan data latih yang terdapat pada persamaan 2.1. Proses perhitungan nilai *Cosine Similarity* memiliki tiga proses yaitu dimulai dengan menghitung perkalian bobot antara data uji dan data latih, serta langkah selanjutnya adalah akar bobot dari data uji dan data latih, kemudian hasil dari perkalian bobot dibagi dengan hasil dari akar bobot. Implementasi perhitungan *Cosine Similarity* ditunjukkan pada kode program 5.1.

```

1 public function display(){
2   $data['latih'] = $this->Data_pasien_model->get_data_latih();
3   $data['uji'] = $this->Data_uji_model->get_data_uji();
4   $jumlah_gejala = count($this->Gejala_model->get_all());
5   $i = 0;
6   $jumlah_data_latih = count($data['latih']);
7   for($i=1;$i<=$jumlah_data_latih;$i++){
8     $hasil = 0;
9     for($j=1;$j<=$jumlah_gejala;$j++){
10      $hasil += (integer) $data['uji'][0]['g'.$j] *
11 (integer) $data['latih'][$i-1]['g'.$j];
12    }
13    $data['hasil'][$i] = $hasil;
14    $tipe[$i] = $data['latih'][$i-1]['tipe'];
15  }
16
17 function display(){
18 $data['latih'] = $this->Data_pasien_model->get_data_latih();
19 $data['uji'] = $this->Data_uji_model->get_data_uji();
20 $jumlah_data_latih = count($data['latih']);
21 $jumlah_gejala = count($this->Gejala_model->get_all());
22 for($i=1;$i<=$jumlah_data_latih;$i++){
23   $sum = 0;
24   for($j=1;$j<=$jumlah_gejala;$j++){
25     $sum += (pow(($data['latih'][$i-1]['g'.$j]),2));
26   }
27   $hasil_data_latih[$i] = sqrt($sum);
28 }
29 for($i=1;$i<=$jumlah_data_latih;$i++){
30   $sum = 0;
31   for($j=1;$j<=$jumlah_gejala;$j++){
32     $sum += (pow(($data['uji'][0]['g'.$j]),2));
33   }
34   $hasil_data_uji[$i] = sqrt($sum);
35 }
36 for($i=1;$i<=$jumlah_data_latih;$i++){
37   $hasil_perkalian[$i] = $hasil_data_latih[$i] *
38 $hasil_data_uji[$j];
39 }
40
41 function display(){
42 $data['hasil1'] = $this->Hasil1_model->get_hasil();
43 $data['hasil2'] = $this->Hasil2_model->get_hasil();
44 $jumlah = count($data['hasil1']);
45 for($i=1;$i<=$jumlah;$i++){
46 $hasil3[$i] = $data['hasil1'][$i-1]['hasil'] /
47 $data['hasil2'][$i-1]['hasil_perkalian'];
48   $tipe[$i] = $data['hasil1'][$i-1]['tipe'];
49 }

```

Kode Program 5.1 Implementasi Algoritme *Cosine Similarity*

1. Baris 2-3 digunakan untuk mengambil data latih dan data uji dari model.
2. Baris 4 digunakan untuk mengambil data jumlah gejala dari model.
3. Baris 5 digunakan untuk inisialisasi i .
4. Baris 6 digunakan untuk mengambil jumlah data latih.
5. Baris 7-15 digunakan untuk menghitung perkalian bobot antara data uji dan data latih.

6. Baris 18-19 digunakan untuk mengambil data latih dan data uji dari model.
7. Baris 20 digunakan untuk mengambil jumlah data latih.
8. Baris 21 digunakan untuk mengambil data jumlah gejala dari model.
9. Baris 22-35 digunakan untuk menghitung akar bobot pada data latih dan data uji.
10. Baris 36-39 digunakan untuk menghitung perkalian hasil dari akar bobot data latih dan data uji.
11. Baris 42-43 digunakan untuk mengambil hasil dari perkalian bobot dan akar bobot dari model.
12. Baris 44 digunakan untuk mengambil data jumlah hasil perkalian bobot.
13. Baris 45-49 digunakan untuk menghitung pembagian dari hasil perkalian bobot dan akar bobot.

5.2.2 Implementasi Algoritme Pengurutan *Cosine Similarity*

Hasil yang didapatkan dari *Cosine Similarity*, kemudian diurutkan mulai dari nilai terbesar hingga terkecil. Implementasi algoritme pengurutan *Cosine Similarity* ditunjukkan pada kode program 5.2.

```

1 function get_hasil_urut() {
2     $this->db->select('hasil');
3     $this->db->select('tipe');
4     $this->db->order_by('hasil', 'DESC');
5     $this->db->from($this->table);
6     return $this->db->get()->result_array();
7 }

```

Kode Program 5.2 Implementasi Algoritme Pengurutan *Cosine Similarity*

1. Baris 2-3 digunakan untuk mengambil data yang ada pada kolom hasil dan tipe.
2. Baris 4 digunakan untuk mengurutkan data dari terbesar hingga terkecil berdasarkan tabel hasil.
3. Baris 5-6 digunakan untuk mengambil data dari hasil tabel hasil yang disimpan dalam bentuk array.

5.2.3 Implementasi Algoritme Pembobotan Setiap Tipe

Proses pembobotan pada setiap tipe menggunakan persamaan 2.2. Implementasi algoritme setiap tipe ditunjukkan pada kode program 5.3.

```

1 function display(){
2   $jumlah_tipe[0] = count($this->Data_pasien_model-
3 >get_jumlah_tipe1());
4   $jumlah_tipe[1] = count($this->Data_pasien_model-
5 >get_jumlah_tipe2());
6   $jumlah = $this->Data_pasien_model->get_jumlah_tipe();
7   $jumlah_terkecil = (integer) $jumlah[0]['jumlah_tipe'];
8   $tipe[0] = "Tipe 1";
9   $tipe[1] = "Tipe 2";
10  $this->Hasil_bobot_model->truncate();
11  for($i=0; $i<2; $i++){
12  $hasil[$i]= 1*pow(($jumlah_tipe[$i]/$jumlah_terkecil),0.5);
13    $data_simpan = array(
14      'hasil' => $hasil[$i],
15      'tipe' => $tipe[$i],
16    );

```

Kode Program 5.3 Implementasi Algoritme Pembobotan Setiap Tipe

1. Baris 2-3 digunakan untuk mengambil data jumlah tipe 1 dari model.
2. Baris 4-5 digunakan untuk mengambil data jumlah tipe 2 dari model.
3. Baris 6-7 digunakan untuk mengambil jumlah tipe terkecil.
4. Baris 8-9 digunakan untuk inialisasi nilai tipe.
5. Baris 10 digunakan untuk mengkosongkan tabel.
6. Baris 11-12 digunakan untuk menghitung pembobotan.
7. Baris 13-16 digunakan untuk menyimpan hasil bobot pada *database*.

