

**PENENTUAN PENERIMA BEASISWA MENGGUNAKAN
METODE K-NEAREST NEIGHBOR DAN WEIGHTED PRODUCT
(STUDI KASUS : MTS AL-ISHLAH TEMBOKREJO MUNCAR)**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Alamsyah Ramadhan

NIM: 145150207111112



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

MALANG

2021



PENGESAHAN

PENENTUAN PENERIMA BEASISWA MENGGUNAKAN METODE K-NEAREST
NEIGHBOR DAN WEIGHTED PRODUCT (STUDI KASUS : MTS AL-ISHLAH
TEMBOKREJO MUNCAR)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :

Alamsyah Ramadhan


NIM: 145450207111112

Skripsi ini telah diuji dan dinyatakan lulus pada
21 Juli 2021


Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II


Wayan Firdaus Mahmudy, S.Si., M.T., Ph.D.

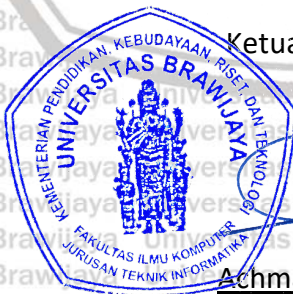
NIK. 197209191997021001


Edy Santoso, S.Si., M.Kom.

NIK. 197404142003121004

Mengetahui

Ketua Jurusan Teknik Informatika




Achmad Basuki, S.T., M.MG., Ph.D.

NIK. 197411182003121002

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, 22 Juli 2021



Alamsyah Ramadhan

NIM: 145150207111112

UNIVERSITAS BRAWI



KATA PENGANTAR

Puji syukur kehadirat Allah SWT yang telah melimpahkan rahmat, taufik dan hidayah-Nya sehingga skripsi yang berjudul “Penentuan Penerima Beasiswa Menggunakan Metode K-Nearest Neighbor Dan Weighted Product (Studi Kasus : Mts Al-Ishlah Tembokrejo Muncar)” ini dapat terselesaikan untuk memenuhi sebagian persyaratan dalam memperoleh gelar Sarjana Komputer.

Penyusunan skripsi ini tidak akan berhasil tanpa bantuan dari berbagai pihak. Oleh karena itu, penulis ingin menyampaikan rasa hormat dan terima kasih kepada:

1. Bapak Wayan Firdaus Mahmudy, S.Si., M.T., Ph.D., selaku dosen pembimbing skripsi yang selalu membimbing dan menuntun penulis hingga terselesaikannya skripsi ini.
2. Bapak Edy Santoso, S.Si., M.Kom., selaku dosen pembimbing dua skripsi yang selalu membimbing dan menuntun penulis hingga terselesaikannya skripsi ini.
3. Bapak Achmad Basuki, S.T., M.MG., Ph.D., selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya.
4. Bapak Adhitya Bhawiyuga, S.Kom., M.Sc., selaku Ketua Program Studi Teknik Informatika Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya.
5. Ayahanda, Ibunda, dan seluruh keluarga besar atas segala nasehat, kasih sayang, perhatian dan kesabarannya di dalam membesarkan dan mendidik penulis, serta yang senantiasa tiada henti-hentinya memberikan doa dan semangat demi terselesaikannya laporan ini.
6. Teman – teman mahasiswa yang berada di Kota Malang yang telah membantu menyelesaikan penelitian ini dan memberikan semangat walau ditengah adanya pandemi virus korona ini.

Penulis menyadari bahwa skripsi ini memiliki banyak kekurangan dan ketidaksempurnaan. Untuk itu penulis mengharapkan kritik dan saran yang dapat membangun. Semoga skripsi ini dapat memberikan manfaat baik bagi perkembangan ilmu pengetahuan dan teknologi.

Malang, 26 Juli 2021



Alamsyah Ramadhan

ABSTRAK

Masyarakat Indonesia yang berkualitas, maju, modern dan mandiri adalah tujuan dari pembangunan Pendidikan nasional. Dalam mewujudkan keberhasilan pendidikan dalam membangun karakter manusia dibutuhkan adanya pendidikan yang akurat, karena pendidikan memiliki pengaruh besar dalam tercapainya tujuan dari pembangunan nasional secara keseluruhan. Berdasarkan data dari BPS Kabupaten Banyuwangi tahun 2017, di desa Tembokrejo anak usia 10 – 14 tahun terdapat 2398 anak, sedangkan hanya terdapat 2 SMP dan 1 Mts. Biaya sekolah menjadi masalah yang dimana mayoritas penduduk bermata pencaharian petani dan nelayan. Dari persoalan tersebut ada solusi yang bernama beasiswa. Beasiswa adalah bentuk pemberian bantuan yang bisa berupa keuangan atau pendidikan yang diberikan untuk pelajar atau mahasiswa yang difungsikan untuk kelangsungan masa pendidikan. Dengan menggunakan algoritme K-Nearest Neighbor dan Weighted Product bisa mengklasifikasikan dan menentukan perankingan calon penerima beasiswa di MTs Al-Ishlah Muncar. Data yang digunakan diperoleh dari pihak sekolah. Dalam penelitian ini menggunakan pengujian tingkat akurasi antara data yang dihasilkan oleh sistem dengan data yang diberikan pihak sekolah dan data perankingan yang diperoleh dari pakar. Tingkat akurasi perhitungan sistem dengan algoritme K-Nearest Neighbor sebesar 94,1% dan untuk algoritme Weighted Product memiliki tingkat akurasi berturut-turut sebesar 33,3%, 54,9% dan 70,5%.

Kata kunci: Beasiswa, K-Nearest Neighbor, Weighted Product, Pendidikan

ABSTRACT

A quality, advanced, modern and independent Indonesian society is the goal of national education development. In realizing the success of education in building human character, accurate education is needed, because education has a major influence in achieving the goals of national development as a whole. Based on data from BPS Banyuwangi Regency in 2017, in Tembokrejo village children aged 10-14 years there are 2398 children, while there are only 2 junior high schools and 1 MTs. School fees are a problem where the majority of the population live as farmers and fishermen. From this problem there is a solution called scholarship. Scholarships are a form of assistance that can be in the form of financial or educational assistance given to students or students that are used for the continuation of the education period. By using the K-Nearest Neighbor and Weighted Product algorithms, you can classify and determine the ranking of prospective scholarship recipients at MTs Al-Ishlah Muncar. The data used were obtained from the school. In this research, testing the level of accuracy between the data generated by the system with the data provided by the school and ranking data obtained from experts. The level of accuracy of system calculations with the K-Nearest Neighbor algorithm is 94.1% and the Weighted Product algorithm has an accuracy rate of 33.3%, 54.9% and 70.5%, respectively.

Keywords: *Scholarship, K-Nearest Neighbor, Weighted Product, Education*



DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
ABSTRACT.....	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xi
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan.....	2
1.4 Manfaat.....	2
1.5 Batasan Masalah.....	2
1.6 Sistematika Pembahasan	2
BAB 2 LANDASAN KEPUSTAKAAN	4
2.1 Kajian Pustaka	4
2.2 Dasar Teori	6
2.2.1 Data Mining	6
2.2.2 Algoritme <i>K-Nearest Neighbor</i>	6
2.2.3 <i>Weighted Product</i>	7
2.2.4 Akurasi	8
BAB 3 METODOLOGI.....	9
3.1 Tipe Penelitian.....	10
3.2 Strategi Penelitian.....	10
3.3 Teknik Pengumpulan Data.....	10
3.4 Peralatan Pendukung Penelitian.....	10
3.5 Implementasi Algoritme	11
BAB 4 PERANCANGAN.....	12

4.1 Deskripsi Umum Sistem	12
4.2 Penyelesaian Masalah Menggunakan <i>K-Nearest Neighbor</i> -WP	13
4.2.1 Normalisasi Data	14
4.2.2 Penyelesaian dengan <i>K-Nearest Neighbor</i>	16
4.2.3 Penyelesaian dengan <i>Weighted Product</i> (WP)	17
4.3 Manualisasi Menggunakan Metode <i>K-Nearest Neighbor</i> -WP	21
4.3.1 Normalisasi	23
4.3.2 Manualisasi <i>K-Nearest Neighbor</i>	24
4.3.3 Manualisasi <i>Weighted Product</i>	25
4.4 Perancangan Antar Muka	27
4.4.1 Antarmuka halaman utama	27
4.4.2 Antarmuka Halaman Hasil	28
4.5 Perancangan Pengujian	28
4.5.1 Rancangan Pengujian Perbandingan Perankingan	29
4.5.2 Rancangan pengujian pengaruh nilai K	29
BAB 5 IMPLEMENTASI	30
5.1 Spesifikasi Sistem	30
5.1.1 Spesifikasi <i>Hardware</i>	30
5.1.2 Spesifikasi <i>Software</i>	31
5.2 Batas Implementasi	31
5.3 Implementasi Membaca File Inputan	31
5.4 Implementasi Algoritme KNN-WP	32
5.4.1 Implementasi Normalisasi Data	32
5.4.2 Implementasi Perhitungan Jarak <i>Ecludian</i>	35
5.4.3 Implementasi Klasifikasi Data Uji Berdasarkan Data Latih	36
5.4.4 Implementasi Input Data Bobot dan Perbaikan Bobot	38
5.4.5 Implementasi Perhitungan Vektor <i>S</i>	39
5.4.6 Implementasi Perhitungan Vektor <i>V</i>	41
5.4.7 Implementasi Perankingan Berdasarkan Vektor <i>V</i>	42
5.5 Implementasi Antarmuka	44
5.5.1 Antarmuka Halaman Utama	44
5.5.2 Antarmuka Halaman Hasil	44
BAB 6 PENGUJIAN DAN ANALISIS	46

6.1 Pengujian Akurasi Algoritme <i>K-Nearest Neighbor</i>	46
6.2 Pengujian Pengaruh Nilai K	48
6.3 Pengujian Akurasi Algoritme <i>Weighted Product</i>	48
BAB 7 PENUTUP	50
7.1 Kesimpulan	50
7.2 Saran	50
DAFTAR REFERENSI	51
LAMPIRAN	52



DAFTAR TABEL

Tabel 2.1 Kajian Pustaka	5
Tabel 4.1 Data Latih	22
Tabel 4.2 Data Uji	23
Tabel 4.3 Normalisasi Data Latih	23
Tabel 4.4 Normalisasi Data Uji	24
Tabel 4.5 Perhitungan Jarak Ecludian Antara Data Latih dan Data Uji	24
Tabel 4.6 Klasifikasi Data Uji	25
Tabel 4.7 Hasil Klasifikasi Data uji	25
Tabel 4.8 Bobot Kriteria	26
Tabel 4.9 Perbaikan Bobot Kriteria	26
Tabel 4.10 Perhitungan Vektor (Si)	26
Tabel 4.11 Menghitung Vektor (Vi)	27
Tabel 4.12 Rancangan Pengujian Perbandingan Perankingan	29
Tabel 4.13 Perbandingan Hasil Perhitungan Pakar dan Hasil Perhitungan Sistem	29
Tabel 4.14 Rancangan Pengujian Pengaruh Nilai K	29
Tabel 5.1 Spesifikasi Hardware	30
Tabel 5.2 Spesifikasi Software	31
Tabel 6.1 Perbandingan Hasil Keluaran Sistem dengan Data Pihak Sekolah dan Data Pihak Pakar	46
Tabel 6.2 Hasil Pengujian Pengaruh Nilai K	48
Tabel 6.3 Perbandingan Hasil Keluaran Sistem dengan Data Pakar	49



DAFTAR GAMBAR

Gambar 3.1 Diagram Alir Metode Penelitian.....	9
Gambar 4.1 Diagram Alir Perancangan Sistem.....	12
Gambar 4.2 Diagram Alir Penyelesaian Masalah Menggunakan	14
Gambar 4.3 Diagram Alir Normalisasi Data	15
Gambar 4.4 Diagram Alir Penyelesaian Dengan <i>K-Nearest Neighbor</i>	16
Gambar 4.5 Diagram Alir Perhitungan Jarak <i>Ecludian</i>	17
Gambar 4.6 Penyelesaian dengan <i>Weighted Product</i>	18
Gambar 4.7 Perbaikan Bobot Kriteria (Wi)	19
Gambar 4.8 Menghitung Vektor (Si)	20
Gambar 4.9 Menghitung Vektor (Vi)	21
Gambar 4.10 Rancangan Antarmuka Halaman Utama.....	28
Gambar 4.11 Rancangan Antarmuka Halaman Hasil.....	28
Gambar 5.1 Implementasi Sistem.....	30
Gambar 5.2 Antarmuka Halaman Utama Sistem.....	44
Gambar 5.3 Antarmuka Halaman Hasil Sistem.....	45

DAFTAR LAMPIRAN

Lampiran 1.1 Data Siswa MTs Al-Ishlah T.A 2018/2019	52
---	----



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Masyarakat Indonesia yang berkualitas, maju, modern dan mandiri adalah tujuan dari pembangunan pendidikan nasional. Dalam mewujudkan keberhasilan pendidikan dalam membangun karakter manusia dibutuhkan adanya pendidikan yang akurat, karena pendidikan memiliki pengaruh besar dalam tercapainya tujuan dari pembangunan nasional secara keseluruhan (Sudarsana, 2016).

Ada lima alasan kenapa pendidikan sangat penting. Pertama, pendidikan memberikan kita banyak pengetahuan. Kedua, pendidikan menjadi dasar untuk meniti karir. Ketiga, karakter yang baik dapat dibangun dengan pendidikan. Keempat, dengan pendidikan kita memiliki gambaran apa yang ada disekitar kita, sehingga tidak membuat kita tersesat. Kelima, pendidikan dapat membantu kemajuan bangsa.

Pada saat ini perkembangan teknologi maju dengan sangat pesat, alangkah baiknya perkembangan ini diikuti dengan pendidikan yang merata secara menyeluruh. Teknologi sudah menjadi bagian dari hidup, semua menjadi mudah karena teknologi. Hal ini yang menjadi dasar mengapa teknologi menjadi hal yang penting untuk saat ini.

Berdasarkan data dari BPS Kabupaten Banyuwangi tahun 2017, di desa Tembokrejo anak usia 10 – 14 tahun terdapat 2398 anak, sedangkan hanya terdapat 2 SMP dan 1 Mts. Biaya sekolah menjadi masalah yang dimana mayoritas penduduk bermata pencaharian petani dan nelayan. Dari persoalan tersebut ada solusi yang bernama beasiswa. Beasiswa adalah bentuk pemberian bantuan yang bisa berupa keuangan atau pendidikan yang diberikan untuk pelajar atau mahasiswa yang difungsikan untuk kelangsungan masa pendidikan.

Ada permasalahan dalam memberikan beasiswa, kuota penerima beasiswa yang sedikit dan jumlah bantuan yang terbatas. Jadi penyedia beasiswa harus memilih dengan bijak. Oleh karena itu dalam menentukan suatu pilihan bisa dilakukan dengan secara sistematis dengan beberapa metode. Salah satu metode pendukung yang cocok digunakan adalah *K-Nearest Neighbor*(KNN). karena metode ini merupakan sebuah metode yang menggunakan algoritma *supervised*, dengan menggunakan metode ini hasil dari *query instance* yang baru diklasifikasikan berdasarkan mayoritas dari kategori. Metode ini bertujuan untuk mengklasifikasikan data uji berdasarkan data latih. Sedangkan untuk mengatasi masalah pemeringkatan di dalam penelitian ini digunakan metode *Weighted Product*(WP). *Weighted Product* merupakan salah satu metode penyelesaian multi kriteria (Diah, 2013) dimana dalam menentukan penerima beasiswa memiliki banyak kriteria yang harus diperhatikan.

Berdasarkan penelitian sebelumnya dari (Kartika, 2017) mengenai penentuan siswa berprestasi di SMPN 3 Mejayan. Penelitian ini menggunakan

metode *K-Nearest Neighbor – Weighted Product* yang menggunakan 5 kriteria mempunyai akurasi secara berturut-turut sebesar 56,67% dan 76,67%. Sedangkan hasil perbandingan perankingan antara data pakar dengan data manual dari pihak sekolah dan data keluaran sistem untuk metode *Weighted Product* memiliki akurasi secara berturut-turut sebesar 11,1% dan 100%. Sehingga pada penelitian ini akan dibuat sebagai referensi untuk membantu guru dalam menentukan penerima beasiswa dengan menggunakan metode *K-Nearest Neighbor – Weighted Product*.

1.2 Rumusan Masalah

Rumusan masalah dalam penelitian ini yaitu:

1. Bagaimana menerapkan metode *K-Nearest Neighbor* dan *Weighted Product* ke dalam penentuan penerima beasiswa.
2. Berapa tingkat akurasi yang diberikan dari hasil implementasi metode *K-Nearest Neighbor* dan *Weighted Product* pada penentuan penerimaan beasiswa.

1.3 Tujuan

Tujuan dari penelitian ini dapat dituliskan sebagai berikut:

1. Menerapkan metode *K-Nearest Neighbor* dan *Weighted Product* untuk mengklasifikasi penentuan penerimaan beasiswa.
2. Menguji tingkat akurasi dari hasil perhitungan metode *K-Nearest Neighbor* dan *Weighted Product* untuk mengklasifikasi penentuan pemberian beasiswa.

1.4 Manfaat

Manfaat yang bisa diambil dari penelitian ini adalah untuk membantu dan memudahkan pihak pemberi beasiswa dalam penentuan penerimaan beasiswa.

1.5 Batasan Masalah

Batasan masalah yang ada dalam penelitian ini adalah:

1. Data yang digunakan sebanyak 221 data siswa MTs Al-Ishlah tahun ajaran 2018/2019.
2. Parameter yang digunakan ada 5 yaitu nilai rapor, tanggungan SPP, kelas, gaji orangtua dan jumlah tanggungan saudara.

1.6 Sistematika Pembahasan

Sistematika pembahasan dalam penelitian ini ada 6 bab, diantaranya sebagai berikut:

BAB I : PENDAHULUAN

Bab ini menjelaskan tentang latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, batasan masalah serta sistematika pembahasan dari penelitian ini.

BAB II : LANDASAN KEPUSTAKAAN

Bab ini menjelaskan tentang penelitian-penelitian terdahulu yang terhubung dalam penelitian ini dengan menggunakan metode *K-Nearest Neighbor* dan *Weighted Product*.

BAB III : METODOLOGI PENELITIAN DAN PERANCANGAN

Pada bab ini menjelaskan tentang metode penelitian yang digunakan dengan pendekatan penelitian baik dari implementasi, eksperimen, desain, kebutuhan, analisis dan pengujian.

