SISTEM PENDUKUNG KEPUTUSAN SELEKSI PENERIMAAN SISWA BARU MENGGUNAKAN METODE ELECTRE - *WEIGHTED PRODUCT* (STUDI KASUS: SMA BRAWIJAYA *SMART SCHOOL* KOTA MALANG)

SKRIPSI

LABORATORIUM KOMPUTASI DAN SISTEM CERDAS

Untuk memenuhi sebagian persyaratan mencapai gelar Sarjana Komputer



Disusun oleh:

MAHARANI PUTRI SEKAR NUGROHO

NIM. 105060807111005

KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI

UNIVERSITAS BRAWIJAYA

PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER

PROGRAM STUDI INFORMATIKA/ILMU KOMPUTER

MALANG

2015

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

SISTEM PENDUKUNG KEPUTUSAN SELEKSI PENERIMAAN SISWA BARU MENGGUNAKAN METODE ELECTRE - *WEIGHTED PRODUCT* (STUDI KASUS: SMA BRAWIJAYA *SMART SCHOOL* KOTA MALANG)

SKRIPSI

LABORATORIUM KOMPUTASI DAN SISTEM CERDAS

Untuk memenuhi sebagian persyaratan mencapai gelar Sarjana Komputer



Disusun oleh:

MAHARANI PUTRI SEKAR NUGROHO

NIM. 105060807111005

Skripsi ini telah diperiksa dan disetujui oleh:

Pembimbing I	Pembimbing II
Arief Andy Soebroto, S.T., M.Kom.	**M. Ali Fauzi, S.Kom., M.Kom.**
NIP. 19720425 199903 1 002	**-**

LEMBAR PENGESAHAN

SISTEM PENDUKUNG KEPUTUSAN SELEKSI PENERIMAAN SISWA BARU MENGGUNAKAN METODE ELECTRE - *WEIGHTED PRODUCT* (STUDI KASUS: SMA BRAWIJAYA *SMART SCHOOL* KOTA MALANG)

SKRIPSI
LABORATORIUM KOMPUTASI DAN SISTEM CERDAS
Untuk memenuhi sebagian persyaratan mencapai gelar Sarjana Komputer

Disusun oleh:
MAHARANI PUTRI SEKAR NUGROHO
NIM. 105060807111005

Setelah dipertahankan di depan Majelis Penguji
pada tanggal 29 Januari 2015
dan dinyatakan memenuhi syarat untuk memperoleh
gelar Sarjana dalam bidang Ilmu Komputer

Penguji I	Penguji II
Nurul Hidayat, S.Pd., M.Sc.	**Wayan Firdaus Mahmudy, S.Si., MT., Ph.D**
NIP. 196804302002121001	**NIP. 197209191997021001**

Penguji III

Barlian Henryranu P, S.T., M.T.
NIK. 82102406110254

Mengetahui,
Ketua Program Studi Informatika / Ilmu Komputer

Drs. Marji, M.T.
NIP. 19670801 199203 1 001

PERNYATAAN
ORISINALITAS SKRIPSI

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 dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan 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, 29 Januari 2015
Penulis,

Maharani Putri S N
NIM. 105060807111005

KATA PENGANTAR

Puji dan syukur Penulis panjatkan kehadirat Allah SWT, karena hanya dengan rahmat dan karunia-Nya Penulis dapat menyelesaikan skripsi dengan judul **"Sistem Pendukung Keputusan Seleksi Penerimaan Siswa Baru Menggunakan Metode Electre - *Weighted Product* (Studi Kasus: SMA Brawijaya *Smart School* Kota Malang)"** dengan baik. Melalui kesempatan ini, Penulis ingin menyampaikan rasa hormat dan terima kasih yang sebesar-besarnya kepada semua pihak yang telah memberikan bantuan dan dukungan selama pengerjaan skripsi, diantaranya:

1. Bapak Arief Andy S., S.T., M.Kom. selaku dosen pembimbing I yang telah banyak memberikan ilmu dan saran selama penyusunan skripsi ini, sekaligus dosen pembimbing akademik yang telah banyak memberikan nasehat-nasehat dan dorongan kepada penulis selama perkuliahan.
2. Bapak M. Ali Fauzi, S.Kom., M.Kom. selaku dosen pembimbing II yang juga telah banyak memberikan ilmu dan saran selama penyusunan skripsi ini.
3. Kedua orang tua, Ayah Gunung Sri Nugroho dan Bunda Leli yang telah memberikan dukungan moral dan material, memberikan semangat, kepercayaan, dan kasih sayang selama ini. Terima kasih atas doa yang tak terhingga, pengorbanan, dan ketulusannya dalam mendampingin penulis. Semoga Allah SWT senantiasa melimpahkan rahmat dan ridho-Nya kepada keduana. Terima kasih Ayah dan Bunda.
4. Kedua adikku, Ahmad Fauzi Akbar Nugroho dan Dara Rizki Zakiyyah Nugroho, yang selalu memberi motivasi bagi penulis agar selalu semangat dalam menjalani perkuliahan.
5. Segenap bapak dan ibu dosen yang telah mendidik dan mengajarkan ilmunya kepada Penulis selama menempuh pendidikan di Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
6. Ni Ketut Novia Nilasari, Fawwaz Ali Akbar, Rohman Widianto, Irwan Nugrahanto, Dio Dharwaman, serta Nur Rokhim Satria Nugraha yang telah menjadi sahabat penulis dari awal perkuliahan hingga saat ini. Terima kasih

atas segala bantuan, semangat, dan doa yang diberikan kepada penulis. Terima kasih teman-teman.

7. Semua teman-teman PTIIK, khususnya Informatika/Ilmu Komputer angkatan 2010 terima kasih atas segala bantuan dan dukungannya selama ini.
8. Semua pihak yang tidak dapat penulis sebutkan satu per satu yang terlibat baik secara langsung maupun tidak langsung demi terselesaikannya skripsi ini.

Semoga jasa dan amal baik mendapatkan balasan dari Allah SWT. Dengan segala kerendahan hati, penulis menyadari sepenuhnya bahwa skripsi ini masih jauh dari sempurna karena keterbatasan materi dan pengetahuan yang dimiliki penulis. Akhirnya, semoga skripsi ini dapat bermanfaat dan berguna bagi pembaca terutama mahasiswa PTIIK Universitas Brawijaya.

Malang, Desember 2014

Penulis

repository.ub.ac.id

ABSTRAK

Maharani Putri Sekar Nugroho. 2015. Sistem Pendukung Keputusan Seleksi Penerimaan Siswa Baru Menggunakan Metode Electre - *Weighted Product* (Studi Kasus: SMA Brawijaya *Smart School* Kota Malang). Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya, Malang. Dosen Pembimbing: Arief Andy S., S.T., M.Kom. dan M. Ali Fauzi, S.Kom., M.Kom

Sekolah merupakan tempat belajar mengajar yang dilakukan secara formal dengan tujuan untuk menghasilkan pribadi yang lebih baik serta memiliki keterampilan dibidang akademik maupun non akademik. Setiap sekolah memiliki proses seleksi dan kriteria yang berbeda untuk mengetahui tingkat kompetensi dan minat dari calon siswa baru. Tingkat kompetensi calon siswa yang berbeda-beda dan proses peminatan kelas yang harus dilakukan ketika siswa baru diterima, terkadang memberikan kesulitan tersendiri bagi pihak sekolah. Sistem pendukung keputusan dapat menjadi solusi untuk membant proses seleksi dan peminatan sesuai dengan kriteria sekolah. Pada sistem tersebut akan menerapkan metode *Elimination and Choice Translating Reality* (ELECTRE) untuk proses seleksi dan metode *Weighted Product* untuk proses peminatan. Hasil pengujian akurasi memiliki tingkat kesesuaian presentase sebesar 84.37%.

Kata Kunci: Seleksi, *Elimination and Choice Translating Reality* (ELECTRE), *Weighted Product,* Seleksi, Peminatan, SPK.

UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

ABSTRACT

Maharani Putri Sekar Nugroho. 2015. ***Decision Support System for the Selection of New Students Admissions Using ELECTRE Method and Weighted Product*** **(SMA Brawijaya *Smart School*: *a Case Study*)**. *Information Technology and Computer Science Program, Brawijaya University, Malang. Advisor*: Arief Andy S., S.T., M.Kom. *and* M. Ali Fauzi, S.Kom., M.Kom

School is a place for formal teaching and learning with the purpose to produce students with good personality and has academic skills and non-academic skills. All schools have admission criteria to identify new student competencies and interests. Admission criteria and requirments are different for each school. Students has different competency to admission requirements and all students placements must be done after admission, may make school admission further away more difficult. Decision Support System can be used to support admission and specialization class placement decision making for school . This system use Elimination and Choice Translating Reality (ELECTRE) for selection and Weighted Product for specialization class placement. This system has a 84.37% accuracy rate.

Keywords: *admission, Elimination and Choice Translating Reality (ELECTRE), Weighted Product, Decision Support System*

UNIVERSITAS BRAWIJAYA

DAFTAR ISI

PERNYATAAN ORISINALITAS SKRIPSI iii
KATA PENGANTAR iv
ABSTRAK vi
ABSTRACT vii
DAFTAR ISI viii
DAFTAR GAMBAR xii
DAFTAR PERSAMAAN xvi
DAFTAR TABEL xvii
BAB I 21
PENDAHULUAN 21
1 Latar 21
1.1 Latar Belakang 21
1.2 Rumusan Masalah 23
1.3 Batasan Masalah 23
1.4 Tujuan 24
1.5 Manfaat 24
1.6 Sistematika Penulisan 25
BAB II 26
TINJAUAN PUSTAKA 26
2 Tinjauan 26
2.1 Kajian Pustaka 26
2.2 SMA Brawijaya *Smart School* (SMA BSS) 35
2.2.1 Seleksi Penerimaan Siswa Baru 36
2.2.2 Kriteria 40

2.2.3 Peminatan 41
2.3 Sistem Pendukung Keputusan 49
2.3.1 Dasar Pengambilan Keputusan 49
2.3.2 Karakteristik Sistem Pendukung Keputusan 51
2.3.3 Komponen Sistem Pendukung Keputusan 52
2.4 *Multiple Attribute Decision Making* (MADM) 53
2.4.1 Elimination Et Choix Tranduisant La Realité (ELECTRE) 54
2.4.2 Weighted Product 59
2.5 Bahasa Pemrograman C# 61
2.6 *Database Management System* MySQL 61
2.7 *Data Flow Diagram* (DFD) 62
2.8 *Entity-Relationship Diagram* (ERD) 70
2.9 Teknik Pemberian Bobot 71
2.9.1 Direct Weight Elication Techinques 71
2.9.2 Indirect Weight Elication Techinques 74
2.10 *Precision and Recall* 75
2.11 Akurasi 76
BAB III 77
METODOLOGI PELAKSANAAN 77
3 Metodologi 77
3.1 Studi Literatur 78
3.2 Pengumpulan Data 78
3.3 Preproses Data 78
3.4 Analisa Kebutuhan 79
3.5 Perancangan Sistem 79
3.5.1 Model Perancangan Sistem 79

3.5.2 Arsitektur Pendukung Keputusan 82
3.6 Implementasi Sistem 83
3.7 Uji Coba Sistem 83
3.8 Kesimpulan 83
4 Analisa Dan Perancangan 84
4.1 Analisa Kebutuhan Perangkat Lunak 84
4.1.1 Identifikasi Pengguna 85
4.1.2 Daftar Kebutuhan Sistem 86
4.2 Perancangan Sistem Pendukung Keputusan 88
4.2.1 Subsistem Manajemen Data 89
4.2.2 Subsistem Basis Pengetahuan 110
4.2.3 Subsistem Manajemen Model 115
4.2.4 Subsistem Antarmuka 159
BAB V 177
IMPLEMENTASI 177
5 implementasi 177
5.1 Spesifikasi Sistem 178
5.1.1 Spesifikasi Perangkat Keras 178
5.1.2 Spesifikasi Perangkat Lunak 178
5.2 Batasan Implementasi 178
5.3 Implementasi Algoritma 180
5.3.1 ELECTRE 180
5.3.2 *Weighted Product* 192
5.4 Implementasi Antarmuka 197
5.4.1 Tampilan Antarmuka Halaman Login 198
5.4.2 Tampilan Antarmuka Halaman Staff IT 198

5.4.3 Tampilan Antarmuka Kepala Sekolah/Ketua PPDB 201
5.4.4 Tampilan Antarmuka Sie Test/Guru BK 203
BAB VI 206
PENGUJIAN DAN ANALISA 206
6 Pengujian 206
6.1 Pengujian Fungsionalitas 206
6.1.1. Skenario Pengujian Fungsionalitas 207
6.2 Pengujian Akurasi Sistem 232
6.2.1 Skenario Pengujian Akurasi 232
6.3 Analisa Pengujian Tingkat Akurasi 250
6.4 Precision and Recall 253
6.4.1 Perhitungan *precision* and *recall* 25 data 253
6.4.2 Perhitungan *precision* and *recall* 32 data 261
6.4.3 Analisis Hasil Pengujian Precision & Recall 269
6.5 Pengujian Akurasi Menggunakan Kuota 270
BAB VII 274
KESIMPULAN DAN SARAN 274
7 Kesimpulan Dan Saran 274
7.1 Kesimpulan 274
7.2 Saran 275
DAFTAR PUSTAKA 276

DAFTAR GAMBAR

Gambar 2.1. Kriteria Input Peminatan Seleksi SMA BSS 33
Gambar 2.2. Kriteria Input Proses Penerimaan Seleksi SMA BSS..... 34
Gambar 2.3. Struktur Organisasi Brawijaya Smart School..... 36
Gambar 2.4 Diagram Blok Proses Penerimaan Siswa Baru..... 37
Gambar 2.5 Diagram Blok Alur Penilaian 38
Gambar 2.6 Diagram Alir Proses Peminatan SMA BSS..... 42
Gambar 2.7 Fase-fase Pengambilan Keputusan 51
Gambar 2.8 Skematik SPK..... 52
Gambar 2.9. Simbol Terminator..... 63
Gambar 2.10. Jenis Terminator 63
Gambar 2.11. Simbol Proses 64
Gambar 2.12. Jenis Proses..... 64
Gambar 2.13. Simbol Data store 65
Gambar 2.14. Jenis Data store..... 65
Gambar 2.15. Konsep Paket Data..... 66
Gambar 2.16. Konsep Alur Data Menyebar..... 67
Gambar 2.17. Konsep Alur Data Menyebar..... 67
Gambar 2.18. Simbol Data flow..... 68
Gambar 2.19. Konsep Data flow 68
Gambar 2.20. Levelisasi DFD..... 69
Gambar 2.21. Simbol ERD..... 71
Gambar 2.22. Macam Relasi 71
Gambar 3.1 Diagram Blok Metodologi Penelitian..... 77
Gambar 3.2 Model Perancanan SPK seleksi penerimaan siswa baru 80
Gambar 3.3 Arsitektur SPK Penerimaan Siswa Baru SMA BSS..... 82
Gambar 4.1 Pohon Perancangan Sistem..... 84
Gambar 4.2 Konteks Diagram..... 90
Gambar 4.3 DFD Level 0 97
Gambar 4.4 DFD Level 1 Proses Kelola Data Pengguna..... 98
Gambar 4.5 DFD Level 1 Proses Kelola Data Tahun Ajaran 98

Gambar 4.7 DFD Level 1 Proses Kelola Data Jalur Masuk 99
Gambar 4.8 DFD Level 1 Proses Kelola Data Kriteria dan Bobot 99
Gambar 4.9 DFD Level 1 Proses Kelola Data Siswa 100
Gambar 4.10. *Entity Relationship Diagram* 100
Gambar 4.11. *Physical Diagram* 102
Gambar 4.12. Kerangka Kerja Model Komputasi 115
Gambar 4.13. Kerangka Kerja Metode ELECTRE 116
Gambar 4.14. Perancangan Algoritma ELECTRE 117
Gambar 4.15. Kerangka Kerja Metode *Weighted Product* 118
Gambar 4.16. Perancangan Algoritma Weighted Product 120
Gambar 4.17. Diagram Alir Matriks Normalisasi ELECTRE 123
Gambar 4.18. Psedocode Matriks Normalisasi ELECTRE 124
Gambar 4.19. Diagram Alir Matriks Normalisasi Terbobot ELECTRE 124
Gambar 4.20. Pseudocode Perhitungan Matriks Normalisasi Terbobot ELECTRE 125
Gambar 4.21. Diagram Alir Penentuan Himpunan Concordance 128
Gambar 4.22. Pseudocode Penentuan Himpunan Concordance 129
Gambar 4.23. Diagram Alir Penentuan Himpunan Discordance 131
Gambar 4.24. Pseudocode Penentuan Himpunan Discordance 132
Gambar 4.25. Diagram Alir Penentuan Matriks Concordance 133
Gambar 4.26. Pseudocode Penentuan Matriks Concordance 134
Gambar 4.27. *Diagram Alir Penentuan Matriks* Discordance 137
Gambar 4.28. Pseudocode Penentuan Matriks Discordance 138
Gambar 4.29. Diagram Alir Perhitungan Threshold Concordance 139
Gambar 4.30. Pseudocode Perhitungan Threshold Concordance 139
Gambar 4.31. Diagram Alir Penentuan Dominan Concordance 141
Gambar 4.32. Pseudocode Penentuan Matriks Dominan Concordance 142
Gambar 4.33. Diagram Alir Perhitungan Threshold Discordance 143
Gambar 4.34. Pseudocode Perhitungan Threshold Discordance 143
Gambar 4.35. Diagram Alir Penentuan Dominan *Discordance* 144
Gambar 4.36. Pseudocode Penentuan Matriks Dominan Discordance 145
Gambar 4.37. Diagram Alir Penentuan Matriks Dominan *Aggregate* 146

Gambar 4.38. Pseudocode Penentuan Matriks Dominan Aggregate 147
Gambar 4.39. Diagram Alir Eliminasi Alternatif..... 149
Gambar 4.40. Pseudocode Eliminasi Alternatif 150
Gambar 4.41. Pseudocode Menghitung Vektor S 155
Gambar 4.42. Alur Data Menghitung Vektor S 156
Gambar 4.43. Pseudocode Menghitung Vektor V..... 157
Gambar 4.44. Pseudocode Menghitung Vektor V..... 158
Gambar 4.45 Sitemap Halaman *Sie Test dan Guru BK* 159
Gambar 4.46 Sitemap Halaman IT 159
Gambar 4.47 Sitemap Halaman Kepala Sekolah/Ketua PPDB..... 160
Gambar 4.48 Perancangan Antarmuka Login 160
Gambar 4.49 Perancangan Antarmuka Halaman Utama Admin..... 161
Gambar 4.50 Perancangan Antarmuka Halaman Data Pengguna 162
Gambar 4.51 Perancangan Antarmuka Halaman Data Tahun Ajaran..... 163
Gambar 4.52 Perancangan Antarmuka Halaman Data Jalur Masuk 164
Gambar 4.53 Perancangan Antarmuka Halaman Data Kelas Peminatan..... 165
Gambar 4.54 Perancangan Antarmuka Utama Kepala Sekolah dan Ketua PPDB 166
Gambar 4.55 Perancangan Antarmuka Halaman Kriteria dan Bobot Peminatan 167
Gambar 4.56 Perancangan Antarmuka Halaman Kriteria dan Bobot Peminatan 168
Gambar 4.57 Perancangan Antarmuka Halaman Data Rekomendasi 169
Gambar 4.58 Perancangan Antarmuka Halaman Data Rekomendasi 170
Gambar 4.59 Perancangan Antarmuka Utama Sie Test dan Guru BK 171
Gambar 4.60 Perancangan Antarmuka Siswa Sie Test dan Guru BK..... 172
Gambar 4.61 Perancangan Antarmuka Psikotes Siswa Sie Test dan Guru BK 174
Gambar 4.62 Perancangan Antarmuka Hasil Seleksi Sie Test dan Guru BK ... 176
Gambar 5.1 Pohon Implemantasi 177
Gambar 5.2 Implementasi Algoritma Matriks Normalisasi 181
Gambar 5.3 Implenetasi Algoritma Matriks Normalisasi Terbobot..... 182

Gambar 5.4 Implentasi Agoritma penentuan himpunan *concordance* ELECTRE 183
Gambar 5.5 Implementasi Algoritma Himpunan *Discordance* 184
Gambar 5.6 Implementasi Algoritma Matriks *Concordance* 185
Gambar 5.7 Implementasi Algoritma Matriks *Discordance* 187
Gambar 5.8 Implementasi *Threshold Concordance* dan *Threshold Discordance* 188
Gambar 5.9 Implementasi Algoritma Matriks Dominan *Concordance* dan *Discordance* 189
Gambar 5.10 Implementasi Algoritma Matriks Dominan *Aggregate* 190
Gambar 5.11 Implementasi Algoritma Alternatif *Less Favorable* 191
Gambar 5.12 Implementasi Algoritma Vektor S 194
Gambar 5.13 Implementasi Algoritma Vektor V 197
Gambar 5.14 Implementasi Antarmuka Halaman Login 198
Gambar 5.15 Implementasi Antarmuka Halaman Pengguna 199
Gambar 5.16 Implementasi Antarmuka Halaman Tahun Ajaran 199
Gambar 5.17 Implementasi Antarmuka Halaman Jenis Minat 200
Gambar 5.18 Implementasi Antarmuka Halaman Jalur Test 201
Gambar 5.19 Implementasi Antarmuka Halaman Data Calon Calon siswa baru 201
Gambar 5.20 Implementasi Antarmuka Halaman Data Psikotes Calon siswa baru 202
Gambar 5.21 Implementasi Antarmuka Halaman Hasil Seleksi 203
Gambar 5.22 Implementasi Antarmuka Halaman Bobot Penerimaan 204
Gambar 5.23 Implementasi Antarmuka Bobot Peminatan 204
Gambar 5.24 Implementasi Antarmuka Hasil Seleksi 205
Gambar 5.25 Implementasi Antarmuka Bobot Peminatan 205
Gambar 6.1 Pohon Pengujian dan Analisa 206
Gambar 6.2 Grafik Tingkat Akruasi Hasil Pengujian Nilai Batas 251
Gambar 6.3 Grafik Tingkat Akruasi Hasil Precision & Recall Kuota 273

DAFTAR PERSAMAAN

Persamaan 2.1 Normalisasi Matriks Keputusan 54

Persamaan 2.2 Matrikss R Hasil Normalisasi 55

Persamaan 2.3 Pembobotan Matriks Ternormalisasi 55

Persamaan 2.4 Matriks Bobot Nilai Kepentingan 55

Persamaan 2.5 Menentukan Himpunan Concordance 56

Persamaan 2.6 Himpunan Concordance 56

Persamaan 2.7 Matriks Concordance 57

Persamaan 2.8 Matriks Discordance 57

Persamaan 2.9 Nilai Treshold Concordance 58

Persamaan 2.10 Matriks Dominan Concordance 58

Persamaan 2.11 Matriks Dominan Discordance 58

Persamaan 2.12 Nilai Treshold Discordance 59

Persamaan 2.13 Nilai Aggregate Dominance Ma 59

Persamaan 2.14 Persamaan Alternatif 60

Persamaan 2.15 Preferensi relatif 60

Persamaan 2.16 Rank-Order Centroid Technique 72

Persamaan 2.17 Rank-Order Centroid Technique W_1 72

Persamaan 2.18 Rank-Order Centroid Technique W_m 72

Persamaan 2.19 Rank Sum 73

Persamaan 2.20 Rank Exponent 73

Persamaan 2.21Rank Reciprocal 74

Persamaan 2.22 Precision 75

Persamaan 2.23Recall 75

Persamaan 2.24Precision 75

DAFTAR TABEL

Tabel 2.1 Kajian Pustaka 28
Tabel 2.2 Contoh Perhitungan Manual 39
Tabel 2.3 Perhitungan Manual SMA BSS 39
Tabel 2.4 Perbandingan Manual SMA BSS 46
Tabel 2.5 Tabel ROC 73
Tabel 2.6 Perhitungan Penentuan Recall dan Precision 75
Tabel 4.1 Identifikasi Pengguna 85
Tabel 4.2 Daftar Kebutuhan Sistem 86
Tabel 4.3 Struktur Tabel Calon siswa baru 103
Tabel 4.4 Struktur Tabel Tahun Ajaran 104
Tabel 4.5 Struktur Tabel Jalur Test 104
Tabel 4.6 Struktur Tabel Jenis Minat 104
Tabel 4.7 Struktur Tabel Nilai Psikotest 104
Tabel 4.8 Struktur Tabel Tahun Ajaran 105
Tabel 4.9 Struktur Tabel Kriteria Bobot Penerimaan 105
Tabel 4.10 Struktur Tabel Kriteria Bobot Peminatan 105
Tabel 4.11 Struktur Tabel Admin 106
Tabel 4.12 Struktur Tabel Admin 106
Tabel 4.13 Struktur Tabel Pengguna 107
Tabel 4.14 Struktur Tabel Nilai Rapor 107
Tabel 4.15 Struktur Tabel Psikotes 108
Tabel 4.16 Struktur Tabel Peminatan 109
Tabel 4.17 Struktur Tabel Rekomendasi BK 109
Tabel 4.18 Struktur Tabel Tes IPS 110
Tabel 4.19 Kriteria Penerimaan Calon Siswa Baru 111
Tabel 4.20. Data Pembobotan Kriteria 111
Tabel 4.21. Kriteria Peminatan Calon Siswa Baru 111
Tabel 4.22 Range Kelompok Peminatan IPA 112
Tabel 4.23 Range Kelompok Peminatan IPS 113
Tabel 4.24 Range Nilai Aspek Psikotes 113

Tabel 4.25 Alternatif Keputusan 113
Tabel 4.26 Bobot Kriteria Peminatan 113
Tabel 4.27 Matriks Keputusan Calon Calon siswa baru (ELECTRE) 120
Tabel 4.28 Nilai Bobot Kriteria (ELECTRE) 121
Tabel 4.29 Tabel Pengambilan Keputusan 150
Tabel 4.30 Data Perhitungan Weighted Product 151
Tabel 4.31 Bobot Peminatan 152
Tabel 4.32 Hasil Pembobotan 153
Tabel 4.33 Matriks Keputusan Data Alternatif 153
Tabel 4.34 Kesimpulan 158
Tabel 5.1Spesifikasi Perangkat Keras 178
Tabel 5.2 Spesifikasi Perangkat Lunak 178
Tabel 6.1 Penjelasan Kasus Uji untuk Pengujian Proses Login 207
Tabel 6.2 Penjelasan Kasus Uji untuk Pengujian Halaman Utama (IT Service) 208
Tabel 6.3 Penjelasan Kasus Uji untuk Pengujian Proses Kelola Data Pengguna 208
Tabel 6.4 Penjelasan Kasus Uji untuk Pengujian Proses Tambah Data Pengguna 209
Tabel 6.5 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Pengguna 210
Tabel 6.6 Penjelasan Kasus Uji untuk Pengujian Proses Hapus Data Pengguna 210
Tabel 6.7 Penjelasan Kasus Uji untuk Pengujian Proses Kelola Data Tahun Ajaran 211
Tabel 6.8 Penjelasan Kasus Uji untuk Pengujian Proses Tambah Data Tahun Ajaran 211
Tabel 6.9 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Tahun Ajaran 212
Tabel 6.10 Penjelasan Kasus Uji untuk Pengujian Proses Hapus Data Tahun Ajaran 213
Tabel 6.11 Penjelasan Kasus Uji untuk Pengujian Proses Kelola Data Jalur Masuk 213
Tabel 6.12 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Jalur Masuk 214

Tabel 6.13 Penjelasan Kasus Uji untuk Pengujian Proses Kelola Data Kelas Peminatan 214
Tabel 6.14 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Kelas Peminatan 215
Tabel 6.15 Penjelasan Kasus Uji untuk Pengujian Halaman Utama (Kepala Sekolah / Ketua PPDB) 216
Tabel 6.16 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Kriteria dan Bobot Seleksi 216
Tabel 6.17 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Kriteria dan Bobot Peminatan 217
Tabel 6.18 Penjelasan Kasus Uji untuk Pengujian Lihat Hasil Rekomendasi Seleksi 218
Tabel 6.19 Penjelasan Kasus Uji untuk Pengujian Lihat Hasil Rekomendasi Peminatan 218
Tabel 6.20 Penjelasan Kasus Uji untuk Pengujian Halaman Utama (Sie Tes / Guru BK) 219
Tabel 6.21 Penjelasan Kasus Uji untuk Pengujian Proses Tambah Data Calon siswa baru 219
Tabel 6.22 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Calon siswa baru 220
Tabel 6.23 Penjelasan Kasus Uji untuk Pengujian Proses Hapus Data Calon siswa baru 221
Tabel 6.24 Penjelasan Kasus Uji untuk Pengujian Proses Tambah Data Calon siswa baru 221
Tabel 6.25 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Calon siswa baru 222
Tabel 6.26 Penjelasan Kasus Uji untuk Pengujian Proses Hapus Data Psikotes Calon siswa baru 223
Tabel 6.27 Penjelasan Kasus Uji untuk Pengujian Lihat Hasil Rekomendasi ... 223
Tabel 6.28 Penjelasan Kasus Uji untuk Pengujian Proses Logout 224
Tabel 6.29 Hasil Pengujian Fungsionalitas Sistem 224
Tabel 6.30 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 0 233

Tabel 6.31 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 3..... 234
Tabel 6.32 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 5..... 236
Tabel 6.33 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 7..... 237
Tabel 6.34 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 8..... 238
Tabel 6.35 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 9..... 239
Tabel 6.36 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 11..... 240
Tabel 6.37 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 0..... 241
Tabel 6.38 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 3..... 242
Tabel 6.39 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 5..... 243
Tabel 6.40 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 7..... 245
Tabel 6.41 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 8..... 246
Tabel 6.42 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 9..... 247
Tabel 6.43 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 11..... 248
Tabel 6.44 Hasil Perbandingan Seleksi dan Peminatan Perhitungan Manual dan Sistem..... 249
Tabel 6.45 Hasil *Precision and Recall* 25 Data dengan Nilai Batas > 0..... 253
Tabel 6.46 Hasil *Precision and Recall* 25 Data dengan Nilai Batas > 3..... 254
Tabel 6.47 Hasil *Precision and Recall* 25 Data dengan Nilai Batas > 5..... 256
Tabel 6.48 Hasil *Precision and Recall* 25 Data dengan Nilai Batas > 7..... 257
Tabel 6.49 Hasil *Precision and Recall* 25 Data dengan Nilai Batas >8..... 258
Tabel 6.50 Hasil *Precision and Recall* 25 Data dengan Nilai Batas > 9..... 259
Tabel 6.51 Hasil *Precision and Recall* 25 Data dengan Nilai Batas > 11..... 260
Tabel 6.52 Hasil *Precision and Recall* 32 Data dengan Nilai Batas > 0..... 261
Tabel 6.53 Hasil *Precision and Recall* 32 Data dengan Nilai Batas > 3..... 262
Tabel 6.54 Hasil *Precision and Recall* 32 Data dengan Nilai Batas > 5..... 263
Tabel 6.55 Hasil *Precision and Recall* 32 Data dengan Nilai Batas > 7..... 264
Tabel 6.56 Hasil *Precision and Recall* 32 Data dengan Nilai Batas > 8..... 266
Tabel 6.57 Hasil *Precision and Recall* 32 Data dengan Nilai Batas > 9..... 267
Tabel 6.58 Hasil *Precision and Recall* 32 Data dengan Nilai Batas >11..... 268
Tabel 6.59 Hasil Seleksi Kuota..... 271
Tabel 6.60 Hasil *Precision and Recall* Kuota..... 272

BAB I
PENDAHULUAN

1.1 Latar Belakang

Sekolah merupakan tempat belajar mengajar yang dilakukan secara formal dengan tujuan untuk menghasilkan pribadi yang lebih baik serta memiliki keterampilan dibidang akademik maupun non akademik [21]. Berdasarkan tujuan tersebut, tidak jarang sekolah-sekolah mengadakan proses seleksi dengan syarat-syarat tertentu untuk mengetahui tingkat kompetensi dan minat dari calon siswa baru. Adanya kriteria, keterbatasan waktu dalam melakukan seleksi, serta banyaknya jumlah pendaftar memberikan kesulitan tersendiri bagi pihak sekolah untuk menyeleksi sekaligus mengelompokkan calon siswa baru sesuai dengan minatnya yaitu IPA atau IPS [24]. Kesulitan tersebut muncul dikarenakan proses seleksi yang digunakan masih menggunakan cara manual dan perbandingan manual. Penggunaan teknologi informasi seperti sistem pendukung keputusan, diharapkan dapat memberikan solusi bagi pengguna dalam membantu proses pengambilan keputusan terkait seleksi penerimaan calon siswa baru baru[10].

Penelitian yang berkaitan dengan sistem pendukung keputusan dengan metode *Elimination and Choice Translating Reality* (ELECTRE) dan *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS) telah dilakukan oleh Rizqi (2009) [22]. Rizqi melakukan penelitian dengan membuat sistem pendukung keputusan pemilihan bahan dasar obat alternatif. Penelitian tersebut menggunakan metode ELECTRE untuk menyeleksi alternatif bahan obat tradisional dan metode TOPSIS yang digunakan untuk melakukan perankingan terhadap alternatif bahan obat tradisional. Hasil dari penelitian tersebut adalah urutan alternatif bahan obat tradisional yang sesuai dengan kriteria yang dimasukkan oleh pengguna. Penelitian ini membuktikan bahwa metode ELECTRE dan TOPSIS dapat melakukan pemilihan obat alternatif berdasarkan kriteria yang telah ditentukan [22]. Penelitian selanjutnya adalah penelitian yang dilakukan oleh Karnajaya dengan judul Rancang

Bangun Sistem Pendukung Keputusan Pemilihan Sepeda Motor Honda Menggunakan Metode Electre Berbasis Web (Studi Kasus di PT. Lumenindo Gilang Cahaya) [15]. Penelitian ini menghasilkan kesimpulan bahwa metode ELECTRE dapat digunakan untuk membandingkan dan memilih sepeda motor Honda sesuai dengan kriteria dan bobot yang telah ditentukan oleh user [15].

Penelitian yang berkaitan dengan penjurusan siswa SMA telah dilakukan oleh Ardianto (2013) dengan judul "Sistem Pendukung Keputusan Pemilihan Jurusan Dengan Metode *Weighted Product* Untuk Siswa-Siswi SMA Negeri 1 SALE". Penelitian ini memiliki dua alternatif penjurusan, yakni IPA dan IPS dengan menggunakan lima kriteria sebagai acuan dalam pengambilan keputusan. Penelitian ini menghasilkan kesimpulan bahwa metode *Weighted Product* dapat diimplementasikan ke sistem pendukung keputusan pemilihan jurusan di SMAN 1 SALE [2].

SMA Brawijaya *Smart School* (SMA BSS) Kota Malang adalah Sekolah Menengah Atas Nasional dalam naungan Universitas Brawijaya dan dipersiapkan untuk diterima di Universitas Brawijaya dan Universitas Negeri lain. Pada tahun ajaran 2012/2013, 85% siswa SMA BSS diterima di Universitas Brawijaya. Jumlah pendaftar calon siswa baru pada tahun ajaran 2013/2014 mengalami peningkatan sebesar 100%, yakni dari 200 pendaftar calon siswa baru pada tahun ajaran 2012/2013 meningkat menjadi 600 pendaftar calon siswa baru. SMA BSS menggunakan kurikulum 2013 dengan proses seleksi yang terdiri dari proses penerimaan dan proses peminatan. SMA BSS melakukan proses seleksi seleksi siswa baru yang sangat ketat untuk memilih calon siswa-siswi terbaik yang nantinya akan dipersiapkan untuk diterima di Universitas Brawijaya [6].

Berdasarkan paparan yang telah dijelaskan, penulis mengusulkan penelitian yang berjudul "Sistem Pendukung Keputusan Seleksi Penerimaan Siswa Baru Menggunakan Metode ELECTRE - *Weighted Product* (Studi Kasus: SMA Brawijaya *Smart School* Kota Malang)". Penelitian ini menggunakan Metode ELECTRE diawal karena pada permasalahan ini dilakukan pengelompokkan terlebih dahulu dengan cara membandingkan nilai pada setiap kriteria dari calon siswa baru. Sehingga metode ELECTRE menghasilkan kelompok calon siswa baru yang diterima dan kelompok calon siswa baru yang tidak diterima. Penelitian ini

dilanjutkan dengan metode *Weighted Product* karena metode ini dapat mencari alternatif terbesar peminatan dengan menggunakan data kelompok calon siswa baru yang diterima berdasarkan perhitungan ELECTRE, sehingga calon siswa baru yang diterima dapat dimasukkan ke dalam kelas peminatan IPA atau IPS. Penelitian ini menggunakan metode ELECTRE karena dapat mengeliminasi alternatif yang kurang sesuai dengan menggunakan konsep *outrangking* [3], dapat sedangkan metode *Weighted Product* dipilih karena dapat memilih alternatif terbaik [8].

1.2 Rumusan Masalah

Rumusan masalah yang diperoleh berdasarkan permasalahan diatas sebagai berikut :

1. Bagaimana merancang Sistem Pendukung Keputusan Seleksi Penerimaan Siswa Baru SMA BSS Kota Malang?
2. Bagaimana mengimplementasikan metode ELECTRE - *Weighted Product* pada seleksi penerimaan siswa baru SMA BSS Kota Malang?
3. Bagaimana pengujian Sistem Pendukung Keputusan Seleksi Penerimaan Siswa Baru SMA BSS Kota Malang menggunakan metode ELECTRE - *Weighted Product* ?

1.3 Batasan Masalah

Adapun batasan-batasan masalah dalam penelitian ini adalah :

1. Data yang digunakan sebagai dasar pengambilan keputusan dalam penelitian ini diperoleh dari SMA Brawijaya *Smart School* (SMA BSS) Kota Malang.
2. Kriteria dan pembobotan yang digunakan sebagai dasar pengambilan keputusan dalam penelitian ini merupakan ketentuan yang dibuat oleh pihak SMA Brawijaya Smart School Kota Malang.
3. Kriteria dan pembobotan yang digunakan sebagai dasar pengambilan keputusan merupakan kriteria dan pembobotan yang digunakan dalam PPDB tahun ajaran 2014/2015 untuk jalur tes.
4. Data yang digunakan sebagai dasar pengambilan keputusan merupakan data PPDB tahun ajaran 2014/2015 yang diterima melalui jalur tes.

5. Jumlah data yang digunakan sebagai dasar pengambilan keputusan adalah sejumlah 32 data calon calon siswa baru yang diperoleh dari SMA Brawijaya Smart School Kota Malang.
6. Kelas peminatan yang digunakan adalah IPA dan IPS.
7. Metode ELECTRE digunakan untuk menghasilkan alternatif siswa baru yang diterima dan yang tidak diterima di SMA BSS sedangkan Metode *Weighted Product* digunakan untuk menentukan peminatan siswa baru yang telah diterima berdasarkan metode ELECTRE.
8. *Output* yang dihasilkan adalah alternatif siswa baru yang diterima dan penempatan kelas peminatan.
9. Implementasi sistem akan menggunakan bahasa pemrograman *C#* dengan menggunakan *Database Management System* MySQl, dan *Microsoft Windows* 7 sebagai sistem operasinya
10. Sistem ini hanya memberikan alternatif dan urutan prioritas siswa baru yang diterima dan kelas peminatan berdasarkan kriteria yang ada. Hasil akhir penerimaan siswa baru tetap dikembalikan kepada pihak SMA BSS.

1.4 Tujuan

Berdasarkan permasalahan yang ada, maka tujuan yang ingin dicapai dari penelitian ini adalah untuk membangun sebuah Sistem Pendukung Keputusan Seleksi Penerimaan Siswa Baru dengan menggunakan metode ELECTRE - *Weighted Product* serta mengukur seberapa besar tingkat akurasi dari sistem tersebut.

1.5 Manfaat

Manfaat dari penelitian ini sebagai berikut :

1. Memberikan kemudahan dan efisiensi dalam proses seleksi penerimaan siswa baru, yang biasanya masih dilakukan dengan cara manual kepada SMA Brawijaya *Smart School*

1.6 Sistematika Penulisan

Pembuatan tugas akhir ini dilakukan dengan sistematika penulisan sebagai berikut:

1. BAB I PENDAHULUAN
2. Berisi latar belakang, rumusan masalah, tujuan, batasan masalah, dan manfaat, serta sistematika penulisan skripsi.
3. BAB II TINJAUAN PUSTAKA

   Berisi teori tentang kajian pustaka yang berhubungan dengan penelitian skripsi ini serta teori dasar tentang SMA Brawijaya *Smart School* (SMA BSS), Sistem Pendukung Keputusan, Metode ELECTRE, Metode *Weighted Product* (WP), dan teori-teori lainnya yang berhubungan dengan skripsi ini.
4. BAB III METODOLOGI PENELITIAN

   Berisi langkah-langkah dan algoritma-algoritma yang digunakan dalam pembuatan sistem pendukung keputusan penerimaan baru SMA BSS dengan menggunakan metode ELECTRE - *Weighted Product*.
5. BAB IV PERANCANGAN

   Berisi analisa kebutuhan sert perancangan sistem pendukung keputusan, mulai dari perancangan model hingga *user interface*.
6. BAB IV IMPLEMENTASI

   Berisi tentang penjelasan implementasi penerimaan siswa baru dengan metode ELECTRE - *Weighted Product*, bagaimana *user interface* sistem, dan *source code* untuk mengembangkan sistem ini.
7. BAB V PENGUJIAN DAN ANALISIS

   Berisi tentang penjelasan proses pengujian dan hasil pengujian dari sistem pendukung keputusan penerimaan siswa baru SMA Brawijaya *Smart School* dengan metode ELECTRE - *Weighted Product*, serta analisis dari pengujian tersebut.
8. BAB VI PENUTUP

   Berisi kesimpulan yang diperoleh dari hasil pengujian dan saran-saran untuk pengembangan.

BAB II
TINJAUAN PUSTAKA

Bab ini berisi pembahasan tentang kajian pustaka dan dasar teori yang berhubungan dengan penelitian. Kajian pustaka dilakukan terhadap hasil penelitian yang dilakukan oleh Andi Rizki [22], Riza Alfita [4], Anak Agung Alit Karnajaya [15], Ingot Seen Sianturi[26]. Dasar teori yang dipergunakan untuk mendukung penelitian ini antara lain Sistem Pendukung Keputusan, SMA Brawijaya *Smart School* (SMA BSS), *Elimination Et Choix Traduisant la Realite* (ELECTRE), *Weighted Product* (WP), Bahasa Pemrograman C#, *Database Management System* MySQL, *Data Flow Diagram* (DFD), *Entity-Relationship Diagram*, Teknik Pemberian Bobot, *Precision* dan *Recall*, serta Akurasi.

2.1 Kajian Pustaka

Berdasarkan judul skripsi yang dibahas, penulis menemukan beberapa hasil penelitian yang relevan untuk mendukung penelitian dalam skripsi ini. Beberapa penelitian lain yang telah dilakukan dengan menggunakan metode ELECTRE - *Weighted Product* ditunjukkan pada Tabel 2.1.

Penelitian pertama dilakukan oleh Andi Rizqi pada tahun 2009. Penelitian yang berjudul Sistem Pendukung Keputusan Untuk Pemilihan Bahan Dasar Obat Alternatif Dengan Metode ELECTRE dan TOPSIS ini bertujuan untuk mencari prioritas bahan dasar obat alternatif. Sistem yang dikembangkan bersifat *customize* dimana informasi mengenai khasiat, bahan dasar obat berupa tanaman, efek samping, kriteria, pembobotan dan prioritas campuran bahan dapat dimasukkan sendiri oleh *user*. Jenis masukan sistem ada tiga, yakni jenis penyakit, nilai bobot tiap kriteria, dan nilai kriteria tiap alternatif. Pembobotan diisi oleh *user* berdasarkan keadaan, yaitu kondisi tubuh serta kondisi lingkungan yang *user* rasakan. Sistem ini melakukan perhitungan dengan menggunakan dua metode, yakni metode ELECTRE dan TOPSIS. Metode ELECTRE dan TOPSIS bekerja secara independen, dimana kedua metode tersebut sama-sama menghasilkan nama tanaman alternatif. Metode ELECTRE digunakan untuk mencari hasil akhir berupa

urutan yang lebih diprioritaskan untuk dipilih sebagai tanaman alternatif obat bagi penderita. Metode TOPSIS digunakan untuk mengurutkan prioritas bahan obat alternatif. Hasil akhir dari sistem ini berupa urutan prioritas untuk bahan dasar obat alternatif untuk *user* berdasarkan kriteria yang telah ditetapkan oleh *user* [22].

Penelitan lainnya yang berkaitan dengan sistem pendukung keputusan juga pernah dilakukan oleh Anak Agung Alit Karnajaya pada tahun 2010. Penelitian yang berjudul Rancang Bangun Sistem Pendukung Keputusan Pemilihan Sepeda Motor Honda Menggunakan Metode ELECTRE Berbasis WEB (Studi Kasus di PT. Lumenindo Gilang Cahaya) ini bertujuan untuk membantu calon pembeli sepada motor Honda untuk memilih sepeda motor yang diinginkan sesuai dengan kriteria-kriteria yang ditentukan oleh calon pembeli itu sendiri. Variabel yang digunakan adalah variable alternatif, kriteria, jenis kepentingan, bobot kepentingan, dan bobot kriteria. Sedangkan untuk beberapa contoh kriteria yang digunakan dalam sistem ini antara lain adalah harga, besar CC, garansi, bagasi, dan jenis velg. *User* akan memilih data-data kriteria yang diinginkan beserta jenis kepentingannya. *User* dapat memilih bobot untuk masing-masing alternatif kriteria yang telah dieliminasi sebelumnya. Metode yang digunakan dalam penelitian ini adalah metode ELECTRE. Proses perhitungan metode ELECTRE dimulai dengan pemilihan data-data kriteria yang diinginkan beserta jenis kepentingannya serta pemilihan data-data alternatif yang telah dieliminasi dari pemilihan kriteria sebelum akan berjalan setelah dilakukan pembobotan. Proses perhitungan metode ELECTRE akan berjalan setelah dilakukan pembobotan. Hasil dari sistem ini menunjukkan bahwa metode ELECTRE dapat diimplementasikan untuk pemilihan sepeda motor Honda, dimana metode ELECTRE dapat melakukan perhitungan sesuai dengan kriteria yang diinginkan oleh *user* [15].

Tabel 2.1 Kajian Pustaka

No	Judul	Obyek (Input)	Metode (Proses)	Hasil (Output)
1.	Sistem Pendukung Keputusan Untuk Pemilihan Bahan Dasar Obat Alternatif Dengan Metode ELECTRE dan TOPSIS [22]	Bahan dasar obat alternatif Input : - Jenis penyakit - Nilai Bobot tiap kriteria - Nilai kriteria tiap alternatif	Metode ELECTRE dan TOPSIS Langkah-langkah: 1. Normalisasi matrik keputusan. 2. Pembobotan pada matrik normalisasi. 3. Menentukan *concordance* dan *disconcordance set* dan hitung matriksnya. 4. Menentukan matrik *concordance* dan *disconcordance*. 5. Menentukan matrik dominan *concordance* dan *disconcordance*. 6. Menentukan *aggregate dominance matrix* 7. Normalisasi matriks keputusan 8. Pembobotan pada matriks ternormalisasi 9. Menentukan solusi ideal positif dan solusi ideal negatif. 10. Menghitung *Separation Measure*. 11. Menghitung kedekatan relatif dengan solusi ideal.	Urutan prioritas untuk bahan dasar obat alternatif yang terangking
2	Rancang Bangun Sistem Pendukung Keputusan Pemilihan Sepeda Motor Honda Menggunakan Metode	Pemilihan motor Honda. Input : - Harga	Metode ELECTRE Langkah-langkah: 1. Pemilihan data-data kriteria yang diinginkan beserta jenis kepentingannya.	

	ELECTRE Berbasis WEB (Studi Kasus di PT. Lumenindo Gilang Cahaya) [15]	- Besar CC - Garansi - Bagasi - Jenis Velg	2. Pemilihan data-data alternatif yang telah dieliminasi dari pemilihan kriteria sebelumnya. 3. Penginputan bobot untuk masing-masing kriteria. 4. Normalisasi matrik keputusan. 5. Pembobotan pada matrik normalisasi. 6. Menentukan *concordance* dan *disconcordance set* dan hitung matriksnya. 7. Menentukan matrik dominan *concordance* dan *disconcordance*. 8. Menentukan *aggregate dominance matrix* 9. Menampilkan urutan prioritas.	Urutan prioritas sepeda motor.
3	Sistem Pendukung Keputusan Pemilihan Jurusan Dengan Metode *Weighted Product* Untuk Siswa-Siswi SMA Negeri 1 SALE [2]	Penjurusan siswa SMA Input : - Nilai ujian nasional mata pelajaran IPA - Nilai ujian nasional mata pelajaran IPS - Nilai tes kompetensi IPA - Nilai tes kompetensi IPS	Metode *Weighted Product* Langkah-langkah: 1. Penentuan bobot preferensi dari setiap kriteria (W) 2. Menghitung vektor S 3. Menentukan nilai vektor V	Prioritas penjurusan IPA/IPS

		– Waktu pengerjaan tes kompetensi		
4	Perancangan Sistem Pendukung Keputusan Penentuan Prioritas Produk Unggulan Daerah Menggunakan Metode *Weighted Product* (WP) [4]	Produk-produk unggulan daerah. Input : – Omset – Tenaga kerja – Target pasar – Asal bahan baku – Teknologi – Spesifikasi kekhasan – Kuantitas Bahan baku	Metode *Weighted Product* Langkah-langkah: 1. Memasukkan kriteria produk unggulan daerah 2. Pembobotan terhadap data 3. Menghitung vektor S 4. Menentukan nilai vektor V	Prioritas produk ungulan dengan nilai tertinggi.
5	Sistem Pendukung Keputusan Untuk Menentukan Pemilihan Jurusan Siswa Dengan Menggunakan Metode *Weighted Product* (Studi Kasus: SMA Swasta HKBP Doloksanggul) [26]	Penjurusan Siswa SMA Input: – Nilai Fisika – Nilai Biologi – Nilai Kimia – Nilai Matematika – Nilai Ranking – Nilai Ekonomi – Nilai Sejarah – Nilai Geografi – Nilai Sosiologi – Nilai Ranking	Metode *Weighted Product* Langkah-langkah: 1. Memasukkan nilai IPA dan IPS siswa 2. Pembobotan terhadap data 3. Menghitung vektor S alternatif IPA dan alternatif IPS 4. Menentukan nilai vektor V alternatif IPA dan alternatif IPS	Prioritas penjurusan IPA/IPS

6	Sistem Pendukung Keputusan Seleksi Penerimaan Siswa Baru Menggunakan Metode ELECTRE - *Weighted Product* (Studi Kasus: SMA Brawijaya *Smart School* Kota Malang)	Penerimaan siswa SMA Input Seleksi: - Nilai tes tulis - Nilai rata-rata rapor - Nilai wawancara siswa dan orang tua - Keminatan - Nilai tes keminatan - Psikotes - Minat Anak - Minat orang tua - Nilai rapor IPA - Nilai rapor IPS - Rekomendasi BK	Metode ELECTRE Langkah-langkah: 1. Pemilihan data-data kriteria yang diinginkan beserta jenis kepentingannya. 2. Pemilihan data-data alternatif yang telah dieliminasi dari pemilihan kriteria sebelumnya. 3. Normalisasi matrik keputusan. 4. Pembobotan pada matrik normalisasi. 5. Menentukan *concordance* dan *disconcordance set* dan hitung matriksnya. 6. Menentukan matrik dominan *concordance* dan *disconcordance*. 7. Menentukan *aggregate dominance matrix* 8. Eliminasi alternatif yang *less favourable* (kurang menguntungkan) Metode *Weighted Product* Langkah-langkah: 1. Mencari data hasil seleksi siswa yang diterima berdasarkan proses perhitungan Electre 2. Pembobotan terhadap data 3. Menghitung vektor S 4. Menentukan nilai vektorV	Alternatif siswa yang diterima di SMA Brawijaya *Smart School* beserta keminatan setiap siswa baru.

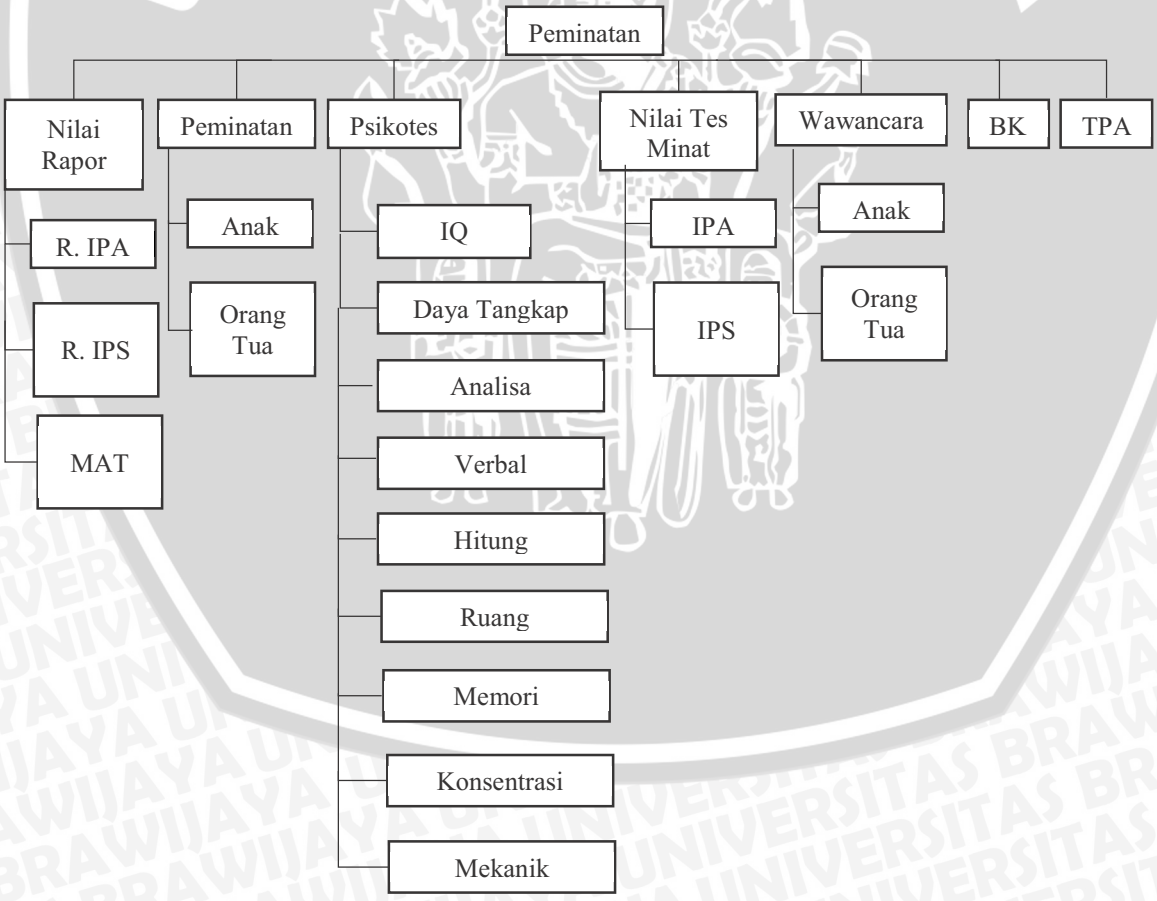
Sumber: [22][15][2][4][26]

Weighted Product juga pernah digunakan oleh Riza Alfita dalam penelitiannya yang berjudul "Perancangan Sistem Pendukung Keputusan Penentuan Prioritas Produk Unggulan Daerah Menggunakan Metode *Weighted Product* (WP)". Metode *Weighted Product* digunakan untuk mencari prioritas produk unggulan daerah berdasarkan kriteria-kriteria tertentu. Kriteria yang digunakan adalah omset, tenaga kerja, target pasar, asal bahan baku, teknologi, spesifikasi kekhasan, dan kuantitas Bahan baku. Aplikasi akan dimulai dengan menginputkan kriteria dan subkriteria yang dibagi dalam dua bagian yaitu kriteria yang bernilai keuntungan dan biaya, dan setelah itu dilanjutkan dengan perhitungan vektor S dan vektor V. Sistem akan melakukan penskala-an berdasarkan nilai interval masing-masing kelompok (sub sektor) dengan kisaran nilai dari 1 sampai 6. Sementara untuk data yang bukan berupa angka, penskala-an dilakukan dengan sistem strata. Masing-masing kriteria (variabel) memiliki bobot yang berbeda-beda disesuaikan dengan tingkat sumbangan kriteria terhadap produk unggulan. Hasil akhir dari sistem ini adalah nilai intensitas tertinggi yang layak disebut Produk Unggulan daerah dan yang lainnya disebut sebagai produk potensial daerah [4].

Ingot Seen Sianturi melakukan penelitian dengan judul "Sistem Pendukung Keputusan Untuk Menentukan Pemilihan Jurusan Siswa Dengan Menggunakan Metode *Weighted Product* (Studi Kasus: SMA Swasta HKBP Doloksanggul)". Metode *Weighted Product* digunakan untuk penjurusan siswa SMA berdasarkan kriteria dari alternatif IPA dan alternatif IPS. Alternatif IPA memiliki 5 kriteria, yakni nilai biologi, nilai fisika, nilai kimia, nilai matematika, dan ranking. Alternatif IPS memiliki 5 kriteria, yakni nilai ekonomi, nilai sejarah, nilai geografi, nilai sosiologi, dan ranking. Nilai bobot untuk masing-masing kriteria telah ditentukan. Metode *Weighted Product* akan menghitungan nilai dan bobot yang telah dimasukkan oleh user, mulai dari pembobotan, perhitungan vektor S, dan penentuan nilai vektor V. Hasil dari penelitian ini meunjukkan bahwa metode *Weighted Product* dapat membantu dalam mengambil keputusan untuk menentukan penjurusan siswa [22].

Berdasarkan dari paparan penelitian yang telah dilakukan, maka penulis mengusulkan penelitian yang berjudul "Sistem Pendukung Keputusan Seleksi Penerimaan Siswa Baru Menggunakan Metode ELECTRE - *Weighted Product* (Studi Kasus: SMA Brawijaya *Smart School* Kota Malang)". Penelitian ini dilakukan untuk mempermudah kinerja pihak SMA BSS dalam melakukan seleksi calon siswa baru. Penelitian ini menggunakan metode ELECTRE - *Weighted Product*. Penelitian ini akan menggunakan beberapa kriteria untuk menentukan siswa dapat diterima dan juga menentukan kelompok keminatannya. Kriteria tersebut antara lain nilai tes tulis, nilai rata-rata rapor, nilai wawancara siswa, nilai wawancara orang tua, , nilai tes keminatan, psikotes, minat anak, minat orang tua, nilai rapor IPA, nilai rapor IPS, dan rekomendasi BK.

Gambar 2.1. Kriteria Input Peminatan Seleksi SMA BSS



Sumber: [Wawancara]



Gambar 2.2. Kriteria Input Proses Penerimaan Seleksi SMA BSS
Sumber: [Wawancara]

Penelitian ini nantinya akan menggunakan metode ELECTRE - *Weighted Product*. Metode ELECTRE digunakan untuk menyeleksi alternatif siswa yang akan diterima oleh SMA BSS. Metode *Weighted Product* digunakan untuk melakukan peminatan calon siswa yang diterima di SMA BSS berdasarkan perhitungan metode ELECTRE. Penggunaan metode ELECTRE dan WP dalam penelitian ini merupakan salah satu pengembangan metode ELECTRE dan TOPSIS yang dilakukan oleh Rizqi A. Penelitian ini menggunakan metode ELECTRE karena dapat mengeliminasi alternatif yang kurang sesuai, sedangkan metode *Weighted Product* dipilih karena metode ini dapat memilih alternatif terbaik.

Langkah-langkah perhitungan dengan menggunakan metode ELECTRE - *Weighted Product* yaitu:

1. Menentukan data dan kriteria yang digunakan sesuai dengan kebutuhan
2. Menghitung normalisasi matriks keputusan
3. Pembobotan pada matriks ternormalisasi
4. Menentukan himpunan *concordace* dan *discordance*
5. Menentukan matriks *concordace* dan *discordance*
6. Menentukan matriks dominan *concordace* dan *discordance*
7. Menentukan *aggregate dominance matrix*
8. Eliminasi alternatif yang *less favourable*
9. Mencari data hasil perhitungan metode ELECTRE
10. Melakukan pembobotan

11. Menghitung nilai vektor S
12. Menghitung nilai vektor V
13. Mencari Nilai Vektor V terbesar.

Setelah semua proses perhitungan selesai dilakukan maka hasil akhir yang diharapkan dari penelitian ini yaitu rekomendasi calon siswa baru yang diterima dalam kelompok peminatan tertentu, yang mana hasil tersebut dapat digunakan sebagai bahan pertimbangan pihak SMA BSS dalam proses seleksi calon siswa baru baru periode selanjutnya.

2.2 SMA Brawijaya *Smart School* (SMA BSS)

Brawijaya *Smart Schools* adalah institusi pendidikan yang menyelenggarakan pendidikan dan pengajaran ditingkat PAUD (Penitipan anak, *playgroup*, dan TK), SD, SMP, SMA dan *University Foundation*. Brawijaya *Smart School* merupakan lembaga pendidikan yang dikelola oleh Unit Pelaksana Teknis (UPT) dan didirikan berdasarkan surat keputusan rektor Universitas Brawijaya [6]. SMA Brawijaya *Smart School* adalah Sekolah Menengah Atas Nasional dalam naungan Universitas Brawijaya, yang dipersiapkan menjadi Sekolah Bertaraf International (SBI) dan bertekad menghasilkan lulusan yang berkualitas internasional yang mampu bersaing dan berkolaborasi secara global [6]. Struktur organisasi dapat dilihat pada Gambar 2.3

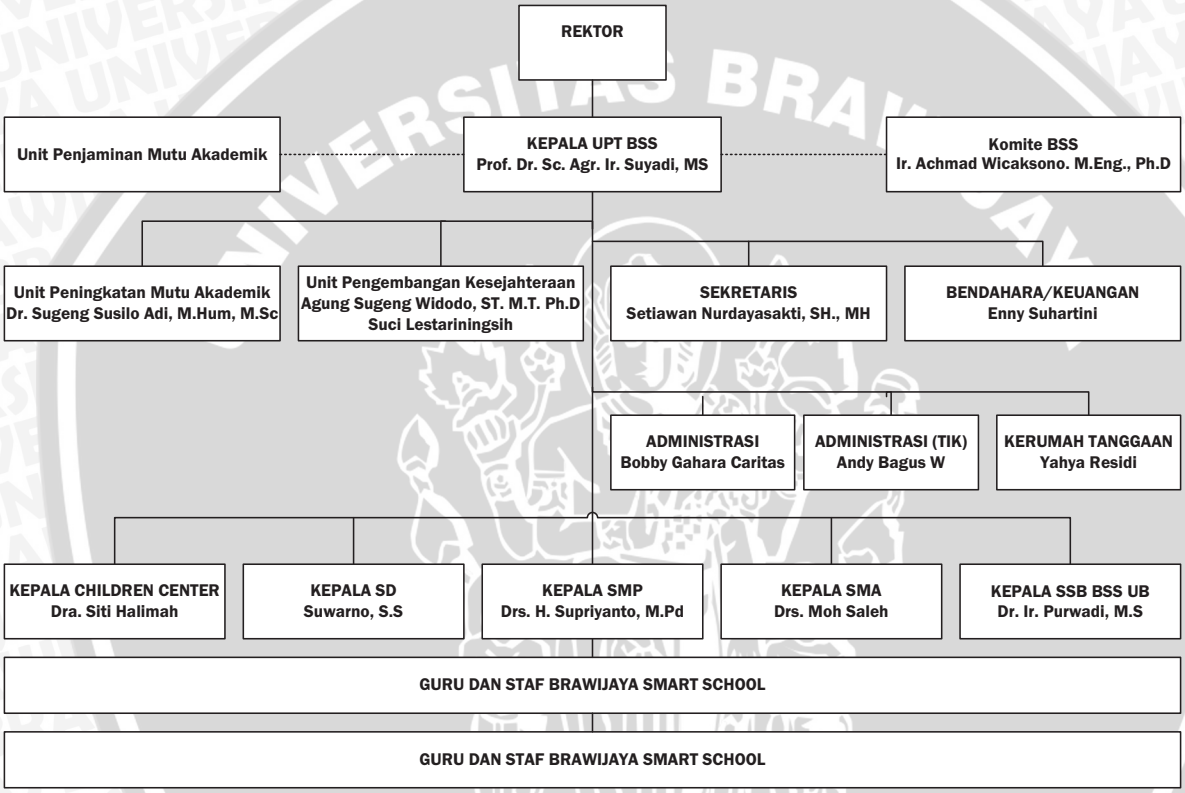
Adapun Misi dari SMA Brawijaya *Smart School* adalah [6] :

- Mewujudkan manusia Indonesia yang cerdas (*smart*), memiliki standar moral yang tinggi dan kompetitif secara internasional, mampu bersaing dan berkolaborasi secara global.

Tujuan dari SMA Brawijaya *Smart School* adalah [6] :

- Menghasilkan lulusan yang berkelas nasional dan internasional sekaligus lulusan berkelas nasional sesuai dengan yang telah dirumuskan dalam UU No. 20/2003 dan dijabarkan dalam PP/2005 yaitu pendidikan menengah umum yang memiliki tujuan untuk meningkatkan kecerdasan, pengetahuan, kepribadian, akhlak mulia, keterampilan untuk hidup mandiri (*life skill*) dan mengikuti pendidikan yang lebih tinggi

- Menghasilkan lulusan yang berprestasi dalam materi pelajaran sehingga :
  1. Dapat diterima di perguruan tinggi ternama yang ada di Indonesia
  2. Lulus ujian sertifikasi internasional dari lembaga pendidik internasional (IBO, IMI atau *Cambridge*)
  3. Lulusan yang dapat memperpendek masa kuliahnya (*acceleration*) di Perguruan Tinggi khususnya Universitas Brawijaya



Gambar 2.3. Struktur Organisasi Brawijaya Smart School
Sumber : [6]

2.2.1 Seleksi Penerimaan Siswa Baru

SMA BSS menggunakan kurikulum 2013, dimana proses seleksi terdiri dari proses penerimaan dan proses peminatan. Pada saat ini penerimaan siswa baru di SMA BSS masih menggunakan sistem manual seperti perhitungan menggunakan *Microsoft Excel*. Seleksi penerimaan siswa baru di SMA BSS menggunakan dua jalur, yakni jalur prestasi dan jalur tes. Jalur prestasi merupakan jalur masuk yang hanya menggunakan nilai rapor, dimana nilai rapor harus memiliki nilai minimal 80 untuk semua mata pelajaran dalam rapor mulai semester I-V. Jalur tes

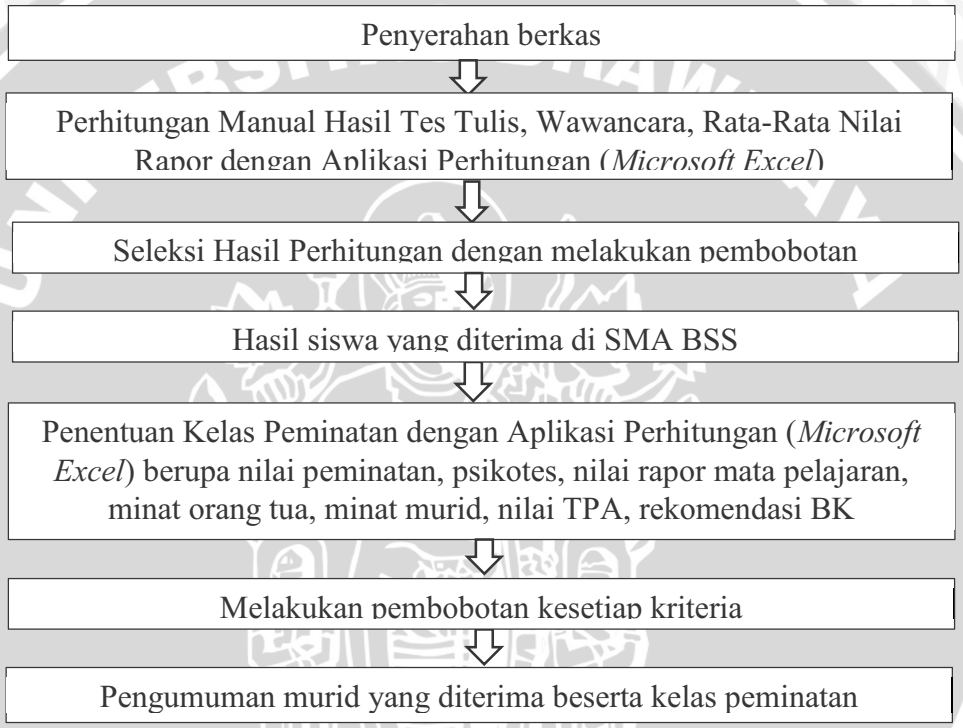
merupakan jalur masuk yang menggunakaan nilai tes yang meliputi wawancara orang tua dan calon siswa baru, rata-rata nilai rapor, dan nilai tes potensial akademik (TPA). Pada tahun pelajaran 2014/2015, pagu SMA BSS terdiri dari 4 kelas IPA dan 2 kelas IPS, dimana terdiri dari 30% calon siswa baru jalur prestasi dan 70% calon siswa baru jalur tes. Jika pagu belum terpenuhi, maka SMA akan membuka kembali pendaftaran [wawancara].

Seleksi dilakukan secara manual dengan akumulasi nilai setiap calon calon siswa baru yang ada. SMA memberikan bobot yang berbeda-beda untuk setiap kriteria penilian.Calon calon siswa baru yang dinyatakan diterima adalah calon siswa baru yang menduduki beberapa (sesuai dengan kuota yang disepakati) peringkat tertinggi [wawancara]. Tahapan proses penerimaan siswa baru dengan jalur reguler di SMA BSS diilustrasikan dalam blok diagram seperti pada Gambar 2.4.



Gambar 2.4 Diagram Blok Proses Penerimaan Siswa Baru
Sumber: [Wawancara]

Sesuai dengan kurikulum 2013, maka rangkaian proses penerimaan calon siswa baru baru dilanjutkan dengan proses mengelompokkan calon siswa baru yang telah diterima dalam tahap seleksi ke dalam kelompok peminatan (IPA atau IPS). Penentuan Kelompok peminatan ditentukan berdasarkan hasil dari tes peminatan, psikotes, nilai rapor mata pelajaran (Matematika, IPA dan IPS), rekomendasi BK di SMP [24]. Tahapan proses penilaian peminatan siswa baru jalur prestasi dengan di SMA BSS diilustrasikan dalam blok diagram seperti pada Gambar 2.5.



Gambar 2.5 Diagram Blok Alur Penilaian
Sumber : [wawancara]

Perhitungan manual yang dilakukan oleh pihak BSS dengan menggunakan salah satu aplikasi perhitungan (*Ms. Excel*). Perhitungan manual yang dilakukan oleh pihak SMA BSS menggunakan rumus sebagai berikut.