5.2.4 Implementasi Algoritme Perhitungan Skor

Sebelum melakukan perhitungan nilai skor, terdapat langkah-langkah yaitu dengan mengambil hasil yang telah diurutkan dari *Cosine Similarity* sebanyak nilai K yang telah ditentukan. Proses perhitungan skor menggunakan persamaan 2.3. Hasil nilai skor didapatkan dengan nilai tipe *diabetes mellitus* yang paling besar untuk digunakan sebagai klasifikasi tipe *diabetes mellitus*. Implementasi dari algoritme perhitungan skor ditunjukkan pada kode program 5.4.

```

1 public function hasil5(){
2   $data['hasil_bobot'] = $this->Hasil_bobot_model-
3   >get_hasil();
4   $data['pengurutan'] = $this->Pengurutan_model->get_hasil();
5   $k = 5;
6   $this->Hasil5_model->truncate();
7   for($i=0;$i<5;$i++){
8     $data_simpan = array(
9       'hasil' =>$data['pengurutan'][$i]['hasil'],
10      'tipe' => $data['pengurutan'][$i]['tipe'],
11    );
12    $this->Hasil5_model->insert($data_simpan);
13  }
14  $hasil[0] = $this->Hasil5_model->get_hasil_tipe1();
15  $hasil[1] = $this->Hasil5_model->get_hasil_tipe2();
16  $tipe[0] = "Tipe 1";
17  $tipe[1] = "Tipe 2";
18
19  function get_hasil_tipe1(){
20    $this->db->select('sum(hasil) as hasil1');
21    $this->db->select('tipe');
22    $this->db->where('tipe', 'Tipe 1');
23    $this->db->from($this->table);
24    return $this->db->get()->result_array();
25  }
26  function get_hasil_tipe2(){
27    $this->db->select('sum(hasil) as hasil2');
28    $this->db->select('tipe');
29    $this->db->where('tipe', 'Tipe 2');
30    $this->db->from($this->table);
31    return $this->db->get()->result_array();
32  }

```

Kode Program 5.4 Implementasi Perhitungan Skor

1. Baris 2-3 digunakan untuk mengambil data hasil bobot dari model.
2. Baris 4 digunakan untuk mengambil hasil pengurutan dari model.
3. Baris 5 digunakan untuk inisialisasi K.
4. Baris 6 digunakan untuk mengkosongkan tabel.
5. Baris 7-11 digunakan untuk mengambil nilai K dari pengurutan *Cosine Similarity*.
6. Baris 12 digunakan untuk menyimpan data.
7. Baris 14-15 digunakan untuk mengambil hasil dari tipe 1 dan tipe 2 dari model.
8. Baris 16-17 digunakan untuk menginisialisasi nilai tipe.
9. Baris 20 digunakan untuk penjumlahan nilai hasil skor tipe 1.
10. Baris 21 digunakan untuk mengambil data kolom tipe.
11. Baris 22 digunakan untuk kondisi dimana data tipe yang diambil adalah tipe 1.
12. Baris 23-25 digunakan untuk mengambil data dari tabel tipe dan menyimpannya ke dalam array.
13. Baris 27 digunakan untuk penjumlahan nilai hasil skor tipe 2.
14. Baris 28 digunakan untuk mengambil data kolom tipe.

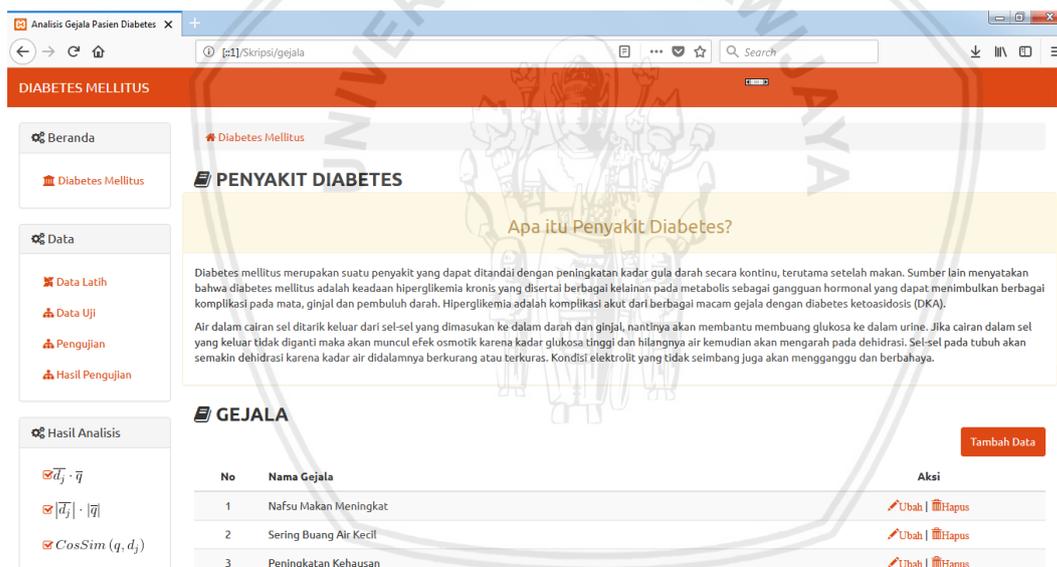
15. Baris 29 digunakan untuk kondisi dimana data tipe yang diambil adalah tipe 2.
16. Baris 30-32 digunakan untuk mengambil data dari tabel tipe dan menyimpannya ke dalam array.

5.3 Implementasi Antarmuka

Antarmuka pada sistem penerapan metode *Neighbor Weighted K-nearest Neighbor* dalam klasifikasi penyakit *diabetes mellitus* dibuat untuk melakukan perhitungan pada setiap proses yang ada. Antarmuka yang ada pada sistem klasifikasi penyakit *diabetes mellitus* adalah halaman beranda dan halaman perhitungan menggunakan metode *Neighbor Weighted K-Nearest Neighbor*. Pada halaman beranda berisi tentang pengertian penyakit diabetes mellitus dan gejalanya, sedangkan pada halaman perhitungan metode *Neighbor Weighted K-Nearest Neighbor* berisi tentang hasil dari langkah-langkah metode *Neighbor Weighted K-nearest Neighbor* beserta hasil klasifikasi tipe penyakit diabetes.

5.3.1 Implementasi Antarmuka Beranda

Pada halaman beranda terdapat pengertian penyakit diabetes mellitus beserta gejalanya yang dapat ditambahkan, diubah dan dihapus sesuai dengan kebutuhan. Implementasi halaman beranda ditunjukkan pada Gambar 5.1.