BAB IV : IMPLEMENTASI

Bab ini menjelaskan tentang seluruh perancangan sistem yang akan diterapkan pada penelitian dalam klasifikasi untuk penentuan pemberian beasiswa kepada siswa.

BAB V : IMPLEMENTASI

Bab ini menjelaskan tentang pengujian dan analisis terhadap sistem untuk mengetahui tingkat keberhasilan dari metode *K-Nearest Neighbor* dan *Weighted Product*.

BAB VI : PENGUJIAN DAN ANALISIS

Pada bagian ini dilakukan pengujian terhadap sistem yang telah dibuat untuk mengetahui tingkat keberhasilan sistem dalam memberikan suatu solusi. Analisis terhadap hasil pengolahan data baik dari masukan maupun proses yang berjalan pada sistem hingga hasil keluaran sistem.

BAB VI : KESIMPULAN DAN SARAN

Setiap proses maupun hasil yang didapatkan selama penelitian akan dibahas secara ringkas pada bab kesimpulan. Saran yang diperoleh dari hasil penelitian dicantumkan dengan harapan akan ada penelitian yang jauh lebih baik untuk dilakukan dari penelitian sebelumnya.

BAB 2 LANDASAN KEPUSTAKAAN

Pada bab ini akan dijelaskan beberapa teori pendukung untuk penelitian penentuan penerima beasiswa di MTs Al-Ishlah menggunakan metode *K-Nearest Neighbor* dan *Weighted Product*. Teori yang akan dibahas mengenai metode *K-Nearest Neighbor* dan *Weighted Product* dalam Penerimaan Beasiswa.

2.1 Kajian Pustaka

Penelitian sebelumnya membahas tentang penentuan siswa-siswa yang berprestasi dengan menggunakan metode *K-Nearest Neighbor-Weighted Product*. Penelitian ini yang dilakukan di SMPN 3 Mejayan dan dalam penelitian ini menggunakan 30 data latih serta 30 data uji. Hasil yang diperoleh dari sistem akan dicocokkan dengan hasil yang diberikan sekolah. Berdasarkan hasil perbandingan antara data dari pihak pakar dengan data manual yang diberikan oleh sekolah dan data hasil perhitungan sistem untuk metode *K-Nearest Neighbor* memiliki akurasi secara sebesar 56,67% dan 76,67%. Sedangkan hasil perbandingan perankingan antara data dari pihak pakar dengan data manual yang diberikan oleh pihak sekolah dan data hasil perhitungan sistem untuk metode *Weighted Product* memiliki akurasi secara sebesar 11,1% dan 100% (Kartika, 2017).

Penelitian yang berikutnya dengan algoritme *Naïve Bayes-Weighted Product* mengklasifikasikan dan memberi rekomendasi jurusan kuliah bagi pelajar SMA. Hasil yang didapatkan dari penelitian ini adalah jurusan yang sistem rekomendasikan berdasarkan nilai-nilai calon mahasiswa. Data calon akan mahasiswa diklasifikasikan menjadi sebuah fakultas dengan menggunakan algoritme *Naïve Bayes*, kemudian data yang diterima dalam fakultas tertentu akan diproses dengan algoritma *Weighted Product* untuk menentukan rekomendasi jurusan. Penelitian ini mempunyai akurasi 95% pada pemilihan fakultas menggunakan *naïve-bayes* dan untuk akurasi jumlah data latih sebanyak 90% dan data uji sebanyak 10% dari total data yang menghasilkan nilai akurasi 93,33%..

Kemudian dari penelitian tentang perbandingan klasifikasi beasiswa toyota astra. Penelitian ini menggunakan *k-nearest neighbor* dan *naïve bayes* sebagai penentu metode klasifikasi pada sistem pendukung keputusan. Penelitian ini mengenai klasifikasi beasiswa yang membandingkan antara metode KNN dan metode *Naïve Bayes*. dimana didapatkan hasil penggunaan metode *K-Nearest Neighbor* memiliki nilai akurasi yang lebih baik dibandingkan dengan metode *Naïve Bayes* (Lestari, 2017).

Tabel 2.1 Kajian Pustaka

No.	Judul	Objek	Metode	Hasil Penelitian
1.	Penentuan Siswa Berprestasi Menggunakan Metode <i>K-Nearest Neighbor</i> dan <i>Weighted Product</i> (Studi Kasus : SMP Negeri 3 Mejayan)	30 data latih dan 30 data uji dengan 5 kriteria, yaitu : rapor, kedisiplinan, ekstrakurikuler, kepribadian, presensi	<i>K-Nearest Neighbor</i> dan <i>Weighted Product</i>	Metode <i>K-Nearest Neighbor</i> sebesar 56,67% dan 76,67%. Untuk perankingan 11,1% dan 100%
2.	Klasifikasi dan Rekomendasi Jurusan Kuliah Bagi Pelajar SMA Menggunakan Algoritme <i>Naïve Bayes-WP</i>	Menggunakan 182 data	<i>Naïve Bayes-WP</i>	Akurasi yang didapat dengan menggunakan seluruh data yaitu 95%, dan untuk akurasi jumlah data latih sebanyak 90% dan data uji sebanyak 10% dari total data yang menghasilkan nilai akurasi 93,33%.
3.	Perbandingan Klasifikasi Beasiswa Toyota Astra menggunakan <i>K-Nearest Neighbor</i> dan <i>Naïve bayes</i> Sebagai Penentu Metode Klasifikasi pada sistem Pendukung Keputusan Penerimaan Beasiswa Toyota Astra	Data didapatkan dari penerimaan beasiswa Toyota Astra di ITS pada tahun 2015 dan 2016, 5 Parameter Beasiswa, 2 status penerimaan, 370 data	<i>K-Nearest Neighbor</i> dan <i>Naïve Bayes</i>	Implementasi dari klasifikasi penerimaan beasiswa menggunakan <i>K-Nearest Neighbor</i> lebih baik dari <i>Naïve Bayes</i> berdasarkan nilai akurasi, presisi, recall, sensitivitas dan spesifitas.
4.	Penentuan Penerimaan Beasiswa		<i>Modified K-Nearest</i>	Pada hasil pengujian, akurasi terkecil pada data latih berjumlah

Menggunakan Metode Modified K-Nearest Neighbor	<i>Neighbor</i> (MKNN)	20 yaitu sebesar 84.8%, data latih berjumlah 30 yaitu sebesar 86.2% dan yang tertinggi data latih berjumlah 40 yaitu sebesar 87.2%.
--	------------------------	---

2.2 Dasar Teori

2.2.1 Data Mining

Secara garis besar data mining digunakan untuk membantu mendapatkan informasi yang berguna. Data mining dikelompokkan menjadi 2 kategori utama, yaitu:

1. Descriptive mining

Descriptive mining adalah proses untuk menemukan karakteristik penting data dalam suatu basis data. Teknik data mining yang termasuk dalam *descriptive mining* adalah *sequential mining*, *clustering* dan *association*.

2. Predictive mining

Predictive mining adalah proses menemukan pola dari data dengan menggunakan beberapa variabel untuk memprediksi variabel lain yang jenis atau nilainya tidak diketahui. Teknik data mining yang termasuk dalam *predictive mining* adalah klasifikasi.

2.2.2 Algoritme K-Nearest Neighbor

Prinsip yang digunakan pada algoritme *K-Nearest Neighbor* adalah membandingkan data uji yang diberikan dengan data latih untuk mencari jarak terdekat antara dua data yang dievaluasi dengan *K* dari tetangga terdekat pada data latih. Dekat atau jauhnya tetangga bisa dihitung berdasarkan variabel. Perhitungan jarak yang umum digunakan adalah *Euclidean Distance*. *Euclidean Distance* dapat dihitung dengan Persamaan (2.1).

$$D(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.1)$$

Dimana:

$D(x, y)$ = Jarak Euclidean antara titik data latih x dengan data uji y

x_i = Sampel data latih

y_i = Data uji

n = Dimensi atribut

Dimana x sebagai variabel data latih dan y sebagai variabel data uji. Kemudian data kontinu harus dinormalisasi atau standardisasi sebelum melakukan klasifikasi. Normalisasi sendiri bertujuan untuk mencegah atribut yang memiliki rentang terlalu besar dengan atribut yang bernilai kecil. Perhitungan min – max

normalisasi dapat digunakan untuk mengubah nilai atribut dapat jatuh pada range tertentu. Min – max dapat dihitung dengan Persamaan (2.2).

$$ndata = \frac{(v-min)*(nmax-nmin)}{max-min} + nmin \quad (2.2)$$

Dimana:

Ndata : data hasil normalisasi

V : data yang akan dinormalisasi

Min : nilai minimum dari data

Max : nilai maksimum dari data

Nmax : skala maksimum yang kita berikan

Nmin : skala minimum yang kita berikan

Tahapan yang digunakan untuk menghitung metode algoritme *K-Nearest Neighbor* (KNN) diawali dengan menentukan parameter *K* (Jumlah tetangga paling dekat). Kemudian menghitung jarak Euclidean masing – masing data latih yang telah diberikan. Hasil dari perhitungan jarak Euclidean diurutkan dari jarak terkecil menuju jarak terbesar dan menentukan nilai *K* yang digunakan.

2.2.3 Weighted Product

Berdasarkan pernyataan Yoon dalam penelitian tahun 1989, metode *Weighted Product* menerapkan sistem perkalian untuk menghubungkan peringkat atribut, peringkat masing – masing atribut dipangkatkan terlebih dahulu dengan bobot atribut yang bersangkutan. Seperti yang ditunjukkan pada persamaan 2.3 (Rudiarsih, 2012).

$$S_i = \prod_{j=1}^n x_{ij}^{w_j} \quad (2.3)$$

Keterangan :

S = Preferensi alternative dituliskan sebagai vektor S dengan $i=1,2,\dots,m$; dimana $\sum w_j = 1$.

w_j merupakan pangkat yang bernilai positif jika termasuk atribut yang menguntungkan, sedangkan untuk atribut yang bersifat merugikan atau merupakan biaya pangkatnya bernilai negative. Preferensi relative untuk setiap alternative ditunjukkan pada Persamaan 2.4 (Rudiarsih, 2012).

$$V_i = \frac{\prod_{j=1}^n x_{ij}^{w_j}}{\prod_{j=1}^n (x_j^*)^{w_j}} \quad (2.4)$$

Keterangan :

V = Preferensi alternative dianalogikan sebagai vektor V

x = Nilai Kriteria

w = Bobot Kriteria / Sub Kriteria

i = Alternatif

j = Kriteria

n = Banyaknya kriteria

* = Banyaknya kriteria yang telah dinilai pada vektor S

2.2.4 Akurasi

Akurasi adalah nilai yang menunjukkan tingkat kesamaan antara data pengukuran sistem dengan data sebenarnya. Penghitungan tingkat akurasi dapat dilakukan pada Persamaan 2.10

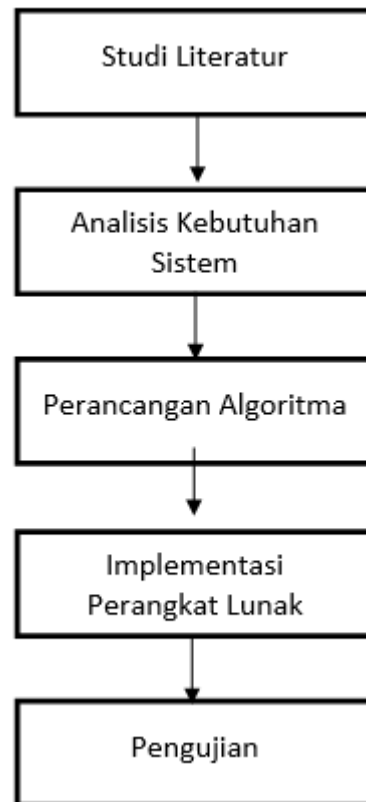
$$Akurasi = \frac{\sum \text{data uji benar}}{\sum \text{jumlah total data uji}} \times 100\% \quad (2.5)$$

Data uji yang benar adalah penjumlahan dari hasil pengukuran sistem yang sama dengan hasil sebenarnya. Sedangkan jumlah total data uji adalah jumlah total data uji yang diprediksi kelasnya.



BAB 3 METODOLOGI

Pada bab ini berisikan tahapan dalam penyelesaian penentuan penerima beasiswa menggunakan metode *K-Nearest Neighbor* dan *weighted product*. Untuk dapat mempermudah dalam penjelasan metodologi yang kami gunakan, maka penulisan menggunakan diagram alir seperti pada Gambar 3.1



Gambar 3.1 Diagram Alir Metode Penelitian

Gambar diatas merupakan tahapan penelitian skripsi yang dijelaskan sebagai berikut:

1. Mengumpulkan beberapa literatur terkait mengenai metode yang akan digunakan selama penelitian skripsi ini berlangsung.
2. Melakukan analisa terhadap kebutuhan sistem.
3. Melakukan perancangan sistem.
4. Mengimplementasikan hasil analisa dan perancangan sistem yang telah dilakukan.
5. Melakukan pengujian terhadap sistem.

3.1 Tipe Penelitian

Penelitian non-implementatif analitik merupakan jenis penelitian yang dipilih. Prosesnya adalah menganalisis ulang penelitian yang sudah pernah dilakukan dengan menggunakan metode yang berbeda. Lalu pada akhirnya dilihat apa saja faktor yang bisa mempengaruhi hasil penelitian.

3.2 Strategi Penelitian

Penelitian dimulai dari studi kepustakaan untuk mencari metode yang tepat dalam memecahkan suatu masalah agar bisa diketemukan solusinya. Referensi yang digunakan harus berkaitan dengan apa yang dibahas pada penelitian ini, yang bersumber dari buku, jurnal, *paper* maupun penelitian terkait. Studi kepustakaan yang diterapkan pada penelitian ini digunakan sebagai acuan dan landasan teori untuk dipelajari serta dianalisis lebih dalam terkait dengan permasalahan yang ada lalu dilanjutkan dengan data siswa MTs Al-Ishlah untuk diproses secara normalisasi data lalu dihitung dengan metode *K-Nearest Neighbor – Weighted Product* serta menghitung evaluasi untuk tahap akhirnya, skenario yang dilakukan ditunjukkan sebagai berikut:

1. Masukan berupa data siswa yang digunakan sebagai data uji. Data dinormalisasi kemudian dihitung menggunakan metode *K-Nearest Neighbor – Weighted Product*.
2. Keluaran berupa hasil klasifikasi apakah apakah siswa tersebut menerima beasiswa atau tidak dan perankingan seluruh penerima beasiswa.

3.3 Teknik Pengumpulan Data

Pengumpulan data yang peneliti lakukan adalah dengan mengambil data siswa di MTs Al-Ishlah tahun ajaran 2018/2019 yang digunakan untuk menentukan penerima beasiswa. Data yang didapatkan berjumlah 221 data, kemudian dibagi menjadi 170 data latih dan 51 data uji.

3.4 Peralatan Pendukung Penelitian

Pada penelitian “Penentuan Penerima Beasiswa Menggunakan Metode *K-Nearest Neighbor* dan *Weighted Product* (Studi Kasus : Mts Al-Ishlah Tembokrejo Muncar)” agar dapat diimplementasikan maka diperlukan analisis kebutuhan sistem. Sistem membutuhkan spesifikasi perangkat keras dan perangkat lunak, yaitu :

1. Spesifikasi kebutuhan hardware, meliputi:
 - a. Komputer atau laptop dengan processor Intel® Core™ i5-7200U CPU @2.70GHz.
 - b. RAM dengan kapasitas 8 GB.
 - c. Hardisk Drive dengan kapasitas 500 GB.
2. Spesifikasi kebutuhan software, meliputi:
 - a. Windows 10 64Bit sebagai sistem operasi

- b. Visual Studio Community sebagai development tools.
- c. Microsoft Excel sebagai penyimpanan data

3.5 Implementasi Algoritme

Implementasi pada sistem menggunakan bahasa pemrograman C#, dari pembuatan proses masukkan data hingga hasil keluaran berupa hasil klasifikasi penerima beasiswa dan perankingan penerima beasiswa. Tahapannya dimulai dari implementasi pada normalisasi hingga penerapan metode *K-Nearest Neighbor – Weighted Product* yang menggunakan bahasa pemrograman C#.



BAB 4 PERANCANGAN

Pada bab perancangan meliputi deskripsi umum sistem, deskripsi data, perancangan sistem, perhitungan manual, perancangan antarmuka. Pada Gambar 4.1 ditunjukkan alur proses yang akan dilakukan pada bab ini.



Gambar 4.1 Diagram Alir Perancangan Sistem

4.1 Deskripsi Umum Sistem

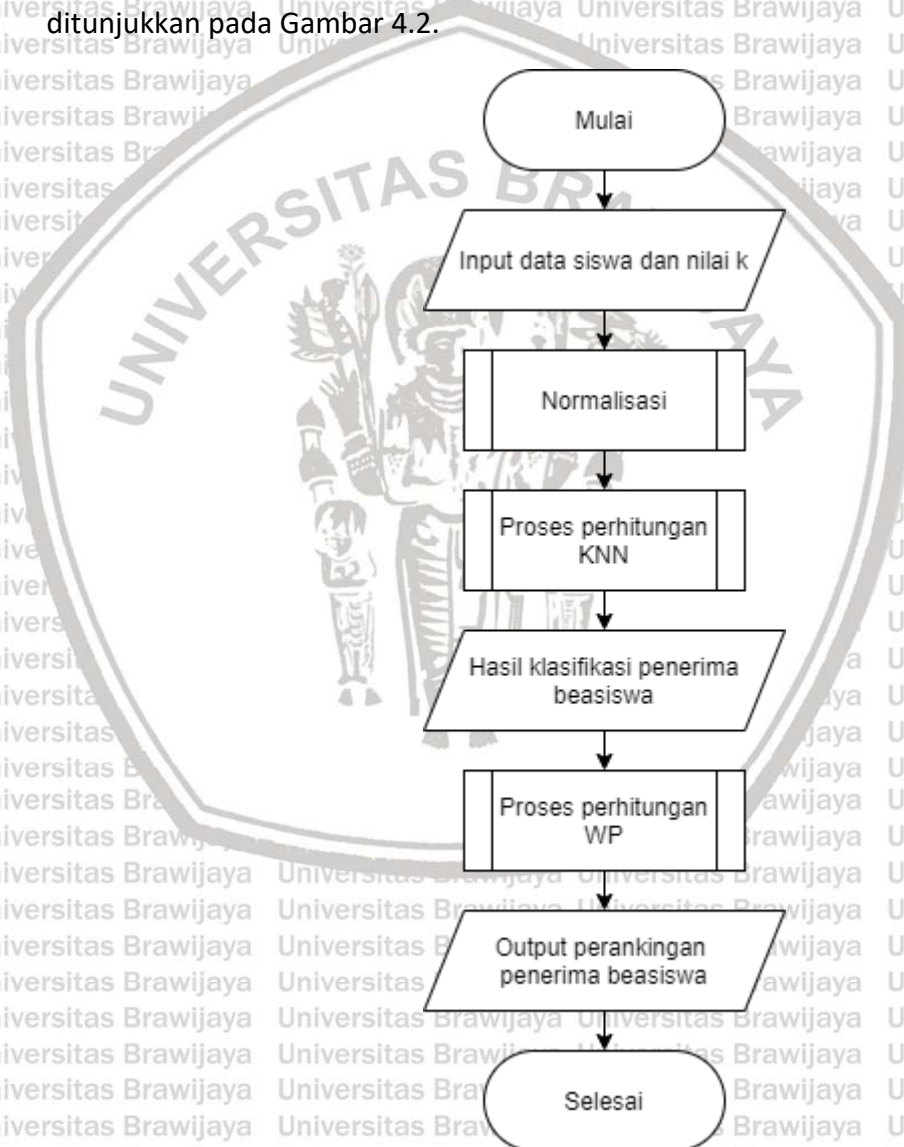
Pada sistem yang akan dikembangkan ini adalah sebuah sistem untuk menentukan penerima beasiswa di MTs. Al-Ishlah, muncar. Metode yang digunakan pada sistem ini adalah metode *K-nearest neighbor-weighted product* menggunakan dataset data komposisi kimiawi pada susu dengan lima parameter yaitu kelas, gaji orang tua, nilai rapor, jumlah saudara, dan tanggungan SPP.