Nilai total = (nilai rata-rata rapor * 30%) + (nilai tes tulis * 40%) + (nilai wawancara calon peserta * 15%) + (nilai wawancara orang tua * 15%)

Contoh perhitungan manual yang dilakukan oleh pihak SMA BSS dapat dilihat melalui paparan di bawah ini:

Tabel 2.2 Contoh Perhitungan Manual

Kriteria	Nilai			
	A01	Nilai*Bobot	A12	Nilai*Bobot
Rata-rata rapor (30)	80.8	2432	78	2340
Nilai Tes (40)	90.67	3626.8	64	2560
Wawancara Siswa (15)	86.0	1290	83	1245
Wawancara Orang Tua (15)	45.0	675	43	645
Total	**8014.3**		6790	

Sumber: [Wawancara]

Tabel 2.3 Perhitungan Manual SMA BSS

No	**No Reg**	**Rata2 Rapor**	**30%**	**Test Tulis**	**40%**	**Wawancara**				**Total**
						Calon Siswa	**15%**	**Orang tua**	**15%**	
1	A01	80.8	2423	90.67	3626.8	86	1290	45	675	8014.3
2	A02	84.8	2543	84	3360	86	1290	41	615	7807.5
3	A03	85.5	2565	81.33	3253.2	85	1275	45	675	7768.2
4	A04	81.5	2445	82.67	3306.8	80	1200	45	675	7626.8
5	A05	81.0	2429	81.33	3253.2	78	1170	45	675	7527.38
6	A06	80.4	2412	78.67	3146.8	81.25	1219	45	675	7452.55
7	A07	83.0	2490	80	3200	70	1050	39	585	7325
8	A08	80.6	2418	73.33	2933.2	75	1125	44	660	7136.2
9*	A09	84	2520	66.67	2666.8	75	1125	43	645	6956.8
10	A10	82.21	2466	68	2720	70	1050	41	615	6851.18
11	A11	91.68	2750	62.67	2506.8	62	930	42	630	6817.2
12	A12	78	2340	64	2560	83	1245	43	645	6790
13*	A13	82.4	2472	65.33	2613.2	71	1065	39	585	6735.2
14	A14	75.4	2262	66.67	2666.8	72	1080	45	675	6683.8
15	A15	78.2	2346	66.67	2666.8	68	1020	38	570	6602.8

Sumber: [Wawancara]

Setelah dilakukan perangkingan seperti yang terlihat pada Tabel 2.3, maka selanjutnya dilakukan proses seleksi. Hasil seleksi yang didapatkan antara lain:

- Calon calon siswa baru nomor A01-A08 adalah siswa yang lolos dalam tahap seleksi, karena nilai total yang didapatkan calon siswa baru berada dalam rentang 6900-8000 atau >8000.
- Calon calon siswa baru nomor A09-A15 kecuali A13 adalah siswa yang tidak lolos dalam tahap seleksi, karena nilai total yang didapatkan calon siswa baru <6900. Pengecualian untuk calon peserta nomor 9 meskipun nilai total diatas batas minimal pihak sekolah memutuskan untuk tidak meloloskan dalam tahap seleksi. Hal ini terjadi karena terdapat kebijakan lain yang dimiliki sekolah yang dilihat dari berbagai aspek, misalkan saja kepribadian calon calon siswa baru tersebut buruk (merokok, tidak sopan, dll.), nilai wawancara, dan lain sebagainya.
- Calon calon siswa baru nomor A13 adalah siswa yang lolos dalam tahap seleksi meskipun nilai total ada dibawah batas. Hal ini terjadi karena terdapat kebijakan lain yang dimiliki sekolah yang dilihat dari berbagai aspek, misalkan saja prestasi akademik maupun non akademik yang dimiliki banyak, kepribadian calon calon siswa baru baik, dan lain sebagainya.

2.2.2 Kriteria

Seleksi penerimaan siswa baru di SMA BSS dilakukan dengan menggunakan beberapa kriteria yang sudah menjadi kecepakatan dengan tujuan untuk tolak ukur kemampuan calon siswa baru. Kriteria yang digunakan berbeda dengan seleksi penerimaan siswa baru SMP BSS, karena seleksi penerimaan siswa SMA BSS memiliki kriteria tambahan untuk proses peminatan calon siswa. Berikut ini merupakan beberapa kriteria yang digunakan pada seleksi penerimaan siswa baru [wawancara] :

1. Nilai rata-rata rapor SMP Semester I-V

   Nilai rata-rata rapor SMP dari semester I-V digunakan baik calon siswa yang menggunakan jalur prestasi maupun jalur tes. Calon siswa harus memiliki nilai minimal 80 untuk semua mata pelajaran dalam rapor mulai semester 1-V. Nilai rapor digunakan untuk mencari nilai rata-rata rapor per semester serta nilai rata-rata rapor dari semester I-V. Nilai rata-rata rapor

semester I-V akan digunakan sebagai kriteria penerimaan calon siswa baru SMA BSS

2. Nilai Wawancara Calon Murid

   Penilaian hasil wawancara akan ditetapkan oleh pihak SMA BSS. Wawancara dilakukan untuk mengetahui kepribadian calon siswa baru. Nilai wawancara dilihat dari beberapa aspek, dan penilaian wawancara diserahkan sepenuhnya kepada pihak panitia penerimaan calon siswa baru SMA BSS

3. Nilai Wawancara Orangtua Calon Murid

   Penilaian hasil wawancara akan ditetapkan oleh pihak SMA BSS. Wawancara orangtua calon murid berbeda dengan wawancara calon murid. Wawancara orangtua lebih ditekankan pada hal administrasi. Nilai wawancara dilihat dari beberapa aspek, dan penilaian wawancara diserahkan sepenuhnya kepada pihak panitia penerimaan calon siswa baru SMA BSS

4. Nilai Tes Tulis

   Tes tulis terdiri dari Tes Potensial Akademik dan Bahasa Inggris, Kedua nilai tersebut akan digabung menjadi satu hasil nilai.

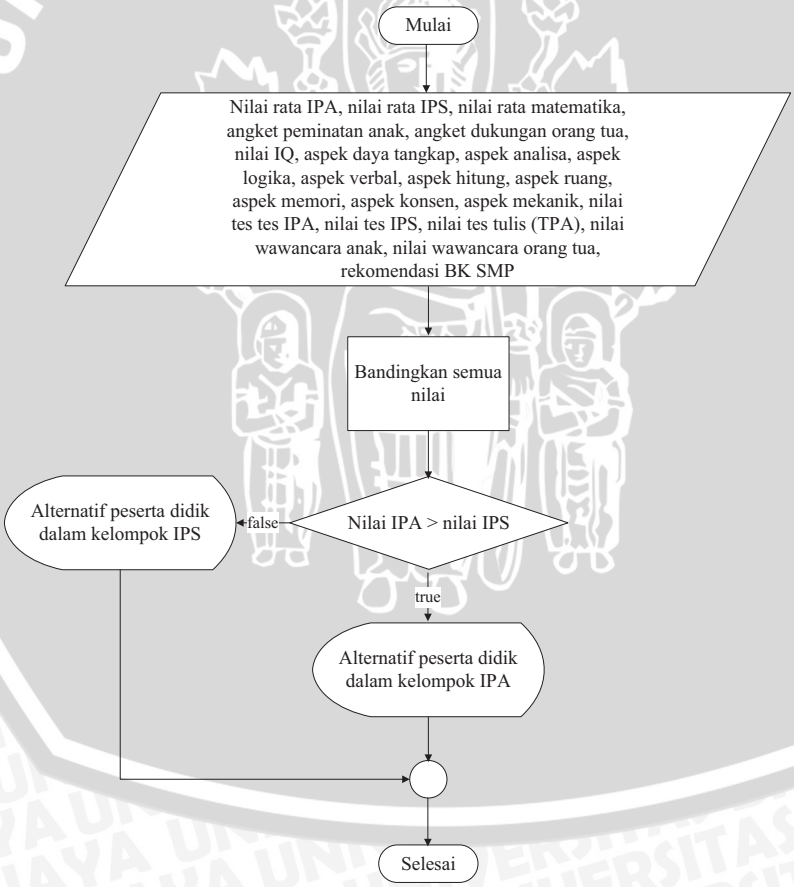
2.2.3 Peminatan

Tahap ini menentukan siswa yang telah diterima masuk ke dalam kelompok peminatan IPA atau IPS. Proses peminatan yang ada di SMA BSS dilakukan ketika calon siswa baru telah lolos dalam tahap seleksi. Proses peminatan calon siswa baru dilakukan oleh Guru BK SMA BSS, yang terbagi menjadi beberapa tahapan yaitu

- Tahapan pengumpulan data merupakan tahapan yang bertujuan untuk mengumpulkan keseluruhan data yang akan digunakan dalam proses peminatan.
- Tahapan identifikasi merupakan tahapan kedua yang dilakukan untuk mengetahui minat dan kemampuan calon siswa baru. Tahapan yang dilakukan adalah semua nilai-nilai yang terkait dengan mata pelajaran dalam kelompok peminatan yaitu mata pelajaran IPA dan IPS dengan kecocokan minat masing-masing calon siswa baru.

- Tahapan penempatan merupakan tahapan akhir yang bertujuan untuk memberikan hasil dari proses penempatan calon siswa baru. Hasil yang didapat antara lain: jika semua nilai yang terkait dengan mata pelajaran IPA baik, maka calon siswa baru tersebut akan dimasukkan dalam kelompok peminatan IPA. Jika semua nilai yang terkait dengan mata pelajaran IPS baik, maka calon siswa baru tersebut akan dimasukkan dalam kelompok peminatan IPS. Jika nilai IPA atau IPS sama-sama baik maka pihak sekolah akan melakukan pertimbangan-pertimbangan lainnya untuk menempatkan calon siswa baru dalam kelompok peminatan tertentu.

Proses peminatan yang dilakukan oleh pihak SMA BSS dapat diilustrasikan dengan menggunakan diagram alir seperti yang terlihat pada Gambar 2.6.



Gambar 2.6 Diagram Alir Proses Peminatan SMA BSS
Sumber: [Wawancara]

Adapun kriteria yang digunakan sebagai dasar penilaian antara lain:

1. Nilai rata-rata rapor

   Nilai rapor SMP dari semester I-V digunakan baik calon siswa yang menggunakan jalur prestasi maupun jalur tes. Nilai rapor digunakan untuk mencari nilai rata-rata rapor per semester serta nilai rata-rata rapor dari semester I-V.

2. Nilai rata-rata Matematika

   Nilai rata-rata rapor SMP yang didapatkan dari nilai mata pelajaran matematika dari semester I-V dengan rentang nilai 0-100.

3. Nilai rata-rata IPA

   Nilai rata-rata rapor SMP yang didapatkan dari nilai mata pelajaran IPA dari semester I-V dengan rentang nilai 0-100.

4. Nilai rata-rata IPS

   Nilai rata-rata rapor SMP yang didapatkan dari nilai mata pelajaran IPS dari semester I-V dengan rentang nilai 0-100.

5. Nilai tes tulis

   Nilai tes tulis adalah nilai tes potensial akademik yang dilakukan oleh pihak SMA BSS dengan rentang nilai 0-100.

6. Wawancara calon siswa baru

   Penilaian hasil wawancara akan ditetapkan oleh pihak SMA BSS. Wawancara dilakukan untuk mengetahui minat calon siswa baru lebih dalam.

7. Wawancara orang tua

   Penilaian hasil wawancara akan ditetapkan oleh pihak SMA BSS.. Nilai wawancara dilihat dari beberapa aspek, dan penilaian wawancara diserahkan sepenuhnya kepada pihak panitia penerimaan calon siswa baru SMA BSS

8. Nilai tes peminatan IPA

   Nilai tes peminatan IPA didapatkan dari *placement test*. SMA BSS melakukan *placemement test* untuk mengukur kemampuan akademik calon siswa baru. *Placement test* terdiri dari berbagai macam soal mata pelajaran yang berhungan dengan mata pelajaran ilmu pengetahuan alam dan ilmu pengetahuan sosial.

9. Nilai tes peminatan IPS

   Nilai tes peminatan IPS didapatkan dari *placement test*. SMA BSS melakukan *placemement test* untuk mengukur kemampuan akademik calon siswa baru. *Placement test* terdiri dari berbagai macam soal mata pelajaran yang berhungan dengan mata pelajaran ilmu pengetahuan alam dan ilmu pengetahuan sosial.

10. Angket calon siswa baru

    Angket calon siswa baru dimaksudkan untuk memperoleh imformasi tentang identitas dan minat dari calon siswa baru. SMA BSS dapat mengetahui keminatan calon siswa baru secara lebih mendalam dengan adanya angket calon siswa baru.

11. Angket orang tua

    Angket orang tua dimaksudkan untuk memperoleh informasi tentang identitas dan harapan peminatan belajar bagi calon siswa baru. SMA BSS dapat mengetahui dukungan orang tua terhadap minat calon siswa baru dengan adanya angket ini.

12. IQ

    Nilai IQ digunakan untuk mengetahui tingkat kecerdasaan calon siswa baru.

    Daya Tangkap

    Nilai daya tangkap digunakan untuk kemampuan analisa calon siswa baru. Nilai analisa didapatkan dari hasil psikotes yang diadakan SMA BSS. Nilai terdiri dari Tinggi (T), Cukup (C), dan Rendah (R).

13. Analisa

    Nilai analisa digunakan untuk kemampuan analisa calon siswa baru. Nilai analisa didapatkan dari hasil psikotes yang diadakan SMA BSS. Nilai terdiri dari Tinggi (T), Cukup (C), dan Rendah (R).

14. Logika

    Nilai logikadigunakan untuk kemampuan logika calon siswa baru. Nilai logika didapatkan dari hasil psikotes yang diadakan SMA BSS. Nilai terdiri dari Tinggi (T), Cukup (C), dan Rendah (R).

15. Verbal

Nilai verbal digunakan untuk kemampuan verbal calon siswa baru. Nilai verbaldidapatkan dari hasil psikotes yang diadakan SMA BSS. Nilai terdiri dari Tinggi (T), Cukup (C), dan Rendah (R).

16. Hitung

    Nilai hitung digunakan untuk kemampuan hitung calon siswa baru. Nilai hitung didapatkan dari hasil psikotes yang diadakan SMA BSS. Nilai terdiri dari Tinggi (T), Cukup (C), dan Rendah (R).

17. Ruang

    Nilai ruang digunakan untuk kemampuan ruang calon siswa baru. Nilai ruang didapatkan dari hasil psikotes yang diadakan SMA BSS. Nilai terdiri dari Tinggi (T), Cukup (C), dan Rendah (R).

18. Memori

    Nilai memori digunakan untuk kemampuan memori calon siswa baru. Nilai memorididapatkan dari hasil psikotes yang diadakan SMA BSS. Nilai terdiri dari Tinggi (T), Cukup (C), dan Rendah (R).

19. Konsentrasi

    Nilai konsentrasi digunakan untuk kemampuan konsentrasi calon siswa baru. Nilai konsentrasi didapatkan dari hasil psikotes yang diadakan SMA BSS. Nilai terdiri dari Tinggi (T), Cukup (C), dan Rendah (R).

20. Mekanik

    Nilai mekanik digunakan untuk kemampuan mekanik calon siswa baru. Nilai mekanik didapatkan dari hasil psikotes yang diadakan SMA BSS. Nilai terdiri dari Tinggi (T), Cukup (C), dan Rendah (R).

21. Rekomendasi BK

    Saran bidang studi diperoleh dari rekomendasi BK SMP.

Proses peminatan calon siswa baru yang dilakukan oleh pihak BSS dengan menggunakan cara melakukan perbandingan manual terhadap nilai yang mewakili IPA dan IPS. Perbandingan manual dilakukan oleh pihak SMA BSS dilakukan sebagai berikut:

Tabel 2.4 Perbandingan Manual SMA BSS

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
A1	79.6	83.2	85.4	81	35	90	**80**	**68**	IPA	IPA	C	C	C-	C	C-	C-	C	C	C	IPA
A2	72	68.4	66.6	65	32	83	**32**	**36**	IPA	IPA	**R**	**R**	**R**	**R**	**R**	C-	C-	C-	C	IPA
A3	73	68.4	68.6	68	33	77	**42**	**24**	**IPS**	**IPS**	C-	C-	C-	**R**	**R**	C-	C	C-	C-	IPS
A4	88.2	84.4	88	90	35	90	**78**	**76**	IPA	IPA	C+	C+	C-	C+	C-	C+	C+	C	C-	IPA
A5	92.4	88.4	89.8	86	35	87	**80**	**66**	IPA	IPA	C	C	C-	C	C-	C-	C	C	C	IPA

Sumber: [Wawancara]

Keterangan Tabel

1 Alternatif
2 Rata-rata MAT SMT I-V
3 Rata-rata IPA SMT I-V
4 Rata-rata IPS SMT I-V
5 Nilai TPA
6 Wawancara Orang tua
7 Wawancara Calon siswa baru
8 Tes Peminatan IPA
9 Tes Peminatan IPS
10 Angket Calon siswa baru
11 Angket Orang Tua
12 Kemampuan Daya Tangkap
13 Kemampuan Analisa
14 Kemampuan Logika
15 Kemampuan Verbal
16 Kemampuan Hitung
17 Kemampuan Ruang
18 Kemampuan Memori
19 Kemampuan Konsentrasi
20 Kemampuan Mekanik
21 Rekomendasi BK SMP

Calon siswa baru masuk kelas peminatan IPS
Calon siswa baru masuk kelas peminatan IPA

Keterangan Alternatif:

1. A1: Peserta A1 masuk ke dalam peminatan IPA.Peserta A1 memiliki nilai rapor MAT semester 1-5 yang cukup baik. Nilai rapor IPA dan IPS dari semester 1-5 cukup berimbang. Hasil nilai dari tes peminatan menunjukan bahwa nilai peminatan IPA peserta A1 jauh lebih tinggi dari nilai peminatan IPS. Hasil psikotest peserta A1 juga menunjukkan bahwa ia memiliki kemampuan daya tangkap hingga kemampuan mekanik yang cukup dan tidak ada yang bernilai rendah. Dari hasil rekomendasi BK SMP dan wawancara, diketahui bahwa calon siswa baru dan orang tua memiliki keinginan untuk ditempatkan pada kelas peminatan IPA. Berdasarkan hasil tersebut, maka pihak SMA BSS merekomendasikan agar peserta A1 masuk ke dalam kelas peminatan IPA.
2. A2: Peserta A2 masuk ke dalam peminatan IPS. Peserta A2 memiliki nilai rapor MAT semester 1-5 yang cukup rendah apabila dibandingkan dengan peserta lainnya. Nilai rapor IPA dan IPS dari semester 1-5 cukup berimbang. Hasil nilai dari tes peminatan cukup rendah namun menunjukan bahwa nilai peminatan IPS peserta A2 lebih tinggi dari nilai peminatan IPA. Hasil psikotest peserta A2 juga menunjukkan bahwa ia memiliki kemampuan daya tangkap, analisa, logika, verbal, dan hitung yang rendah. Meskipun dari hasil rekomendasi BK SMP dan wawancara, diketahui bahwa calon siswa baru dan orang tua memiliki keinginan untuk ditempatkan pada kelas peminatan IPA, namun berdasarkan pertimbangan beberapa kriteria lainnya maka pihak SMA BSS merekomendasikan agar peserta A2 masuk ke dalam kelas peminatan IPS.
3. A3: Peserta A3 masuk ke dalam peminatan IPS. Peserta A3 memiliki nilai rapor MAT semester 1-5 yang cukup rendah apabila dibandingkan dengan peserta lainnya. Nilai rapor IPA dan IPS dari semester 1-5 cukup berimbang. Hasil nilai dari tes peminatan cukup rendah akan tetapi menunjukan bahwa nilai peminatan IPA peserta A2 lebih tinggi dari nilai peminatan IPS. Hasil psikotest peserta A3 juga menunjukkan bahwa ia memiliki kemampuan daya tangkap, analisa, logika, ruang, memori, konsentrasi, dan mekanik yang cukup namun kemampuan verbal dan kemampuan hitung peserta A3

adalah rendah. Dari hasil rekomendasi BK SMP dan wawancara, diketahui bahwa calon siswa baru dan orang tua memiliki keinginan untuk ditempatkan pada kelas peminatan IPS. Berdasarkan hasil tersebut, maka pihak SMA BSS merekomendasikan agar peserta A3 masuk ke dalam kelas peminatan IPS.

4. A4 : Peserta A4 masuk ke dalam peminatan IPA.Peserta A4 memiliki nilai rapor MAT semester 1-5 yang cukup baik. Nilai rapor IPA dan IPS dari semester 1-5 cukup berimbang. Hasil nilai dari tes peminatan cukup berimbang, namun dapat menunjukan bahwa nilai peminatan IPA peserta A4 lebih tinggi dari nilai peminatan IPS. Hasil psikotest peserta A4 juga menunjukkan bahwa ia memiliki kemampuan daya tangkap hingga kemampuan mekanik yang cukup dan tidak ada yang bernilai rendah. Dari hasil rekomendasi BK SMP dan wawancara, diketahui bahwa calon siswa baru dan orang tua memiliki keinginan untuk ditempatkan pada kelas peminatan IPA. Berdasarkan hasil tersebut, maka pihak SMA BSS merekomendasikan agar peserta A4 masuk ke dalam kelas peminatan IPA.
5. A5 : Peserta A5 masuk ke dalam peminatan IPA.Peserta A5 memiliki nilai rapor MAT semester 1-5 yang cukup baik. Nilai rapor IPA dan IPS dari semester 1-5 cukup berimbang. Hasil nilai dari tes peminatan menunjukan bahwa nilai peminatan IPA peserta A5 lebih jauh tinggi dari nilai peminatan IPS. Hasil psikotest peserta A5 juga menunjukkan bahwa ia memiliki kemampuan daya tangkap hingga kemampuan mekanik yang cukup dan tidak ada yang bernilai rendah. Dari hasil rekomendasi BK SMP dan wawancara, diketahui bahwa calon siswa baru dan orang tua memiliki keinginan untuk ditempatkan pada kelas peminatan IPA. Berdasarkan hasil tersebut, maka pihak SMA BSS merekomendasikan agar peserta A5 masuk ke dalam kelas peminatan IPA.

2.3 Sistem Pendukung Keputusan

Pada awalnya, Sistem Pendukung Keputusan didefinisikan sebagai sebuah sistem yang ditujukan untuk mendukung para manajerial mengambil sebuah keputusan dalam situasi keputusan semiterstruktur. Sistem pendukung keputusan dimaksudkan untuk menjadi tambahan bagi para pengambil keputusan utnuk memperluas kemampuan mereka, namun tidak untuk menggantikan penilian mereka. Little (1970) mendefinisikan Sistem Pendukung Keputusan sebagai sebuah model berbasis set prosedur yang digunakan untuk memproses data dan penilaian untuk membantu seorang manajer dalam mengambil sebuah keputusan. Little berpendapat bahwa agar sebuah sistem pendukung keputusan tersebut sukses, sistem harus dibuat dengan sederhana, kokoh, mudah untuk mengontrol, adaptif, lengkap berisi mengenai isu-isu penting, dan mudah untuk dikomunikasikan [27: 103].

Marko Bohanec mendefinisikan sistem pendukung keputusan adalah sebuah sistem komputer yang berbasis komputer interaktif yang dibuat untuk membantu pembuat keputusan dalam mengidentifikasi masalah dan memecahkan masalah dengan menggunakan data dan model yang ada [5]. Bonczek dkk (1980) mendefiniskan sistem pendukung keputusan sebagai sebuah sistem berbasis computer yang terdiri dari tiga komponen yang saling berinteraksi, yakni sistem bahasa (sebuah mekanisme yang menyediakan komunikasi antara *user* dan komponen-komponen sistem pendukung keputusan), sistem pengetahuan (repositori pengetahuan domain masalah yang tersimpan di sistem pendukung keputusan baik sebagai data atau prosedur-prosedur), dan sistem pengolahan masalah (penghubung antara dua komponen lainnya, berisi satu atau lebih kemampuan umum manipulasi masalah yang dibutuhkan untuk proses pengambilan keputusan) [27: 104].

2.3.1 Dasar Pengambilan Keputusan

Simon (1977) mengatakan bahwa dasar pengambilan keputusan melibatkan tiga fase utama: *intelligence*, *design*, dan *choice*. Simon kemudian menambahkan fase keempat yakni *implementation* [9:49]. Penjelasan dari keempat fase tersebut adalah sebagai berikut [27:51-70]:

1. Fase *Intelligence*

   Fase pengetahuan dalam dasar pengambilan keputusan melibatkan pada pengamatan lingkungan, baik sesekali maupun secara terus-menerus. Pengamatan lingkungan ini meliputi berbagai macam kegiatan yang bertujuan untuk mengidentifikasi situasi masalah atau peluang. Data masukan diperoleh, diproses, dan diuji dalam rangka mengidentifikasikan masalah.

2. Fase *Design*

   Tahap ini merupakan proses menemukan, mengembangkan, dan menganalisis alternatif tindakan yang bisa dilakukan. Hal ini termasuk memahami masalah dan solusi pengujian untuk berbagai kemungkinan. Tahap ini meliputi proses untuk mengerti masalah, menurunkan solusi dan menguji kelayakan solusi.

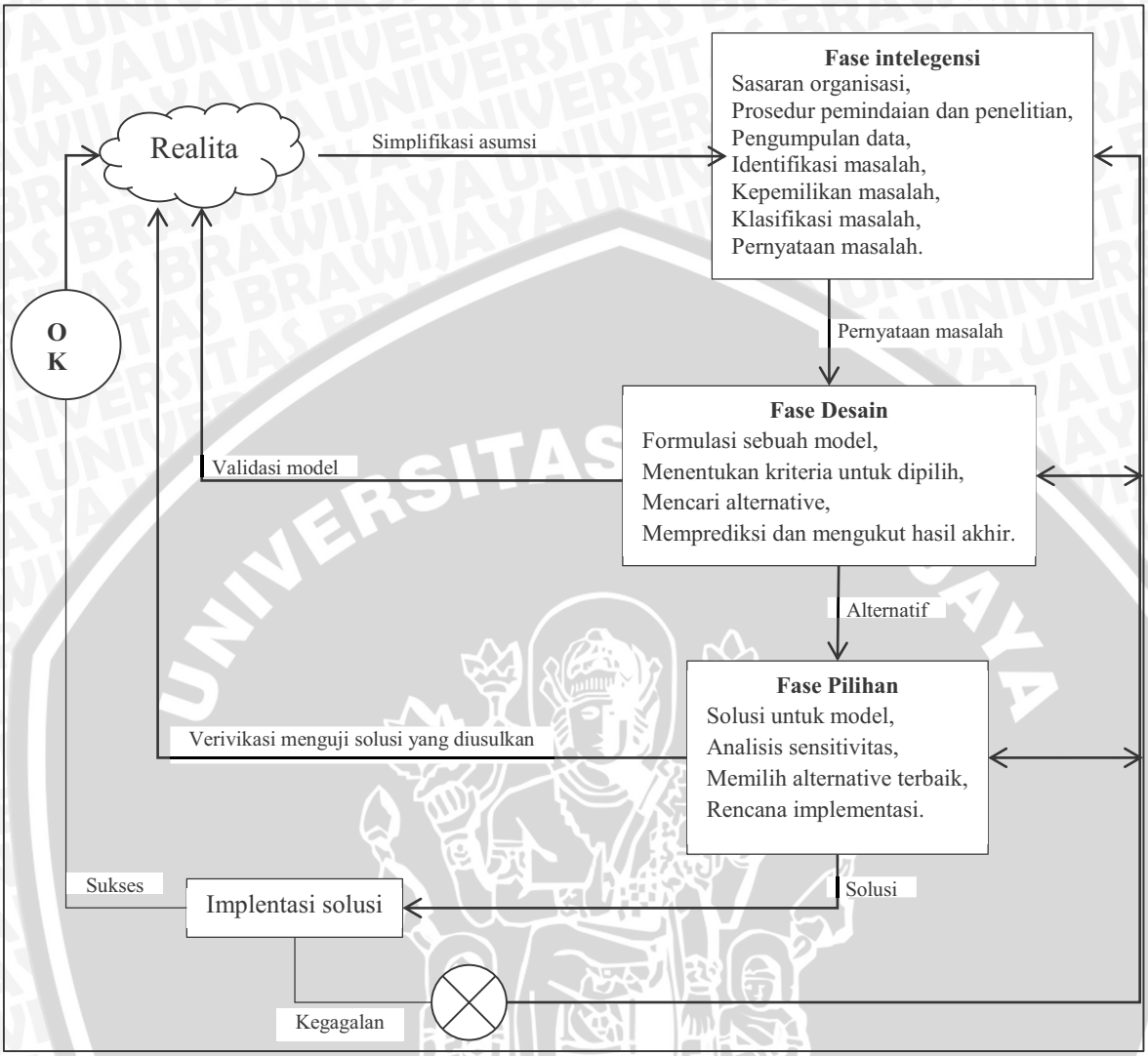
3. Fase *Choice*

   Fase ini adalah fase penting dalam pengambilan keputusan. Batas antara fase desain dan fase pilihan seringkali tidak jelas karena pada waktu tertentu kedua fase ini dapat dilakukan secara bersamaan dan seringkali terjadi perpindahan fase, baik dari fase desain ke fase pilihan maupun dari fase pilihan kembali ke fase desain. Pada tahap ini dilakukan proses pemilihan diantara berbagai alternatif tindakan yang mungkin utnuk dijalankan. Hasil pemilihan tersebut kemudian diimplementasikan dalam proses pengambilan keputusan.

4. Fase *Implementation*

   Fase ini merupakan fase untuk mengimplementasikan solusi yang direkomendasikan sebelumnya.

Skenario fase pengambilan keputusan dapat dilihat pada Gambar 2.7.



Gambar 2.7 Fase-fase Pengambilan Keputusan

Sumber : [27: 50]

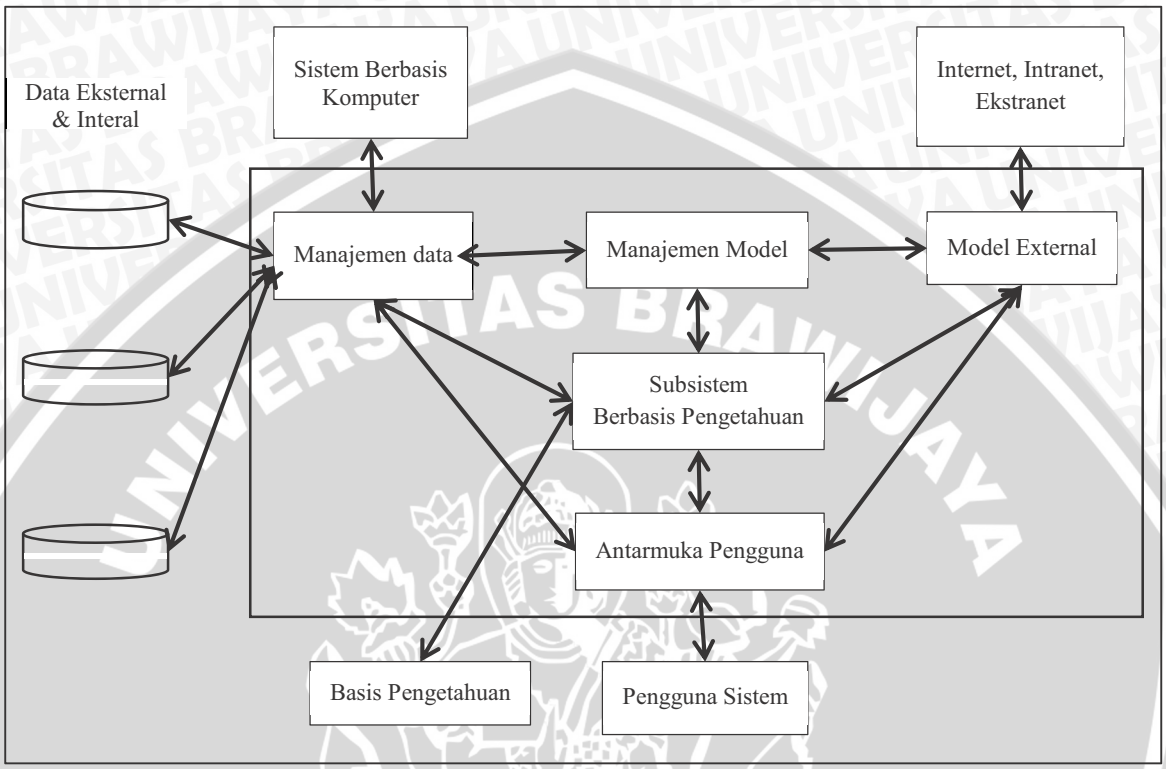
2.3.2 Karakteristik Sistem Pendukung Keputusan

Sebuah sistem dapat dikatakan sebagai sistem pendukung keputusan apabila memenuhi karakteristik-karakteristik seperti dibawah ini [5]:

1. Menggabungkan dua komponen utama yaitu, data dan model
2. Dibuat untuk membantu proses pengambilan keputusan dalam menyelesaikan permasalahan semi terstruktur dan tidak terstruktur
3. Digunakan untuk mendukung keputusan, bukan untuk menggantikan tugas si pembuat keputusan
4. Digunakan untuk meningkatkan efektifitas

2.3.3 Komponen Sistem Pendukung Keputusan

Aplikasi Sistem Pendukung Keputusan (SPK) dapat terdiri dari subsistem seperti ditunjukkan pada Gambar 2.8



Gambar 2.8 Skematik SPK
Sumber: [27:144]

- **Subsistem manajemen data**

  Subsistem manajemen data memasukkan satu database yang berisi data yang relevan untuk situasi dan dikelola oleh perangkat lunak yang disebut sistem manajemen database (DBMS). Subsistem manajemen data bisa terhubung dengan *data warehouse*.

- **Subsistem manajemen model**

  Merupakan paket perangkat lunak yang memasukkan model keuangan, statistik, ilmu manajemen, atau model kuantitatif lainnya yang memberikan kapabilitas analitik dan manajemen perangkat lunak yang tepat termasuk menyertakan bahasa pemodelan untuk membangun model kustom. Bahasa-bahasa pemodelan untuk membangun model-model kustom juga

dimasukkan. Perangkat lunak ini sering disebut sistem manajemen basis model (MBMS).

- **Subsistem antarmuka pengguna**

  Pengguna berkomunikasi dengan dan memerintahkan SPK melalui subsistem ini. Pengguna dianggab sebagai bagian dari sistem. Para peneliti menegaskan bahwa beberapa kontribusi unik dari SPK berasal dari interaksi yang intensif antara komputer dan pembuat keputusan.

- **Subsistem manajemen berbasis pengetahuan**

  Subsistem ini dapat mendukung semua subsistem lain atau bertindak sebagai komponen independen. Ia memberikan inteligensi untuk memperbesar pengetahuan si pengambil keputusan. Subsistem ini dapat diinterkoneksikan dengan repositori pengetahuan perusahaan yang kadang-kadang disebut basis pengetahuan organisasional.

Sistem Pendukung Keputusan harus mencakup tiga komponen utama dari DBMS, MBMS, dan antarmuka pengguna. Subsistem manajemen berbasis pengetahuan hanya bersifat opsional, namun bisa memberikan banyak manfaat dengan menyediakan intelegensi ke tiga komponen utama. Layaknya pada sistem informasi manajemen, pengguna dapat dianggap sebagai salah satu komponen sistem pendukung keputusan [27:145].

2.4 *Multiple Attribute Decision Making* (MADM)

Multi Atribute Decision Making merupakan sebuah metode untuk menyelesaikan permasalahan dengan cara mencari alternatif paling optimal dari keseluruhan alternatif yang ada dengan kriteria tertentu. Dalam metode MADM terdapat beberapa cara yang dapat digunakan untuk menyelesaikan sebuah permasalahan, diantaranya yaitu [23]:

a) *Simple Additive Weighting Method* (SAW)
b) *Weighted Product* (WP)
c) ELECTRE
d) *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS)
e) *Analytic Hierarchy Process* (AHP)

2.4.1 Elimination Et Choix Tranduisant La Realité (ELECTRE)

Metode ELECTRE (merupakan singkatan dari *Elimination and Choice Translating Reality*) pertama kali diperkenalkan oleh Benayoun, dkk pada tahun 1966. Metode ELECTRE merupakan salah satu metode pengambilan keputusan multikriteria dengan konsep *outranking* dengan menggunakan perbandingan berpasangan dari alternatif-alternatif berdasarkan setiap kriteria yang sesuai[28: 14]. Metode ELECTRE digunakan pada kondisi dimana alternatif yang kurang sesuai dengan kriteria akan dieliminasi, dan alternatif yang sesuai dapat dihasilkan. Suatu alternatif dikatakan mendominasi alternatif lainnya jika satu atau lebih kriterianya melebihi (dibandingkan dengan kriteria dari alternatif yang lain) dan sama dengan kriteria lain yang tersisa. ELECTRE cocok digunakan untuk masalah keputusan yang melibatkan sedikit kriteria dengan jumlah alternatif yang besar [12].

Langkah-langkah yang digunakan dalam penyelesaian masalah menggunakan metode ELECTRE adalah sebagai berikut [28: 14-17]:

Langkah 1: Normalisasi matrik keputusan

Dalam prosedur ini, setiap atribut diubah menjadi nilai yang *comparable*. Setiap normalisasi dari nilai rij dapat dilakukan dengan rumus sebagai berikut [28: 14]:

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x^2 ij}} \quad \text{..... (2-1),}$$

dimana:

- r_{ij} adalah nilai normalisasi
- x_{ij} adalah nilai elemen setiap kriteria
- i = 1, 2, 3,…, m. dimana m adalah banyaknya alternatif.
- j = 1, 2, 3,…, n. dimana n adalah banyaknya kriteria.

Sehingga didapat matriks R hasil normalisasi,

$$R = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{m1} & r_{m2} & \cdots & r_{mn} \end{bmatrix} \quad \text{.....(2-2),}$$

Persamaan 2.2 Matriks R Hasil Normalisasi

R adalah matriks yang telah dinormalisasi, dimana m menyatakan alternatif, n menyatakan kriteria dan r_{ij} adalah normalisasi pengukuran pilihan dari alternatif ke-*I* dalam hubungannya dengan kriteria ke-*j*.

Langkah 2: Pembobotan pada matrik yang telah dinormalisasi.

Setelah dinormalisasi, setiap kolom dari matrik R dikalikan dengan bobot-bobot (w_j) yang ditentukan oleh pembuat keputusan. Sehingga, *weighted normalized matrix* dapat dihitung dengan menggunakan rumus berikut [28: 14-15]:

$$V = R * W \quad \text{.....(2-3),}$$

Persamaan 2.3 Pembobotan Matriks Ternormalisasi

dimana:

V = nilai *weighted normalized matrix*

R = nilai matriks ternormalisasi

W = nilai bobot kepentingan

W dapat dituliskan dalam matriks pada persamaan 2-4 dimana *n* merupakan banyaknya kriteria

$$W = \begin{bmatrix} w_1 & 0 & \cdots & 0 \\ 0 & w_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & w_n \end{bmatrix} \quad \text{.....(2-4)}$$

Langkah 3: Menentukan *concordance* dan *discordance set*.

Untuk setiap pasang dari alternatif *k* dan *l* (*k,l* = 1,2,3,…,*m* dan k ≠ *l*) kumpulan kriteria *J* dibagi dmenjadi dua *subsets*, yaitu *concordance* dan *discordance*. Bilamana sebuah kriteria dalam suatu alternatif termasuk *concordance* adalah [28: 15]:

$$C_{kl} = \{j, v_{kj} \geq v_{lj}\} \quad \text{.....(2-5),}$$

dimana:

C_{kl} = himpunan *concordance*

v_{kj} = nilai kriteria pada baris k

v_{lj} = nilai kriteria pada baris l

j = 1, 2, 3, …, *n*, dimana *n* adalah banyaknya kriteria

k, l = 1, 2, 3,.., *m*.(*m* adalah banyaknya alternatif) dan $k \neq l$

Sebaliknya, komplementer dari subset ini adalah *discordance*, yaitu bila:

$$D_{kl} = \{j, v_{kj} < v_{lj}\} \quad \text{.....(2-6),}$$

dimana:

D_{kl} = himpunan *discordance*

v_{kj} = nilai kriteria pada baris k

v_{lj} = nilai kriteria pada baris l

j = 1, 2, 3, …, *n*. dimana *n* adalah banyaknya kriteria

k, l = 1, 2, 3,.., *m*.(*m* adalah banyaknya alternatif) dan $k \neq l$

Langkah 4 : Hitung matriks *concordance* dan *discordance*.

a. *Concordance*

Untuk menentukan nilai dari elemen-elemen pada matriks *concordance* adalah dengan menjumlahkan bobo-bobot yang termasuk dalam subset *concordance*, secara sistematis adalah pada rumus [28: 16]:

$$c_{kl} = \sum_{j \in ckl} w_j \quad \text{.....(2-7),}$$

dimana:

c_{kl} = matriks *concordance*

C_{kl} = himpunan *concordance*

W_j = bobot kepentingan setiap kriteria *j*

j = 1, 2, 3, …, *n.* dimana *n* adalah banyaknya kriteria.

k, l = 1, 2, 3,.., *m*.(*m* adalah banyaknya alternatif) dan $k \neq l$

Sehingga matrik *concordance* yang dihasilkan adalah:

$$C = \begin{bmatrix} - & C_{12} & C_{13} & C_{1n} \\ C_{21} & - & \cdots & C_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{m1} & C_{m2} & C_{m3} & - \end{bmatrix}$$

b. *Discordance*

Untuk menentukan nilai dari elemen-elemen pada matriks *discordance* adalah dengan membagi maksimum selisih nilai kriteria yang termasuk dalam subset *discordance* dengan maksimum selisih nilai seluruh kriteria yang ada, secara matematisnya adalah:

$$d_{kl} = \frac{max\{|v_{kj} - v_{lj}|\}_{j \in D_{kl}}}{max\{|v_{kj} - v_{lj}|\}_{\forall j}} \quad \text{.....(2-8),}$$

dimana:

d_{kl} = matriks *discordance*

D_{kl} = himpunan *discordance*

v_{kj} = nilai kriteria pada baris *k*

v_{lj} = nilai kriteria pada baris *l*

j = 1, 2, 3, …, *n.* dimana *n* adalah banyaknya kriteria.

k, l = 1, 2, 3,.., *m*.(*m* adalah banyaknya alternatif) dan $k \neq l$

Sehingga matrik *discordance* yang dihasilkan adalah:

$$D = \begin{bmatrix} - & d_{12} & d_{13} & d_{1n} \\ d_{21} & - & \cdots & d_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{m1} & d_{m2} & d_{m3} & - \end{bmatrix}$$

Langkah 5: Menentukan matrik dominan *concordance* dan *discordance*.

a. *Concordance,* dengan rumus [28: 17] :

Matrik dominan *concordance* dapat dibangun dengan bantuan nilai *threshold*, yaitu dengan membandingkan setiap nilai elemen matriks *concordance* dengan nilai *threshold*.

$C_{kl} \geq \underline{c}$ .....(2-9),

dengan ketentuan :

$$f_{kl} = \begin{cases} 0, & c_{kl} < \underline{c} \\ 1, & c_{kl} \geq \underline{c} \end{cases}$$

Dengan nilai *threshold c*, adalah:

$\underline{c} = \frac{\sum_{k=1}^{n}\sum_{l=1}^{n} c_{kl}}{m(m-1)}$ .....(2-10),

dimana:

$\underline{c}$ = Nilai *threshold concordance*

C_{kl} = himpunan *concordance*

m = banyak alternatif yang dimiliki

f_{kl} = matriks dominan *corcondance*

b. *Discordance*, dengan rumus [28: 17] :

Untuk membangun matrik dominan *discordance* juga menggunakan bantuan nilai *threshold*, yaitu:

$\underline{d} = \frac{\sum_{k=1}^{m}\sum_{l=1}^{m} C_{kl}}{m*(m-1)}$ ..... (2-11),

dimana:

$\underline{d}$ = Nilai *threshold discordance*

d_{kl} = himpunan *discordance*

m = banyak alternatif yang dimiliki

g_{kl} = matriks dominan *discordance*

Untuk menentukan nilai matriks g, digunkaan rumus berikut :

$d_{kl} \geq \underline{d}$ .....(2-12),

dimana:

$\underline{d}$ = Nilai *threshold discordance*

d_{kl} = himpunan *discordance*

m = banyak alternatif yang dimiliki

g_{kl} = matriks dominan *discordance*

dengan ketentuan :

$$g_{kl} = \begin{cases} 0, & d_{kl} < \underline{d} \\ 1, & d_{kl} \geq \underline{d} \end{cases}$$

Langkah 6 : Menentukan *aggregate dominance matrix*

Langkah selanjutnya adalah menentukan *aggregate dominance matrix* sebagai matriks *e*, yang setiap elemennya merupakan perkalian Antara elemen matriks *F* dengan elemen matriks *G*, sebagai berikut [28: 17]:

$e_{kl} = f_{kl} x g_{kl}$ .....(2-13),

dimana:

e_{kl} = nilai *aggregate dominance matrix*

f_{kl} = matriks dominan *concordance*

g_{kl} = matriks dominan *discordance*

Langkah 7: Eliminasi alternatif yang *less favourable*.

Matriks *E* memberikan urutan pilihan dari setiap alternatif, yaitu bila $e_{kl} = 1$ maka alternatif A_k merupakan pilihan yang lebih baik dari A_1 . Sehingga baris dalam matriks *E* yang memiliki jumlah $e_{kl} = 1$ paling sedikit dapat dieliminasi. Dengan demikian alternatif terbaik adalah yang mendominasi alternatif lainnya.

2.4.2 Weighted Product

Metode *weighted product* merupakan salah satu metode yang ada dalam penyelesaian permasalahan MADM (*Multi Atribute Decision Making*). Menurut

Yoon dalam kusumadewi (2006:19), Metode *weighted product* menggunakan perkalian untuk menghubungkan rating atribut, dimana rating setiap atribut harus dipangkatkan dulu dengan bobot atribut yang bersangkutan. Proses ini sama halnya dengan proses normalisasi [9]. Alternatif terbaik adalah alternatif terbaik dari alternatif lainnya [8].

Preferensi alternatif yang diberikan pada persamaan 2-14 berikut [9]:

$$S_i = \prod_{j=1}^{n} {x_{ij}}^{w_j} \quad \text{.....(2-14),}$$

dimana:

S : *Preferensi* alternatif dianologikan sebagai vektor S

x : Nilai kriteria

W : Bobot kriteria/subkriteria

i : Alternatif (1,2,3,…n)

j : Kriteria

n : Banyaknya kriteria

dimana $\sum W_j = 1. W_j$ adalah pangkat bernilai positif untuk atribut yang memberikan keuntungan dan bernilai negatif untuk atribut yang berbentuk biaya. Preferensi relatif dari setiap alternatif diberikan pada persamaan 2-15 berikut [9]:

$$V_i = \frac{\prod_{j=1}^{n} {x_{ij}}^{w_i}}{\prod_{j=1}^{n} ({x_j}^*)} \quad \text{.....(2-15),}$$

dimana:

V : Preferensi relatif dari setiap alternative dianalogikan sebagai vektor V

x : Nilai Kriteria

W : Bobot kriteria/subkriteria

i : Alternatif (1,2,3,…n)

j : Kriteria

n : Banyaknya kriteria

* : Banyaknya kriteria yang telah dinilai pada vektor S

2.5 Bahasa Pemrograman C#

Bahasa Pemrograman C# adalah sebuah Bahasa pemrograman yang dibuat dan dikembangkan Microsoft bersama dengan *platform* .NET. Bahasa pemrograman C# merupakan *high-level language* yang mirip dengan Java dan C++. Bahasa pemrograman C# merupakan salah satu Bahasa pemrograman terpopuler di dunia yang dipakai oleh banyak developer. C# banyak digunakan di anak perusahaan Micrososft, organisasi, dan developer. Berbeda dengan Java, C# hanya bisa dikembangkan oleh oleh Microsoft dan tidak terbuka untuk umum [18: 18]

2.6 *Database Management System* MySQL

Database Management System (DBMS) merupakan sistem yang digunakan admin *database* untuk mengatur dan mengembangkan *database* kedalam data yang lebih spesiifik [13: 26]. Databse sendiri adalah kumpulan data yang tersimpan kedalam table dengan menggunkaan model-model tertentu [17: 26-27]. MySQL adalah sebuah DBMS yang didesain berbasis *open source*. MySQL dapat mendukung untuk membangun beberapa tabel dan menghubungkan tabel-tabel tersebut dengan menggunakan kolom dalam tabel [17: 33]. MySQL adalah sebuah aplikasi DBMS yang berbasis *open source*. MySQL mempunyai beberapa fitur utama yaitu [17: 33] :

- Cepat dan Efektif

  MqSQL dikembangkan dengan menggunakan bahasa C/C++ dengan menggunakan multithreaded sehingga membuat sistem lebih cepat.

- Tipe kolom

  MySQL mendukung banyak tipe kolom atau tipe data yang dapat digunakan, seperti signed/unsigned integer, bytes long, float, double, char, varchar, text, blob, date, time, datetime, timestamp, year, set dan enum.

- Full-feature pada perintah SQL

  MySQL mendukung semua standart perintah SQL, seperti select, insert, delete dan join selaipun. Selain intu MySQL juga mendukung perintah show untuk memberikan informasi mengenai sistem.

Fungsi

MySQL mendukung penggunaan fungsi pada perintah SQl, seperti fungsi AVG(), SUM() dan masih banyak yang lainnya.

- Keamanan

  MySQL memberikan keamanan dengan menggunakan sistem *privilege* dan *password*.

- Skalabilitas

  MySQL mendukung admin untuk membuat ribuan table dan jutaan baris dalam databasenya.

- Karakter set

  MySQL mendukung beberapa macam karakter set dan pesan eror pada output yang dihasilkan.

2.7 *Data Flow Diagram* (DFD)

Data Flow Diagram (DFD) adalah sebuah diagram yang bertujuan untuk menggambarkan logika model dan menjelaskan perubahan aliran data yang terjadi dalam sistem. Untuk menjelaskan proses aliran data maka digunakan simbol-simbol/notasi-notasi tertentu, sedangkan untuk mendeskripsikan proses dekomposisi sistem yang lebih rinci atau dekomposisi [16:85].

Berikut ini adalah symbol-simbol yang digunakan dalam DFD [19:2-8]:

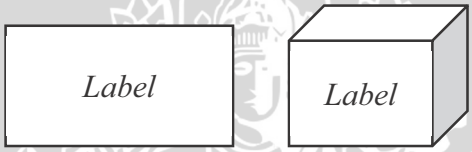
- **Komponen Terminator/Entitas Luar**

  Teminator mewakili entitas eksternal yang berkomunikasi dengan sistem yang sedang dikembangkan. Biasanya terminator dikenal dengan nama entitas luar (*eksternal entity*). Terdapat dua jenis terminator yaitu [19:2-3]:

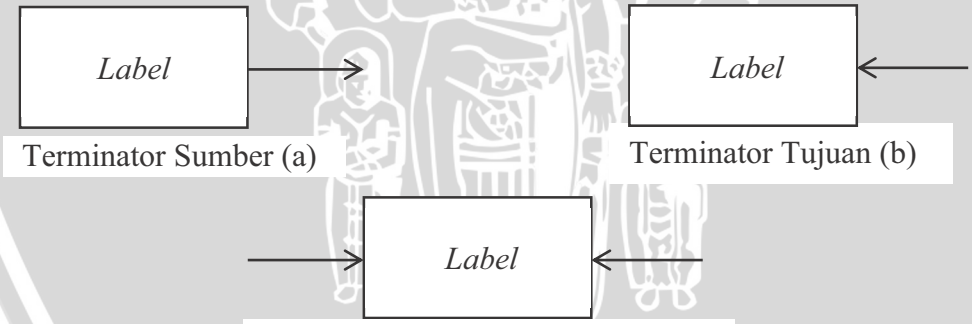
  1. Teminator Sumber (*source*): merupakan terminator yang menjadi sumber, ditunjukkan pada Gambar 2.10 (a)
  2. Terminator Tujuan (*sink*): merupakan terminator yang menjadi tujuan data/informasi sistem, ditunjukkan pada Gambar 2.10 (b)

  Beberapa hal yang perlu diperhatikan dalam menggunakan symbol terminator ini antara lain [16:87-88]:

- Terminator merupakan entitas sumber/tujuan dari aliran data eksternal
- Terminator merupakan entitas yang hanya dapat mengirim/menerima data
- Termiantor merupakan entitas yang harus diberi label yang menggunakan kata benda
- Terminator merupakan bagian/lingkungan luar sistem berupa orang, sekelompok orang, organisasi atau lainnya yang dapat berinteraksi dengan sistem
- Terminator merupakan entitas yang menerima informasi dari sistem, memberikan informasi baru dalam sistem, dam menjalankan sistem.



Gambar 2.9. Simbol Terminator
Sumber : [16: 88]



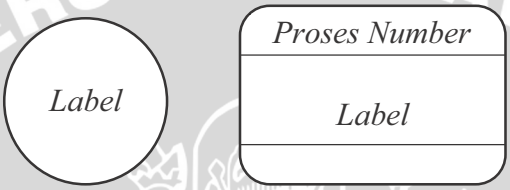
Gambar 2.10. Jenis Terminator
Sumber : [16: 2]

- **Proses**

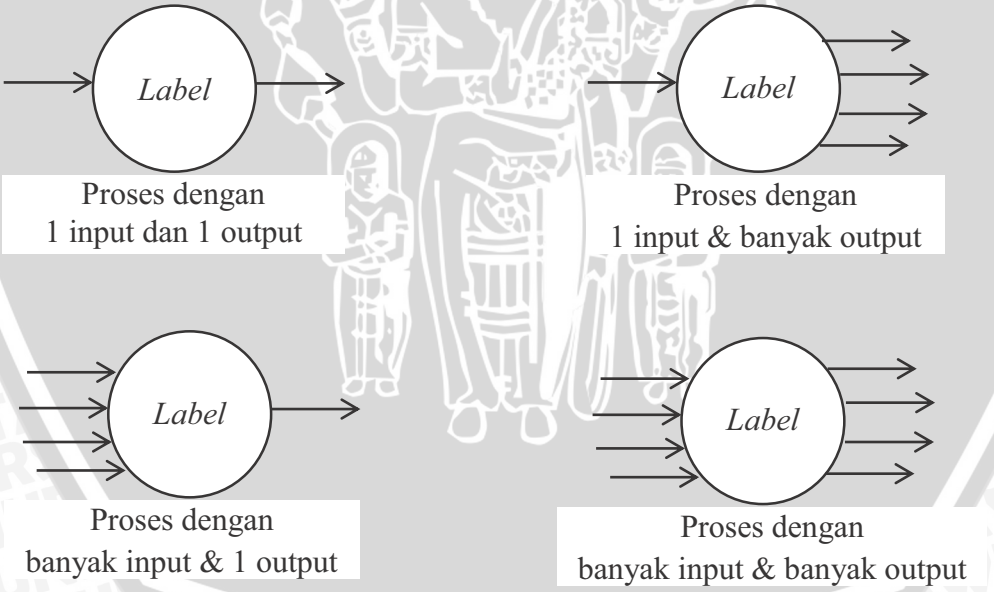
Proses adalah komponen yang menggambarkan bagian dari sistem yang dibagun dengan mengubah input menjadi output, simbol proses dapat dilihat pada Gambar 2.11. Proses dapat dibedakan menjadi 4 kemungkinan terjadinya proses yang dapat dilihat pada Gambar 2.12 [19: 3].

Beberapa hal yang perlu diperhatikan dalam menggunakan simbol proses ini antara lain [16:86]:

- Proses harus menerima aliran data sebagai input dan menghasilkan aliran data sebagai output
- Proses dapat dibagi menjadi beberapa proses yang lebih detail (sub-proses)
- Proses merupakan komponen yang harus diberi label berupa kata kerja
- Proses merupakan internal sistem yang dapat dihubungkan dengan komponen terminator, data store atau proses melalui alur data.
- Terdapat paling sedikit 1 input aliran data dan 1 output aliran data



Gambar 2.11. Simbol Proses
Sumber : [16: 86]



Gambar 2.12. Jenis Proses
Sumber : [19: 2]

- **Data Store**

Data store adalah komponen yang digunakan untuk model kumpulan data pada sistem yang dibagun, simbol proses dapat dilihat pada Gambar

2.11. Berdasarkan aliran data yang ada data store dapat dibedakan menjadi 2 jenis yaitu [19:5] :

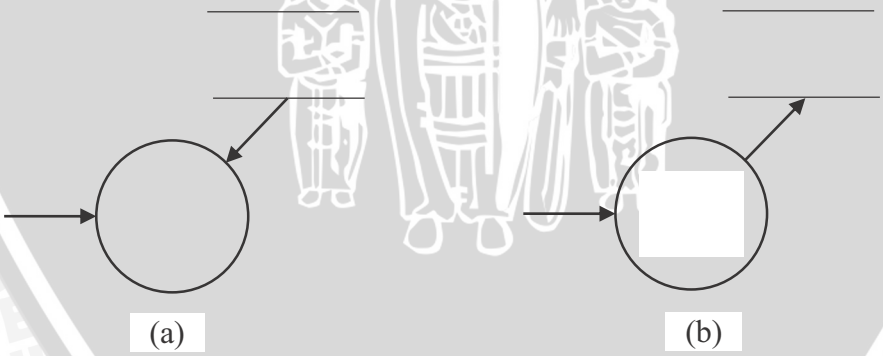
- Alur data dari data store, yang merupakan pembacaaan atau pengaksesan data dalam sistem, ditunjukkan pada Gambar 2.14 (a).
- Alur data ke data store, yang merupakan pengupdatean data dalam sistem seperti menghapus atau mengubah data dalam sistem, ditunjukkan pada Gambar 2.14 (b).

Beberapa hal yang perlu diperhatikan dalam menggunakan simbol data store ini antara lain [16: 87-88]:

- Data store hanya dihubungkan dengan kompunen proses
- Data store merupakan komponen yang harus diberi label berupa kata benda jamak
- Data store merupakan internal sistem yang dapat dihubungkan dengan komponen proses melalui alur data



Gambar 2.13. Simbol Data store
Sumber : [16: 87]



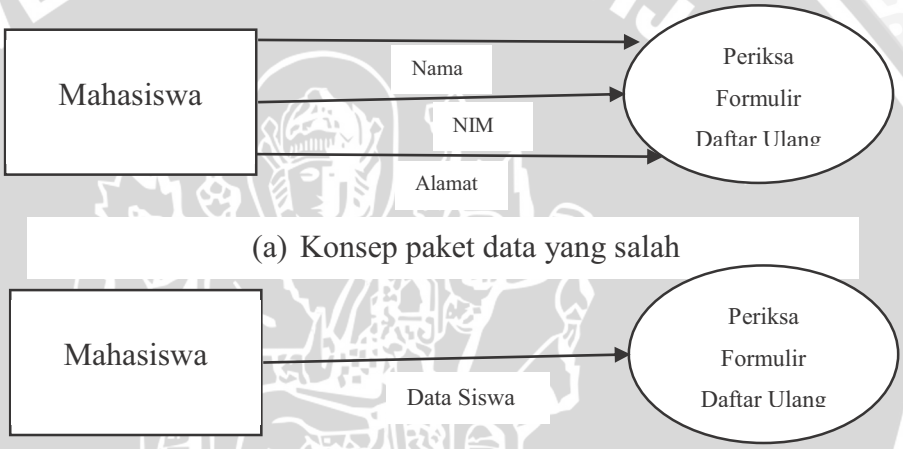
Gambar 2.14. Jenis Data store
Sumber : [19: 2]

- **Data Flow / Alur Data**

Data flow adalah komponen yang digunakan sebagai model aliran data yang digambarkan dengan anak panah, sekaligus menunjukkan arah aliran data menuju atau keluar dari proses, simbol data flow dapat dilihat pada

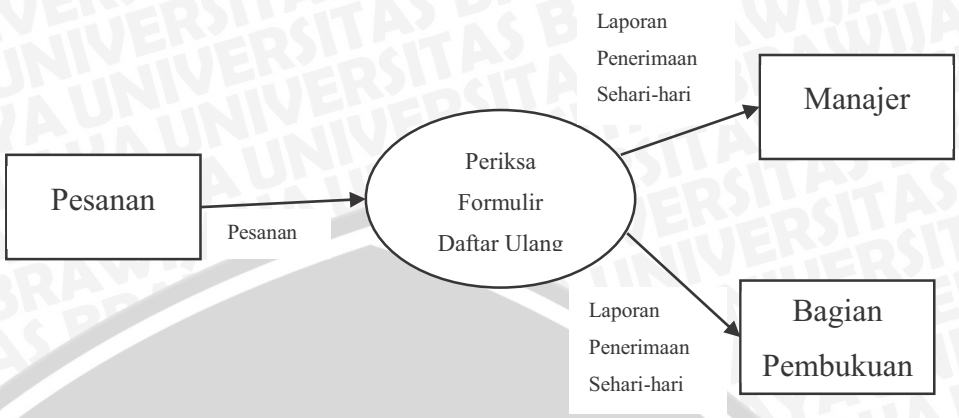
Gambar 2.18. Terdapat 4 konsep yang digunakan dalam penggambaran alur data, yaitu [19: 6-8] :

- **Konsep paket data**, merupakan alur data yang menggabungkan beberapa data berhubungan menjadi satu paket data yang sama. Dengan ketentuan alur data tersebut mempunyai sumber dan tujuan yang sama. Contohnya, terdapat data pribadi siswa yang terdiri dari Nama, Nomer Induk, Tempat Tanggal Lahir, Alamat dan lainnya. Data-data tersebut dapat diringkas menjadi satu paket data yaitu paket Data Siswa. Penggambaran konsep paket data dapat dilihat pada Gambar 2.19 (a).



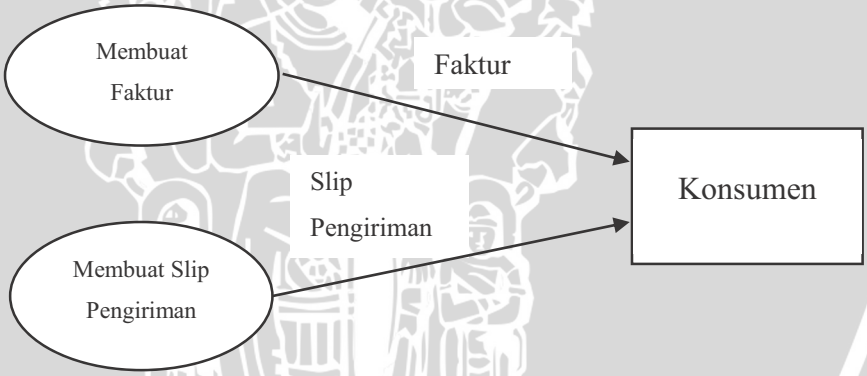
Gambar 2.15. Konsep Paket Data
Sumber : [19: 6]

- **Konsep alur data menyebar**, merupakan alur data yang mempunyai sumber sama dan menyebar ke tujuan yang berbeda. Penggambaran konsep alur data menyebar dapat dilihat pada Gambar 2.19 (b).



Gambar 2.16. Konsep Alur Data Menyebar
Sumber : [19: 7]

- **Konsep alur data mengumpul**, merupakan alur data dari beberapa sumber data yang berbeda dan mengumpul ke tujuan yang sama. Penggambaran konsep alur data mengumpul dapat dilihat pada Gambar 2.19 (c).

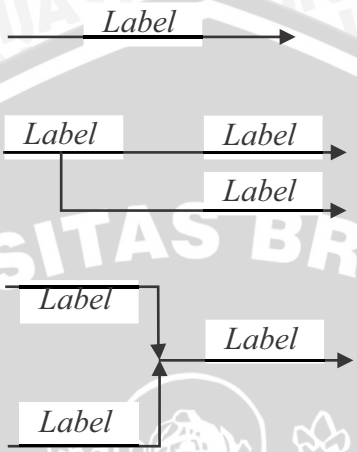


Gambar 2.17. Konsep Alur Data Menyebar
Sumber : [19: 7]

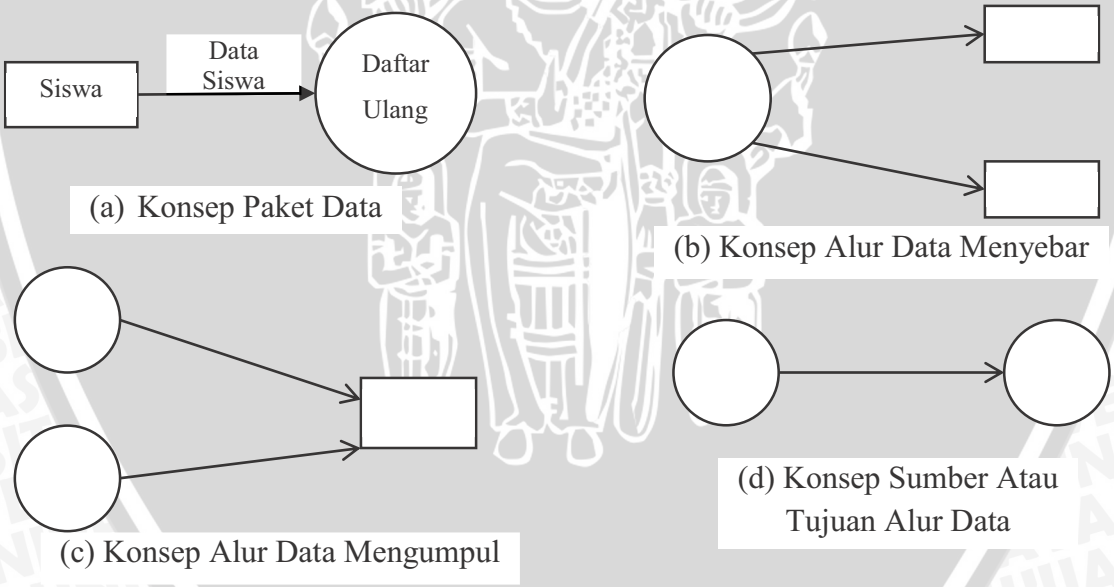
- **Konsep sumber atau tujuan alur data**, merupakan alur data dengan ketentuan harus mengandung minimal satu proses. Terdapat beberapa contoh peggunaannya diantaranya :
  - Alur data yang bersumber atau bertujuan pada suatu proses.
  - Alur data yang bersumber dan bertujuan pada suatu proses.

  Penggambaran konsep sumber atau tujuan alur data dapat dilihat pada Gambar 2.19 (d). Beberapa hal yang perlu diperhatikan dalam menggunakan simbol data flow ini antara lain [18: 87-88]:

- Data flow merupakan konektor yang berfungsi untuk menghubungkan proses, data store, dan terminator.
- Data flow merupakan komponen yang harus diberi label berupa kata benda



Gambar 2.18. Simbol Data flow
Sumber : [19: 87]

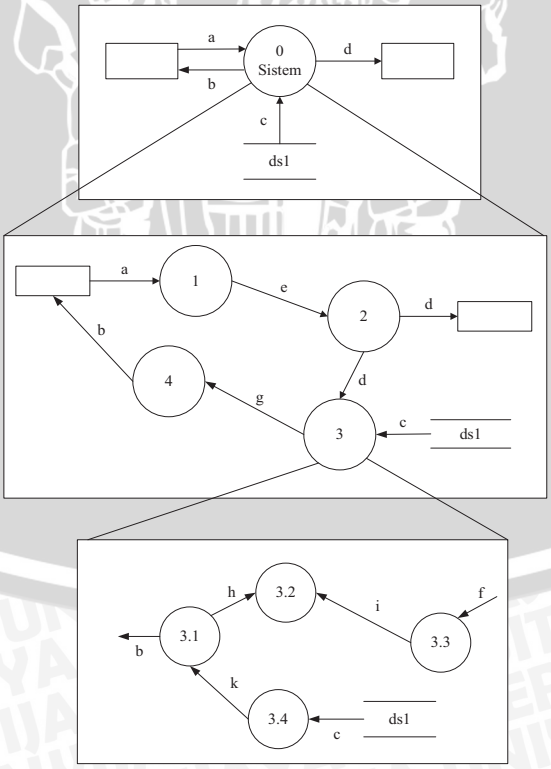


Gambar 2.19. Konsep Data flow
Sumber : [19: 6-8]

Penggambaran DFD yang konsisten, memiliki beberapa garis besar langkah-langkah dalam pembuatannya yaitu [16][19]:

1) Mengidentifikasi entitas yang akan digunakan pada sistem.

2) Mengidentifikasi input dan output yang akan digunakan oleh entitas.
3) Membuat Diagram Konteks (diagram context), yang menggambarkan sistem dan merupakan diagram level tertinggi dari DFD.
4) Membuat Diagram Level Zero, yang merupakan penjabaran detail dari diagram konteks dengan ketentuan masuk/keluar alur data pada masing-masing proses memperhatikan konsep keseimbangan dengan diagram level sebelumnya. Penomeran dilakukan dengan cara memberi nomer pada proses utama (urutan tidak mempengaruhi).
5) Membuat Diagram Level Satu, yang merupakan dekomposisi dari diagram level zero dengan membagi proses ke sub proses dengan ketentuan masuk/keluar alir data pada proses memperhatikan konsep keseimbangan dengan diagram level sebelumnya. Penomeran dilakukan dengan cara mengikuti nomer proses level sebelumnya ditambah nomer sub proses itu sendiri.
6) DFD Level Dua, Tiga, dst, yang merupakan dekomposisi dari level sebelumnya dengan aturan yang sama dengan level satu.



Gambar 2.20. Levelisasi DFD
Sumber : [19: 16]

2.8 *Entity-Relationship Diagram* (ERD)

Entity Relationship Diagram (ERD) adalah teknik penggambaran secara umum untuk struktur data dan desain sistem database. ERD digunakan untuk mendeskripsikan hubungan antara tabel penyimpanan (data) atau dapat dikatakan pemodelan data. [16: 125-127]. ERD menggunakan sejumlah notasi dan simbol untuk menggambarkan struktur dan hubungan antar data. Ada tiga komponen utama pada ERD yaitu [16: 127-133]:

- ***Entity***

  Entity merupakan perwakilan dari orang, tempat, benda, peristiwa, atau konsep yang dibutuhkan untuk menyimpan data. Simbol dari entitas dapat dilihat pada Gambar 2.21 (a).

- **Atribut**

  Atribut merupakan *element* yang dimiliki oleh *entity*. Atribut terdapat beberapa macam seperti *primary key*, *foreign key*, *candidate key*, *group,* dan *subsetting* kriteria. Simbol dari atribut dapat dilihat pada Gambar 2.21 (b).

- **Relasi**

  Relasi merupakan hubungan antara entitas yang ada, baik antar satu entitas atau lebih entitas. Relasi ini mewakili suatu peristiwa antar entitas atau hanya sebagai relasi logis antar entitas. Simbol dari relasi dapat dilihat pada Gambar 2.21 (c).

  Dalam ERD terdapat mcam-macan relasi yang memungkinkan dalam pemodelan data, diantaranya [16: 130-133] :

  - Satu Ke Satu (*One to One*)

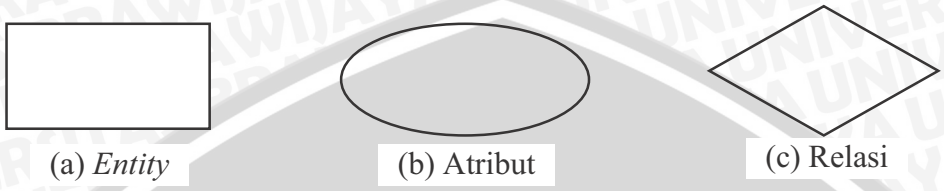
    Relasi antara entitas A dengan entitas B, dimana entitas A mempunyai satu hubungan dengan entitas B dan begitu sebaliknya, yang digambarkan pada Gambar 2.22 (a).