Gambar 5.1 Implementasi Antarmuka Beranda

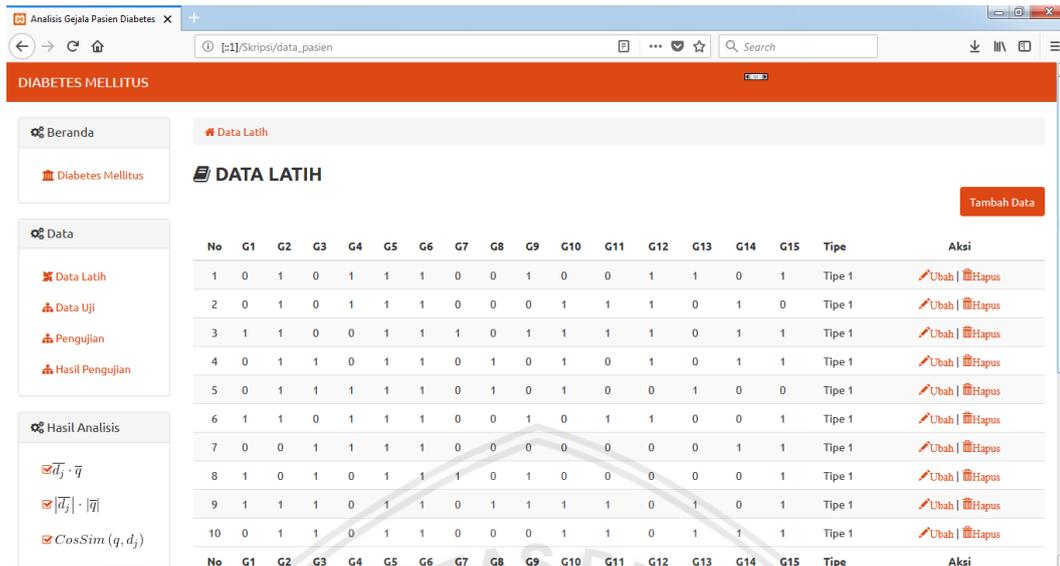
5.3.2 Implementasi Antarmuka Perhitungan *Neighbor Weighted K-Nearest Neighbor*

Pada halaman perhitungan metode *Neighbor Weighted K-Nearest Neighbor* terdapat beberapa tab menu yaitu data dan hasil analisis. Pada halaman data berisi tentang data latih dan data uji. Pada halaman hasil analisis berisi tentang langkah-langkah metode *Neighbor Weighted K-Nearest Neighbor*.

5.3.2.1 Implementasi Antarmuka Data Latih

Pada halaman data latih akan menampilkan tabel data yang telah dimasukkan pada *database*. Tabel yang ditampilkan berupa nomor data, atribut-atribut yang digunakan dan

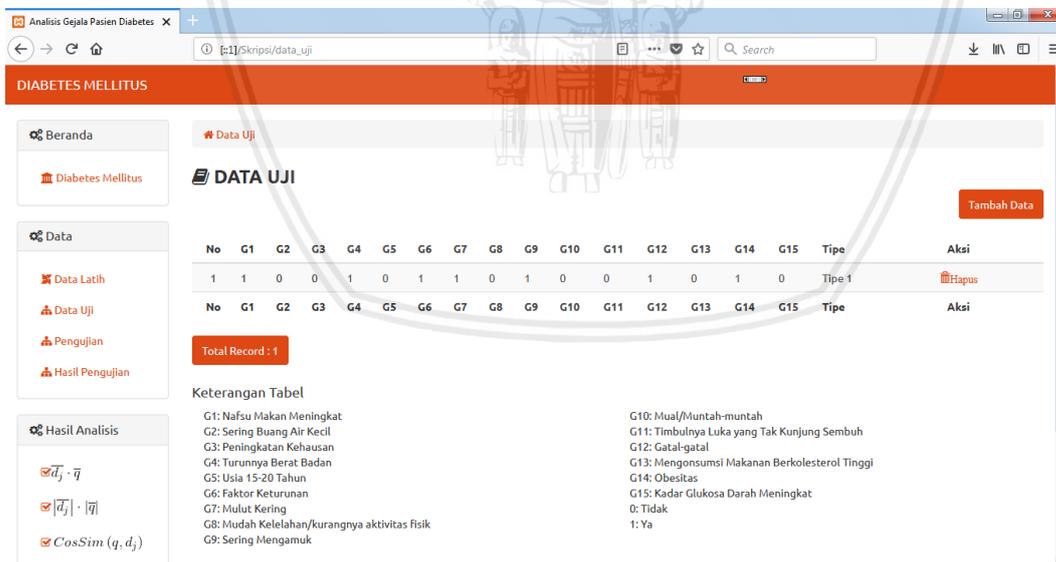
tipe dari *diabetes mellitus*. Pada Halaman data latih dapat menambahkan data, menghapus data dan mengubah data. Implementasi antarmuka data latih ditunjukkan pada Gambar 5.2.



Gambar 5.2 Implementasi Antarmuka Data Latih

5.3.2.2 Implementasi Antarmuka Data Uji

Pada halaman data latih akan menampilkan hasil masukkan dari pengguna yang disimpan pada *database*. Halaman data latih berisi tentang nomor data, atribut-atribut beserta penjelasannya, dapat menambahkan data, dan menghapus data. Implementasi antarmuka data uji ditunjukkan pada Gambar 5.3.



Gambar 5.3 Implementasi Antarmuka Data Uji

5.3.2.3 Implementasi Antarmuka Perhitungan *Cosine similarity*

Pada halaman antarmuka perhitungan *Cosine similarity* akan menampilkan hasil dari perhitungan *Cosine similarity*. Implementasi antarmuka perhitungan *Cosine similarity* ditunjukkan pada Gambar 5.4, Gambar 5.5 dan Gambar 5.6.

DIABETES MELLITUS

Beranda

Diabetes Mellitus

Data

Data Latih

Data Uji

Pengujian

Hasil Pengujian

Hasil Analisis

$\overline{d_j} \cdot \bar{q}$

$|\overline{d_j}| \cdot |\bar{q}|$

$CosSim(q, d_j)$

Hasil Pengurutan

Langkah 1

No	Hasil	Tipe
1	4	Tipe 1
2	4	Tipe 1
3	6	Tipe 1
4	3	Tipe 1
5	2	Tipe 1
6	5	Tipe 1
7	3	Tipe 1
8	4	Tipe 1
9	3	Tipe 1
10	2	Tipe 1