Proses pertama dari sistem, sistem melakukan pembacaan data yang telah disimpan dalam file *excel* dengan *extension* *.xls* yang telah dibagi menjadi dua lembar *worksheet* yaitu data latih dan data uji untuk disimpan sementara pada memori sistem sebelum digunakan untuk proses selanjutnya. Proses kedua dari sistem, sistem akan melakukan normalisasi data dari data yang telah dimasukkan pada proses pertama. Kemudian sistem akan melakukan perhitungan *K-Nearest Neighbor* yang dimulai dengan menghitung jarak *Ecludian* antara data latih dengan data uji. Selanjutnya hasil dari perhitungan jarak *Ecludian* antara data latih terhadap data uji diurutkan dari yang terkecil dan menentukan kelompok data hasil uji berdasarkan label mayoritas dari *K* tetangga terdekat. Proses ketiga yaitu sistem melakukan perhitungan *Weighted Product* dengan menggunakan hasil dari perhitungan *K-Nearest Neighbor* yang dimulai dengan penentuan bobot setiap parameter kemudian dilakukan perhitungan vektor *S*, lalu dilanjutkan dengan

perhitungan vektor V . Proses keempat, data hasil perhitungan vektor V dilakukan perankingan berdasarkan hasil perhitungan yang paling besar. Proses kelima, melakukan perhitungan tingkat akurasi pada sistem yang dikembangkan.

4.2 Penyelesaian Masalah Menggunakan *K-Nearest Neighbor-WP*

Proses awal sistem dalam menentukan penerima beasiswa dengan memasukkan data berupa data nilai dari parameter siswa, dilanjutkan dengan normalisasi data agar range pada data tidak terlalu jauh. Sistem akan memulai proses klasifikasi KNN dengan menghitung jarak *Euclidean*. Hasil dari perhitungan jarak *Euclidean* akan diklasifikasikan berdasarkan label mayoritas dari K tetangga terdekat. Proses yang terakhir adalah melakukan perhitungan *Weighted Product*. Hasil keluaran berupa ranking penerima beasiswa. Diagram alir proses sistem ditunjukkan pada Gambar 4.2.



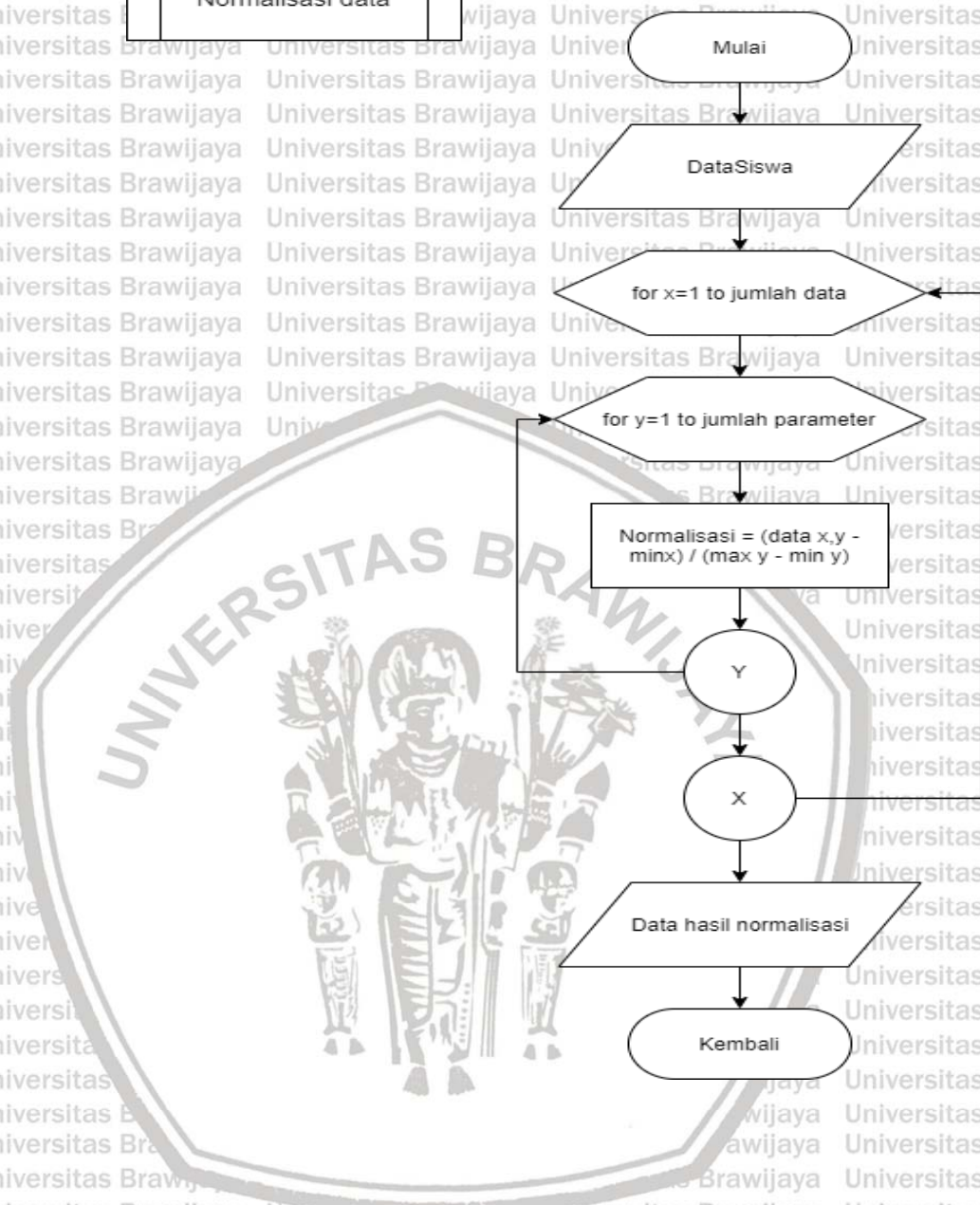
**Gambar 4.2 Diagram Alir Penyelesaian Masalah Menggunakan
*K-Nearest Neighbor – Weighted Product***

Dari Gambar 4.2 dapat dijelaskan bahwa data yang diinputkan adalah data siswa yang terdapat di file data *excle*. Algoritme *K-Nearest Neighbor* berfungsi untuk mendapatkan hasil klasifikasi calon penerima beasiswa, sistem akan menampilkan *output* berupa hasil klasifikasi calon penerima beasiswa. Setelah memiliki hasil diterima maka sistem akan mengolah data uji dengan menggunakan algoritme *Weighted Product* untuk mendapatkan rekomendasi calon penerima beasiswa. Sistem akan menampilkan *output* berupa prioritas penerima beasiswa dari data uji yang telah diinputkan berdasarkan peringkat tertinggi hingga terendah.

4.2.1 Normalisasi Data

Data yang digunakan dalam proses klasifikasi menggunakan metode *K-Nearest Neighbor-WP* harus dinormalisasi terlebih dahulu yang bertujuan untuk range data berada pada range $[0,1]$ sehingga data satu dengan data yang lain tidak terlalu jauh dalam range. Proses perhitungan normalisasi dimulai saat user memasukkan nilai parameter siswa yang kemudian oleh sistem akan dilakukan perulangan dengan sejumlah data siswa yang telah dimasukkan sebelumnya, kemudian dilakukan perhitungan normalisasi. Diagram alir normalisasi data ditunjukkan pada Gambar 4.3.

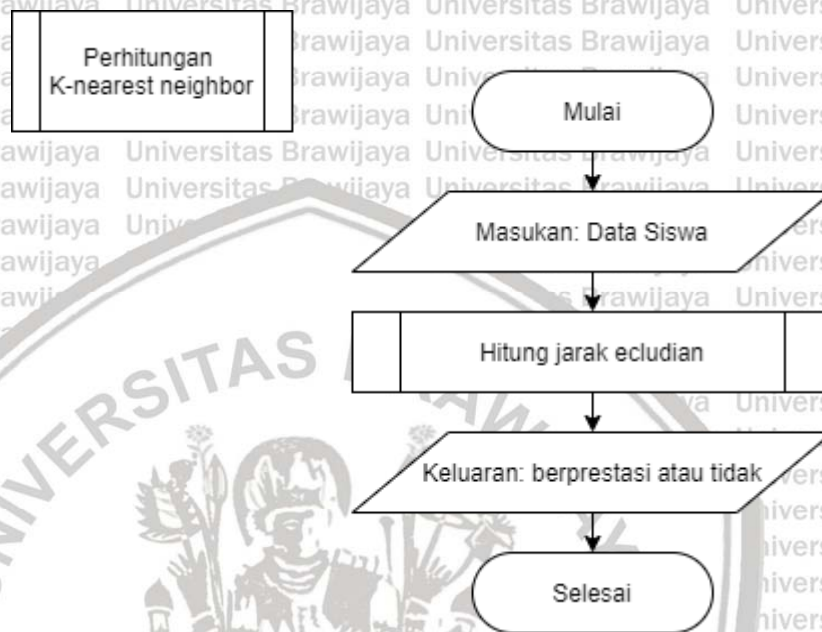
Normalisasi data



Gambar 4.3 Diagram Alir Normalisasi Data

4.2.2 Penyelesaian dengan *K-Nearest Neighbor*

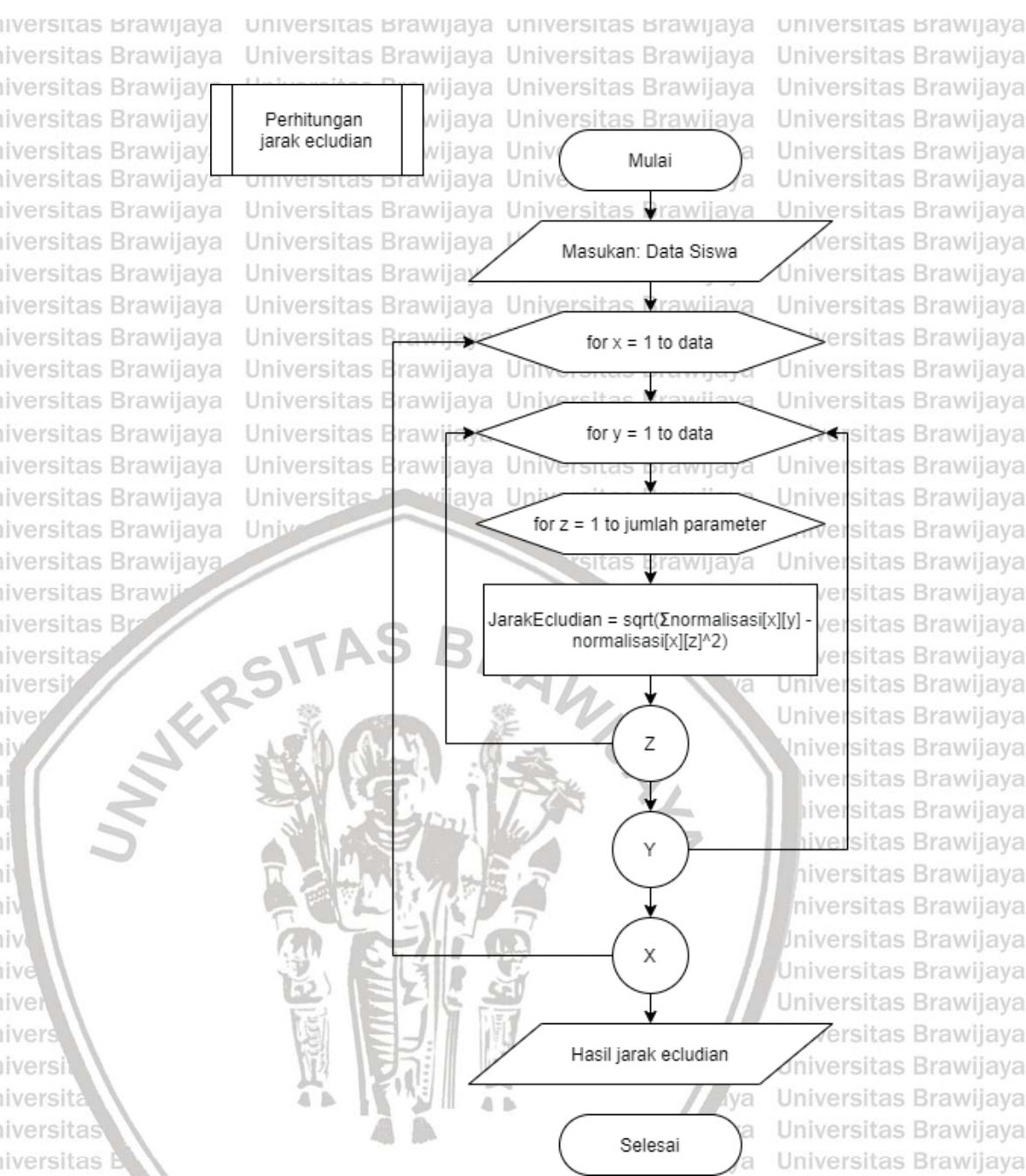
Pada proses klasifikasi *K-Nearest Neighbor* ini berfungsi sebagai pendukung pengambilan keputusan yang berdasarkan dari hasil yang menentukan penerima beasiswa. Proses perhitungan yang dilakukan dalam *K-Nearest Neighbor* meliputi perhitungan jarak *Euclidean* dan menentukan kelompok data hasil uji berdasarkan *K* tetangga terdekat. Diagram alir proses klasifikasi KNN ditunjukkan pada Gambar 4.4.



Gambar 4.4 Diagram Alir Penyelesaian Dengan *K-Nearest Neighbor*

4.2.2.1 Perhitungan Jarak *Euclidian*

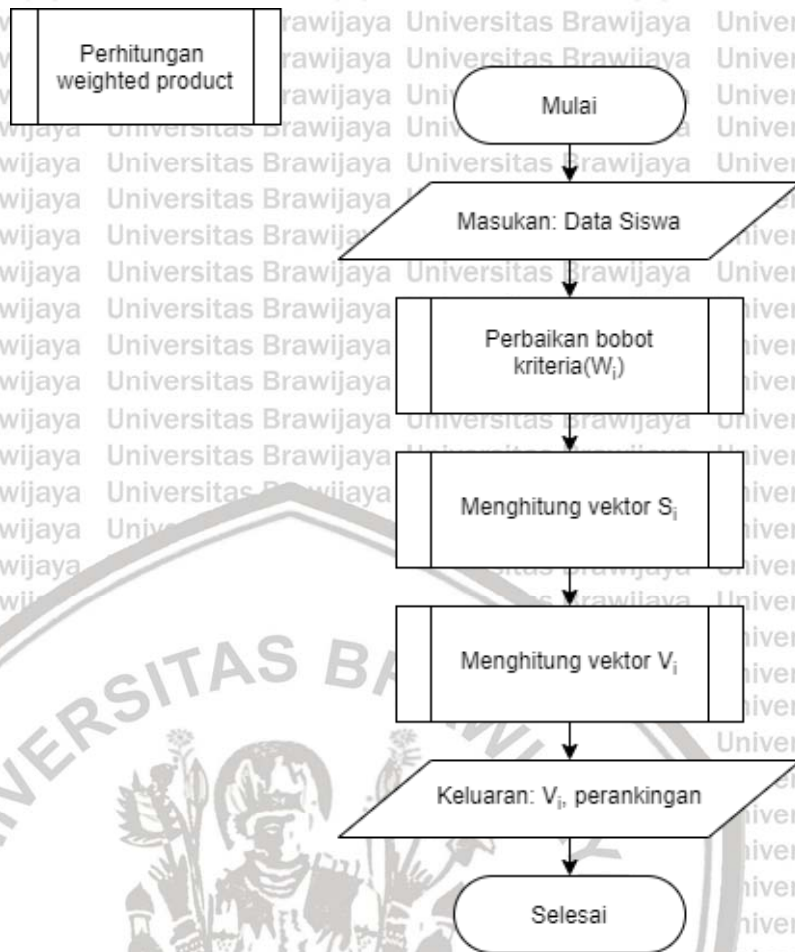
Pada perhitungan jarak *Euclidean*, pertama dimulai dengan memasukkan dataset nilai parameter setiap komposisi yang nantinya akan dilanjutkan dengan menggunakan proses inisialisasi awal. Selanjutnya akan digunakan Persamaan (2.1) yang berfungsi untuk melakukan perhitungan jarak *Euclidean* dan menghasilkan nilai jarak *Euclidean*. Diagram alir proses perhitungan jarak *Euclidean* ditunjukkan pada Gambar 4.5.