  - Satu Ke Banyak (*One to Many*)

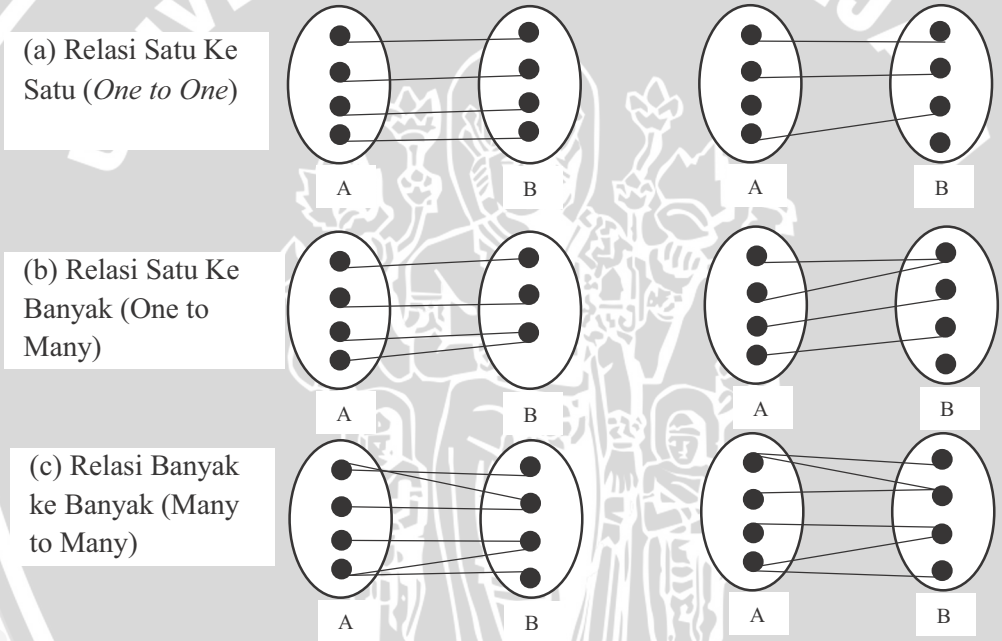
    Relasi antara entitas A dengan entitas B, dimana entitas A mempunyai banyak hubungan dengan entitas B, sedangkan entitas B hanya mempunyai satu hubungan dengan antitas A, yang digambarkan pada Gambar 2.22(b).

- Banyak ke Banyak (*Many to Many*)

  Relasi antara entitas A dengan entitas B, dimana entitas A mempunyai banyak hubungan dengan entitas B dan begitu sebaliknya, yang digambarkan pada Gambar 2.22(c).



Gambar 2.21. Simbol ERD
Sumber : [16: 128]



Gambar 2.22. Macam Relasi
Sumber : [16: 130]

2.9 Teknik Pemberian Bobot

Menurut Dennis M. Buede [7], Teknik Pemberian Bobot dapat dibagi menjadi dua kategori, yakni pembobotan secara langsung (*Direct Weight Elication Techinques*) dan pembobotan secara tidak langsung (*Indirect Weight Elication Techinques*) yang dijelaskan sebagai berikut:

2.9.1 Direct Weight Elication Techinques

Pembobotan dengan cara langsung dapat dilakukan dengan cara pemberian rangking dari sebuah kriteria, dimulai dengan rangking yang paling tinggi sampai

yang paling rendah. Selanjutnya bobot dapat dihitung dengan menggunakan salah satu persamaan yang diinginkan.

2.9.1.1 *Rank-Order Centroid Technique* (ROC)

ROC merupakan salah satu metode yang paling mudah dalam melakukan pembobotan. Metode ini menggunakan rangking tingkat kepentingan atau prioritas dari kriteria yang ada. Kriteria akan disusun berdasarakan urutan dari kriteria yang paling penting hingga kriteria yang paling tidak penting [7]. Ada beberapa rumus yang dapat digunakan untuk melakukan pembobotan ROC yaitu [7]:

$$W_i = \frac{1}{M}\sum_{n=1}^{M}\frac{1}{n} \quad \text{(2-16)}$$

$$W_1 = \left(1 + \frac{1}{2} + \frac{1}{3} + \cdots \frac{1}{M}\right)/k \quad \text{(2-17)},$$

$$W_m = \left(0 + 0 + 0 + \cdots \frac{1}{M}\right)/k \quad \text{(2-18)},$$

dimana:

W_i= nilai bobot keperntingan kriteria ke-i

M= banyaknya kriteria

n= ranking dari kriteria

i= 1,2,3,…, m. dimana m adalah banyaknya kriteria

Sebagai contoh terdapat 3 kriteria yang akan dicari nilai bobotnya, dimana M=3. Maka perhitungan bobot dengan menggunakan metode ROC adalah sebagai berikut:

$$W_1 = \left(1 + \frac{1}{2} + \frac{1}{3}\right)/3 = 0.61$$

$$W_2 = \left(0 + \frac{1}{2} + \frac{1}{3}\right)/3 = 0.28$$

$$W_3 = \left(0 + 0 + \frac{1}{3}\right)/3 = 0.11$$

Berikut adalah tabel perhitungan ROC dengan 3 kriteria[29]:

Tabel 2.5 Tabel ROC

Ranking	Kriteria	
	2	3
1	0.7500	0.6111
2	0.2500	0.2778
3		0.1111

Sumber: [29]

Jika terdapat beberapa kriteria yang memiliki prioritas yang sama, maka nilai bobot yang digunakan adalah nilai rata-rata dari bobot kriteria yang memiliki prioritas sama. Misalkan dalam permasalahan di atas, bobot 2 dan bobot 3 memiliki prioritas yang sama, maka [29]:

$$C_1 = W_1 = 0.61$$

$$C_2 = \frac{W_2 + W_3}{2} = \frac{0.28 + 0.11}{2} = 0.1945$$

$$C_3 = \frac{W_2 + W_3}{2} = \frac{0.28 + 0.11}{2} = 0.1945$$

2.9.1.2 Rank Sum

$$wt_i = \frac{K - r_1 + 1}{\sum_{j=1}^{K} K - r_1 + 1} \quad \text{(2-19)},$$

dimana:

r_i = ranking dari kriteria ke-i

K = banyaknya kriteria

w_i = normalisasi kriteria ke-i

i = 1,2,3,…, m. dimana m adalah banyaknya kriteria

2.9.1.3 Rank Exponent

$$wt_i = \frac{(K - r_1 + 1)^z}{\sum_{j=1}^{K} (K - r_1 + 1)^z} \quad \text{(2-20)},$$

dimana:

r_i = ranking dari kriteria ke-i

K = banyaknya kriteria

w_i = normalisasi kriteria ke-i

i = 1,2,3,…, m. dimana m adalah banyaknya kriteria

z = nilai kepentingan kriteria, semakin tinggi nilai z semakin penting kriteria tersebut

2.9.1.4 Rank Reciprocal

$$wt_i = \frac{1/r_1}{\sum_{j=1}^{K}(1/r_1)} \quad \text{(2-21)},$$

dimana:

r_i = ranking dari kriteria ke-i

K = banyaknya kriteria

w_i = normalisasi kriteria ke-i. i = 1,2,3,…, m. dimana m adalah banyaknya kriteria

2.9.2 Indirect Weight Elication Techinques

Pemberian bobot dengan teknik *indirect* dapat diperoleh dengan beberapa cara dari teknik perbandingan berpasangan dan penggunaan penyesuaian grafis pada komputer. Teknik-teknik pada pemberian bobot secara tidak langsung biasanya lebih unggul dibandingkan dengan pembobotan secara tidak langsung dalam kemampuannya untuk menangkap keinginan pembuat keputusan diseluruh tujuannya [7].

Teknik perbandingan berpasangan yang paling sering digunakan adalah *Analytical Hierarchy Process* (AHP) [Saaty, 1980], *Trade Offs* [Watson and Buede, 1987], *Balance Team Approach* [Watson and Buede, 1987], *Lottery Questions* [Keeney and Raiffa, 1987] [7].

2.9.2.1 *Analytical Hierarchy Process* (AHP)

AHP dapat digunakan untuk mengetahui bobot dari kriteria-kriteria yang ada. Dalam AHP, pengguna akan diminta untuk membandingkan setiap kriteria dengan seluruh kriteria yang ada. AHP tidak perlu mengetahui peringkat masing-masing kriteria yang ada, dan hanya meminta pengguna untuk membandingkan dua kriteria secara bersamaan tanpa memperhatikan peringkat. Langkah-langkah AHP dapat dilihat sebagai berikut[7]:

1. Pada proses ini, pengguna akan diminta untuk membandingkan dua objek secara bersamaan. Perbandingan dilakukan dengan memberikan skala numerik dari 1 (memiliki tingkat kepentingan yang sama) hingga 9 (salah satu objek memiliki tingkat kepentingan yang jauh lebih tinggi dibandingkan dengan objek lainnya).
2. Nilai perbandingan akan dimasukkan ke dalam sebuah matriks yang nantinya akan digunakan untuk menentukan nilai *eigenvector*.
3. Normalisasi pada nilai *eigenvector*.

2.10 *Precision and Recall*

Precision atau ketepatan adalah jumlah prediksi nilai positif yang benar memiliki nilai positif. *Recall* atau perolehan adalah jumlah data positif yang diprediksi bernilai positif dengan benar, dengan kata lain *recall* adalah kemampuan sistem dalam mengambil data yang relevan[20] . Tabel perhitungan penentuan *recall* dan *precision* dapat dilihat pada Tabel 2.6.

Tabel 2.6 Perhitungan Penentuan *Recall* dan *Precision*

Precision dan *Recall*		Nilai Sebenarnya	
		Retrieved	*Not Retrieved*
Hasil Metode	*Retrieved*	*True Positive* (TP)	*FalsePositive* (FP)
	Not Retrieved	*False Negative* (FN)	*True Negative* (Tn)

Sumber: [14].

Rumus dari *precision* dapat dilihat pada Persamaan 22 [14].

$$Precision\ (p) = \frac{TP}{TP+FP} \quad \text{(2-22)},$$

Rumus dari *precision* dapat dilihat pada Persamaan 23[14].

$$Recall(r) = \frac{TP}{TP+FN} \quad \text{(2-23)},$$

Rumus dari *F-Measure* (F_1) dapat dilihat pada Persamaan 24[1].

$$F - Measure\ (F_1) = 2x\frac{precision.recall}{precision+recall} \quad \text{(2-24)},$$

2.11 Akurasi

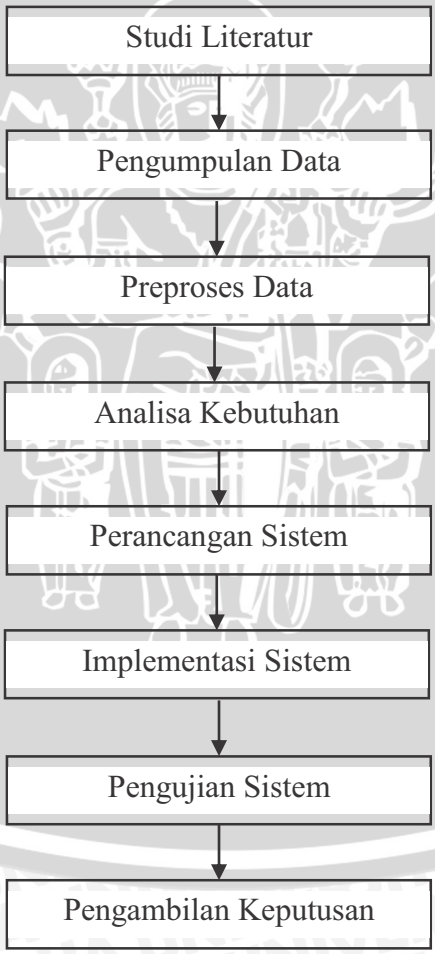
Akurasi adalah ukuran kedekatan hasil pengukuran terhadap angka sebenarnya (*true value / reference value*). Pada penelitian ini pengujian akurasi dilakukan untuk mengetahui kemampuan sistem dalam membuat keputusan [13]. Akurasi dilakukan dengan menghitung jumlah diagnosis yang tepat dibagi dengan jumlah data, atau dapat dirumuskan dengan persamaan berikut ini [13] :

$$Akurasi = \frac{\sum data\ uji\ benar}{\sum data\ uji} * 100\% \quad \text{.....(2-25)}$$

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB III
METODOLOGI PELAKSANAAN

Bab metodologi ini akan membahas tentang metode yang digunakan dalam pembuatan sistem pendukung keputusan untuk penerimaan siswa baru SMA BSS dengan menggunakan metode ELECTRE - *Weighted Product*. Metodologi penelitian ini dilakukan melalui beberapa tahapan, yaitu studi literature, pengumpulan data, analisa dan perancangan, implementasi sistem, pengujian sistem, dan pengambilan keputusan. Tahapan-tahapan dalam penelitian tersebut diilustrasikan dalam diagram blok metode penelitian pada Gambar 3.1.



Gambar 3.1 Diagram Blok Metodologi Penelitian
Sumber : [Perancangan]

3.1 Studi Literatur

Tahapan studi literature pada penelitian ini adalah mempelajari literature dari beberapa bidang informasi dan pustaka yang berkaitan dengan pembuatan sistem pendukung keputusan penerimaan siswa SMA BSS. Literatur diperoleh dari buku, internet, penjelasan dari pihak SMA Brawijaya *Smart School*, dan juga dari bimbingan dengan dosen pembimbing. Adapun teori yang dipelajari diantaranya:

- Sistem Pendukung Keputusan
- Metode MADM
- Metode *Elimination and Choice Expressing Reality* (ELECTRE)
- Metode *Weighted Product*
- Rekayasa Lunak
- Pemrograman dengan menggunakan Bahasa Pemrograman C#
- DBMS MySQL
- Tejnik Pemberian Bobot
- Proses Pengujian Sistem
- Akurasi

3.2 Pengumpulan Data

Pengumpulan data dilakukan di SMA Brawijaya *Smart School* Kota Malang. Pengumpulan data dilakukan dengan cara melakukan izin dan wawancara kepada pihak SMA Brawijaya *Smart School*. Data pada penelitian ini menggunakan data penerimaan siswa baru SMA Brawijaya *Smart School* tahun ajaran 2014/2015. Data-data tersebut akan digunakan untuk menghitung tingkat akurasi sistem akan dibangun.

3.3 Preproses Data

Preproses data merupakan tahapan yang dilakukan untuk menyeleksi data dari semua alternatif yang telah diterkumpul dalam tahap pengumpulan data. Alternatif data yang telah diseleksi kemudian dipisahkan sesuai dengan kriteria-kriteria yang telah ditentukan.

3.4 Analisa Kebutuhan

Analisa kebutuhan dilakukan dengan menentukan kebutuhan apa saja yang dibutuhkan untuk membangun sistem pendukung keputusan. Berikut ini kebutuhan yang digunakan dalam pembuatan sistem pendukung keputusan penerimaa siswa baru SMA BSS:

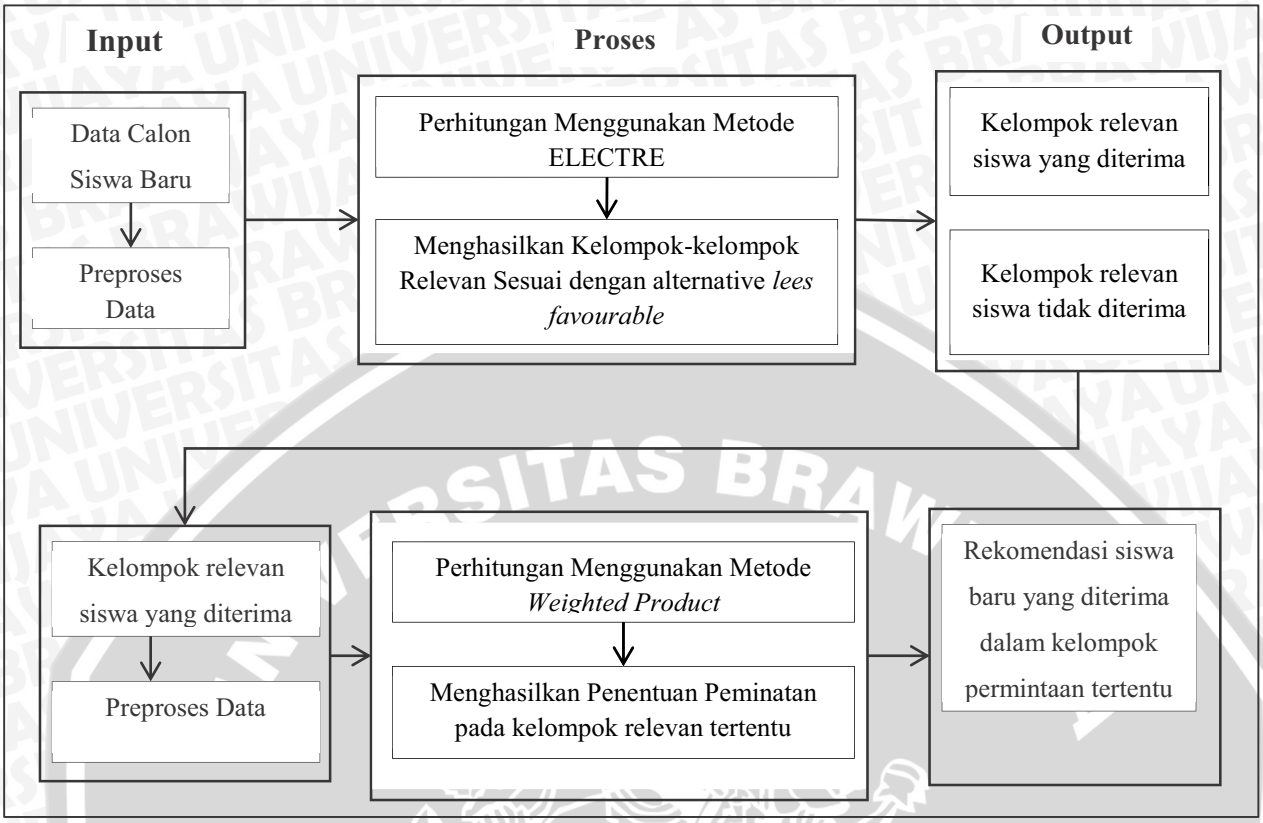
1. Kebutuhan *Hardware*, meliputi:
    - Laptop dengan *Processor* Intel® Core™ i5-2410M CPU @ 2.30GHz
    - Memori 4 GB
2. Kebutuhan *Software*, meliputi:
    - Sistem operasi *Windows 7 Home Premium* 64-bit
    - C#
    - XAMPP
3. Data yang dibutuhkan, meliputi:
    - Data seleksi calon siswa baru SMA Brawijaya *Smart School* Kota Malang

3.5 Perancangan Sistem

Dalam fase ini akan dibuat rancangan langkah kerja dari sistem secara menyeluruh, baik dari segi model maupun dari segi arsitektur untuk mempermudah implementasi dan pengujian. Langkah kerja dalam sistem disesuaikan dengan arsitektur sistem pendukung keputusan.

3.5.1 Model Perancangan Sistem

Model perancangan sistem menjelaskan mengenai cara kerja sistem secara terstruktur mulai dari input yang dimasukkan hingga mendapatkan hasil. Diagram model perancangan sistem dapat dilihat pada Gambar 3.2.



Gambar 3.2 Model Perancanan SPK seleksi penerimaan siswa baru
Sumber : [Perancangan]

Pada gambar 3.2 terdiri dari tiga proses utama, yaitu:

- Input

  Input pada sistem ini yaitu berdasarkan pada kriteria yang digunakan, yaitu nilai rata-rata rapor semester I-V, wawancara anak, wawancara orang tua, nilai tes. Calon siswa sebagai alternatif calon siswa yang diterima, kriteria sebagai parameter yang digunakan untuk menyeleksi siswa baru yang diterima, dan bobot kriteria yang merupakan pembobotan dari pembuat keputusan yaitu pihak SMA BSS.

- Proses

  Proses perhitungan pada penelitian ini menggunakan metode ELECTRE dan metode *Weighted Product* yang hasilnya adalah menentukan calon siswa baru yang layak diterima atau tidak beserta kelas peminatan. Metode ELECTRE akan menghasilkan output kelompok-kelompok yang relevan, pada penelitian ini ELECTRE untuk mengelompokkan calon siswa baru

kedalam kelompok terbaik (A) dan kelompok yang tidak diterima (B). Metode *Weighted Product* akan menghasilkan peminatan dari kelompok data yang ada, pada penelitian ini digunakan untuk mencari alternatif peminatan terbaik dari calon siswa baru yang berada dikelompok diterima(A). Langkah-langkah perhitungan dengan menggunakan metode ELECTRE - *Weighted Product* yaitu diawali dengan menggunakan metode ELECTRE[15] :

1. Menghitung matriks normalisasi
2. Pembobotan pada matriks ternormalisasi
3. Menentukan himpunan *concordace* dan *disordance*
4. Menentukan matriks *concordace* dan *disordance*
5. Menentukan matriks dominan *concordace* dan *disordance*
6. Menentukan *aggregate dominance matrix*
7. Eliminasi alternatif yang *less favourable*

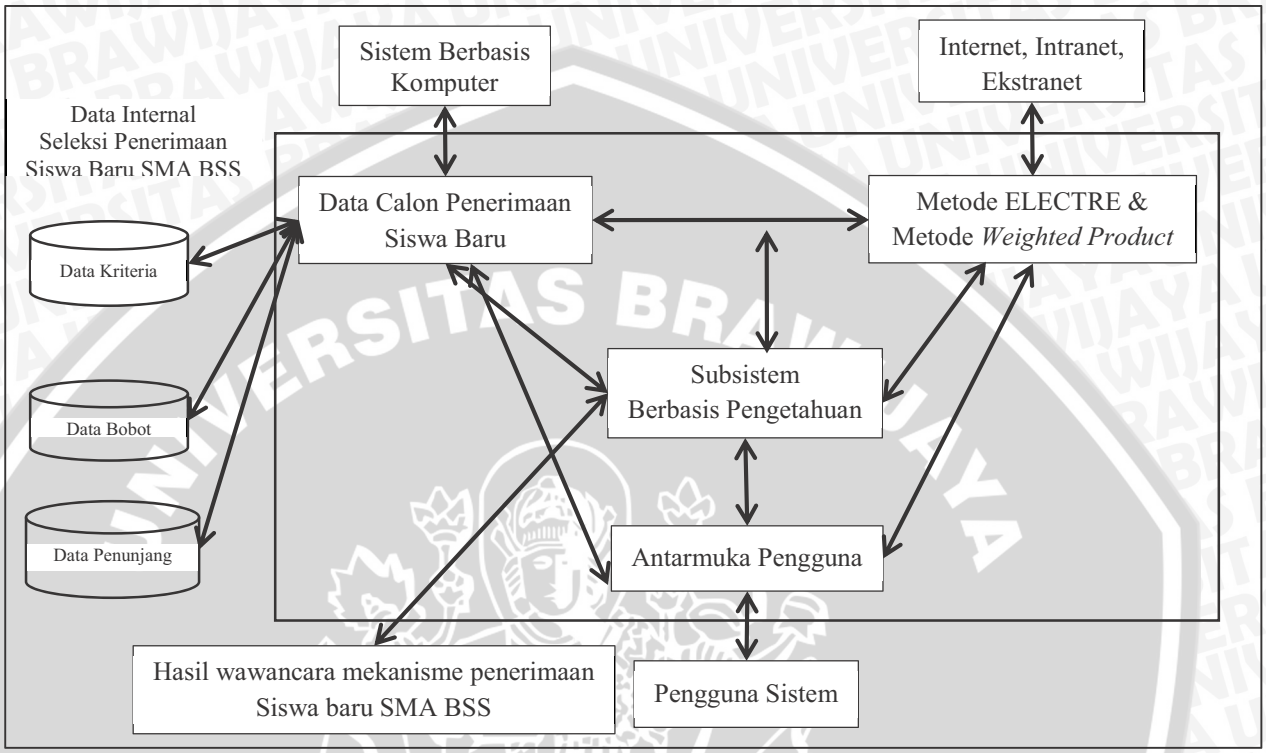
Dari hasil perhitungan eliminasi alternatif yang *less favourable* menggunakan metode ELECTRE, maka akan dibentuk kelompok-kelompok yaitu kelompok yang diterima (A) dan kelompok yang tidak diterima (B). Kelompok yang diterima adalah kelompok yang memiliki nilai e = 1 lebih banyak dari yang lainnya. Sedangkan kelompok yang belum diterima adalah kelompok yang memiliki nilai e = 1 lebih sedikit dari yang lainnya atau tidak memiliki nilai e = 1. Kelompok yang belum diterima akan dihitung kembali menggunakan langkah-langkah metode *Weighted Product* berikut [5] :

1. Pengambilan Data
2. Pembobotan
3. Menghitung vektor S
4. Menentukan Vektor V

Dari hasil perhitungan vektor S menggunakan metode *Weighted Product*, maka akan diketahui kelas peminatan setiap siswa baru yang diterima di SMA BSS. Kelas peminatan dibagi menjadi dua, yakni kelas peminatan IPA/MIA (Matematika dan Ilmu Alam) dan kelas peminatan IPS/IIS (Ilmu-ilmu Sosial)

3.5.2 Arsitektur Pendukung Keputusan

Arsitektur sistem pendukung keputusan yang akan digunakan dapat dilihat pada Gambar 3.3



Gambar 3.3 Arsitektur SPK Penerimaan Siswa Baru SMA BSS
Sumber : [Perancangan]

Gambar 3.3 menjelaskan arsitektur SPK yang mewakili beberapa komponen SPK. Pada subsistem basis pengetahuan menjelaskan proses pembentukan alternatif sesuai dengan kriteria yang telah dibentuk pada basis pengetahuan organisasional. Subsistem manajemen data pada Gambar 3.3 diwakili oleh data eksternal yang berfungsi untuk pengelolaan data calon calon siswa baru. Subsistem manajemen model pada Gambar 3.3 terlihat pada penggunaan metode ELECTRE - *Weighted Product* yang berfungsi untuk menganalisa dan menyelesaikan permasalahan. Antarmuka pengguna berfungsi sebagai perantara antara sistem dan pengguna.

3.6 Implementasi Sistem

Impementasi adalah fase membangun sistem yang mengacu pada perancangan sistem dan menerapkan hal yang telah didapatkan dalam proses studi literatur. Fase-fase yang ada dalam implementasi antara lain:

- Implementasi *interface*, menggunakan *software* C#
- Implementasi basis data, dengan menggunakan sebuah DBMS MySQL pada Server localhost (XAMPP) yang bertujuan untuk memudahkan melakukan manipulasi dan penyimpan data.
- Implementasi algoritma, melakukan perhitungan dengan metode ELECTRE - *Weighted Product* kedalam bahasa pemrograman Java dan menggunakan C#
- Implementasi ini akan menghasilkan keluaran calon siswa baru yang layak diterima dan yang tidak.

3.7 Uji Coba Sistem

Pengujian dilakukan untuk mengetahui apakah sistem telah berjalan sesuai dengan yang diharapkan. Pengujian sistem yang dilakukan antara lain;

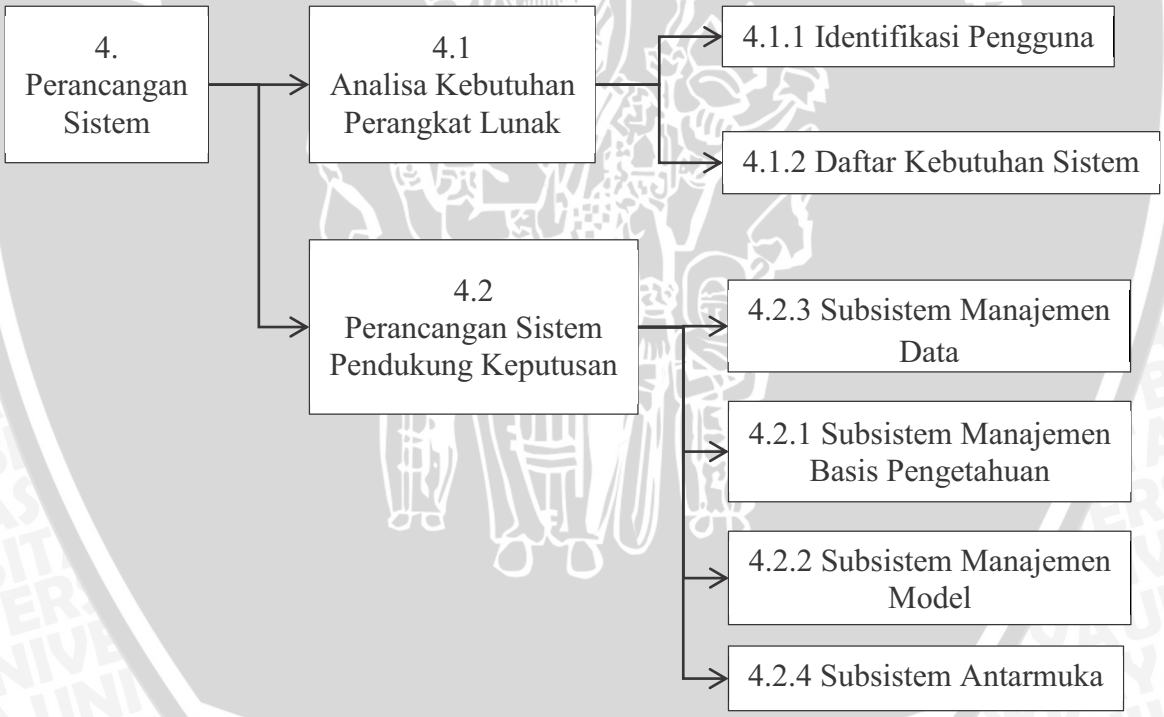
1. Pengujian fungsionalitas, dilakukan dengan menggunakan tabel yang menjelaskan hubungan kesesuaian antara fungsi hasil kerja sistem dengan daftar kebutuhan sistem.
2. Pengujian akurasi, dilakukan dengan membandingan data output sistem perhitungan manual dengan data output perhitungan sistem.

3.8 Kesimpulan

Penarikan kesimpulan dilakukan setelah semua tahapan perancangan, implementasi, dan pengujian terhadap sistem dengan metode ELECTRE - *Weighted Product* telah selesai dikerjakan. Penulisan saran digunakan untuk memperbaiki kekurangan dari sistem yang telah dibangun, sehingga dapat memberikan pertimbangan untuk mengembangkan sistem maupun metode lebih lanjut.

BAB IV
PERANCANGAN

Bab ini menjelaskan tentang analisis kebutuhan dan perancangan Sistem pendukung Keputusan seleksi penerimaan siswa baru SMA menggunakan metode ELECTRE - *Weighted Product*. Tahap analisis kebutuhan perangkat lunak terdiri dari identifikasi pengguna dan daftar kebutuhan sistem, sedangkan tahap perancangan sistem keputusan terdiri dari perancangan subsistem manjemen data, subsistem basis pengetahuan, subsistem manajemen model dan susbsistem antarmuka. Alur perancangan sistem dapat dilihat lebih jelasnya dalam pohon perancangan pada Gambar 4.1.



Gambar 4.1 Pohon Perancangan Sistem
Sumber: [Perancangan]

4.1 Analisa Kebutuhan Perangkat Lunak

Sistem pendukung keputusan seleksi penerimaan calon siswa baru baru SMA dengan menggunakan metode ELECTRE - *Weighted Product* ini merupakan

upaya dalam menangani permasalah seleksi calon siswa baru baru yang terbaik dengan kriteria yang sudah ditentukan. Analisa kebutuhan sistem merupakan tahapan awal perancangan yang diawali dengan mengindetifikasi pengguna yang teribat dalam sistem dan penjabaran kebutuhan sistem. Analisa kebutuhan ini memiliki tujuan untuk memberikan gambaran informasi kebutuhan yangakan digunakan dalam tahapan perancangan. Berikut ini merupakan penjelasan dari masing-masing tahapannya:

4.1.1 Identifikasi Pengguna

Tahapan identifikasi pengguna ini bertujuan untuk mengidentifikasi siapa saja pengguna yang akan berinteraksi dengan sistem. Pasa sistem ini telah ditentukan pengguna yang berinteraksi dengan sistem, yaitu pengguna dan admin. Pada Tabel 4.1 merupakan penjelasan mengenai identifikasi pengguna. Pada Tabel 4.1, kolom pengguna disebutkan pengguna-pengguna yang berperan dalam sistem dan pada kolom lainnya akan menunjukkan deskripsi dari masing-masing pengguna.

Tabel 4.1 Identifikasi Pengguna

Pengguna	Deskripsi Pengguna
Kepala Sekolah (KS)/Kepala PPDB (KP)	Kepala sekolah dan kepala PPDB merupakan pengguna yang mempunyai hak untuk menentukan bobot dari sistem penerimaan calon siswa baru SMA BSS Kota Malang. Kepala sekolah dan kepala PPDB dapat melakukan perubahan terhadap pembobotan yang sudah ada. Selain itu kepala sekolah/Kepala PPDB SMA BSS dapat melihat hasil seleksi dan peminatan yang telah dilakukan oeh Sie Test dan Guru BK. Akses kepala sekolah atau kepala PPDB dapat dilakukan setelah melakukan proses *login* sebagai kepala sekolah atau kepala PPDB.
Sie Test (ST)/ Guru BK (GBK)	Sie Test/Guru BK merupakan pengguna yang mempunyai hak untuk memantau data yang berhubungan dengan calon siswa baru yang hendak diseleksi. Sie Test/Guru BK merupakan pengguna yang memiliki hak untuk menginputkan data calon siswa baru baru, memperbaharui data siswa baru, dan melakukan perhitungan ELECTRE-*Weighted Product* yang menghasilkan output rekomendasi calon siswa baru baru yang diterima beserta kelas peminatannya. Akses Sie Test/Guru BK dapat dilakukan setelah melakukan proses *login* sebagai Sie Test/Guru BK
IT *Service*	IT merupakan pengguna pengguna yang mempunyai hak untuk memantau sistem, seperti mengolah data pengguna, seperti

(IT)	menambah pengguna baru, menghapus pengguna yang telah ada, dan memperbaharui data pengguna. Akses IT dapat dilakukan setelah melakukan proses *login* sebagai IT.

Sumber: Perancangan

4.1.2 Daftar Kebutuhan Sistem

Tahapan ini bertujuan untuk menjelaskan kebutuhan sistem yang harus dipenuhi saat pengguna melakukan sebuah aksi. Daftar kebutuhan ini terdiri dari sebuah kolom yang merupakan hal-hal yang harus disediakan oleh sistem, sedangkan pada kolom yang lain menunjukkan pengguna dan aksi yang menampilkan fungsionalitas masing-masing kebutuhan tersebut. Daftar kebutuhan fungsionalitas sistem dapat dilihat pada Tabel 4.2.

Tabel 4.2 Daftar Kebutuhan Sistem

Kebutuhan	Pengguna	Aksi
Sistem menyediakan antarmuka untuk melakukan *Login*	KS, KP, GBK, IT, ST	Melakukan *Login*
Sistem menyediakan menu untuk mengolah data pengguna	IT	Mengolah data pengguna
Sistem menyediakan proses untuk *insert* data pengguna	IT	*Insert* data pengguna
Sistem menyediakan proses untuk *edit* data pengguna	IT	*Edit* data pengguna
Sistem menyediakan proses untuk *delete* data pengguna	IT	*Delete* data pengguna
Sistem menyediakan menu untuk mengolah data tahun ajaran	IT	Mengolah data tahun ajaran
Sistem menyediakan proses untuk *insert* data tahun ajaran	IT	*Insert* data tahun ajaran
Sistem menyediakan proses untuk *edit* data tahun ajaran	IT	*Edit* data tahun ajaran
Sistem menyediakan proses untuk *delete* data tahun ajaran	IT	*Delete* data tahun ajaran
Sistem menyediakan proses untuk kelola data kelas peminatan	IT	Kelola data kelas peminatan
Sistem menyediakan proses untuk *edit* data kelas peminatan	IT	*Edit* data kelas peminatan
Sistem menyediakan proses untuk kelola data jalur masuk	IT	Kelola data jalur masuk
Sistem menyediakan proses untuk *edit* data jalur masuk	IT	*Edit* data jalur masuk

Sistem menyediakan menu untuk mengolah data kriteria dan bobot seleksi	KS, KP	Mengolah data kriteria dan bobot seleksi
Sistem menyediakan proses untuk *edit* data kriteria dan bobot seleksi	KS, KP	*Edit* data kriteria dan bobot seleksi
Sistem menyediakan menu untuk mengolah data kriteria dan bobot peminatan	KS, KP	Mengolah data kriteria dan bobot peminatan
Sistem menyediakan proses untuk *edit* data kriteria dan bobot peminatan	KS, KP	*Edit* data kriteria dan bobot peminatan
Sitem menyediakan menu untuk melihat hasil rekomendasi seleksi	KS, KP	Menampilkan hasil rekomendasi seleksi
Sitem menyediakan menu untuk melihat hasil rekomendasi peminatan	KS, KP	Menampilkan hasil rekomendasi peminatan
Sistem menyediakan menu untuk mengolah data penerimaan calon siswa baru baru	ST, GBK	Mengolah data penerimaan calon siswa baru baru
Sistem menyediakan proses untuk *insert* data penerimaan calon siswa baru baru	ST, GBK	*Insert* data penerimaan calon siswa baru baru
Sistem menyediakan proses untuk *edit* data penerimaan calon siswa baru baru	ST, GBK	*Edit* data penerimaan calon siswa baru baru
Sistem menyediakan proses untuk *delete* data penerimaan calon siswa baru baru	ST, GBK	*Delete* data penerimaan calon siswa baru baru
Sistem menyediakan proses untuk *insert* data psikotes calon siswa baru baru	ST, GBK	*Insert* data psikotes calon siswa baru baru
Sistem menyediakan proses untuk *edit* data psikotes calon siswa baru baru	ST, GBK	*Edit* data psikotes calon siswa baru baru
Sistem menyediakan proses untuk *delete* data psikotes calon siswa baru baru	ST, GBK	*Delete* data psikotes calon siswa baru baru
Sistem menyediakan menu untuk menampilkan hasil rekomendasi dari proses perhitungan yang telah dilakukan oleh sistem.	ST, GBK, KP, KS	Melihat hasil rekomendasi hasil perhitungan sistem
Sistem menyediakan antarmuka untuk melakukan *Logout*	KS, KP, GBK, IT, ST	*Logout*

Sumber: Perancangan

4.2 Perancangan Sistem Pendukung Keputusan

Tahapan perancangan sistem pendukung keputusan ini bertujuan untuk merancang kinerja sistem dan menentukan fitur-fitur sistem secara keseluruhan berdasar model informasi yang telah dibuat pada tahapan analisa kebutuhan sistem. Pada tahap ini akan dilakukan perancangan subsistem yang terdapat dalam arsitektur sistem pendukung keputusan untuk merekomendasikan siswa baru yang diterima. Subsistem yang akan dirancangan, yaitu susbsistem manajemen data, susbsistem basis pengetahuan, subsistem manajemen model dan subsistem antarmuka. Untuk subsistem manajemen data digunakan beberapa pemodelan data, yaitu *Data Flow Diagram* (DFD), *Entity Relation diagram* (ERD) dan *Physical diagram*.

Pada Gambar 3.3 telah dijelaskan arsitektur sistem pendukung keputusan yang digunakan untuk perancangan sistem. Subsistem yang terdapat dalam sistem pendukung keputusan ini antara lain:

a) Subsistem Basis Pengetahuan

   Menjelaskan tentang kriteria-kriteria yang digunakan pada Sistem Pendukung Keputusan Penerimaan calon siswa baru sebagai bahan pertimbangan dalam melakukan perhitungan dengan metode ELECTRE - *Weighted Product*.

b) Subsistem Manajemen Data

   Menjelaskan tentang pemodelan basis data dan relasi yang akan digunakan Sistem Pendukung Keputusan penerimaan Calon siswa baru perhitungan dengan metode ELECTRE - *Weighted Product*.

c) Subsistem Manajemen model

   Menjelaskan tentang cara perhitungan dan melakukan perancangan algoritma pada Sistem Pendukung Keputusan penerimaan Calon siswa baru dengan metode ELECTRE - *Weighted Product*.

d) Subsistem Antarmuka

   Menjelaskan tentang perancangan desain antarmuka sistem yang baik untuk Sistem Pendukung Keputusan penerimaan Calon siswa baru dengan metode ELECTRE - *Weighted Product*.

Untuk memberikan keputusan, sistem akan menggunakan metode ELECTRE - *Weighted Product* untuk menyeleksi dan merangking alternatif keputusan berdasarkan bobot masing-masing alternatif tersebut. Penggunaan metode ELECTRE - *Weighted Product* didukung oleh basis pengetahuan yang sebelumnya telah disimpan dalam basis data yang merupakan bagian dari subsistem manajemen data.

4.2.1 Subsistem Manajemen Data

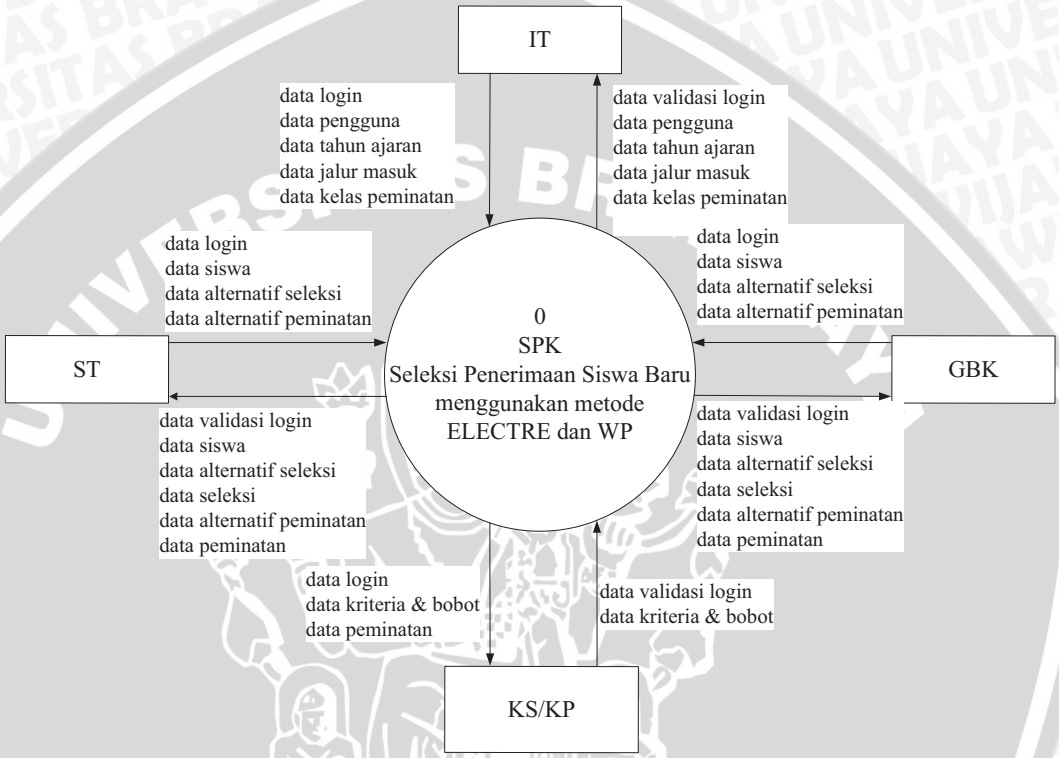
Perancangan yang dibutuhkan dalam sistem manajemen data ini adalah proses aliran data dan basis data. Manajemen data menggunakan data internal, dimana data disimpan dan diakses secara langsung dari *database* yang ada di dalam sistem. Data internal terdiri dari data kriteria, data bobot, dan data penunjang (data yang berisi informasi siswa). Untuk lebih detailnya subsistem manajemen data menggunakan beberapa pemodelan data, yaitu *Data Flow Diagram* (DFD), *Entity Relation diagram* (ERD) dan *Physical Diagram* (PD). DFD digunakan pada proses perancangan aliran data. ERD dan PD digunakan untuk merancang basis data yang sesuai dengan data dari basis pengetahuan untuk digunakan dalam proses perhitungan dengan menggunakan metode ELECTRE - *Weighted Product*.

4.2.1.1 *Data Flow Diagram* (DFD)

Proses aliran data menggambarkan proses yang terjadi antara pengguna dengan sistem. Proses aliran data ini dimodelkan dalam bentuk diagram yang biasa disebut dengan *Data Flow Diagram* (DFD). DFD dalam sistem ini terbagi menjadi beberapa level untuk memodelkannya, level tersebut antara lain: *Context Diagram,* DFD Level 0 dan DFD Level 1. *Context Diagram* merupakan gambaran secara umum proses aliran data yang terjadi dalam sistem, baik aliran data yang masuk maupun keluar dari sistem. DFD Level 0 menggambarkan proses aliran data yang lebih rinci dibandingkan dengan *context diagram*. DFD Level 1 menggambarkan proses aliran data yang lebih rinci yang terjadi dalam sistem. Berikut ini merupakan gambaran dari DFD yang digunakan untuk menggambarkan proses aliran data yang terjadi dalam sistem ini.

4.2.1.1.1 Diagram Konteks

Diagram konteks merupakan gambaran umum aliran data yang terdapat pada sistem. Terdapat dua proses aliran data yaitu aliran data yang berasal dari Panitia PPDB/Kepala Sekolah, Sie Test, Guru BK, dan IT seperti pada Gambar 4.2.



Gambar 4.2 Konteks Diagram
Sumber: [Perancangan]

Berikut ini merupakan penjelasan untuk masing-masing pengguna dalam SPK seleksi penerimaan siswa baru menggunakan metode ELECTRE - *Weighted Product*, antara lain:

- IT merupakan pengguna yang bertugas untuk mengelola data pengguna sistem, tahun ajaran, jalur masuk dan kelas peminatan.
- ST merupakan pengguna yang menggunakan SPK untuk mengolah data calon siswa baru untuk proses seleksi dengan menggunakan metode ELECTRE - *Weighted Product*

- GBK merupakan pengguna yang menggunakan SPK untuk mengolah data calon siswa baru untuk proses peminatan dengan menggunakan metode ELECTRE - *Weighted Product*
- KS/KP merupakan pengguna yang menggunakan SPK untuk mengolah data kriteria dan bobot yang digunakan dalam proses seleksi dan peminatan sekaligus melihat hasil rekomendasi yang diberikan oleh sistem.

Beberapa penggunaan paket data yang terdapat pada *context diagram* diatas, antara lain:

- Paket data login merupakan paket data yang digunakan pada saat proses login yang dilakukan oleh semua pengguna. Paket data login berisi data *username* dan data *password*.
- Paket data validasi login merupakan paket data yang dihasilkan dari proses login yang dilakukan oleh semua pengguna. Paket data validasi login berisi data username dan data hak akses.
- Paket data pengguna merupakan paket data yang berasal dari entitas IT menuju sistem atau sebaliknya dan digunakan untuk menyimpan data pengguna. Paket data pengguna berisi:
  - Data username,
  - Data password, dan
  - Data hak akses
- Data tahun ajaran merupakan data yang berasal dari entitas IT menuju sistem atau sebaliknya dan digunakan untuk menyimpan data tahun ajaran.
- Data jalur masuk merupakan data yang berasal dari entitas IT menuju sistem atau sebaliknya dan digunakan untuk menyimpan data jalur masuk.
- Data kelas peminatan merupakan data yang berasal dari entitas IT menuju sistem atau sebaliknya dan digunakan untuk menyimpan data kelas peminatan.

- Paket data kriteria & bobot paket data yang berasal/menuju dari entitas KS/KP dan digunakan untuk menyimpan kriteria dan nilai bobot masing-masing kriteria. Paket data kriteria & bobot berisi:
  - Data nama kriteria
  - Data nilai bobot
- Paket data siswa merupakan paket data yang berasal dari entitas ST/GBK menuju sistem atau sebaliknya dan digunakan untuk menyimpan data dari peserta didik. Paket data siswa berisi:
  - Data nama peserta didik
  - Data nomor registrasi
  - Data tahun ajaran peserta didik
  - Data jalur tes peserta didik
  - Data jenis kelamin
  - Data asal sekolah
  - Data nilai tes tulis
  - Data nilai rata-rata matematika
  - Data nilai rata-rata IPA
  - Data nilai rata-rata IPS
  - Data nilai rata-rata Agama
  - Data nilai rata-rata B.Indonesia
  - Data nilai rata-rata B.Inggris
  - Data nilai rata-rata IPS
  - Data nilai rata-rata PKn
  - Data nilai rata-rata TIK
  - Data nilai rata-rata Penjas
  - Data nilai rata-rata Mulok
  - Data nilai rata-rata Seni Budaya
  - Data nilai wawancara anak
  - Data nilai wawancara orang tua
  - Data minat anak
  - Data minat orang tua
  - Data nilai tes peminatan IPA

- Data nilai tes peminatan IPS
- Data nilai IQ
- Data nilai aspek kecerdasan daya tangkap
- Data nilai aspek kecerdasan analisa
- Data nilai aspek kecerdasan logika
- Data nilai aspek kecerdasan verbal
- Data nilai aspek kecerdasan hitung
- Data nilai aspek kecerdasan ruang
- Data nilai aspek kecerdasan memori
- Data nilai aspek kecerdasan konsen
- Data nilai aspek kecerdasan mekanik
- Data rekomendasi BK

- Paket data alternatif seleksi merupakan paket yang berasal dari entitas ST menuju sistem atau sebaliknya dan digunakan untuk proses perhitungan seleksi. Paket data alternatif seleksi berisi:
  - Data nilai tes tulis
  - Data nilai rata-rata rapor
  - Data nilai wawancara anak
  - Data nilai wawancara orang tua
  - Data nilai bobot kriteria tes tulis
  - Data nilai bobot kriteria rata-rata rapor
  - Data nilai bobot kriteria wawancara anak
  - Data nilai bobot kriteria wawancara orang tua
- Paket data alternatif peminatan merupakan paket yang berasal dari entitas GBK menuju sistem atau sebaliknya dan digunakan untuk proses perhitungan peminatan. Paket data alternatif peminatan berisi:
  - Data nilai tes tulis
  - Data nilai rata-rata matematika
  - Data nilai rata-rata IPA
  - Data nilai rata-rata IPS
  - Data nilai wawancara anak
  - Data nilai wawancara orang tua

- Data minat anak
- Data minat orang tua
- Data nilai tes peminatan IPA
- Data nilai tes peminatan IPS
- Data nilai IQ
- Data nilai aspek kecerdasan daya tangkap
- Data nilai aspek kecerdasan analisa
- Data nilai aspek kecerdasan logika
- Data nilai aspek kecerdasan verbal
- Data nilai aspek kecerdasan hitung
- Data nilai aspek kecerdasan ruang
- Data nilai aspek kecerdasan memori
- Data nilai aspek kecerdasan konsen
- Data nilai aspek kecerdasan mekanik
- Data rekomendasi BK
- Data nilai bobot kriteria tes tulis
- Data nilai bobot rata-rata matematika
- Data nilai bobot rata-rata IPA
- Data nilai bobot rata-rata IPS
- Data nilai bobot kriteria wawancara anak
- Data nilai bobot kriteria wawancara orang tua
- Data nilai bobot kriteria minat anak
- Data nilai bobot kriteria minat orang tua
- Data nilai bobot tes peminatan IPA
- Data nilai bobot peminatan IPS
- Data nilai bobot kriteria IQ
- Data nilai bobot kriteria aspek kecerdasan daya tangkap
- Data nilai bobot kriteria aspek kecerdasan analisa
- Data nilai bobot kriteria aspek kecerdasan logika
- Data nilai bobot kriteria aspek kecerdasan verbal
- Data nilai bobot kriteria aspek kecerdasan hitung
- Data nilai bobot kriteria aspek kecerdasan ruang

- Data nilai bobot kriteria aspek kecerdasan memori
- Data nilai bobot kriteria aspek kecerdasan konsen
- Data nilai bobot kriteria aspek kecerdasan mekanik
- Data nilai bobot kriteria rekomendasi BK

- Paket data seleksi merupakan paket data yang berasal dari sistem menuju entitas ST dan digunakan untuk menyimpan output dari proses perhitungan seleksi. Paket data seleksi berisi:
  - Data tahun ajaran peserta didik
  - Data nama peserta didik
  - Data nomor registrasi
  - Data jalur tes peserta didik
  - Data *content*
- Paket data peminatan merupakan paket data yang berasal dari sistem menuju entitas GBK dan digunakan untuk menyimpan output dari proses perhitungan peminatan. Paket data peminatan berisi:
  - Data tahun ajaran peserta didik
  - Data nama peserta didik
  - Data nomor registrasi
  - Data jalur tes peserta didik
  - Data hasil perhitungan seleksi
  - Data *content*

4.2.1.1.2 Data Flow Diagram Level 0

Data Flow Diagram Level 0 merupakan rincian dari konteks diagram. Proses dalam DFD level 0 ini lebih detail dimana terdapat delapan proses yang dilakukan sistem secara umum. Delapan proses tersebut dapat dilihat pada gambar 4.3 adalah sebagai berikut :

1.0. Proses Login merupakan proses yang dilakukan oleh semua kelompok pengguna sistem, yaitu Kepala Sekolah, Ketua PPDB, Sie Test, Guru BK dan *IT Service*. Proses ini dilakukan dengan cara memasukkan *username* dan *password*.

2.0. Proses Pengolahan Data Pengguna merupakan proses yang dilakukan oleh pengguna *IT Service*. Proses ini menyediakan antarmuka untuk melakukan lihat, tambah, ubah, dan hapus data pengguna.

3.0. Proses Pengolahan Data Tahun Ajaran merupakan proses yang dilakukan oleh pengguna *IT Service*. Proses ini menyediakan antarmuka untuk melakukan lihat, tambah, ubah dan hapus data tahun ajaran.

4.0. Proses Pengolahan Data Tahun Ajaran merupakan proses yang dilakukan oleh pengguna *IT Service*. Proses ini menyediakan antarmuka untuk melakukan lihat dan ubah data jalur masuk.

5.0. Proses Pengolahan Data Tahun Ajaran merupakan proses yang dilakukan oleh pengguna *IT Service*. Proses ini menyediakan antarmuka untuk melakukan lihat dan ubah data jalur masuk.

6.0. Proses Pengolahan Data Kriteria dan Bobot merupakan proses yang dilakukan oleh pengguna Kepala Sekolah dan Ketua PPDB. Proses ini memberikan antarmuka untuk melakukan lihat dan ubah data kriteria dan bobot.

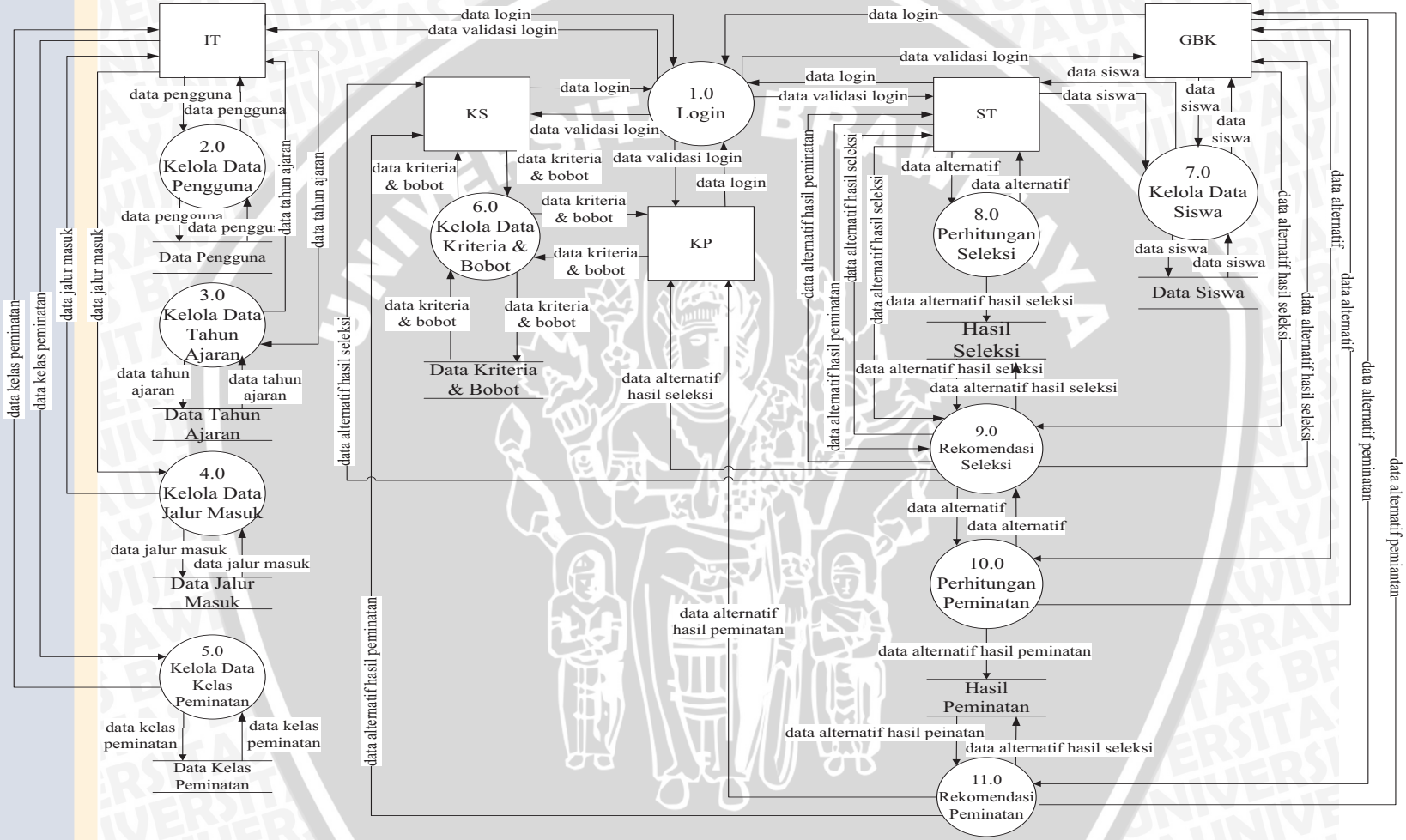
7.0. Proses Pengolahan Data Siswa merupakan proses yang dilakukan oleh pengguna Sie Test dan Guru BK. Proses ini memberikan antarmuka untuk melakukan lihat, tambah, ubah, dan hapus data calon siswa.

8.0. Proses Perhitungan Seleksi merupakan proses perhitungan seleksi yang dihitung berdasarkan masukkan kriteria input yang dimasukkan sebelumnya.

9.0. Proses Perhitungan Peminatan merupakan proses perhitungan peminatan yang dihitung berdasarkan masukkan kriteria input yang dimasukkan sebelumnya.

10.0. Proses Rekomendasi Seleksi merupakan proses yang dilakukan oleh pengguna Sie Test untuk melihat hasil akhir alternatif calon siswa baru yang lolos dalam tahap seleksi.

11.0. Proses Rekomendasi Peminatan merupakan proses yang dilakukan oleh pengguna Guru BK untuk melihat hasil akhir alternatif calon siswa baru yang masuk dalam kelompok peminatan tertentu.

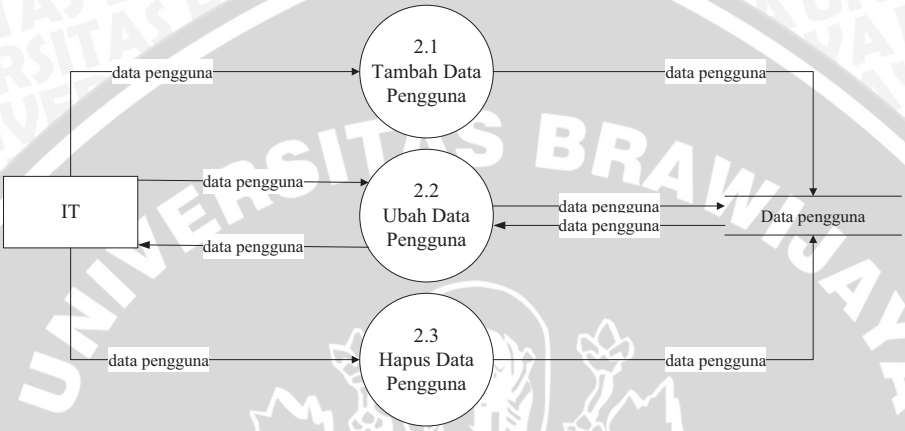


Gambar 4.3 DFD Level 0
Sumber: [Perancangan]

4.2.1.1.3 DFD Level 1

a. DFD Level 1 Proses Kelola Data Pengguna

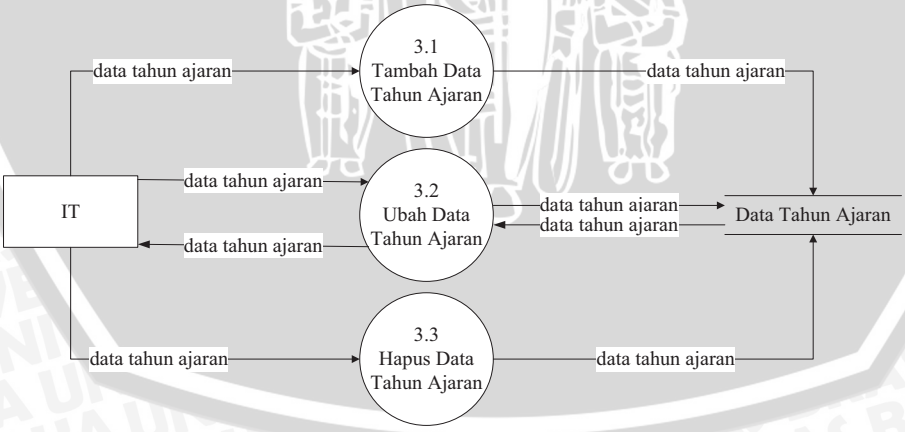
Gambar 4.4 menunjukkan sub proses yang terdapat dalam proses kelola data pengguna. Subproses tersebut didekomposisikan menjadi beberapa proses antara lain tambah, ubah, hapus data pengguna.



Gambar 4.4 DFD Level 1 Proses Kelola Data Pengguna

b. DFD Level 1 Proses Kelola Data Tahun Ajaran

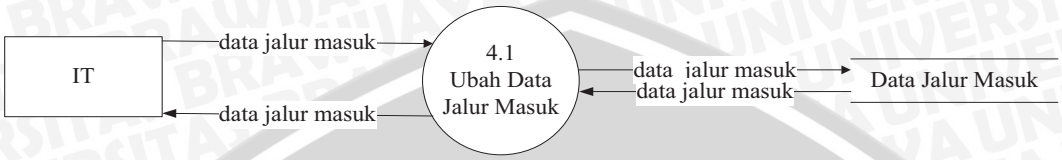
Gambar 4.5 menunjukkan sub proses yang terdapat dalam proses kelola data tahun ajaran. Subproses tersebut didekomposisikan menjadi beberapa proses antara lain tambah, ubah, hapus data tahun ajaran.



Gambar 4.5 DFD Level 1 Proses Kelola Data Tahun Ajaran

c. DFD Level 1 Proses Kelola Data Jalur Masuk

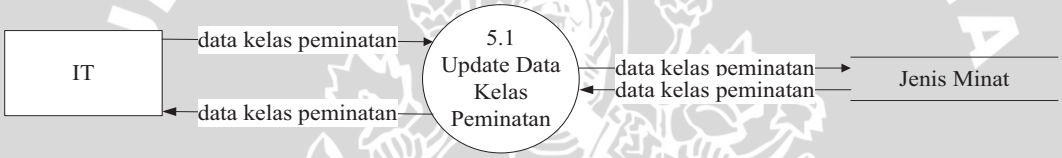
Gambar 4.6 menunjukkan sub proses yang terdapat dalam proses kelola data jalur masuk. Subproses tersebut didekomposisikan menjadi beberapa proses antara lain tambah, ubah, hapus data jalur masuk.



Gambar 4.6 DFD Level 1 Proses Kelola Data Jalur Masuk

d. DFD Level 1 Proses Kelola Data Kelas Peminatan

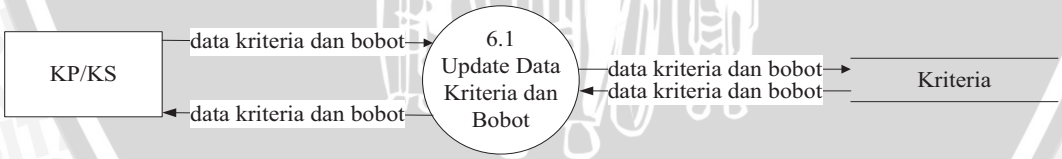
Gambar 4.7 menunjukkan sub proses yang terdapat dalam proses kelola data kelas peminatan. Subproses tersebut didekomposisikan menjadi beberapa proses antara lain tambah, ubah, hapus data peminatan.



Gambar 4.7 DFD Level 1 Proses Kelola Data Jalur Masuk

e. DFD Level 1 Proses Kriteria dan Bobot

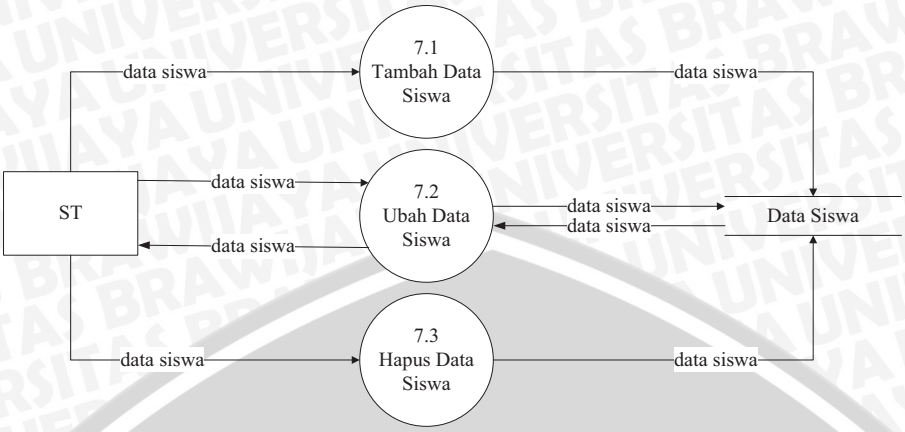
Gambar 4.7 menunjukkan sub proses yang terdapat dalam proses kelola data kriteria dan bobot. Subproses tersebut didekomposisikan menjadi beberapa proses antara lain ubah data kriteria dan bobot.



Gambar 4.8 DFD Level 1 Proses Kelola Data Kriteria dan Bobot

f. DFD Level 1 Proses Kelola Data Siswa

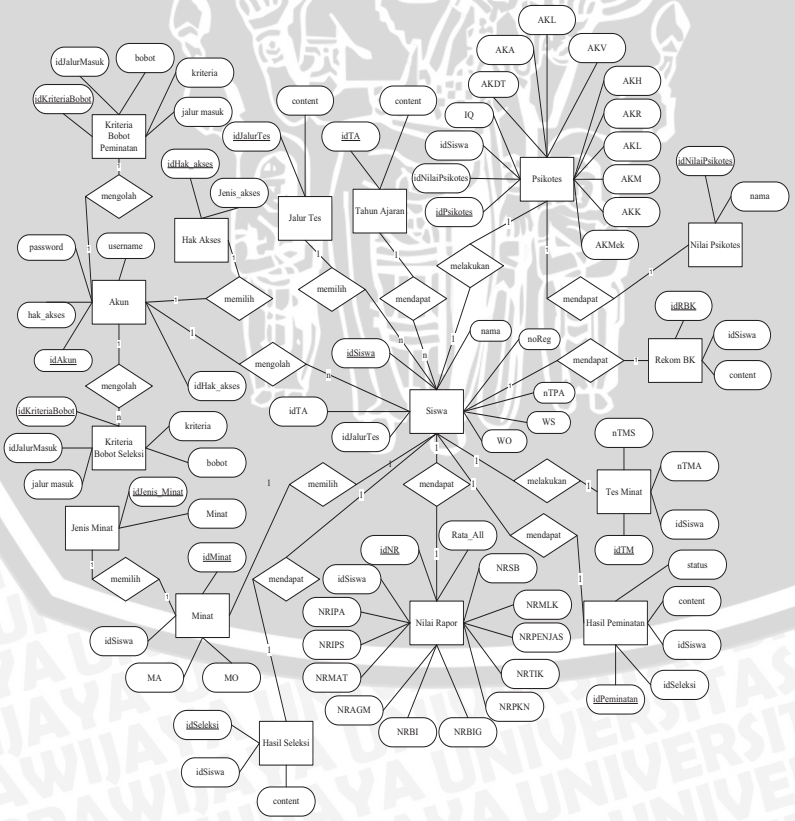
Gambar 4.9 menunjukkan sub proses yang terdapat dalam proses kelola data siswa. Subproses tersebut didekomposisikan menjadi beberapa proses antara lain tambah, ubah, hapus data siswa.



Gambar 4.9 DFD Level 1 Proses Kelola Data Siswa

4.2.1.2 *Entity Relation Diagram* (ERD)

Pada perancangan basis data dari sistem ini digunakan enam tabel yang berfungsi untuk menyimpan data yang berasal dari basis pengetahuan sebelumnya. Adapun perancangan basis data dengan menggunakan pemodelan *Entity Relationship Diagram* sistem ini dapat dilihat pada Gambar 4.10.



Gambar 4.10. *Entity Relationship Diagram*
Sumber : [Perancangan]

Setiap pengguna dalam sistem memiliki perannya masing-masing sesuai dengan otoritasnya yaitu IT, ST, KP dan KS. KP dan KS disini dapat melakukan penilaian kepentingan kriteria serta nilai bobotnya yang akan disimpan pada entitas 'kriteria bobot'. Entitas 'kriteria bobot' ini mempunyai *primary key*. Hasil dari pengisian 'kriteria bobot' adalah 'nama', dan 'nilaiBobot'.