Gambar 5.4 Implementasi Antarmuka Perhitungan *Cosine Similarity*

DIABETES MELLITUS

Beranda

Diabetes Mellitus

Data

Data Latih

Data Uji

Pengujian

Hasil Pengujian

Hasil Analisis

$\overline{d_j} \cdot \bar{q}$

$|\overline{d_j}| \cdot |\bar{q}|$

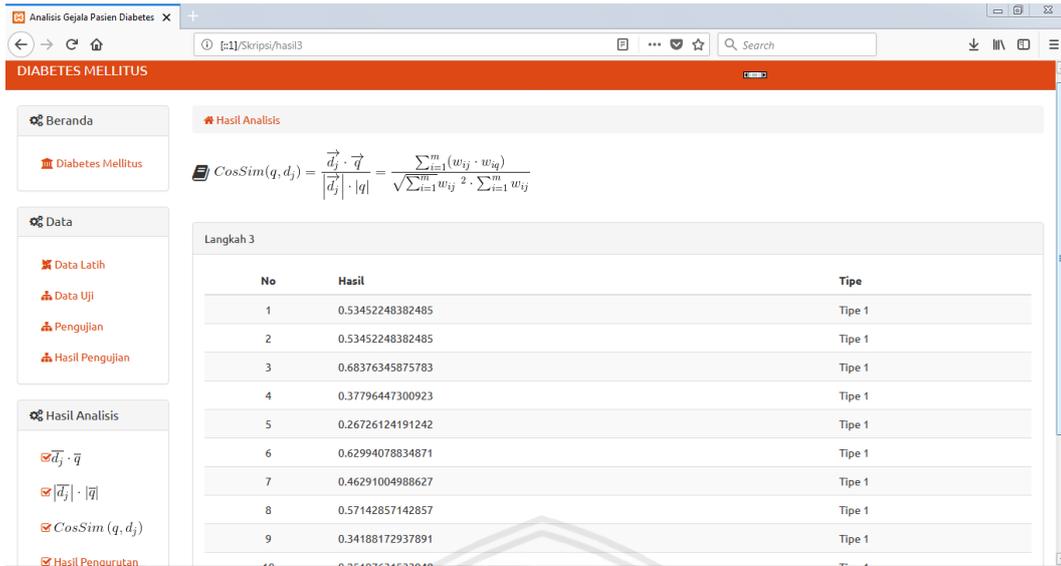
$CosSim(q, d_j)$

Langkah 2

No	Data Latih	Data Uji	Hasil Perkalian
1	2.4494897427832	2.6457513110646	6.4807406984079
2	2.6457513110646	2.6457513110646	7
3	3.1622776601684	2.6457513110646	8.3666002653408
4	3	2.6457513110646	7.9372539331938
5	2.8284271247462	2.6457513110646	7.4833147735479
6	3	2.6457513110646	7.9372539331938
7	3.1622776601684	2.6457513110646	8.3666002653408
8	2.8284271247462	2.6457513110646	7.4833147735479
9	2.6457513110646	2.6457513110646	7
10	2.8284271247462	2.6457513110646	7.4833147735479

Gambar 5.5 Implementasi Antarmuka Perhitungan *Cosine Similarity*

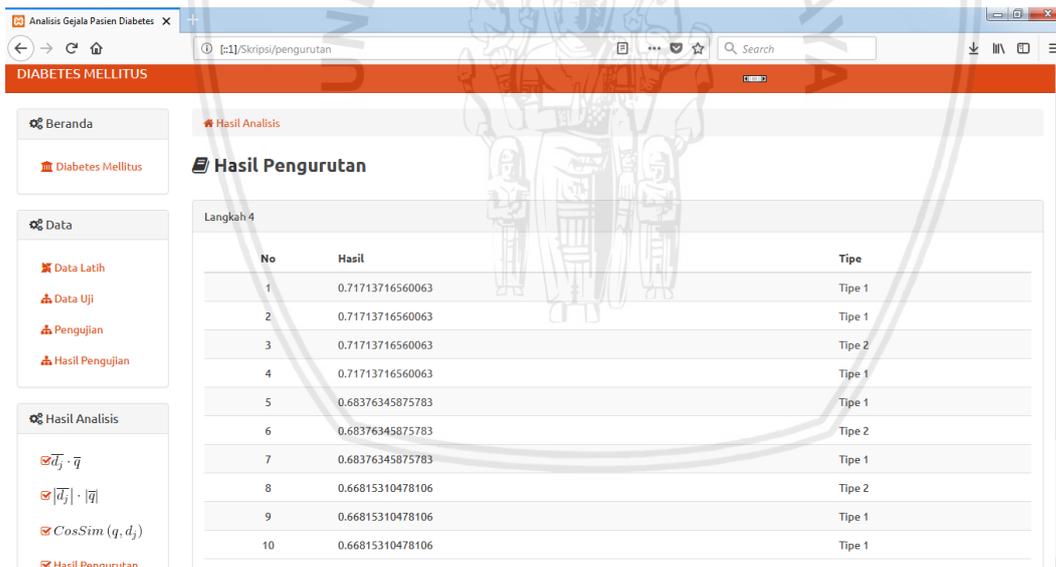




Gambar 5.6 Implementasi Antarmuka Perhitungan *Cosine Similarity*

5.3.2.4 Implementasi Antarmuka Pengurutan Hasil *Cosine Similarity*

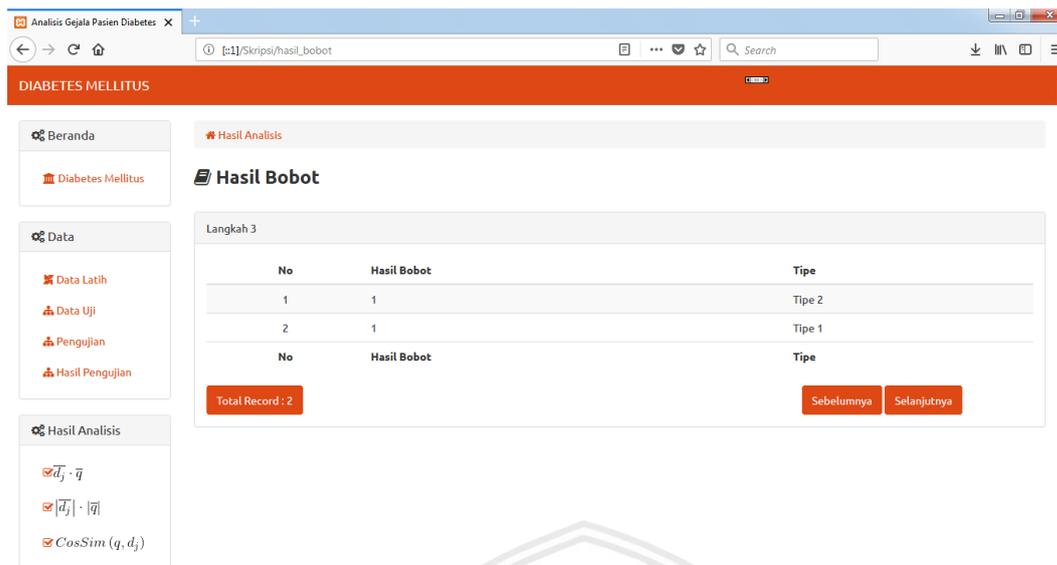
Pada halaman antarmuka ini akan mengurutkan hasil dari *Cosine Similarity* mulai dari terbesar hingga terkecil. Implementasi antarmuka pengurutan hasil *Cosine Similarity* ditunjukkan pada Gambar 5.7.