Gambar 4.5 Diagram Alir Perhitungan Jarak Ecludian

4.2.3 Penyelesaian dengan *Weighted Product (WP)*

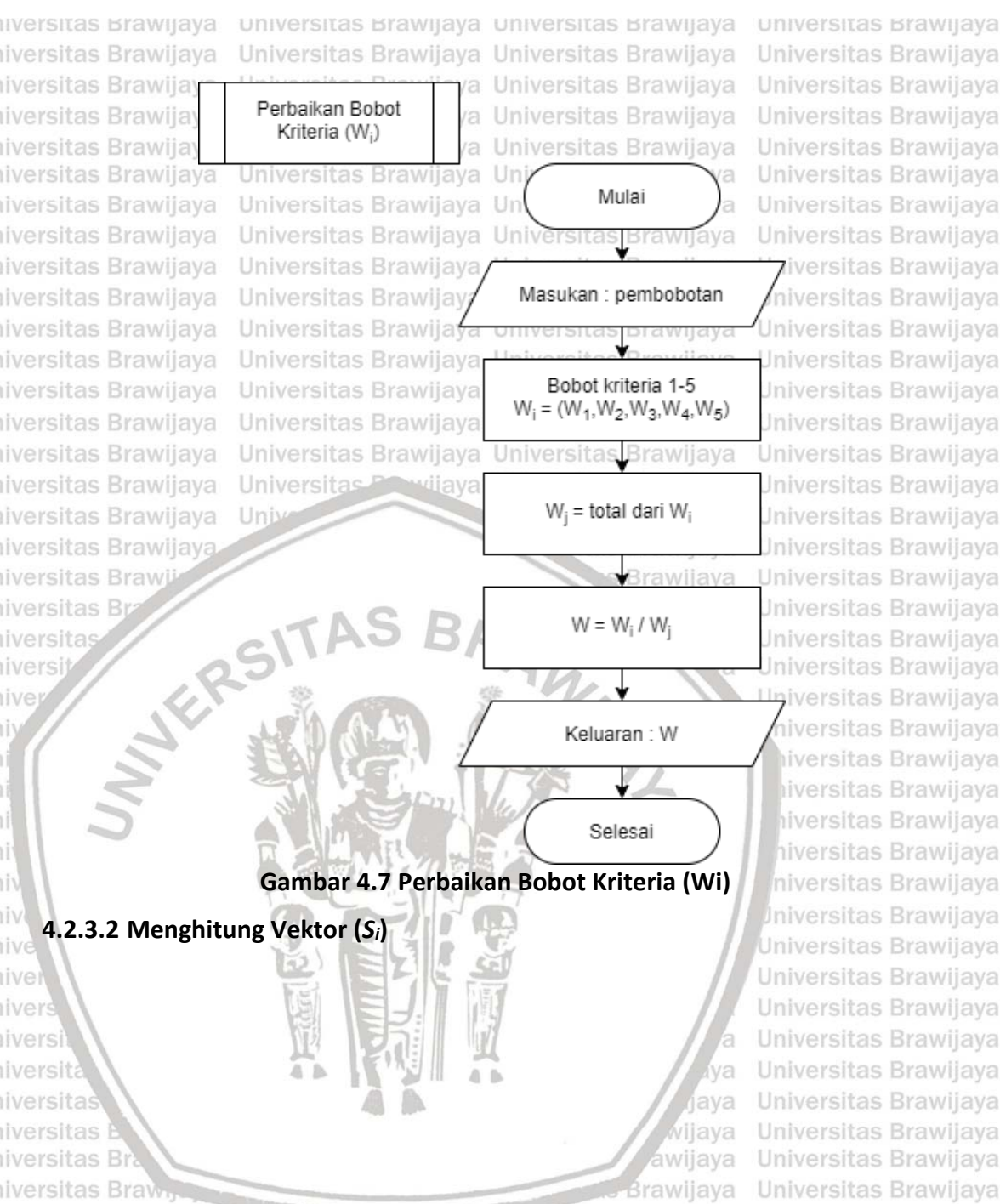
Pada bagian ini membahas tentang perancangan penelitian ini untuk memperoleh rekomendasi calon penerima beasiswa menggunakan algoritme WP. Langkah pertama dari bagian ini adalah melakukan perbaikan bobot awal dari setiap parameter yang digunakan untuk menghitung nilai Vektor S. langkah selanjutnya menghitung nilai vektor S untuk digunakan dalam perhitungan Vektor V. kemudian dilakukan perhitungan vektor V yang hasilnya nanti akan digunakan untuk perankingan calon penerima beasiswa. Alur proses algoritme WP dapat dilihat pada Gambar 4.6.



Gambar 4.6 Penyelesaian dengan *Weighted Product*

4.2.3.1 Perbaikan Bobot Kriteria (W_i)

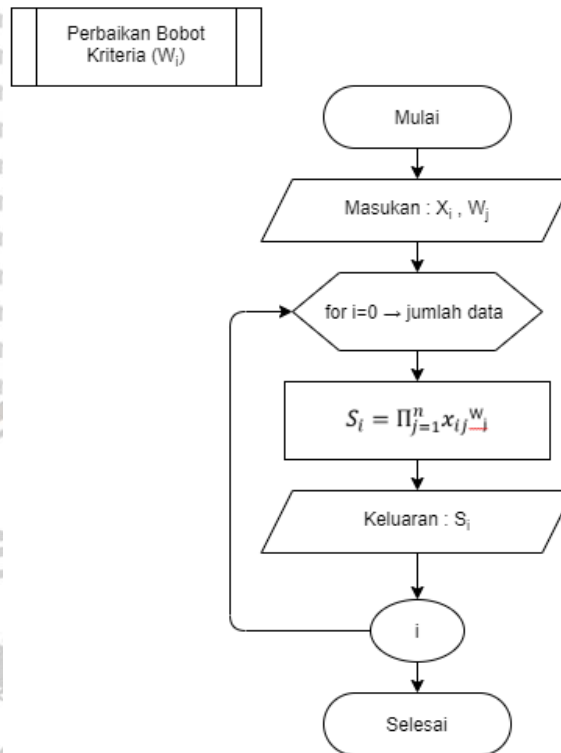
Untuk perbaikan bobot kriteria dengan cara bobot setiap parameter di bagi dengan seluruh jumlah nilai bobot pada setiap parameter. Agar lebih memudahkan pemahaman dapat dilihat pada Gambar 4.7.



Gambar 4.7 Perbaikan Bobot Kriteria (Wi)

4.2.3.2 Menghitung Vektor (S_i)

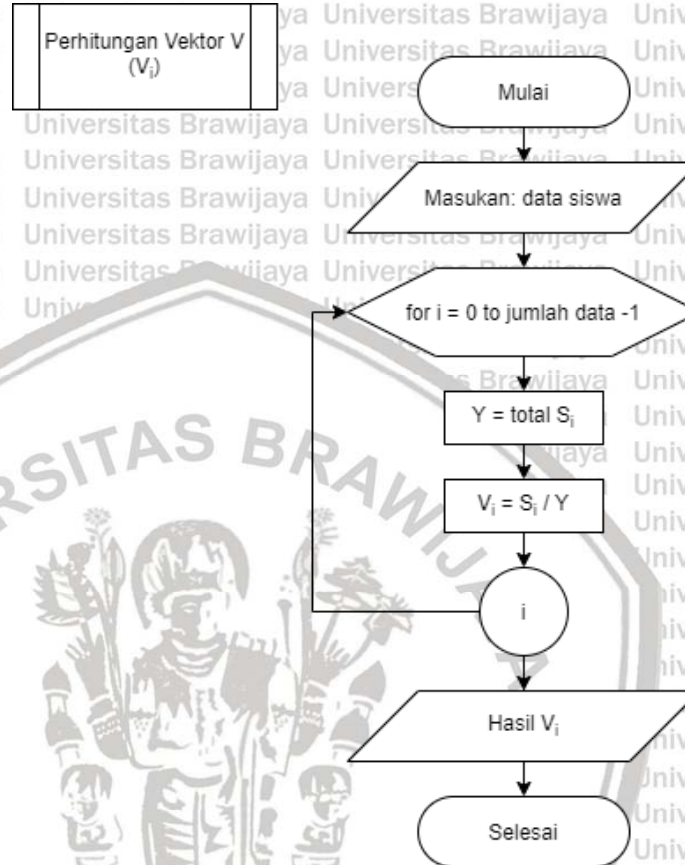
Untuk menghitung vektor (S_i) dengan cara mengalikan setiap nilai parameter pada data siswa yang sudah dipangkatkan dengan nilai perbaikan bobot. Agar lebih memudahkan pemahaman dapat dilihat pada Gambar 4.8.



Gambar 4.8 Menghitung Vektor (S_i)

4.2.3.3 Menghitung Vektor (V_i)

Untuk menghitung vektor (V_i) dengan cara nilai vektor (S_i) yang sebelumnya sudah di cari pada tiap siswa dibagi dengan jumlah seluruh nilai vektor (S). Agar lebih memudahkan pemahaman dapat dilihat pada Gambar 4.9.



Gambar 4.9 Menghitung Vektor (V_i)

4.3 Manualisasi Menggunakan Metode *K-Nearest Neighbor-WP*

Pada penelitian ini, metode yang digunakan dalam menentukan calon penerima beasiswa adalah menggunakan metode *K-Nearest Neighbor-WP*. Berikut merupakan tahapan proses yang akan dilakukan pada perhitungan manualisasi.

1. Melakukan normalisasi data, untuk *range* pada setiap data tidak terlalu jauh yaitu nilai sebesar antara 0 sampai 1.
2. Menentukan nilai dari K , yang berfungsi untuk mengetahui manakah tetangga terdekat dari setiap data. Pada perhitungan manualisasi ini, nilai K adalah $K = 3$.
3. Melakukan perhitungan jarak *Euclidean* antara data latih terhadap data uji
4. Menentukan kelompok data hasil perhitungan jarak *Ecludian* berdasarkan label mayoritas dari K tetangga terdekat.

5. Melakukan perhitungan *Weighted Product* dimulai dengan menentukan bobot setiap parameter.
6. Melakukan perbaikan bobot dengan cara setiap bobot dibagi dengan jumlah seluruh bobot.
7. Melakukan perhitungan vektor *S*.
8. Melakukan perhitungan vektor *V*.
9. Melakukan perankingan dari seluruh hasil dari perhitungan vektor *V*.

Data yang digunakan untuk perhitungan manualisasi pada penelitian ini adalah data nilai parameter dari data siswa yang terdiri dari 20 data dataset seperti pada Tabel, dari dataset tersebut akan dipisahkan menjadi 15 data latihan dan 5 data uji. Untuk lebih jelasnya, data latihan terdapat pada Tabel 4.1 dan untuk data uji terdapat pada Tabel 4.2.

Tabel 4.1 Data Latihan

No	Nama	Kelas	Gaji orang tua	Jumlah Saudara	Tanggungans SPP	Nilai	Klasifikasi
1	Elisa Mutiara Dewi	1	Rp 800,000.00	2	220000	1116.50	Tidak Beasiswa
2	Fatwahul Rohmah	1	Rp 300,000.00	4	1396000	1209.00	Beasiswa
3	Fitriya	1	Rp 800,000.00	2	400000	1230	Tidak Beasiswa
4	Moh. Firman Syabana	1	Rp 250,000.00	3	996000	1227.5	Beasiswa
5	Moh. Riski	1	Rp 1,300,000.00	1	0	1254	Tidak Beasiswa
6	M Reza Febriyadi	2	Rp 900,000.00	3	500000	1318	Tidak Beasiswa
7	M Rohman	2	Rp 500,000.00	2	1931000	1219	Beasiswa
8	Nanik Sri Wahyuni	2	Rp 600,000.00	1	0	1348.5	Tidak Beasiswa
9	Wahyu Cucu Pamungkas	2	Rp 500,000.00	3	1546000	1252.5	Beasiswa
10	Wilda Nunafiah	2	Rp 800,000.00	4	400000	1373	Tidak Beasiswa
11	Aisyah Okta Romadhona	3	Rp 1,000,000.00	2	0	1265	Tidak Beasiswa
12	Alfan tio Saputra	3	Rp 200,000.00	4	3686000	1268	Beasiswa
13	Amin Wafa	3	Rp 500,000.00	2	225000	1232	Tidak Beasiswa
14	M Febriyanto Aditama	3	Rp 750,000.00	1	200000	1103	Tidak Beasiswa

15	M Mukhlisin	3	Rp 350,000.00	5	2738000	1206	Beasiswa
----	-------------	---	---------------	---	---------	------	----------

Tabel 4.2 Data Uji

No	Nama	Kelas	Gaji orang tua	Jumlah Saudara	Tanggungan SPP	Nilai
1	Rimba Intan Laura	1	Rp 300,000.00	2	1396000	1363
2	Aditya	1	Rp 800,000.00	3	180000	1236
3	Della Inka Febriana	2	Rp 300,000.00	5	2151000	1315
4	Dhea Vivi Indriyani	2	Rp 750,000.00	2	0	1313
5	Ulfa Febrianti	3	Rp 500,000.00	5	3446000	1205

4.3.1 Normalisasi

Langkah kedua setelah menentukan nilai K adalah melakukan perhitungan normalisasi data untuk setiap data latih dan data uji yang berfungsi agar nilai *range* dari data tidak terlalu jauh yaitu sebesar antara 0 sampai 1. Pada proses ini menggunakan Persamaan (2.2).

1. Normalisasi data pada tabel data latih kolom pertama baris pertama.

$$V' = \frac{1 - 1}{3 - 1} (1 - 0) + 0 = 0$$

2. Normalisasi data pada tabel data uji kolom pertama baris kelima.

$$V' = \frac{3 - 1}{3 - 1} (1 - 0) + 0 = 1$$

Tabel 4.3 Normalisasi Data Latih

No	Nama	Kelas	Gaji orang tua	Jumlah Saudara	Tanggungan SPP	Nilai
1	Elisa Mutiara Dewi	0	0.55	0.25	0.06	0.05
2	Fatwahul Rohmah	0	0.09	0.75	0.38	0.39
3	Fitriya	0	0.55	0.25	0.11	0.47
4	Moh. Firman Syabana	0	0.05	0.50	0.27	0.46
5	Moh. Riski	0	1.00	0.00	0.00	0.56
6	M Reza Febriyadi	0.5	0.64	0.50	0.14	0.80
7	M Rohman	0.5	0.27	0.25	0.52	0.43
8	Nanik Sri Wahyuni	0.5	0.36	0.00	0.00	0.91
9	Wahyu Cucu Pamungkas	0.5	0.27	0.50	0.42	0.55
10	Wilda Nunafiah	0.5	0.55	0.75	0.11	1.00
11	Aisyah Okta Romadhona	1	0.73	0.25	0.00	0.60
12	Alfan tio Saputra	1	0.00	0.75	1.00	0.61
13	Amin Wafa	1	0.27	0.25	0.06	0.48
14	M Febriyanto Aditama	1	0.50	0.00	0.05	0.00

15	M Mukhlisin	1	0.14	1.00	0.74	0.38
----	-------------	---	------	------	------	------

Tabel 4.4 Normalisasi Data Uji

No	Nama	Kelas	Gaji orang tua	Jumlah Saudara	Tanggung SPP	Nilai
1	Rimba Intan Laura	0.00	0.00	0.00	0.41	1.00
2	Aditya	0.00	1.00	0.33	0.05	0.20
3	Della Inka Febriana	0.50	0.00	1.00	0.62	0.70
4	Dhea Vivi Indriyani	0.50	0.90	0.00	0.00	0.68
5	Ulfa Febrianti	1.00	0.40	1.00	1.00	0.00

4.3.2 Manualisasi *K-Nearest Neighbor*

4.3.2.1 Menentukan nilai *K*

Langkah pertama dari perhitungan manualisasi metode *K-Nearest Neighbor* adalah menentukan nilai *K*, yang berfungsi untuk menentukan manakah tetangga terdekatnya. Pada perhitungan manualisasi ini *K* = 3.

4.3.2.2 Perhitungan jarak *Euclidean* antara data latih dan data uji

Perhitungan jarak *Euclidean* data latih dengan data uji bertujuan untuk mendapatkan nilai jarak *Euclidean* data uji dengan data latih, perhitungannya dilakukan menggunakan Persamaan (2.1), sebagai contoh $d(1,1)$.

$$\begin{aligned}
 d(1,1) &= \sqrt{(0 - 0)^2 + (0 - 0,55)^2 + (0 - 0,25)^2 + (0,41 - 0,06)^2 + (1 - 0,5)^2} \\
 &= 1,382
 \end{aligned}$$

Tabel 4.5 Perhitungan Jarak *Euclidean* Antara Data Latih dan Data Uji

Data	1	2	3	4	5
1	1.382	0.235	1.677	1.603	3.673
2	0.940	1.145	0.723	1.945	1.698
3	0.728	0.292	1.677	0.745	2.600
4	0.561	1.057	0.933	1.603	2.121
5	1.358	0.246	2.908	0.525	3.673
6	1.269	1.027	0.904	0.351	2.187
7	0.976	1.313	0.718	0.795	1.490
8	0.805	1.527	1.567	0.339	3.328
9	1.024	1.319	0.387	0.836	1.410
10	1.448	1.529	0.718	0.800	2.378
11	1.916	1.247	1.990	0.599	2.030
12	2.068	3.244	0.711	2.878	0.596
13	1.528	1.615	1.502	1.002	1.689
14	2.373	1.400	2.560	1.130	1.904

15	2.515	2.702	0.632	2.726	0.281
----	-------	-------	-------	-------	-------

4.3.2.3 Menentukan klasifikasi data uji

Setelah menghitung jarak ecludian antara data latih dengan data uji, maka ditentukan kelompok data sejumlah nilai k berdasarkan label mayoritas dari tertangga terdekat. Dapat dilihat pada Tabel 4.6.

Tabel 4.6 Klasifikasi Data Uji

data	1	Ket.	data	2	Ket.	data	3	Ket.	data	4	Ket.	data	5	Ket.
4	0.56	B	1	0.23	TB	9	0.39	B	8	0.34	TB	15	0.28	B
3	0.73	TB	5	0.25	TB	15	0.63	B	6	0.35	TB	12	0.60	B
8	0.80	TB	3	0.29	TB	12	0.71	B	5	0.53	TB	9	1.41	B
2	0.94		6	1.03		7	0.72		11	0.60		7	1.49	
7	0.98		4	1.06		10	0.72		3	0.75		13	1.69	
9	1.02		2	1.15		2	0.72		7	0.79		2	1.70	
6	1.27		11	1.25		6	0.90		10	0.80		14	1.90	
5	1.36		7	1.31		4	0.93		9	0.84		11	2.03	
1	1.38		9	1.32		13	1.50		13	1.00		4	2.12	
10	1.45		14	1.40		8	1.57		14	1.13		6	2.19	
13	1.53		8	1.53		1	1.68		1	1.60		10	2.38	
11	1.92		10	1.53		3	1.68		4	1.60		3	2.60	
12	2.07		13	1.62		11	1.99		2	1.95		8	3.33	
14	2.37		15	2.70		14	2.56		15	2.73		1	3.67	
15	2.52		12	3.24		5	2.91		12	2.88		5	3.67	

Berdasarkan pengurutan nilai K , hingga didapatkan sesuai Tabel 4.7.