Pengguna ST dan GBK dapat melakukan pengelolaan data siswa pada enam entitas yakni:

- Entitas 'Peserta Didik'
- Entitas 'Psikotes'
- Entitas 'Nilai Rapor'
- Entitas 'Peminatan
- Entitas 'Rekom BK'
- Entitas 'Tes Minat'

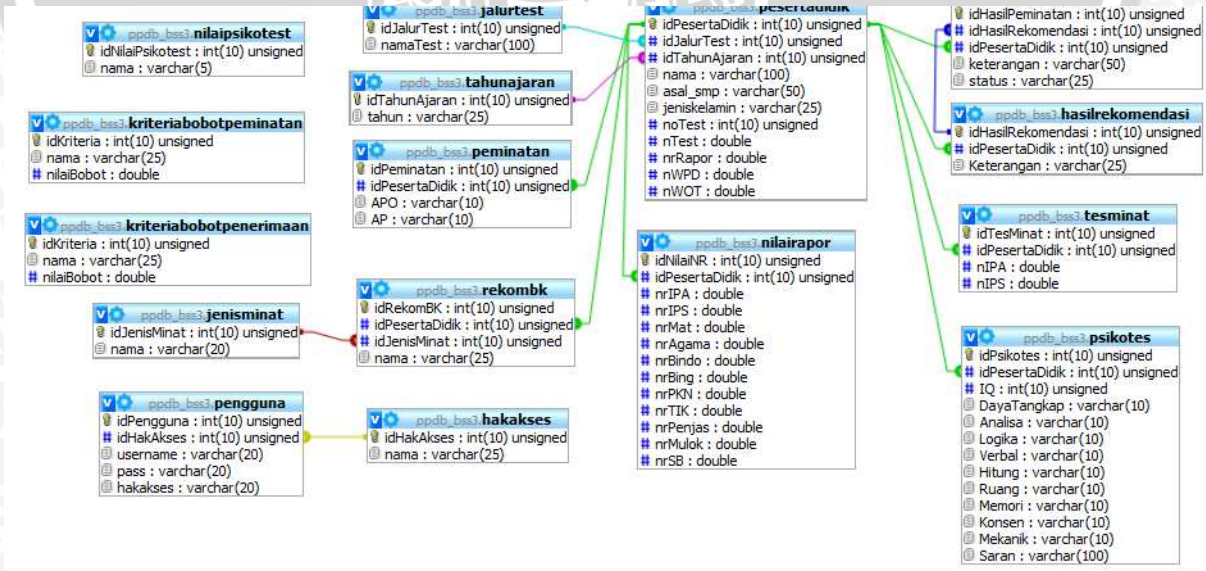
ST dan GBK melakukan perhitungan hingga menghasilkan hasil rekomendasi pada entitas 'Hasil Rekomendasi'dan 'Hasil Peminatan'. Entitas 'Peserta Didik' ini mempunyai primary key idPesertaDidik dan enam foreign key yang diambil dari relasi antar entitas 'Peserta Didik' dengan entitas 'Jalur Test', 'Tahun Ajaran', 'Psikotes', 'Nilai Rapor', 'Peminatan', 'Rekom BK', dan 'Tes Minat'. Dimana satu entitas 'Rekom BK', 'Tes Minat', 'Psikotes', 'Nilai Rapor', 'Peminatan' direlasikan dengan beberapa entitas 'Calon siswa baru', sehingga setiap siswa akan mempunyai nilai tes minat, rekomendasi BK, nilai rapor, psikotes, dan peminatan. Entitas 'Nilai Rapor' memiliki *primary key* idNilaiNR yang dapat direlasikan dengan entitas 'Peserta Didik'. Entitas 'Psikotes' memiliki *primary key* idPsikotes yang dapat direlasikan dengan entitas 'Peserta Didik'. Entitas 'Peminatan' *primary key* idPeminatan yang dapat direlasikan dengan entitas 'Peserta Didik'. Entitas 'Tes Minat' memiliki *primary key* idTesMinat yang dapat direlasikan dengan entitas 'Peserta Didik'. Entitas 'Rekom BK' *primary key* idRekomBK yang dapat direlasikan dengan entitas 'Peserta Didik'.

Entitas 'Hasil Rekomendasi' berelasi dengan entitas 'Peserta Didik' yang merupakan *master* data calon siswa baru. Setiap nilai yang diambil dari entitas 'Peserta Didik' akan dilakukan proses perhitungan seleksi dengan menggunakan

nilai yang diambil dari entitas ‘Kriteria Bobot’ yaitu masing-masing nilai kriteria dan bobot yang digunakan. Hasil dari proses perhitungan seleksi akan disimpan ke dalam entitas ‘Hasil Rekomendasi’ yang mempunyai primary key idHasilRekomendasi, foreign key idPesertaDidik siswa baru yang didapatkan dari relasi entitas ‘Peserta Didik’, dan keterangan untuk memberikan keterangan diterima atau tidak. Hasil dari proses perhitungan peminatan akan disimpan ke dalam entitas ‘Hasil Peminatan’ yang mempunyai primary key idHasilPeminatan, foreign key idPesertaDidik siswa baru yang didapatkan dari relasi entitas ‘Peserta Didik’, dan keterangan untuk memberikan keterangan hasil kelas peminatan.Satu entitas ‘Peserta Didik’ direlasikan dengan beberapa entitas ‘Hasil Rekomendasi’ dan ‘Hasil peminatan’, sehingga setiap siswa akan mempunyai satu hasil rekomendasi.

4.2.1.3 *Physical Diagram* (PD)

Sistem ini menggunakan MySQL sebagai *Server Database Management System* (DBMS). Dibuatnya *physical diagram* ini bertujuan untuk mempermudah dalam pengembangan dan pengelolaan *database* serta mempermudah dalam membuat *query Data Definition Language* (DDL). Gambar 4.11 menunjukkan *physical diagram* yang dirancang untuk membangun SPK seleksi penerimaan calon siswa baru baru ini.



Gambar 4.11. *Physical Diagram*
Sumber : [Perancangan]

Struktur dari masing-masing tabel dalam *Physical Diagram* SPK seleksi penerimaan calon siswa baru baru sebagai berikut:

1. **Tabel 'Calon siswa baru'**
   Tabel calon siswa baru ini digunakan untuk menyimpan nilai kriteria-kriteria yang digunakan dalam sistem serta data yang diperlukan dalam seleksi penerimaan calon siswa baru baru. Tabel calon siswa baru dapat dilihat pada Tabel 4.3.

Tabel 4.3 Struktur Tabel Calon siswa baru

No	Nama Field	Tipe	Lebar	Keterangan
1	idPesertaDidik	INTEGER		Kode calon siswa baru(*Primary Key*)
2	idJalurTest	INTEGER		Kode jalur test (*Foreign Key*)
3	idTahunAjaran	INTEGER		Kode tahun ajaran (*Foreign Key*)
4	nama	VARCHAR	100	Nama dari calon siswa baru
5	asal_smp	VARCHAR	50	Asal Sekolah dari calon siswa baru
6	jeniskelamin	VARCHAR	25	Jenis kelamin dari calon siswa baru
7	NoTest	INTEGER	10	Nomer test dari calon siswa baru
8	nTest	DOUBLE	100	Nilai TPA dari calon siswa baru
9	nRapor	DOUBLE		Rata-rata rapor dari Calon siswa baru
10	nWPD	DOUBLE		Jumlah nilai wawancara Calon siswa baru
11	nWOT	DOUBLE		Jumlah nilai wawancara orang tua Calon siswa baru

Sumber: [Perancangan]

2. **Tabel 'Tahun Ajaran'**
   Tabel tahun ajaran digunakan untuk menyimpan tahun ajaran baru seleksi penerimaan pserta didik yang digunakan dalam sistem. Tabel tahun ajaran dapat dilihat pada Tabel 4.4.

Tabel 4.4 Struktur Tabel Tahun Ajaran

No	Nama Field	Tipe	Lebar	Keterangan
1	idTahunAjaran	INTEGER		Kode tahun ajaran (*Primary Key*)
2	tahun	VARCHAR	25	Tahun ajaran (tahun)

Sumber: [Perancangan]

3. **Tabel 'Jalur Test'**

Tabel jalur test digunakan untuk menyimpan jalur test seleksi penerimaan pserta didik yang digunakan dalam sistem. Tabel jalur test dapat dilihat pada Tabel 4.5.

Tabel 4.5 Struktur Tabel Jalur Test

No	Nama Field	Tipe	Lebar	Keterangan
1	idJalurTest	INTEGER		Kode jalur test (*Primary Key*)
2	namaTest	VARCHAR	100	Tahun test

Sumber: [Perancangan]

4. **Tabel 'Jenis Minat'**

Tabel jenis minat digunakan untuk menyimpan jenis minat baru seleksi penerimaan pserta didik yang digunakan dalam sistem. Tabel jenis minat dapat dilihat pada Tabel 4.6.

Tabel 4.6 Struktur Tabel Jenis Minat

No	Nama Field	Tipe	Lebar	Keterangan
1	idJenisMinat	INTEGER		Kode jenis minat (*Primary Key*)
2	nama	VARCHAR	20	Nama jenis minat

Sumber: [Perancangan]

5. **Tabel 'Nilai Psikotest'**

Tabel nilai psikotest digunakan untuk menyimpan nilai psikotest seleksi penerimaan pserta didik yang digunakan dalam sistem. Tabel nilai psikotest dapat dilihat pada Tabel 4.7.

Tabel 4.7 Struktur Tabel Nilai Psikotest

No	Nama Field	Tipe	Lebar	Keterangan
1	idNilaiPsikotest	INTEGER		Kode nilai psikotest (*Primary Key*)
2	nama	VARCHAR	5	Nilai psikotest

Sumber: [Perancangan]

6. **Tabel 'Hak Akses'**

Tabel hak akses digunakan untuk menyimpan hak akses untuk seleksi penerimaan pserta didik yang digunakan dalam sistem. Tabel hak akses dapat dilihat pada Tabel 4.8.

Tabel 4.8 Struktur Tabel Tahun Ajaran

No	Nama Field	Tipe	Lebar	Keterangan
1	idHakAkses	INTEGER		Kode hak akses (*Primary Key*)
2	nama	VARCHAR	25	Jenis hak akses

Sumber: [Perancangan]

7. **Tabel 'Kriteria Bobot Penerimaan'**

Tabel kriteria bobot digunakan untuk menyimpan kriteria bobot yang digunakan dalam sistem. Tabel kriteria bobot dapat dilihat pada Tabel 4.9.

Tabel 4.9 Struktur Tabel Kriteria Bobot Penerimaan

No	Nama Field	Tipe	Lebar	Keterangan
1	idKriteria	INTEGER		Kode kriteria (*Primary Key*)
2	nama	VARCHAR	25	Keterangan dari masing-masing bobot
3	nilaiBobot	DOUBLE		Nilai dari masing-masing bobot

Sumber: [Perancangan]

8. **Tabel 'Kriteria Bobot Peminatan'**

Tabel kriteria bobot digunakan untuk menyimpan kriteria bobot peminatan yang digunakan dalam sistem. Tabel kriteria bobot peminatan dapat dilihat pada Tabel 4.10.

Tabel 4.10 Struktur Tabel Kriteria Bobot Peminatan

No	Nama Field	Tipe	Lebar	Keterangan
1	idKriteria	INTEGER		Kode kriteria (*Primary Key*)
2	nama	VARCHAR	25	Keterangan dari masing-masing bobot
3	nilaiBobot	DOUBLE		Nilai dari masing-masing bobot

Sumber: [Perancangan]

9. Tabel 'Hasil Rekomendasi'

Tabel hasil rekomendasi digunakan untuk menyimpan data hasil rekomendasi yang digunakan dalam sistem. Tabel hasil rekomendasi dapat dilihat pada tabel 4.11.

Tabel 4.11 Struktur Tabel Admin

No	Nama Field	Tipe	Lebar	Keterangan
1	idHasilRekomendasi	INTEGER		Kode hasil rekomendasi (*Primary Key*)
2	idPesertaDidik	INTEGER		Kode calon siswa baru (*Foreign Key*)
3	Keterangan	VARCHAR	25	Keterangan dari masing-masing calon siswa baru

Sumber: [Perancangan]

10. Tabel 'Hasil Peminatan'

Tabel hasil peminatan digunakan untuk menyimpan data hasil peminatan yang digunakan dalam sistem. Tabel hasil peminatan dapat dilihat pada tabel 4.12.

Tabel 4.12 Struktur Tabel Admin

No	Nama Field	Tipe	Lebar	Keterangan
1	idHasilPeminatan	INTEGER		Kode hasil peminatan (*Primary Key*)
2	idHasilRekomendasi	INTEGER		Kode hasil rekomendasi (*foreign key*)
3	idPesertaDidik	INTEGER		Kode calon siswa baru (*Foreign Key*)
4	Keterangan	VARCHAR	50	Keterangan dari masing-masing calon siswa baru
5	status	VARCHAR	25	Status dari masing-masing calon siswa baru

Sumber: [Perancangan]

11. Tabel 'Pengguna'

Tabel pengguna digunakan untuk menyimpan data pengguna yang digunakan dalam sistem. Tabel pengguna dapat dilihat pada tabel 4.13.

Tabel 4.13 Struktur Tabel Pengguna

No	Nama Field	Tipe	Lebar	Keterangan
1	idPengguna	INTEGER	10	Kode pengguna (*Primary Key*)
2	idHakAkses	INTEGER	10	Kode hak akses (*foreign Key*)
3	username	VARCHAR	20	Username yang digunakan oleh pengguna
4	Pass	VARCHAR	20	Password yang digunakan oleh pengguna
5	hakakses	VARCHAR	20	Hak akses yang diberikan oleh pengguna

Sumber: [Perancangan]

12. Tabel 'Nilai Rapor'

Tabel nilai rapor digunakan untuk menyimpan data nilai rapor calon siswa baru yang digunakan dalam sistem. Tabel nilai rapor dapat dilihat pada tabel 4.14.

Tabel 4.14 Struktur Tabel Nilai Rapor

No	Nama Field	Tipe	Lebar	Keterangan
1	idNilaiNR	INTEGER	10	Kode nilai rapor (*Primary Key*)
2	idPesertaDidik	INTEGER	10	Kode calon siswa baru (*foreign Key*)
3	nrIPA	DOUBLE		Nilai rata-rata IPA calon siswa baru
4	nrIPS	DOUBLE		Nilai rata-rata IPS calon siswa baru
5	nrMat	DOUBLE		Nilai rata-rata matematika calon siswa baru
6	nrBindo	DOUBLE		Nilai rata-rata bahasa inggris calon siswa baru
7	nrAgama	DOUBLE		Nilai rata-rata agama calon siswa baru
8	nrBing	DOUBLE		Nilai rata-rata bahasa inggris calon siswa baru

9	nrPKN	DOUBLE		Nilai rata-rata PKN calon siswa baru
10	nrTIK	DOUBLE		Nilai rata-rata teknologi informasi calon siswa baru
11	nrSB	DOUBLE		Nilai rata-rata seni budaya calon siswa baru
12	nrPenjas	DOUBLE		Nilai rata-rata olah raga calon siswa baru
13	nrMulok	DOUBLE		Nilai rata-rata muatan lokal calon siswa baru

Sumber: [Perancangan]

13. Tabel 'Psikotes'

Tabel psikotes digunakan untuk menyimpan hasil psikotes data calon siswa baru yang digunakan dalam sistem. Tabel psikotest dapat dilihat pada tabel 4.15.

Tabel 4.15 Struktur Tabel Psikotes

No	Nama Field	Tipe	Lebar	Keterangan
1	idPsikotes	INTEGER		Kode psikotes (*Primary Key*)
2	idPesertaDidik	INTEGER		Kode calon siswa baru (*foreign Key*)
3	IQ	VARCHAR	5	Nilai IQ calon siswa baru
4	Daya tangkap	VARCHAR	5	Nilai kemampuan daya tangkap calon siswa baru
5	Analisa	VARCHAR	5	Nilai kemampuan analisa calon siswa baru
6	Logika	VARCHAR	5	Nilai kemampuan logika calon siswa baru
7	Verbal	VARCHAR	5	Nilai kemampuan verbal calon siswa baru
8	Hitung	VARCHAR	5	Nilai kemampuan hitung calon siswa baru

9	Ruang	VARCHAR	5	Nilai kemampuan ruang calon siswa baru
10	Memori	VARCHAR	5	Nilai kemampuan memori calon siswa baru
11	Konsen	VARCHAR	5	Nilai kemampuan konsentrasi calon siswa baru
12	Mekanik	VARCHAR	5	Nilai kemampuan mekanik calon siswa baru
13	Saran	VARCHAR	100	Saran yang diberikan oleh psikolog

Sumber: [Perancangan]

14. Tabel 'Peminatan'

Tabel peminatan digunakan untuk menyimpan data peminatan calon siswa baru yang digunakan dalam sistem. Tabel peminatan dapat dilihat pada tabel 4.16.

Tabel 4.16 Struktur Tabel Peminatan

No	Nama Field	Tipe	Lebar	Keterangan
1	idPeminatan	INTEGER		Kode peminatan(*Primary Key*)
2	idPesertaDidik	INTEGER		Kode calon siswa baru(*foreign Key*)
3	APO	VARCHAR	10	Hasil pemintanan orangtua calon siswa baru
4	AP	VARCHAR	10	Hasil pemintanan calon siswa baru

15. Tabel 'Rekomendasi BK'

Tabel rekomendasi BK ini digunakan untuk menyimpan hasil rekomendasi BK yang diperlukan dalam seleksi penerimaan calon siswa baru baru. Tabel rekomendasi BK dapat dilihat pada Tabel 4.17.

Tabel 4.17 Struktur Tabel Rekomendasi BK

No	Nama Field	Tipe	Lebar	Keterangan
1	idRekomBK	INTEGER		Kode rekomendasi BK(*Primary Key*)
2	idPesertaDidik	INTEGER		Kode calon siswa baru(*foreign Key*)

3	idJenisMinat	INTEGER		Kode jenis minat (*Foreign Key*)
4	nama	VARCHAR	100	Nama dari calon siswa baru

16. Tabel 'Tes Minat'

Tabel admin digunakan untuk menyimpan data tes peminatan calon siswa baru yang digunakan dalam sistem. Tabel tes minat dapat dilihat pada tabel 4.18.

Tabel 4.18 Struktur Tabel Tes IPS

No	Nama Field	Tipe	Lebar	Keterangan
1	idPeminatan	INTEGER		Kode peminatan(*Primary Key*)
2	idPesertaDidik	INTEGER		Kode calon siswa baru(*foreign Key*)
3	nIPA	DOUBLE		Hasil peminatan IPA calon siswa baru
4	nIPS	DOUBLE		Hasil pemintanan IPS calon siswa baru

4.2.2 Subsistem Basis Pengetahuan

Subsistem basis pengetahuan merupakan pengembangan dari Basis Pengetahuan. Basis pengetahuan merupakan hasil wawancara dengan SMA BSS (yang ditunjukkan pada lampiran wawancara) untuk mengetahui proses penerimaan siswa baru secara keseluruhan sehingga dapat dikembangan menjadi subsistem basis pengetahuan yang lebih terperinci dan detail. Subsistem basis pengetahuan ini merupakan subsistem yang dapat mendukung subsistem lainnya. Berdasarkan arsitektur SPK, sistem ini memerlukan pengetahuan mengenai penerimaan calon siswa baru baru yaitu kriteria-kriteria yang digunakan dalam proses pengambilan keputusan beserta pembobotannya. Subsistem ini mendukung penyimpanan data yang berada dalam subsistem manajemen data yaitu basis data SPK di MySQL. Pemodelan yang digunakan untuk memproses perhitungan data yaitu metode ELECTRE-*Weighted Product*. Kriteria yang akan digunakan untuk pengambilan keputusan penerimaan calon siswa baru baru berdasarkan hasil wawancara yang

dilakukan pada SMA BSS dapat dilihat pada tabel 4.19 dan sesuai dengan penjelasan pada Subbab 2.2.2

Tabel 4.19 Kriteria Penerimaan Calon Siswa Baru

Kriteria		Penjelasan
C1	Nilai rata-rata rapor	Nilai rata-rata rapor SMP semester I-V (0-100)
C2	Jumlah nilai tes	Nilai tes berupa Tes Potensial Akademik (0-100)
C3	Wawancara calon siswa baru	Penilaian wawancara calon siswa baru (0-100)
C4	Wawancara orang tua	Penilaian wawancara orang tua (0-45)

Sumber: [Perancangan]

Data pembobotan dalam sistem merupakan pembobotan yang telah ditentukan oleh pihak SMA BSS. Data pembobotan ini dilakukan untuk kriteria-kriteria yang akan digunakan dalam proses perhitungan nantinya, yaitu kriteria C1 sampai dengan C4. Pembobotan ini digunakan untuk proses seleksi penerimaan melalui jalur tes. Pembobotan dilakukan dengan menggunakan bobot secara presentase yang dapat dilihat pada Tabel 4.20. dibawah ini.

Tabel 4.20. Data Pembobotan Kriteria

Kriteria	Nilai bobot (w) dalam persem
Nilai Tes	40
Nilai Rapor	30
Wawancara anak	15
Wawancara Orang tua	15

Sumber: [Perancangan]

Kriteria yang akan digunakan untuk pengambilan keputusan peminatan calon siswa baru baru berdasarkan hasil wawancara yang dilakukan pada SMA BSS dapat dilihat pada Tabel 4.21 dan sesuai dengan penjelasan pada Subbab 2.2.3.

Tabel 4.21. Kriteria Peminatan Calon Siswa Baru

Kriteria		Penjelasan
C1	Nilai rata-rata Matematika	Nilai rata-rata rapor mapel Matematika semester I-V (0-100)
C2	Nilai rata-rata IPA	Nilai rata-rata rapor mapel IPA semester I-V (0-100)
C3	Nilai rata-rata IPS	Nilai rata-rata rapor mapel IPA semester I-V (0-100)

C4	Nilai tes tulis	Nilai tes potensial akademik (0-100)
C5	Wawancara calon siswa baru	Penilaian wawancara calon siswa baru (0-100)
C6	Wawancara orang tua	Penilaian wawancara orang tua (0-45)
C7	Nilai tes peminatan IPA	Hasil *placement test* mata pelajaran IPA (0-100)
C8	Nilai tes peminatan IPS	Hasil *placement test* mata pelajaran IPS (0-100)
C9	Angket calon siswa baru	Pertanyaan-pertanyaan yang berhubungan dengan peminatan calon siswa baru (untuk mengetahui lebih jauh minat IPA/IPS calon siswa baru)
C10	Angket orang tua	Pertanyaan-pertanyaan yang berhubungan dengan peminatan calon siswa baru (untuk mengetahui lebih dukungan orang tua terhadap minat IPA/IPS calon siswa baru)
C11	IQ	Nilai IQ calon siswa baru (70-130)
C12	Daya tangkap	Hasil kemampuan daya tangkap calon siswa baru berdasarkan psikotes (R-T)
C13	Analisa	Hasil kemampuan analisa calon siswa baru berdasarkan psikotes (R-T)
C14	Logika	Hasil kemampuan logika calon siswa baru berdasarkan psikotes (R-T)
C15	Verbal	Hasil kemampuan verbal calon siswa baru berdasarkan psikotes (R-T)
C16	Hitung	Hasil kemampuan hitung calon siswa baru berdasarkan psikotes (R-T)
C17	Ruang	Hasil kemampuan ruang calon siswa baru berdasarkan psikotes (R-T)
C18	Memori	Hasil kemampuan memori calon siswa baru berdasarkan psikotes (R-T)
C19	Konsentrasi	Hasil kemampuan konsentrasi calon siswa baru berdasarkan psikotes (R-T)
C20	Mekanik	Hasil kemampuan mekanik calon siswa baru berdasarkan psikotes (R-T)
C21	Rekomendasi BK	Rekomendasi dari guru BK SMP (IPA/IPS)

Sumber: [Perancangan]

Untuk mempermudah proses perhitungan, maka dilakukan konversi nilai terhadap nilai inputan yang berbentuk angka maupun inputan yang bukan berbentuk angka. Konversi nilai dapat dilihat sebagai berikut.

Tabel 4.22 Range Kelompok Peminatan IPA

Kelompok Peminatan	Konversi
IPA	3

IPS	2
Terserah	1

Sumber: [Perancangan]

Tabel 4.23 Range Kelompok Peminatan IPS

Kelompok Peminatan	Konversi
IPA	2
IPS	3
Terserah	1

Sumber: [Perancangan]

Tabel 4.24 Range Nilai Aspek Psikotes

Nilai Aspek Psikotes	Konversi
R	1
C-	2
C	3
C+	4
T	5

Sumber: [Perancangan]

Keputusan yang menjadi keluaran dari perhitungan sistem didapat dari perhitungan kriteria pengambilan keputusan yang didasarkan pada alternatif keputusan yang telah ditetapkan. Alternatif keputusan yang menjadi output dapat dilihat pada Tabel 4.25.

Tabel 4.25 Alternatif Keputusan

Keputusan	Kriteria yang digunakan
A1 (IPA)	C1, C2, C4, C5, C6, C7, C9, C10, C11, C12, C14, C16, C19, C20, C21
A2 (IPS)	C1, C3, C4, C5, C6, C8, C9, C10, C11, C13, C15, C17, C18, C19, C21

Sumber: [Perancangan]

Sedangkan bobot yang digunakan pada proses pemilihan kelas peminatan, bobot yang digunakan dapat dilihat pada Tabel 4.26.

Tabel 4.26 Bobot Kriteria Peminatan

	Kriteria	Bobot
C1	Nilai rata-rata matematika	1

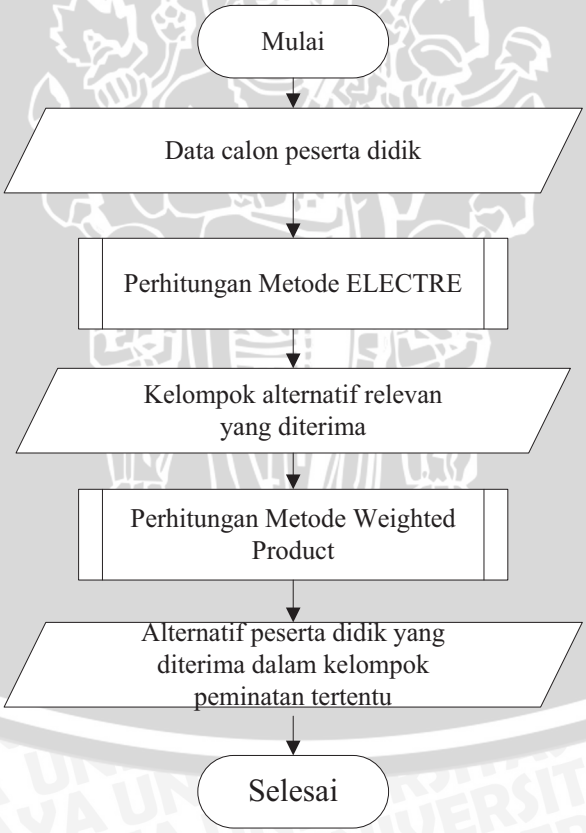
C2	Nilai rata-rata IPA	1
C3	Nilai rata-rata IPS	1
C4	Nilai tes tulis	1
C5	Nilai wawancara calon calon siswa baru	1
C6	Nilai wawancara orang tua	1
C7	Nilai tes peminatan IPA	1
C8	Nilai tes peminatan IPS	1
C9	Angket peminatan anak	1
C10	Angket dukungan orang tua	1
C11	IQ	1
C12	Aspek kecerdasan daya tangkap	1
C13	Aspek kecerdasan analisa	1
C14	Aspek kecerdasan logika	1
C15	Aspek kecerdasan verbal	1
C16	Aspek kecerdasan hitung	1
C17	Aspek kecerdasan ruang	1
C18	Aspek kecerdasan memori	1
C19	Aspek kecerdasan konsen	1
C20	Aspek kecerdasan mekanik	1
C21	Rekomendasi BK	1

Seluruh kriteria memiliki nilai bobot satu. Hal ini disebabkan karena seluruh kriteria memiliki tingkat kepentingan yang sama. Berdasarkan hasil wawancara dengan SMA BSS, C1 memiliki tingkat kepentingan yang sama dengan C2 hingga C21, dan seterusnya. SMA BSS tidak bisa menentukan tingkat kepentingan antar kriteria karena SMA BSS melihat kriteria secara keseluruhan. Sehingga tidak dimungkinkan untuk memberikan tingkat kepentingan antar kriteria. Hal inilah yang menyebabkan proses penentuan nilai bobot tidak bisa menggunakan Teknik Pemberian Bobot yang telah dijelaskan pada Subbab 2.10. Pada Teknik Pemberian Bobot, baik secara langsung maupun tidak langsung, dibutuhkan sebuah tingkat

kepentingan atau ranking antar kriteria. Tingkat kepentingan atau ranking dibutuhkan untuk mengetahui urutan kriteria dari yang terpenting hingga yang paling tidak penting. Dengan begitu, bobot akan dihitung sesuai dengan tingkat kepentingan atau ranking masing-masing kriteria. Setiap kriteria akan memiliki bobot yang berbeda-beda sesuai dengan tingkat kepentingannya.

4.2.3 Subsistem Manajemen Model

Subsistem ini berfungsi untuk memproses perhitungan data dengan menggunakan metode ELECTRE - *Weighted Product*. Kriteria yang akan digunakan untuk pengambilan keputusan berdasarkan hasil wawancara pada pihak SMA BSS yaitu seperti pada Tabel 4.19 dan Tabel 4.21. Selain kriteria, sistem juga membutuhkan pembobotan yang sudah ditentukan oleh pihak SMA BSS sebagai dasar penerimaan calon siswa baru baru yang dapat dilihat pada Tabel 4.20 dan Tabel 4.26.



Gambar 4.12. Kerangka Kerja Model Komputasi
Sumber: [Perancangan]



Gambar 4.13. Kerangka Kerja Metode ELECTRE
Sumber: [Perancangan]

Dari diagram alir pada Gambar 4.13 maka dapat dibuat pseudocode untuk menerapkan cara kerja dari metode ELECTRE. Algoritma tersebut dapat dilihat pada Gambar 4.14 dibawah ini

```
Nama algoritma: Perhitungan ELECTRE
Deklarasi:
  - m adalah integer, untuk menyimpan jumlah alternative
  - n adalah integer, untuk menyimpan jumlah kriteria
  - Matriks c[m][n] adalah Double, untuk menyimpan matriks
    keputusan yang berisi nilai masing-masing calon calon siswa
    baru pada setiap kriteria.
  - Matriks bobot[n] adalah Double, variabel untuk menampung nilai
    bobot dari setiap kriteria.
  - Mariks R[m][n] adalah Double, untuk menyimpan matiks
    normalisasi.
```

- Matriks V[m][n] adalah Double, untuk menyimpan matiks normalisasi terbobot.
- Matriks C_{kl}[m][m][n] untuk menyimpan nilai himpunan *concordance*.
- Matriks D_{kl}[m][m][n] untuk menyimpan nilai himpunan *discordance*.
- Matriks c_{kl}[m][n] untuk menyimpan nilai matriks *concordance*.
- Matriks d_{kl}[m][n] untuk menyimpan nilai matriks *discordance*.
- $\underline{c}$ untuk menyimpan nilai *threshold concordance*.
- $\underline{d}$ untuk menyimpan nilai *threshold discordance*.
- Matriks f_{kl}[m][n] untuk menyimpan nilai domain matriks *concordance*.
- Matriks g_{kl}[m][n] untuk menyimpan nilai domain matriks *discordance*.
- Matriks E untuk menyimpan nilai *aggregate dominance matrix*

e_{kl} untuk menyimpan nilai matriks E setiap elemenya.

Deskripsi:

Input:

- Memasukkan nilai kriteria setiap calon calon siswa baru pada setiap kriteria, diantaranya adalah:
  - Nilai Rapor
  - Nilai Tes Tulis
  - Nilai Wawancara Siswa
  - Nilai Wawancara Orang tua
- Memasukkan nilai bobot masing-masing kriteria, diantaranya:
  - Nilai Bobot Rapor
  - Nilai Bobot Tes Tulis
  - Nilai Bobot Wawancara Siswa
  - Nilai Bobot Wawancaara Orang Tua

Proses:

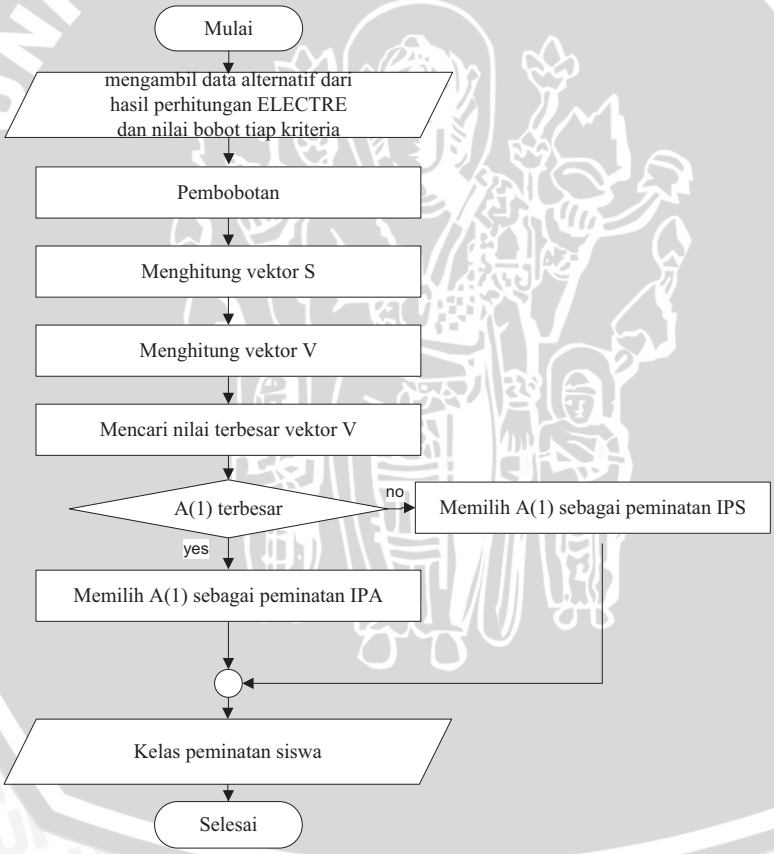
1. Membentuk matriks Keputusan.
2. Menghitung matriks normalisasi.
3. Menghitng matriks normalisasi terbobot dengan menggunakan bobot yang sudah ditentukan.
4. Menentukan himpunan *concordace* dan *discordance*.
5. Menentukan matriks *concordance* dan *discordance*, dengan menjumlahkan bobot yang termasuk pada himpunan *concordance* dan *discordance*
6. Menentukan matriks *concordance* dan *discordance*, dengan bantuan nilai *threshold concordance* dan *discordance*
7. Menentukan *aggregate dominance matrix*, yang merupakan perkalian antara elemen matriks concordance dan discordance.
8. Eliminasi alternatif yang *less favourable*, dengan melihat nilai pada *anggregate dominance matrix*.
9. Menentukan klasifikasi 2 kelompok calon siswa baru yaitu kelompok diterima (A) dan kelompok tidak diterima (B).

OUTPUT:

- 2 kelompok calon siswa baru yaitu kelompok diterima (A) dan kelompok tidak diterima (B)

Gambar 4.14. Perancangan Algoritma ELECTRE
Sumber: [Perancangan]

Melalui Gambar 4.13 dijelaskan bagaimana proses perhitungan yang ada dalam metode ELECTRE dan pada Gambar 4.15 akan dijelaskan bagaimana proses perhitungan dengan metode *Weighted Product* yang akan diterapkan dalam sistem nantinya. Dalam subsistem ini akan dimodelkan bagaimana sistem akan memproses data dengan menggunakan metode ELECTRE - *Weighted Product* ini. Proses perhitungan sistem ini menggunakan metode ELECTRE mulai dari inputan dari pengguna hingga menghasilkan kelompok klasifikasi kelompok yang diterima dan yang tidak diterima. Kemudian perhitungan dilanjutkan dengan menggunakan metode *Weighted Product*. Metode *Weighted Product* ini digunakan untuk menentukan kelas peminatan calon siswa baru yang diterima SMA BSS berdasarkan perhitungan ELECTRE.



Gambar 4.15. Kerangka Kerja Metode *Weighted Product*
Sumber: [Perancangan]

```
Nama algoritma: Perhitungan Weighted Product
Deklarasi:
-   m adalah integer, variable untuk menyimpan jumlah alternatif
-   n adalah integer, variable untuk menyimpan jumlah kriteria
```

- c[i] adalah Integer, variabel untuk menampung nilai setiap alternatif terhadap kriteria.
- bobot[i] adalah Integer, variabel untuk menampung nilai bobot dari kriteria yang ada.
- w[i] adalah Double, variabel untuk menampung nilai perbaikan setiap bobot.
- S[i] adalah Double, variabel untuk menampung perhitungan vektor s.
- V[i] adalah Double, variabel untuk menmpung perhitungan vektor v.

Input:

- Nilai masing-masing calon calon siswa baru, antara lain:
  - -Nilai rata-rata MAT
  - -Nilai rata-rata IPA
  - -Nilai rata-rata IPS
  - -Nilai tes tulis
  - -Nilai wawancara anak
  - -Nilai wawancara orang tua
  - -Nilai tes peminatan IPA
  - -Nilai tes peminatan IPS
  - -Minat anak
  - -Minat orang tua
  - -Nilai IQ
  - -Nilai aspek kecerdasan daya tangkap
  - -Nilai aspek kecerdasan analisa
  - -Nilai aspek kecerdasan logika
  - -Nilai aspek kecerdasan verbal
  - -Nilai aspek kecerdasan hitung
  - -Nilai aspek kecerdasan ruang
  - -Nilai aspek kecerdasan memori
  - -Nilai aspek kecerdasan konsentrasi
  - -Nilai aspek kecerdasan mekanik
  - -Rekomendasi BK

- Nilai bobot masing-masing kriteria, antara lain:
  - -Nilai bobot rata-rata MAT
  - -Nilai bobot rata-rata IPA
  - -Nilai bobot rata-rata IPS
  - -Nilai bobot tes tulis
  - -Nilai bobot wawancara anak
  - -Nilai bobot wawancara orang tua
  - -Nilai bobot tes peminatan IPA
  - -Nilai bobot tes peminatan IPS
  - -Minat bobot anak
  - - Minat bobot orang tua
  - - Nilai bobot IQ
  - - Nilai bobot aspek kecerdasan daya tangkap
  - - Nilai bobot aspek kecerdasan analisa
  - - Nilai bobot aspek kecerdasan
  - - Nilai bobot aspek kecerdasan analisa
  - - Nilai bobot aspek kecerdasan logika
  - - Nilai bobot aspek kecerdasan verbal
  - - Nilai bobot aspek kecerdasan hitung
  - - Nilai bobot aspek kecerdasan ruang
  - - Nilai bobot aspek kecerdasan memori
  - - Nilai bobot aspek kecerdasan konsentrasi

```
        -  Nilai bobot aspek kecerdasan mekanik
        -  Nilai bobot Rekomendasi BK

Proses:
1. Menghitung jumlah nilai bobot yang dimasukkan
2. Menghitung nilai pembobotan untuk masing-masing kriteria
3. Menghitung nilai vektor S dengan menggunakan perulangan
   sebanyak alternatif data yang dihasilkan
4. Menghitung nilai vektor V dengan menggunakan perulangan
   sebanyak alternatif data yang dihasilkan
5. Mencari nilai maksimal vektor V

Output:
   • Calon siswa baru yang diterima dalam kelompok peminatan
     tertentu
```

Gambar 4.16. Perancangan Algoritma *Weighted Product*
Sumber: [Perancangan]

Untuk mengetahui cara kerja sistem yang akan diimplementasikan maka digunakan pemodelan dengan cara membuat contoh kasus seleksi penerimaan calon siswa baru baru. Pemodelan yang dilakukan tersebut mengacu pada diagram blok sistem SPK pada Gambar 3.2 dan kerangka kerja untuk masing-masing metode yang digunakan dalam proses perhitungan mengacu pada Gambar 4.13 dan Gambar 4.15. Sistem akan melakukan perhitungan menggunakan metode ELECTRE dan dilanjutkan dengan perhitungan *Weighted Product.* Berikut adalah langkah-langkah yang dilakukan oleh sistem:

4.2.3.1 Proses Perhitungan ELECTRE

Proses perhitungan ELECTRE ini dilakukan mulai dari pengambilan data alternatif yang akan dihitung yang nantinya akan menghasilkan output dua kelompok calon siswa baru. Berikut ini adalah langkah-langkah perhitungan ELECTRE:

1. Ambil data alternatif

Sebagai contoh pengguna akan menggunakan 5 data alternatif keputusan sebagai inputan. Data alternatif tersebut dapat dilihat pada Tabel berikut.

Tabel 4.27 Matriks Keputusan Calon Calon siswa baru (ELECTRE)

Alternatif	C1	C2	C3	C4
A1	90.7	83.2	90	35
A2	84	86.6	87	35
A3	81.3	81.1	90	35
A4	82.7	86.6	90	33
A5	81.3	86.2	90	35

Sumber: [Perancangan]

2. Ambil nilai bobot tiap kriteria

Bobot yang digunakan merupakan bobot yang telah ditentukan oleh SMA BSS sebagaimana yang telah dijelaskan pada Tabel 4.20. Nilai Bobot setiap kriteria ditunjukkan pada Tabel 4.28 berikut:

Tabel 4.28 Nilai Bobot Kriteria (ELECTRE)

Kriteria	Nilai Bobot Kriteria (w)
Hasil Seleksi Rapor	$W1 = 40/100 = 0.4$
Hasil Sekeksi Tes Tulis	$W2 = 30/100 = 0.3$
Wawancara Siswa	$W3 = 15/100 = 0.15$
Wawancara Orang Tua	$W4 = 15/100 = 0.15$

Sumber: [Perancangan]

3. Langkah 1 : Menghitung matriks normalisasi

Langkah pertama yang dilakukan pada proses perhitungan ELECTRE adalah melakukan normalisasi terhadap nilai matriks keputusan calon calon siswa baru yang telah dimasukkan ke dalam sistem pendukung keputusan. Proses ini dilakukan untuk mempermudah dalam membandingkan nilai dari setiap alternatifnya. Diagram alir menghitung matriks normalisasi ditunjukkan dalam Gambar 4.17. Penerapan cara kerja dari menghitung matriks normalisasi dengan *pseudocode* ditunjukkan pada Gambar 4.18.

Perhitungan matriks normalisasi menggunakan rumus Persamaan (2-1). Nilai matriks normalisasi (alternatif i, kriteria j) merupakan hasil pembagian antara nilai matriks kputusan (alternatif I, kriteria j) dengan akar jumlah nilai alternatif i pada setiap kriteria j. Contoh perhitungan berdasarkan pada Tabel 4.27 sehingga dapat dijabarkan sebagai berikut:

- Alternatif A2 terhadap Kriteria C1

$$r_{21} = \frac{x_{21}}{\sqrt{\sum_{i=2}^{m} x_{i1}^2}} = \frac{84}{\sqrt{90.7^2 + 84^2 + 81.3^2 + 82.7^2 + 81.3^2}}$$

$$= \frac{90.7}{187.99} = 0.446$$

- Alternatif A1 terhadap Kriteria C2

$$r_{12} = \frac{x_{12}}{\sqrt{\sum_{i=1}^{m} x_{i2}^2}} = \frac{83.2}{\sqrt{83.2^2 + 86.6^2 + 81.1^2 + 86.6^2 + 86.2^2}}$$

$$= \frac{83.2}{189.55} = 0.438$$

- Alternatif A3 terhadap Kriteria C1

$$r_{31} = \frac{x_{31}}{\sqrt{\sum_{i=1}^{m} x_{i1}^2}} = \frac{81.3}{\sqrt{90.7^2 + 84^2 + 81.3^2 + 82.7^2 + 81.3^2}}$$

$$_{11} = \frac{81.3}{187.99} = 0.432$$

- Alternatif A3 terhadap Kriteria C4

$$r_{34} = \frac{x_{33}}{\sqrt{\sum_{i=1}^{m} x_{i1}^2}} = \frac{35}{\sqrt{35^2 + 35^2 + 35^2 + 33^2 + 35^2}}$$

$$r_{11} = \frac{35}{77.38} = 0.452$$

Perhitungan dilakukan untuk seluruh alternatif terhadap kriteria yang ada. Sehingga hasil matriks ternormalisasi R untuk data pada Tabel 4.27 adalah sebagai berikut:

$$R = \begin{bmatrix} 0.482 & 0.439 & 0.450 & 0.452 \\ 0.447 & 0.457 & 0.434 & 0.452 \\ 0.432 & 0.428 & 0.450 & 0.452 \\ 0.440 & 0.457 & 0.450 & 0.426 \\ 0.432 & 0.454 & 0.450 & 0.452 \end{bmatrix}$$



Gambar 4.17. Diagram Alir Matriks Normalisasi ELECTRE
Sumber: [Perancangan]

```
NAMA ALGORITMA:
PERHITUNGAN MATRIKS NORMALISASI ELECTRE

DEKLARASI:
    - M_keputusan1[m] [n] adalah Double, digunakan untuk
      menyimpan matriks keputusan yang berisi nilai masing-
      masing calon calon siswa baru pada setiap kriteria.
    - n adalah integer, merupakan indeks kriteria.
    - m adalah integer, merupakan indeks alternatif.
    - x[n] adalah Double, digunakan untuk menyimpan nilai akar
      jumlah nilai kuadrat setiap kolom kriteria pada matriks
      keputusan.
    - sum[n] adalah Double, digunakan untuk menyimpan jumlah
      nilai kuadrat setiap kolom kriteria pada matriks
      keputusan.
    - r[m][n] adalah Double, digunakan untuk menyimpan matriks
      normalisasi.

DESKRIPSI:
INPUT:
```

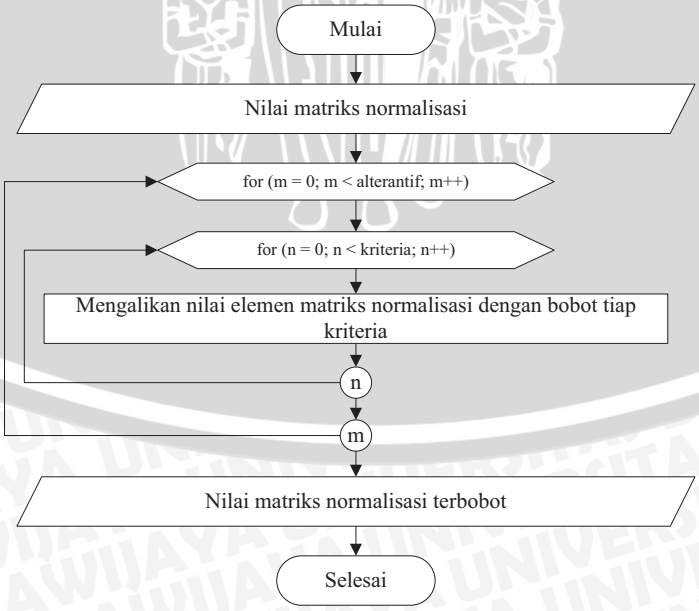
```
    - Memasukkan nilai masing-masing calon calon siswa baru,
      pada M_keputusan1[m] [n].
PROSES:
  1. Menghitung jumlah nilai kuadrat setiap kolom a pada matriks
     keputusan, lalu nilai disimpan pada sum[n].
  2. Menghitung nilai akar setiap element sum[n] lalu nilai
     disimpan pada x[n].
  3. Membagi masing-masing nilai element pada matriks_keputusan
     calon calon siswa baru dengan nilai matriks_akar, untuk
     setiap matriks_keputusan[m][n]:
     - Apabila n=1, maka akan dibagi dengan x[1]
     - Apabila n=2, maka akan dibagi dengan x[2]
     - Dst,
     Hasil pembagian akan disimpan pada r[m][n].

OUTPUT:
Matrik normalisasi, yang merupakan hasil normalisasi dari
matriks keputusan.
```

Gambar 4.18. *Psedocode* Matriks Normalisasi ELECTRE
Sumber: [Perancangan]

4. Langkah 2: Pembobotan pada matriks ternomalisasi

Langkah selanjutnya adalah melakukan pembobotan pada matriks normalisasi. Proses ini dilakukan apabila nilai matriks normalisasi sudah terbentuk dan terdapat nilai bobot untuk setiap kriteria yang ada. Diagram alir pembobotan pada matriks normalisasi ditunjukkan dalam Gambar 4.19. dan perenapan cara kerja dari pembobotan pada matriks normalisasi dengan *pseudocode* ditunjukkan pada Gambar 4.20.



Gambar 4.19. Diagram Alir Matriks Normalisasi Terbobot ELECTRE
Sumber: [Perancangan]

```
NAMA ALGORITMA:
PERHITUNGAN MATRIKS NORMALISASI TERBOBOT ELECTRE

DEKLARASI:
    - r[m][n] adalah Double, digunakan untuk menyimpan matriks
      normalisasi.
    - n adalah integer, merupakan indeks kriteria.
    - m adalah integer, merupakan indeks alternatif.
    - w[n] adalah Double, digunakan untuk menyimpan nilai bobot
      untuk setiap kriteria yang ada.
    - v[m][n] adalah Double, digunakan untuk menyimpan hasil
      matriks normalisasi terbobot.

DESKRIPSI:

INPUT:
    -  r[m][n] yang berisi nilai matriks normalisasi.
    -  w[n] yang berisi nilai bobot dari setiap kriteria.

PROSES:
   1. Mengalikan masing-masing nilai element pada matriks_R
      dengan nilai matriks_bobot, untuk setiap r[a][b]:
      -  Apabila m=1, maka akan dikali dengan w[1]
      -  Apabila m=2, maka akan dikali dengan w[2]
      -  Dst,
      Hasil kali akan disimpan pada v[m][n]

OUTPUT:
   -  Matrik normalisasi terbobot yang merupakan hasil perkalian
      antara nilai matriks normalisasi dengan nilai matriks dari
      setiap kriteria yang ada.
```

Gambar 4.20. *Pseudocode* Perhitungan Matriks Normalisasi Terbobot ELECTRE
Sumber: [Perancangan]

Perhitungan matriks normalisasi terbobot menggunakan rumus Persamaan (2-3). Nilai matriks normalisasi terbobot (alternatif i, kriteria j) merupakan hasil kali dari nilai matriks normalisasi (R) alternatif i, kriteria j dengan nilai bobot (w) kriteria j yang dapat dilihat pada Tabel 4.23. Sehingga dapat dihitung dengan cara sebagai berikut:

- Pembobotan pada matriks normalisasi alternatif 2 kriteria 1

$$v_{21} = w_1 r_{21} = (0.40)(0.446) = 0.178$$

- Pembobotan pada matriks normalisasi alternatif 1 kriteria 2

$$v_{12} = w_1 r_{12} = (0.30)(0.456) = 0.131$$

- Pembobotan pada matriks normalisasi alternatif 3 kriteria 1

$$v_{31} = w_1 r_{31} = (0.15)(0.432) = 0.173$$

- Pembobotan pada matriks normalisasi alternatif 3 kriteria 4

$$v_{34} = w_3 r_{34} = (0.15)(0.452) = 0.067$$

perhitungan dilakukan untuk seluruh alternatif terhadap kriteria yang ada. Sehingga hasil matriks ternormalisasi V untuk data pada Tabel 4.27 adalah sebagai berikut:

$$V = \begin{bmatrix} 0.193 & 0.132 & 0.067 & 0.068 \\ 0.179 & 0.137 & 0.065 & 0.068 \\ 0.173 & 0.128 & 0.067 & 0.068 \\ 0.176 & 0.137 & 0.067 & 0.064 \\ 0.173 & 0.136 & 0.067 & 0.068 \end{bmatrix}$$

5. Langkah 3: Menentukan himpunan *concordance* dan *discordance*

Langkah selanjutnya adalah menentukan himpunan *concordance* dan *discordance*. Proses ini dapat dilakukan setelah terbentuk nilai matriks normalisasi terbobot.

a. *Concordance*

Diagram alir penentuan himpunan *concordance* ditunjukkan pada Gambar 4.21. dan penerapan cara kerja dari penetuan himpunan *concordance* dengan *pseudocode* ditunjukkan pada gambar 4.22. Perhitungan matriks normalisasi menggunakan rumus Persamaan (2-5). Setiap nilai pada alternatif k dibangdingkan dengan alternatif l pada setiap kriteria j. Perbandingan dilakukan dengan cara: jika nilai alternatif k kriteria j ≥ nilai alternatif l kriteria j, maka kriteria j termasuk ke dalam himpunan himpunan *concordane* (C_{kl}) atau disimbolkan dengan nilai j. Perbandingan dilakukan berdasarkan hasil perhitungan pembobotan matriks ternormalisasi, dimana:

$$V = \begin{bmatrix} 0.193 & 0.132 & 0.067 & 0.068 \\ 0.179 & 0.137 & 0.065 & 0.068 \\ 0.173 & 0.128 & 0.067 & 0.068 \\ 0.176 & 0.137 & 0.067 & 0.064 \\ 0.173 & 0.136 & 0.067 & 0.068 \end{bmatrix}$$

Sehingga dapat dihitung dengan cara seperti dibawah ini:

- Untuk alternatif 2 dibanding dengan alternatif 1 pada kriteria 1:

Melakukan perbandingan antara nilai alternatif 2 pada kriteria 1 (v_{21}) dengan alternatif 1 pada kriteria 1 (v_{11}). Apabila nilai $v_{21} \geq v_{11}$, maka kriteria 1 masuk ke dalam himpunan *concordance*, dan sebaliknya jika nilai $v_{21} < v_{11}$, maka kriteria 1 tidak termasuk ke dalam himpunan *concordance*.

$C_{21} = \{1, v_{21}\ dibanding\ v_{11}\} = \{1, 0.179 < 0.193\}$

v_{21} memiliki nilai lebih kecil dari v_{11}, maka kriteria 1 merupakan bukan himpunan *concordance* untuk alternatif 2 terhadap alternatif 1

- Untuk alternatif 1 terhadap alternatif 3 pada kriteria 2:

Melakukan perbandingan antara nilai alternatif 1 pada kriteria 2 (v_{12}) dengan alternatif 3 pada kriteria 2 (v_{32}). Apabila nilai $v_{12} \geq v_{32}$, maka kriteria 2 masuk ke dalam himpunan *concordance*, dan sebaliknya jika nilai $v_{12} < v_{32}$, maka kriteria 2 tidak termasuk ke dalam himpunan *concordance*.

$C_{12} = \{2, v_{12}\ dibanding\ v_{32}\} = \{2, 0.132 > 0.128\}$

V_{12} memiliki nilai lebih besar dari v_{32}, maka kriteria 2 merupakan himpunan *concordance* untuk alternatif 2 terhadap alternatif 3

- Untuk alternatif 3 terhadap alternatif 5 pada kriteria 1:

Melakukan perbandingan antara nilai alternatif 3 pada kriteria 1 (v_{31}) dengan alternatif 5 pada kriteria 1 (v_{51}). Apabila nilai $v_{31} \geq v_{51}$, maka kriteria 1 masuk ke dalam himpunan *concordance*, dan sebaliknya jika nilai $v_{31} < v_{51}$, maka kriteria 1 tidak termasuk ke dalam himpunan *concordance*.

$C_{31} = \{1, v_{31}\ dibanding\ v_{51}\} = \{1, 0.173 > 0.172\}$

v_{31} memiliki nilai lebih besar dari v_{51}, maka kriteria 1 merupakan himpunan *concordance* untuk alternatif 3 terhadap alternatif 5

- Untuk alternatif 3 terhadap alternatif 2 pada kriteria 4:

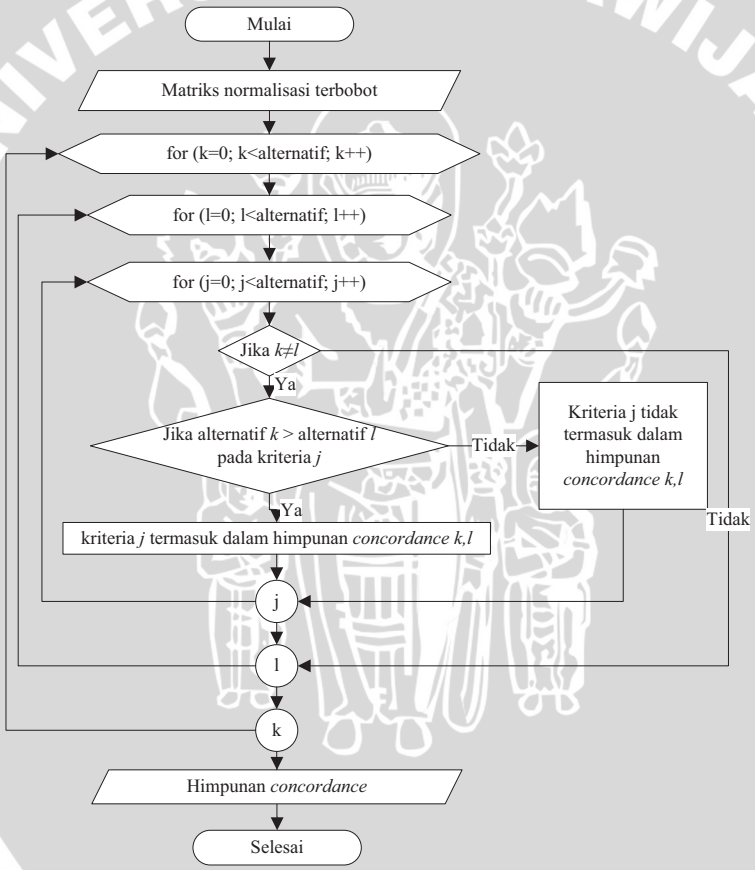
Melakukan perbandingan antara nilai alternatif 3 pada kriteria 4 (v_{34}) dengan alternatif 2 pada kriteria 4 (v_{24}). Apabila nilai $v_{34} \geq v_{24}$, maka kriteria 4 masuk ke dalam himpunan *concordance*, dan sebaliknya jika nilai $v_{34} < v_{24}$, maka kriteria 4 tidak termasuk ke dalam himpunan *concordance*.

$C_{34} = \{4, v_{34}\ dibanding\ v_{44}\} = \{4, 0.0680 > 0.0680\}$

v_{34} memiliki nilai lebih besar dari v_{44}, maka kriteria 3 merupakan himpunan *concordance* untuk alternatif 4 terhadap alternatif 4

Perhitungan dilakukan untuk seluruh alternatif pada seluruh kriteria yang ada. Sehingga nantinya didapatkan himpunan *concordance* sebagai berikut:

C12={1,3,4}	C34={3,4}	C24={1,2,4}	C51={2,3,4}
C13={1,2,3,4}	C35={1,3,4}	C25={1,2,4}	C52={3,4}
C14={1,3,4}	C41={2,3}	C31={3,4}	C53={1,23,4}
C15={1,3,4}	C42={2,3}	C32={3,4}	C54={3,4}
C21={2,4}	C43={1,2,3}	C23={1,2,4}	C45={1,2,3}



Gambar 4.21. Diagram Alir Penentuan Himpunan *Concordance*
Sumber: [Perancangan]

```
NAMA ALGORITMA:
PENENTUAN HIMPUNAN CONCORDANCE ELECTRE
DEKLARASI:
    - v[m][n] adalah Double, digunakan untuk menyimpan hasil
      matriks normalisasi terbobot.
    - j adalah integer, merupakan indeks kriteria.
```

```
    - k adalah integer, merupakan indeks alternatif.
    - l adalah integer, merupakan indeks alternatif.
    - C[m][m][n] adalah integer, digunakan untuk menyimpan
      himpunan concordance.
DESKRIPSI:
INPUT:
    - v[m][n] yang berisi nilai matriks normalisasi terbobot.
PROSES:
  1. Membandingkan setiap alternatif ke-k dengan alternatif ke-
     l pada setiap kriteria ke-j selama k ≠ l. Jika a=b, maka k
     akan membandingkan pada nilai l selanjutnya.
  2. Kedua alternatif dibandingkan dengan ketentuan jika v baris
     ke-k ≥ v baris ke-l pada kolom ke-j, maka kriteria ke-j
     masuk ke dalam himpunan concordance dengan nilai=j.
  3. Hasil banding setiap kolom ke-j pada setiap perbandingan
     baris j,k akan disimpan pada C[k][l][j].
OUTPUT:
  - Matrik himpunan concordance yang merupakan hasil
    perbandingan nilai v baris ke=j dan j pada setiap kolom ke-
    j
```

Gambar 4.22. *Pseudocode* Penentuan Himpunan *Concordance*
Sumber: [Perancangan]

b. *Discordance*

Diagram alir penentuan himpunan *discordance* ditunjukkan pada gambar 4.23. dan penerapan cara kerja dari penetuan himpunan *discordance* dengan *pseudocode* ditunjukkan pada gambar 4.24.

Perhitungan matriks normalisasi menggunakan rumus Persamaan (2-6). Setiap nilai pada alternatif k dibangdingkan dengan alternatif l pada setiap kriteria j. Perbandingan dilakukan dengan cara: jika nilai alternatif k kriteria j < nilai alternatif l kriteria j, maka kriteria j termasuk ke dalam himpunan himpunan *discordane* (D_{kl}) atau disimbolkan dengan nilai j. Perbandingan dilakukan berdasarkan hasil perhitungan pembobotan matriks ternormalisasi, dimana:

$$V = \begin{bmatrix} 0.193 & 0.132 & 0.067 & 0.068 \\ 0.179 & 0.137 & 0.065 & 0.068 \\ 0.173 & 0.128 & 0.067 & 0.068 \\ 0.176 & 0.137 & 0.067 & 0.064 \\ 0.173 & 0.136 & 0.067 & 0.068 \end{bmatrix}$$

Sehingga dapat dihitung dengan cara seperti dibawah ini:

- Untuk alternatif 1 terhadap alternatif 2 pada kriteria 1:

Melakukan perbandingan antara nilai alternatif 2 pada kriteria 1 (v_{21}) dengan alternatif 1 pada kriteria 1 (v_{11}). Apabila nilai $v_{21}<v_{11}$, maka kriteria 1 masuk ke

dalam himpunan *discordance*, dan sebaliknya jika nilai $v_{21} \geq v_{11}$, maka kriteria 1 tidak termasuk ke dalam himpunan *discordance*.

$D_{21} = \{1, v_{21}\ dibanding\ v_{22}\} = \{1, 0.179 < 0.193\}$

v_{21} memiliki nilai lebih kecil dari v_{11}, maka kriteria 1 merupakan himpunan *discordance* untuk alternatif 2 terhadap alternatif 1

- Untuk alternatif 1 terhadap alternatif 3 pada kriteria 2:

Melakukan perbandingan antara nilai alternatif 1 pada kriteria 2 (v_{12}) dengan alternatif 3 pada kriteria 2 (v_{32}). Apabila nilai $v_{12} < v_{32}$, maka kriteria 2 masuk ke dalam himpunan *discordance*, dan sebaliknya jika nilai $v_{12} \geq v_{32}$, maka kriteria 2 tidak termasuk ke dalam himpunan *discordance*.

$D_{12} = \{2, v_{12}\ dibanding\ v_{32}\} = \{2, 0.132 > 0.128\}$

V_{12} memiliki nilai lebih besar dari v_{32}, maka kriteria 2 bukan merupakan himpunan *discordance* untuk alternatif 2 terhadap alternatif 3

- Untuk alternatif 3 terhadap alternatif 5 pada kriteria 1:

Melakukan perbandingan antara nilai alternatif 3 pada kriteria 1 (v_{31}) dengan alternatif 5 pada kriteria 1 (v_{51}). Apabila nilai $v_{31} < v_{51}$, maka kriteria 1 masuk ke dalam himpunan *discordance*, dan sebaliknya jika nilai $v_{31} \geq v_{51}$, maka kriteria 1 tidak termasuk ke dalam himpunan *discordance*.

$D_{31} = \{1, v_{31}\ dibanding\ v_{51}\} = \{1, 0.173 > 0.172\}$

v_{31} memiliki nilai lebih besar dari v_{51}, maka kriteria 1 bukan merupakan himpunan *discordance* untuk alternatif 3 terhadap alternatif 5

- Untuk alternatif 3 terhadap alternatif 2 pada kriteria 3:

Melakukan perbandingan antara nilai alternatif 3 pada kriteria 4 (v_{34}) dengan alternatif 2 pada kriteria 4 (v_{24}). Apabila nilai $v_{34} < v_{24}$, maka kriteria 4 masuk ke dalam himpunan *discordance*, dan sebaliknya jika nilai $v_{34} \geq v_{24}$, maka kriteria 4 tidak termasuk ke dalam himpunan *discordance*.

$D_{34} = \{4, v_{34}\ dibanding\ v_{44}\} = \{4, 0.0680 > 0.0680\}$

v_{34} memiliki nilai lebih besar dari v_{44}, maka kriteria 3 bukan merupakan himpunan *discordance* untuk alternatif 4 terhadap alternatif

- Untuk alternatif 3 terhadap alternatif 2 pada kriteria 3:

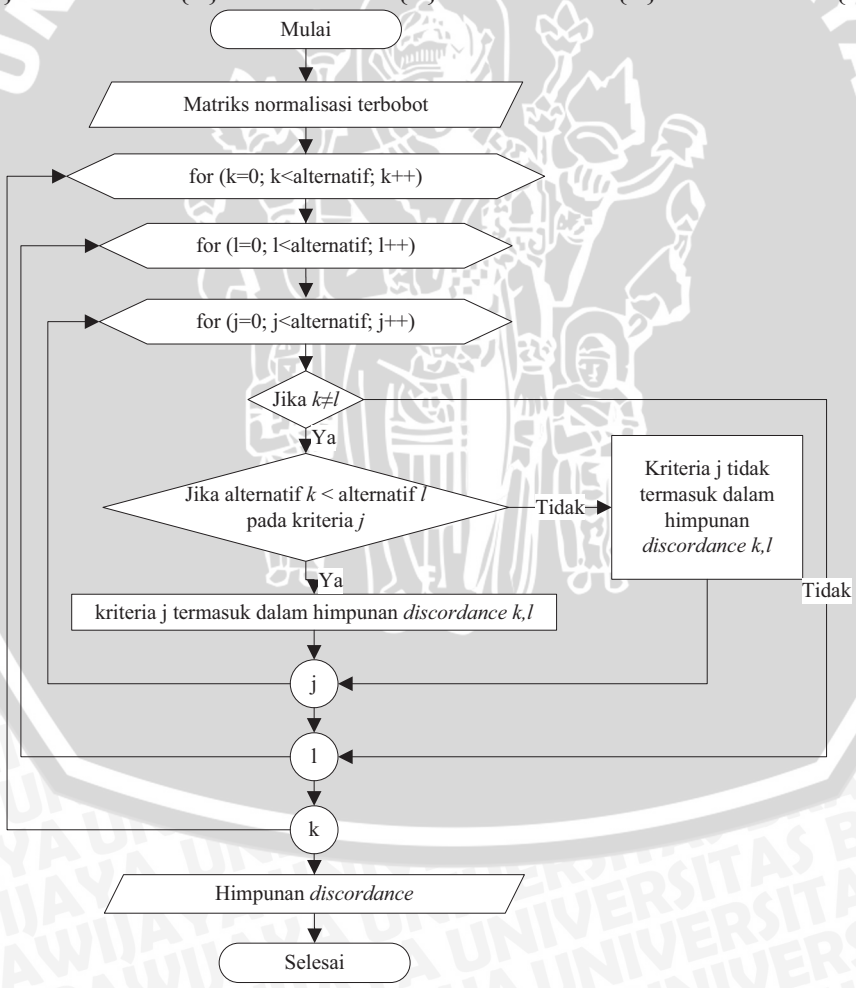
Melakukan perbandingan antara nilai alternatif 3 pada kriteria 4 (v_{34}) dengan alternatif 2 pada kriteria 4 (v_{24}). Apabila nilai $v_{34}<v_{24}$, maka kriteria 4 masuk ke dalam himpunan *discordance*, dan sebaliknya jika nilai $v_{34}\geq v_{24}$, maka kriteria 4 tidak termasuk ke dalam himpunan *discordance*.

$D_{34} = \{4, v_{34}\ dibanding\ v_{44}\} = \{4, 0.0680 > 0.0680\}$

v_{34} memiliki nilai lebih besar dari v_{44}, maka kriteria 3 bukan merupakan himpunan *discordance* untuk alternatif 4 terhadap alternatif

Perhitungan dilakukan untuk seluruh alternatif pada seluruh kriteria yang ada. Sehingga nantinya didapatkan himpunan *discordance* sebagai berikut:

D12={2}	D21={1,3}	D31={1,2}	D41={1,4}	D51={1}
D13={-}	D23={3}	D32={1,2}	D42={1,4}	D52={1,2}
D14={2}	D24={3}	D34={1,2}	D43={1,2}	D53={-}
D15={2}	D25={3}	D35={2}	D45={4}	D54={1,2}



Gambar 4.23. Diagram Alir Penentuan Himpunan Discordance
Sumber: [Perancangan]

```
NAMA ALGORITMA:
PENENTUAN HIMPUNAN DISCORDANCE ELECTRE

DEKLARASI:
    - v[m][n] adalah Double, digunakan untuk menyimpan hasil
      matriks normalisasi terbobot.
    - j adalah integer, merupakan indeks kriteria.
    - k adalah integer, merupakan indeks alternatif.
    - l adalah integer, merupakan indeks alternatif.
    - D[m][m][n] adalah integer, digunakan untuk menyimpan
      himpunan concordance.
INPUT:
    - v[m][n] yang berisi nilai matriks normalisasi terbobot.
PROSES:
  1. Membandingkan setiap alternatif ke-k dengan alternatif ke-
     l pada setiap kriteria ke-j selama k ≠ l. Jika k=l, maka k
     akan membandingkan pada nilai l selanjutnya.
  2. Kedua alternatif dibandingkan dengan ketentuan jika v baris
     ke-k < v baris ke-l pada kolom ke-j, maka kriteria ke-j
     masuk ke dalam himpunan discordance dengan nilai=j.
  3. Hasil banding setiap kolom ke-j pada setiap perbandingan
     baris k,l akan disimpan pada D[k][l][j].
OUTPUT:
    - Matrik himpunan discordance yang merupakan hasil
      perbandingan nilai matriks_V baris ke=l dan k pada setiap
      kolom ke-j.
```

Gambar 4.24. *Pseudocode* Penentuan Himpunan *Discordance*
Sumber: [Perancangan]

6. Langkah 4: Menentukan matriks *concordance* dan *discordance*

a. *Concordance*

Diagram alir penentuan matriks *concordance* ditunjukkan pada gambar 4.25. dan penerapan cara kerja dari penetuan matriks *concordance* dengan *pseudocode* ditunjukkan pada gambar 4.26. Perhitungan matriks *concordance* menggunakan rumus Persamaan (2-7). Nilai dari matriks *concordance* pada baris ke k kolom l merupakan penjumlahan dari nilai bobot kriteria yang termasuk dalam himpunan *concordance* (C_{kl}). Perhitungan matriks *concordance* berdasarkan hasil dari himpunan concordance yang telah didapatkan sebelumnya, dimana:

C12={1,3,4}	C34={3,4}	C25={1,2,4}	C52={3,4}
C13={1,2,3,4}	C35={1,3,4}	C31={3,4}	C53={1,23,4}
C14={1,3,4}	C41={2,3}	C32={3,4}	C54={3,4}
C15={1,3,4}	C42={2,3}	C23={1,2,4}	C45={1,2,3}
C21={2,4}	C43={1,2,3}	C24={1,2,4}	C51={2,3,4}

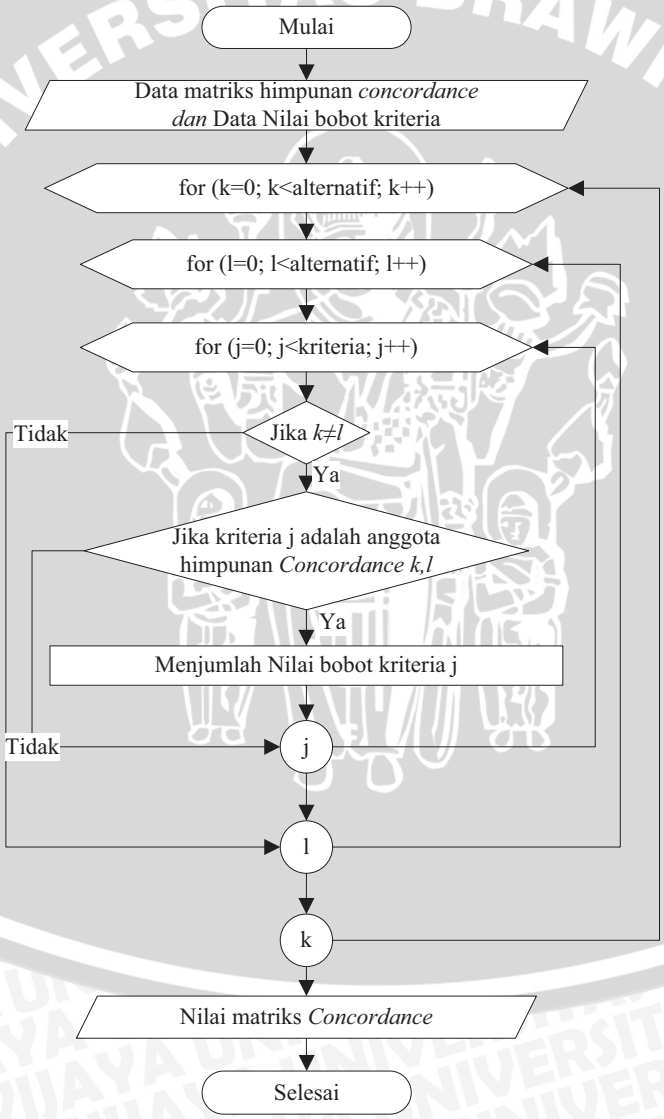
Nilai bobot yang digunakan adalah nilai bobot yang mengacu pada Tabel 4.20.Sehingga dapat dihitung dengan cara seperti di bawah ini:

$C_{21} = w_2 + w_4 = 0.3 + 0.15 = 0.45$

$C_{12} = w_1 + w_3 + w_4 = 0.4 + 0.15 + 0.15 = 0.7$

$C_{24} = w_1 + w_2 + w_4 = 0.4 + 0.3 + 0.15 = 0.85$

Dan seterusnya untuk seluruh himpunan *concordance*.