Gambar 5.7 Implementasi Antarmuka Pengurutan Hasil *Cosine Similarity*

5.3.2.5 Implementasi Antarmuka Hasil Perhitungan Bobot

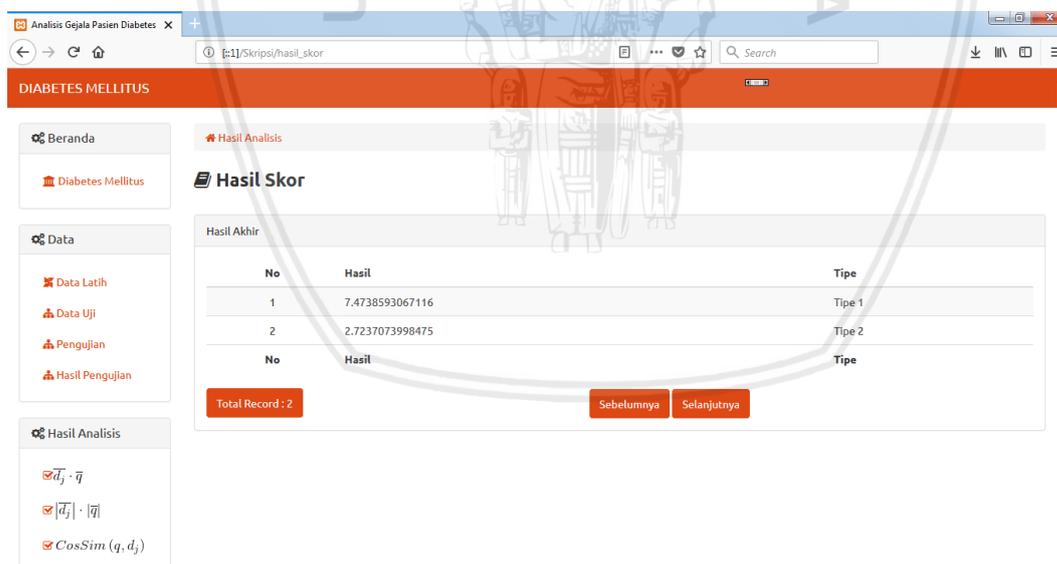
Pada halaman ini akan menampilkan hasil perhitungan untuk bobot setiap tipe diabetes mellitus. Pemberian bobot ini akan digunakan untuk menghitung nilai skor. Implementasi antarmuka hasil perhitungan bobot ditunjukkan pada Gambar 5.8.



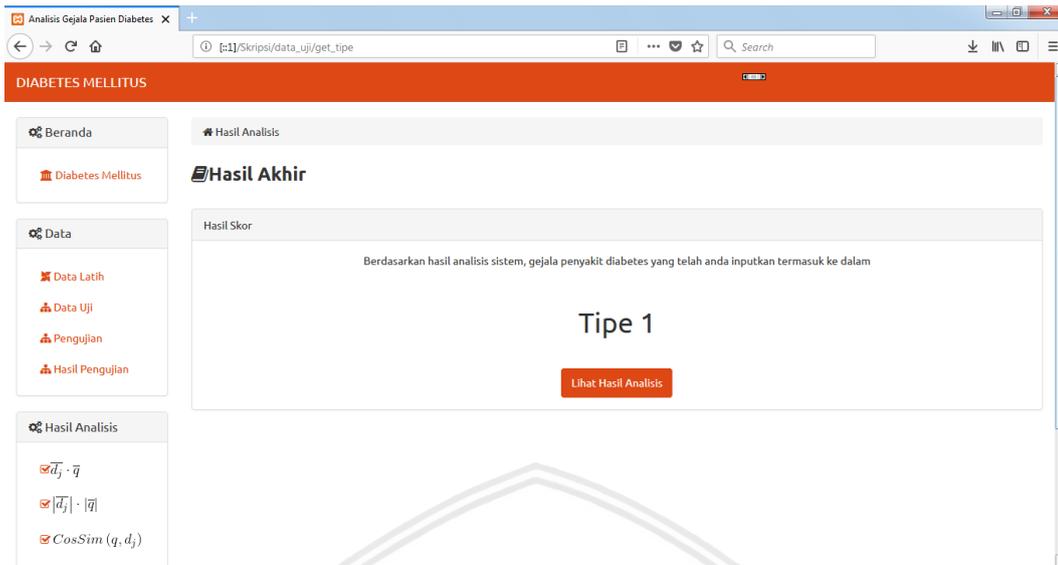
Gambar 5.8 Implementasi Antarmuka Hasil Perhitungan Bobot

5.3.2.6 Implementasi Antarmuka Hasil Perhitungan Skor

Pada halaman ini akan menampilkan hasil perhitungan skor dengan mengambil proses *Cosine Similarity* yang telah diurutkan dan diambil berdasarkan nilai K yang telah ditentukan. Selain itu terdapat halaman hasil akhir yang digunakan untuk menampilkan tipe diabetes mellitus. Implementasi antarmuka hasil perhitungan skor ditunjukkan pada Gambar 5.9 dan halaman hasil akhir ditunjukkan pada Gambar 5.10.



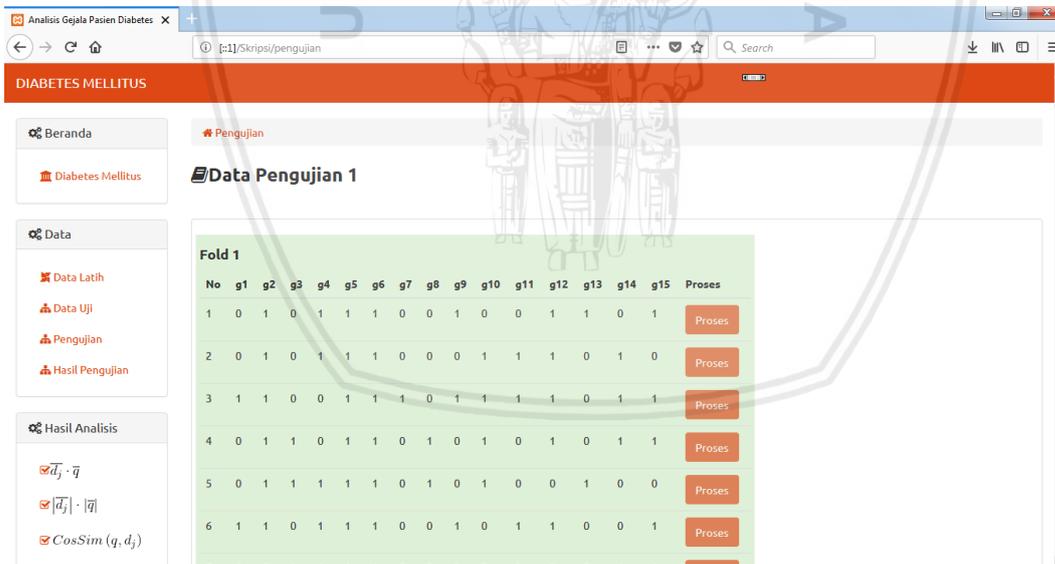
Gambar 5.9 Implementasi Antarmuka Hasil Perhitungan Skor