Tabel 4.7 Hasil Klasifikasi Data uji

No	Nama	klasifikasi
1	Rimba Intan Laura	Tidak Menerima Beasiswa
2	Aditya	Tidak Menerima Beasiswa
3	Della Inka Febriana	Menerima Beasiswa
4	Dhea Vivi Indriyani	Tidak Menerima Beasiswa
5	Ulfa Febrianti	Menerima Beasiswa

4.3.3 Manualisasi *Weighted Product*

Tahap ini akan melanjutkan perhitungan dari hasil perhitungan KNN yang telah dilakukan dengan metode *Weighted product*. Untuk menghitung nilai vector S dimulai dari memasukkan nilai parameter siswa masing-masing pada kriteria 1, kriteria 2, kriteria 3, kriteria 4 dan kriteria 5 yang kemudian nilai-nilai tersebut akan dipangkatkan dengan nilai bobot prioritas kriteria lalu didapatkan nilai alternatif dari setiap kriteria. Sehingga dapat dihitung dari data siswa dengan menentukan

Bobot dari masing-masing kriteria. Bobot dari setiap kriteria dapat dilihat pada Tabel 4.8.

Tabel 4.8 Bobot Kriteria

Kriteria	Bobot
Kelas	5
Gaji Orang tua	10
Jumah Saudara	25
Tanggungsan SPP	20
Nilai	40
Jumlah	100

4.3.3.1 Perbaikan Bobot Kriteria (W_i)

Bobot yang sudah ditentukan pada setiap kriteria akan lebih mudah untuk dihitung dalam metode weighted product apabila sudah dilakukan perbaikan bobot. Bobot dari setiap kriteria akan dibagi dengan jumlah total bobot seluruh kriteria, seperti yang ditampilkan pada Tabel 4.9.

Tabel 4.9 Perbaikan Bobot Kriteria

Kriteria	Label	W_i
Kelas	W1	0.05
Gaji Orang tua	W2	0.1
Jumah Saudara	W3	0.25
Tanggungsan SPP	W4	0.2
Nilai	W5	0.4

4.3.3.2 Menghitung Vektor (S_i)

Setelah dilakukannya perbaikan bobot kriteria, langkah selanjutnya adalah mencari vektor S dengan cara melakukan perhitungan sesuai dengan persamaan, seperti pada contoh berikut:

$$Vektor S = (2^{0.05})(300000^{0.1})(5^{0.25})(2151000^{0.2})(1315^{0.4}) = 1784.86$$

Sehingga didapat hasil perhitungan vector S, seperti pada Tabel 4.10.

Tabel 4.10 Perhitungan Vektor (S_i)

No	Nama	Kriteria					Vektor S
		k1	k2	k3	k4	k5	
1	Della Inka Febriana	2	Rp 300,000.00	5	2151000	1315	1784.86
2	Ulfa Febriyanti	3	Rp 500,000.00	5	3446000	1205	2034.01

4.3.3.3 Menghitung Vektor (V_i)

Menghitung nilai Vektor V dimulai dengan mencari jumlah seluruh nilai vektor S lalu didapatkan nilai penjumlahan seluruh vektor S, seperti pada contoh berikut:

$$\sum S = 1784.86 + 2034.01 = 3818.87$$

Kemudian nilai vector V didapatkan sesuai persamaan, nilai vector S dibagi dengan jumlah total nilai vector S . selanjutnya melakukan perankingan nilai vektor V berdasarkan nilai terbesar sampai nilai terkecil. Berikut adalah perhitungan dan hasil perhitungan nilai vektor V :

$$\text{Vektor } V = \frac{1784.86}{3818.87} = 0.000261857$$

Tabel 4.11 Menghitung Vektor (V_i)

No	Nama	Vektor S	Vektor V	ranking
1	Della Inka Febriana	1784.86	0.000261857	2
2	Ulfa Febriyanti	2034.01	0.000523715	1

4.4 Perancangan Antar Muka

Pada perancangan antarmuka sistem ini akan digambarkan secara garis besar bagaimana rancangan tampilan dari sistem yang akan diimplementasikan, antarmuka adalah merupakan alat komunikasi antara pengguna dan sistem penerimaan beasiswa, agar dapat mudah dipahami oleh pengguna maka antarmuka akan didesain sederhana mungkin. Pada perancangan antarmuka di sistem ini akan menampilkan rancangan halaman utama sistem dan halaman hasil penerimaan beasiswa.

4.4.1 Antarmuka halaman utama

halaman utama dari sistem ini adalah halaman pertama yang akan ditampilkan dari sistem penerimaan beasiswa siswa ini. Pada halaman utama ini terdapat dua tombol *choose file* yang berfungsi agar *user* dapat memasukkan data latihan serta data uji dengan format berupa .xls atau .xlsx yang berada pada satu *Excel* namun berbeda *sheet*. Kemudian setelah itu agar *user* dapat memasukkan nilai k dibuat *textbox* sehingga *user* dapat memasukkan nilai k pada *textbox* tersebut. Setelah itu terdapat tombol inisialisasi yang berfungsi untuk menampilkan masukkan data latihan dan data uji yang selanjutnya digunakan untuk melakukan perhitungan, hasil perhitungan menggunakan metode *K-Nearest Neighbor – Weighted Product*. Tampilan antarmuka halaman utama ditunjukkan pada Gambar 4.10.

Gambar 4.10 Rancangan Antarmuka Halaman Utama

4.4.2 Antarmuka Halaman Hasil

Dalam halaman hasil sistem penerimaan beasiswa ini, pengguna bisa melihat keluaran berupa klasifikasi yang diproses oleh sistem dengan menggunakan metode *Modified K-Nearest Neighbor* berupa hasil status penerimaan beasiswa dan hasil akurasi. Terdapat tombol yang berfungsi untuk dapat melihat proses sistem seperti melihat normalisasi seluruh data latih, normalisasi seluruh data uji, hasil perhitungan jarak *Euclidean*, hasil perbaikan bobot tiap parameter, hasil perhitungan vektor *S* dan hasil perhitungan vektor *V* pada tab diantara tab data dan tab hasil. Tampilan antarmuka halaman hasil penerimaan beasiswa ditunjukkan pada Gambar 4.11.

Gambar 4.11 Rancangan Antarmuka Halaman Hasil

4.5 Perancangan Pengujian

Sebagai proses pengujian untuk mengetahui tingkat akurasi dari perbandingan hasil keluaran sistem dengan hasil keluaran sekolah dan hasil

keluaran pakar. Data sampel parameter pengujian beasiswa mahasiswa diambil secara acak.

4.5.1 Rancangan Pengujian Perbandingan Perankingan

Pada bagian sub bab ini akan dilakukan pencarian tingkat akurasi antara hasil keluaran sistem, hasil keluaran dari pakar dan hasil keluaran dari sekolah.

Tabel 4.12 Rancangan Pengujian Perbandingan Perankingan

No	Hasil Perhitungan Sekolah		N1	Hasil Perhitungan Pakar		N2	Hasil Perhitungan Sistem	
	Nama	Keterangan		Nama	Keterangan		Nama	Keterangan
1								
2								
3								
4								

Proses perancangan pengujian selanjutnya dilakukan perubahan nilai bobot untuk mencari nilai akurasi yang terbaik

Tabel 4.13 Perbandingan Hasil Perhitungan Pakar dan Hasil Perhitungan Sistem

Hasil Perhitungan Pakar	Hasil Perhitungan Sistem	N
Nama	Nama	

4.5.2 Rancangan pengujian pengaruh nilai K

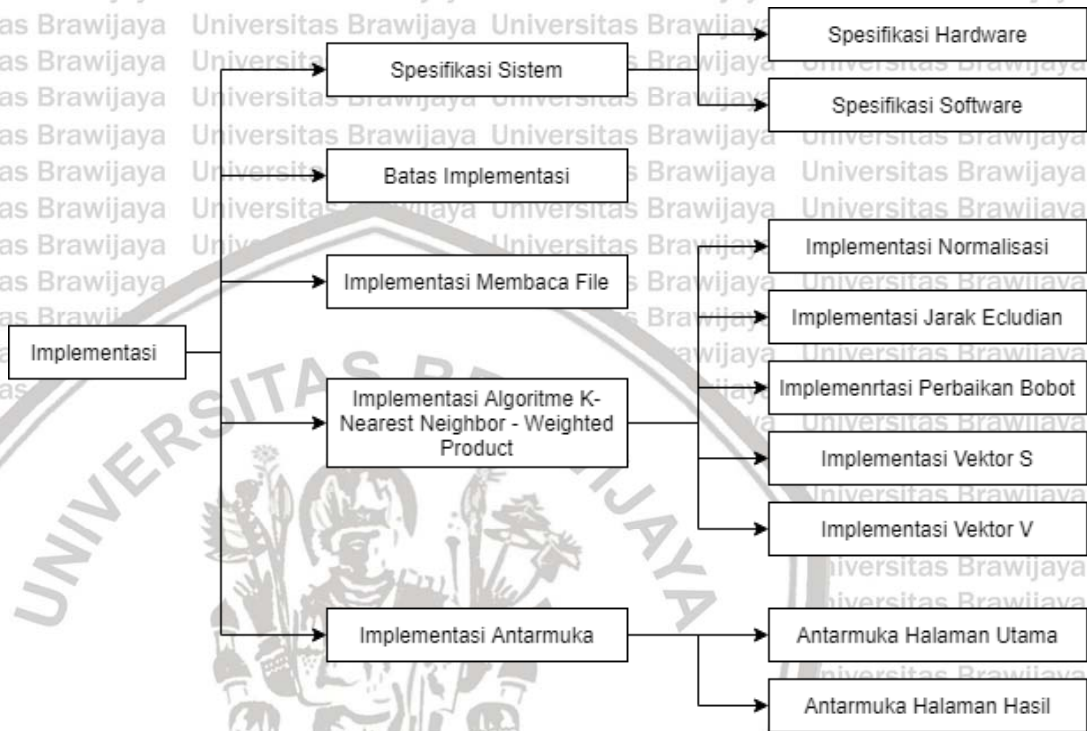
Pada rancangan pengujian pengaruh nilai K berfungsi mengetahui pengaruh nilai k yang berbeda-beda terhadap nilai akurasi dengan menggunakan jumlah data uji yang telah ditentukan, pengujian yang dilakukan. Rancangan ini ditunjukkan pada Tabel 4.14.

Tabel 4.14 Rancangan Pengujian Pengaruh Nilai K

Nilai K	Jumlah Data	Tingkat Akurasi (%)
1		
2		
....		
n		

BAB 5 IMPLEMENTASI

Bab ini akan menjelaskan spesifikasi, batasan implementasi dan hasil dari implementasi sistem itu sendiri. Antarmuka system dalam menentukan beasiswa menggunakan metode *K-Nearest Neighbor – Weighted Product* juga dijelaskan pada bab ini. Tahapan implementasi akan dijelaskan sesuai dalam Gambar 5.1.



Gambar 5.1 Implementasi Sistem

5.1 Spesifikasi Sistem

Spesifikasi hardware dan spesifikasi software berikut akan digunakan dalam menentukan beasiswa menggunakan metode *K-Nearest Neighbor – Weighted Product*.

5.1.1 Spesifikasi Hardware

Dalam Tabel 5.1 dapat diketahui spesifikasi hardware yang akan digunakan dalam sistem untuk menentukan beasiswa dengan metode *K-Nearest Neighbor – Weighted Product*.

Tabel 5.1 Spesifikasi Hardware

Nama Komponen	Spesifikasi
Prosesor	Intel® Core™ i5-7200U CPU @2.70GHz
Memori	8 GB
Hardisk	500 GB

5.1.2 Spesifikasi Software

Tabel 5.2 akan menjelaskan spesifikasi software apa saja yang akan digunakan untuk menentukan beasiswa dengan metode *K-Nearest Neighbor – Weighted Product*.

Tabel 5.2 Spesifikasi Software

Nama	Spesifikasi
Sistem Operasi	Windows 10
Bahasa Pemrograman	C#
Tools Pemrograman	Visual Studio Community

5.2 Batas Implementasi

Dalam mengimplementasikan sistem penentuan beasiswa dengan metode *K-Nearest Neighbor – Weighted Product* terdapat berbagai macam batasan. Batasan implementasi tersebut adalah sebagai berikut:

- A. Sistem penentuan penerima beasiswa menggunakan Bahasa pemrograman C#.
- B. Sistem menggunakan *K-Nearest Neighbor – Weighted Product* dalam mengolah data.
- C. Nilai masukan atau *input* yang digunakan oleh sistem adalah data siswa MTs Al-Ishlah, nilai k dan bobot tiap parameter.
- D. Nilai keluaran atau *output* pada sistem ini berupa perangkingan siswa yang menerima beasiswa berdasarkan perhitungan *Weighted Product*.

5.3 Implementasi Membaca File Inputan

Proses ini adalah hal yang pertama kali dilakukan pengguna dalam menjalankan sistem, yaitu menginputkan data yang telah dikumpulkan dan disimpan dalam bentuk format .xls ataupun .xlsx. Pada tabel *Source Code* 5.1 bisa dilihat implementasi membaca file yang telah dimasukkan pengguna dan akan ditampilkan pada antarmuka halaman awal.

```

1 public DataTable ReadExcel(string fileName, string
2   fileExt, string sheet)
3   {
4       string conn = string.Empty;
5       DataTable dtexcel = new DataTable();
6       if (fileExt.CompareTo(".xls") == 0)
7           conn =
8   @"provider=Microsoft.Jet.OLEDB.4.0;Data Source=" +
9   fileName + ";Extended Properties='Excel
10  8.0;HRD=Yes;IMEX=1';";

```

```

11     else
12         conn =
13         @"Provider=Microsoft.ACE.OLEDB.12.0;Data Source=" +
14         fileName + ";Extended Properties='Excel
15         12.0;HDR=Yes;IMEX=1';";
16         using (OleDbConnection con = new
17         OleDbConnection(conn))
18         {
19             try
20             {
21                 OleDbDataAdapter oleAdpt = new
22                 OleDbDataAdapter("select * from [" + sheet + "$]",
23                 con);
24                 oleAdpt.Fill(dtexcel);
25             }
26             catch (Exception ex)
27             {
28                
29                 MessageBox.Show(ex.Message.ToString());
30             }
31         }
32         return dtexcel;
33     }

```

Source Code 5.1 Membaca File Inputan

5.4 Implementasi Algoritme KNN-WP

5.4.1 Implementasi Normalisasi Data

Proses ini adalah proses normalisasi data yang telah dimasukkan oleh pengguna sebelum diproses dalam perhitungan *K-Nearest Neighbor – Weighted Product*. Dapat dilihat pada tabel *Source Code 5.2*.

```

1     private void normalisasi()
2     {
3
4         dataGridViewDataUji.Sort (dataGridViewDataUji.Columns[
5         0], ListSortDirection.Ascending);
6
7         dataGridViewDataLatih.Sort (dataGridViewDataLatih.Colu
8         mns[0], ListSortDirection.Ascending);
9
10        for (int i=0; i<
11        dataGridViewDataLatih.Columns.Count; i++)
12        {
13            dataGridViewNormalisasiDataLatih.Columns.Add("",
14            dataGridViewDataLatih.Columns[i].HeaderText.ToString(
15            ));
16        }
17
18        dataGridViewNormalisasiDataLatih.Rows.Add();

```



```

19
20     for (int i = 0; i <
21     dataGridViewDataLatih.Rows.Count; i++)
22     {
23         DataGridViewRow row =
24         (DataGridViewRow) dataGridViewNormalisasiDataLatih.Rows
25         [0].Clone();
26         DataGridViewRow rowC =
27         dataGridViewDataLatih.Rows[i];
28         row.Cells[0].Value =
29         Convert.ToInt32(rowC.Cells[0].Value.ToString());
30         for (int j = 1; j <
31         dataGridViewDataLatih.Columns.Count; j++)
32         {
33             if (j == 1)
34             {
35                 row.Cells[j].Value =
36                 rowC.Cells[j].Value.ToString();
37             }
38             else if (j ==
39             dataGridViewDataLatih.Columns.Count-1)
40             {
41                 row.Cells[j].Value =
42                 rowC.Cells[j].Value.ToString();
43             }
44             else
45             {
46                 double max =
47                 dataGridViewDataLatih.Rows.Cast<DataGridViewRow>().Ma
48                 x(r => Convert.ToDouble(r.Cells[j].Value));
49                 double min =
50                 dataGridViewDataLatih.Rows.Cast<DataGridViewRow>().Mi
51                 n(r => Convert.ToDouble(r.Cells[j].Value));
52                 double val =
53                 Math.Round(((Convert.ToDouble(rowC.Cells[j].Value.ToS
54                 tring()) - min) / (max - min)) * 1, 6);
55                 val = Double.IsNaN(val) ||
56                 Double.IsInfinity(val) ? 0 : val;
57                 row.Cells[j].Value = val;
58             }
59         }
60         dataGridViewNormalisasiDataLatih.Rows.Add(row);
61     }
62     dataGridViewNormalisasiDataLatih.Rows.Remove(dataGrid
63     ViewNormalisasiDataLatih.Rows[0]);
64     for (int i = 0; i <
65     dataGridViewDataUji.Columns.Count; i++)
66
67

```

```
68         {
69     69
70     dataGridViewNormalisasiDataUji.Columns.Add("",
71     dataGridViewDataUji.Columns[i].HeaderText.ToString())
72     ;
73     }
74
75     dataGridViewNormalisasiDataUji.Rows.Add();
76
77     for (int i = 0; i <
78     dataGridViewDataUji.Rows.Count; i++)
79     {
80         DataGridViewRow row =
81         (DataGridViewRow) dataGridViewNormalisasiDataUji.Rows[
82         0].Clone();
83         DataGridViewRow rowC =
84         dataGridViewDataUji.Rows[i];
85         row.Cells[0].Value =
86         Convert.ToInt32(rowC.Cells[0].Value.ToString());
87         for (int j = 1; j <
88         dataGridViewDataUji.Columns.Count; j++)
89         {
90             if (j == 1)
91             {
92                 row.Cells[j].Value =
93                 rowC.Cells[j].Value.ToString();
94             }
95             else if (j ==
96             dataGridViewDataLatih.Columns.Count - 1)
97             {
98                 row.Cells[j].Value =
99                 rowC.Cells[j].Value.ToString();
100            }
101            else
102            {
103                double max =
104                dataGridViewDataUji.Rows.Cast<DataGridViewRow>().Max(
105                f => Convert.ToDouble(f.Cells[j].Value));
106                double min =
107                dataGridViewDataUji.Rows.Cast<DataGridViewRow>().Min(
108                f => Convert.ToDouble(f.Cells[j].Value));
109                double val =
110                Math.Round(((Convert.ToDouble(rowC.Cells[j].Value.ToS
111                tring()) - min) / (max - min)) * 1, 6);
112                val = Double.IsNaN(val) ||
113                Double.IsInfinity(val) ? 0 : val;
114                row.Cells[j].Value = val;
115            }
116        }
117    }
```



```

117 dataGridViewNormalisasiDataUji.Rows.Add(row);
118 }
119 }
120 dataGridViewNormalisasiDataUji.Rows.Remove(dataGridVi
121 ewNormalisasiDataUji.Rows[0]);
122 }