Gambar 4.25. Diagram Alir Penentuan Matriks *Concordance*
Sumber: [Perancangan]

NAMA ALGORITMA:
PENENTUAN MATRIKS *CONCORDANCE* ELECTRE

DEKLARASI:
- c[m][m][n] adalah integer, digunakan untuk menyimpan himpunan *concordance*.
- j adalah integer, merupakan indeks kriteria.
- k adalah integer, merupakan indeks alternatif.
- l adalah integer, merupakan indeks alternatif.
- Matrix_C[m][m] adalah Double, digunakan untuk menyimpan nilai matriks *concordance*.
- w[n] adalah Double, digunakan untuk menyimpan nilai bobot masing-masing kriteria.

DESKRIPSI:
INPUT:
- C[m][m][n] yang berisi matriks himpunan *concordance*.

PROSES:
1. Menjumlahkan nilai bobot kriteria, dimana kriteria yag masuk ke dalam himpunan *concordance* memiliki ketentuan k ≠ 1 dan nilai element C[k][1][j] ≠ 0. Jika nilai k=1, maka akan menuju ke nilai 1 selanjutnya.
2. Penjumlahan dilakukan dengan cara:
    - Jika nilai himpunan yang ada pada matrik C[k][l][j] = 1, maka nilai bobot kriteria yang dijumlahkan adalah nilai bobot kriteria 1.
    - Jika nilai himpunan yang ada pada matrik C[k][l][j] = 2, maka nilai bobot kriteria yang dijumlahkan adalah nilai bobot kriteria 2.
    - Dst.
3. Hasil penjumlahan bobot kriteria yang terpasuk pada himpunan *concordance* C[k][l][j] akan disimpan dalam Matrix_C[m][m]

OUTPUT:
- Matrik *concordance* yang merupakan hasil penjumlahan nilai bobot kriteria yang termasuk dalam himpunan *concordance*.

Gambar 4.26. *Pseudocode* Penentuan Matriks *Concordance*
Sumber: [Perancangan]

Perhitungan dilakukan untuk seluruh alternatif pada seluruh kriteria yang ada. Sehingga nantinya didapatkan matriks *concordance* sebagai berikut:

$$C = \begin{bmatrix} - & 0.70 & 1.00 & 0.70 & 0.70 \\ 0.45 & - & 0.85 & 0.85 & 0.85 \\ 0.30 & 0.30 & - & 0.30 & 0.70 \\ 0.45 & 0.45 & 0.85 & - & 0.85 \\ 0.60 & 0.30 & 1.00 & 0.30 & - \end{bmatrix}$$

b. *Discordance*

Diagram alir penentuan matriks *discordance* ditunjukkan pada Gambar 4.27. dan penerapan cara kerja dari penetuan matriks *discordance* dengan *pseudocode* ditunjukkan pada gambar 4.28.

Perhitungan matriks *discordance* menggunakan rumus Persamaan (2-8). Nilai matriks *discordance* pada baris ke-k kolom l merupakan hasil pembagian dari maksimal nilai X dengan maksimal nilai Y. Nilai X adalah selisih antara nilai matriks normalisasi terbobot alternatif k kriteria j dengan nilai matriks normalisasi terbobot alternatif l kriteria j, dimana kriteria j merupakan anggota himpunan dari himpunan *discordance* (D_{kl}). Nilai Y adalah nilai selisih antara nilai matriks normalisasi terbobot alternatif k kriteria j dengan nilai matriks normalisasi terbobot alternatif l kriteria j.

Nilai matriks normalisasi yang digunakan adalah sebagai berikut:

$$V = \begin{bmatrix} 0.193 & 0.132 & 0.067 & 0.068 \\ 0.179 & 0.137 & 0.065 & 0.068 \\ 0.173 & 0.128 & 0.067 & 0.068 \\ 0.176 & 0.137 & 0.067 & 0.064 \\ 0.173 & 0.136 & 0.067 & 0.068 \end{bmatrix}$$

Sedangkan untuk himpunan *discordance* yang digunakan adalah sebagai berikut:

D12={2}	D31={1,2}	D23={3}
D13={-}	D32={1,2}	D24={3}
D14={2}	D34={1,2}	D25={3}
D15={2}	D35={2}	D51={1}
D21={1,3}	D41={1,4}	D52={1,2}
D42={1,4}	D43={1,2}	D45={4}
D54={1,2}	D53={-}	

Sehingga dapat dihitung dengan cara:

$$d_{21} = \frac{max\{|v_{2j} - v_{1j}|\}_{j \in D_{21}}}{max\{|v_{2j} - v_{1j}|\}_{\forall j}}$$

$$= \frac{max\{|0.178 - 0.192|\}}{max\{|0.178 - 0.192|; |0.137 - 0.131|; |0.065 - 0.067|; |0.678 - 0.678|\}}$$

$$= \frac{max\{|0.014256|\}}{0.014256} = 1$$

$$d_{12} = \frac{max\{|v_{1j} - v_{2j}|\}_{j \in D_{12}}}{max\{|v_{1j} - v_{2j}|\}_{\forall j}}$$

$$= \frac{max\{|0|\}}{max\{|0.192 - 0.178|; |0.131 - 0.137|; |0.067 - 0.065|; |0.678 - 0.678|\}}$$

$$= \frac{max\{|0|\}}{0.014256} = 0$$

$$d_{31} = \frac{max\{|v_{1j} - v_{3j}|\}_{j \in D_{31}}}{max\{|v_{1j} - v_{3j}|\}_{\forall j}}$$

d_{12}

$$= \frac{max\{|0.172 - 0.192|\}}{max\{|0.178 - 0.192|; |0.137 - 0.131|; |0.065 - 0.067|; |0.678 - 0.678|\}}$$

$$= 1$$

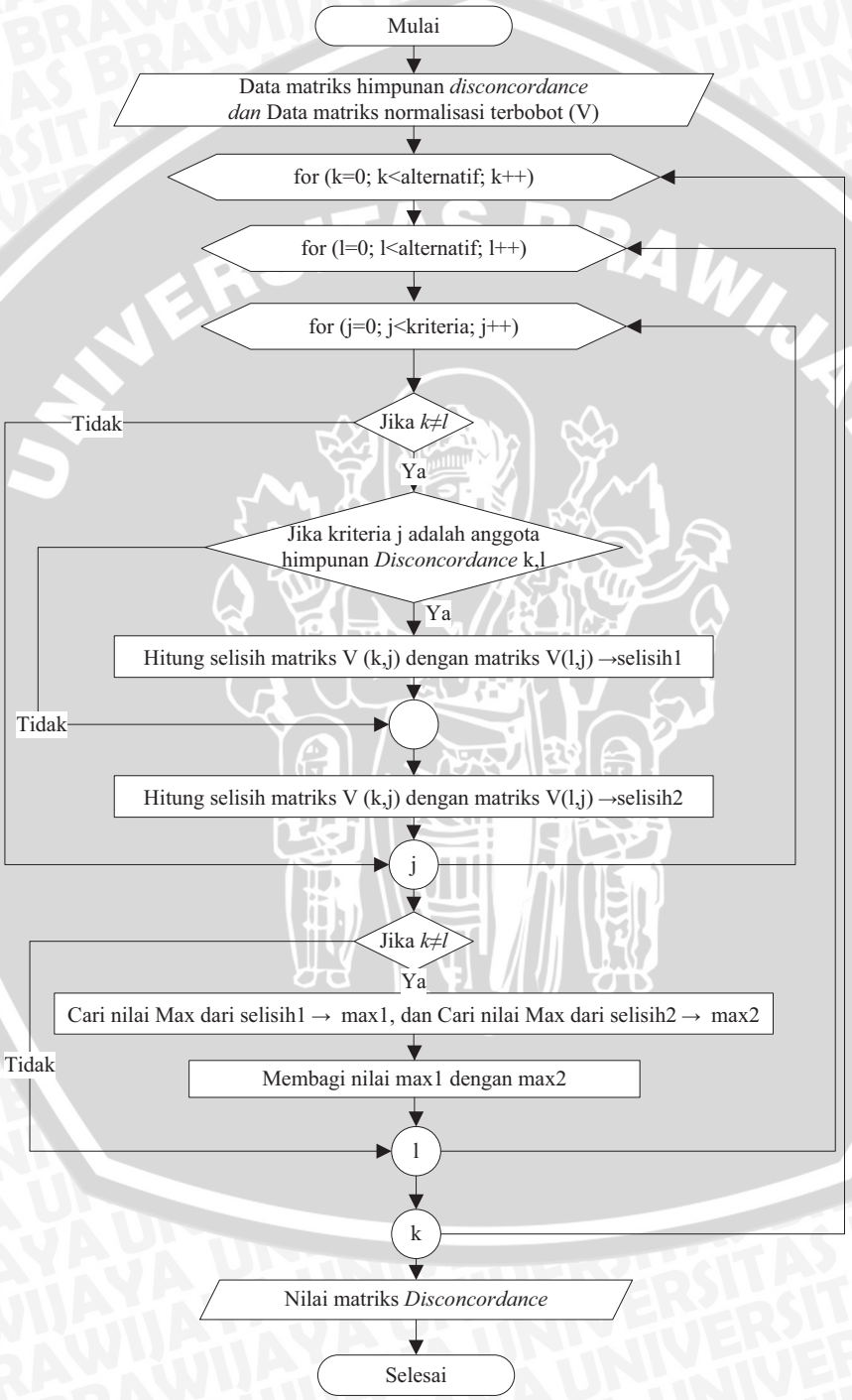
$$d_{34} = \frac{max\{|v_{3j} - v_{4j}|\}_{j \in D_{31}}}{max\{|v_{3j} - v_{4j}|\}_{\forall j}}$$

$$= \frac{max\{|0|\}}{max\{||0.172 - 0.175|; |0.128 - 0.137|; |0.067 - 0.067|; |0.067 - 0.0.63||\}}$$

$$= 0$$

Perhitungan dilakukan untuk seluruh alternatif pada seluruh kriteria yang ada. Sehingga nantinya didapatkan matriks *discordance* sebagai berikut:

$$D = \begin{bmatrix} - & 0.3774 & 0 & 0.3161 & 0.2359 \\ 1.1755 & - & 0.2879 & 0.6456 & 0.4356 \\ 1.3427 & 1.6610 & - & 1.3427 & 1 \\ 1.2277 & 1.7135 & 0.4460 & - & 1 \\ 1 & 1.1152 & 0 & 0.9391 & - \end{bmatrix}$$



Gambar 4.27. *Diagram Alir Penentuan Matriks* Discordance
Sumber: [Perancangan]

```
NAMA ALGORITMA:
PENENTUAN MATRIKS DISCORDANCE ELECTRE

DEKLARASI:
    - D[m][m][n] adalah integer, digunakan untuk menyimpan
      himpunan discordance.
    - j adalah integer, merupakan indeks kriteria.
    - k adalah integer, merupakan indeks alternatif.
    - l adalah integer, merupakan indeks alternatif.
    - Matrix_D[m][m] adalah Double, digunakan untuk menyimpan
      nilai matriks discordance.

DESKRIPSI:

INPUT:
    - D[m][m][n] yang berisi matriks himpunan discordance.

PROSES:
   1. Selama k ≠ l: mencari selisih antara nilai v alternatif ke-
      k kriteria ke-j dengan nilai v alternatif ke-l kriteria ke-
      j, dimana kriteria ke-j merupakan anggota himpunan
      discordance. Nilai selisih dapat disimbolkan dengan
      selisih1[n].
   2. Selama k ≠ l: mencari selisih antara nilai v alternatif ke-
      k kriteria ke-j dengan nilai v alternatif ke-l kriteria ke-
      j. Nilai selisih dapat disimbolkan dengan selisih2[n].
   3. Cari nilai maksimal dari selisih1 dan selisih2, lalu nilai
      tersebut disimpan dalam variable max1 dan max2.
   4. Membagi variable max1dan max2. Hasil bagi ini merupakan
      nilai element matriks discordance alternatif l dengan
      alternatif s yang akan disimpan dalam matrix_D[k][l].

OUTPUT:
    - Matrik discordance yang merupakan hasil bagi dari nilai max
      selisih v pada kriteria j yang merupakan anggota himpunan
      discordance dengan nilai max selisih v pada kriteria j.
```

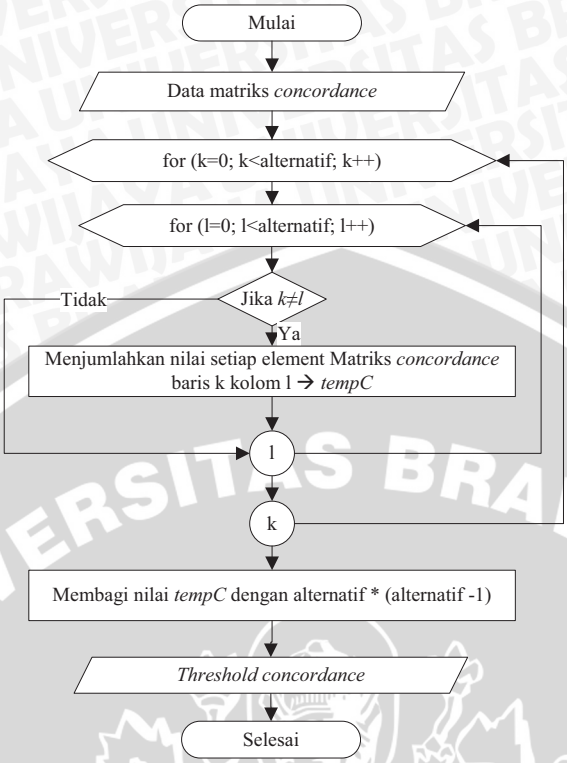
Gambar 4.28. *Pseudocode* Penentuan Matriks *Discordance*
Sumber: [Perancangan]

7. Langkah 5: Menentukan matriks dominan *concordance* dan *discordance*.

Penentuan matriks dominan *concordance* dan *discordance* dapat dilakukan setelah terbentuknya matriks *concordance* dan *discordance*. Terdapat dua bagian pada langkah ini, yakni:

a. *Concordance*

Diagram alir perhitungan nilai *threshold concordance* ditunjukkan pada gambar 4.29. dan penerapan cara kerja dari perhitungan nilai *threshold concordance* dengan *pseudocode* ditunjukkan pada gambar 4.30.



Gambar 4.29. Diagram Alir Perhitungan *Threshold Concordance*
Sumber: [Perancangan]

```
NAMA ALGORITMA: PENENTUAN THRESHOLD CONCORDANCE ELECTRE
DEKLARASI:
    - Matrix_C[m][m] adalah Double, digunakan untuk menyimpan
      matriks concordance.
    - k adalah integer, merupakan indeks alternatif.
    - l adalah integer, merupakan indeks alternatif.
    - thres_c digunakan untuk menyimpan nilai threshold
      concordance.

INPUT:
    - Matrix_C[m][m] yang berisi matriks concordance.

PROSES:
  1. Untuk setiap k ≠ l, maka akan dilakukan penjumlahan nilai
     setiap element matrix_C.
  2. tempC merupakan hasil penjumlahan dari nilai elemen
     Matrix_C[m][m]
  3. Membagi tempc  dengan jumlah alternatif(m) yang dikalikan
     dengan jumlah alternatif dikurangi 1(m-1),lalu disimpan
     pada variable c.

OUTPUT:
thres_C yang merupakan hasil pembagian dari tempC dengan jumlah
alternatif(m) yang dikalikan dengan jumlah alternatif dikurangi
1(m-1).
```

Gambar 4.30. *Pseudocode* Perhitungan *Threshold Concordance*
Sumber: [Perancangan]

Perhitungan *threshold* c̲ menggunakan rumus Persamaan (2-9). Nilai *threshold* c̲ merupakan hasil pembagian dari hasil penjumlahan setiap element matriks *concordance* pada baris ke k kolom ke j dengan hasil kali dari jumlah alternatif(m) dikali dengan jumlah alternatif dikurang 1 (m-1). Matriks *concordance* yang akan digunakan dalam langkah ini merupakan hasil dari perhitungan dari langkah 4. Berikut ini merupakan nilai dari matriks *conrodance*.

$$C = \begin{bmatrix} - & 0.70 & 1.00 & 0.70 & 0.70 \\ 0.45 & - & 0.85 & 0.85 & 0.85 \\ 0.30 & 0.30 & - & 0.30 & 0.70 \\ 0.45 & 0.45 & 0.85 & - & 0.85 \\ 0.60 & 0.30 & 1.00 & 0.30 & - \end{bmatrix}$$

Sehingga dapat dihitung dengan cara:

$$\underline{c} = \frac{\sum_{k=1}^{n}\sum_{l=1}^{n} c_{kl}}{m(m-1)}$$

$$\underline{c} = \frac{0.7+1+0.7+0.45+0.85+0.85+0.85+0.3+0.3+0.3+0.7+0.45+0.45+0.85+0.85+0.6+0.3+1+0.3}{5(5-1)}$$

$$\underline{c} = \frac{12.5}{5(5-1)} = 0.625$$

Setelah didapatkan nilai *threshold* c̲, maka langkah selanjutnya adalah menentukan matriks dominan *concordace*. Diagram alir penentuan matriks dominan *concordance* ditunjukkan pada gambar 4.31. dan penerapan cara kerja dari penentuan matriks dominan *concordance* dengan *pseudocode* ditunjukkan pada gambar 4.32.

Perhitungan matriks dominan *concordance* (*f*) menggunakan rumus Persamaan (2-10). Nilai matriks dominan *concordance* pada baris k kolom l merupakan hasil perbandingan dari nilai matriks *concordace* pada bari k kolom l dengan *threshold* c̲. Sehingga dapat dihitung dengan cara seperti dibawah ini:

- Untuk matriks dominan *concordance* (*f*) baris 2 kolom 1

  $C_{21} = \{ c_{21}\ dibanding\ \underline{c}\} = \{0.45\ dibanding\ 0.625\} = \{\ 0.45 < 0.625\}$

  c_{21} memiliki nilai lebih kecil dari $\underline{c}$, maka nilai element matriks dominan *concordance* (*f*) baris 2 kolom 1 adalah 0.

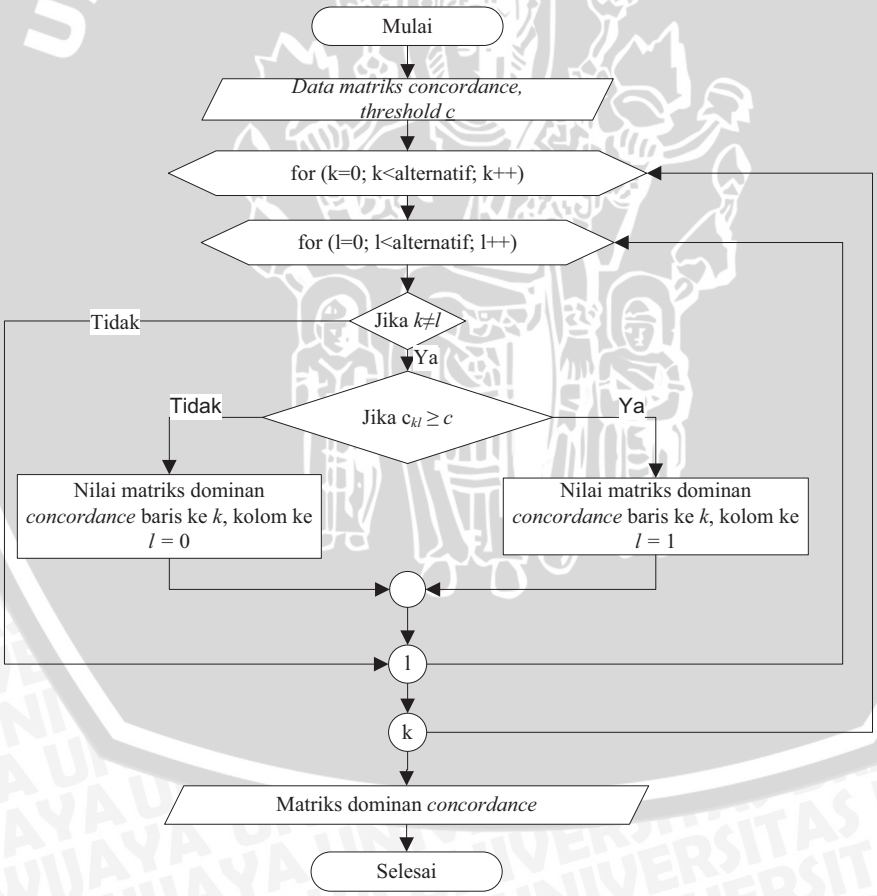
- Untuk matriks dominan *concordance* (*f*) baris1 kolom 2

$$C_{12} = \{ v_{12}\ dibanding\ \underline{c}\} = \{0.7\ dibanding 0.625\} = \{\ 0.7 > 0.625\}$$

c_{12} memiliki nilai kurang dari $\underline{c}$, maka nilai element matriks dominan *concordance* (*f*) baris 1 kolom 2 adalah 1.

Perhitungan dilakukan untuk seluruh alternatif pada seluruh kriteria yang ada. Sehingga nantinya didapatkan matriks dominan *concordance* sebagai berikut:

$$F = \begin{bmatrix} - & 1 & 1 & 1 & 1 \\ 0 & - & 1 & 1 & 1 \\ 0 & 0 & - & 0 & 1 \\ 0 & 0 & 1 & - & 1 \\ 0 & 0 & 1 & 0 & - \end{bmatrix}$$



Gambar 4.31. Diagram Alir Penentuan Dominan *Concordance*
Sumber: [Perancangan

NAMA ALGORITMA:
PENENTUAN MATRIKS DOMINAN *CONCORDANCE* ELECTRE

DEKLARASI:
- Matrix_C[m][m] adalah Double, digunakan untuk menyimpan matriks *concordance*.
- k adalah integer, merupakan indeks alternatif.
- l adalah integer, merupakan indeks alternatif.
- thres_c digunakan untuk menyimpan nilai *threshold concordance*.
- F[m][m] adalah integer, digunakan untuk menyimpan matriks dominan *concordance*.

DESKRIPSI:

INPUT:
- Matrix_C[m][m] yang berisi matriks *concordance*.

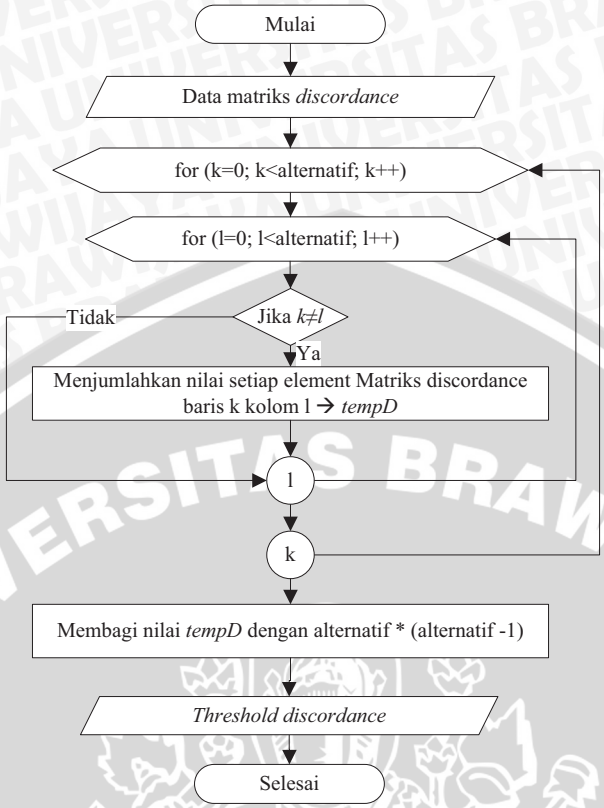
PROSES:
1. Untuk setiap k ≠ l, maka matrix_C baris ke-k kolom ke-l akan dibandingan dengan nilai *threshold concordance* (thres_c) dengan ketentuan:
   - jika matrix_C baris ke-k kolom ke-l < nilai *threshold concordance* (thres_c), maka nilai F baris ke-a kolom ke-j adalah 0.
   - jika matrix_C baris ke-k kolom ke-j ≥ nilai *threshold concordance* (thres_c), maka nilai matriks_F baris ke-k kolom ke-j adalah 1.

OUTPUT:
matriks dominan *concordance*, yang merupakan hasil perbandingan antara matriks *concordance* dengan *threshold*.

Gambar 4.32. *Pseudocode* Penentuan Matriks Dominan *Concordance*
Sumber: [Perancangan]

b. *Discordance*

Diagram alir perhitungan nilai *threshold discordance* ditunjukkan pada gambar 4.33. dan penerapan cara kerja dari perhitungan nilai *threshold discordance* dengan *pseudocode* ditunjukkan pada gambar 4.34.



Gambar 4.33. Diagram Alir Perhitungan *Threshold Discordance*
Sumber: [Perancangan]

```
NAMA ALGORITMA: PENENTUAN THRESHOLD DISCORDANCE ELECTRE
DEKLARASI:
    - Matrix_D[m][m] adalah Double, digunakan untuk menyimpan
      matriks concordance.
    - k adalah integer, merupakan indeks alternatif.
    - l adalah integer, merupakan indeks alternatif.
    - thres_d digunakan untuk menyimpan nilai threshold
      discordance.

INPUT:
    - Matrix_D[m][m] yang berisi matriks discordance.

PROSES:
   1. Untuk setiap k ≠ l, maka akan dilakukan penjumlahan nilai setiap
      element matrix_D.
   2. tempD merupakan hasil perhitungan dari jumlah nilai elemen
      matrix_D.
   3. Membagi tempD dengan lternatif(m) yang dikalikan dengan jumlah
      alternatif dikurangi l(m-l), lalu disimpan pada thres_d.

OUTPUT:
thres_d yang merupakan hasil pembagian dari tempD dengan
lternatif(m) yang dikalikan dengan jumlah alternatif dikurangi l(m-
l).
```

Gambar 4.34. *Pseudocode* Perhitungan *Threshold Discordance*
Sumber: [Perancangan]

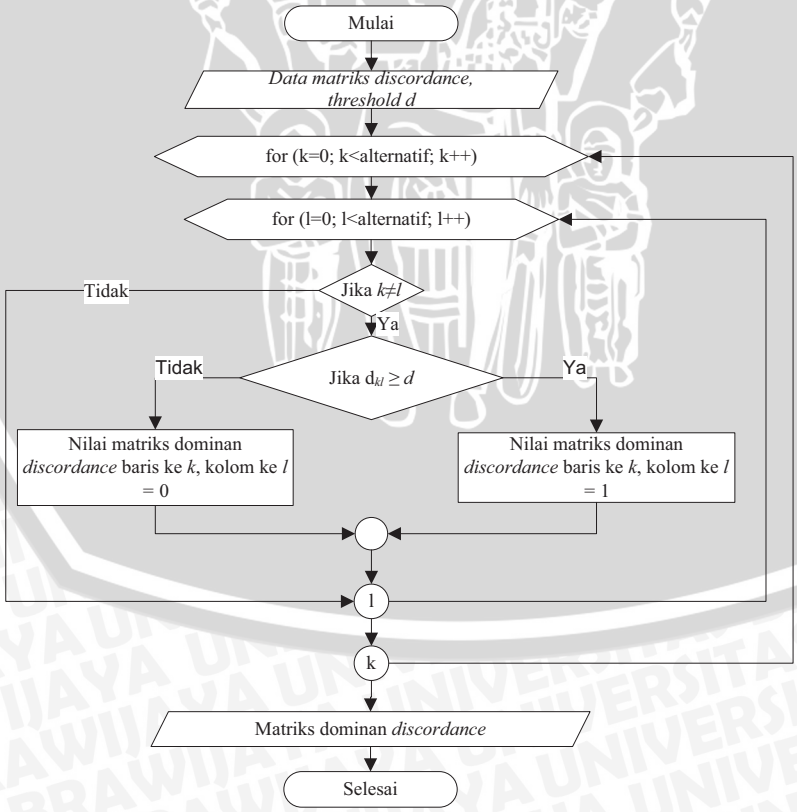
Perhitungan *threshold* d menggunakan rumus Persamaan (2-11). Nilai *threshold* d merupakan hasil pembagian dari hasil penjumlahan setiap element matriks *discordance* pada baris ke k kolom ke l dengan hasil kali dari jumlah alternatif(m) dikali dengan jumlah alternatif dikurang 1 (m-1). Sehingga dapat dihitung dengan cara:

$$\underline{d} = \frac{\sum_{k=1}^{n}\sum_{l=1}^{n} d_{kl}}{m(m-1)}$$

$$= \frac{0.3375+0+0.31361+0.2360+1+0.2880+0.6456+0.4357+1+1+1+1+1+1+0.4461+1+1+1+0+0.7684}{5(5-1)}$$

$$= 0.813$$

Setelah didapatkan nilai *threshold* d, maka langkah selanjutnya adalah menentukan matriks dominan *discordace*. Diagram alir penentuan matriks dominan *discordance* ditunjukkan pada gambar 4.35. dan penerapan cara kerja dari penentuan matriks dominan *discordance* dengan *pseudocode* ditunjukkan pada gambar 4.36.



Gambar 4.35. Diagram Alir Penentuan Dominan *Discordance*
Sumber: [Perancangan]

NAMA ALGORITMA:
PENENTUAN MATRIKS DOMINAN *DISCORDANCE* ELECTRE

DEKLARASI:
- Matrix_D[m][m] adalah Double, digunakan untuk menyimpan matriks *discordance*.
- k adalah integer, merupakan indeks alternatif.
- l adalah integer, merupakan indeks alternatif.
- Thres_D digunakan untuk menyimpan nilai *threshold discordance*.
- G[m][m] adalah integer, digunakan untuk menyimpan matriks dominan *discordance*.

INPUT:
- MatriX_D[m][m] yang berisi matriks *discordance*.

PROSES:
1. Untuk setiap k ≠ l, maka matrix_D baris ke-k kolom ke-l akan dibandingan dengan nilai *threshold discordance* (Thres_D) dengan ketentuan:
   - jika matrix_D baris ke-k kolom ke-j < nilai *threshold discordance* (Thres_D), maka nilai G baris ke-k kolom ke-j adalah 0.
   - jika matrix_D baris ke-k kolom ke-k ≥ nilai *threshold discordance* (Thres_D), maka nilai G baris ke-k kolom ke-j adalah 1.

OUTPUT:
matriks dominan *discordance*, yang merupakan hasil perbandingan antara matriks *discordance* dengan *threshold*.

Gambar 4.36. *Pseudocode* Penentuan Matriks Dominan *Discordance*
Sumber: [Perancangan]

Perhitungan matriks dominan *discordance* (*g*) menggunakan rumus Persamaan (2-12). Nilai matriks dominan *discordance* pada baris k kolom l merupakan hasil perbandingan dari nilai matriks *discordace* pada baris k kolom l dengan *threshold* d. Sehingga dapat dihitung dengan cara seperti dibawah ini:

- Untuk matriks dominan *discordance* (*g*) baris 1 kolom 2

$g_{12} = \{v_{12}\ dibanding\ d\} = \{1.175\ dibanding\ 0.813\}$

$\{0.377 < 0.813\}$

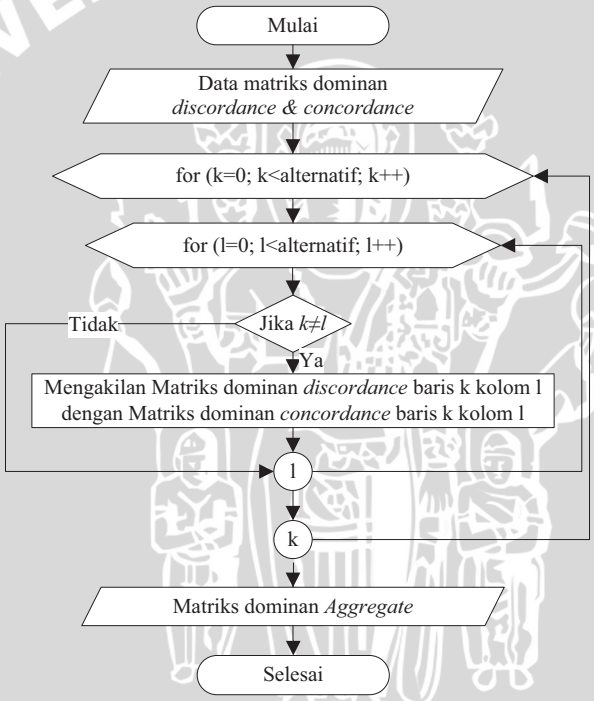
g_{12} memiliki nilai lebih kecil dari d, maka nilai matriks dominan *discordance* (*g*) baris 1 kolom 2 adalah 1.

Perhitungan dilakukan untuk seluruh alternatif pada seluruh kriteria yang ada. Sehingga nantinya didapatkan matriks dominan *concordance* sebagai berikut:

$$G = \begin{bmatrix} - & 0 & 0 & 0 & 0 \\ 1 & - & 0 & 0 & 0 \\ 1 & 1 & - & 1 & 1 \\ 1 & 1 & 0 & - & 1 \\ 1 & 1 & 0 & 1 & - \end{bmatrix}$$

8. Langkah 6: Menentukan *Aggregate Dominance Matrix*

Diagram alir penentuan matriks dominan *aggregate* ditunjukkan pada gambar 4.37. dan penerapan cara kerja dari penentuan matriks dominan *aggregate* dengan *pseudocode* ditunjukkan pada gambar 4.38.



Gambar 4.37. Diagram Alir Penentuan Matriks Dominan *Aggregate*
Sumber: [Perancangan]

```
NAMA ALGORITMA:
PENENTUAN MATRIKS DOMINAN AGGREGATE ELECTRE

DEKLARASI:
    - k adalah integer, merupakan indeks alternatif.
    - l adalah integer, merupakan indeks alternatif.
    - F[m][m] adalah integer, digunakan untuk menyimpan matriks
      dominan concordance.
    - G[m][m] adalah integer, digunakan untuk menyimpan matriks
      dominan discordance.
    - E[m][m] adalah integer, digunakan untuk menyimpan matriks
      dominan aggregate.
```

```
DESKRIPSI:

INPUT:
    -  F[m][m] yang berisi matriks dominan concordance.
    -  G[m][m] yang berisi matriks dominan discordance.

PROSES:
   1. Untuk setiap k ≠ l, maka dilakukan pengalian antara F baris ke-
      k kolom ke-j dengan D baris ke-k kolom ke-j. Hasil perkalian
      ini menghasilkan matriks dominan aggregate (E) baris ke-k kolom
      ke-j.

OUTPUT:
matriks dominan aggregate,yang merupakan hasil perkalian antara
matriks dominan concordance dengan matriks dominan discordance.
```

Gambar 4.38. *Pseudocode* Penentuan Matriks Dominan *Aggregate*
Sumber: [Perancangan]

Perhitungan matriks dominan *aggregate* (*e*) menggunakan rumus Persamaan (2-13). Nilai matriks dominan *aggregate* pada baris k kolom l merupakan hasil perkalian dari nilai matriks *concordance* pada baris k kolom l dengan nilai matriks *discordance* pada bari k kolom l. Matriks dominan *concordance* adalah sebagai berikut:

$$F = \begin{bmatrix} - & 1 & 1 & 1 & 1 \\ 0 & - & 1 & 1 & 1 \\ 0 & 0 & - & 0 & 1 \\ 0 & 0 & 1 & - & 1 \\ 0 & 0 & 1 & 0 & - \end{bmatrix}$$

Matriks dominan *discordance* adalah sebagai berikut:

$$G = \begin{bmatrix} - & 0 & 0 & 0 & 0 \\ 1 & - & 0 & 0 & 0 \\ 1 & 1 & - & 1 & 1 \\ 1 & 1 & 0 & - & 1 \\ 1 & 1 & 0 & 1 & - \end{bmatrix}$$

Sehingga dapat dihitung dengan cara:

- Untuk *aggregate dominance matrix* baris 2 kolom 1

$e_{21} = f_{21} * g_{21} = 0 * 1 = 0$

Maka nilai *aggregate dominance matrix* baris 2 kolom 1 adalah 0

- Untuk *aggregate dominance matrix* baris 1 kolom 2

$e_{12} = f_{12} * g_{12} = 1 * 0 = 0$

Maka nilai *aggregate dominance matrix* baris 1 kolom 2 adalah 0

- Untuk *aggregate dominance matrix* baris 3 kolom 1

$e_{31} = f_{31} * g_{31} = 0 * 1 = 0$

Maka nilai *aggregate dominance matrix* baris 3 kolom 1 adalah 0

- Untuk *aggregate dominance matrix* baris 3 kolom 4

$e_{34} = f_{34} * g_{34} = 0 * 1 = 0$

Maka nilai *aggregate dominance matrix* baris 3 kolom 4 adalah 0

Perhitungan dilakukan untuk seluruh alternatif pada seluruh kriteria yang ada. Sehingga nantinya didapatkan matriks dominan *concordance* sebagai berikut:

$$E = \begin{bmatrix} - & 0 & 0 & 0 & 0 \\ 0 & - & 0 & 0 & 0 \\ 0 & 0 & - & 0 & 1 \\ 0 & 0 & 0 & - & 1 \\ 0 & 0 & 0 & 0 & - \end{bmatrix}$$

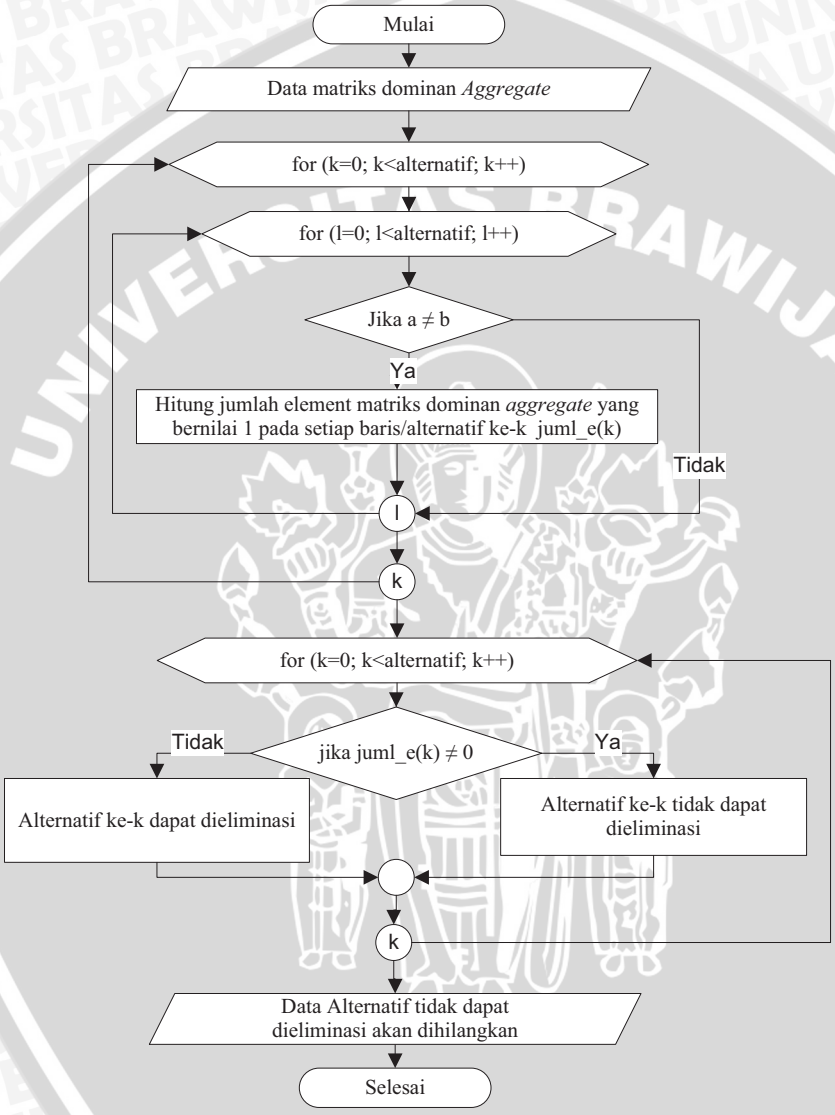
9. Langkah 7: Eliminasi alternatif yang *less favourable*

Langkah selanjutnya yaitu eliminasi alternatif yang *less favourable*. Berdasarkan nilai *aggregate dominance matrix* yang telah didapatkan sebelumnya, dengan ketentuan setiap baris yang ada pada *aggregate dominance matrix* mewakili setiap alternatif calon siswa baru ke-k.

$$E = \begin{bmatrix} - & 0 & 0 & 0 & 0 \\ 0 & - & 0 & 0 & 0 \\ 0 & 0 & - & 0 & 1 \\ 0 & 0 & 0 & - & 1 \\ 0 & 0 & 0 & 0 & - \end{bmatrix} \begin{matrix} A1 \\ A2 \\ A3 \\ A4 \\ A5 \end{matrix}$$

Kesimpulan, A1, A2, dan A5 dapat dieliminasi karena memiliki elemen yang bernilai 0, dengan kata lain A1, A2, A5 termasuk ke dalam kelompok yang tidak diterima. Sedangkan A3 dan A4 memiliki elemen yang memiliki nilai 1 sehingga masuk ke dalam kelompok yang diterima. Berdasarkan nilai agregasi matriks dominan E yang telah didapatkan sebelumnya dapat disimpulkan bahwa alternatif dapat dieliminasi karena memiliki nilai element bernilai satu lebih sedikit dibandingkan dengan alternatif alternatif. Dengan demikian, alternatif alternatif masuk ke dalam kelompok yang diterima yang akan dilanjutkan pada perhitungan

menggunakan metode *Weighted Product*, sedangkan alternatif, alternated masuk ke dalam kelompok yang tidak diterima. Diagram alir eliminasi alternatif ditunjukkan pada gambar 4.39. dan penerapan cara kerja dari eliminasi alternatif dengan *pseudocode* ditunjukkan pada gambar 4.40.



Gambar 4.39. Diagram Alir Eliminasi Alternatif
Sumber: [Perancangan]

```
NAMA ALGORITMA:
ELIMINASI ALTERNATIF LESS FAVOURABLE

DEKLARASI:
    - k adalah integer, merupakan indeks alternatif.
    - l adalah integer, merupakan indeks alternatif.
    - eliminasi[a] adalah integer, banyaknya element E yang
      bernilai 1
```

```
    - E[m][m] adalah integer, digunakan untuk menyimpan matriks
      dominan aggregate.

DESKRIPSI:
INPUT:
    - E[m][m] yang berisi matriks dominan aggregate.

PROSES:
   1. Cari jumlah/banyaknya elemen E baris ke k kolom j yang bernilai
      1 pada setiap baris k, lalu simpan pada matriks eliminasi[a].
   2. Cari nilai maksimal dari eliminasi[a].
   3. Jika nilai e pada indeks ke-k tersebut adalah yang terbesar
      atau ≠ 0, maka alternatif ke-k dapat dieliminasi.

OUTPUT:
Alternatif dapat dieleiminasi ketika element matriks_E bernilai ≠ 0.
```

Gambar 4.40. *Pseudocode* Eliminasi Alternatif
Sumber: [Perancangan]

10. Pengambilan Keputusan

Berdasarkan langkah 1-9 yang telah dijabarkan sebelumnya, maka didapatkan 2 kelompok calon siswa baru. Dua kelompok tersebut adalah kelompok calon siswa baru yang diterima dan kelompok calon siswa baru yang tidak diterima. Kelompok calon siswa baru tersebut dapat dilihat pada Tabel 4.29 di bawah ini:

Tabel 4.29 Tabel Pengambilan Keputusan

Alternatif	Kelompok Calon siswa baru
A1	Tidak Diterima
A2	Tidak Diterima
A3	Diterima
A4	Diterima
A5	Tidak Diterima

4.2.3.2 Proses Perhitungan *Weighted Product*

Proses perhitungan dimulai dengan mengambil data alternatif yang didapatkan dari hasil proses perhitungan ELECTRE yakni kelompok calon siswa baru yang diterima. Data alternatif tersebut akan digunakan untuk menentukan kelas

peminatan dengan menggunakan metode *Weighted Product*. Berikut ini merupakan langkah-langkah yang dilakukan dalam perhitungan *Weighted Product*:

1. Langkah 1: Ambil Data Alternatif

Pada proses ini, data yang diambil adalah data hasil seleksi perhitungan ELECTRE yang telah dilakukan sebelumnya. Data yang digunakan adalah data yang masuk ke dalam kelompok diterima sesuai dengan perhitungan ELECTRE. Sebagai contoh, pada perhitungan ELECTRE sebelumnya, data yang digunakan adalah data A1, A2, A3, A4, dan A5. Data A1, A2, A5 adalah alternatif yang masuk ke dalam kelompok yang tidak diterima sedangkan data A3, A4 masuk ke dalam kelompok yang diterima. Maka dalam perhitungan *Weighted Product*, hanya data A3 dan A4 yang akan dihitung oleh metode *Weighted Product*. Data dapat dilihat pada Tabel 4.30.

Tabel 4.30 Data Perhitungan *Weighted Product*

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
D3	80.2	79	80	81.3	35	90	64	58	IPA	IPA	105	C	C	C	C-	C-	C	C	C	C+	IPA
D4	87.4	87	87.2	82.7	33	90	76	68	IPA	IPA	115	C+	C+	C	C-	C+	T	C+	C	C	IPA

Sumber: [Perancangan]

Keterangan Tabel 4.30:

1 Calon siswa baru
2 Nilai Rata-Rata Mat
3 Nilai Rata-Rata IPA
4 Nilai Rata-Rata IPS
5 Nilai Tes Tulis (TPA)
6 Wawancara Orang Tua
7 Wawancara Calon siswa baru
8 Tes Peminatan IPA
9 Tes Peminatan IPS
10 Angket Peminatan Anak
11 Angkat Dukungan Orang Tua
12 IQ
13 Aspek Daya Tangkap
14 Aspek Analisa
15 Aspek Logika
16 Aspek Verbal
17 Aspek Hitung
18 Aspek Ruang
19 Aspek Memori
20 Aspek Konsentrasi
21 Aspek Mekanik
22 Rekomendasi BK

2. Langkah 2: Ambil nilai bobot tiap kriteria

Bobot yang digunakan dalam perhtiungan ini merupakan bobot yang telah dijelaskan pada Tabel 4.21. Nilai bobot untuk masing-masing kriteria ditunjukkan pada Tabel 4.31 berikut.

Tabel 4.31 Bobot Peminatan

	Kriteria	Bobot		Kriteria	Bobot
C1	Nilai rata-rata matematika	1	C12	Aspek kecerdasan daya tangkap	1
C2	Nilai rata-rata IPA	1	C13	Aspek kecerdasan analisa	1
C3	Nilai rata-rata IPS	1	C14	Aspek kecerdasan logika	1
C4	Nilai tes tulis	1	C15	Aspek kecerdasan verbal	1
C5	Nilai Wawancara calon siswa baru	1	C16	Aspek kecerdasan hitung	1
C6	Nilai wawancara orang tua	1	C17	Aspek kecerdasan ruang	1
C7	Nilai tes peminatan IPA	1	C18	Aspek kecerdasan memori	1
C8	Nilai tes peminatan IPS	1	C19	Aspek kecerdasan konsen	1
C9	Angket peminatan anak	1	C20	Aspek kecerdasan mekanik	1
C10	Angket dukungan orang tua	1	C21	Rekomendasi BK	1
C11	IQ	1			

3. Langkah 3: Melakukan Pembobotan Bobot Terhadap Data

Pembobotan terhadap data dilakukan sesuai dengan cara membagi bobot kriteria dengan jumlah total kriteria yang ada. Sistem akan melakukan pembobotan kriteria (w) dengan perhitungan sebagai berikut:

$$W_1 = \frac{C_1}{C_1 + C_2 + C_3 + \cdots . + C_{21}} = \frac{1}{1 + 1 + 1 + \cdots . + 1} = \frac{1}{15}$$

$$= 0.067$$

$$W_2 = \frac{C_2}{C_1 + C_2 + C_3 + \cdots . + C_{21}} = \frac{1}{1 + 1 + 1 + \cdots . + 1} = \frac{1}{15}$$

$= 0.067$

$$W_3 = \frac{C_3}{C_1 + C_2 + C_3 + \cdots . + C_{21}} = \frac{1}{1 + 1 + 1 + \cdots . + 1} = \frac{1}{15}$$

$= 0.067$

$$W_4 = \frac{C_4}{C_1 + C_2 + C_3 + \cdots . + C_{21}} = \frac{1}{1 + 1 + 1 + \cdots . + 1} = \frac{1}{15}$$

$= 0.067$

Dan seterusnya hingga W_{21}.

Tabel 4.32 Hasil Pembobotan

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15
W1	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067
W2	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067	0.067

4. Langkah 4: Membuat Matriks data alternatif sesuai dengan kriteria

Data hasil seleksi yang telah diambil sebelumnya akan diubah menjadi matriks keputusan, dimana nilai berupa angka dan nilai berupa bukan angka akan dikonversi terlebih dahulu sesuai dengan range yang telah ditentukan pada Tabel 4.22. hingga Tabel 4.24. Matriks keputusan akan dibuat berdasarkan Tabel 4.30. Maka matriks keputusan konversi dapat dilihat pada Tabel 4.33.

Tabel 4.33 Matriks Keputusan Data Alternatif

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15
A1	80.2	79	81.3	35	90	64	1	1	105	3	3	2	3	4	1
A2	80.2	80	81.3	35	90	58	1	1	105	3	2	3	3	3	1

Sumber: [Perancangan]

5. Langkah 5: Menghitung Vektor S

Langkah selanjutnya adalah menghitung Vektor S, dimana data akan dikalikan tetapi sebelumnya dilakukan pemangkatan dengan bobot dari masing-masing kriteria. Diagram alir perhitungan vektor S ditunjukkan pada gambar 4.41. dan penerapan cara kerja dari perhitungan vektor S dengan *pseudocode* ditunjukkan

pada gambar 4.42. Perhitungan Vektor S menggunakan Persamaan 2-14. Nilai bobot bisa bernilai negatif apabila kriteria tersebut merugikan.

S_1

$= 80.2^{0.067} x 79^{0.067} x 81.3^{0.067} x 35^{0.067} x 90^{0.067} x 64^{0.067} x 3^{0.067} x 3^{0.067} x\ 105^{0.067} x 3^{0.067}$

$3^{0.067} x 2^{0.067} x 3^{0.067} x 4^{0.067} x 3^{0.067} = 10.1$

S_2

$= 80.2^{0.067} x 80^{0.067} x 81.3^{0.067} x 35^{0.067} x 90^{0.067} x 58^{0.067} x 2^{0.067} x 2^{0.067} x 105^{0.067} x 3^{0.067} x 2^{0.067}$

$x 3^{0.067} x 3^{0.067} x 3^{0.067} x 2^{0.067}$

$= 9.1$

NAMA ALGORITMA:
Menghitung Nilai Vektor S
DEKLARASI:

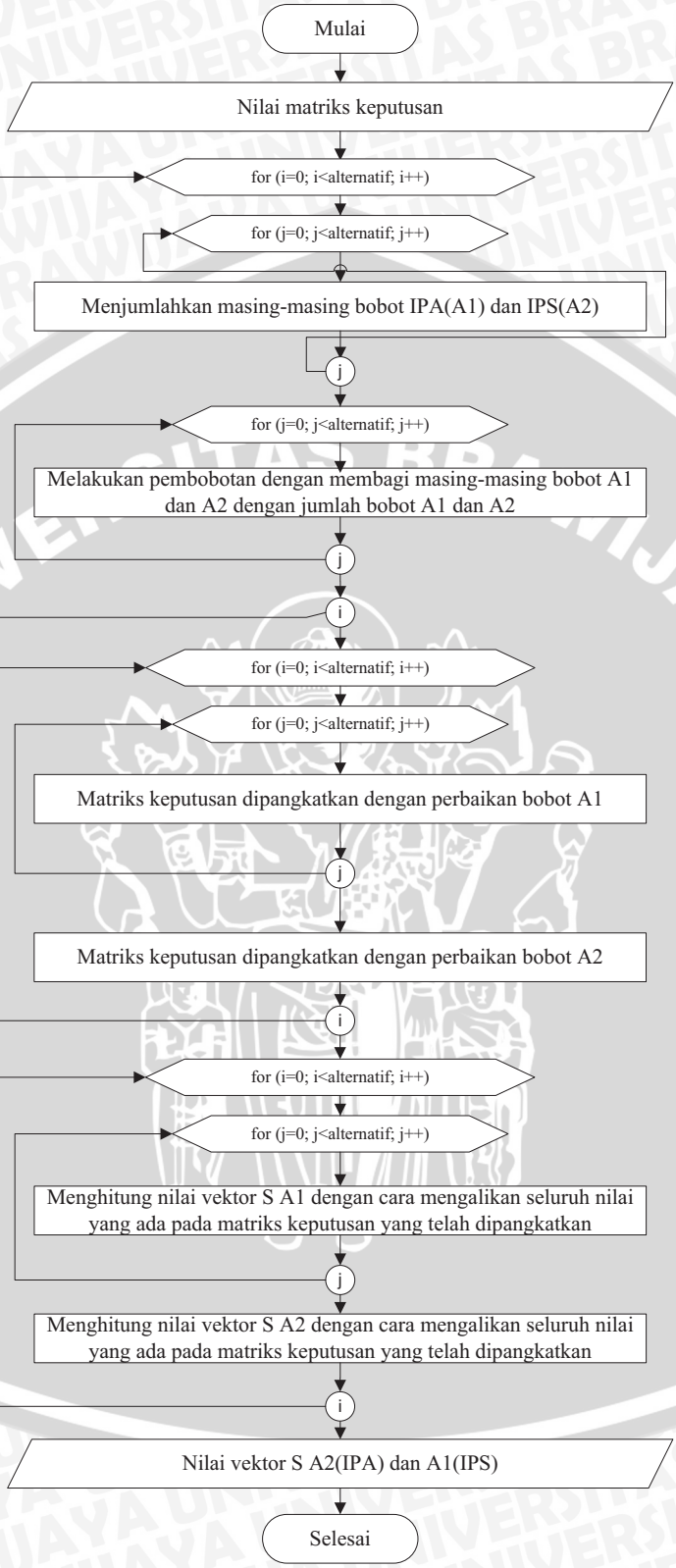
- BobotIPA[c] adalah Double, digunakan untuk hasil pembagian anatara bobot alternatif c dengan totalIPA pada peminatan IPA.
- BobotIPS[c] adalah Double, digunakan untuk hasil pembagian anatara bobot alternatif c dengan totalIPS pada peminatan IPS.
- c adalah integer, merupakan indeks kriteria.
- m2 adalah integer, merupakan indeks alternatif siswa.
- n2 adalah integer, merupakan indeks alternatif peminatan.
- Matriks_s[m2][n2][c] adalah string, digunakan untuk menyimpan data alternatif siswa ke-i dan kriteria j pada masing-masing alternatif penjurusan.
- Matriks_s_convert[[m2][n2][c] adalah double, digunakan untuk menyimpan data alternatif siswa ke-i dan kriteria j pada masing-masing alternatif penjurusan.
- Matriks_w[z,n2,c] adalah double, digunakan untuk menyimpan bobot untuk setiap alternatif.
- totalIPA[c] adalah double, digunakan untuk menyimpan jumlah bobot untuk alternatif IPA yang didapat dari matriks_w.
- totalIPS[c] adalah double, digunakan untuk menyimpan jumlah bobot untuk alternatif IPS yang didapat dari matriks_w.
- bobotIPA[c] adalah double, digunakan untuk menyimpan hasil pembagian yang didapat dari matriks_w dengan totalIPA.
- bobotIPS[c] adalah double, digunakan untuk menyimpan hasil pembagian yang didapat dari matriks_w dengan totalIPS.
- Mpow[m2, n2, c] adalah double, digunakan untuk menyimpan hasil dari Matriks_s_convert[[m2][n2][c] dipangkatkan dengan masing-masing bobot alternatif (bobotIPA/bobotIPS).
- T_ipa[m2] adalah double, digunakan untuk mengkalikan seluruh hasil Mpow[m2, n2, c]. t_ipa adalah hasil vektor s dari alternatif IPA.
- T_ips[m2] adalah double, digunakan untuk mengkalikan seluruh hasil Mpow[m2, n2, c]. t_ips adalah hasil vektor s dari alternatif IPs.

DESKRIPSI:
INPUT:

```
      - Matriks_s[m2][n2][c] adalah string, digunakan untuk menyimpan
        data alternatif siswa ke-i dan kriteria j pada masing-masing
        alternatif penjurusan.
      - Matriks_w[z,n2,c] adalah double, digunakan untuk menyimpan
        bobot untuk setiap alternatif.


PROSES:
      - Mengambil seluruh data alternatif yang dihasilkan dari
        perhitungan ELECTRE, lalu data tersebut disimpan pada
        Matriks_s[m2][n2][c].
      - Melakukan konversi Matriks_s[m2][n2][c], lalu hasilnya
        disimpan pada Matriks_s_convert[[m2][n2][c].
      - Mengambil bobot untuk masing-masing kriteria dan disimpan pada
        Matriks_w[z,n2,c].
      - Menghitung nilai totalIPA dan totalIPS dengan cara
        menjumlahkan Matriks_w[z,n2,c].
      - Menghitung nilai bobotIPA dan bobotIPS dengan cara membagi
        Matriks_w[z,n2,c] dengan totalIPA dan totalIPS.
      - Nilai dari Matriks_s_convert[[m2][n2][c] dipangkatkan dengan
        bobotIPA dan bobotIPS, dan disimpan ke dalam variable
        Mpow[m2, n2, c].
      - Mengitung T_ipa[m2] dengan cara mengalikan seluruh Mpow[m2,
        n2, c].
      - Mengitung T_ips[m2] dengan cara mengalikan seluruh Mpow[m2,
        n2, c].
      -
OUTPUT:
      - Nilai dari t_ipa[m2] dan t_ips[m2]
```

Gambar 4.41. *Pseudocode* Menghitung Vektor S
Sumber: [Perancangan]



Gambar 4.42. Alur Data Menghitung Vektor S
Sumber: [Perancangan]

6. Langkah 6: Menghitung Vektor V

Langkah selanjutnya adalah menghitung Vektor V, dengan cara membagi hasil masing-masing vektor S dengan jumlah seluruh vektor S. Diagram alir perhitungan vektor V ditunjukkan pada gambar 4.44. dan penerapan cara kerja dari perhitungan vektor V dengan *pseudocode* ditunjukkan pada gambar 4.43. Perhitungan Vektor V menggunakan Persamaan 2-15.

$$V_1 = \frac{S_1}{S_1 + S_2} = \frac{10.1}{10.1 + 9.1} = 0.52$$

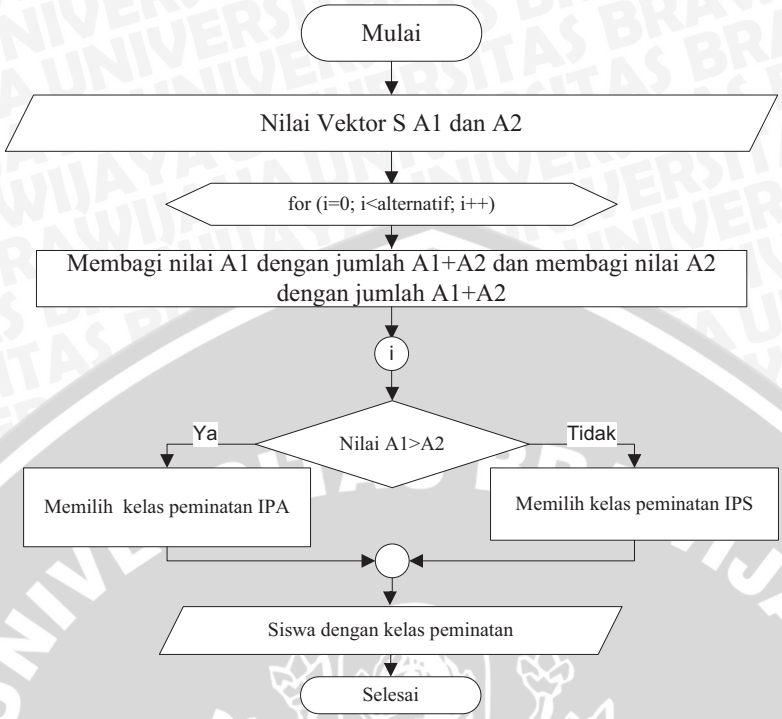
$$V_2 = \frac{S_1}{S_1 + S_2} = \frac{10.1}{10.1 + 9.1} = 0.47$$

```
NAMA ALGORITMA:
Menghitung Nilai Vektor V
DEKLARASI:
      - v_ipa_nilai[m2] adalah double, digunakan untuk menyimpan
        nilai vektor V alternatif IPA.
      - v_ips_nilai[m2] adalah double, digunakan untuk menyimpan
        nilai vektor V alternatif IPS.

DESKRIPSI:
INPUT:
      - T_ipa[m2] adalah double, digunakan untuk mengkalikan
        seluruh hasil Mpow[m2, n2, c]. t_ipa adalah hasil vektor
        s dari alternatif IPA.
      - T_ips[m2] adalah double, digunakan untuk mengkalikan
        seluruh hasil Mpow[m2, n2, c]. t_ipa adalah hasil vektor
        s dari alternatif IPs.
PROSES:
      - Melakukan penjumlahan t_ipa[m2] dan t_ips[m2].
      - Membagi nilai t_ipa[m2] dengan nilai penjumlahan
        t_ipa[m2] dan t_ips[m2, lalu disimpan pada
        v_ipa_nilai[m2].
      - Membagi nilai t_ips[m2] dengan nilai penjumlahan
        t_ipa[m2] dan t_ips[m2, lalu disimpan pada
        v_ips_nilai[m2].

OUTPUT:
      - Nilai dari v_ipa_nilai[m2] dan v_ips_nilai[m2]
```

Gambar 4.43. *Pseudocode* Menghitung Vektor V
Sumber: [Perancangan]



Gambar 4.44. *Pseudocode* Menghitung Vektor V
Sumber: [Perancangan]

7. Langkah 7: Ambil Keputusan

Pengambilan keputusan bergantung pada nilai Vektor S. Sistem akan memilih nilai tertinggi sebagai alternatif terbaik dalam mengambil keputusan. Dari perhitungan vektor V pada Langkah 6 di atas, maka didapatkan hasil rekomendasi bahwa A1 masuk dalam kelas peminatan IPA. Hal ini dikarenakan nilai $V_1 > V_2$ yang berarti nilai peminatan IPA lebih besar dari nilai peminatan IPS.

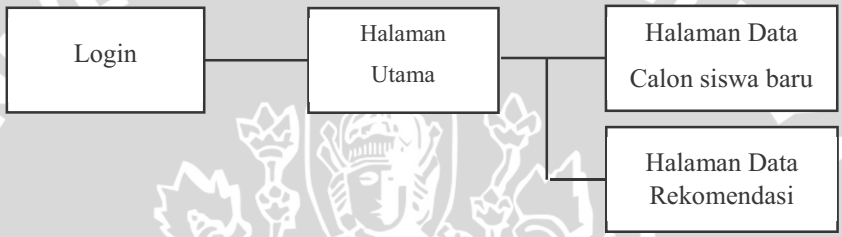
Hasil perhitungan untuk keseluruhan metode yang digunakan ditunjukkan pada Tabel 4.34.

Tabel 4.34 Kesimpulan

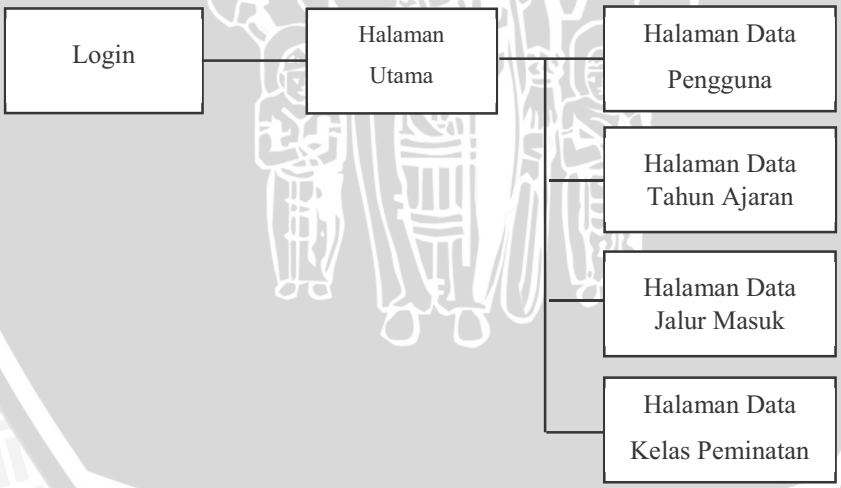
Calon siswa baru ke-	Hasil Perhitungan ELECTRE	Hasil Perhitungan WP
1	Tidak Diterima	(Tidak dihitung)
2	Tidak Diterima	(Tidak dihitung)
3	Diterima	IPA
4	Diterima	IPA
5	Tidak Diterima	(Tidak dihitung)

4.2.4 Subsistem Antarmuka

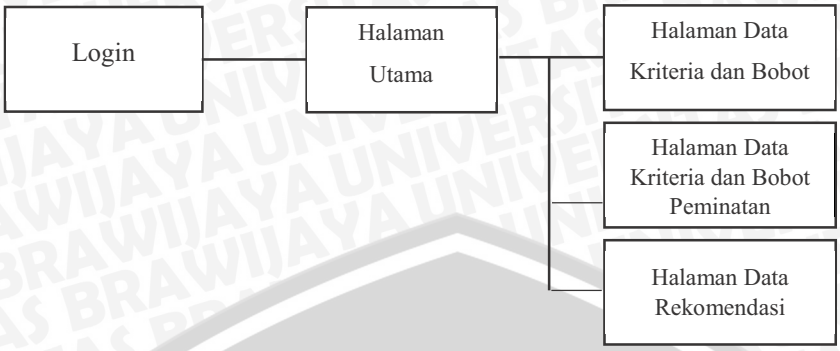
Subsistem antarmuka ini bertujuan agar pengguna dapat berkomunikasi dan memberikan perintah pada sistem. Subsistem antarmuka dalam sistem ini akan dijelaskan melalui alur *sitemap, flowchart* dan desain antarmuka tiap halaman. Subsistem ini dibuat untuk menggambarkan aplikasi yang dibuat. Berikut ini *sitemap* yang ditujukan untuk masing-masing pengguna, *sitemap* untuk IT dapat dilihat pada Gambar 4.46, *sitemap* untuk Kepala Sekolah dan Ketua PPDB dapat dilihat pada Gambar 4.45, *sitemap* untuk kepala sekolah dapat dilihat pada Gambar 4.45, *sitemap* untuk Sie-Test/Guru BK dapat dilihat pada Gambar 4.47.



Gambar 4.45 Sitemap Halaman *Sie Test dan Guru BK*
Sumber: [Perancangan]



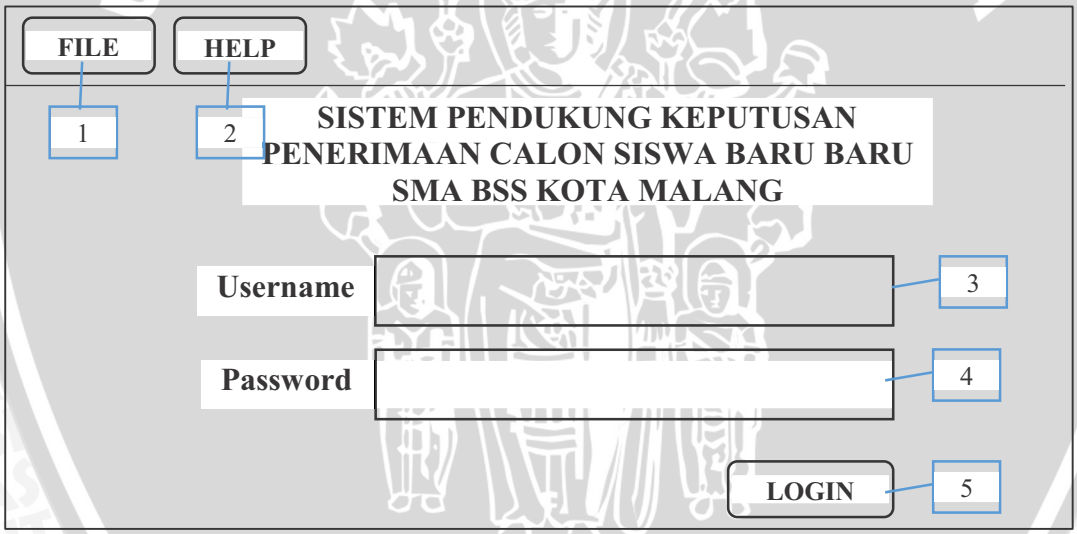
Gambar 4.46 *Sitemap* Halaman IT
Sumber: [Perancangan]



Gambar 4.47 *Sitemap* Halaman Kepala Sekolah/Ketua PPDB
Sumber: [Perancangan]

4.2.4.1 Halaman *Login*

Pada halaman *login*, IT, PPDB, Kepala sekolah, c-test, atau BK dapat memulai untuk menjalankan sistem dengan menginputkan *username* dan *password*. Perancangan tampilan untuk halaman *login* ditunjukkan pada Gambar 4.48.