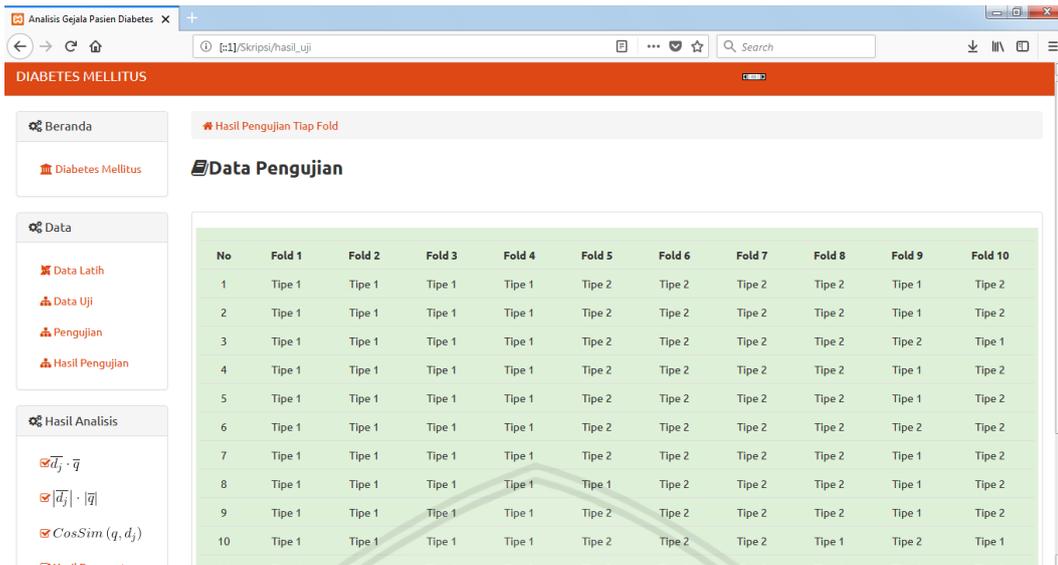
Gambar 5.10 Implementasi Antarmuka Hasil Perhitungan Skor

5.3.3 Implementasi Antarmuka Pengujian *K-fold Cross Validation*

Pada halaman ini akan menampilkan proses pengujian pada setiap *fold* dan hasil dari pengujian *K-Fold Cross Validation*. Implementasi antarmuka pengujian dapat dilihat pada gambar 5.11 dan implementasi hasil pengujian dapat dilihat pada gambar 5.12.



Gambar 5.11 Implementasi Antarmuka Pengujian



Gambar 5.12 Implementasi Antarmuka Hasil Pengujian



BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini menjelaskan tentang pengujian dan analisis sistem hasil penerapan metode *Neighbor Weighted K-Nearest Neighbor* dalam klasifikasi penyakit diabetes mellitus. Pengujian yang akan dijelaskan dalam bab ini, yaitu pengujian *K-Fold Cross Validation*. Pengujian dilakukan untuk mengetahui tingkat akurasi sistem.

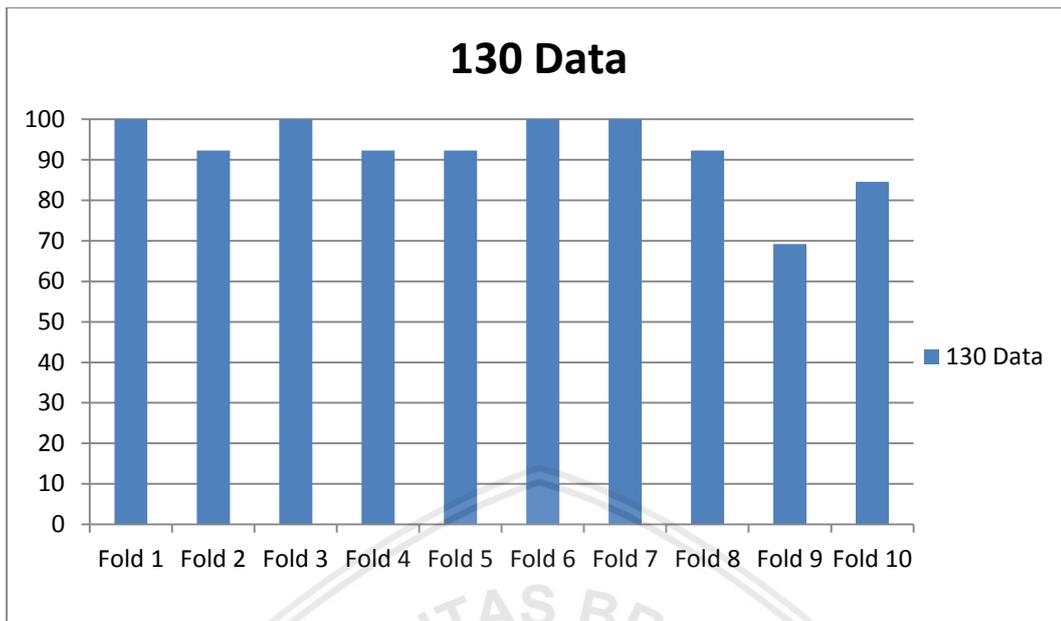
6.1 Pengujian dan Analisis *K-Fold Cross Validation*

Pengujian dengan menggunakan *K-Fold Cross Validation* bertujuan untuk mengetahui akurasi sistem berdasarkan data uji terhadap data latih. Data yang digunakan adalah data uji sebanyak 13 data, nilai $K=15$ dan nilai $E=2$. Pada pengujian ini diperoleh hasil yang berbeda-beda terhadap akurasi sistem. Hasil pengujian *K-Fold Cross Validation* dapat dilihat pada Table 4.8.

Tabel 6.1 Hasil Pengujian dan Analisis *K-Fold Cross Validation*

Jumlah data uji	Nilai E	Nilai K	Jumlah Data Latih	Akurasi Data Latih (%)
13	2	15	117	100
			117	92.3
			117	100
			117	92.3
			117	92.3
			117	100
			117	100
			117	92.3
			117	69.2
			117	84.6
Rata-Rata Akurasi (%)				92.3

Pengujian K-Fold Cross Validation



Gambar 6.1 Grafik Pengujian K-Fold Cross Validation

Berdasarkan pada Tabel 6.1 hasil rata-rata akurasi pada keseluruhan *fold* adalah 92.3% . Akurasi terendah didapatkan pada *fold* 9 dengan akurasi sebesar 69.2%, sedangkan akurasi tertinggi sebesar 100% didapatkan pada beberapa *fold* yaitu *fold* 1, *fold* 3, *fold* 6, dan *fold* 7. Dalam hal ini akurasi terendah didapatkan karena pada saat mengambil data sebanyak nilai $K=15$ yang telah diurutkan dari terbesar hingga terkecil, memiliki data yang lebih banyak pada tipe 1 atau tipe 2. Hal tersebut yang menjadi perbedaan antara hasil klasifikasi data keluaran pada sistem dan data klasifikasi rumah sakit. Selain itu pengujian ini digunakan untuk mengetahui hasil klasifikasi pada setiap data uji atau data *fold* terhadap data latih.

6.2 Pengujian dan Analisis Pengaruh Nilai E

Pengujian pengaruh nilai E bertujuan untuk mengetahui perubahan akurasi sistem ketika nilai E berubah. Pengujian ini dilakukan dengan mengubah nilai E sebanyak tiga kali. Data yang digunakan adalah data uji sebanyak 13 data dan nilai $K=15$. Hasil pengujian nilai E terhadap data latih dapat dilihat pada tabel 6.2, tabel 6.3 dan tabel 6.4.