```

Source Code 5.2 Normalisasi Data

5.4.2 Implementasi Perhitungan Jarak Ecludian

Proses ini adalah perhitungan jarak ecludian distance antara data uji terhadap data latih. Proses perhitungan jarak *Euclidian* akan dilakukan antara data uji pertama terhadap seluruh data latih dan begitu juga data uji berikutnya. Dapat dilihat pada tabel *Source Code 5.3*.

```

1 public void euclidean()
2 {
3
4 dataGridViewDataUji.Sort(dataGridViewDataUji.Columns[
5 0], ListSortDirection.Ascending);
6
7 dataGridViewDataLatih.Sort(dataGridViewDataLatih.Colu
8 mns[0], ListSortDirection.Ascending);
9
10 dataGridViewNormalisasiDataLatih.Sort(dataGridViewNor
11 malisasiDataLatih.Columns[0],
12 ListSortDirection.Ascending);
13
14 dataGridViewNormalisasiDataUji.Sort(dataGridViewNorma
15 lisasiDataUji.Columns[0],
16 ListSortDirection.Ascending);
17
18 int c = 0;
19
20
21 dataGridViewEuclideanDataLatihUji.Columns.Add("",
22 "No");
23     for (int i = 0; i <
24 dataGridViewDataUji.Rows.Count; i++)
25 {
26
27 dataGridViewEuclideanDataLatihUji.Columns.Add("", "
28 + ++c);
29     }
30
31 dataGridViewEuclideanDataLatihUji.Columns.Add("",
32 "resiko");
33
34 dataGridViewEuclideanDataLatihUji.Rows.Add();
35
36 c = 0;
37

```

```

38         for (int i = 0; i <
39             dataGridViewDataLatih.Rows.Count; i++)
40         {
41             DataGridViewRow rowE =
42                 (DataGridViewRow) dataGridViewEuclideanDataLatihUji.Rows[0].Clone();
43             DataGridViewRow row =
44                 dataGridViewNormalisasiDataLatih.Rows[i];
45             rowE.Cells[0].Value = ++c;
46             for (int j = 0; j <
47                 dataGridViewDataUji.Rows.Count; j++)
48             {
49                 DataGridViewRow rowC =
50                     dataGridViewNormalisasiDataUji.Rows[j];
51                 double finalCount = 0;
52                 for (int m = 2; m <= 6; m++)
53                 {
54                     double times =
55                         Convert.ToDouble(row.Cells[m].Value.ToString()) -
56                         Convert.ToDouble(rowC.Cells[m].Value.ToString());
57                     times = Math.Pow(times, 2);
58                     finalCount += times;
59                 }
60                 finalCount =
61                     Math.Round(finalCount, 6);
62                 rowE.Cells[j + 1].Value =
63                     Math.Round(Math.Sqrt(finalCount), 6);
64             }
65             rowE.Cells[dataGridViewDataUji.Rows.Count + 1].Value
66             = row.Cells[7].Value.ToString();
67             dataGridViewEuclideanDataLatihUji.Rows.Add(rowE);
68         }
69
70         dataGridViewEuclideanDataLatihUji.Columns[dataGridViewDataUji.Rows.Count + 1].Visible = false;
71         dataGridViewEuclideanDataLatihUji.Rows.Remove(dataGridViewEuclideanDataLatihUji.Rows[0]);
72     }
73 }
74
75
76
77
78
79
80

```

Source Code 5.3 Perhitungan Jarak Euclidian

5.4.3 Implementasi Klasifikasi Data Uji Berdasarkan Data Latih

Hasil dari perhitungan jarak euclidian distance antara data uji terhadap data latih akan diklasifikasikan berdasarkan nilai k tetangga terdekat. Dapat dilihat pada tabel *Source Code 5.4*.

```

1 public void hasil()

```



```
2      {
3
4      dataGridViewDataUji.Sort(dataGridViewDataUji.Columns[
5      0], ListSortDirection.Ascending);
6
7      dataGridViewDataLatih.Sort(dataGridViewDataLatih.Colum
8      nns[0], ListSortDirection.Ascending);
9
10     dataGridViewNormalisasiDataLatih.Sort(dataGridViewNor
11     malisasiDataLatih.Columns[0],
12     ListSortDirection.Ascending);
13
14     dataGridViewNormalisasiDataUji.Sort(dataGridViewNorma
15     lisasiDataUji.Columns[0],
16     ListSortDirection.Ascending);
17
18     dataGridViewEuclideanDataLatihUji.Sort(dataGridViewEu
19     clideanDataLatihUji.Columns[0],
20     ListSortDirection.Ascending);
21
22     int c = 0;
23
24
25     dataGridViewHasilKlasifikasi.Columns.Add("", "Data
26     Uji");
27
28     dataGridViewHasilKlasifikasi.Columns.Add("", "Data
29     Asli");
30
31     dataGridViewHasilKlasifikasi.Columns.Add("",
32     "Prediksi Sistem");
33     dataGridViewHasilKlasifikasi.Rows.Add();
34
35     for (int i = 0; i <
36     dataGridViewEuclideanDataLatihUji.Columns.Count - 2;
37     i++)
38     {
39         DataGridViewRow row =
40         (DataGridViewRow) dataGridViewHasilKlasifikasi.Rows[0]
41         .Clone();
42         DataGridViewRow rowN =
43         dataGridViewDataUji.Rows[i];
44
45         double sumB = 0;
46         double sumTB = 0;
47
48
49         dataGridViewEuclideanDataLatihUji.Sort(dataGridViewEu
50         clideanDataLatihUji.Columns[i + 1],
51         ListSortDirection.Ascending);
52
53         row.Cells[0].Value = ++c;
54     }
```

```

55     for (int j = 0; j < k; j++)
56     {
57         string dataUjiResiko =
58         (string)dataGridViewEuclideanDataLatihUji.Rows[j].Cells
59         [dataGridViewNormalisasiDataUji.Rows.Count +
60         1].Value;
61
62         if
63         (dataUjiResiko.Trim().Equals("B"))
64         {
65             sumB++;
66         }
67         else
68         {
69             sumTB++;
70         }
71     }
72
73     string final = sumB > sumTB ? "B" :
74     "TB";
75
76     row.Cells[1].Value =
77     rowN.Cells[dataGridViewDataUji.Columns.Count -
78     1].Value.ToString();
79     row.Cells[2].Value = final;
80
81     dataGridViewHasilKlasifikasi.Rows.Add(row);
82     }
83
84     dataGridViewEuclideanDataLatihUji.Sort(dataGridViewEu
85     clideanDataLatihUji.Columns[0],
86     ListSortDirection.Ascending);
87
88     dataGridViewHasilKlasifikasi.Rows.Remove(dataGridView
89     HasilKlasifikasi.Rows[0]);
90     }

```

Source Code 5.4 Klasifikasi Data Uji Terhadap Data Latih

5.4.4 Implementasi Input Data Bobot dan Perbaikan Bobot

Proses yang dilakukan setelah melakukan klasifikasi adalah perhitungan menggunakan algoritme Weighted Product. Hal yang pertama kali dilakukan adalah input data bobot tiap parameter, selanjutnya nilai bobot akan dilakukan perbaikan bobot. Dapat dilihat pada tabel *Source Code 5.5*.

```

1     public void perbaikan()
2     {
3         double[] nKriteria = {
4         Convert.ToInt32(numericUpDownKelas.Value),
5         Convert.ToInt32(numericUpDownGaji.Value),
6         Convert.ToInt32(numericUpDownSaudara.Value),
7         Convert.ToInt32(numericUpDownSPP.Value),
8         Convert.ToInt32(numericUpDownNilai.Value) };

```



```

9         double sum =
10        Convert.ToInt32(numericUpDownKelas.Value) +
11        Convert.ToInt32(numericUpDownGaji.Value) +
12        Convert.ToInt32(numericUpDownSaudara.Value) +
13        Convert.ToInt32(numericUpDownSPP.Value) +
14        Convert.ToInt32(numericUpDownNilai.Value);
15
16        dataGridViewPerbaikan.Columns.Add("",
17        "Kriteria");
18        dataGridViewPerbaikan.Columns.Add("",
19        "label");
20        dataGridViewPerbaikan.Columns.Add("",
21        "Wi");
22        dataGridViewPerbaikan.Rows.Add();
23
24        for (int i = 1; i <= 5; i++)
25        {
26            DataGridViewRow row =
27            (DataGridViewRow) dataGridViewPerbaikan.Rows[0].Clone(
28            );
29            row.Cells[0].Value =
30            kriteriaSwitch(i);
31            row.Cells[1].Value = "W"+i;
32            row.Cells[2].Value = nKriteria[i-
33            1]/sum;
34            dataGridViewPerbaikan.Rows.Add(row);
35        }
36
37        dataGridViewPerbaikan.Rows.Remove(dataGridViewPerbaik
38        an.Rows[0]);
39    }

```

Source Code 5.5 Input Data Bobot dan Perbaikan Bobot

5.4.5 Implementasi Perhitungan Vektor S

Proses perhitungan vektor S dilakukan setelah perbaikan bobot. Pada proses ini setiap nilai parameter siswa akan dipangkatkan dengan nilai bobot setiap parameter. melakukan klasifikasi perhitungan menggunakan algoritme Weighted Product. Hal yang pertama kali dilakukan adalah input data bobot tiap parameter, selanjutnya nilai bobot akan dilakukan perbaikan bobot. Dapat dilihat pada tabel

Source Code 5.6.

```

1    public void vektorS()
2    {
3        int c = 0;
4
5        dataGridViewVektorS.Columns.Add("",
6        "No");
7        dataGridViewVektorS.Columns.Add("",
8        "Nama");
9        dataGridViewVektorS.Columns.Add("",
10       "Kelas");
11    }

```

```
12         dataGridViewVektorS.Columns.Add("", "Gaji
13     Orang Tua");
14     dataGridViewVektorS.Columns.Add("",
15     "Jumlah Saudara");
16     dataGridViewVektorS.Columns.Add("",
17     "Tanggung an SPP");
18     dataGridViewVektorS.Columns.Add("",
19     "Nilai");
20     dataGridViewVektorS.Columns.Add("",
21     "Vektor S");
22     dataGridViewVektorS.Rows.Add();
23
24     for (int i = 0; i <
25     dataGridViewHasilKlasifikasi.Rows.Count; i++)
26     {
27         DataGridViewRow rowC =
28     dataGridViewHasilKlasifikasi.Rows[i];
29         DataGridViewRow row =
30     (DataGridViewRow) dataGridViewVektorS.Rows[0].Clone();
31         DataGridViewRow rowN =
32     dataGridViewDataUji.Rows[i];
33         double sumVektorS = 1;
34
35         for (int j = 0; j < 5; j++)
36         {
37             DataGridViewRow rowP =
38     dataGridViewPerbaikan.Rows[j];
39
40             sumVektorS *=
41     Math.Pow(Convert.ToDouble(rowN.Cells[j+2].Value),
42     Convert.ToDouble(rowP.Cells[2].Value));
43         }
44
45         row.Cells[0].Value = ++c;
46         row.Cells[1].Value =
47     rowN.Cells[1].Value;
48         row.Cells[2].Value =
49     rowN.Cells[2].Value;
50         row.Cells[3].Value =
51     rowN.Cells[3].Value;
52         row.Cells[4].Value =
53     rowN.Cells[4].Value;
54         row.Cells[5].Value =
55     rowN.Cells[5].Value;
56         row.Cells[6].Value =
57     rowN.Cells[6].Value;
58         row.Cells[7].Value =
59     Math.Round(sumVektorS, 2);
60
61
62     dataGridViewVektorS.Rows.Add(row);
63     }
64 }
```



```

65
66 dataGridViewVektorS.Rows.Remove(dataGridViewVektorS.R
ows[0]);
}

```

Source Code 5.6 Perhitungan Vektor S

5.4.6 Implementasi Perhitungan Vektor V

Proses perhitungan vektor V dimulai dengan menjumlahkan seluruh nilai vektor S yang akan digunakan untuk perhitungan dalam proses ini. Proses selanjutnya adalah membagi nilai vektor S dengan hasil penjumlahan seluruh vektor S , maka nilai vektor V akan didapatkan. Bisa dilihat pada tabel *Source Code 5.7*.

```

1 public void vektorV()
2 {
3     int c = 0;
4
5     dataGridViewVektorV.Columns.Add("",
6     "No");
7     dataGridViewVektorV.Columns.Add("",
8     "Nama");
9     dataGridViewVektorV.Columns.Add("",
10    "Vektor S");
11    dataGridViewVektorV.Columns.Add("",
12    "Vektor V");
13    dataGridViewVektorV.Columns.Add("",
14    "Kelas");
15    dataGridViewVektorV.Columns.Add("", "Gaji
16    Orang Tua");
17    dataGridViewVektorV.Columns.Add("",
18    "Jumlah Saudara");
19    dataGridViewVektorV.Columns.Add("",
20    "Tanggung SPP");
21    dataGridViewVektorV.Columns.Add("",
22    "Nilai");
23    dataGridViewVektorV.Rows.Add();
24
25    double sumVektorS = 0;
26
27    for (int i = 0; i <
28    dataGridViewVektorS.Rows.Count; i++)
29    {
30        sumVektorS +=
31        Convert.ToDouble(dataGridViewVektorS.Rows[0].Cells[7]
32        .Value);
33    }
34
35    for (int i = 0; i <
36    dataGridViewVektorS.Rows.Count; i++)
37    {
38        DataGridViewRow row =
39        (DataGridViewRow) dataGridViewVektorV.Rows[0].Clone();
40

```

```

41 DataGridViewRow rowN =
42 dataGridViewVektorS.Rows[i];
43 row.Cells[0].Value = ++c;
44 row.Cells[1].Value =
45 rowN.Cells[1].Value;
46 row.Cells[2].Value =
47 rowN.Cells[7].Value;
48 row.Cells[3].Value =
49 Convert.ToDouble(rowN.Cells[7].Value) / sumVektorS;
50 row.Cells[4].Value =
51 rowN.Cells[2].Value;
52 row.Cells[5].Value =
53 rowN.Cells[3].Value;
54 row.Cells[6].Value =
55 rowN.Cells[4].Value;
56 row.Cells[7].Value =
57 rowN.Cells[5].Value;
58 row.Cells[8].Value =
59 rowN.Cells[6].Value;
60 dataGridViewVektorV.Rows.Add(row);
61 }
62 dataGridViewVektorV.Columns[4].Visible =
63 false;
64 dataGridViewVektorV.Columns[5].Visible =
65 false;
66 dataGridViewVektorV.Columns[6].Visible =
67 false;
68 dataGridViewVektorV.Columns[7].Visible =
69 false;
70 dataGridViewVektorV.Columns[8].Visible =
71 false;
72
73 dataGridViewVektorV.Rows.Remove(dataGridViewVektorV.R
74 ows[0]);
75 }

```

Source Code 5.7 Perhitungan Vektor V

5.4.7 Implementasi Perankingan Berdasarkan Vektor V

Proses ini akan menampilkan seluruh data siswa calon penerima beasiswa seperti kelas, gaji orang tua, jumlah saudara, tanggungan SPP dan nilai raport. Perankingan dilakukan berdasarkan nilai vektor V yang paling besar. Bisa dilihat pada tabel Source Code 5.8.

```

1 public void vektorRanking()
2 {
3     int c = 0;
4     dataGridViewRanking.Columns.Add("",
5     "Ranking");
6     dataGridViewRanking.Columns.Add("",
7     "Nama");
8
9

```



```

10     dataGridviewRanking.Columns.Add("",
11     "Kelas");
12     dataGridviewRanking.Columns.Add("", "Gaji
13     Orang Tua");
14     dataGridviewRanking.Columns.Add("",
15     "Jumlah Saudara");
16     dataGridviewRanking.Columns.Add("",
17     "Tanggungans PPA");
18     dataGridviewRanking.Columns.Add("",
19     "Nilai");
20     dataGridviewRanking.Rows.Add();
21
22
23     dataGridviewVektorV.Sort(dataGridviewVektorV.Columns[
24     3], ListSortDirection.Descending);
25
26     for (int i = 0; i <
27     dataGridviewVektorV.Rows.Count; i++)
28     {
29         DataGridViewRow row =
30         (DataGridViewRow) dataGridviewRanking.Rows[0].Clone();
31         DataGridViewRow rowN =
32         dataGridviewVektorV.Rows[i];
33         row.Cells[0].Value = ++c;
34         row.Cells[1].Value =
35         rowN.Cells[1].Value;
36         row.Cells[2].Value =
37         rowN.Cells[4].Value;
38         row.Cells[3].Value =
39         rowN.Cells[5].Value;
40         row.Cells[4].Value =
41         rowN.Cells[6].Value;
42         row.Cells[5].Value =
43         rowN.Cells[7].Value;
44         row.Cells[6].Value =
45         rowN.Cells[8].Value;
46
47         dataGridviewRanking.Rows.Add(row);
48     }
49     dataGridviewVektorV.Sort(dataGridviewVektorV.Columns[
50     0], ListSortDirection.Ascending);
51
52     dataGridviewRanking.Rows.Remove(dataGridviewRanking.R
53     ows[0]);

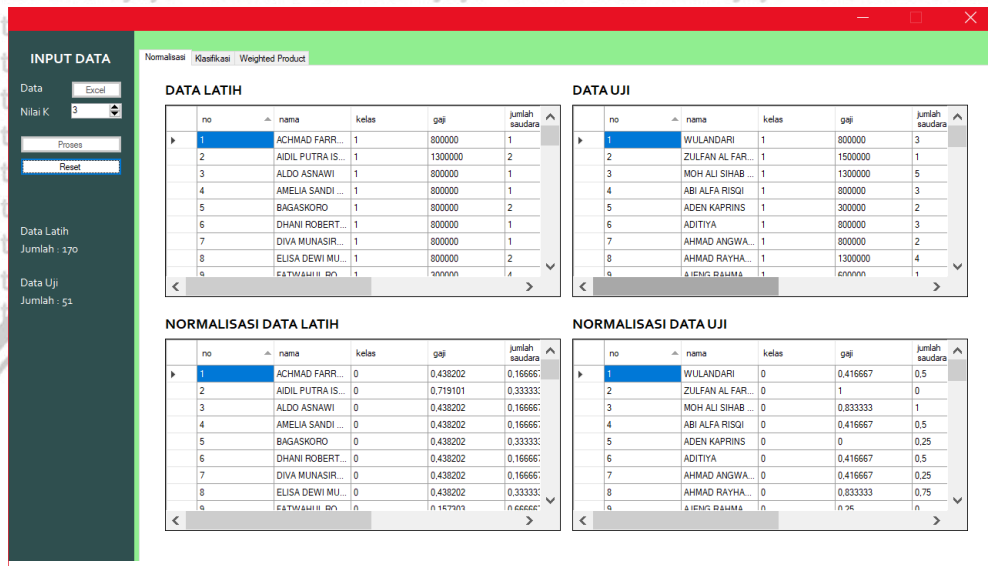
```

Source Code 5.8 Perankingan Berdasarkan Vektor V

5.5 Implementasi Antarmuka

5.5.1 Antarmuka Halaman Utama

Antarmuka halaman utama adalah halaman awal yang ditampilkan ketika pertama kali menjalankan program. Dalam halaman utama terdapat tombol untuk menginputkan data berupa file dalam bentuk excel, kolom untuk menentukan nilai K , jumlah data latih, jumlah data uji dan normalisasi data latih dan data uji. Dalam Gambar 5.2 dapat dilihat implementasi dari antarmuka halaman utama pada sistem ini.