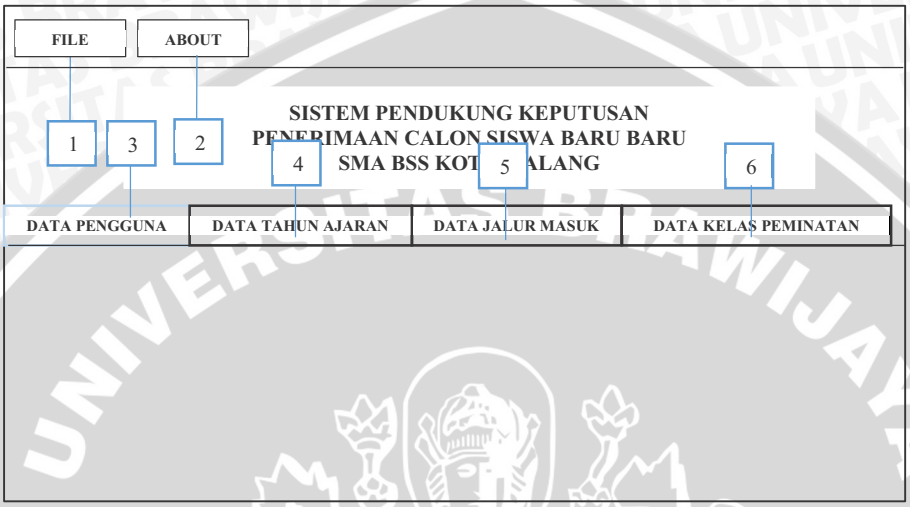
Gambar 4.48 Perancangan Antarmuka Login
Sumber: [Perancangan]

Keterangan Gambar 4.48:

1. *Taskbar File* (berisi menu *Exit*)
2. *Taskbar Help* (berisi menu *Help* dan *About*)
3. Form untuk menginputkan *username*
4. Form untuk menginputkan *password*
5. Tombol *Login*

4.2.4.2 Halaman *Home* IT

Setelah berhasil melakukan login, maka pengguna akan masuk ke halaman utama yang merupakan halaman utama dari aplikasi ini. Halaman ini ditujukan untuk pengguna *IT Service*. Pada halaman ini terdapat beberapa tab menu. Perancangan tampilan untuk halaman utama ditunjukkan pada Gambar 4.49.



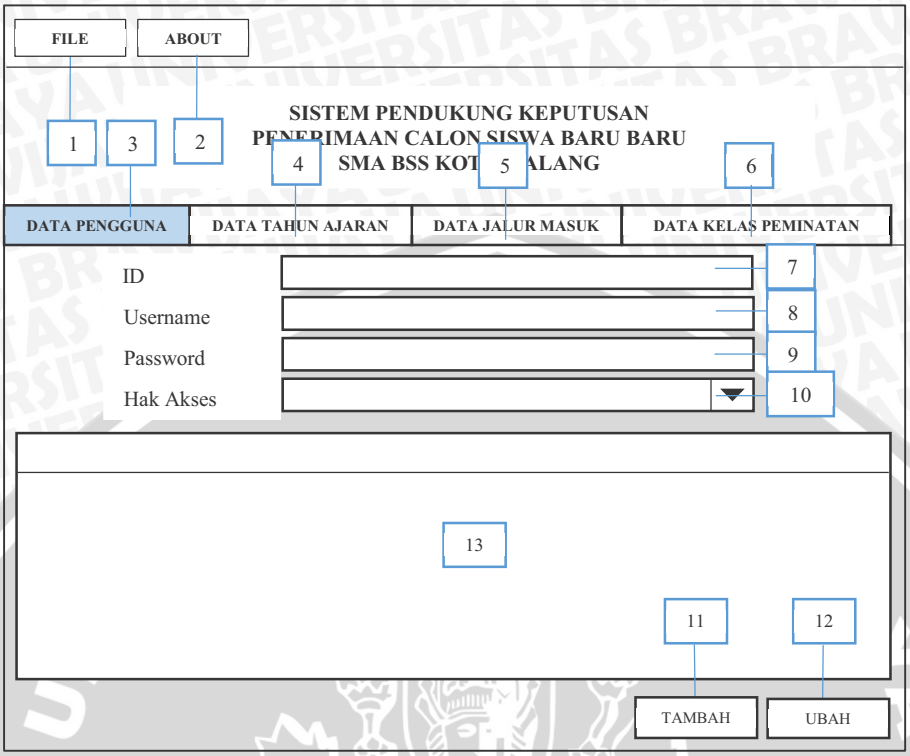
Gambar 4.49 Perancangan Antarmuka Halaman Utama Admin
Sumber: [Perancangan]

Keterangan Gambar 4.49:

1. *Taskbar File* (dengan *dropdown* menu *logout* dan *exit*)
2. *Taskbar About* (dengan *dropdown* menu *about*)
3. Tab Menu Data Pengguna
4. Tab Menu Data Tahun Ajaran
5. Tab Menu Data Jalur Masuk
6. Tab Menu Data Kelas Peminatan

4.1.1.1. Halaman Data Pengguna

Setelah berhasil melakukan login dan memilih tab menu data pengguna, maka pengguna akan masuk ke halaman data pengguna yang merupakan halaman yang digunakan untuk mengelola data pengguna sistem. Kelola data yang dapat dilakukan antara lain menambah, mengubah, menghapus dan melihat data pengguna sistem. Halaman ini ditujukan untuk pengguna *IT Service.* Perancangan tampilan untuk halaman data pengguna ditunjukkan pada Gambar 4.50.



Gambar 4.50 Perancangan Antarmuka Halaman Data Pengguna

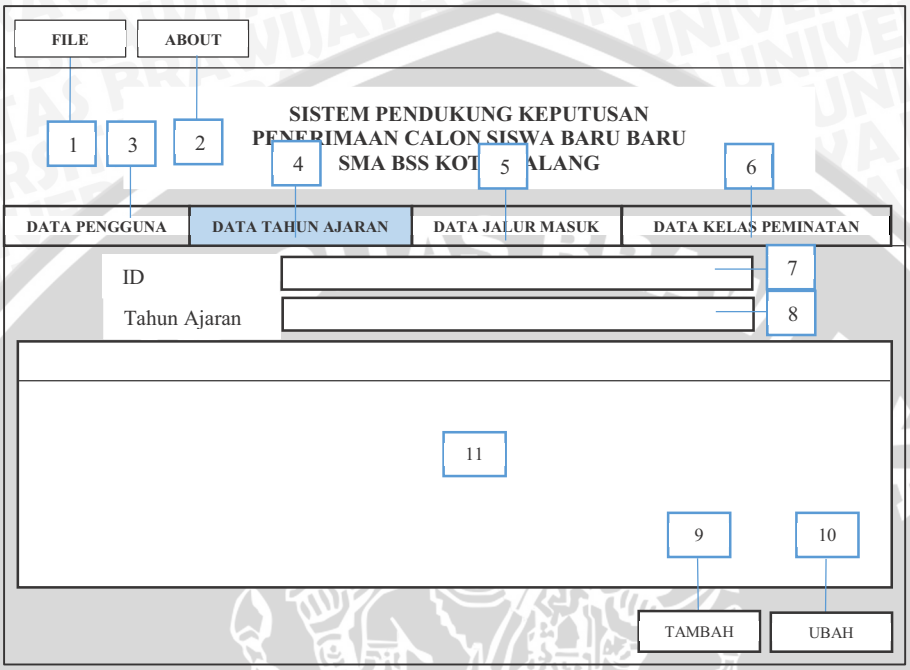
Keterangan Gambar 4.50:

1. *Taskbar File* (dengan *dropdown* menu *logout* dan *exit*)
2. *Taskbar About* (dengan *dropdown* menu *about*)
3. Tab Menu Data Pengguna
4. Tab Menu Data Tahun Ajaran
5. Tab Menu Data Jalur Masuk
6. Tab Menu Data Kelas Peminatan
7. Form untuk menampilkan ID pengguna
8. Form untuk menginputkan *username*
9. Form untuk menginputkan *password*
10. *Combobox* untuk menampilkan hak akses
11. Tombol untuk menambahkan data pengguna baru
12. Tombol untuk mengubah data pengguna yang dipilih
13. Tabel untuk menampilkan data pengguna

4.1.1.2. Halaman Data Tahun Ajaran

Setelah berhasil melakukan login dan memilih tab menu data tahun ajaran, maka pengguna akan masuk ke halaman data tahun ajaran yang merupakan halaman

yang digunakan untuk mengelola data tahun ajaran. Kelola data yang dapat dilakukan antara lain menambah, mengubah, menghapus dan melihat data tahun ajaran. Halaman ini ditujukan untuk pengguna *IT Service.* Perancangan tampilan untuk halaman data tahun ajaran ditunjukkan pada Gambar 4.51.



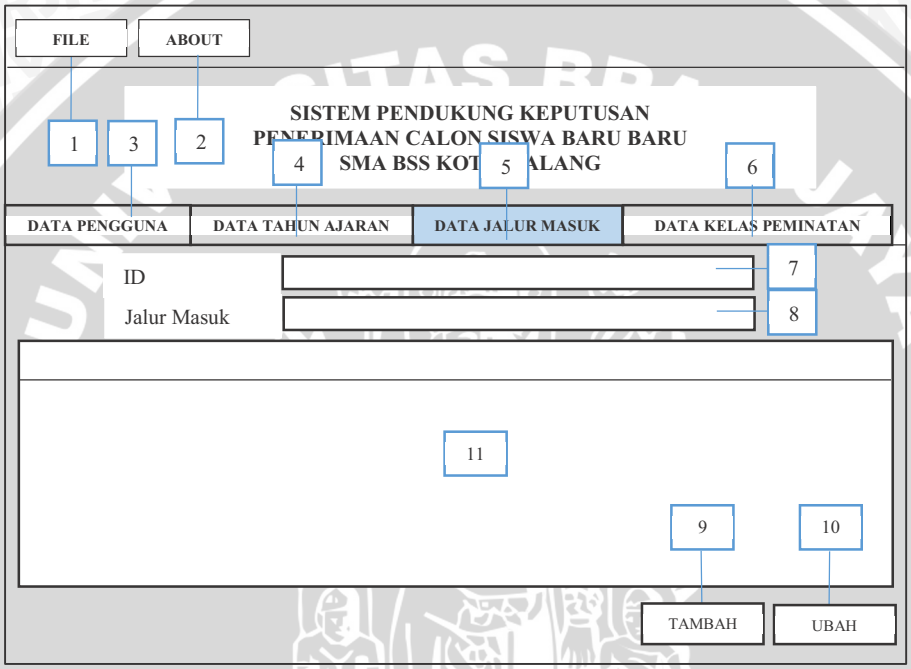
Gambar 4.51 Perancangan Antarmuka Halaman Data Tahun Ajaran

Keterangan Gambar 4.51:

1. *Taskbar File* (dengan *dropdown* menu *logout* dan *exit*)
2. *Taskbar About* (dengan *dropdown* menu *about*)
3. Tab Menu Data Pengguna
4. Tab Menu Data Tahun Ajaran
5. Tab Menu Data Jalur Masuk
6. Tab Menu Data Kelas Peminatan
7. Form untuk menampilkan ID tahun ajaran
8. Form untuk menginputkan tahun ajaran
9. Tombol untuk menambahkan data tahun ajaran
10. Tombol untuk mengubah data tahun ajaran yang dipilih
11. Tabel untuk menampilkan data tahun ajaran

4.1.1.3. Halaman Data Jalur Masuk

Setelah berhasil melakukan login dan memilih tab menu data jalur masuk, maka pengguna akan masuk ke halaman data jalur masuk yang merupakan halaman yang digunakan untuk mengelola data jalur masuk. Kelola data yang dapat dilakukan antara lain menambah, mengubah, menghapus dan melihat data jalur masuk. Halaman ini ditujukan untuk pengguna *IT Service*. Perancangan tampilan untuk halaman data jalur masuk ditunjukkan pada Gambar 4.52.



Gambar 4.52 Perancangan Antarmuka Halaman Data Jalur Masuk

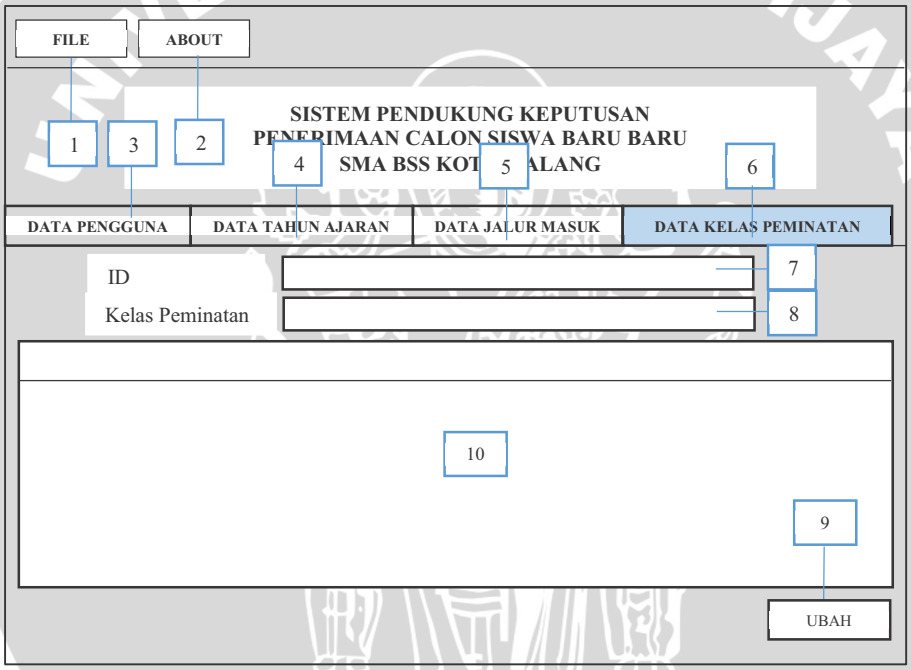
Keterangan Gambar 4.52:

1. *Taskbar File* (dengan *dropdown* menu *logout* dan *exit*)
2. *Taskbar About* (dengan *dropdown* menu *about*)
3. Tab Menu Data Pengguna
4. Tab Menu Data Tahun Ajaran
5. Tab Menu Data Jalur Masuk
6. Tab Menu Data Kelas Peminatan
7. Form untuk menampilkan ID jalur masuk
8. Form untuk menginputkan jalur masuk
9. Tombol untuk menambahkan data jalur masuk
10. Tombol untuk mengubah data jalur masuk yang dipilih

11. Tabel untuk menampilkan data jalur masuk

4.1.1.4. Halaman Data Kelas Peminatan

Setelah berhasil melakukan login dan memilih tab menu data kelas peminatan, maka pengguna akan masuk ke halaman data kelas peminatan yang merupakan halaman yang digunakan untuk mengelola data kelas peminatan. Kelola data yang dapat dilakukan antara lain menambah, mengubah, menghapus dan melihat data kelas peminatan. Halaman ini ditujukan untuk pengguna *IT Service.* Perancangan tampilan untuk halaman data kelas peminatan ditunjukkan pada Gambar 4.53.



Gambar 4.53 Perancangan Antarmuka Halaman Data Kelas Peminatan

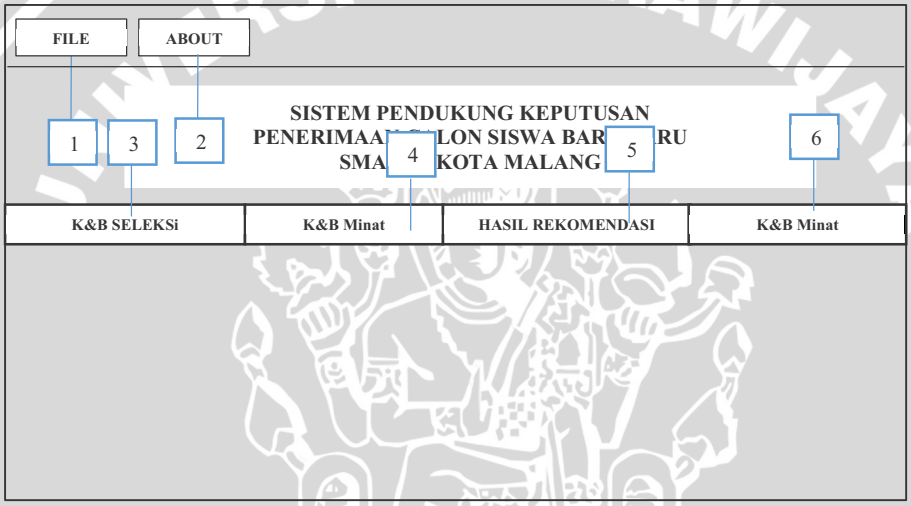
Keterangan Gambar 4.53:

1. *Taskbar File* (dengan *dropdown* menu *logout* dan *exit*)
2. *Taskbar About* (dengan *dropdown* menu *about*)
3. Tab Menu Data Pengguna
4. Tab Menu Data Tahun Ajaran
5. Tab Menu Data Jalur Masuk
6. Tab Menu Data Kelas Peminatan
7. Form untuk menampilkan ID kelas peminatan

8. Form untuk menginputkan kelas peminatan
9. Tombol untuk mengubah data kelas peminatan yang dipilih
10. Tabel untuk menampilkan kelas peminatan

4.1.1.5. Halaman Utama (Kepala Sekolah dan Ketua PPDB)

Setelah berhasil melakukan login, maka pengguna akan masuk ke halaman utama yang merupakan halaman utama dari aplikasi ini. Halaman ini ditujukan untuk pengguna kepala sekolah dan ketua ppdb. Pada halaman ini terdapat beberapa tab menu. Perancangan tampilan untuk halaman utama ditunjukkan pada Gambar 4.54.



Gambar 4.54 Perancangan Antarmuka Utama Kepala Sekolah dan Ketua PPDB

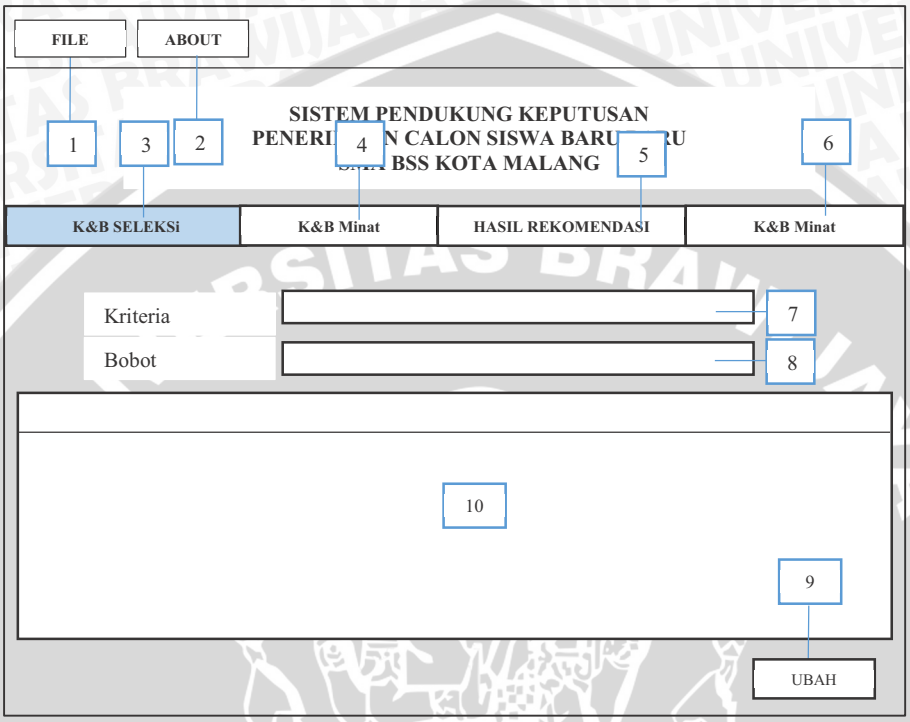
Keterangan Gambar 4.54:

1. *Taskbar File* (dengan *dropdown* menu *logout* dan *exit*)
2. *Taskbar About* (dengan *dropdown* menu *about*)
3. Tab Menu Kriteria dan Bobot Seleksi
4. Tab Menu Kriteria dan Bobot Peminatan
5. Tab Menu Hasil Rekomendasi
6. Tab Menu Hasil Peminatan

4.1.1.6. Halaman Data Kriteria dan Bobot Seleksi

Setelah berhasil melakukan login, dan memilih tab menu kriteria dan bobot seleksi, maka pengguna akan masuk ke halaman kriteria dan bobot seleksi yang merupakan halaman yang digunakan untuk mengelola data kriteria dan bobot seleksi. Halaman ini diggunakan untuk mengelola data kriteria dan bobot seleksi.

Kelola data yang dapat dilakukan antara lain mengubah dan melihat data kriteria dan bobot seleksi. Halaman ini ditujukan untuk pengguna Kepala Sekolah/Ketua PPDB. Perancangan tampilan untuk halaman data kriteria dan bobot ditunjukkan pada Gambar 4.55.



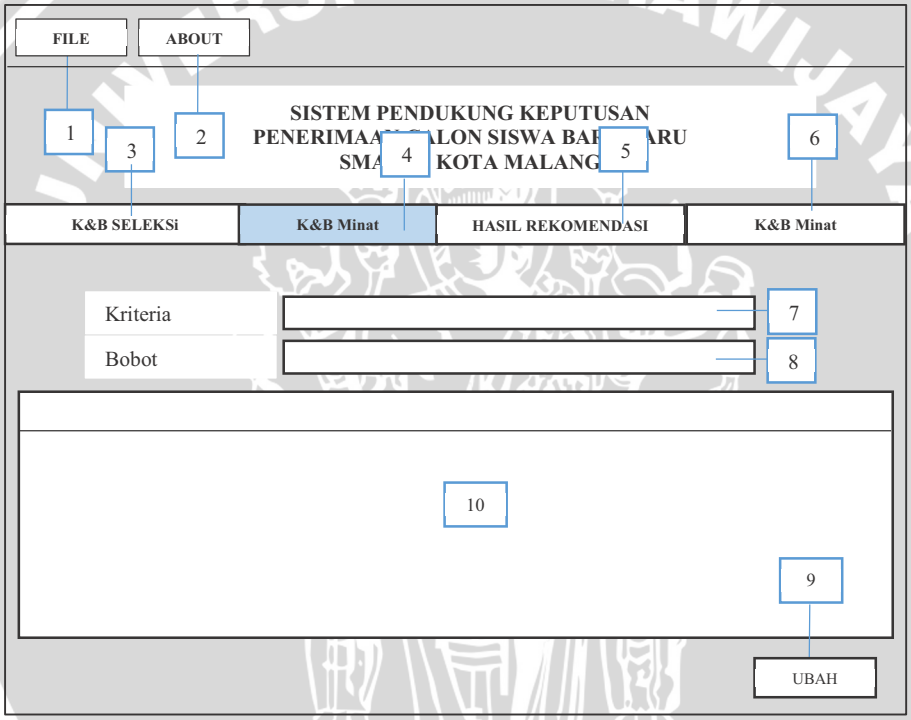
Gambar 4.55 Perancangan Antarmuka Halaman Kriteria dan Bobot Peminatan

Keterangan Gambar 4.55:

1. *Taskbar File* (dengan *dropdown* menu *logout* dan *exit*)
2. *Taskbar About* (dengan *dropdown* menu *about*)
3. Tab Menu Kriteria dan Bobot Seleksi
4. Tab Menu Kriteria dan Bobot Peminatan
5. Tab Menu Hasil Rekomendasi
6. Form untuk menampilkan nama kriteria dan bobot seleksi
7. Form untuk menginputkan bobot
8. Tombol untuk mengubah data kriteria dan bobot yang dipilih
9. Tabel untuk menampilkan kriteria dan bobot
10. Menampilkan data bobot dan kriteria

4.1.1.7. Halaman Data Kriteria dan Bobot Peminatan

Setelah berhasil melakukan login, dan memilih tab menu kriteria dan bobot seleksi, maka pengguna akan masuk ke halaman kriteria dan bobot seleksi yang merupakan halaman yang digunakan untuk mengelola data kriteria dan bobot seleksi. Halaman ini diggunakan untuk mengelola data kriteria dan bobot seleksi. Kelola data yang dapat dilakukan antara lain mengubah dan melihat data kriteria dan bobot peminatan. Halaman ini ditujukan untuk pengguna Kepala Sekolah/Ketua PPDB. Perancangan tampilan untuk halaman data kriteria dan bobot ditunjukkan pada Gambar 4.56.



Gambar 4.56 Perancangan Antarmuka Halaman Kriteria dan Bobot Peminatan

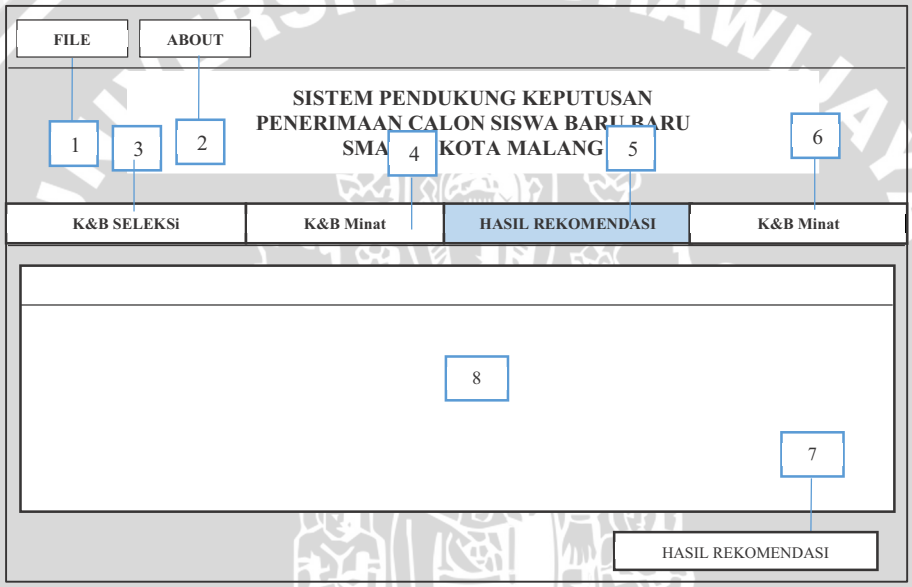
Keterangan Gambar 4.56:

1. *Taskbar File* (dengan *dropdown* menu *logout* dan *exit*)
2. *Taskbar About* (dengan *dropdown* menu *about*)
3. Tab Menu Kriteria dan Bobot Seleksi
4. Tab Menu Kriteria dan Bobot Peminatan
5. Tab Menu Hasil Rekomendasi
6. Tab Menu Hasil Peminatan
7. Form untuk menampilkan Inama kriteria dan bobot seleksi

8. Form untuk menginputkan bobot
9. Tombol untuk mengubah data kriteria dan bobot yang dipilih
10. Tabel untuk menampilkan kriteria dan bobot

4.1.1.8. Halaman Data Rekomendasi

Halaman ini merupakan halaman yang ditampilkan pada saat kepala sekolah/ketua panitia telah memilih menu Data Rekomendasi. Dalam halaman ini sie tes dapat melihat data hasil rekomendasi dan melakukan hapus data hasil rekomendasi. Perancangan tampilan dari halaman Data Rekomendasi ditunjukkan pada Gambar 4.57.



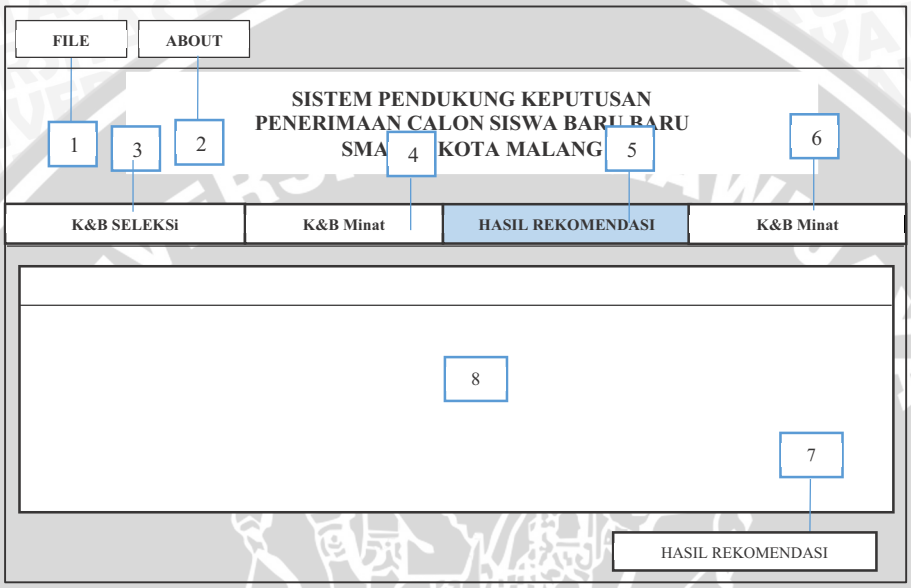
Gambar 4.57 Perancangan Antarmuka Halaman Data Rekomendasi

Keterangan Gambar 4.57:

1. *Taskbar File* (dengan *dropdown* menu *logout* dan *exit*)
2. *Taskbar About* (dengan *dropdown* menu *about*)
3. Tab Menu Kriteria dan Bobot Seleksi
4. Tab Menu Kriteria dan Bobot Peminatan
5. Tab Menu Hasil Rekomendasi
6. Tab Menu Hasil Peminatan
7. Tombol untuk menampilkan hasil rekomendasi
8. Tabel untuk menampilkan data rekomendasi

4.1.1.9. Halaman Data Rekomendasi

Halaman ini merupakan halaman yang ditampilkan pada saat kepala sekolah/ketua panitia telah memilih menu Data Rekomendasi. Dalam halaman ini sie tes dapat melihat data hasil rekomendasi dan melakukan hapus data hasil rekomendasi. Perancangan tampilan dari halaman Data Rekomendasi ditunjukkan pada Gambar 4.58.



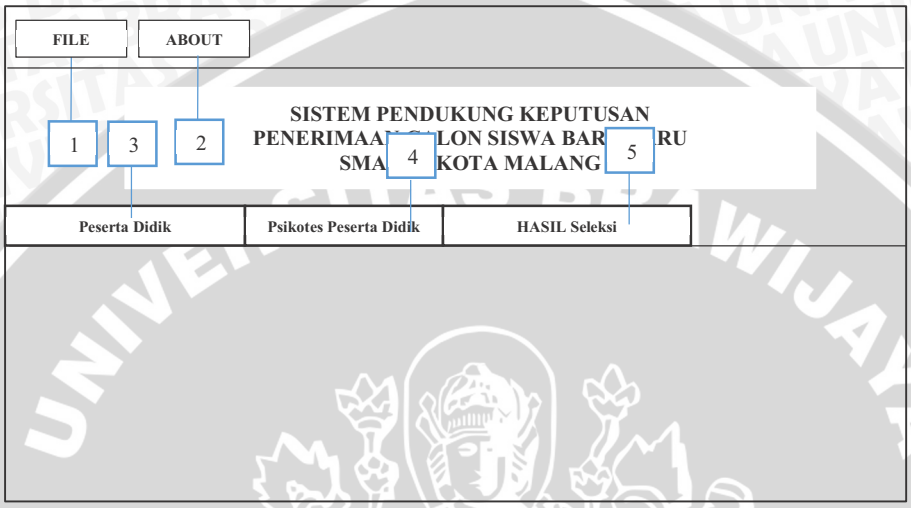
Gambar 4.58 Perancangan Antarmuka Halaman Data Rekomendasi

Keterangan Gambar 4.58:

1. *Taskbar File* (dengan *dropdown* menu *logout* dan *exit*)
2. *Taskbar About* (dengan *dropdown* menu *about*)
3. Tab Menu Kriteria dan Bobot Seleksi
4. Tab Menu Kriteria dan Bobot Peminatan
5. Tab Menu Hasil Rekomendasi
6. Tombol bantuan untuk menampilkan bantuan dalam menjalankan sistem
7. Tombol untuk menampilkan hasil rekomendasi
8. Tabel untuk menampilkan data rekomendasi

4.1.1.10. Halaman Utama (Sie Test dan Guru BK)

Setelah berhasil melakukan login, maka pengguna akan masuk ke halaman utama yang merupakan halaman utama dari aplikasi ini. Halaman ini ditujukan untuk pengguna sie test dan guru BK. Pada halaman ini terdapat beberapa tab menu. Perancangan tampilan untuk halaman utama ditunjukkan pada Gambar 4.59.



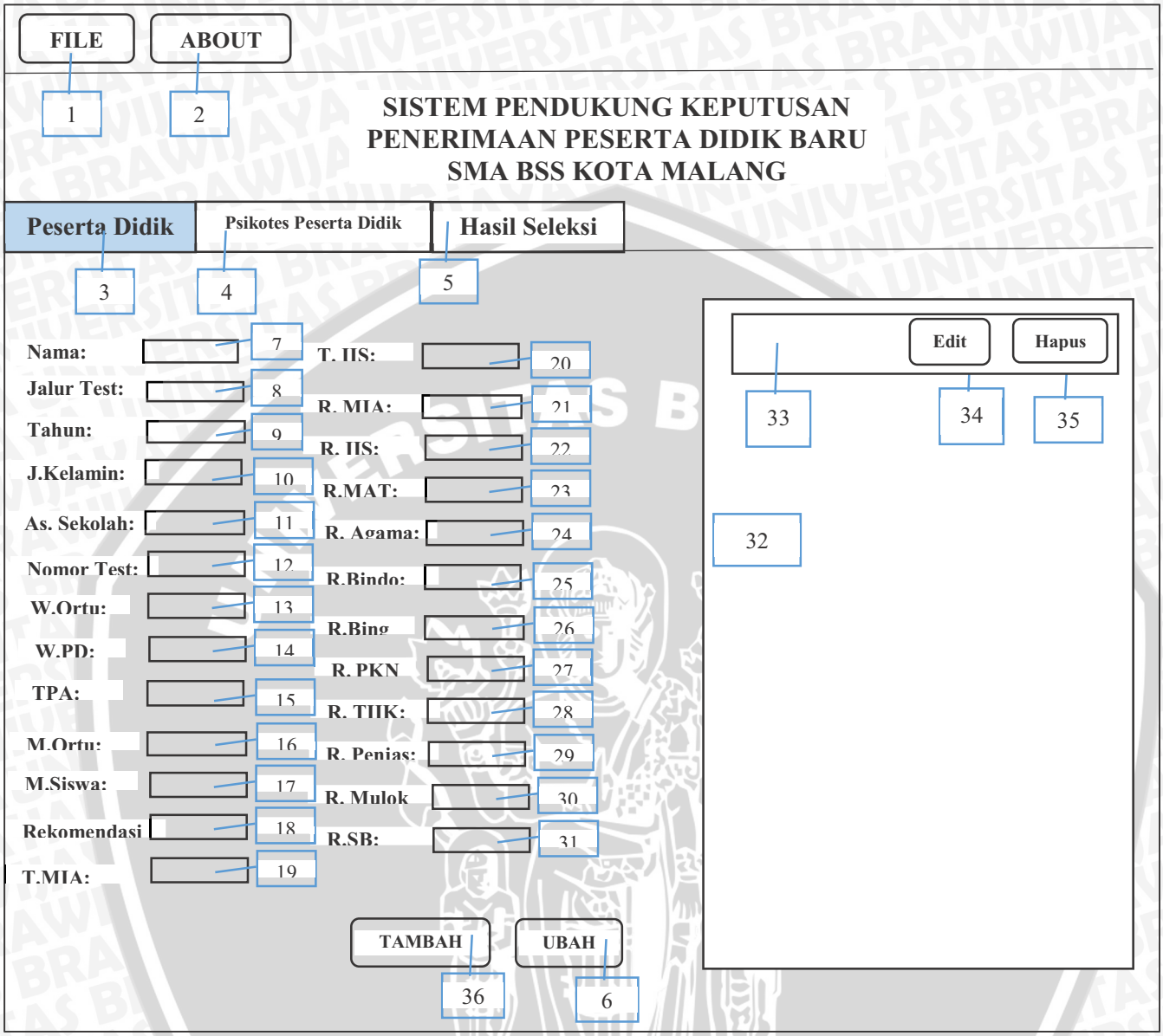
Gambar 4.59 Perancangan Antarmuka Utama Sie Test dan Guru BK

Keterangan Gambar 4.59:

1. *Taskbar File* (dengan *dropdown* menu *logout* dan *exit*)
2. *Taskbar About* (dengan *dropdown* menu *about*)
3. Tab Menu Peserta Didik
4. Tab Menu Psikotes Peserta Didik
5. Tab Menu Hasil Seleksi

4.1.1.11. Halaman Siswa (Sie Test dan Guru BK)

Halaman ini adalah halaman tab siswa yang berfungsi untuk melihat, menambah, menghapus, dan mengubah data siswa. Perancangan tampilan untuk halaman utama ditunjukkan pada Gambar 4.60.



Gambar 4.60 Perancangan Antarmuka Siswa Sie Test dan Guru BK

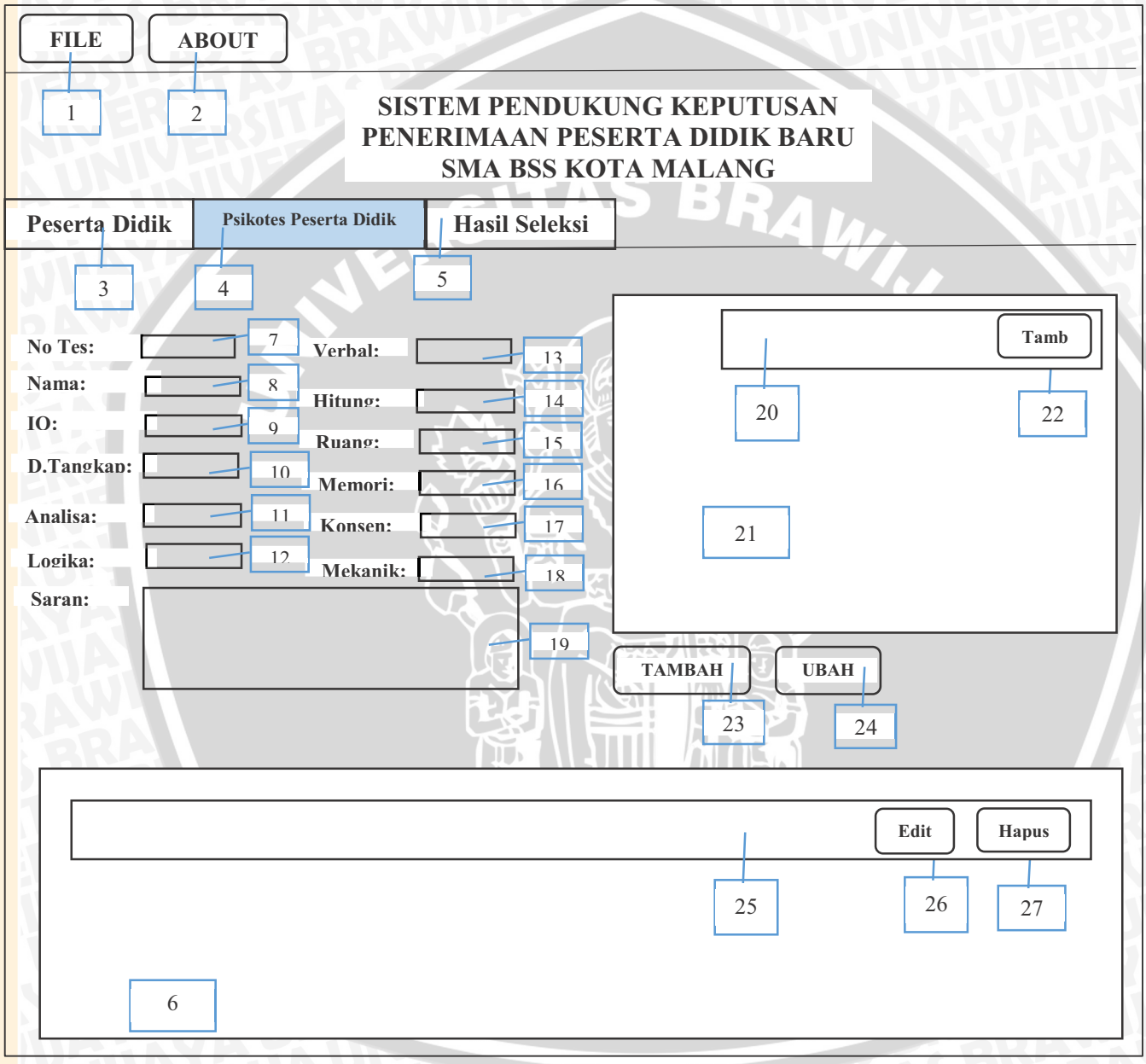
Keterangan Gambar 4.60:

1. *Taskbar File* (dengan *dropdown* menu *logout* dan *exit*)
2. *Taskbar About* (dengan *dropdown* menu *about*)
3. Tab Menu Peserta Didik
4. Tab Menu Psikotes Peserta Didik
5. Tab Menu Hasil Seleksi
6. *Field* untuk mengisi nama siswa
7. Tombol untuk menyimpan perubahan data siswa
8. *Field* untuk mengisi jalur test siswa

9. *Field* untuk mengisi tahun seleksi siswa
10. *Field* untuk mengisi jenis kelamin siswa
11. *Field* untuk mengisi asal sekolah siswa
12. *Field* untuk mengisi nomor test siswa
13. *Field* untuk mengisi nilai wawancara orang tua siswa
14. *Field* untuk mengisi nilai wawancara siswa
15. *Field* untuk mengisi nilai tes potensi akademik siswa
16. *Field* untuk mengisi minat orang tua siswa
17. *Field* untuk mengisi minat siswa
18. *Field* untuk mengisi rekomendasi peminatan SMP BK siswa
19. *Field* untuk mengisi nilai tes peminatan IPA siswa
20. *Field* untuk mengisi nilai tes peminatan IPS siswa
21. *Field* untuk mengisi nilai rata-rata rapor IPA siswa
22. *Field* untuk mengisi nilai rata-rata rapor IPS siswa
23. *Field* untuk mengisi nilai rata-rata rapor MAT siswa
24. *Field* untuk mengisi nilai rata-rata rapor Agama siswa
25. *Field* untuk mengisi nilai rata-rata rapor Bahasa Indonesia siswa
26. *Field* untuk mengisi nilai rata-rata rapor Bahasa Inggris siswa
27. *Field* untuk mengisi nilai rata-rata rapor PKN siswa
28. *Field* untuk mengisi nilai rata-rata rapor TIK siswa
29. *Field* untuk mengisi nilai rata-rata rapor Penjaskes siswa
30. *Field* untuk mengisi nilai rata-rata rapor Mulok siswa
31. *Field* untuk mengisi nilai rata-rata rapor Seni Budaya siswa
32. Tabel untuk menampilkan seluruh data siswa
33. *Field* untuk menampilkan data siswa
34. Tombol untuk melakukan perubahan data siswa
35. Tombol untuk menghapus data siswa
36. Tombol untuk tambah siswa dan menyimpannya ke dalam *database*

4.1.1.12. Halaman Psikotes Siswa (Sie Test dan Guru BK)

Halaman ini adalah halaman tab siswa yang berfungsi untuk melihat, menambah, menghapus, dan mengubah data psikotes siswa. Perancangan tampilan untuk halaman utama ditunjukkan pada Gambar 4.61.



Gambar 4.61 Perancangan Antarmuka Psikotes Siswa Sie Test dan Guru BK

Keterangan Gambar 4.61:

1. *Taskbar File* (dengan *dropdown* menu *logout* dan *exit*)
2. *Taskbar About* (dengan *dropdown* menu *about*)

3. Tab Menu Peserta Didik
4. Tab Menu Psikotes Peserta Didik
5. Tab Menu Hasil Seleksi
6. Tabel untuk menampilkan seluruh data psikotes siswa
7. *Field* untuk menampilkan nomor siswa
8. *Field* untuk mengisi nama siswa
9. *Field* untuk mengisi IQ siswa
10. *Field* untuk mengisi kemampuan daya tangkap siswa
11. *Field* untuk mengisi kemampuan analisa siswa
12. *Field* untuk mengisi kemampuan logika siswa
13. *Field* untuk mengisi kemampuan verbal siswa
14. *Field* untuk mengisi kemampuan hitung siswa
15. *Field* untuk mengisi kemampuan ruang siswa
16. *Field* untuk mengisi kemampuan memori siswa
17. *Field* untuk mengisi kemampuan konsestrasi siswa
18. *Field* untuk mengisi kemampuan mekanik siswa
19. *Field* untuk mengisi saran siswa
20. Tabel untuk menampilkan data siswa yang hendak ditambah nilai psikotesnya
21. Tabel untuk menampilkan seluruh data siswa
22. Tombol untuk memilih siswa yang hendak ditambah nilai psikotes
23. Tombol untuk menyimpan nilai psikotes ke database
24. Tombol untuk menyimpan perubahan nilai psikotes ke database
25. Tabel untuk menampilkan data psikotes siswa
26. Tombol untuk mengubah data psikotes siswa
27. Tombol untuk menghapus data psikotes siswa

4.1.1.13. Halaman seleksi Siswa (Sie Test dan Guru BK)

Halaman ini adalah halaman tab untuk melakukan proses seleksi dan melihat hasil seleksi. Perancangan tampilan untuk halaman utama ditunjukkan pada Gambar 4.62.



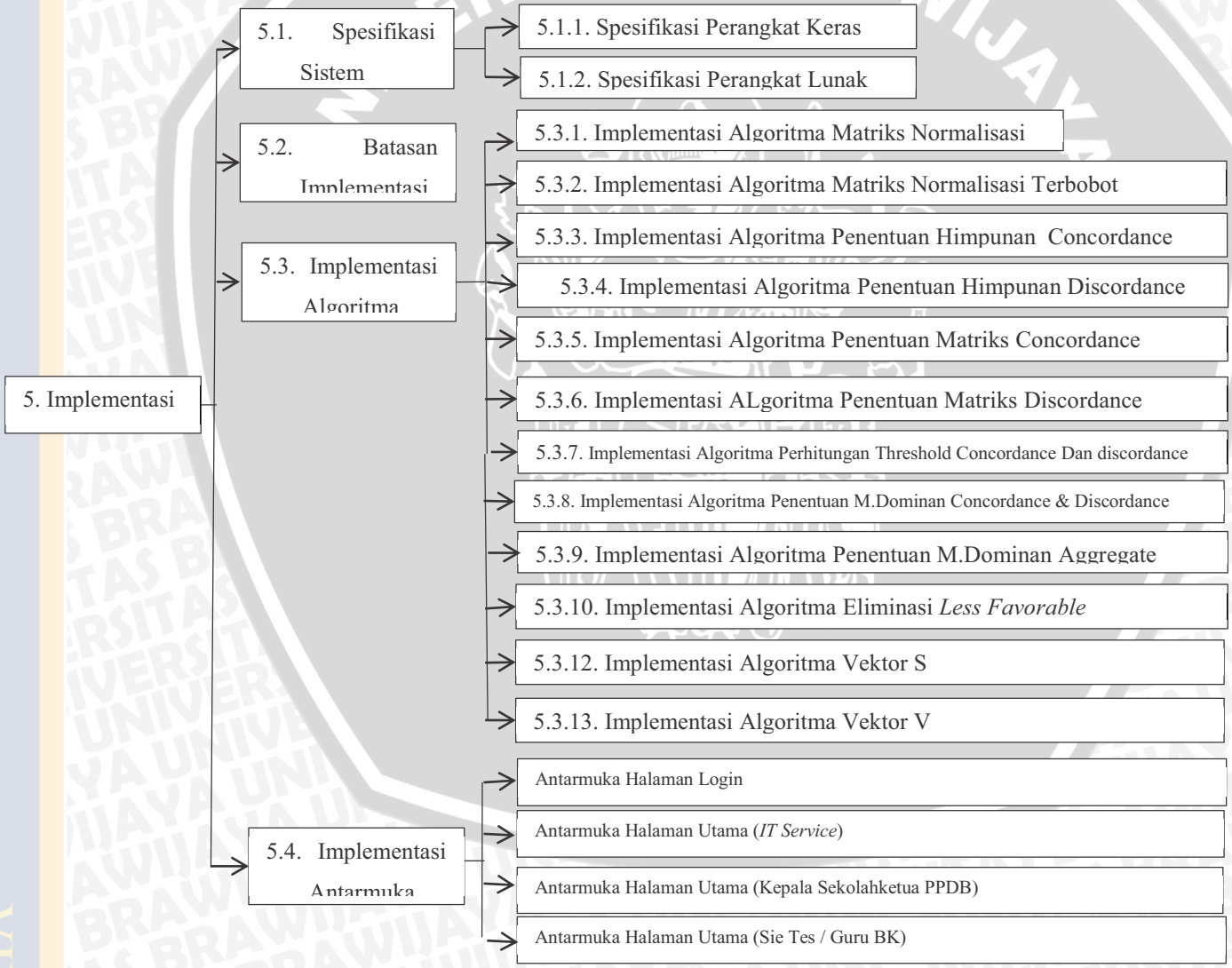
Gambar 4.62 Perancangan Antarmuka Hasil Seleksi Sie Test dan Guru BK

Keterangan Gambar 4.62.

1. *Taskbar File* (dengan *dropdown* menu *logout* dan *exit*)
2. *Taskbar About* (dengan *dropdown* menu *about*)
3. Tab Menu Peserta Didik
4. Tab Menu Psikotes Peserta Didik
5. Tab Menu Hasil Seleksi
6. Tabel untuk menampilkan hasil seleksi siswa
7. *Field* untuk memilih tahun test
8. *Field* untuk memilih jalur test
9. *Field* untuk mengisi kuota (jika ada)
10. Tombol untuk melakukan perhitungan
11. Tombol untuk melihat hasil perhitungan
12. Tabel untuk menampilkan hasil perhitungan

BAB V
IMPLEMENTASI

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



Gambar 5.1 Pohon Implemantasi

5.1 Spesifikasi Sistem

Spesifikasi sistem sistem yang akan dijelaskan pada sub bab ini adalah spesifikasi perangkat lunak dan perangkat keras yang digunakan untuk implementasi sistem agar sistem dapat berfungsi sesuai dengan kebutuhan.

5.1.1 Spesifikasi Perangkat Keras

Pengembangan sistem pendukung keputusan penerimaan calon siswa baru baru SMA Brawijaya *Smart School* (SMA BSS) ini menggunakan seperangkat komputer dengan spesifikasi perangkat keras yang ditunjukkan dalam Tabel 5.1

Tabel 5.1Spesifikasi Perangkat Keras

Nama Komponen	Spesifikasi
Prosesor	Intel ® Core™ i5-2410M CPU @ 2.30GHz
Memori	4096 MB
Harddisk	500 GB

Sumber: Implementasi

5.1.2 Spesifikasi Perangkat Lunak

Pengembangan sistem pendukung keputusan penerimaan calon siswa baru baru SMA Brawijaya *Smart School* (SMA BSS) ini menggunakan seperangkat komputer dengan spesifikasi perangkat lunak yang ditunjukkan dalam Tabel 5.2

Tabel 5.2 Spesifikasi Perangkat Lunak

Nama	Spesifikasi
Sistem Operasi	Microsoft Windows 7 Home Edition
Bahasa Pemrograman	C#
Tools Pemrograman	Microsoft Visual Studio 2010
DBMS	MySQL
Tools DBMS	

Sumber: Implementasi

5.2 Batasan Implementasi

Batasan implementasi merupakan batasan implementasi yang bisa dilakukan oleh sistem sesuai dengan perancangan awal sistem. Batasan implementasi

dijelaskan secara rinci agar penelitian ini memiliki ruang lingkup yang jelas. Beberapa batasan yang digunakan dalam mengimplementasikan sistem pendukung keputusan penerimaan calon siswa baru baru SMA Brawijaya *Smart School* (SMA BSS) adalah sebagai berikut:

a. Sistem dibangun berdasarkan ruang lingkup *desktop application* dengan menggunakan bahasa pemrograman C#.
b. Data-data yang digunakan dalam sistem disimpan dalam *Database Management System* (DBMS) MySQL.
c. Metode yang digunakan dalam menyelesaikan masalah adalah Metode ELECTRE yang digunakan untuk melakukan seleksi calon siswa baru dan *Weighted Product* yang digunakan untuk menentukan kelas peminatan calon siswa baru yang telah diseleksi.
d. Sistem pendukung keputusan ini dibuat untuk melakukan seleksi terhadap calon calon siswa baru jalur tes
e. Inputan dari pengguna yang diolah dalam sistem pendukung ini adalah data calon siswa baru dan nilai bobot.
f. Kriteria yang digunakan adalah empat kriteria untuk metode ELECTRE dan 21 kriteria untuk metode *Weighted Product*, seperti yang dijelaskan dalam Bab 2.
g. Nilai bobot dalam proses ELECTRE - *Weighted Product* hanya dapat diubah oleh pengguna sistem dengan jabatan sebagai Kepala Sekolah dan Ketua PPDB.
h. Pengelolaan data pengguna sistem hanya berhak dilakukan oleh pengguna sistem dengan jabatan sebagai Staff IT.
i. Pengelolaan data calon calon siswa baru dalam sistem hanya berhak dilakukan oleh pengguna sistem dengan jabatan sebagai Sie Test dan Guru BK.
j. Output yang diterima oleh pengguna dari sistem pendukung keputusan ini adalah hasil rekomendasi calon siswa baru baik yang diterima maupun yang tidak diterima, serta penempatan kelas peminatan bagi siswa yang diterima.
k. Sistem pendukung keputusan ini hanya memberikan rekomendasi sesuai dengan hasil perhitungan terhadap data. Keputusan akhir dari diterima atau

tidak diterima calon calon siswa baru diserahkan sepenuhnya kepada Kepala Sekolah dan Sie Test.

l. Setiap pengguna sistem yang akan menggunakan sistem diwajibkan untuk login terlebih dahulu.

m. Sistem pendukung keputusan ini hanya memberikan rekomendasi sesuai dengan hasil perhitungan terhadap data. Keputusan akhir dari diterima atau tidak diterima calon peserta didik diserahkan sepenuhnya kepada pihak sekolah SMA BSS.

n. Setiap pengguna sistem yang akan menggunakan sistem diharuskan untuk login terlebih dahulu.

5.3 Implementasi Algoritma

Sistem pendukung keputusan seleksi penerimaan peserta didik baru dengan menggunakan metode ELECTRE - *Weighted Product*, memiliki proses utama perhitungan antara lain:

- Untuk perhitungan ELECTRE: penentuan matriks normalisasi, penentuan matriks normalisasi terbobot, penentuan himpunan *concordance* dan *discordance,* penentuan matriks *concordance* dan *discordance*, penentuan matriks dominan *concordance* dan *discordance*, penentuan *aggregate dominance matrix* dan proses eliminasi alternatif yang *less favourable*.
- Untuk perhitungan *Weighted Product*: penentuan nilai vektor S dan penentuan nilai vektor V.

Implementasi algoritma ini mengacu pada bab perancangan dengan subbab 4.2.3.1 dan subbab 4.2.3.2. Untuk rincian lebih lanjut dapat dilihat melalui paparan berikut ini.

5.3.1 ELECTRE

5.3.1.1 Menghitung Matriks Normalisasi

Proses normalisasi ELECTRE dilakukan dengan cara menjumlahkan nilai kuadrat seluruh data alternatif (baris) masing-masing kriteria (kolom) lalu jumlah tersebut diakar. Setelah diakar, nilai masing-masing data alternatif dibagi dengan hasil akar sesuai dengan kriteria-nya. Implementasi algoritma menghitung matriks

normalisasi berdasarkan rancangan pada Gambar 4.18 Ditunjukkan pada gambar 5.2.

```
1   M_keputusan1 = new double[m, n];
2   x = new double[n];
3   sum = new double[n];
4   for (int b = 0; b < n; b++)
5   {
      M_keputusan1[data, b] =
6   double.Parse(dataReader1.GetString(b).ToString());
7   }
8   for (int a = 0; a < n; a++)
9   {
10    for (int b = 0; b < m; b++)
11    {
12      sum[a] += Math.Pow(M_keputusan1[b, a], 2);
13    }
14      x[a] = Math.Sqrt(sum[a]);
15  }
16  r = new double[m, n];
17  for (int a = 0; a < m; a++)
18  {
19   for (int b = 0; b < n; b++)
20    {
21      r[a, b] = M_keputusan1[a, b] / x[b];
22    }
23  }
```

Gambar 5.2 Implementasi Algoritma Matriks Normalisasi
Sumber: [Implementasi]

Penjelasan algoritma perhitungan matriks normalisasi pada Gambar 5.2 Yaitu:

1. Deklarasi variable untuk menyimpan matriks keputusan ditunjukkan pada baris 1
2. Deklarasi variable untuk menyimpan nilai akar jumlah nilai kuadrat setiap kolom kriteria pada matriks keputusan ditunjukkan pada baris 2
3. Deklarasi variable untuk menyimpan jumlah nilai kuadrat setiap kolom kriteria pada matriks keputusan ditunjukkan pada baris 3
4. Proses membentuk matriks keputusan ditunjukkan pada baris 4-7
5. Proses menghitung jumlah nilai kuadrat setiap kolom kriteria pada matriks keputusan dan hasil tersebut diakar ditunjukkan pada baris 8-12. Terdapat

proses perulangan sebanyak n dan didalamnya terdapat proses perulangan sebanyak m.

6. Proses menghitung nilai akar setiap elemen sum[n] dan disimpan pada variable x
7. Deklarasi variable untuk menyimpan matriks normalisasi ditunjukkan pada baris 11.
8. Proses hitung pembagian antara nilai matriks keputusan dengan nilai x[n] ditunjukkan pada baris 17-23. Terdapat proses perulangan sebanyak n dan didalamnya terdapat proses perulangan sebanyak m

5.3.1.2 Menghitung Matriks Normalisasi Terbobot

Proses normalisasi terbobot ELECTRE dilakukan dengan mengkalikan matriks normalisasi dengan bobotnya masing-masing. Implementasi algoritma menghitung matriks normalisasi berdasarkan rancangan pada Gambar 4.20 Ditunjukkan pada gambar 5.3.

```
1  v = new double[m, n];
2  for (int a = 0; a < m; a++)
3  {
4   for (int b = 0; b < n; b++)
5   {
6     v[a, b] = r[a, b] * w[b];
7   }
8  }
```

Gambar 5.3 Implenetasi Algoritma Matriks Normalisasi Terbobot

Penjelasan algoritma perhitungan matriks normalisasi terbobot pada Gambar 5.3 Yaitu:

1. Deklarasi variable untuk menyimpan nilai matriks normalisasi terbobot ditunjukkan pada baris 1.
2. Proses perhitungan matrik normalisasi terbobot v[m][n] dengan mengkalikan matriks normalisasi dengan bobotnya masing-masing. Terdapat proses perulangan sebanyak n dan didalamnya terdapat proses perulangan sebanyak m

5.3.1.3 Penentuan Himpunan *Concordance* ELECTRE

Proses penentuan himpunan *concordance* ELECTRE dilakukan dengan cara membandingkan satu alternatif dengan alternatif lainnya pada setiap krtiteria. Implementasi algoritma menghitung matriks normalisasi berdasarkan rancangan pada Gambar 4.22 Ditunjukkan pada gambar 5.4.

```
1  C = new int[m, m, n];
2  for (int a = 0; a < m; a++)
3  {
4   for (int b = 0; b < m; b++)
5   {
6    for (int j = 0; j < n; j++)
7    {
8     if (a != b)
9     {
       if (v[a, j] > v[b, j] || v[a, j] == v[b, j])
10      { C[a, b, j]  = j;}
11      else if (v[a, j] < v[b, j])
12      {
13         C[a, b, j] = 99;
14      }
15     }
16    }
17   }
19 }
```

Gambar 5.4 Implentasi Agoritma penentuan himpunan *concordance* ELECTRE

Penjelasan algoritma penentuan himpunan *concordance* pada Gambar 5.4 yaitu:

1. Deklarasi variable untuk menyimpan himpunan *concordance* ditunjukkan pada baris 1.
2. Proses penentuan himpunan *concordance* ditunjukkan pada baris 2-19. Setiap alternatif ke-a akan dibandingkan dengan alternatif ke-b pada setiap kriteria ke-j selama a≠b. Proses perbandingan ditunjukkan pada baris 2-8.
3. Kedua alternatif dibandingkan dengan ketentuan jika v baris ke-a ≥ v baris ke-b pada kolom ke j, maka kriteria ke-j masuk ke dalam himpunan *concordance* dengan nilai=j. Jika v baris ke-a ≤ v baris ke-b pada kolom ke

j, maka himpunan *concordance* pada kriteria tersebut akan bernilai 99. Proses ditunjukkan pada baris 8-15.

4. Nilai hasil banding setiap kolom akan disimpan pada C[a][b][j]. Proses ditunjukkan pada baris 10 dan baris 13.

5.3.1.4 Penentuan Himpunan *Discordance* ELECTRE

Proses penentuan himpunan *discordance* ELECTRE dilakukan dengan cara membandingkan satu alternatif dengan alternatif lainnya pada setiap krtiteria. Implementasi algoritma menghitung matriks normalisasi berdasarkan rancangan pada Gambar 4.24 Ditunjukkan pada gambar 5.5.

```
1  D = new int[m, m, n];
2  for (int a = 0; a < m; a++)
3  {
4   for (int b = 0; b < m; b++)
5   {
6    for (int j = 0; j < n; j++)
7    {
8       if (a != b)
9       {
        if (v[a, j] > v[b, j] || v[a, j] == v[b, j])
10      { D[a, b, j] = 99; }
11      else if (v[a, j] < v[b, j])
12      {
13        D[a, b, j] = j;
14      }
15      }
16     }
17   }
19 }
```

Gambar 5.5 Implementasi Algoritma Himpunan *Discordance*

Penjelasan algoritma himpunan *discordance* pada Gambar 5.5 Yaitu:

1. Deklarasi variable untuk menyimpan himpunan *discordance* ditunjukkan pada baris 1.
2. Proes penentuan himpunan *discordance* ditunjukkan pada baris 2-19. Setiap alternatif ke-a akan dibandingkan dengan alternatif ke-b pada setiap kriteria ke-j selama a≠b. Proses perbandingan ditunjukkan pada baris 2-8.

3. Kedua alternatif dibandingkan dengan ketentuan jika v baris ke-a < v baris ke-b pada kolom ke j, maka kriteria ke-j masuk ke dalam himpunan *discordance* dengan nilai=j. Jika v baris ke-a ≥ v baris ke-b pada kolom ke j, maka himpunan *discordance* pada kriteria tersebut akan bernilai 99(0). Proses ditunjukkan pada baris 9-14.
4. Nilai hasil banding setiap kolom akan disimpan pada D[a][b][j]. Proses ditunjukkan pada baris 10 dan baris 13.

5.3.1.5 Menentukan Matriks *Concordance*

Proses menentukan matriks *concordance* ELECTRE dilakukan dengan cara menjumlahkan nilai bobot kriteria. Implementasi algoritma menghitung matriks normalisasi berdasarkan rancangan pada Gambar 4.26 Ditunjukkan pada gambar 5.6.

```
1  matrix_C = new double[m, m];
2  for (int a = 0; a < m; a++)
3  {
4   for (int b = 0; b < m; b++)
5   {
6   for (int j = 0; j < n; j++)
7   {
8   if (a != b)
9   {
    if (C[a, b, j] != 99) { int index = C[a, b, j]; matrix_C[a, b] +=
10 w[index]; }
11  }
12 }
13 }
14 }
```

Gambar 5.6 Implementasi Algoritma Matriks *Concordance*

Penjelasan algoritma matriks *concordance* pada Gambar 5.6 Yaitu:

1. Deklarasi variable untuk menyimpan himpunan matriks *concordance* ditunjukkan pada baris 1.
2. Baris 9-8 merupakan deklarasi proses penentuan matriks *concordance*. Nilai bobot kriteria dijumlahkan, dimana kriteria yang masuk ke dalam himpunan *concordance* memiliki ketentuan a≠b dan nilai element C[a][b][j] ≠99. Jika nilai a=b, maka akan menuju ke nilai b selanjutnya.

3. Melakukan proses penjumlahan dengan cara:

- Jika nilai himpunan pada matriks C[a][b][j] = 1, maka nilai bobot kriteria yang dijumlahkan adalah nilai bobot kriteria 1
- Jika nilai himpunan pada matriks C[a][b][j] = 2, maka nilai bobot kriteria yang dijumlahkan adalah nilai bobot kriteria 2
- Dst. Proses ini ditunjukkan pada baris 9-10

4. Hasil penjumlahan bobot kriteria yang termasuk pada himpunan *concordance* C[a][b][j] akan disimpan pada matrix_C[a,b]

5.3.1.6 Menentukan Matriks *Discordance*

Proses menentukan matriks *discordance* ELECTRE dilakukan dengan cara menjumlahkan nilai bobot kriteria. Implementasi algoritma menghitung matriks normalisasi berdasarkan rancangan pada Gambar 4.28 Ditunjukkan pada gambar 5.7.

```
 1   matrix_D = new double[m, m];
 2   selisih1 = new double[n];
 3  selisih2 = new double[n];
 4   max1 = new double[m, m];
 5   max2 = new double[m, m];
 6   for (int a = 0; a < m; a++)
 7   {
 8    for (int b = 0; b < m; b++)
 9    {
10         for (int j = 0; j < n; j++)
11         {
12           selisih1[j] = 0; selisih2[j] = 0;
13            if (a != b)
14            {
15             if (D[a, b, j] != 99) { selisih1[j] = Math.Abs(v[a, j]
  - v[b, j]); }
16             selisih2[j] = Math.Abs(v[a, j] - v[b, j]);
17              }
18           }
19           if (a != b)
20           {
21               max1[a, b] = nilaiMax(n, selisih1);
22               max2[a, b] = nilaiMax(n, selisih2);
23               matrix_D[a, b] = max1[a, b] / max2[a, b];
24           }
```

```
25          }
26  }
```

Gambar 5.7 Implementasi Algoritma Matriks *Discordance*

penjelasan algoritma algoritma matriks *discordance* pada Gambar 5.7 Yaitu:

1. Deklrasi variable untuk menyimpan himpunan matriks *discordance* ditunjukkan pada baris 1.
2. Deklarasi variable untuk menyimpan selisih antara alternatif a dan alternatif b pada kriteria j yang masuk ke dalam anggota himpunan *discordance* ditunjukkan pada baris 2.
3. Deklarasi variable untuk menyimpan selisih antara alternatif a dan alternatif b pada kriteria j dari seluruh kriteria ditunjukkan pada baris 3.
4. Deklarasi variable untuk menyimpan nilai maksimal dari nilai selisih1 sebanyak data.
5. Baris 5 merupakan deklarasi variable untuk menyimpan nilai maksimal dari nilai selisih 2 sebanyak data ditunjukkan pada baris 4.
6. Deklarasi nilai selisih1 pada kriteria-j dan nilai selisih2 pada kriteria-j adalah 0 ditunjukkan pada baris 6
7. Proses perhitungan selisih1 ditunjukkan pada baris 14. Jika nilai a≠b, maka akan dilakukan pengecekan pada D[a, b, j]. Jika nilai D[a, b, j] ≠ 99, maka nilai selisih1 adalah absolut v[a, j] – v[b, j].
8. Proses perhitungan selisih2 ditunjukkan pada baris 15. Jika nilai a≠b, maka akan dilakukan pengecekan pada D[a, b, j]. Jika nilai D[a, b, j] ≠ 99, maka nilai selisih1 adalah absolut v[a, j] – v[b, j].
9. Proses perhitungan max1, max2 dan matrix_D ditunjukkan pada baris 19-24. Jika nilai a≠b, maka akan dilakukan proses perhitungan max1[a, b] dan max2[a,b] yang merupakan nilai maksimal(nilaiMax) dari selisih1 dan selesih2
10. Nilai dari matrix_D[a,b] didapatkan dari hasil bagi max1[a, b] dengan max2[a, b]. Proses ditunjukkan pada baris 23.

5.3.1.7 Penentuan *Threshold Concordance* dan *Threshold Discordance*

Proses menentukan *threshold concordance* dan *threshold discordance* ELECTRE diimplementasikan berdasarkan rancangan pada Gambar 4.30 untuk *threshold concordance* dan Gambar 4.34 *threshold discordance*. Implementasi ditunjukkan pada gambar 5.8.

```
1  double tempC = 0, tempD = 0;
2  for (int a = 0; a < m; a++)
3  {
4     for (int b = 0; b < m; b++)
5     {
6        if (a != b)
7        {
8           tempC += matrix_C[a, b];
9           tempD += matrix_D[a, b];
10       }
11    }
12 }
13                thres_C = tempC / (m * (m - 1));
14                thres_D = tempD / (m * (m - 1));
```

Gambar 5.8 Implementasi *Threshold Concordance* dan *Threshold Discordance*

Penjelasan algoritma *threshold concordance* dan *threshold discordance* pada Gambar 5.8 Yaitu:

1. Deklarasi variable untuk menyimpan jumlah dari seluruh nilai matrix_C pada *concordance* dan untuk menyimpan jumlah dari seluruh nilai matrix_D pada *discordance* ditunjukkan pada baris 1.
2. Proses perhitungan tempC dan dan tempD ditunjukkan pada baris 10-13. Jika nilai a≠b, maka tempC akan terus menyimpan nilai dari matrix_C[a, b], dan tempD akan menyimpan nilai dari matrix_D[a, b]. Proses ini akan terus dilakukan sebanyak a dan b serta nilai a ≠ nilai b. Proses ini akan dilakukan sesuai dengan perulangan pada baris 4 dan 2.
3. tempC merupakan hasil penjumlahan nilai elemen matrix_C[a][b] yang ditunjukkan pada baris 8. tempD merupakan hasil penjumlahan nilai elemen matrix_D[a][b] yang ditunjukkan pada baris 9.

4. Proses perhitungan nilai *threshold concordance* ditunjukkan pada baris 13. Nilai thres_C didapatkan dengan cara membagi nilai tempC dengan hasil dari jumlah baris dikali dengan jumlah baris dikurang 1.
5. Proses perhitungan nilai *threshold discordance* ditunjukkan pada baris 14. Nilai thres_D didapatkan dengan cara membagi nilai tempD dengan hasil dari jumlah baris dikali dengan jumlah baris dikurang 1.

5.3.1.8 Penentuan Matriks Dominan *Concordance* dan *Discordance*

Proses menentukan matriks dominan *concordance* dan matriks dominan *discordance* ELECTRE diimplementasikan berdasarkan rancangan pada Gambar 4.32 untuk matriks dominan *concordance* dan Gambar 4.36 matriks dominan *discordance*. Implementasi ditunjukkan pada gambar 5.8.

```
1    for (int a = 0; a < m; a++)
2    {
3        for (int b = 0; b < m; b++)
4        {
5          if (a != b)
6          {
             if (matrix_C[a, b] > thres_C || matrix_C[a, b] == thres_C)
7  { F[a, b] = 1; } else { F[a, b] = 0; }
             if (matrix_D[a, b] > thres_D || matrix_D[a, b] == thres_D)
8  { G[a, b] = 1; } else { G[a, b] = 0; }
9           }
10       }
11   }
```

Gambar 5.9 Implementasi Algoritma Matriks Dominan *Concordance* dan *Discordance*

Penjelasan Algoritma Matriks Dominan *Concordance* dan *Discordance* pada Gambar 5.9 Yaitu:

1. Proses untuk menghitung matriks dominan *concordance* dan *discordance* ditunjukkan pada baris 1-10. Jika nilai a≠b, maka akan dilakukan pengecekan pada matriks *concordance*. Jika nilai matrix_C > thres_C atau nilai matrix_C = thres_C, maka nilai nilai matriks dominan *concordance* adalah 1, jika tidak maka nilai matriks dominan *concordance* adalah 0. Pengecekan juga dilakukan pada matriks *discordance*. Jika nilai matrix_D > thres_D atau nilai matrix_D = thres_D maka nilai nilai matriks dominan

discordance adalah 1, jika tidak maka nilai matriks dominan *discordance* adalah 0. Proses ini akan terus dilakukan sesuai dengan perulangan pada baris 4 dan 2.