Tabel 6.2 Hasil Pengujian dan Analisis Pengaruh Nilai E=2

Jumlah data uji	Nilai K	Nilai E	Jumlah Data Latih	Akurasi Data Latih (%)
13	15	2	117	100
			117	92.3
			117	100
			117	92.3
			117	92.3
			117	100



			117	100
			117	92.3
			117	69.2
			117	84.6
Rata-Rata Akurasi (%)				92.3

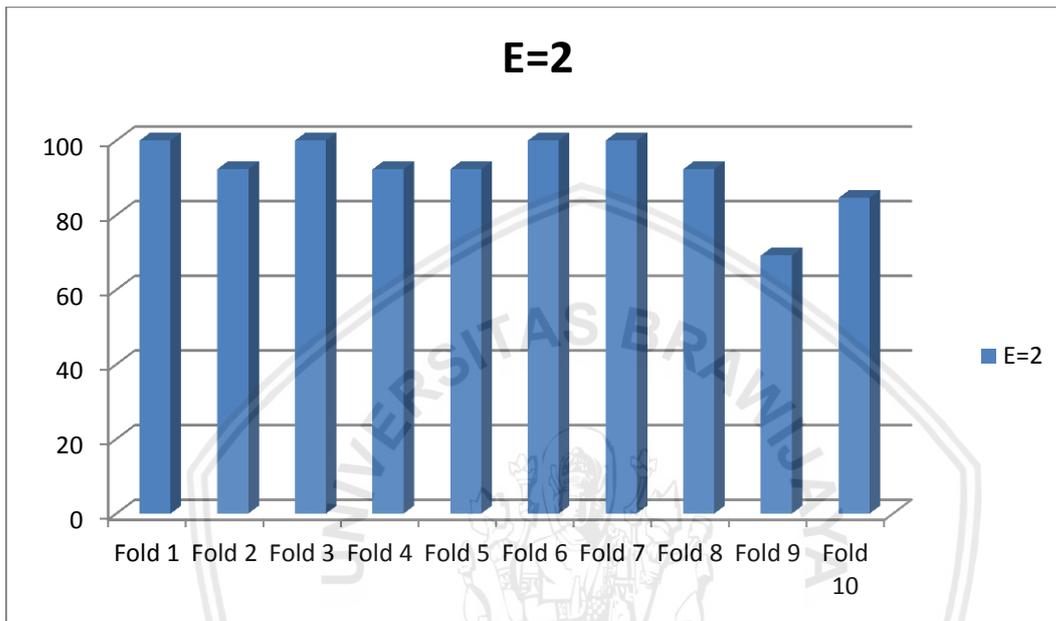
Tabel 6.3 Hasil Pengujian dan Analisis Pengaruh Nilai E=4

Jumlah data uji	Nilai K	Nilai E	Jumlah Data Latih	Akurasi Data Latih (%)
13	15	4	117	100
			117	92.3
			117	100
			117	92.3
			117	92.3
			117	100
			117	100
			117	92.3
			117	69.2
			117	84.6
Rata-Rata Akurasi (%)				92.3

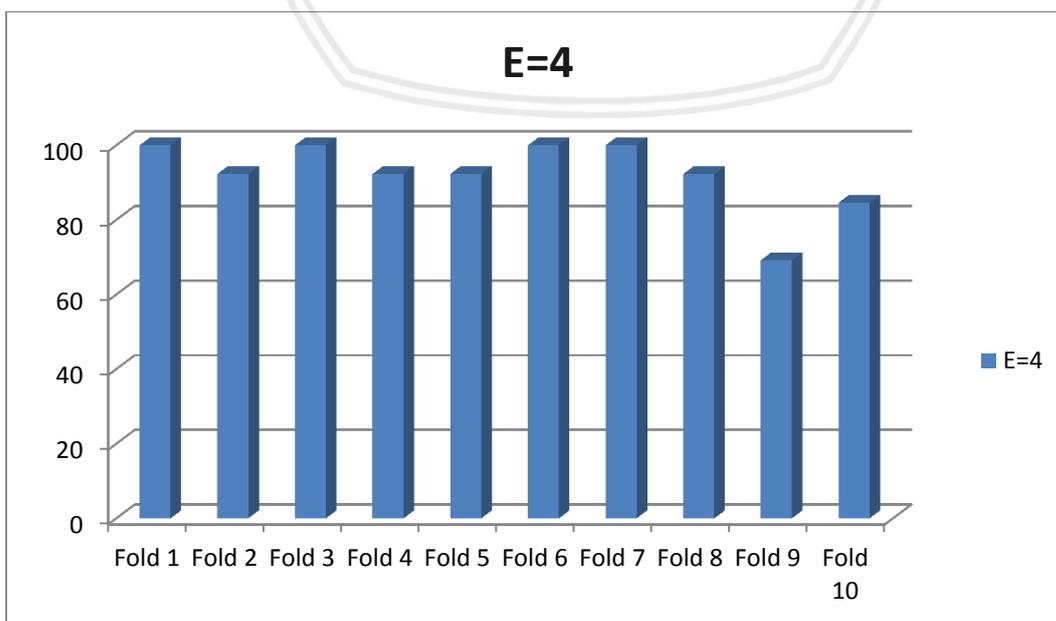
Tabel 6.4 Hasil Pengujian dan Analisis Pengaruh Nilai E=6

Jumlah data uji	Nilai K	Nilai E	Jumlah Data Latih	Akurasi Data Latih (%)
13	15	6	117	100
			117	92.3
			117	100
			117	92.3
			117	92.3
			117	100
			117	100

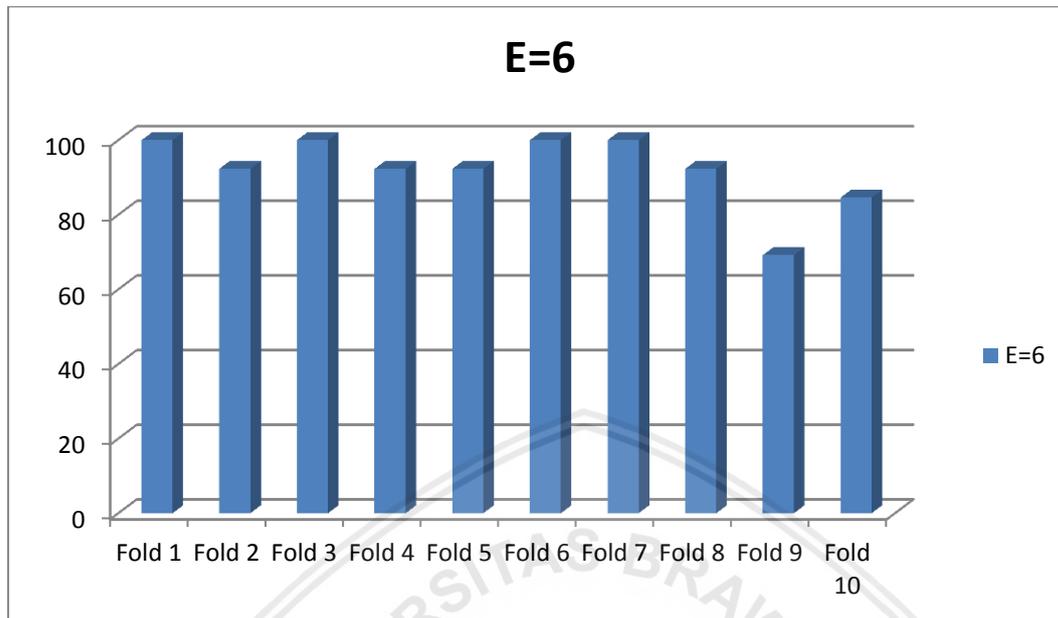
			117	92.3
			117	69.2
			117	84.6
Rata-Rata Akurasi (%)				92.3