Gambar 5.2 Antarmuka Halaman Utama Sistem

5.5.2 Antarmuka Halaman Hasil

Antarmuka halaman hasil dalam sistem ini terdapat proses perhitungan Weighted Product yang dimana hasil yang didapatkan berupa perankingan siswa – siswa yang menerima beasiswa berdasarkan bobot dari setiap kriteria. Berdasarkan hal tersebut, dalam antarmuka halaman hasil ditampilkan kolom inputan bobot dari setiap kriteria, perbaikan bobot, perhitungan vektor S , perhitungan vektor V dan yang terakhir perankingan. Dapat dilihat pada Gambar 5.3 tampilan antarmuka halaman hasil.

INPUT DATA

Data: Excel
 Nilai K: 3
 Proses
 Reset

Data Latih: Jumlah: 170
 Data Uji: Jumlah: 51

INPUT DATA

Kelas: 10
 Gaji Orang Tua: 20
 Jumlah Saudara: 15
 Tanggungan SPP: 25
 Nilai: 30

PERBAIKAN BOBOT

Kriteria	label
Kelas	W1
Gaji Orang Tua	W2
Jumlah Saudara	W3
Tanggungan SPP	W4
Nilai	W5

VEKTOR (Si)

No	Nama	Kelas	Gaji Orang Tua	Jumlah Saudara	Tanggungan SPP	Nilai
1	ADEN KAPRINS	1	300000	2	370000	1279
2	ULFA FEBRIYAN...	3	500000	5	3446000	1205
3	YUDA PRATAMA	3	500000	4	4141000	1210

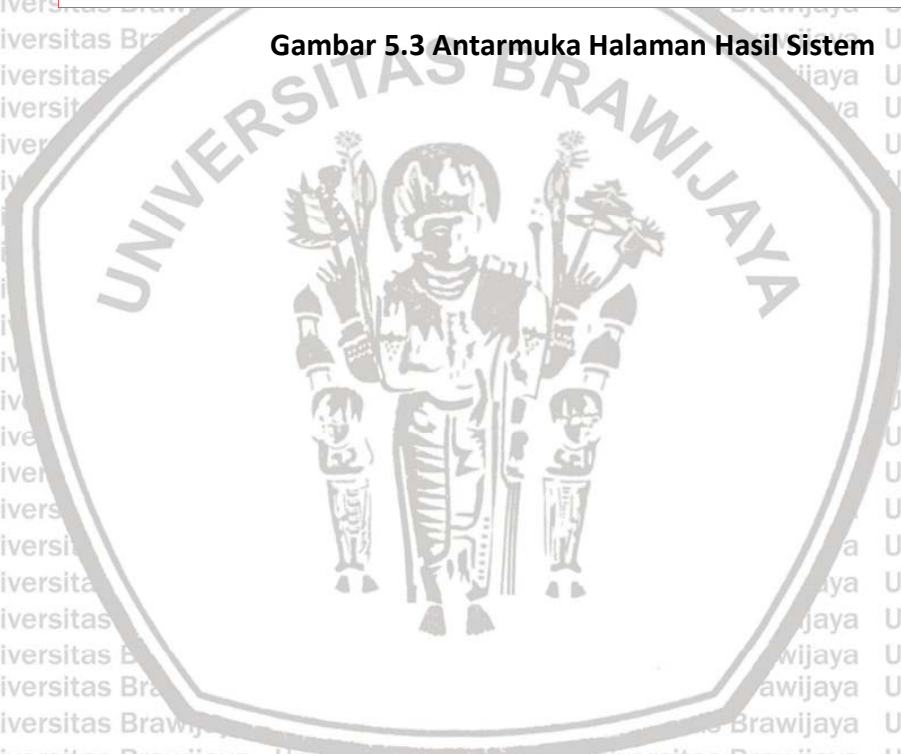
VEKTOR (Vi)

No	Nama	Vektor S	Vektor V
1	ADEN KAPRINS	2915.35	0.333333333333...
2	ULFA FEBRIYAN...	7095.39	0.811267943814...
3	YUDA PRATAMA	7193.28	0.822460424991...

RANKING

Ranking	Nama	Kelas	Gaji Orang Tua	Jumlah Saudara	Tanggungan SPP	Nilai
1	YUDA PRATAMA	3	500000	4	4141000	1210
2	ULFA FEBRIYAN...	3	500000	5	3446000	1205
3	ADEN KAPRINS	1	300000	2	370000	1279

Gambar 5.3 Antarmuka Halaman Hasil Sistem



BAB 6 PENGUJIAN DAN ANALISIS

Bab ini akan menjelaskan proses dan hasil dari pengujian pada sistem berdasarkan implementasi yang dilakukan, beserta analisis dari hasil pengujian yang dilakukan.

6.1 Pengujian Akurasi Algoritme *K-Nearest Neighbor*

Pengujian akurasi dilakukan untuk mengetahui tingkat kecocokan hasil dari sistem yang berupa urutan nama penerima beasiswa berdasarkan nilai rekomendasi tertinggi. Data yang digunakan berjumlah 51 data uji. Hasil rekomendasi yang diperoleh dari perhitungan sistem dicocokkan dengan hasil dari sekolah.

Tujuan dari pengujian akurasi adalah untuk mengetahui tingkat kecocokan data penerima beasiswa antara sistem yang dibuat dengan menggunakan metode *K-Nearest Neighbor* dan *Weighted Product* dengan hasil dari pihak MTs Al-Ishlah.

Prosedur dalam pengujian ini dilakukan dengan membandingkan hasil perhitungan menggunakan *K-Nearest Neighbor* dan *Weighted Product*, dari hasil perhitungan penentuan penerima beasiswa di MTs. Al-Ishlah yang dilakukan pihak sekolah dengan hasil perhitungan sistem.

Pada perhitungan akurasi terdapat 51 data uji dan digunakan masing-masing 10 data siswa dari tiap tingkat. Nilai dapat dikatakan benar (1) apabila data penerima beasiswa dari pihak sekolah sesuai dengan data hasil keluaran sistem. Begitu juga sebaliknya, dapat dikatakan salah (0) apabila data tidak sesuai.

Dalam pengujian ini nama lengkap siswa akan diiniasialisasikan menjadi $S(x)$ di mana S adalah singkatan dari siswa dan x adalah nomor urut siswa pada tabel data. Missal siswa berinisial A dengan nomor urut satu 1 pada tabel siswa, maka pada pengujian ini menjadi S1 begitu juga dengan siswa berinisial G pada nomor urut 7 menjadi S7 pada pengujian ini.

Pada Tabel 6.1 terdapat perbandingan data calon penerima beasiswa yang diperoleh dari pakar dengan pihak sekolah dan perbandingan data calon penerima beasiswa dari pakar dengan hasil perhitungan sistem menggunakan metode *K-Nearest Neighbor*.

Tabel 6.1 Perbandingan Hasil Keluaran Sistem dengan Data Pihak Sekolah dan Data Pihak Pakar

No.	Pihak Pakar		N1	Pihak Sekolah		N2	Hasil Sistem	
	Nama	Keterangan		Nama	Keterangan		Nama	Keterangan
1	S1	TB	1	S1	TB	1	S1	TB
2	S2	TB	1	S2	TB	1	S2	TB
3	S3	B	1	S3	B	1	S3	B
4	S4	TB	1	S4	TB	1	S4	TB
5	S5	TB	1	S5	TB	1	S5	TB

6	S6	TB	1	S6	TB	1	S6	TB
7	S7	B	1	S7	B	1	S7	B
8	S8	TB	1	S8	TB	1	S8	TB
9	S9	TB	1	S9	TB	1	S9	TB
10	S10	TB	1	S10	TB	1	S10	TB
11	S11	TB	1	S11	TB	1	S11	TB
12	S12	TB	1	S12	TB	1	S12	TB
13	S13	TB	1	S13	TB	1	S13	TB
14	S14	TB	1	S14	TB	1	S14	TB
15	S15	B	1	S15	B	1	S15	B
16	S16	TB	1	S16	TB	1	S16	TB
17	S17	TB	1	S17	TB	1	S17	TB
18	S18	TB	1	S18	TB	0	S18	B
19	S19	TB	1	S19	TB	1	S19	TB
20	S20	TB	1	S20	TB	1	S20	TB
21	S21	B	1	S21	B	1	S21	B
22	S22	TB	1	S22	TB	1	S22	TB
23	S23	TB	1	S23	TB	1	S23	TB
24	S24	TB	1	S24	TB	1	S24	TB
25	S25	TB	1	S25	TB	1	S25	TB
26	S26	TB	1	S26	TB	0	S26	B
27	S27	TB	1	S27	TB	1	S27	TB
28	S28	TB	1	S28	TB	1	S28	TB
29	S29	TB	1	S29	TB	1	S29	TB
30	S30	TB	1	S30	TB	1	S30	TB
31	S31	TB	1	S31	TB	1	S31	TB
32	S32	B	1	S32	B	0	S32	TB
33	S33	TB	1	S33	TB	1	S33	TB
34	S34	TB	1	S34	TB	1	S34	TB
35	S35	TB	1	S35	TB	1	S35	TB
36	S36	TB	1	S36	TB	1	S36	TB
37	S37	TB	1	S37	TB	1	S37	TB
38	S38	TB	1	S38	TB	1	S38	TB
39	S39	TB	1	S39	TB	1	S39	TB
40	S40	B	1	S40	B	1	S40	B
41	S41	TB	1	S41	TB	1	S41	TB
42	S42	TB	1	S42	TB	1	S42	TB
43	S43	TB	1	S43	TB	1	S43	TB
44	S44	B	1	S44	B	1	S44	B
45	S45	TB	1	S45	TB	1	S45	TB
46	S46	TB	1	S46	TB	1	S46	TB
47	S47	TB	1	S47	TB	1	S47	TB
48	S48	B	1	S48	B	1	S48	B
49	S49	TB	1	S49	TB	1	S49	TB



50	S50	TB	1	S50	TB	1	S50	TB
51	S51	B	1	S51	B	1	S51	B

Pada tabel 6.1 dapat dilihat terdapat variabel N1 adalah hasil perbandingan calon penerima beasiswa antara data yang diperoleh dari pakar dengan data yang diperoleh dari sekolah. Variabel N2 adalah hasil perbandingan calon penerima beasiswa antara data yang diperoleh dari sekolah dengan data yang didapatkan dari sistem. Variabel TB yang memiliki arti tidak menerima beasiswa dan variabel B yang memiliki arti menerima beasiswa. Tingkat akurasi yang diperoleh dari pengujian tersebut dengan menggunakan persamaan 2.5.

$$\text{Tingkat Akurasi } N2 = \frac{\sum \text{Data Uji Sama}}{\sum \text{Total Data Uji}} \times 100\%$$

$$\text{Tingkat Akurasi } N2 = \frac{48}{51} \times 100\% = 94,1\%$$

Berdasarkan hasil perhitungan tingkat akurasi diatas, nilai tingkat akurasi penentuan penerima beasiswa menggunakan metode K-Nearest Neighbor sebesar 94,1%.

6.2 Pengujian Pengaruh Nilai K

Pengujian ini digunakan untuk mencari seberapa besar pengaruh nilai k terhadap nilai hasil akurasi, untuk nilai k yang akan digunakan dalam penelitian ini adalah sebesar k=1 sampai dengan k=5. Total keseluruhan dari jumlah dataset yang digunakan dalam penelitian ini adalah sebanyak 221 data, dimana berarti pada data uji 25% berisi 51 data uji dan 170 data latih. Pada Tabel 6.2 akan ditunjukkan nilai tingkat akurasi dari pengujian yang telah dilakukan.

Tabel 6.2 Hasil Pengujian Pengaruh Nilai K

K	Data Uji	Tingkat Akurasi
1	25%	92,15%
2	25%	96,07%
3	25%	94,11%
4	25%	96,07%
5	25%	96,07%

6.3 Pengujian Akurasi Algoritme *Weighted Product*

Pengujian tingkat akurasi dilakukan untuk melihat tingkat kecocokan hasil perankingan dari sistem yang mengimplementasikan algoritme *Weighted Product* dengan data yang didapatkan dari pakar. Perbandingan dilakukan dengan menggunakan 10 data siswa yang telah diklasifikasikan mendapatkan beasiswa menggunakan algoritme *K-Nearest Neighbor*. Bisa dilihat pada Tabel 6.3 contoh

perbandingan perankingan antara data yang diperoleh dari pakar dengan data hasil keluaran sistem.

Tabel 6.3 Perbandingan Hasil Keluaran Sistem dengan Data Pakar

Data Pakar	N	Hasil Sistem
Nama		Nama
S48	1	S48
S51	1	S51
S44	1	S44
S26	1	S26
S40	1	S40
S18	1	S18
S21	1	S21
S3	1	S3
S15	1	S15
S7	1	S7

Variabel N diberikan nilai 1 apabila perankingan keluaran sistem sesuai dengan perankingan oleh pakar. Berdasarkan Tabel 6.3, didapatkan perbandingan perankingan penerima beasiswa antara data dari pihak pakar dengan data hasil keluaran sistem, dengan bobot kriteria kelas = 5, gaji orang tua = 35, jumlah saudara = 25, tanggungan SPP = 20 dan nilai rapor = 15. Tingkat akurasi yang diperoleh dari pengujian tersebut dengan menggunakan persamaan 2.5.

$$\text{Tingkat Akurasi } N = \frac{\sum \text{Data Uji Sama}}{\sum \text{Total Data Uji}} \times 100\%$$

$$\text{Tingkat Akurasi } N = \frac{10}{10} \times 100\% = 100\%$$

BAB 7 PENUTUP

Pada bab ini berisikan kesimpulan dari penelitian yang telah dilakukan dan pemberian saran kepada peneliti selanjutnya untuk dikembangkan lebih lanjut.

7.1 Kesimpulan

Berdasarkan hasil dari perancangan, pengujian serta implementasi yang telah dilakukan dapat ditarik kesimpulan sebagai berikut:

1. Penentuan penerima beasiswa dapat diimplementasikan menggunakan Metode *K-Nearest Neighbor – Weighted Product* dengan menggunakan 5 parameter dan 2 kelas klasifikasi.
2. Hasil Pengujian ini mendapatkan akurasi dijelaskan dibawah ini:
 - a. Klasifikasi calon penerima beasiswa dengan menggunakan algoritme *K-Nearest Neighbor*, didapatkan nilai akurasi sebesar 94,1%
 - b. Menurut hasil pengujian nilai *K*, nilai akurasi tertinggi sebesar 96,07% dengan nilai *K*= 2, 4 dan 5 dan yang terendah yaitu menggunakan nilai *K*=1 dengan akurasi 92,15%.
 - c. Untuk perankingan menggunakan algoritme *Weighted Product*, dengan nilai bobot kriteria kelas = 5, gaji orang tua = 35, jumlah saudara = 25, tanggungan SPP = 20 dan nilai rapor = 15, didapatkan nilai akurasi sebesar 100%.

7.2 Saran

Berdasarkan kesimpulan yang telah didapat, ada beberapa saran untuk pembaca jika ingin melakukan penelitian lebih lanjut, sebagai berikut :

1. Pada penelitian ini hanya menggunakan 1 MTs atau 1 tempat saja dan menggunakan data 1 tahun, dikarenakan lingkup penelitian terlalu kecil. Untuk peneliti selanjutnya disarankan mencari data penerima beasiswa yang lebih banyak tidak hanya dalam kurun waktu 1 tahun saja.
2. Dalam penelitian ini tidak memperhitungkan jenis beasiswa yang diberikan oleh pihak penyedia beasiswa, sehingga dalam penelitian berikutnya dapat ditambahkan klasifikasi baru berupa jenis beasiswa yang diberikan.

DAFTAR REFERENSI

Diah, A. k., 2013. Sistem Pendukung Keputusan Rekrutmen Karyawan Produksi Menggunakan Metode Weighted Product Pada PT. Ploss Asia Semarang.

Kartika, J. I., 2017. Penentuan Siswa Berprestasi Menggunakan Metode K-Nearest Neighbor dan Weighted Product (Studi Kasus : SMP Negeri 3 Mejayan).

Lestari, D. R. D., 2017. Perbandingan Klasifikasi Beasiswa Toyota Astra Menggunakan K-Nearest Neighbor Classifier Dan Naïve Bayes Sebagai Penentu Metode Klasifikasi Pada Sistem Pendukung Keputusan Penerimaan Beasiswa Toyota Astra (Studi Kasus: Institut Teknologi Sepuluh Nopember).

M. et al., 2016. Implementasi Metode Naïve Bayes Classifier Untuk Seleksi Asisten Praktikum Pada Simulasi Hadoop Multinode Cluster. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, Volume 3.

Prasetya, D. E., Cholissodin, I. & Widodo, A. W., 2016. Sistem Pengklasifikasi Jurusan Kuliah Untuk Pelajar Sma Menggunakan Algoritma AHP-Naïve Bayes. *Doro Jurnal*, Volume 8.

Rudiarsih, N., 2012. *Sistem Pendukung Keputusan untuk Proses Penentuan Rumah Tangga Miskin Menggunakan Metode Weighted Product*. [Online] Available at: <http://filkom.ub.ac.id/doro/archives/detail/DR00026201306> [Accessed 6 November 2017].