5.3.1.9 Penentuan Matriks Dominan *Aggregate*

Proses menentukan matriks dominan *aggregate* ELECTRE diimplementasikan berdasarkan rancangan pada Gambar 4.38 untuk matriks dominan *concordance* dan Gambar 4.35 matriks dominan *discordance*. Implementasi ditunjukkan pada gambar 5.8.

```
1   E = new double[m, m];
2   for (int a = 0; a < m; a++)
3   {
4    for (int b = 0; b < m; b++)
5    {
6     if (a != b)
7     {
8      E[a, b] = F[a, b] * G[a, b];
9     }
10   }
11  }
```

Gambar 5.10 Implementasi Algoritma Matriks Dominan *Aggregate*

Penjelasan Algoritma Matriks Dominan *Aggregate* pada Gambar 5.10 Yaitu:

1. Deklarasi variable untuk matriks dominan *aggregate* ditunjukkan pada baris 1.
2. Proses penentuan matriks dominan *aggregate* ditunjukkan pada baris 2-11. Nilai dari matriks dominan *aggregate* didapatkan dengan cara mengalikan nilai dari matriks dominan *concordance* dan matriks dominan *discordance* selama a≠b. Proses ini akan terus dilakukan sesuai dengan perulangan pada baris 4 dan baris 2.

5.3.1.10 Eliminasi Alternatif *Less Favorable*

Proses eliminasi alternatif *less favorable* ELECTRE diimplementasikan berdasarkan rancangan pada Gambar 4.40 untuk matriks dominan *concordance* dan Gambar 4.35 matriks dominan *discordance*. Implementasi ditunjukkan pada gambar 5.8.

```
1   double[] eliminasi = new double[m];
```

```
2   for (int a = 0; a < m; a++)
3   {
4    eliminasi[a] = 0;
5    for (int b = 0; b < m; b++)
6    {
7     if (a != b)
8     {
9      if (E[a, b] == 1) { eliminasi[a]++; }
10    }
11   }
12   }
13  for (int a = 0; a < m; a++)
14  {
15  if (eliminasi[a] > 0)
16  {
          hasil[a] = "Diterima";
    }
    else
    {
     hasil[a] = "Tidak Diterima";
    }
                }
```

Gambar 5.11 Implementasi Algoritma Alternatif *Less Favorable*

Penjelasan Algoritma Alternatif *Less Favorable*perhitungan matriks normalisasi pada Gambar 5.11 Yaitu:

1. Deklarasi variable untuk eliminasi alternatif *less favorable* ditunjukkan pada baris 1
2. Deklarasi yang menyatakan bahwa nilai awal dari eliminasi alternatif *less favorable* adalah 0 ditunjukkan pada baris 4
3. Proses perhitungan alternatif *less favorable* ditunjukkan pada baris 2-12. Jika nilai a≠b maka akan dilakukan pengecekan pada nilai matriks dominan *aggregate*. Jika nilai matriks dominan *aggregate* = 1, dan disimpan pada matriks eliminasi[a]. Proses ini akan dilakukan sesuai dengan perulangan baris 5 dan 2.
4. Jika nilai e pada indeks ke-a tersebut adalah yang terbesar atau ≠0, maka alternatif ke-a dapat dieliminasi

5.3.2 *Weighted Product*

5.3.2.1 Menghitung Nilai Vektor S

Proses menghitung nilai vektor S diimplementasikan berdasarkan rancangan pada Gambar 4.41. Implementasi ditunjukkan pada gambar 5.12.

```
matriks_w = new double[z, n2, c];
matriks_s = new string[m2, n2, c];
double[, ,] matriks_s_convert = new double[m2, n2, c];
double totalIPA = 0;
double[] bobotIPA = new double[c];
double[] bobotIPS = new double[c];
double totalIPS = 0;

double[, ,] Mpow = new double[m2, n2, c];
double[] t_ipa = new double[m2];
double[] t_ips = new double[m2];

for (int i = 0; i < m2; i++) //perulangan untuk baris
{
 for (int j = 0; j < c; j++) //perulangan untuk kolom
 {
  matriks_s[i, 0, j] = dataReader1.GetString(j);
 }
}
for (int i = 0; i < m2; i++)
{
 for (int j = 0; j < c; j++) //perulangan untuk kolom
 {
  matriks_s[i, 1, j] = dataReader.GetString(j);
 }
}
for (int i = 0; i < m2; i++)
{
  for (int j = 0; j < c; j++)
  {
if (matriks_s[i, 0, j] == "terserah" || matriks_s[i, 0, j] == "IPA" || matriks_s[i, 0, j] == "IPS" || matriks_s[i, 0, j] == "R" || matriks_s[i, 0, j] == "C-" || matriks_s[i, 0, j] == "C" || matriks_s[i, 0, j] == "C+" || matriks_s[i, 0, j] == "T")
    {
      int konversi = convertpsikotes(matriks_s[i, 0, j]) + convertkelas(matriks_s[i, 0, j], true);
      matriks_s_convert[i, 0, j] = konversi;
      }
```

```
36          else
37          {
38          matriks_s_convert[i, 0, j] = Convert.ToDouble(matriks_s[i, 0, j]);
39          }
40      }
41  for (int j = 0; j < c; j++)
42  {
    if (matriks_s[i, 1, j] == "terserah" || matriks_s[i, 1, j] == "IPA" ||
    matriks_s[i, 1, j] == "IPS" || matriks_s[i, 1, j] == "R" || matriks_s[i,
    1, j] == "C-" || matriks_s[i, 1, j] == "C" || matriks_s[i, 1, j] == "C+"
43  || matriks_s[i, 1, j] == "T")
44   {
       int    konversi    =    convertpsikotes(matriks_s[i,    1,    j])    +
45  convertkelas(matriks_s[i, 1, j], false);
46      matriks_s_convert[i, 1, j] = konversi;
47   }
48      else
49    {
50    matriks_s_convert[i, 1, j] = Convert.ToDouble(matriks_s[i, 1, j]);
51     }
52    }
53  }
54  for (int i = 0; i < z; i++)
55  {
56   for (int j = 0; j < c; j++)
57   {
58
59  matriks_w[i, 0, j] = double.Parse(dataReader1.GetString(0).ToString());
60    totalIPA += matriks_w[i, 0, j];
61    }
62      for (int j = 0; j < c; j++)
63      {
64       bobotIPA[j] = matriks_w[i, 0, j] / totalIPA;
65      }
66  }
67  for (int i = 0; i < z; i++)
68  {
69    for (int j = 0; j < c; j++) //pengulangan sebanyak kolom
70     {
       matriks_w[i,                  1,                  j]                  =
    double.Parse(dataReader1.GetString(0).ToString());    //mengambil    bobot
71  untuk setiap kolom
72       totalIPS += matriks_w[i, 1, j];
73     }
74  for (int j = 0; j < c; j++)
```

```
{
    bobotIPS[j] = matriks_w[i, 0, j] / totalIPS;
}
}
for (int i = 0; i < m2; i++)
{
 for (int j = 0; j < c; j++)
 {
   Mpow[i, 0, j] = System.Math.Pow(matriks_s_convert[i, 0, j], bobotIPA[j]);

  }
  for (int j = 0; j < c; j++)
 {
   Mpow[i, 1, j] = System.Math.Pow(matriks_s_convert[i, 1, j], bobotIPA[j]);
  }
  for (int i = 0; i < m2; i++)
  {
      for (int j = 0; j < c; j++)
      {
     Mpow[i, 0, j] = System.Math.Pow(matriks_s_convert[i, 0, j], bobotIPA[j]);
       if (j == 0)
       t_ipa[i] += Mpow[i, 0, j];
       else
       t_ipa[i] *= Mpow[i, 0, j];
    }
    for (int j = 0; j < c; j++)
      {
       Mpow[i, 1, j] = System.Math.Pow(matriks_s_convert[i, 1, j], bobotIPA[j]);
        if (j == 0)
        t_ips[i] += Mpow[i, 1, j];
         else
         t_ips[i] *= Mpow[i, 1, j];
       }
}
```

Gambar 5.12 Implementasi Algoritma Vektor S

Penjelasan algoritma vektor S pada Gambar 5.12 Yaitu:

1. Deklarasi variable untuk bobot alternatif ditunjukkan pada baris 1

2. Deklarasi variable untuk menyimpan data alternatif dari calon calon siswa baru dan kriteria-keriteria dari peminatan IPA dan IPS ditunjukkan pada baris 2
3. Deklarasi variable untuk menyimpan hasil konversi dari data alternatif calon calon siswa baru dan kriteria-keriteria dari peminatan IPA dan IPS ditunjukkan pada baris 3
4. Deklarasi variable untuk menyimpan jumlah bobot untuk alternatif IPA ditunjukkan pada baris 4
5. Deklarasi variable untuk menyimpan hasil pembagian bobot alternatif dengan jumlah bobot pada peminatan IPA ditunjukkan pada baris 5
6. Deklarasi variable untuk menyimpan hasil pembagian bobot alternatif dengan jumlah bobot pada peminatan IPS ditunjukkan pada baris 6
7. Deklarasi variable untuk menyimpan jumlah bobot untuk alternatif IPS ditunjukkan pada baris 7
8. Dekalrasi variable untuk menyimpan hasil dari matriks_s_convert yang dipangkatkan dengan masing-masing bobot alternatif (bobotIPA/bobotIPS) ditunjukkan pada baris 9
9. Deklarasi variable untuk menyimpan hasil kali dari seluruh hasil Mpow pada peminatan IPA ditunjukkan pada baris 10
10. Deklarasi variable untuk menyimpan hasil kali dari seluruh hasil Mpow pada peminatan IPS ditunjukkan pada baris 11
11. Proses mengambil data calon calon siswa baru sebanyak kriteria utnuk alternatif IPA. Data yang telah diambil disimpan pada matriks_s ditunjukkan pada baris 13-19
12. Proses mengambil data calon calon siswa baru sebanyak kriteria utnuk alternatif IPS. Data yang telah diambil disimpan pada matriks_s ditunjukkan pada baris 22-26
13. Proses konversi data alternatif IPA. Nilai matriks_s akan dikonversi lalu nilainya akan disimpan pada matriks_s_convert. Proses ini akan dilakukan sebanyak alternatif calon calon siswa baru dan sebanyak kriteria ditunjukkan pada baris 27-40

14. Proses konversi data alternatif IPS. Nilai matriks_s akan dikonversi lalu nilainya akan disimpan pada matriks_s_convert. Proses ini akan dilakukan sebanyak alternatif calon calon siswa baru dan sebanyak kriteria ditunjukkan pada baris 41-53
15. Proses untuk mencari bobotIPA alternatif-j. Proses ini dilakukan dengan mengambil terlebih dahulu bobot masing-masing kriteria dan disimpan pada variable matriks_w, lalu nilai-nilai pada matriks_w dijumlahkan dan disimpan pada totalIPA. Lalu akan dilakukan perulangan sebanyak kriteria untuk mencari bobotIPA kriteria ke-j. bobotIPA dapat dihitung dengan cara membagi matriks_w dengan totalIPA ditunjukkan pada baris 54-66
16. Proses untuk mencari bobotIPS alternatif-j. Proses ini dilakukan dengan mengambil terlebih dahulu bobot masing-masing kriteria dan disimpan pada variable matriks_w, lalu nilai-nilai pada matriks_w dijumlahkan dan disimpan pada totalIPS. Lalu akan dilakukan perulangan sebanyak kriteria untuk mencari bobotIPS kriteria ke-j. bobotIPS dapat dihitung dengan cara membagi matriks_w dengan totalIPS ditunjukkan pada baris 67-68
17. Proses untuk mencari hasil pangkat antara matriks_s_convert pada kriteria j dengan bobotIPA kriteria j. Proses perulangan dilakukan sebanyak data alternatif calon calon siswa baru. Pada baris 81-85, dilakukan proses pemangkatan matriks_s_convert kriteria j dengan bobotIPA kriteria j pada peminatan IPA dan dilakukan sebanyak kriteria. Pada baris 86-89, dilakukan proses pemangkatan matriks_s_convert kriteria j dengan bobotIPS kriteria j pada peminatan IPS dan dilakukan sebanyak kriteria ditunjukkan pada baris 79-89
18. Proses perhitungan vektor S ditunjukkan pada baris 90-107. Proses perulangan akan dilakukan sebanyak alternatif. Baris 92-99 adalah proses perhitungan vektor S IPA yang hasilnya akan disimpan pada variable t_ipa. Baris 95 menjelaskan bahwa apabila kriteria j adalah 0, maka nilai t_ipa akan ditambahkan dengan Mpow. Namun jika tidak, maka t_ipa merupakan hasil perkalian Mpow untuk seluruh alternatif. Proses ini akan dilakukan sebanyak kriteria. Baris 100-107 adalah proses perhitungan vektor S IPS yang hasilnya akan disimpan pada variable t_ips. Baris 104 menjelaskan

bahwa apabila kriteria j adalah 0, maka nilai t_ips akan ditambahkan dengan Mpow. Namun jika tidak, maka t_ips merupakan hasil perkalian Mpow untuk seluruh alternatif. Proses ini akan dilakukan sebanyak kriteria.

5.3.2.2 Menghitung Nilai Vektor V

Proses menghitung nilai vektor V diimplementasikan berdasarkan rancangan pada Gambar 4.43. Implementasi ditunjukkan pada gambar 5.13.

```
1  double[] v_ipa_nilai = new double[m2];
2  double[] v_ips_nilai = new double[m2];
3  for (int i = 0; i < m2; i++)
4  {
5    v_ipa_nilai[i] = t_ipa[i] / (t_ipa[i] + t_ips[i]);
6    v_ips_nilai[i] = t_ips[i] / (t_ipa[i] + t_ips[i]);
7  }
```

Gambar 5.13 Implementasi Algoritma Vektor V

Penjelasan algoritma vektor V pada Gambar 5.13 Yaitu:

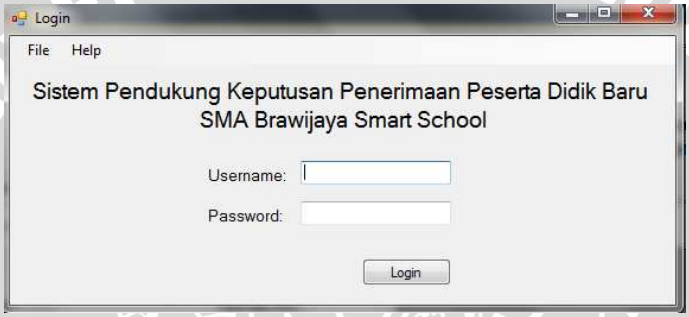
1. Deklarasi variable untuk nilai vektor V IPA ditunjukkan pada baris 1
2. Deklarasi variable untuk nilai vektor V IPS ditunjukkan pada baris 2
3. Proses perhitungan nilai vektor V IPA dan vektor V IPS ditunjukkan pada baris 3-7.
4. Nilai vektor V IPA didapatkan dari nilai t_ipa yang dibagi dengan jumlah t_ipa ditambah t_ips ditunjukkan pada baris 5.
5. Nilai vektor V IPS didapatkan dari nilai t_ips yang dibagi dengan jumlah t_ipa ditambah t_ips ditunjukkan pada baris 6.

5.4 Implementasi Antarmuka

Implementasi antarmuka calon siswa baru baru SMA Brawijaya *Smart School* (SMA BSS) ini terdiri dari empat bagian utama. Tiga bagian utama tersebut antara lain halaman login, halaman staff IT, halaman Kepala Sekolah dan Ketua PPDB, serta halaman Sie Test dan Guru BK.

5.4.1 Tampilan Antarmuka Halaman Login

Halaman login digunakan untuk mengidentifikasi pengguna yang akan masuk ke dalam sistem. Halaman ini adalah halaman yang akan muncul pertama kali ketika program dijalankan. Untuk bisa masuk ke dalam sistem, pengguna harus memasukkan data-data login, yakni *username* dan *password*. Ketika pengguna menekan tombol 'Login', sistem akan melakukan identifikasi hasil inputan tersebut apakah tersimpan dalam database. Jika data berhasil ditemukan dalam database, maka pengguna akan diteruskan ke dalam sistem bergantung dari jenis pengguna tersebut (IT, Kepala Sekolah, Ketua PPDB, Sie Test, atau Guru BK). Implementasi antarmuka halaman login ditunjukan dalam Gambar 5.14



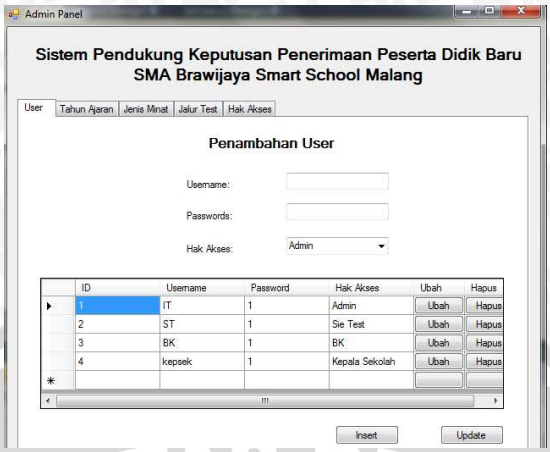
Gambar 5.14 Implementasi Antarmuka Halaman Login
Sumber: Implementasi

5.4.2 Tampilan Antarmuka Halaman Staff IT

Halaman Staff IT merupakan halaman yang disediakan setelah pengguna dengan hak akses 'IT' berhasil login. Halaman staff IT ini memiliki beberapa tab halaman, antara lain halaman pengguna, halaman tahun ajaran, halaman jenis minat, halaman jalur test, dan halaman hak akses.

1. Halaman Pengguna

   Halaman Pengguna digunakan untuk mengelola data pengguna sistem yang tersimpan di dalam database. Staff IT bisa menambah data pengguna baru, mengubah data pengguna, dan menghapus data pengguna yang ada di dalam database. Implementasi antarmuka halaman pengguna ditunjukkan dalam Gambar 5.15



Gambar 5.15 Implementasi Antarmuka Halaman Pengguna
Sumber: Implementasi

Staff IT bisa menambah data pengguna baru dengan mengisi terlebih dahulu username, password, dan hak akses lalu menekan tombol 'Insert'. Staff IT juga bisa mengubah data pengguna yang ada dengan menekan tombol 'Ubah' ditempat data yang akan diubah, dan data tersebut akan muncul pada tempat username, password, dan hak akses. Apabila staff IT telah mengubah data tersebutt, maka staff IT dapat menekan tombol 'Update' yang berada di kanan bawah untuk menyimpan hasil perubahan ke dalam database.

2. Halaman Tahun Ajaran

Halaman TahunAjaran digunakan untuk mengelola data tahun ajaran yang tersimpan di dalam database. Staff IT bisa menambah tahun ajaran, mengubah data tahun ajaran, dan menghapus data tahun ajaran yang ada di dalam database. Implementasi antarmuka halaman pengguna ditunjukkan dalam Gambar 5.16

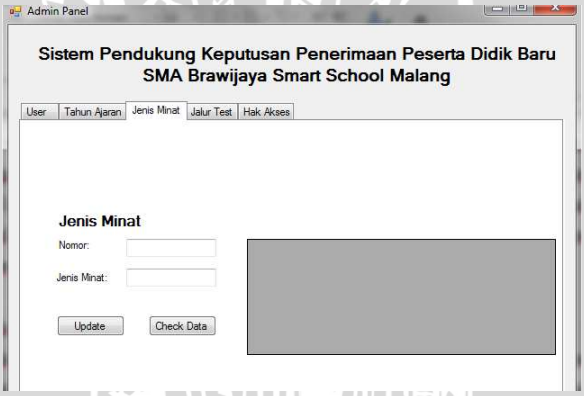


Gambar 5.16 Implementasi Antarmuka Halaman Tahun Ajaran
Sumber: Implementasi

Staff IT bisa menambah data Tahun Ajaran dengan mengisi terlebih dahulu nama tahun ajaran di *field* 'Tahun Ajaran' lalu menekan tombol 'Insert'. Staff IT juga bisa mengubah data Tahun Ajaran yang ada dengan menekan tombol 'Ubah' ditempat data yang akan diubah, dan data tersebut akan muncul pada tempat nomor dan tahun ajaran. Apabila staff IT telah mengubah data tersebutt, maka staff IT dapat menekan tombol 'Update'. Staff IT juga bisa menghapus data Tahun Ajaran yang ada dengan menekan sel ditempat data yang akan diubah, dan data tersebut akan muncul pada tempat nomor dan tahun ajaran, lalu menekan tombol 'Delete'.

3. Halaman Jenis Minat

Halaman Jenis Minat digunakan untuk mengelola data jenis minat yang tersimpan di dalam database. Staff IT hanya bisa mengubah data tahun ajaran yang ada di dalam database. Implementasi antarmuka halaman pengguna ditunjukkan dalam Gambar 5.17

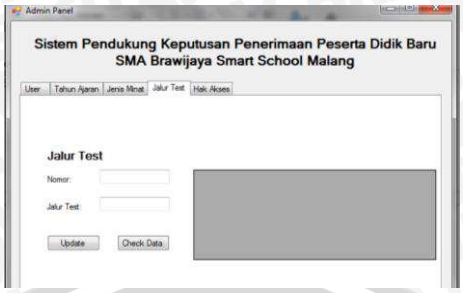


Gambar 5.17 Implementasi Antarmuka Halaman Jenis Minat
Sumber: Implementasi

Staff IT bisa mengubah data Jenis Minar yang ada dengan menekan sel di tempat data yang akan diubah, dan data tersebut akan muncul pada tempat nomor dan jenis minat. Apabila staff IT telah mengubah data tersebut, maka staff IT dapat menekan tombol 'Update'.

4. Halaman Jalur Test

Halaman Jenis Minat digunakan untuk mengelola data jalur test yang tersimpan di dalam database. Staff IT hanya bisa mengubah data jalur test yang ada di dalam database. Implementasi antarmuka halaman pengguna ditunjukkan dalam Gambar 5.18



Gambar 5.18 Implementasi Antarmuka Halaman Jalur Test
Sumber: Implementasi

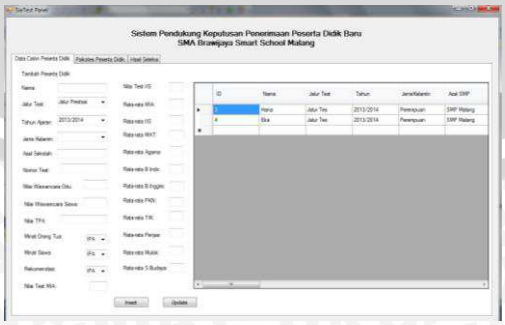
Staff IT bisa mengubah data jalur test yang ada dengan menekan sel di tempat data yang akan diubah, dan data tersebut akan muncul pada tempat nomor dan jalur test. Apabila staff IT telah mengubah data tersebut, maka staff IT dapat menekan tombol 'Update'.

5.4.3 Tampilan Antarmuka Kepala Sekolah/Ketua PPDB

Sie Test/Guru BK merupakan halaman yang disediakan setelah pengguna dengan hak akses 'Sie Test' atau 'Guru BK' berhasil login. Halaman Sie Test ini memiliki beberapa tab halaman, antara lain halaman data calon calon siswa baru, halaman psikotes calon siswa baru, dan halaman hasil seleksi,

5.4.3.1 Halaman Data Calon Calon siswa baru

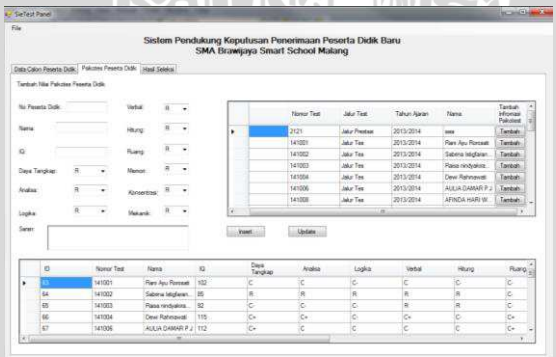
Halaman calon siswa baru digunakan untuk menampilkan data-data calon calon siswa baru yang tersimpan di dalam database. Halaman calon siswa baru digunakan sekaligun untuk melakukan proses menambah data calon calon siswa baru baru, merubah data calon calon siswa baru yang telah disimpan di dalam database, dan menghapus data calon calon siswa baru. Data calon calon siswa baru ditampilkan secara terurut berdasarkan ID yang dimiliki. Implementasi antarmuka halaman data calon calon siswa baru ditunjukkan dalam Gambar 5.19



Gambar 5.19 Implementasi Antarmuka Halaman Data Calon Calon siswa baru
Sumber: Implementasi

Sie Test/Guru BK dapat menambah data baru calon calon siswa baru dengan cara mengisi semua *field* yang ada pada halaman data calon calon siswa baru. Setelah mengisi seluruh *field* yang ada, Sie Test/Guru BK dapat menekan tombol 'Insert' yang ada di bawah untuk menyimpan data ke dalam database. Sie Test/BK dapat mengubah data calon calon siswa baru yang telah tersimpan di dalam database dengan cara menekan tombol 'Ubah' yang berada di sisi kanan datagrid yang menampilkan seluruh data calon siswa baru. Setelah tombol 'Ubah' ditekan, maka data dari calon calon siswa baru yang hendak diubah akan muncul ke *field* data yang berada disisi kiri. Sie Test/Guru BK dapat mengubah data sesuai dengan kebutuhan dan menyimpan hasil perubahan tersebut ke dalam database dengan menekan tombol 'Update'. Untuk menghapus data calon calon siswa baru, Sie Test/BK hanya perlu menekan tombol 'Hapus'.

Halaman calon siswa baru digunakan untuk menampilkan data-data calon calon siswa baru yang tersimpan di dalam database. Halaman calon siswa baru digunakan sekaligun untuk melakukan proses menambah data calon calon siswa baru baru, merubah data calon calon siswa baru yang telah disimpan di dalam database, dan menghapus data calon calon siswa baru. Data calon calon siswa baru ditampilkan secara terurut berdasarkan ID yang dimiliki. Implementasi antarmuka halaman data calon calon siswa baru ditunjukkan dalam Gambar 5.20



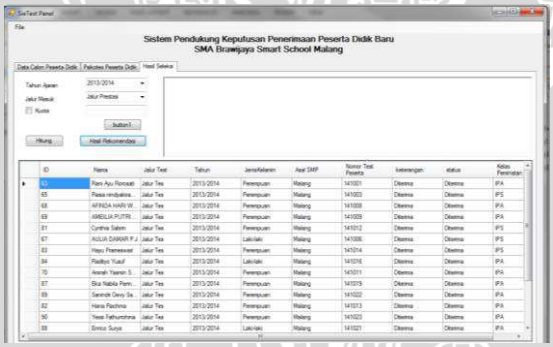
Gambar 5.20 Implementasi Antarmuka Halaman Data Psikotes Calon siswa baru
Sumber: Implementasi

Sie Test/Guru BK dapat menambah data baru calon calon siswa baru dengan memilih terlebih dahulu data calon calon siswa baru yang ingin ditambahkan informasi psikotes. Form tambah data psikotes berada di sisi kana form. Sie Test/Guru BK cara mengisi semua *field* yang ada pada halaman psikotes data calon calon siswa baru. Setelah mengisi seluruh *field* yang ada, Sie Test/Guru BK dapat

menekan tombol ‘Insert’ untuk menyimpan data ke dalam database. Sie Test/BK dapat mengubah data calon calon siswa baru yang telah tersimpan di dalam database dengan cara menekan tombol ‘Ubah’ yang berada di sisi kanan datagrid yang menampilkan seluruh data calon siswa baru. Setelah tombol ‘Ubah’ ditekan, maka data psikotes dari calon calon siswa baru yang hendak diubah akan muncul ke *field* data yang berada disisi kiri. Sie Test/Guru BK dapat mengubah data sesuai dengan kebutuhan dan menyimpan hasil perubahan tersebut ke dalam database dengan menekan tombol ‘Update’. Untuk menghapus data psikotes calon calon siswa baru, Sie Test/BK hanya perlu menekan tombol ‘Hapus’.

5.4.3.2 Halaman Hasil Seleksi

Halaman hasil seleksi digunakan untuk melakukan proses seleksi dan peminatan. Sie Test/Guru BK dapat memilih tahun dari data yang hendak diolah. Fasilitas kuota ditambahkan untuk melihat calon calon siswa baru yang diterima dan yang menjadi cadangan. Untuk mengetahui hasil seleksi secara keseluruhan, Sie Test/Guru BK hanya perlu menekan tombol ‘Hasil Rekomendasi’ untuk melihat hasil seleksi secara keseluruhan.



Gambar 5.21 Implementasi Antarmuka Halaman Hasil Seleksi
Sumber: Implementasi

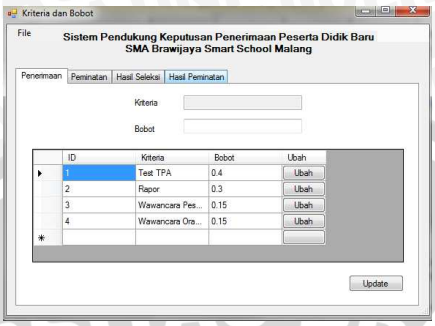
5.4.4 Tampilan Antarmuka Sie Test/Guru BK

Kepala Sekolah/Ketua PPDB merupakan halaman yang disediakan setelah pengguna dengan hak akses ‘Kepala Sekolah’ atau ‘Ketua PPDB’ berhasil login. Halaman Kepala Sekolah/Ketua PPDB ini memiliki beberapa tab halaman, antara lain halaman bobot penerimaan, bobot peminatan, hasil seleksi, dan hasil peminatan.

5.4.4.1 Halaman Bobot Penerimaan

Halaman calon siswa baru digunakan untuk menampilkan data-data calon calon siswa baru yang tersimpan di dalam database. Halaman bobot penerimaan

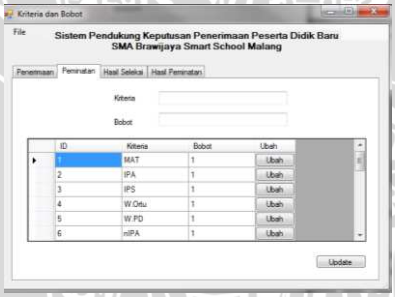
digunakan untuk melakukan edit bobot kriteria yang digunakan untuk seleksi penerimaan. Implementasi antarmuka halaman datbobot penerimaan ditunjukkan dalam Gambar 5.22



Gambar 5.22 Implementasi Antarmuka Halaman Bobot Penerimaan
Sumber: Implementasi

5.4.4.2 Halaman Bobot Peminatan

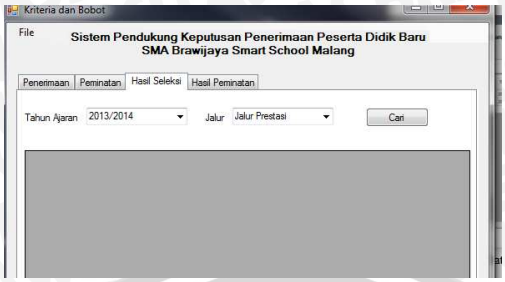
Halaman calon siswa baru digunakan untuk menampilkan data-data calon calon siswa baru yang tersimpan di dalam database. Halaman bobot penerimaan digunakan untuk melakukan edit bobot kriteria yang digunakan untuk proses peminatan. Implementasi antarmuka halaman datbobot penerimaan ditunjukkan dalam Gambar 5.23



Gambar 5.23 Implementasi Antarmuka Bobot Peminatan
Sumber: Implementasi

5.4.4.3 Halaman Hasil Seleksi

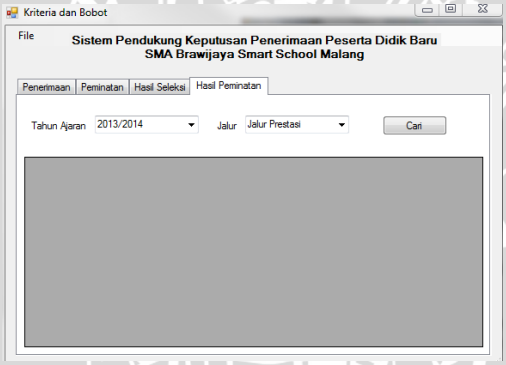
Halaman calon siswa baru digunakan untuk menampilkan data-data calon calon siswa baru yang tersimpan di dalam database. Halaman hasil seleksi digunakan untuk melihat hasil seleksi yang telah dilakukan perhitungannya oleh Sie Test/Guru BK. Kepala sekolah/Ketua PPDB bisa memilih data hasil seleksi berdasarkan tahun dan jalur tes. Implementasi antarmuka halaman hasil seleksi ditunjukkan dalam Gambar 5.24



Gambar 5.24 Implementasi Antarmuka Hasil Seleksi
Sumber: Implementasi

5.4.4.4 Halaman Hasil Peminatan

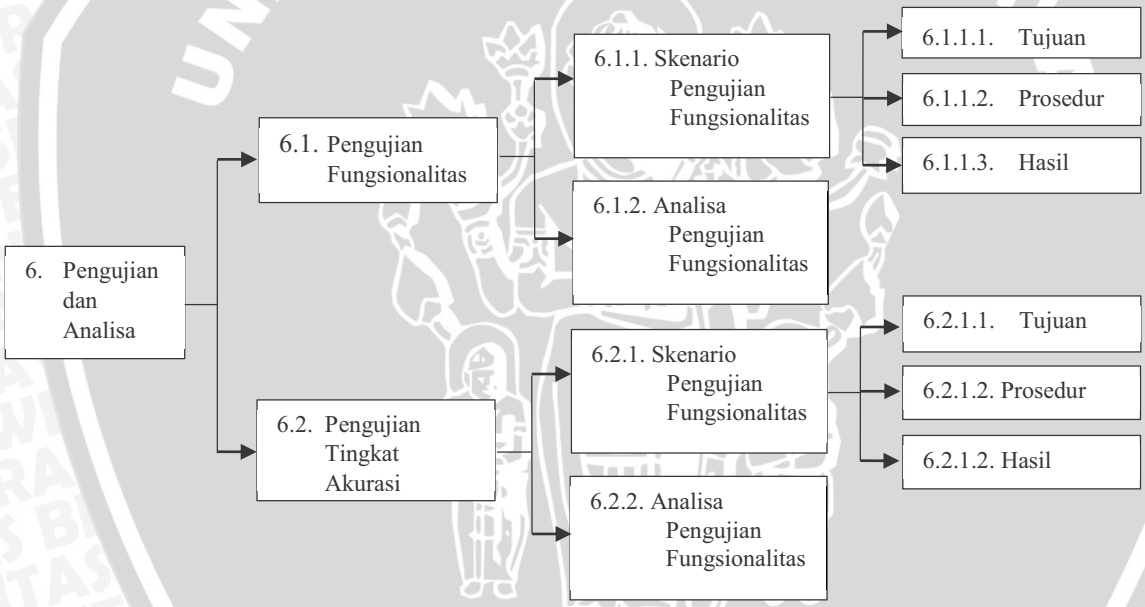
Halaman calon siswa baru digunakan untuk menampilkan data-data calon calon siswa baru yang tersimpan di dalam database. Halaman hasil seleksi digunakan untuk melihat hasil peminatan yang telah dilakukan perhitungannya oleh Sie Test/Guru BK. Kepala sekolah/Ketua PPDB bisa memilih data hasil seleksi berdasarkan tahun dan jalur tes. Implementasi antarmuka halaman hasil pak Aan ditunjukkan dalam Gambar 5.25



Gambar 5.25 Implementasi Antarmuka Bobot Peminatan
Sumber: Implementasi

BAB VI
PENGUJIAN DAN ANALISA

Pada bab ini membahas mengenai proses pengujian sistem pendukung keputusan seleksi penerimaan calon siswa baru baru. Proses pengujian dilakukan melalui dua tahap pengujian, yaitu pengujian fungsionalitas dan pengujian akurasi. Pengujian fungsionalitas digunakan untuk menguji apakah sistem yang dibangun telah sesuai dengan kebutuhan sistem. Pengujian akurasi digunakan untuk mengukur tingkat akurasi dari sistem. Pengujian akurasi dilakukan dengan membandingkan hasil dari keluaran sistem dan hasil perhitungan manual yang dilakukan oleh pihak SMA BSS. Diagram alir proses pengujian dan analisis ditunjukkan pada Gambar 6.1.



Gambar 6.1 Pohon Pengujian dan Analisa

6.1 Pengujian Fungsionalitas

Pada bagian ini akan dijelaskan tentang skenario pengujian yang akan dilakukan pada skenario pengujian pertama yaitu pengujian fungsionalitas sistem bedasarkan daftar kebutuhan sistem. Pengujian fungsionalitas merupakan pengujian yang dilakukan terhadap sistem untuk mengetahui apakah sistem yang dibangun telah sesuai dengan daftar kebutuhan sistem yang telah ditentukan. Daftar kebutuhan yang digunakan dalam proses pengujian fungsionalitas ditunjukkan pada Tabel 4.2. Pada tabel tersebut terdapat 22 kebutuhan yang harus terdapat pada

sistem yang dibangun, 22 daftar kebutuhan yang telah diimplementasikan pada sistem tersebut nantinya akan diuji dengan pengujian fungsionalitas untuk mengatahui tingkat kesesuaian antara kinerja sistem dengan daftar kinerja sistem yang telah disebutkan.

6.1.1. Skenario Pengujian Fungsionalitas

Sub bab berikut akan menjelaskan tentang tujuan, prosedur, serta hasil akhir yang didapatkan dari skenario pengujian pertama yang merupakan skenario pengujian fungsionalitas.

6.1.1.1. Tujuan

Pengujian ini dilakukan untuk mengetahui apakah sistem yang dibangun telah sesuai dengan daftar kebutuhan sistem yang telah ditentunkan.

6.1.1.2. Prosedur

Prosedur pengujian fungsionalitas dilakukan dengan cara membuat kasus uji pengujian untuk setiap daftar kebutuhan sistem yang telah ditentukan pada Tabel 4.2. Berdasarkan daftar kebutuhan yang ditunjukkan pada Tabel 4.2 terdapat 22 daftar kebutuhan sistem yang nantinya akan diuji. Setiap kasus uji daftar kebutuhan sistem akan berisi tentang nama kasus uji yang dilakukan, tujuan pengujian, prosedur pengujian, dan hasil yang diharapkan. Adapun kasus uji yang digunakan untuk pengujian fungsionalitas adalah sebagai berikut :

6.1.1.2.1. Kasus Uji Login

Kasus uji login menjelaskan tentang pengujian fungsionalitas proses login seperti yang ditunjukkan pada Tabel 6.1.

Tabel 6.1 Penjelasan Kasus Uji untuk Pengujian Proses Login

Nama Kasus Uji	Kasus Uji Login
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk kebutuhan masuk ke dalam sistem dengan tujuan agar pengguna dapat masuk/menggunakan sistem pendukung keputusan seleksi penerimaan calon siswa baru baru
Prosedur Uji	1. Sistem dijalankan ketika program dieksekusi 2. Pengguna akan langsung dihadapkan kepada halaman login. 3. Pengguna mengisi *username* pengguna dan *password* ke dalam kolom yang telah disediakan.

	4. Pengguna mengeklik tombol login.
Hasil yang diharapkan	1. Sistem dapat mengakses *database* sistem. 2. Sistem dapat melakukan pemeriksaan data login yang telah dimasukan oleh pengguna. 3. Sistem dapat menampilkan pesan peringatan ketika data login yang dimasukan tidak sesuai dengan data login yang tersimpan dalam *database* sistem. 4. Sistem dapat menampilkan halaman utama sesuai dengan hak akses masing-masing pengguna ketika data login sesuai dengan data pada *database*.

6.1.1.2.2. Kasus Uji Halaman Utama (*IT Service*)

Kasus uji halaman utama menjelaskan tentang pengujian fungsionalitas proses masuk dalam halaman utama setelah pengguna dengan hak akses admin berhasil melakukan proses login, ditunjukkan pada Tabel 6.2.

Tabel 6.2 Penjelasan Kasus Uji untuk Pengujian Halaman Utama (*IT Service*)

Nama Kasus Uji	Kasus Uji Halaman Utama
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk masuk ke halaman utama admin.
Prosedur Uji	1. Pengguna dengan hak akses sebagai admin masuk ke dalam sistem pada halaman utama admin. 2. Pengguna dihadapkan pada halaman utama admin dengan tab menu Data Pengguna, Data Tahun Ajaran, Data Jalur Masuk, dan Data Kelas Peminatan.
Hasil yang diharapkan	1. Sistem dapat menampilkan data pada masing-masing tab menu yang dipilih

a. Kasus Uji Halaman Kelola Data Pengguna

Kasus uji kelola data pengguna menjelaskan tentang pengujian fungsionalitas proses kelola data pengguna pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai admin seperti yang ditunjukkan pada Tabel 6.3.

Tabel 6.3 Penjelasan Kasus Uji untuk Pengujian Proses Kelola Data Pengguna

Nama Kasus Uji	Kasus Uji Kelola Data Pengguna
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk mengelola data pengguna.

Prosedur Uji	1. Pengguna dengan hak akses sebagai admin masuk ke dalam sistem pada halaman utama admin. 2. Pengguna memilih tab menu Data Pengguna 3. Pengguna dihadapkan pada halaman utama admin yang memuat data-data pengguna dengan menu-menu yang digunakan untuk menambah, mengubah, menghapus dan melihat data pengguna.
Hasil yang diharapkan	1. Sistem dapat menampilkan data pengguna yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang digunakan untuk menambah, mengubah, menghapus, ataupun melihat data pengguna pada sistem.

b. Kasus Uji Kelola Tambah Data Pengguna

Kasus uji tambah data pengguna menjelaskan tentang pengujian fungsionalitas proses menambah data pengguna pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai admin seperti yang ditunjukkan pada Tabel 6.4.

Tabel 6.4 Penjelasan Kasus Uji untuk Pengujian Proses Tambah Data Pengguna

Nama Kasus Uji	Kasus Uji Tambah Data Pengguna
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses tambah data pengguna.
Prosedur Uji	1. Pengguna dengan hak akses sebagai admin masuk ke dalam sistem pada halaman utama admin. 2. Pengguna memilih menu Data Pengguna 3. Pengguna mengisi data pengguna baru dalam *field* yang telah disediakan. 4. Pengguna mengeklik tombol tambah.
Hasil yang diharapkan	1. Sistem dapat menyimpan data pengguna baru yang telah dimasukkan ke dalam *database* sistem. 2. Sistem dapat menampilkan data pengguna terbaru pada halaman utama admin

c. Kasus Uji Kelola Ubah Data Pengguna

Kasus uji ubah data pengguna menjelaskan tentang pengujian fungsionalitas proses mengubah data pengguna pada sistem yang hanya dapat dilakukan oleh

pengguna dengan hak akses sebagai admin seperti yang ditunjukkan pada Tabel 6.5.

Tabel 6.5 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Pengguna

Nama Kasus Uji	Kasus Uji Tambah Data Pengguna
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses ubah data pengguna.
Prosedur Uji	1. Pengguna dengan hak akses sebagai admin masuk ke dalam sistem pada halaman utama admin. 2. Pengguna memilih menu Data Pengguna 3. Pengguna memilih data yang diubah dalam tabel data pengguna 4. Pengguna mengubah data yang ada dalam *field* yang telah disediakan 5. Pengguna mengeklik tombol ubah
Hasil yang diharapkan	1. Sistem dapat menyimpan data pengguna yang telah diubah ke dalam *database* sistem. 2. Sistem dapat menampilkan data pengguna terbaru pada halaman utama admin

d. Kasus Uji Kelola Hapus Data Pengguna

Kasus uji hapus data pengguna menjelaskan tentang pengujian fungsionalitas proses menghapus data pengguna pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai admin seperti yang ditunjukkan pada Tabel 6.6.

Tabel 6.6 Penjelasan Kasus Uji untuk Pengujian Proses Hapus Data Pengguna

Nama Kasus Uji	Kasus Uji Hapus Data Pengguna
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses hapus data pengguna.
Prosedur Uji	1. Pengguna dengan hak akses sebagai admin masuk ke dalam sistem pada halaman utama admin. 2. Pengguna memilih tab menu Data Pengguna 3. Pengguna memilih data pengguna yang akan dihapus kemudian mengeklik tombol Hapus untuk menghapus data pengguna yang telah dipilih.
Hasil yang diharapkan	1. Sistem dapat menghapus data pengguna yang tersimpan dalam *database* sistem. 2. Sistem dapat menampilkan data pengguna terbaru pada halaman utama admin

e. Kasus Uji Kelola Data Tahun Ajaran

Kasus uji kelola data tahun ajaran menjelaskan tentang pengujian fungsionalitas proses kelola data tahun ajaran pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai admin seperti yang ditunjukkan pada Tabel 6.7.

Tabel 6.7 Penjelasan Kasus Uji untuk Pengujian Proses Kelola Data Tahun Ajaran

Nama Kasus Uji	Kasus Uji Kelola Data Tahun Ajaran
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk mengelola data tahun ajaran.
Prosedur Uji	1. Pengguna dengan hak akses sebagai admin masuk ke dalam sistem pada halaman utama admin. 2. Pengguna memilih tab menu Data Tahun Ajaran 3. Pengguna dihadapkan pada halaman utama admin yang memuat data-data tahun ajaran dengan menu-menu yang digunakan untuk menambah, mengubah, menghapus dan melihat data tahun ajaran
Hasil yang diharapkan	1. Sistem dapat menampilkan data tahun ajaran yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang digunakan untuk menambah, mengubah, menghapus, ataupun melihat data tahun ajaran pada sistem.

f. Kasus Uji Kelola Tambah Data Tahun Ajaran

Kasus uji tambah data tahun ajaran menjelaskan tentang pengujian fungsionalitas proses menambah data tahun ajaran pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai admin seperti yang ditunjukkan pada Tabel 6.8.

Tabel 6.8 Penjelasan Kasus Uji untuk Pengujian Proses Tambah Data Tahun Ajaran

Nama Kasus Uji	Kasus Uji Tambah Data Tahun Ajaran
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses tambah data tahun ajaran.
Prosedur Uji	1. Pengguna dengan hak akses sebagai admin masuk ke dalam sistem pada halaman utama admin.

	2. Pengguna memilih tab menu Data Tahun Ajaran 3. Pengguna mengisi data tahun ajaran baru dalam *field* yang telah disediakan. 4. Pengguna mengeklik tombol tambah.
Hasil yang diharapkan	1. Sistem dapat menyimpan data tahun ajaran baru yang telah dimasukkan ke dalam *database* sistem. 2. Sistem dapat menampilkan data tahun ajaran terbaru pada halaman utama admin

g. Kasus Uji Kelola Ubah Data Tahun Ajaran

Kasus uji ubah data tahun ajaran menjelaskan tentang pengujian fungsionalitas proses mengubah data tahun ajaran pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai admin seperti yang ditunjukkan pada Tabel 6.9.

Tabel 6.9 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Tahun Ajaran

Nama Kasus Uji	Kasus Uji Ubah Data Tahun Ajaran
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses ubah data tahun ajaran.
Prosedur Uji	1. Pengguna dengan hak akses sebagai admin masuk ke dalam sistem pada halaman utama admin. 2. Pengguna memilih tab menu Data Tahun Ajaran 3. Pengguna memilih data yang diubah dalam tabel data tahun ajaran 4. Pengguna mengubah data yang ada dalam *field* yang telah disediakan 5. Pengguna mengeklik tombol ubah
Hasil yang diharapkan	1. Sistem dapat menyimpan data tahun ajaran baru yang telah dimasukkan ke dalam *database* sistem. 2. Sistem dapat menampilkan data tahun ajaran terbaru pada halaman utama admin

h. Kasus Uji Kelola Hapus Data Tahun Ajaran

Kasus uji hapus data tahun ajaran menjelaskan tentang pengujian fungsionalitas proses menghapus data tahun ajaran pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai admin seperti yang ditunjukkan pada Tabel 6.10.

Tabel 6.10 Penjelasan Kasus Uji untuk Pengujian Proses Hapus Data Tahun Ajaran

Nama Kasus Uji	Kasus Uji Hapus Data Tahun Ajaran
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses hapus data tahun ajaran.
Prosedur Uji	1. Pengguna dengan hak akses sebagai admin masuk ke dalam sistem pada halaman utama admin. 2. Pengguna memilih tab menu Data Tahun Ajaran 3. Pengguna memilih data tahun ajaran yang akan dihapus kemudian mengeklik tombol Hapus untuk menghapus data tahun ajaran yang telah dipilih.
Hasil yang diharapkan	1. Sistem dapat menghapus data tahun ajaran yang tersimpan dalam *database* sistem. 2. Sistem dapat menampilkan data tahun ajaran terbaru pada halaman utama admin

i. Kasus Uji Kelola Data Jalur Masuk

Kasus uji kelola data jalur masuk menjelaskan tentang pengujian fungsionalitas proses kelola data jalur masuk pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai admin seperti yang ditunjukkan pada Tabel 6.11.

Tabel 6.11 Penjelasan Kasus Uji untuk Pengujian Proses Kelola Data Jalur Masuk

Nama Kasus Uji	Kasus Uji Kelola Data Jalur Masuk
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk mengelola data jalur masuk
Prosedur Uji	1. Pengguna dengan hak akses sebagai admin masuk ke dalam sistem pada halaman utama admin. 2. Pengguna memilih tab menu Data Jalur Masuk 3. Pengguna dihadapkan pada halaman utama admin yang memuat data jalur masuk dengan menu yang digunakan untuk mengubah dan melihat data tahun ajaran
Hasil yang diharapkan	1. Sistem dapat menampilkan data jalur masuk yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang digunakan untuk mengubah ataupun melihat data jalur masuk pada sistem.

j. Kasus Uji Kelola Ubah Data Jalur Masuk

Kasus uji ubah data jalur masuk menjelaskan tentang pengujian fungsionalitas proses mengubah data jalur masuk pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai admin seperti yang ditunjukkan pada Tabel 6.12.

Tabel 6.12 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Jalur Masuk

Nama Kasus Uji	Kasus Uji Ubah Data Jalur Masuk
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses ubah data jalur masuk.
Prosedur Uji	1. Pengguna dengan hak akses sebagai admin masuk ke dalam sistem pada halaman utama admin. 2. Pengguna memilih tab menu Data Jalur Masuk 3. Pengguna memilih data yang diubah dalam tabel data jalur masuk 4. Pengguna mengubah data yang ada dalam *field* yang telah disediakan 5. Pengguna mengeklik tombol ubah
Hasil yang diharapkan	1. Sistem dapat menyimpan data jalur masuk baru yang telah dimasukkan ke dalam *database* sistem. 2. Sistem dapat menampilkan data jalur masuk terbaru pada halaman utama admin

k. Kasus Uji Kelola Data Kelas Peminatan

Kasus uji kelola data kelas peminatan menjelaskan tentang pengujian fungsionalitas proses kelola data kelas peminatan pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai admin seperti yang ditunjukkan pada Tabel 6.13.

Tabel 6.13 Penjelasan Kasus Uji untuk Pengujian Proses Kelola Data Kelas Peminatan

Nama Kasus Uji	Kasus Uji Kelola Data Kelas Peminatan
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk mengelola data kelas peminatan
Prosedur Uji	1. Pengguna dengan hak akses sebagai admin masuk ke dalam sistem pada halaman utama admin. 2. Pengguna memilih tab menu Data Kelas Peminatan

	3. Pengguna dihadapkan pada halaman utama admin yang memuat data kelas peminatan dengan menu yang digunakan untuk mengubah dan melihat data kelas peminatan
Hasil yang diharapkan	1. Sistem dapat menampilkan data kelas peminatan yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang digunakan untuk mengubah ataupun melihat data kelas peminatan pada sistem.

l. Kasus Uji Kelola Ubah Data Kelas Peminatan

Kasus uji ubah data kelas peminatan menjelaskan tentang pengujian fungsionalitas proses mengubah data kelas peminatan pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai admin seperti yang ditunjukkan pada Tabel 6.14.

Tabel 6.14 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Kelas Peminatan

Nama Kasus Uji	Kasus Uji Ubah Data Kelas Peminatan
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses ubah data kelas peminatan
Prosedur Uji	1. Pengguna dengan hak akses sebagai admin masuk ke dalam sistem pada halaman utama admin. 2. Pengguna memilih tab menu Data Kelas Peminatan 3. Pengguna memilih data yang diubah dalam tabel data kelas peminatan 4. Pengguna mengubah data yang ada dalam *field* yang telah disediakan 5. Pengguna mengeklik tombol ubah
Hasil yang diharapkan	1. Sistem dapat menyimpan data kelas peminatan baru yang telah dimasukkan ke dalam *database* sistem. 2. Sistem dapat menampilkan data kelas peminatan terbaru pada halaman utama admin

6.1.1.2.3. Kasus Uji Halaman Utama (Kepala Sekolah / Ketua PPDB)

Kasus uji halaman utama menjelaskan tentang pengujian fungsionalitas proses masuk dalam halaman utama setelah pengguna dengan hak akses kepala sekolah/ ketua panitia berhasil melakukan proses login, ditunjukkan pada Tabel 6.15.

Tabel 6.15 Penjelasan Kasus Uji untuk Pengujian Halaman Utama (Kepala Sekolah / Ketua PPDB)

Nama Kasus Uji	Kasus Uji Halaman Utama
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk masuk ke halaman utama kepala sekolah / ketua panitia.
Prosedur Uji	1. Pengguna dengan hak akses sebagai kepala sekolah / ketua panitia masuk ke dalam sistem pada halaman utama kepala sekolah / ketua panitia. 2. Pengguna dihadapkan pada halaman utama admin dengan tab menu Data Kriteria dan Bobot Seleksi, Kriteria dan Bobot Peminatan, Data Seleksi dan Data Rekomendasi
Hasil yang diharapkan	Sistem dapat menampilkan data pada masing-masing tab menu yang dipilih

a. Kasus Uji Kelola Ubah Data Kriteria dan Bobot Seleksi

Kasus uji ubah data kritera dan bobot seleksi menjelaskan tentang pengujian fungsionalitas proses mengubah data kritera dan bobot seleksi pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai kepala sekolah / ketua panitia seperti yang ditunjukkan pada Tabel 6.16.

Tabel 6.16 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Kriteria dan Bobot Seleksi

Nama Kasus Uji	Kasus Uji Ubah Data Kriteria dan Bobot Seleksi
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses ubah data kriteria dan bobot seleksi
Prosedur Uji	1. Pengguna dengan hak akses sebagai kepala sekolah / ketua panitia masuk ke dalam sistem pada halaman utama kepala sekolah / ketua panitia. 2. Pengguna memilih tab menu Data Kriteria dan Bobot Seleksi 3. Pengguna memilih data yang diubah dalam tabel data kriteria dan bobot seleksi 4. Pengguna mengubah data yang ada dalam *field* yang telah disediakan 5. Pengguna mengeklik tombol ubah
Hasil yang diharapkan	1. Sistem dapat menyimpan data kriteria dan bobot seleksi baru yang telah dimasukkan ke dalam *database* sistem.

	2. Sistem dapat menampilkan data kriteria dan bobot seleksi terbaru pada halaman utama kepala sekolah / ketua panitia

b. Kasus Uji Ubah Kriteria dan Bobot Peminatan

Kasus uji ubah data kritera dan bobot seleksi menjelaskan tentang pengujian fungsionalitas proses mengubah data kritera dan bobot peminatan pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai kepala sekolah / ketua panitia seperti yang ditunjukkan pada Tabel 6.17.

Tabel 6.17 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Kriteria dan Bobot Peminatan

Nama Kasus Uji	Kasus Uji Ubah Data Kriteria dan Bobot Peminatan
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses ubah data kriteria dan bobot peminatan
Prosedur Uji	1. Pengguna dengan hak akses sebagai kepala sekolah / ketua panitia masuk ke dalam sistem pada halaman utama kepala sekolah / ketua panitia. 2. Pengguna memilih tab menu Data Kriteria dan Bobot Peminatan 3. Pengguna memilih data yang diubah dalam tabel data kriteria dan bobot peminatan 4. Pengguna mengubah data yang ada dalam *field* yang telah disediakan 5. Pengguna mengeklik tombol ubah
Hasil yang diharapkan	1. Sistem dapat menyimpan data kriteria dan bobot peminatan baru yang telah dimasukkan ke dalam *database* sistem. 2. Sistem dapat menampilkan data kriteria dan peminatan terbaru pada halaman utama kepala sekolah / ketua panitia

c. Kasus Uji Lihat Hasil Rekomendasi Seleksi

Kasus uji lihat hasil rekomendasi seleksi menjelaskan tentang pengujian fungsionalitas proses melihat hasil rekomendasi hasil proses seleksi pada sistem yang hanya dilakukan oleh pengguna dengan hak akses sebagai kepala sekolah / ketua panitia seperti yang ditunjukkan pada Tabel 6.18.

Tabel 6.18 Penjelasan Kasus Uji untuk Pengujian Lihat Hasil Rekomendasi Seleksi

Nama Kasus Uji	Kasus Uji Lihat Hasil Rekomendasi Seleksi
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses pencarian rekomendasi.
Prosedur Uji	1. Pengguna dengan hak akses sebagai kepala sekolah / ketua panitia masuk ke dalam sistem pada halaman utama kepala sekolah / ketua panitia. 2. Pengguna memilih tab menu Hasil Rekomendasi Seleksi 3. Pengguna memilih data yang akan dicari dari *database* dengan memilih nilai yang ada pada *combobox* 4. Pengguna mengeklik tombol hasil rekomendasi
Hasil yang diharapkan	Sistem dapat menampilkan rekomendasi calon siswa baru yang diterima / tidak diterima sesuai dengan nilai *combobox*

d. Kasus Uji Lihat Hasil Rekomendasi Peminatan

Kasus uji lihat hasil rekomendasi peminatan menjelaskan tentang pengujian fungsionalitas proses melihat hasil rekomendasi hasil proses peminatan pada sistem yang dapat dilakukan oleh pengguna dengan hak akses sebagai kepala sekolah / ketua panitia seperti yang ditunjukkan pada Tabel 6.19.

Tabel 6.19 Penjelasan Kasus Uji untuk Pengujian Lihat Hasil Rekomendasi Peminatan

Nama Kasus Uji	Kasus Uji Lihat Hasil Rekomendasi Peminatan
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses pencarian rekomendasi.
Prosedur Uji	1. Pengguna dengan hak akses sebagai kepala sekolah / ketua panitia masuk ke dalam sistem pada halaman utama kepala sekolah / ketua panitia. 2. Pengguna memilih tab menu Hasil Rekomendasi Peminatan 3. Pengguna memilih data yang akan dicari dari *database* dengan memilih nilai yang ada pada *combobox* 4. Pengguna mengeklik tombol hasil rekomendasi

Hasil yang diharapkan	Sistem dapat menampilkan rekomendasi calon siswa baru yang diterima dalam kelompok peminatan tertentu

6.1.1.2.4. Kasus Uji Halaman Utama (Sie Tes/ Guru BK)

Kasus uji halaman utama menjelaskan tentang pengujian fungsionalitas proses masuk dalam halaman utama setelah pengguna dengan hak akses sie tes / guru BK berhasil melakukan proses login, ditunjukkan pada Tabel 6.20.

Tabel 6.20 Penjelasan Kasus Uji untuk Pengujian Halaman Utama (Sie Tes / Guru BK)

Nama Kasus Uji	Kasus Uji Halaman Utama
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk masuk ke halaman utama sie tes / guru BK.
Prosedur Uji	1. Pengguna dengan hak akses sie tes / guru BK masuk ke dalam sistem pada halaman utama sie tes / guru BK. 2. Pengguna dihadapkan pada halaman utama admin dengan tab menu Data Calon siswa baru, Data Psikotes dan Data Rekomendasi
Hasil yang diharapkan	Sistem dapat menampilkan data pada masing-masing tab menu yang dipilih

a. Kasus Uji Kelola Tambah Data Calon siswa baru

Kasus uji tambah data calon siswa baru menjelaskan tentang pengujian fungsionalitas proses menambah data calon siswa baru pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai sie tes / guru BK seperti yang ditunjukkan pada Tabel 6.21.

Tabel 6.21 Penjelasan Kasus Uji untuk Pengujian Proses Tambah Data Calon siswa baru

Nama Kasus Uji	Kasus Uji Tambah Data Calon siswa baru
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses tambah data calon siswa baru.
Prosedur Uji	1. Pengguna dengan hak akses sebagai sie tes / guru BK masuk ke dalam sistem pada halaman utama sie tes / guru BK. 2. Pengguna memilih tab menu Data Calon siswa baru 3. Pengguna mengisi data calon siswa baru baru dalam *field* yang telah disediakan. 4. Pengguna mengeklik tombol tambah.

Hasil yang diharapkan	1. Sistem dapat menyimpan data calon siswa baru baru yang telah dimasukkan ke dalam *database* sistem. 2. Sistem dapat menampilkan data calon siswa baru terbaru pada halaman utama sie tes / guru BK

b. Kasus Uji Kelola Ubah Data Calon siswa baru

Kasus uji ubah data calon siswa baru menjelaskan tentang pengujian fungsionalitas proses mengubah data calon siswa baru pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai sie tes / guru BK seperti yang ditunjukkan pada Tabel 6.22.

Tabel 6.22 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Calon siswa baru

Nama Kasus Uji	Kasus Uji Ubah Data Tahun Calon siswa baru
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses ubah data calon siswa baru.
Prosedur Uji	1. Pengguna dengan hak akses sebagai sie tes / guru BK masuk ke dalam sistem pada halaman utama admin. 2. Pengguna memilih tab menu Data Calon siswa baru 3. Pengguna memilih data yang diubah dalam tabel data calon siswa baru 4. Pengguna mengubah data calon siswa baru yang ada dalam *field* yang telah disediakan 5. Pengguna mengeklik tombol ubah
Hasil yang diharapkan	1. Sistem dapat menyimpan data calon siswa baru baru yang telah dimasukkan ke dalam *database* sistem. 2. Sistem dapat menampilkan data calon siswa baru terbaru pada halaman utama sie tes / guru BK

c. Kasus Uji Kelola Hapus Data Calon siswa baru

Kasus uji hapus data calon siswa baru menjelaskan tentang pengujian fungsionalitas proses menghapus data calon siswa baru pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai sie tes / guru BK seperti yang ditunjukkan pada Tabel 6.23.

Tabel 6.23 Penjelasan Kasus Uji untuk Pengujian Proses Hapus Data Calon siswa baru

Nama Kasus Uji	Kasus Uji Hapus Data Calon siswa baru
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses hapus data calon siswa baru.
Prosedur Uji	1. Pengguna dengan hak akses sebagai sie tes / guru BK masuk ke dalam sistem pada halaman utama sie tes / guru BK. 2. Pengguna memilih tab menu Data Calon siswa baru 3. Pengguna memilih data calon siswa baru yang akan dihapus kemudian mengeklik tombol Hapus untuk menghapus data calon siswa baru yang telah dipilih.
Hasil yang diharapkan	1. Sistem dapat menghapus data calon siswa baru yang tersimpan dalam *database* sistem. 2. Sistem dapat menampilkan data calon siswa baru terbaru pada halaman utama admin

d. Kasus Uji Kelola Tambah Data Psikotes Calon siswa baru

Kasus uji tambah data calon siswa baru menjelaskan tentang pengujian fungsionalitas proses menambah data calon siswa baru pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai sie tes / guru BK seperti yang ditunjukkan pada Tabel 6.24.

Tabel 6.24 Penjelasan Kasus Uji untuk Pengujian Proses Tambah Data Calon siswa baru

Nama Kasus Uji	Kasus Uji Tambah Data Calon siswa baru
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses tambah data calon siswa baru.
Prosedur Uji	1. Pengguna dengan hak akses sebagai sie tes / guru BK masuk ke dalam sistem pada halaman utama sie tes / guru BK. 2. Pengguna memilih tab menu Data Calon siswa baru 3. Pengguna mengisi data calon siswa baru baru dalam *field* yang telah disediakan. 4. Pengguna mengeklik tombol tambah.
Hasil yang diharapkan	1. Sistem dapat menyimpan data calon siswa baru baru yang telah dimasukkan ke dalam *database* sistem.

	2. Sistem dapat menampilkan data calon siswa baru terbaru pada halaman utama sie tes / guru BK

e. Kasus Uji Kelola Ubah Psikotes Data Calon siswa baru

Kasus uji ubah data calon siswa baru menjelaskan tentang pengujian fungsionalitas proses mengubah data calon siswa baru pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai sie tes / guru BK seperti yang ditunjukkan pada Tabel 6.25.

Tabel 6.25 Penjelasan Kasus Uji untuk Pengujian Proses Ubah Data Calon siswa baru

Nama Kasus Uji	Kasus Uji Ubah Data Tahun Calon siswa baru
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses ubah data calon siswa baru.
Prosedur Uji	1. Pengguna dengan hak akses sebagai sie tes / guru BK masuk ke dalam sistem pada halaman utama admin. 2. Pengguna memilih tab menu Data Calon siswa baru 3. Pengguna memilih data yang diubah dalam tabel data calon siswa baru 4. Pengguna mengubah data calon siswa baru yang ada dalam *field* yang telah disediakan 5. Pengguna mengeklik tombol ubah
Hasil yang diharapkan	1. Sistem dapat menyimpan data calon siswa baru baru yang telah dimasukkan ke dalam *database* sistem. 2. Sistem dapat menampilkan data calon siswa baru terbaru pada halaman utama sie tes / guru BK

f. Kasus Uji Kelola Hapus Data Psikotes Calon siswa baru

Kasus uji hapus data calon siswa baru menjelaskan tentang pengujian fungsionalitas proses menghapus data calon siswa baru pada sistem yang hanya dapat dilakukan oleh pengguna dengan hak akses sebagai sie tes / guru BK seperti yang ditunjukkan pada Tabel 6.26.

Tabel 6.26 Penjelasan Kasus Uji untuk Pengujian Proses Hapus Data Psikotes Calon siswa baru

Nama Kasus Uji	Kasus Uji Hapus Data Calon siswa baru
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses hapus data calon siswa baru.
Prosedur Uji	1. Pengguna dengan hak akses sebagai sie tes / guru BK masuk ke dalam sistem pada halaman utama sie tes / guru BK. 2. Pengguna memilih tab menu Data Calon siswa baru 3. Pengguna memilih data calon siswa baru yang akan dihapus kemudian mengeklik tombol Hapus untuk menghapus data calon siswa baru yang telah dipilih.
Hasil yang diharapkan	1. Sistem dapat menghapus data calon siswa baru yang tersimpan dalam *database* sistem. 2. Sistem dapat menampilkan data calon siswa baru terbaru pada halaman utama admin

g. Kasus Uji Lihat Data Rekomendasi

Kasus uji lihat hasil rekomendasi hasil seleksi menjelaskan tentang pengujian fungsionalitas proses melihat hasil rekomendasi hasil proses perhitungan sistem yang dapat dilakukan oleh pengguna dengan hak akses sebagai sie tes / guru BK seperti yang ditunjukkan pada Tabel 6.27.

Tabel 6.27 Penjelasan Kasus Uji untuk Pengujian Lihat Hasil Rekomendasi

Nama Kasus Uji	Kasus Uji Lihat Hasil Rekomendasi
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsionalitas dari proses pencarian rekomendasi.
Prosedur Uji	1. Pengguna dengan hak akses sebagai sie tes / guru BK masuk ke dalam sistem pada halaman utama sie tes / guru BK 2. Pengguna memilih tab menu Hasil Rekomendasi 3. Pengguna memilih data yang akan dicari dengan memilih nilai yang ada pada *combobox* dan mengisi kuota yang ada 4. Pengguna mengeklik tombol hasil rekomendasi
Hasil yang diharapkan	Sistem dapat menampilkan rekomendasi calon siswa baru yang diterima

h. Kasus Uji Logout

Kasus uji logout menjelaskan tentang pengujian fungsionalitas proses logout seperti yang ditunjukkan pada Tabel 6.28.

Tabel 6.28 Penjelasan Kasus Uji untuk Pengujian Proses Logout

Nama Kasus Uji	Kasus Uji Logout
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem dapat memenuhi kebutuhan fungsional untuk kebutuhan keluar dari sistem
Prosedur Uji	1. Pengguna sudah masuk ke dalam sistem 2. Pengguna memilih menu File dan submenu Logout.
Hasil yang diharapkan	1. Sistem akan menampilkan pesan untuk memastikan pengguna benar-benar akan keluar dari sistem. 2. Sistem dapat menampilkan halaman login

6.1.1.3. Hasil Akhir

Berdasarkan kasus uji terhadap daftar kebutuhan sistem yang telah dijelaskan didapatkan hasil dari proses pengujian fungsionalitas sistem seperti yang ditunjukkan pada Tabel 6.29.