Gambar 6.2 Grafik Pengujian Pengaruh Nilai E=2



Gambar 6.3 Diagram Pengujian Pengaruh Nilai E=4



Gambar 6.4 Grafik Pengujian Pengaruh Nilai E=6

Berdasarkan pada tabel 6.2 hingga tabel 6.4, hasil akurasi pengaruh nilai E pada masing-masing *fold* cenderung stabil. Hasil akurasi tertinggi terdapat pada *fold 1*, *fold 3*, *fold 6* dan *fold 7* yaitu sebesar 100%. Kemudian hasil akurasi terendah didapatkan pada *fold 9* yaitu sebesar 69.2%. Dalam hal ini menunjukkan bahwa perubahan nilai E tidak terlalu berpengaruh pada hasil akurasi data latih klasifikasi penyakit diabetes mellitus.

BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan hasil dari pengujian dan analisis pada penerapan metode *Neighbor Weighted K-Nearest Neighbor* dalam klasifikasi penyakit *diabetes Mellitus* dapat disimpulkan sebagai berikut:

1. Cara untuk Klasifikasi penyakit *diabetes mellitus* dengan metode NWKNN adalah memasukkan data pasien yang memiliki 15 atribut gejala. Kemudian data tersebut akan diproses untuk menentukan nilai *similarity* antara data latih dan data uji, yang nantinya akan diproses hingga mendapatkan nilai skor. Dimana tipe *diabetes mellitus* yang mendapatkan nilai tertinggi pada hasil skor yang masuk dalam klasifikasi.
2. Berdasarkan hasil pengujian *K-Fold Cross Validation*, didapatkan rata-rata akurasi sebesar 92.3%. Akurasi tertinggi pada *fold 1, fold 3, fold 6, dan fold 7* terhadap data latih sebanyak 117 data, yaitu dengan mendapatkan akurasi sebesar 100%. Sedangkan pada *fold 9* menunjukkan hasil akurasi terendah dengan akurasi sebesar 69.2%. Hal ini menunjukkan bahwa terdapat perbedaan hasil klasifikasi keluaran sistem dan hasil klasifikasi rumah sakit karena saat mengambil data sebanyak nilai $K=15$ yang telah diurutkan dari terbesar hingga terkecil, memiliki data lebih banyak pada tipe 1 atau tipe 2 yang membuat hasil klasifikasi berbeda.
3. Berdasarkan pada hasil pengujian pengaruh nilai E , didapatkan akurasi pada masing-masing *fold* cenderung stabil tidak terjadi adanya perubahan. Dalam hal ini menunjukkan bahwa perubahan nilai E tidak terlalu berpengaruh pada hasil akurasi data latih klasifikasi penyakit diabetes mellitus.

1.2 Saran

Saran yang dapat diberikan dari penelitian penerapan metode NWKNN dalam klasifikasi penyakit diabetes mellitus untuk penelitian selanjutnya adalah sebagai berikut:

1. Data yang digunakan pada penelitian ini sebanyak 130 data, dimana data tersebut bersifat seimbang pada masing-masing tipe diabetes mellitus. Diharapkan penelitian selanjutnya untuk menambahkan jumlah data dan fitur.

Untuk peneliti selanjutnya diharapkan dapat mengembangkan sistem dengan kombinasi dari metode NWKNN, agar membantu meningkatkan hasil akurasi.

DAFTAR REFERENSI

- Fadila, P. N., Indriati, & Ratnawati, D. E. (2016). Identifikasi Jenis Attention Deficit Hyperactivity Disorder (ADHD) Pada Anak Usia Dini Menggunakan Neighbor Weighted K-Nearest Neighbor (NWKNN). *Jurnal Teknologi Informasi dan Ilmu Komputer*, 194-200.
- Fernanda, S. I., Ratnawati, D. E., & Adikara, P. P. (2017). Identifikasi Penyakit Diabetes Mellitus Menggunakan Metode Modified K-Nearest Neighbor (MKNN). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 1(-), 507-513.
- Indriati, & Ridok, A. (2016). Sentiment Analysis for Review Mobile Application Using Neighbor Weighted K-Nearest Neighbor(NWKNN). *Journal of Enviromental Engineering & Sustainable Technology*, 23-32.
- Johar, A., Yanosma, D., & Anggriani, K. (2016). Implementasi Metode K-Nearest Neighbor (KNN) dan Simple Additive . *Jurnal Pseudocode*, III(-), 98-112.
- Khaleel, A. H., Al-Suhail, G. A., & Hussan, B. M. (2017). A Weighted Voting of K-Nearest Neighbor Algorithm for Diabetes Mellitus. *International Journal of Computer Science and Mobile Computing*, 43-51.
- Khamis, H. S., Cheruiyot, K. W., & Kimani, S. (2014). Application of K-Nearest Neighbor Classification in Medical Data Mining. *International Journal of Information and Communication Technology Research*, 4, 121-128.
- Krisandi, N., Helmi, & Prihandoso, B. (2013). Algoritma K-Nearest Neighbor Dalam Klasifikasi Data Hasil Produksi Kelapa Sawit Pada PT. MINAMAS Kecamatan Parindu. *Buletin Ilmiah Math. Stat. dan Terapannya (Bimaster)*, 02, 33-38.
- Rivaldi, A., Adikara, P. P., & Adinugroho, S. (2018). Klasifikasi Penyimpangan Tumbuh Kembang pada Anak Menggunakan Metode Neighbor Weighted K-Nearest Neighbor. *Pengembangan Teknologi Informasi dan Ilmu Komputer*, 2843-2850.
- Saxena, K., Khan, D., & Singh, S. (2014). Diagnosis of Diabetes Mellitus using K Nearest Neighbor Algorithm. *International Journal of Computer Science Trends and Technology*, 2(4), 36-43.
- Sutanto. (2010). *Cekal Penyakit Modern*. Yogyakarta: C.V ANDI OFFSET .
- Tan, S. (2005). Neighbor-Weighted K-Nearest Neighbor for Unbalance Text Corpus. *Expert System with Applications*, 667-671.
- Trisnawati, S. K., & Setyorogo, S. (2013). Faktor Risiko Kejadian Diabetes Melits Tipe II Di puskesmas Kecamatan engkareng Jakarta Barat Tahun 2012. *Jurnal Ilmiah Kesehatan*, 5, 6-11.



V, V. V., & Ravikumar, A. (2014). Study of Data Mining Algorithm for Prediction and Diagnosis of Diabetes Mellitus. *International Journal of Computer Application* , 12-16.