Sudarsana, I. K., 2016. Pemikiran Tokoh Pendidikan Dalam Buku Lifelong Learning: Policies, Practices, And Programs (Perspektif Peningkatan Mutu Pendidikan Di Indonesia). *Jurnal Penjaminan Mutu 1 (1)*, 1-14.

LAMPIRAN

Lampiran 1.1 Data Siswa MTs Al-Ishlah T.A 2018/2019

No	Nama	Kelas	Gaji	Jumlah saudara	Tanggungan spp	Rapor
1	Achmad farrel ramadhan	1	Rp 800.000,00	1	250000	1228
2	Aidil putra islami pasha	1	Rp 1.300.000,00	2	0	1169,50
3	Aldo asnawi	1	Rp 800.000,00	1	270000	1143,00
4	Amelia sandi agustin	1	Rp 800.000,00	1	300000	1212,00
5	Bagaskoro	1	Rp 800.000,00	2	500000	1070,50
6	Dhani robertian	1	Rp 800.000,00	1	200000	1073,50
7	Diva munasiroh	1	Rp 800.000,00	1	150000	1138,00
8	Elisa dewi mutiara	1	Rp 800.000,00	2	220000	1116,50
9	Fatwahul rohmah	1	Rp 300.000,00	4	1396000	1209,00
10	Fitriya	1	Rp 800.000,00	2	400000	1230
11	Habibah	1	Rp 1.300.000,00	1	0	1264
12	Ishomudin noersi zawawi	1	Rp 1.300.000,00	4	150000	1253,5
13	Laura vitsa maharani	1	Rp 1.000.000,00	0	0	1215
14	M. Andriansyah	1	Rp 800.000,00	1	100000	1214
15	M. Refqi pratama	1	Rp 650.000,00	0	0	1266
16	M. Virdaus ramadhan	1	Rp 1.300.000,00	2	200000	1098,00
17	Moh. Farel irfansah	1	Rp 1.500.000,00	2	0	1182,00
18	Moh. Nizar irawan buton	1	Rp 800.000,00	2	120000	1114,50
19	Moh. Yongki saputra	1	Rp 1.000.000,00	1	0	1124,00
20	Moh. Zamrotul fikri	1	Rp 800.000,00	2	100000	1153,50
21	Mohamad anam farhan	1	Rp 1.500.000,00	2	0	1215
22	Muhammad qoiyum	1	Rp 200.000,00	4	1396000	1245
23	Nova purwandinata	1	Rp 800.000,00	1	210000	1124,00
24	Rasia ilhamsah	1	Rp 800.000,00	1	130000	1131,50
25	Ratna cantika dewi	1	Rp 1.300.000,00	0	0	1238
26	Rayhan ferdiyansyah	1	Rp 800.000,00	3	350000	977,00
27	Rehan danuarta	1	Rp 800.000,00	1	110000	1095,50
28	Riski wahyuni	1	Rp 800.000,00	2	150000	1131,50
29	Ronald rafialdo febriandanu	1	Rp 800.000,00	2	140000	1137,50
30	Sahroni	1	Rp 300.000,00	4	1446000	1138,50
31	Shinici kudo	1	Rp 800.000,00	0	50000	1095,50
32	Siti aisyah	1	Rp 250.000,00	3	1319000	1254
33	Wardaniatus zahroh	1	Rp 1.300.000,00	2	0	1058,00
34	Wildan author bala	1	Rp 1.300.000,00	2	0	1056,00
35	Wulandari	1	Rp 800.000,00	3	270000	1165,50
36	Zulfan al farizi muchlis	1	Rp 1.500.000,00	1	0	1311,5
37	Moh ali sihab budin	1	Rp 1.300.000,00	5	0	1169,50
38	Abi alfa risqi	1	Rp 800.000,00	3	245000	1245
39	Aden kaprins	1	Rp 300.000,00	2	370000	1279

40	Aditiya	1	Rp 800.000,00	3	180000	1236
41	Ahmad angwar dimas ginam siyan	1	Rp 800.000,00	2	125000	1164,00
42	Ahmad rayhan hudaifah	1	Rp 1.300.000,00	4	200000	1330
43	Ajeng rahma dani	1	Rp 600.000,00	1	400000	1314,5
44	Almaddjid atoqillah	1	Rp 800.000,00	2	180000	1245
45	Amelia	1	Rp 800.000,00	2	150000	1315,5
46	Andrian saputro	1	Rp 800.000,00	2	130000	1116,00
47	Anisa audianto	1	Rp 1.300.000,00	3	100000	1361
48	Anisatul fariha	1	Rp 800.000,00	4	500000	1305
49	Ariska dwi ristani	1	Rp 1.300.000,00	2	100000	1282
50	Arisma dwi ristani	1	Rp 20.000,00	2	1396000	1277
51	Azizatul mas'ula	1	Rp 1.500.000,00	2	0	1313,5
52	Citra bunga mayasari	1	Rp 1.500.000,00	2	0	1350
53	Dewi agustin enjel monica	1	Rp 30.000,00	2	1476000	1244
54	Enggar audia salsabela putri	1	Rp 1.300.000,00	2	100000	1372
55	Febrianti dwi arlinda	1	Rp 800.000,00	2	0	1366
56	Feriska amelia	1	Rp 1.250.000,00	1	100000	1269
57	Huriyatul aizah	1	Rp 1.300.000,00	3	0	1341,5
58	Izzul rohiqim makhtum	1	Rp 1.500.000,00	2	0	1214,00
59	M. Halim farisi	1	Rp 280.000,00	6	1319000	1221
60	M. Irwansyah tegar maulana	1	Rp 1.500.000,00	3	0	1292
61	Maufidatul afriza	1	Rp 1.500.000,00	2	0	1352
62	Moh. Alvin hidayatullah	1	Rp 800.000,00	3	300000	1252,5
63	Moh. Firman syabana	1	Rp 250.000,00	3	996000	1227,5
64	Moh. Riski	1	Rp 1.300.000,00	1	0	1254
65	Muh. Naufal izzudin daffa	1	Rp 1.800.000,00	2	0	1240,5
66	Muhamad maulana ishak	1	Rp 800.000,00	1	50000	1211,00
67	Nadine safinatun aflah	1	Rp 1.500.000,00	2	0	1369
68	Naila istifania	1	Rp 800.000,00	2	100000	1377,5
69	Nailatul hasanah	1	Rp 1.300.000,00	2	0	1331,5
70	Qudrotul aini	1	Rp 800.000,00	3	350000	1373,5
71	Rimba intan laura	1	Rp 300.000,00	2	1396000	1363
72	Rosa ameliya putri	1	Rp 800.000,00	2	200000	1322
73	Shakilla natasya putri	1	Rp 500.000,00	2	175000	1347
74	Tio tri saputra	1	Rp 600.000,00	4	900000	1176,00
75	Wendra femas putra ramadhani	1	Rp 1.500.000,00	2	0	1268
76	Ahmad abul abbas	2	Rp 1.500.000,00	3	80000	1222
77	Ahmad bustomi bayazid	2	Rp 1.000.000,00	4	500000	1171
78	Alvin putra wijaya	2	Rp 800.000,00	2	0	1268,5
79	Angga tri putra	2	Rp 1.000.000,00	1	0	1278
80	Ayu isma ramadhani	2	Rp 1.200.000,00	2	0	1383
81	Cici aulia putri	2	Rp 900.000,00	3	100000	1187
82	Davis ikbal arrosyid	2	Rp 300.000,00	4	690000	1319,5
83	Dhea vivi indriyani	2	Rp 750.000,00	2	0	1313

84	Dimas al farisi	2	Rp 1.000.000,00	2	55000	1141
85	Destra nazariel akbar	2	Rp 1.000.000,00	1	0	1354,5
86	Dwi ratna ningrum	2	Rp 750.000,00	2	70000	1418,5
87	Fatin azzahra al islamiyah	2	Rp 1.500.000,00	1	0	1364
88	Fatur ega maulana	2	Rp 750.000,00	3	500000	1364,5
89	Fauziyatul ilmi	2	Rp 1.000.000,00	2	80000	1309
90	Fransiska dian novita	2	Rp 800.000,00	2	80000	1353,5
91	Hafiluh sagarul qoyyis	2	Rp 1.000.000,00	1	0	1182
92	Innes irmala dewi	2	Rp 600.000,00	1	0	1335,5
93	Kaigo vemas al bukhori	2	Rp 1.000.000,00	1	0	1295
94	Khoirul umam	2	Rp 1.500.000,00	2	0	1279,5
95	Marham efendi	2	Rp 900.000,00	1	50000	1036,5
96	Maulana bagus saputra	2	Rp 1.000.000,00	3	200000	1097,5
97	Maulida reza saila huda risqiana	2	Rp 750.000,00	2	0	1371
98	Maya mina indriani	2	Rp 1.000.000,00	2	0	1313
99	Mega nur alis	2	Rp 950.000,00	3	0	1368,5
100	Mohammad daniyal kurniawan	2	Rp 1.000.000,00	1	0	1303
101	Moch. Nabil akbar	2	Rp 950.000,00	2	0	1258,5
102	Muhammad rian andika	2	Rp 1.000.000,00	1	0	1172,5
103	Nicolas widyawan putra	2	Rp 750.000,00	1	0	1328,5
104	Nira dwi fantri	2	Rp 600.000,00	1	0	1370,5
105	Nola dwi agustin	2	Rp 1.000.000,00	1	0	1323,5
106	Pinky ramadhani	2	Rp 950.000,00	1	0	1290,5
107	Putri wardani	2	Rp 800.000,00	2	0	1297,5
108	Reyno verdy ramadhansyah	2	Rp 1.000.000,00	2	0	1198
109	Reza dwi permata sari	2	Rp 1.000.000,00	2	0	1357,5
110	Riska wazi'ah	2	Rp 1.000.000,00	3	150000	1423,5
111	Safa aurellia azahra	2	Rp 600.000,00	1	0	1305,5
112	Silviana wichdatul aulia	2	Rp 750.000,00	1	0	1389
113	Sofiatun nisa	2	Rp 1.000.000,00	1	0	1336
114	Adi jendro hayuningrat	2	Rp 680.000,00	1	0	1297
115	Ahmad danil rizal efendi	2	Rp 950.000,00	1	0	1196,5
116	Al masih isa muhamad	2	Rp 1.000.000,00	1	0	1372,5
117	Anggi novi rahmawati	2	Rp 1.500.000,00	2	0	1314
118	Bagus setyo	2	Rp 1.500.000,00	2	0	1177,5
119	Dela ameliya putri	2	Rp 650.000,00	3	2031000	1305
120	Della inka febriana	2	Rp 300.000,00	5	2151000	1315
121	Dwi yuliyati putri	2	Rp 900.000,00	2	100000	1248,5
122	Farel wurdiansyah	2	Rp 800.000,00	2	190000	1276
123	Gita gutawa	2	Rp 800.000,00	3	400000	1305,5
124	Helen wily amertha	2	Rp 1.500.000,00	3	0	1319,5
125	Ilham farel	2	Rp 1.200.000,00	3	0	1204
126	Ilyas wardana putra	2	Rp 1.500.000,00	2	0	1240
127	Intan belgis khumairoh	2	Rp 800.000,00	1	45000	1349

128	Intan nur aini	2	Rp 1.100.000,00	2	0	1303
129	Ivan agus bastian	2	Rp 1.100.000,00	1	0	1187,5
130	Kristina	2	Rp 800.000,00	1	100000	1316,5
131	Meisa wilyana salam	2	Rp 1.500.000,00	3	0	1338
132	Muhammad davin winata putra	2	Rp 800.000,00	2	0	1181,5
133	Muhammad reza febriyadi	2	Rp 900.000,00	3	500000	1243,5
134	Muhammad rohman	2	Rp 500.000,00	2	1931000	1219
135	Nanik sri wahyuni	2	Rp 600.000,00	1	0	1348,5
136	Nazwa ainur rohima	2	Rp 1.100.000,00	3	100000	1282,5
137	Neza auliya pratama putri	2	Rp 1.500.000,00	2	0	1303
138	Neza indira pratama putri	2	Rp 1.000.000,00	4	250000	1327
139	Putri rila agustina	2	Rp 900.000,00	4	850000	1304,5
140	Satrio aditiya nugroho	2	Rp 800.000,00	3	250000	1185
141	Septiayu isyana putri	2	Rp 800.000,00	2	175000	1330,5
142	Simon aminullah	2	Rp 900.000,00	2	0	1067,5
143	Siti nur alia	2	Rp 800.000,00	3	275000	1318
144	Tiara helmalia sari	2	Rp 1.500.000,00	2	0	1294,5
145	Wahyu cucu pamungkas	2	Rp 500.000,00	3	1546000	1252,5
146	Wilda nunafiah	2	Rp 800.000,00	4	400000	1373
147	Yunita ababil	2	Rp 1.000.000,00	3	0	1311,5
148	Ahmad afan rifa'i	3	Rp 950.000,00	3	0	1194,5
149	Ahmad afandi	3	Rp 400.000,00	4	3476000	1209
150	Annisa izmia rohman	3	Rp 800.000,00	1	0	1445,5
151	Cahana kanti widodo	3	Rp 1.500.000,00	2	0	1400
152	Cahani kanti widodo	3	Rp 900.000,00	3	0	1402,5
153	Devone sendy dwi artha	3	Rp 1.000.000,00	2	0	1402
154	Dia avina ismi	3	Rp 1.500.000,00	1	0	1422,5
155	Diah ayu safitri	3	Rp 800.000,00	1	130000	1395
156	Dimas pradana	3	Rp 800.000,00	2	230000	1247
157	Dina oktavia sabila	3	Rp 800.000,00	3	345000	1395,5
158	Diva maulina	3	Rp 1.500.000,00	3	0	1406,5
159	Eka sriwahyuni	3	Rp 1.500.000,00	2	0	1320
160	Felia meita sari	3	Rp 950.000,00	2	0	1342
161	Fitria monica	3	Rp 950.000,00	1	0	1292
162	Fitria wahyu ningseh	3	Rp 800.000,00	1	0	1340,5
163	Galih pradana	3	Rp 850.000,00	1	0	1226,5
164	Habib firmansyah	3	Rp 800.000,00	2	0	1290
165	Jaka kurnia	3	Rp 750.000,00	1	0	1271
166	Maya jihan nafani	3	Rp 800.000,00	2	0	1308
167	Muhammad hasbi assidqi	3	Rp 1.000.000,00	2	0	1405,5
168	Natasya salsa billa	3	Rp 800.000,00	2	250000	1381
169	Noval ibnu zaky	3	Rp 1.200.000,00	2	0	1397
170	Putri dwi ardella	3	Rp 1.200.000,00	3	0	1406
171	Rafli royhul akbar	3	Rp 1.200.000,00	3	0	1316,5

172	Rara kemuning	3	Rp 600.000,00	2	0	1429
173	Revi eka pratama	3	Rp 1.000.000,00	1	0	1314
174	Ricky yohana	3	Rp 1.000.000,00	2	0	1181
175	Risma fadilatul zanna	3	Rp 600.000,00	1	0	1338
176	Sahara fernanda	3	Rp 1.000.000,00	2	0	1167
177	Sakka arya jaya	3	Rp 1.500.000,00	2	0	1336,5
178	Sherli vila prastiwi	3	Rp 1.000.000,00	2	0	1335
179	Siti nur afinda	3	Rp 800.000,00	3	325000	1360,5
180	Tegar prasetyo	3	Rp 800.000,00	1	0	1327,5
181	Vicky hadi putra	3	Rp 800.000,00	1	125000	1334,5
182	Wahid maulana f	3	Rp 900.000,00	2	0	1330
183	Widia agustin	3	Rp 800.000,00	2	100000	1395,5
184	Abil alsya elisa	3	Rp 750.000,00	1	0	1326
185	Ahmad dani	3	Rp 1.000.000,00	3	0	1173
186	Ahmad ridho maulana	3	Rp 1.000.000,00	2	0	1151
187	Aisyah okta romadhona	3	Rp 1.000.000,00	2	0	1265
188	Alfan tio saputra	3	Rp 200.000,00	4	3686000	1268
189	Amin wafa	3	Rp 500.000,00	2	225000	1232
190	Andini sofi putri	3	Rp 750.000,00	3	450000	1328
191	Andre candra kirana	3	Rp 800.000,00	1	125000	1289
192	Antika wulandari	3	Rp 750.000,00	3	155000	1294,5
193	Dede alung nurmawati	3	Rp 500.000,00	1	800000	1294
194	Dela sintiana	3	Rp 1.000.000,00	2	0	1291
195	Dinda septia pratama	3	Rp 950.000,00	3	0	1102
196	Eka tio hafis hidayat	3	Rp 1.000.000,00	2	0	1368
197	Fariz djibran	3	Rp 750.000,00	2	235000	1306
198	Feby dwi andika	3	Rp 500.000,00	2	3146000	1279
199	Fernanda pamungkas	3	Rp 500.000,00	2	2151000	1060
200	Fitria andini	3	Rp 1.000.000,00	3	230000	1330
201	Halimatus sakdiyah	3	Rp 500.000,00	6	2151000	1265
202	Ilyas kurniawan	3	Rp 500.000,00	2	400000	1353
203	Indri vila anggraini	3	Rp 750.000,00	2	175000	1385
204	Lira amelia	3	Rp 750.000,00	2	100000	1353,5
205	Moh. Arif kurniawan	3	Rp 500.000,00	2	300000	1143
206	Moh. Femas indi sarif saputra	3	Rp 750.000,00	2	150000	1230
207	Mohammad arifin ilham	3	Rp 750.000,00	3	500000	1102
208	Mohammad rosit	3	Rp 900.000,00	0	0	1090
209	Muhammad febriyanto aditama	3	Rp 750.000,00	1	200000	1103
210	Muhammad mukhlisin	3	Rp 350.000,00	5	2738000	1206
211	Muhammat ikfan afandi	3	Rp 950.000,00	2	0	1266
212	Nazeril dimas wahyudi	3	Rp 1.000.000,00	2	0	1053
213	Novia luzain wardana yusra	3	Rp 750.000,00	2	450000	1330
214	Raditya akbar pratama	3	Rp 950.000,00	2	0	1200
215	Rhisanda marellita cahya	3	Rp 1.000.000,00	2	0	1380

216	Sandi ferdiansyah	3	Rp 500.000,00	3	700000	1143
217	Tendi arfianto cahya putra	3	Rp 750.000,00	2	120000	1130
218	Ulfa febriyanti	3	Rp 500.000,00	5	3446000	1205
219	Veni nurfaida	3	Rp 750.000,00	1	100000	1150
220	Vika anggiana pradevi	3	Rp 750.000,00	2	300000	1318
221	Yuda pratama	3	Rp 500.000,00	4	4141000	1210