Tabel 6.29 Hasil Pengujian Fungsionalitas Sistem

No	Nama kasus uji	Hasil yang diharapkan	Hasil yang didapatkan	Kesimpulan
1	***Login***	1. Sistem dapat mengakses *database* sistem. 2. Sistem dapat melakukan pemeriksaan data *login* yang telah dimasukan oleh pengguna. 3. Sistem dapat menampilkan pesan peringatan ketika data *login* yang dimasukan tidak sesuai dengan data *login* yang tersimpan dalam *database* sistem.	1. Sistem dapat mengakses *database* sistem. 2. Sistem dapat melakukan pemeriksaan data *login* yang telah dimasukan oleh pengguna. 3. Sistem dapat menampilkan pesan peringatan ketika data *login* yang dimasukan tidak sesuai dengan data *login* yang tersimpan dalam *database* sistem.	Sukses

		4. Sistem dapat menampilkan halaman utama sesuai dengan hak akses masing-masing pengguna ketika data *login* sesuai dengan data pada *database*.	4. Sistem dapat menampilkan halaman utama sesuai dengan hak akses masing-masing pengguna ketika data *login* sesuai dengan data pada *database*.	
2	**Kelola Data Pengguna**	1. Sistem dapat menampilkan data pengguna yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang digunakan untuk menambah, mengubah, menghapus, ataupun melihat data pengguna pada sistem.	1. Sistem dapat menampilkan data pengguna yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang digunakan untuk menambah, mengubah, menghapus, ataupun melihat data pengguna pada sistem.	Sukses
3	**Tambah Data Pengguna**	1. Sistem dapat menyimpan data pengguna baru yang dimasukkan ke dalam *database* sistem. 2. Sistem dapat menampilkan data pengguna pada halaman utama admin	1. Sistem dapat menyimpan data pengguna baru yang dimasukkan ke dalam *database* sistem. 2. Sistem dapat menampilkan data pengguna pada halaman utama admin	Sukses
4	**Ubah Data Pengguna**	1. Sistem dapat mengubah data pengguna yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data pengguna terbaru pada halaman utama admin	1. Sistem dapat mengubah data pengguna yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data pengguna terbaru pada halaman utama admin	Sukses

5	**Hapus Data Pengguna**	1. Sistem dapat menghapus data pengguna yang tersimpan dalam *database* sistem. 2. Sistem dapat menampilkan data pengguna terbaru pada halaman utama admin	1. Sistem dapat menghapus data pengguna yang tersimpan dalam *database* sistem. 2. Sistem dapat menampilkan data pengguna terbaru pada halaman utama admin	Sukses
6	**Kelola Data Tahun Ajaran**	1. Sistem dapat menampilkan data tahun ajaran yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang digunakan untuk menambah, mengubah, menghapus, ataupun melihat data tahun ajaran pada sistem.	1. Sistem dapat menampilkan data tahun ajaran yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang digunakan untuk menambah, mengubah, menghapus, ataupun melihat data tahun ajaran pada sistem.	Sukses
7	**Tambah Data Tahun Ajaran**	1. Sistem dapat menyimpan data tahun ajaran baru yang dimasukkan ke dalam *database* sistem. 2. Sistem dapat menampilkan data tahun ajaran pada halaman utama admin	1. Sistem dapat menyimpan data tahun ajaran baru yang dimasukkan ke dalam *database* sistem. 2. Sistem dapat menampilkan data tahun ajaran pada halaman utama admin	Sukses
8	**Ubah Data Tahun Ajaran**	1. Sistem dapat mengubah data tahun ajaran yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data tahun ajaran terbaru pada halaman utama admin	1. Sistem dapat mengubah data tahun ajaran yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data tahun ajaran terbaru pada halaman utama admin	Sukses

9	**Hapus Data Tahun Ajaran**	1. Sistem dapat menghapus data tahun ajaran yang tersimpan dalam *database* sistem. 2. Sistem dapat menampilkan data tahun ajaran terbaru pada halaman utama admin	1. Sistem dapat menghapus data tahun ajaran yang tersimpan dalam *database* sistem. 2. Sistem dapat menampilkan data tahun ajaran terbaru pada halaman utama admin	Sukses
10	**Kelola Data Jalur Masuk**	1. Sistem dapat menampilkan data jalur masuk yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang mengubah ataupun melihat data jalur masuk pada sistem.	1. Sistem dapat menampilkan data jalur masuk yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang mengubah ataupun melihat data jalur masuk pada sistem.	Sukses
11	**Ubah Data Jalur Masuk**	1. Sistem dapat mengubah data jalur masuk yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data jalur masuk terbaru pada halaman utama admin	1. Sistem dapat mengubah data jalur masuk yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data jalur masuk terbaru pada halaman utama admin	Sukses
12	**Kelola Data Kelas Peminatan**	1. Sistem dapat menampilkan data kelas peminatan yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang digunakan untuk mengubah ataupun melihat data kelas	1. Sistem dapat menampilkan data kelas peminatan yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang digunakan untuk mengubah ataupun melihat data kelas	Sukses

		peminatan pada sistem.	peminatan pada sistem.	
13	**Ubah Data Kelas Peminatan**	1. Sistem dapat mengubah data kelas peminatan yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data kelas peminatan terbaru pada halaman utama admin	1. Sistem dapat mengubah data tahun ajaran yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data tahun ajaran terbaru pada halaman utama admin	Sukses
14	**Kelola Data Kriteria dan Bobot Seleksi**	1. Sistem dapat menampilkan data kriteria dan bobot seleksi yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang digunakan untuk mengubah ataupun melihat data kriteria dan bobot seleksi pada sistem.	1. Sistem dapat menampilkan data kriteria dan bobot seleksi yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang digunakan untuk mengubah ataupun melihat data kriteria dan bobot seleksi pada sistem.	Sukses
15	**Ubah Data Kriteria dan Bobot Seleksi**	1. Sistem dapat mengubah data kriteria dan bobot seleksi yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data kriteria dan bobot seleksi terbaru pada halaman utama kepala sekolah / ketua panitia	1. Sistem dapat mengubah data kriteria dan bobot seleksi yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data kriteria dan bobot seleksi terbaru pada halaman utama kepala sekolah / ketua panitia	Sukses
16	**Kelola Data Kriteria dan Bobot Peminatan**	1. Sistem dapat menampilkan data kriteria dan bobot peminatan yang	1. Sistem dapat menampilkan data kriteria dan bobot peminatan yang	Sukses

		tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang digunakan untuk mengubah ataupun melihat data kriteria dan bobot peminatan pada sistem.	tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan menu yang digunakan untuk mengubah ataupun melihat data kriteria dan bobot peminatan pada sistem.	
17	**Ubah Data Kriteria dan Bobot Peminatan**	1. Sistem dapat mengubah data kriteria dan bobot peminatan yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data kriteria dan bobot peminatan terbaru pada halaman utama kepala sekolah / ketua panitia	1. Sistem dapat mengubah data kriteria dan bobot peminatan yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data kriteria dan bobot peminatan terbaru pada halaman utama kepala sekolah / ketua panitia	Sukses
18	**Lihat Hasil Rekomendasi Seleksi**	Sistem dapat menampilkan rekomendasi peserta didik yang diterima / tidak diterima yang telah disimpan dalam *database*	Sistem dapat menampilkan rekomendasi peserta didik yang diterima / tidak diterima yang telah disimpan dalam *database*	Sukses
19	**Lihat Hasil Rekomendasi Peminatan**	Sistem dapat menampilkan rekomendasi peserta didik yang diterima dalam kelompok peminatan tertentu yang telah disimpan dalam *database*	Sistem dapat menampilkan rekomendasi peserta didik yang diterima / tidak diterima yang telah disimpan dalam *database*	Sukses
20	**Kelola Data Peserta Didik**	1. Sistem dapat menampilkan data peserta didik yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan	1. Sistem dapat menampilkan data peserta didik yang tersimpan dalam *database* sistem. 2. Sistem dapat menyediakan	Sukses

		menu yang digunakan untuk menambah, mengubah, menghapus, ataupun melihat data peserta didik pada sistem.	menu yang digunakan untuk menambah, mengubah, menghapus, ataupun melihat data peserta didik pada sistem.	
21	**Tambah Data Peserta Didik**	1. Sistem dapat menyimpan data peserta didik baru yang dimasukkan ke dalam *database* sistem. 2. Sistem dapat menampilkan data peserta didik pada halaman utama sie tes / guru BK	1. Sistem dapat menyimpan data peserta didik baru yang dimasukkan ke dalam *database* sistem. 2. Sistem dapat menampilkan data peserta didik pada halaman utama sie tes / guru BK	Sukses
22	**Ubah Data Peserta Didik**	1. Sistem dapat mengubah data pesrta didik yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data peserta didik terbaru pada halaman utama sie tes / guru BK	1. Sistem dapat mengubah data peserta didik yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data peserta didik terbaru pada halaman utama sie tes / guru BK	Sukses
23	**Hapus Data Peserta Didik**	1. Sistem dapat menghapus data peserta didik yang tersimpan dalam *database* sistem. 2. Sistem dapat menampilkan data peserta didik terbaru pada halaman utama sie tes / guru BK	1. Sistem dapat menghapus data peserta didik yang tersimpan dalam *database* sistem. 2. Sistem dapat menampilkan data peserta didik terbaru pada halaman utama sie tes / guru BK	Sukses
24	**Tambah Psikotest Data Peserta Didik**	1. Sistem dapat menyimpan data psikotest peserta didik baru yang dimasukkan ke	1. Sistem dapat menyimpan data psikotest peserta didik baru yang dimasukkan ke	Sukses

		dalam *database* sistem. 2. Sistem dapat menampilkan data peserta didik pada halaman utama sie tes / guru BK	dalam *database* sistem. 2. Sistem dapat menampilkan data psikotest peserta didik pada halaman utama sie tes / guru BK	
25	**Ubah Data psiktoest Peserta Didik**	1. Sistem dapat mengubah data psikotest pesrta didik yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data psikotest peserta didik terbaru pada halaman utama sie tes / guru BK	1. Sistem dapat mengubah data peserta didik yang ada dalam *database* sistem. 2. Sistem dapat menampilkan data psikotest peserta didik terbaru pada halaman utama sie tes / guru BK	Sukses
26	**Hapus Data Psiktoest Peserta Didik**	1. Sistem dapat menghapus data psikotest peserta didik yang tersimpan dalam *database* sistem. 2. Sistem dapat menampilkan data psikotest peserta didik terbaru pada halaman utama sie tes / guru BK	1. Sistem dapat menghapus data psikotest peserta didik yang tersimpan dalam *database* sistem. 2. Sistem dapat menampilkan data psikotest peserta didik terbaru pada halaman utama sie tes / guru BK	Sukses
27	**Lihat Hasil Rekomendasi**	Sistem dapat menampilkan rekomendasi peserta didik yang diterima dalam kelompok peminatan tertentu yang telah disimpan dalam *database*	Sistem dapat menampilkan rekomendasi peserta didik yang diterima dan kelas peminatannya / tidak diterima yang telah disimpan dalam *database*	Sukses
28	***Logout***	1. Sistem akan menampilkan pesan untuk menanyakan pada pengguna apakah akan keluar dari sistem.	1. Sistem akan menampilkan pesan untuk menanyakan pada pengguna apakah akan keluar dari sistem.	Sukses

		2. Sistem dapat menampilkan halaman login	2. Sistem dapat menampilkan halaman login	

6.2 Pengujian Akurasi Sistem

Pengujian tingkat akurasi dilakukan untuk mengetahui performa dari sistem pendukung keputusan seleksi penerimaan siswa baru menggunakan metode ELECTRE - *Weighted Product*. Pengujian akurasi dilakukan dengan cara membandingkan data hasil keputusan sistem dengan data hasil perhitungan manual yang dilakukan oleh pihak SMA BSS.

6.2.1 Skenario Pengujian Akurasi

Sub bab berikut akan menjelaskan tentang tujuan, prosedur, serta hasil akhir yang didapatkan dari scenario pengujian kedua yang merupakan scenario pengujian tingkat akurasi.

6.2.1.1 Tujuan

Tujian dari pengujian akurasi adalah untuk mengetahui seberapa banyak kesesuaian data antara hasil keputusan sistem dengan hasil keputusan dari perhitugan manual pihak SMA BSS. Data yang digunakan sebagai data uji sebanyak 25 data yang kemudian data tersebut akan dievaluasi dengan hasil keputusan sistem yang menggunakan metode ELECTRE - *Weighted Product*.

6.2.1.1 Prosedur

Prosedur pengujian tingkat akurasi dilakukan dengan cara mencocokkan perhitungan manual pihak SMA BSS dengan hasil keputusan sistem. Dalam pengujian ini digunakan sejumlah data calon siswa dan pembobotan yang digunakan sebagai acuan didapatkan dari hasil wawancara dengan pihak SMA BSS. Pengujian tingkat akurasi digunakan terhadap dua hasil yang didapatkan dari perhitungan sistem yaitu hasil perhitungan seleksi dan peminatan. Adapaun skenario yang digunakan untuk mengetahui tingkat akurasi dari perhitungan sistem, yaitu:

6.2.1.1.1 Pengujian akurasi dengan menggunakan 25 data

Skenario pengujian yang pertama digunakan data calon siswa sebanyak 25 data, dengan menggunakan nilai pembobotan yang didapatkan dari hasil wawancara dengan pihak SMA BSS. Berikut ini skenario pengujian akurasi dengan menggunakan 25 data calon siswa

a. Perhitungan sistem dengan menggunakan nilai batas > 0

Pada uji eksperimen ini digunakan nilai batas > 0 terhadap jumlah nilai 1 yang terdapat dalam keseluruhan nilai matriks *e* pada metode ELECTRE yang digunakan dalam tahapan proses seleksi. Hasil yang didapatkan dari uji eksperimen ditunjukkan pada Tabel 6.30.

Tabel 6.30 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 0

Alternatif siswa ke-	Perhitngan Sistem		Data Aktual		Keterangan
	Seleksi	Kelompok Peminatan	Seleksi	Kelompok Peminatan	
1	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
2	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
3	Diterima	IPS	Tidak Diterima	-	Tidak Sesuai
4	Diterima	IPS	Diterima	IPA	Tidak Sesuai
5	Diterima	IPS	Tidak Diterima	-	Tidak Sesuai
6	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
7	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
8	Diterima	IPA	Diterima	IPA	Sesuai
9	Diterima	IPS	Diterima	IPS	Sesuai
10	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
11	Diterima	IPS	Diterima	IPA	Tidak Sesuai
12	Diterima	IPA	Diterima	IPA	Sesuai
13	Diterima	IPA/IPS	Diterima	IPA	Sesuai
14	Diterima	IPA	Diterima	IPA	Sesuai
15	Diterima	IPA	Diterima	IPA	Sesuai

16	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
17	Diterima	IPA	Diterima	IPA	Sesuai
18	Diterima	IPA	Diterima	IPA	Sesuai
19	Diterima	IPA	Diterima	IPA	Sesuai
20	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
21	Diterima	IPS	Tidak Diterima	-	Tidak Sesuai
22	Diterima	IPA	Diterima	IPA	Sesuai
23	Diterima	IPS	Tidak Diterima	-	Tidak Sesuai
24	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
25	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai

Hasil pengujian dengan menggnakan nilai batas > 0 dari 25 data kasus didapatkan hasil 11 data kasus sesuai dengan data aktual dan 14 data kasus yang tidak sesuai dengan data aktual. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25) maka didapatkan hasil sebagai berikut.

$$Akurasi = \frac{11}{25} \times 100\% = 44\%$$

b. Perhitungan sistem dengan menggunakan nilai batas > 3

Pada uji eksperimen ini digunakan nilai batas > 3 terhadap jumlah nilai 1 yang terdapat dalam keseluruhan nilai matriks *e* pada metode ELECTRE yang digunakan dalam tahapan proses seleksi. Hasil yang didapatkan dari uji eksperimen ditunjukkan pada Tabel 6.31.

Tabel 6.31 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 3

Alternatif siswa **ke-**	**Perhitngan Sistem**		**Data Aktual**		**Keterangan**
	Seleksi	**Kelompok Peminatan**	**Seleksi**	**Kelompok Peminatan**	
1	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
2	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
3	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
4	Diterima	IPS	Diterima	IPA	Sesuai

5	Diterima	IPS	Tidak Diterima	-	Tidak Sesuai
6	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
7	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
8	Diterima	IPA	Diterima	IPA	Sesuai
9	Diterima	IPS	Diterima	IPS	Sesuai
10	Tida kDiterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
11	Diterima	IPS	Diterima	IPA	Tidak Sesuai
12	Diterima	IPA	Diterima	IPA	Sesuai
13	Diterima	IPA/IPS	Diterima	IPA	Sesuai
14	Diterima	IPA	Diterima	IPA	Sesuai
15	Diterima	IPA	Diterima	IPA	Sesuai
16	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
17	Diterima	IPA	Diterima	IPA	Sesuai
18	Diterima	IPA	Diterima	IPA	Sesuai
19	Diterima	IPA	Diterima	IPA	Sesuai
20	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
21	Diterima	IPS	Tidak Diterima	-	Tidak Sesuai
22	Diterima	IPA	Diterima	IPA	Sesuai
23	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Tidak Sesuai
24	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
25	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai

Hasil pengujian dengan menggnakan nilai batas > 3 dari 25 data kasus didapatkan hasil 16 data kasus sesuai dengan data aktual dan 9 data kasus yang tidak sesuai dengan data aktual. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25) maka didapatkan hasil sebagai berikut.

$$Akurasi = \frac{16}{25} \times 100\% = 64\%$$

c. Perhitungan sistem dengan menggunakan nilai batas > 5

Pada uji eksperimen ini digunakan nilai batas > 5 terhadap jumlah nilai 1 yang terdapat dalam keseluruhan nilai matriks *e* pada metode ELECTRE yang digunakan dalam tahapan proses seleksi. Hasil yang didapatkan dari uji eksperimen ditunjukkan pada Tabel 6.32.

Tabel 6.32 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 5

Alternatif siswa ke-	Perhitngan Sistem		Data Aktual		Keterangan
	Seleksi	Kelompok Peminatan	Seleksi	Kelompok Peminatan	
1	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
2	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
3	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
4	Diterima	IPS	Diterima	IPA	Tidak Sesuai
5	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
6	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
7	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
8	Diterima	IPA	Diterima	IPA	Sesuai
9	Diterima	IPS	Diterima	IPS	Sesuai
10	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
11	Diterima	IPS	Diterima	IPA	Tidak Sesuai
12	Diterima	IPA	Diterima	IPA	Sesuai
13	Diterima	IPA/IPS	Diterima	IPA	Sesuai
14	Diterima	IPA	Diterima	IPA	Sesuai
15	Diterima	IPA	Diterima	IPA	Sesuai
16	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
17	Diterima	IPA	Diterima	IPA	Sesuai
18	Diterima	IPA	Diterima	IPA	Sesuai
19	Diterima	IPA	Diterima	IPA	Sesuai
20	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
21	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
22	Diterima	IPA	Diterima	IPA	Sesuai
23	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
24	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
25	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai

Hasil pengujian dengan menggnakan nilai batas > 5 dari 25 data kasus didapatkan hasil 19 data kasus sesuai dengan data aktual dan 6 data kasus yang tidak sesuai dengan data aktual. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25) maka didapatkan hasil sebagai berikut.

$$Akurasi = \frac{19}{25} \times 100\% = 76\%$$

d. Perhitungan sistem dengan menggunakan nilai batas > 7

Pada uji eksperimen ini digunakan nilai batas > 7 terhadap jumlah nilai 1 yang terdapat dalam keseluruhan nilai matriks *e* pada metode ELECTRE yang digunakan dalam tahapan proses seleksi. Hasil yang didapatkan dari uji eksperimen ditunjukkan pada Tabel 6.33.

Tabel 6.33 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 7

Alternatif siswa **ke-**	**Perhitngan Sistem**		**Data Aktual**		**Keterangan**
	Seleksi	**Kelompok Peminatan**	**Seleksi**	**Kelompok Peminatan**	
1	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Tidak Sesuai
2	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
3	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
4	Diterima	IPS	Diterima	IPA	Tidak Sesuai
5	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
6	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
7	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
8	Diterima	IPA	Diterima	IPA	Sesuai
9	Diterima	IPS	Diterima	IPS	Sesuai
10	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
11	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
12	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Sesuai
13	Diterima	IPA/IPS	Diterima	IPA	Sesuai
14	Diterima	IPA	Diterima	IPA	Sesuai
15	Diterima	IPA	Diterima	IPA	Sesuai
16	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
17	Diterima	IPA	Diterima	IPA	Sesuai
18	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Sesuai
19	Diterima	IPA	Diterima	IPA	Sesuai
20	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
21	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
22	Diterima	IPA	Diterima	IPA	Sesuai
23	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
24	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
25	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai

Hasil pengujian dengan menggnakan nilai batas > 7 dari 25 data kasus didapatkan hasil 19 data kasus sesuai dengan data aktual dan 6 data kasus yang tidak sesuai dengan data aktual. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25) maka didapatkan hasil sebagai berikut.

$$Akurasi = \frac{19}{25} \times 100\% = 76\%$$

e. Perhitungan sistem dengan menggunakan nilai batas > 8

Pada uji eksperimen ini digunakan nilai batas > 8 terhadap jumlah nilai 1 yang terdapat dalam keseluruhan nilai matriks *e* pada metode ELECTRE yang digunakan dalam tahapan proses seleksi. Hasil yang didapatkan dari uji eksperimen ditunjukkan pada Tabel 6.34.

Tabel 6.34 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 8

Alternatif siswa **ke-**	**Perhitngan Sistem**		**Data Aktual**		**Keterangan**
	Seleksi	**Kelompok Peminatan**	**Seleksi**	**Kelompok Peminatan**	
1	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
2	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
3	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
4	Diterima	IPS	Diterima	IPA	Tidak Sesuai
5	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
6	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
7	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
8	Diterima	IPA	Diterima	IPA	Sesuai
9	Diterima	IPS	Diterima	IPS	Sesuai
10	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
11	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
12	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
13	Diterima	IPA/IPS	Diterima	IPA	Sesuai
14	Diterima	IPA	Diterima	IPA	Sesuai
15	Diterima	IPA	Diterima	IPA	Sesuai
16	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
17	Diterima	IPA	Diterima	IPA	Sesuai
18	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
19	Diterima	IPA	Diterima	IPA	Sesuai

20	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
21	TidakDiterima	TidakDihitung	Tidak Diterima	-	Sesuai
22	Diterima	IPA	Diterima	IPA	Sesuai
23	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
24	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
25	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai

Hasil pengujian dengan menggnakan nilai batas > 8 dari 25 data kasus didapatkan hasil 20 data kasus sesuai dengan data aktual dan 5 data kasus yang tidak sesuai dengan data aktual. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25)) maka didapatkan hasil sebagai berikut.

$$Akurasi = \frac{20}{25} \times 100\% = 80\%$$

f. Perhitungan sistem dengan menggunakan nilai batas > 9

Pada uji eksperimen ini digunakan nilai batas > 9 terhadap jumlah nilai 1 yang terdapat dalam keseluruhan nilai matriks *e* pada metode ELECTRE yang digunakan dalam tahapan proses seleksi. Hasil yang didapatkan dari uji eksperimen ditunjukkan pada Tabel 6.35.

Tabel 6.35 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 9

Alternatif siswa **ke-**	**Perhitngan Sistem**		**Data Aktual**		**Keterangan**
	Seleksi	**Kelompok Peminatan**	**Seleksi**	**Kelompok Peminatan**	
1	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
2	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
3	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
4	Diterima	IPS	Diterima	IPA	Tidak Sesuai
5	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
6	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
7	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
8	Diterima	IPA	Diterima	IPA	Sesuai
9	Diterima	IPS	Diterima	IPS	Sesuai
10	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
11	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
12	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
13	Diterima	IPA/IPS	Diterima	IPA	Sesuai
14	Diterima	IPA	Diterima	IPA	Sesuai
15	Diterima	IPA	Diterima	IPA	Sesuai
16	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
17	Diterima	IPA	Diterima	IPA	Sesuai
18	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
19	Diterima	IPA	Diterima	IPA	Sesuai

20	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
21	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
22	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
23	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
24	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
25	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai

Hasil pengujian dengan menggnakan nilai batas > 9 dari 25 data kasus didapatkan hasil 19 data kasus sesuai dengan data aktual dan 6 data kasus yang tidak sesuai dengan data aktual. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25) maka didapatkan hasil sebagai berikut.

$$Akurasi = \frac{19}{25} \times 100\% = 76\%$$

g. Perhitungan sistem dengan menggunakan nilai batas > 11

Pada uji eksperimen ini digunakan nilai batas > 11 terhadap jumlah nilai 1 yang terdapat dalam keseluruhan nilai matriks *e* pada metode ELECTRE yang digunakan dalam tahapan proses seleksi. Hasil yang didapatkan dari uji eksperimen ditunjukkan pada Tabel 6.36.

Tabel 6.36 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 11

Alternatif siswa **ke-**	**Perhitngan Sistem**		**Data Aktual**		**Keterangan**
	Seleksi	**Kelompok Peminatan**	**Seleksi**	**Kelompok Peminatan**	
1	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
2	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
3	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
4	Diterima	IPS	Diterima	IPA	Tidak Sesuai
5	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
6	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
7	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
8	Diterima	IPA	Diterima	IPA	Sesuai
9	Diterima	IPS	Diterima	IPS	Sesuai
10	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
11	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
12	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
13	Diterima	IPA/IPS	Diterima	IPA	Sesuai
14	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
15	Diterima	IPA	Diterima	IPA	Sesuai
16	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
17	Diterima	IPA	Diterima	IPA	Sesuai
18	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
19	Diterima	IPA	Diterima	IPA	Sesuai
20	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai
21	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
22	Tidak Diterima	Tidak Dihitung	Diterima	IPA	Tidak Sesuai

23	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
24	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai
25	Tidak Diterima	Tidak Dihitung	Tidak Diterima	-	Sesuai

Hasil pengujian dengan menggnakan nilai batas > 11 dari 25 data kasus didapatkan hasil 18 data kasus sesuai dengan data aktual dan 7 data kasus yang tidak sesuai dengan data aktual. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25) maka didapatkan hasil sebagai berikut.

$$Akurasi = \frac{18}{25} \times 100\% = 72\%$$

6.2.1.1.2 Pengujian akurasi dengan menggunakan 32 data

Skenario pengujian yang pertama digunakan data calon siswa sebanyak 32 data, dengan menggunakan nilai pembobotan yang didapatkan dari hasil wawancara dengan pihak SMA BSS. Berikut ini skenario pengujian akurasi dengan menggunakan 32 data calon siswa.

a. Perhitungan sistem dengan menggunakan nilai batas > 0

Pada uji eksperimen ini digunakan nilai batas > 0 terhadap jumlah nilai 1 yang terdapat dalam keseluruhan nilai matriks *e* pada metode ELECTRE yang digunakan dalam tahapan proses seleksi. Hasil yang didapatkan dari uji eksperimen ditunjukkan pada Tabel 6.37.

Tabel 6.37 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 0

Nomor	Data Aktual		Sistem		Keterangan
1	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
2	Tidak Diterima	-	Tidak Diterima	-	Sesuai
3	Tidak Diterima	-	Diterima	IPS	Tidak Sesuai
4	Diterima	IPA	Diterima	IPS	Tidak Sesuai
5	Tidak Diterima	-	Diterima	IPS	Tidak Sesuai
6	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
7	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
8	Diterima	IPA	Diterima	IPA	Sesuai
9	Diterima	IPA/IPS	Diterima	IPS	Sesuai
10	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
11	Diterima	IPA	Diterima	IPS	Tidak Sesuai
12	Diterima	IPA	Diterima	IPA	Sesuai
13	Diterima	IPA	Diterima	IPA /IPS	Sesuai
14	Diterima	IPA	Diterima	IPA	Sesuai
15	Diterima	IPA	Diterima	IPA	Sesuai
16	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
17	Diterima	IPA	Diterima	IPA	Sesuai

18	Diterima	IPA	Diterima	IPA	Sesuai
19	Diterima	IPA	Diterima	IPA	Sesuai
20	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
21	Tidak Diterima	-	Diterima	IPS	Tidak Sesuai
22	Diterima	IPA	Diterima	IPA	Sesuai
23	Tidak Diterima	-	Diterima	IPS	Tidak Sesuai
24	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
25	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
26	Diterima	IPA	Diterima	IPA	Sesuai
27	Diterima	IPA	Diterima	IPA	Sesuai
28	Diterima	IPA	Diterima	IPA	Sesuai
29	Diterima	IPA	Diterima	IPA	Sesuai
30	Diterima	IPA	Diterima	IPA	Sesuai
31	Diterima	IPA	Diterima	IPA	Sesuai
32	Diterima	IPA/IPS	Diterima	IPA	Sesuai

Hasil pengujian dengan menggnakan nilai batas > 0 dari 32 data kasus didapatkan hasil 19 data kasus sesuai dengan data aktual dan 13 data kasus yang tidak sesuai dengan data aktual. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25) maka didapatkan hasil sebagai berikut.

$$\text{Akurasi} = \frac{18}{32} * 100\% \ = 56.25\%$$

b. Skenario sistem dengan menggunakan nilai batas > 3

Pada uji eksperimen ini digunakan nilai batas > 3 terhadap jumlah nilai 1 yang terdapat dalam keseluruhan nilai matriks *e* pada metode ELECTRE yang digunakan dalam tahapan proses seleksi. Hasil yang didapatkan dari uji eksperimen ditunjukkan pada Tabel 6.38.

Tabel 6.38 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 3

Nomor	Data Aktual		Hasil Seleksi Sistem		Keterangan
1	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
2	Tidak Diterima	-	Tidak Diterima	-	Sesuai
3	Tidak Diterima	-	Tidak Diterima	-	Sesuai
4	Diterima	IPA	Diterima	IPS	Tidak Sesuai
5	Tidak Diterima	-	Diterima	IPS	Tidak Sesuai
6	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
7	Tidak Diterima	-	Tidak Diterima	-	Sesuai
8	Diterima	IPA	Diterima	IPA	Sesuai
9	Diterima	IPA/IPS	Diterima	IPS	Sesuai
10	Tidak Diterima	-	Tidak Diterma	-	Sesuai
11	Diterima	IPA	Diterima	IPS	Tidak Sesuai
12	Diterima	IPA	Diterima	IPA	Sesuai

13	Diterima	IPA	Diterima	IPA/IPS	Sesuai
14	Diterima	IPA	Diterima	IPA	Sesuai
15	Diterima	IPA	Diterima	IPA	Sesuai
16	Tidak Diterima	-	Tidak Diterima	-	Sesuai
17	Diterima	IPA	Diterima	IPA	Sesuai
18	Diterima	IPA	Diterima	IPA	Sesuai
19	Diterima	IPA	Diterima	IPA	Sesuai
20	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
21	Tidak Diterima	-	Diterima	IPS	Tidak Sesuai
22	Diterima	IPA	Diterima	IPA	Sesuai
23	Tidak Diterima	-	Tidak Diterima	-	Sesuai
24	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
25	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
26	Diterima	IPA	Diterima	IPA	Sesuai
27	Diterima	IPA	Diterima	IPA	Sesuai
28	Diterima	IPA	Diterima	IPA	Sesuai
29	Diterima	IPA	Diterima	IPA	Sesuai
30	Diterima	IPA	Diterima	IPA	Sesuai
31	Diterima	IPA	Diterima	IPA	Sesuai
32	Diterima	IPA/IPS	Diterima	IPA	Sesuai

Hasil pengujian dengan menggnakan nilai batas > 3 dari 32 data kasus didapatkan hasil 23 data kasus sesuai dengan data aktual dan 9 data kasus yang tidak sesuai dengan data aktual. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25) maka didapatkan hasil sebagai berikut.

$$\text{Akurasi} = \frac{23}{32} \times 100\% = 71.8\%$$

c. Skenario sistem dengan menggunakan nilai batas > 5

Pada uji eksperimen ini digunakan nilai batas > 35 terhadap jumlah nilai 1 yang terdapat dalam keseluruhan nilai matriks *e* pada metode ELECTRE yang digunakan dalam tahapan proses seleksi. Hasil yang didapatkan dari uji eksperimen ditunjukkan pada Tabel 6.39.

Tabel 6.39 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 5

Nomor	Data Aktual		Hasil Seleksi Sistem		Keterangan
1	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
2	Tidak Diterima	-	Tidak Diterima	-	Sesuai
3	Tidak Diterima	-	Tidak Diterima	-	Sesuai
4	Diterima	IPA	Diterima	IPS	Tidak Sesuai
5	Tidak Diterima	-	Diterima	IPS	Tidak Sesuai
6	Tidak Diterima	-	Tidak Diterima	-	Sesuai

7	Tidak Diterima	-	Tidak Diterima	-	Sesuai
8	Diterima	IPA	Diterima	IPA	Sesuai
9	Diterima	IPA/IPS	Diterima	IPS	Sesuai
10	Tidak Diterima	-	Tidak Diterima	-	Sesuai
11	Diterima	IPA	Diterima	IPS	Tidak Sesuai
12	Diterima	IPA	Diterima	IPA	Sesuai
13	Diterima	IPA	Diterima	IPA/IPS	Sesuai
14	Diterima	IPA	Diterima	IPA	Sesuai
15	Diterima	IPA	Diterima	IPA	Sesuai
16	Tidak Diterima	-	Tidak Diterima	-	Sesuai
17	Diterima	IPA	Diterima	IPA	Sesuai
18	Diterima	IPA	Diterima	IPA	Sesuai
19	Diterima	IPA	Diterima	IPA	Sesuai
20	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
21	Tidak Diterima	-	Tidak Diterima	-	Sesuai
22	Diterima	IPA	Diterima	IPA	Tidak Sesuai
23	Tidak Diterima	-	Tidak Diterima	-	Sesuai
24	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
25	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
26	Diterima	IPA	Diterima	IPA	Sesuai
27	Diterima	IPA	Diterima	IPA	Sesuai
28	Diterima	IPA	Diterima	IPA	Sesuia
29	Diterima	IPA	Diterima	IPA	Sesuai
30	Diterima	IPA	Diterima	IPA	Sesuai
31	Diterima	IPA	Diterima	IPA	Sesuai
32	Diterima	IPA/IPS	Diterima	IPA	Sesuai

Hasil pengujian dengan menggnakan nilai batas > 5 dari 32 data kasus didapatkan hasil 27 data kasus sesuai dengan data aktual dan 5 data kasus yang tidak sesuai dengan data aktual. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25) maka didapatkan hasil sebagai berikut.

$$\text{Akurasi} = \frac{25}{32} * 100\% = 78.125\%$$

d. Skenario sistem dengan menggunakan nilai batas > 7

Pada uji eksperimen ini digunakan nilai batas > 7 terhadap jumlah nilai 1 yang terdapat dalam keseluruhan nilai matriks *e* pada metode ELECTRE yang digunakan dalam tahapan proses seleksi. Hasil yang didapatkan dari uji eksperimen ditunjukkan pada Tabel 6.40.

Tabel 6.40 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 7

Nomor	Data Aktual		Hasil Seleksi Sistem		Keterangan
1	Tidak Diterima	-	Tidak Diterima		Sesuai
2	Tidak Diterima	-	Tidak Diterima	-	Sesuai
3	Tidak Diterima	-	Tidak Diterima	-	Sesuai
4	Diterima	IPA	Diterima	IPS	Tidak Sesuai
5	Tidak Diterima	-	Tidak Diterima		Sesuai
6	Tidak Diterima	-	Tidak Diterima		Sesuai
7	Tidak Diterima	-	Tidak Diterima	-	Sesuai
8	Diterima	IPA	Diterima	IPA	Sesuai
9	Diterima	IPA/IPS	Diterima	IPS	Sesuai
10	Tidak Diterima	-	Tidak Diterma	-	Sesuai
11	Diterima	IPA	Diterima	IPS	Tidak Sesuai
12	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
13	Diterima	IPA	Diterima	IPA/IPS	Sesuai
14	Diterima	IPA	Diterima	IPA	Sesuai
15	Diterima	IPA	Diterima	IPA	Sesuai
16	Tidak Diterima	-	Tidak Diterima	-	Sesuai
17	Diterima	IPA	Diterima	IPA	Sesuai
18	Diterima	IPA	Diterima	IPA	Sesuai
19	Diterima	IPA	Diterima	IPA	Sesuai
20	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
21	Tidak Diterima	-	Tidak Diterima	-	Tidak Sesuai
22	Diterima	IPA	Diterima	IPA	Sesuai
23	Tidak Diterima	-	Tidak Diterima	-	Sesuai
24	Tidak Diterima	-	Tidak Diterima	-	Sesuai
25	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
26	Diterima	IPA	Diterima	IPA	Sesuai
27	Diterima	IPA	Diterima	IPA	Sesuai
28	Diterima	IPA	Diterima	IPA	Sesuai
29	Diterima	IPA	Diterima	IPA	Sesuai
30	Diterima	IPA	Diterima	IPA	Sesuai
31	Diterima	IPA	Diterima	IPA	Sesuai
32	Diterima	IPA/IPS	Diterima	IPA	Sesuai

Hasil pengujian dengan menggnakan nilai batas > 7 dari 32 data kasus didapatkan hasil 27 data kasus sesuai dengan data aktual dan 5 data kasus yang tidak sesuai dengan data aktual. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25) maka didapatkan hasil sebagai berikut.

$$\text{Akurasi} = \frac{27}{32} \times 100\% = 84.37\%$$

e. Skenario sistem dengan menggunakan nilai batas > 8

Pada uji eksperimen ini digunakan nilai batas > 8 terhadap jumlah nilai 1 yang terdapat dalam keseluruhan nilai matriks *e* pada metode ELECTRE yang digunakan dalam tahapan proses seleksi. Hasil yang didapatkan dari uji eksperimen ditunjukkan pada Tabel 6.41.

Tabel 6.41 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 8

Nomor	Data Aktual		Hasil Seleksi Sistem		Keterangan
1	Tidak Diterima	-	Tidak Diterima		Sesuai
2	Tidak Diterima	-	Tidak Diterima	-	Sesuai
3	Tidak Diterima	-	Tidak Diterima	-	Sesuai
4	Diterima	IPA	Diterima	IPS	Tidak Sesuai
5	Tidak Diterima	-	Tidak Diterima		Sesuai
6	Tidak Diterima	-	Tidak Diterima		Sesuai
7	Tidak Diterima	-	Tidak Diterima	-	Sesuai
8	Diterima	IPA	Diterima	IPA	Sesuai
9	Diterima	IPA/IPS	Diterima	IPS	Sesuai
10	Tidak Diterima	-	Tidak Diterma	-	Sesuai
11	Diterima	IPA	Diterima	IPS	Tidak Sesuai
12	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
13	Diterima	IPA	Diterima	IPA/IPS	Sesuai
14	Diterima	IPA	Diterima	IPA	Sesuai
15	Diterima	IPA	Diterima	IPA	Sesuai
16	Tidak Diterima	-	Tidak Diterima	-	Sesuai
17	Diterima	IPA	Diterima	IPA	Sesuai
18	Diterima	IPA	Tidak Diterima	IPA	Sesuai
19	Diterima	IPA	Diterima	IPA	Sesuai
20	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
21	Tidak Diterima	-	Tidak Diterima	-	Tidak Sesuai
22	Diterima	IPA	Diterima	IPA	Sesuai
23	Tidak Diterima	-	Tidak Diterima	-	Sesuai
24	Tidak Diterima	-	Tidak Diterima	-	Sesuai
25	Tidak Diterima	-	Tidak Diterima	IPA	Tidak Sesuai
26	Diterima	IPA	Diterima	IPA	Sesuai
27	Diterima	IPA	Tidak Diterima	IPA	Sesuai
28	Diterima	IPA	Diterima	IPA	Sesuai
29	Diterima	IPA	Diterima	IPA	Sesuai
30	Diterima	IPA	Diterima	IPA	Sesuai
31	Diterima	IPA	Diterima	IPA	Sesuai
32	Diterima	IPA/IPS	Diterima	IPA	Sesuai

Hasil pengujian dengan menggnakan nilai batas > 8 dari 32 data kasus didapatkan hasil 26 data kasus sesuai dengan data aktual dan 6 data kasus yang tidak sesuai dengan data aktual. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25) maka didapatkan hasil sebagai berikut.

$$\text{Akurasi} = \frac{26}{32} \times 100\% = 81.25\%$$

f. Skenario sistem dengan menggunakan nilai batas > 9

Pada uji eksperimen ini digunakan nilai batas > 9 terhadap jumlah nilai 1 yang terdapat dalam keseluruhan nilai matriks *e* pada metode ELECTRE yang digunakan dalam tahapan proses seleksi. Hasil yang didapatkan dari uji eksperimen ditunjukkan pada Tabel 6.42.

Tabel 6.42 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 9

Nomor	Data Aktual		Hasil Seleksi Sistem		Keterangan
1	Tidak Diterima	-	Tidak Diterima		Sesuai
2	Tidak Diterima	-	Tidak Diterima	-	Sesuai
3	Tidak Diterima	-	Tidak Diterima	-	Sesuai
4	Diterima	IPA	Diterima	IPS	Tidak Sesuai
5	Tidak Diterima	-	Tidak Diterima		Sesuai
6	Tidak Diterima	-	Tidak Diterima		Sesuai
7	Tidak Diterima	-	Tidak Diterima	-	Sesuai
8	Diterima	IPA	Diterima	IPA	Sesuai
9	Diterima	IPA/IPS	Diterima	IPS	Sesuai
10	Tidak Diterima	-	Tidak Diterma	-	Sesuai
11	Diterima	IPA	Diterima	IPS	Tidak Sesuai
12	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
13	Diterima	IPA	Diterima	IPA/IPS	Sesuai
14	Diterima	IPA	Diterima	IPA	Sesuai
15	Diterima	IPA	Diterima	IPA	Sesuai
16	Tidak Diterima	-	Tidak Diterima	-	Sesuai
17	Diterima	IPA	Diterima	IPA	Sesuai
18	Diterima	IPA	Tidak Diterima	IPA	Sesuai
19	Diterima	IPA	Diterima	IPA	Sesuai
20	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
21	Tidak Diterima	-	Tidak Diterima	-	Tidak Sesuai
22	Diterima	IPA	Diterima	IPA	Sesuai
23	Tidak Diterima	-	Tidak Diterima	-	Sesuai
24	Tidak Diterima	-	Tidak Diterima	-	Sesuai
25	Tidak Diterima	-	Tidak Diterima	IPA	Tidak Sesuai
26	Diterima	IPA	Diterima	IPA	Sesuai
27	Diterima	IPA	Tidak Diterima	IPA	Sesuai

28	Diterima	IPA	Diterima	IPA	Sesuai
29	Diterima	IPA	Diterima	IPA	Sesuai
30	Diterima	IPA	Diterima	IPA	Sesuai
31	Diterima	IPA	Diterima	IPA	Sesuai
32	Diterima	IPA/IPS	Tidak Diterima	-	Sesuai

Hasil pengujian dengan menggnakan nilai batas > 9 dari 32 data kasus didapatkan hasil 25 data kasus sesuai dengan data aktual dan 7 data kasus yang tidak sesuai dengan data aktual. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25) maka didapatkan hasil sebagai berikut.

$$\text{Akurasi} = \frac{25}{32} \times 100\% = 78.12\%$$

g. Skenario sistem dengan menggunakan nilai batas > 11

Pada uji eksperimen ini digunakan nilai batas > 11 terhadap jumlah nilai 1 yang terdapat dalam keseluruhan nilai matriks *e* pada metode ELECTRE yang digunakan dalam tahapan proses seleksi. Hasil yang didapatkan dari uji eksperimen ditunjukkan pada Tabel 6.43.

Tabel 6.43 Hasil Pengujian Tingkat Akurasi dengan Nilai Batas > 11

Nomor	Data Aktual		Hasil Seleksi Sistem		Keterangan
1	Tidak Diterima	-	Tidak Diterima		Sesuai
2	Tidak Diterima	-	Tidak Diterima	-	Sesuai
3	Tidak Diterima	-	Tidak Diterima	-	Sesuai
4	Diterima	IPA	Diterima	IPS	Tidak Sesuai
5	Tidak Diterima	-	Tidak Diterima		Sesuai
6	Tidak Diterima	-	Tidak Diterima		Sesuai
7	Tidak Diterima	-	Tidak Diterima	-	Sesuai
8	Diterima	IPA	Diterima	IPA	Sesuai
9	Diterima	IPA/IPS	Diterima	IPS	Sesuai
10	Tidak Diterima	-	Tidak Diterma	-	Sesuai
11	Diterima	IPA	Tidak Diterima	IPS	Tidak Sesuai
12	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
13	Diterima	IPA	Diterima	IPA/IPS	Sesuai
14	Diterima	IPA	Diterima	IPA	Sesuai
15	Diterima	IPA	Diterima	IPA	Sesuai
16	Tidak Diterima	-	Tidak Diterima	-	Sesuai
17	Diterima	IPA	Diterima	IPA	Sesuai
18	Diterima	IPA	Tidak Diterima	IPA	Sesuai
19	Diterima	IPA	Tidak Diterima	IPA	Sesuai
20	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai

21	Tidak Diterima	-	Tidak Diterima	-	Tidak Sesuai
22	Diterima	IPA	Diterima	IPA	Sesuai
23	Tidak Diterima	-	Tidak Diterima	-	Sesuai
24	Tidak Diterima	-	Tidak Diterima	-	Sesuai
25	Tidak Diterima	-	Tidak Diterima	IPA	Tidak Sesuai
26	Diterima	IPA	Diterima	IPA	Sesuai
27	Diterima	IPA	Tidak Diterima	IPA	Sesuai
28	Diterima	IPA	Diterima	IPA	Sesuai
29	Diterima	IPA	Diterima	IPA	Sesuai
30	Diterima	IPA	Diterima	IPA	Sesuai
31	Diterima	IPA	Diterima	IPA	Sesuai
32	Diterima	IPA/IPS	Tidak Diterima	-	Sesuai

Hasil pengujian dengan menggnakan nilai batas > 11 dari 32 data kasus didapatkan hasil 24 data kasus sesuai dengan data aktual dan 8 data kasus yang tidak sesuai dengan data aktual. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25) maka didapatkan hasil sebagai berikut.

$$\text{Akurasi} = \frac{24}{32} \times 100\% = 75\%$$

6.2.1.2 Hasil Akhir

Pengujian tingkat akurasi dilakukan dengan menggunakan data sebanyak 25 calon peserta dan 32 data calon siswa didapatkan hasil dengan rincian sebagai berikut.

Pada uji eksperimen ini digunakan nilai batas > 7 terhadap jumlah nilai 1 yang terdapat dalam keseluruhan nilai matriks E pada metode ELECTRE yang digunakan dalam tahapan proses seleksi. Hasil yang didapatkan dari uji eksperimen ditunjukkan pada Tabel 6.44.

Tabel 6.44 Hasil Perbandingan Seleksi dan Peminatan Perhitungan Manual dan Sistem

Nomor	Data Aktual		Hasil Seleksi Sistem		Keterangan
1	Tidak Diterima	-	Tidak Diterima		Sesuai
2	Tidak Diterima	-	Tidak Diterima	-	Sesuai
3	Tidak Diterima	-	Tidak Diterima	-	Sesuai
4	Diterima	IPA	Diterima	IPS	Tidak Sesuai
5	Tidak Diterima	-	Tidak Diterima		Sesuai
6	Tidak Diterima	-	Tidak Diterima		Sesuai
7	Tidak Diterima	-	Tidak Diterima	-	Sesuai
8	Diterima	IPA	Diterima	IPA	Sesuai
9	Diterima	IPA/IPS	Diterima	IPS	Sesuai
10	Tidak Diterima	-	Tidak Diterma	-	Sesuai

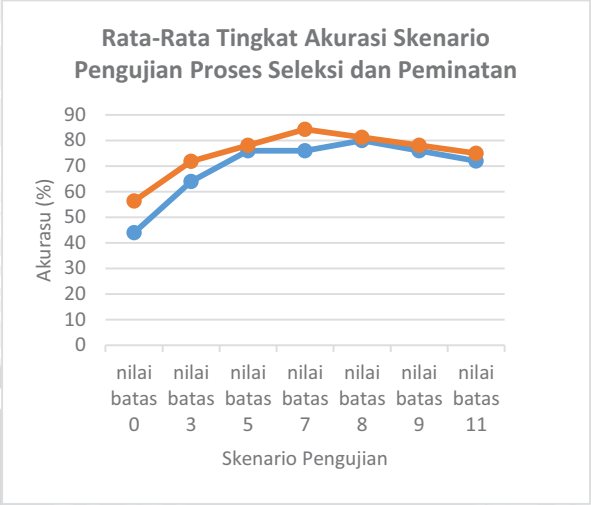
11	Diterima	IPA	Diterima	IPS	Tidak Sesuai
12	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
13	Diterima	IPA	Diterima	IPA/IPS	Sesuai
14	Diterima	IPA	Diterima	IPA	Sesuai
15	Diterima	IPA	Diterima	IPA	Sesuai
16	Tidak Diterima	-	Tidak Diterima	-	Sesuai
17	Diterima	IPA	Diterima	IPA	Sesuai
18	Diterima	IPA	Diterima	IPA	Sesuai
19	Diterima	IPA	Diterima	IPA	Sesuai
20	Diterima	IPA	Tidak Diterima	-	Tidak Sesuai
21	Tidak Diterima	-	Tidak Diterima	-	Tidak Sesuai
22	Diterima	IPA	Diterima	IPA	Sesuai
23	Tidak Diterima	-	Tidak Diterima	-	Sesuai
24	Tidak Diterima	-	Tidak Diterima	-	Sesuai
25	Tidak Diterima	-	Diterima	IPA	Tidak Sesuai
26	Diterima	IPA	Diterima	IPA	Sesuai
27	Diterima	IPA	Diterima	IPA	Sesuai
28	Diterima	IPA	Diterima	IPA	Sesuai
29	Diterima	IPA	Diterima	IPA	Sesuai
30	Diterima	IPA	Diterima	IPA	Sesuai
31	Diterima	IPA	Diterima	IPA	Sesuai
32	Diterima	IPA/IPS	Diterima	IPA	Sesuai

Hasil pengujian dengan menggnakan nilai batas > 7 dari 32 data kasus didapatkan hasil 27 data kasus sesuai dengan data aktual dan 5 data kasus yang tidak sesuai dengan data aktual. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25) maka didapatkan hasil sebagai berikut.

$$\text{Akurasi} = \frac{27}{32} \times 100\% = 84.37\%$$

6.3 Analisa Pengujian Tingkat Akurasi

Pada analisis hasil tingkat akurasi akan dijelaskan mengenai analisa hasil terhadap 2 skenario pengujian dengan 7 percobaan yang dilakukan menggunakan algoritma ELECTRE - *Weighted Product*. Pengujian dilakukan dengan membandingkan hasil akhir sistem dengan data aktual. Berdasarkan 25 data siswa tahun ajaran 2014/2015 dari jalur tes didapatkan beberapa hasil sesuai dengan scenario yang digunakan. Hasil akurasi scenario pengujian dapat dilihat pada Gambar 6.2



Gambar 6.2 Grafik Tingkat Akruasi Hasil Pengujian Nilai Batas
Sumber: [Pengujian]

Pada skenario 1 (yang ditunjukkan dengan grafik biru) uji coba 1 digunakan data sebanyak 25 data siswa baru dan melakukan perubahan terhadap nilai batas pada metode ELECTRE. Uji coba 1 menggunakan nilai batas > 0 dengan menghasilkan nilai akurasi sebesar 44%. Tingkat akurasi yang dihasilkan hanya 44% ini disebabkan akrena terlalu banyak nilai yang lebih atau sama dengan 1, sehingga hasil keputusan tidak sesuai dengan data aktual. Pada skenario 2 (yang ditunjukkan dengan grafik oranye) uji coba 1 digunakan data sebanyak 32 data siswa baru dan melakukan perubahan terhadap nilai batas pada metode ELECTRE dengan menggunakan nilai batas > 0, sama seperti skenario 1 uji coba 1, tingkat akurasi yang dihasilkan hanya sebesar 56.33%.

Pada skenario 1 (yang ditunjukkan dengan grafik biru) uji coba 2 digunakan data sebanyak 25 data siswa baru dan melakukan perubahan terhadap nilai batas pada metode ELECTRE. Uji coba 2 menggunakan nilai batas > 3 dengan menghasilkan nilai akurasi sebesar 64%. Tingkat akurasi mengalami peningkatan sebesar 20% dari uji coba pertama. Pada skenario 2 (yang ditunjukkan dengan grafik oranye) uji coba 2 digunakan data sebanyak 32 data siswa baru dan melakukan perubahan terhadap nilai batas pada metode ELECTRE dengan menggunakan nilai batas > 0, sama seperti skenario 1 uji coba 1, nilai akurasi meningkat menjadi 71.8%

Pada skenario 1 (yang ditunjukkan dengan grafik biru) uji coba 3 digunakan data sebanyak 25 data siswa baru dan melakukan perubahan terhadap nilai batas pada metode ELECTRE. Uji coba 3 menggunakan nilai batas > 5 dengan menghasilkan nilai akurasi sebesar 76%. Tingkat akurasi kembali meningkat sebesar 12% dari uji coba 2. Pada skenario 2 (yang ditunjukkan dengan grafik oranye) uji coba 3 digunakan data sebanyak 32 data siswa baru dan melakukan perubahan terhadap nilai batas pada metode ELECTRE dengan menggunakan nilai batas >5, berbeda skenario 1 uji coba 3, dimana nilai akurasi scenario 2 uji coba 3 hanya meningkat sebesar 6.32% dari uji coba 3 yakni menjadi 78.12%.

Pada skenario 1 (yang ditunjukkan dengan grafik biru) uji coba 4 digunakan data sebanyak 25 data siswa baru dan melakukan perubahan terhadap nilai batas pada metode ELECTRE. Uji coba 4 menggunakan nilai batas > 7 dengan menghasilkan nilai akurasi sebesar 76 %. Tingkat akurasi tidak mengalami perubahan dari uji coba 3. Pada skenario 2 (yang ditunjukkan dengan grafik oranye) uji coba 4 digunakan data sebanyak 32 data siswa baru dan melakukan perubahan terhadap nilai batas pada metode ELECTRE dengan menggunakan nilai batas >7, berbeda skenario 1 uji coba 4, nilai akurasi scenario 2 uji coba 4 mengalami peningkatan menjadi 84.37%.

Pada skenario 1 (yang ditunjukkan dengan grafik biru) uji coba 5 digunakan data sebanyak 25 data siswa baru dan melakukan perubahan terhadap nilai batas pada metode ELECTRE. Uji coba 5 menggunakan nilai batas > 8 akurasi kembali meningkat dengan menghasilkan nilai akurasi sebesar 80%. Pada skenario 2 (yang ditunjukkan dengan grafik oranye) uji coba 5 digunakan data sebanyak 32 data siswa baru dan melakukan perubahan terhadap nilai batas pada metode ELECTRE dengan menggunakan nilai batas >8, berbeda skenario 1 uji coba 5, nilai akurasi scenario 2 uji coba 4 mulai mengalami penurunan menjadi 81.25%

Pada skenario 1 (yang ditunjukkan dengan grafik biru) uji coba 6 digunakan data sebanyak 25 data siswa baru dan melakukan perubahan terhadap nilai batas pada metode ELECTRE. Uji coba 6 menggunakan nilai batas > 9 akurasi mulai menurun dengan menghasilkan nilai akurasi sebesar 76%. Pada skenario 2 (yang ditunjukkan dengan grafik oranye) uji coba 5 digunakan data sebanyak 32 data siswa baru dan melakukan perubahan terhadap nilai batas pada metode ELECTRE

dengan menggunakan nilai batas >8 dan tingkat akurasi kembali menurun pada 78.15%.

Pada skenario 1 (yang ditunjukkan dengan grafik biru) uji coba 7 digunakan data sebanyak 25 data siswa baru dan melakukan perubahan terhadap nilai batas pada metode ELECTRE. Uji coba 7 menggunakan nilai batas > 11 akurasi kembali menurun dengan menghasilkan nilai akurasi sebesar 72%. Pada skenario 2 (yang ditunjukkan dengan grafik oranye) uji coba 7 digunakan data sebanyak 32 data siswa baru dan melakukan perubahan terhadap nilai batas pada metode ELECTRE dengan menggunakan nilai batas >11 dan tingkat akurasi kembali menurun pada 75%.

Pada pengujian ini didapatkan tingkat akurasi terbaik dan terburuk terhadap dua scenario percobaan sebagai berikut:

- Akurasi terbaik ada pada skenario 2 uji coba 4 dengan nilai batas > 4 pada ELECTRE yang menghasilkan nilai akurasi sebesar 84,37%
- Akruasi terburuk ada pada skenario 1 uji coba 1 dengan nilai batas > 0 pada ELECTRE yang menghasilkan nilai akurasi sebesar 44%

Maka dapat disimpulkan bahwa pengubahan nilai batas pada metode ELECTRE mempengaruhi tingkat akurasi perhitungan sistem. Tidak hanya nilai batas, namun jumlah datapun mempengaruhi tingkat akurasi yang dimiliki oleh sistem.

6.4 Precision and Recall

6.4.1 Perhitungan *precision* and *recall* 25 data.

a. Pengujian Precision and Recall dilakukan untuk kasus pengujian 25 data dengan nilai batas > 0 berdasarkan hasil perhitungan pada Tabel 6.30

Tabel 6.45 Hasil *Precision and Recall* 25 Data dengan Nilai Batas > 0

Penerimaan		Nilai Sebenarnya (Seleksi)		Jumlah	Nilai Sebenarnya (IPA)			Jumlah	Nilai Sebenarnya (IPS)			Jumlah
		T	F			T	F			T	F	
Hasil Metode	T	12	11	23	T	9	0	9	T	1	2	3
	F	1	1	2	F	3	0	3	F	0	0	0
Jumlah		13	12	25	J	12	0	12	J	1	2	3

Hasil perhitungan *precision and recall* dengan menggnakan nilai batas > 0 dari 32 data kasus didapatkan *True Positive* (TP) sebanyak 12, *False Positive* (FP)

sebanyak 11, *False Negative* (FN) sebanyak 1, dan *True Negative* (TN) sebanyak 1.

- Tingkat *precision* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-22) maka didapatkan hasil sebagai berikut.

$$Precision\ (P)\ Seleksi = \frac{12}{23} = 36.36\%$$

$$Precision\ (P)\ IPA = \frac{9}{9} = 100\%$$

$$Precision\ (P)\ IPS = \frac{1}{3} = 75\%$$

- Tingkat *recall* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-23) maka didapatkan hasil sebagai berikut.

$$Recall\ (R)\ Seleksi = \frac{12}{13} = 92.30\%$$

$$Recall\ (R)\ IPA = \frac{9}{12} = 75\%$$

$$Recall\ (R)\ IPS = \frac{1}{1} = 100\%$$

- *F1 Measure* dapat dihitung dengan Persamaan (2-24), maka didapatkan hasil sebagi berikut.

$$F1\ Measure\ Seleksi = \frac{2PR}{P+R} = \frac{2x36.36x92.30}{36.36+92.30} = 52.16\%$$

$$F1\ Measure\ IPA = \frac{2PR}{P+R} = \frac{2x100x75}{100+75} = 85.71\%$$

$$F1\ Measure\ IPS = \frac{2PR}{P+R} = \frac{2x75x100}{75+100} = 85.71\%$$

b. Pengujian Precision and Recall dilakukan untuk kasus pengujian 25 data dengan nilai batas > 3 berdasarkan hasil perhitungan pada Tabel 6.31

Tabel 6.46 Hasil *Precision and Recall* 25 Data dengan Nilai Batas > 3

Penerimaan		Nilai Sebenarnya (Seleksi)		Jumlah	Nilai Sebenarnya (IPA)			Jumlah	Nilai Sebenarnya (IPS)			Jumlah
		T	F			T	F			T	F	
	T	12	6	18	T	9	0	9	T	1	2	3

Hasil Metode	F	1	6	7	F	3	0	3	F	0	0	0
Jumlah		13	12	25	J	12	0	12	J	1	2	3

Hasil perhitungan *precision and recall* dengan menggnakan nilai batas > 3 dari 32 data kasus didapatkan *True Positive* (TP) sebanyak 12, *False Positive* (FP) sebanyak 6, *False Negative* (FN) sebanyak 1, dan *True Negative* (TN) sebanyak 6.

- Tingkat *precision* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-22) maka didapatkan hasil sebagai berikut.

$$Precision\ (P)\ Seleksi = \frac{12}{18} = 66.67\%$$

$$Precision\ (P)\ IPA = \frac{9}{9} = 100\%$$

$$Precision\ (P)\ IPS = \frac{1}{3} = 33.33\%$$

- Tingkat *recall* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-23) maka didapatkan hasil sebagai berikut.

$$Recall\ (R)\ Seleksi = \frac{12}{13} = 92.30\%$$

$$Recall\ (R)\ IPA = \frac{9}{12} = 75\%$$

$$Recall\ (R)\ IPS = \frac{1}{1} = 100\%$$

- *F1 Measure* dapat dihitung dengan Persamaan (2-24), maka didapatkan hasil sebagi berikut.

$$F1\ Measure\ Seleksi = \frac{2PR}{P+R} = \frac{2x66.67x92.30}{66.67+92.30} = 77.41\%$$

$$F1\ Measure\ IPA = \frac{2PR}{P+R} = \frac{2x100x75}{100+75} = 85.71\%$$

$$F1\ Measure\ IPS = \frac{2PR}{P+R} = \frac{2x33.33x100}{33.33+100} = 49.99\%$$

c. Pengujian Precision and Recall dilakukan untuk kasus pengujian 25 data dengan nilai batas > 5 berdasarkan hasil perhitungan pada Tabel 6.32

Tabel 6.47 Hasil *Precision and Recall* 25 Data dengan Nilai Batas > 5

Penerimaan		Nilai Sebenarnya (Seleksi)		Jumlah	Nilai Sebenarnya (IPA)			Jumlah	Nilai Sebenarnya (IPS)			Jumlah
		T	F			T	F			T	F	
Hasil Metode	T	12	3	15	T	9	0	9	T	1	2	3
	F	1	9	10	F	3	0	3	F	0	0	0
Jumlah		13	12	25	J	12	0	12	J	1	2	3

Hasil perhitungan *precision and recall* dengan menggnakan nilai batas > 5 dari 32 data kasus didapatkan *True Positive* (TP) sebanyak 12, *False Positive* (FP) sebanyak 3, *False Negative* (FN) sebanyak 1, dan *True Negative* (TN) sebanyak 9.

- Tingkat *precision* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-22) maka didapatkan hasil sebagai berikut.

$$Precision\ (P)Seleksi = \frac{12}{15} = 80\%$$

$$Precision\ (P)IPA = \frac{9}{9} = 100\%$$

$$Precision\ (P)\ IPS = \frac{1}{3} = 33.33\%$$

- Tingkat *recall* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-23) maka didapatkan hasil sebagai berikut.

$$Recall\ (R)Seleksi = \frac{12}{13} = 92.30\%$$

$$Recall\ (R)IPA = \frac{9}{12} = 75\%$$

$$Recall\ (R)\ IPS = \frac{1}{1} = 100\%$$

- *F1 Measure* dapat dihitung dengan Persamaan (2-24), maka didapatkan hasil sebagi berikut.

$$F1\ Measure\ Seleksi = \frac{2PR}{P+R} = \frac{2x80x92.30}{80+92.30} = 85.71\%$$

$$F1\ Measure\ IPA = \frac{2PR}{P+R} = \frac{2x100x75}{100+75} = 85.71\%$$

$$F1\ Measure\ IPS = \frac{2PR}{P+R} = \frac{2x33.33x100}{33.33+100} = 49.99\%$$

d. Pengujian Precision and Recall dilakukan untuk kasus pengujian 25 data dengan nilai batas > 7 berdasarkan hasil perhitungan pada Tabel 6.33

Tabel 6.48 Hasil *Precision and Recall* 25 Data dengan Nilai Batas > 7

Penerimaan		Nilai Sebenarnya (Seleksi)		Jumlah	Nilai Sebenarnya (IPA)			Jumlah	Nilai Sebenarnya (IPS)			Jumlah
		T	F			T	F			T	F	
Hasil Metode	T	9	1	10	T	7	0	7	T	1	1	2
	F	4	11	15	F	5	0	5	F	0	0	0
Jumlah		13	12	25	J	12	0	12	J	1	1	2

Hasil perhitungan *precision and recall* dengan menggnakan nilai batas >7 dari 32 data kasus didapatkan *True Positive* (TP) sebanyak 9, *False Positive* (FP) sebanyak 1, *False Negative* (FN) sebanyak 4, dan *True Negative* (TN) sebanyak 11.

- Tingkat *precision* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-22) maka didapatkan hasil sebagai berikut.

$$Precision\ (P)Seleksi = \frac{9}{10} = 90\%$$

$$Precision\ (P)IPA = \frac{7}{7} = 100\%$$

$$Precision\ (P)IPS = \frac{1}{2} = 50\%$$

- Tingkat *recall* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-23) maka didapatkan hasil sebagai berikut.

$$Recall\ (R)Seleksi = \frac{9}{13} = 69.23\%$$

$$Recall\ (R)IPA = \frac{7}{12} = 58.33\%$$

$$Recall\ (R)IPS = \frac{1}{1} = 100\%$$

- *F1 Measure* dapat dihitung dengan Persamaan (2-25), maka didapatkan hasil sebagi berikut.

$$F1\ Measure\ Seleksi = \frac{2PR}{P+R} = \frac{2x90x69.23}{90+69.23} = 78.26\%$$

$$F1\ Measure\ IPA = \frac{2PR}{P+R} = \frac{2x100x58.33}{100+58.33} = 73.26\%$$

$$F1\ Measure\ IPS = \frac{2PR}{P+R} = \frac{2x50x100}{50+100} = 66.67\%$$

e. Pengujian Precision and Recall dilakukan untuk kasus pengujian 25 data dengan nilai batas > 8 berdasarkan hasil perhitungan pada Tabel 6.34

Tabel 6.49 Hasil *Precision and Recall* 25 Data dengan Nilai Batas >8

Penerimaan		Nilai Sebenarnya (Seleksi)		Jumlah	Nilai Sebenarnya (IPA)			Jumlah	Nilai Sebenarnya (IPS)			Jumlah
		T	F			T	F			T	F	
Hasil Metode	T	9	0	9	T	7	0	7	T	1	1	2
	F	4	12	16	F	5	0	5	F	0	0	0
Jumlah		13	12	25	J	12	0	12	J	1	1	2

Hasil perhitungan *precision and recall* dengan menggnakan nilai batas > 8 dari 32 data kasus didapatkan *True Positive* (TP) sebanyak 9, *False Positive* (FP) sebanyak 0, *False Negative* (FN) sebanyak 4, dan *True Negative* (TN) sebanyak 12.

- Tingkat *precision* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-22) maka didapatkan hasil sebagai berikut.

$$Precision\ (P)Seleksi = \frac{9}{9} = 100\%$$

$$Precision\ (P)IPA = \frac{7}{7} = 100\%$$

$$Precision\ (P)IPS = \frac{1}{2} = 50\%$$

- Tingkat *recall* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-23) maka didapatkan hasil sebagai berikut.

$$Recall\ (R)Seleksi = \frac{9}{13} = 69.23\%$$

$$Recall\ (R)IPA = \frac{7}{12} = 58.33\%$$

$$Recall\ (R)IPS = \frac{1}{1} = 100\%$$

- *F1 Measure* dapat dihitung dengan Persamaan (2-24), maka didapatkan hasil sebagi berikut.

$$F1\ Measure\ Seleksi = \frac{2PR}{P+R} = \frac{2x100x69.23}{100+69.23} = 81.81\%$$

$$F1\ Measure\ IPA = \frac{2PR}{P+R} = \frac{2x100x58.33}{100+58.33} = 73.26\%$$

$$F1\ Measure\ IPS = \frac{2PR}{P+R} = \frac{2x50x100}{50+100} = 66.67\%$$

f. Pengujian Precision and Recall dilakukan untuk kasus pengujian 25 data dengan nilai batas > 9 berdasarkan hasil perhitungan pada Tabel 6.35

Tabel 6.50 Hasil *Precision and Recall* 25 Data dengan Nilai Batas > 9

Penerimaan		Nilai Sebenarnya (Seleksi)		Jumlah	Nilai Sebenarnya (IPA)			Jumlah	Nilai Sebenarnya (IPS)			Jumlah
		T	F			T	F			T	F	
Hasil Metode	T	8	0	8	T	6	0	6	T	1	1	2
	F	5	12	17	F	6	0	6	F	0	0	0
Jumlah		13	12	25	J	12	0	12	J	1	1	2

Hasil perhitungan *precision and recall* dengan menggnakan nilai batas > 9 dari 32 data kasus didapatkan *True Positive* (TP) sebanyak 8, *False Positive* (FP) sebanyak 0, *False Negative* (FN) sebanyak 5, dan *True Negative* (TN) sebanyak 12.

- Tingkat *precision* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-22) maka didapatkan hasil sebagai berikut.

$$Precision\ (P)Seleksi = \frac{8}{8} = 100\%$$

$$Precision\ (P)IPA = \frac{6}{6} = 100\%$$

$$Precision\ (P)IPS = \frac{1}{2} = 50\%$$

- Tingkat *recall* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-23) maka didapatkan hasil sebagai berikut.

$$Recall\ (R)Seleksi = \frac{8}{13} = 58.33\%$$

$$Recall\ (R)IPA = \frac{6}{12} = 50\%$$

$$Recall\ (R)IPS = \frac{1}{1} = 100\%$$

- *F1 Measure* dapat dihitung dengan Persamaan (2-24), maka didapatkan hasil sebagi berikut.

$$F1\ Measure = \frac{2PR}{P+R} = \frac{2x100x58.33}{100+58.33} = 73.68\%$$

$$F1\ Measure\ IPS = \frac{2PR}{P+R} = \frac{2x100x50}{100+50} = 66.67\%$$

$$F1\ Measure\ IPS = \frac{2PR}{P+R} = \frac{2x50x100}{50+100} = 66.67\%$$

g. Pengujian Precision and Recall dilakukan untuk kasus pengujian 25 data dengan nilai batas > 11 berdasarkan hasil perhitungan pada Tabel 6.36

Tabel 6.51 Hasil *Precision and Recall* 25 Data dengan Nilai Batas > 11

Penerimaan		Nilai Sebenarnya (Seleksi)		Jumlah	Nilai Sebenarnya (IPA)			Jumlah	Nilai Sebenarnya (IPS)			Jumlah
		T	F			T	F			T	F	
Hasil Metode	T	7	0	7	T	5	0	6	T	1	1	2
	F	6	12	18	F	7	0	6	F	0	0	0
Jumlah		13	12	25	J	12	0	12	J	1	1	2

Hasil perhitungan *precision and recall* dengan menggnakan nilai batas > 11 dari 32 data kasus didapatkan *True Positive* (TP) sebanyak 7, *False Positive* (FP) sebanyak 0, *False Negative* (FN) sebanyak 6, dan *True Negative* (TN) sebanyak 12.

- Tingkat *precision* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-22) maka didapatkan hasil sebagai berikut.

$$Precision\ (P)Seleksi = \frac{7}{7} = 100\%$$

$$Precision\ (P)IPA = \frac{5}{5} = 100\%$$

$$Precision\ (P)IPS = \frac{1}{2} = 50\%$$

- Tingkat *recall* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-23) maka didapatkan hasil sebagai berikut.

$$Recall\ (R)Seleksi = \frac{7}{13} = 53.84\%$$

$$Recall\ (R)IPA = \frac{5}{12} = 41.67\%$$

$$Recall\ (R)IPS = \frac{1}{1} = 100\%$$

- *F1 Measure* dapat dihitung dengan Persamaan (2-24), maka didapatkan hasil sebagi berikut.

$$F1\ Measure\ Seleksi = \frac{2PR}{P+R} = \frac{2x100x53.84}{100+53.84} = 69.99\%$$

$$F1\ Measure\ IPA = \frac{2PR}{P+R} = \frac{2x100x41.67}{100+41.67} = 58.82\%$$

$$F1\ Measure\ IPS = \frac{2PR}{P+R} = \frac{2x50x100}{100+50} = 66.67\%$$

6.4.2 Perhitungan *precision* and *recall* 32 data.

a. Pengujian Precision and Recall dilakukan untuk kasus pengujian 32 data dengan nilai batas > 0 berdasarkan hasil perhitungan pada Tabel 6.37

Tabel 6.52 Hasil *Precision and Recall* 32 Data dengan Nilai Batas > 0

Penerimaan		Nilai Sebenarnya (Seleksi)		Jumlah	Nilai Sebenarnya (IPA)			Jumlah	Nilai Sebenarnya (IPS)			Jumlah
		T	F			T	F			T	F	
Hasil Metode	T	19	11	30	T	16	7	23	T	1	6	7
	F	1	1	2	F	3	0	3	F	0	0	0
Jumlah		20	12	32	J	19	7	26	J	1	6	7

Hasil perhitungan *precision and recall* dengan menggnakan nilai batas > 0 dari 32 data kasus didapatkan *True Positive* (TP) sebanyak 19, *False Positive* (FP) sebanyak 11, *False Negative* (FN) sebanyak 1, dan *True Negative* (TN) sebanyak 1.

- Tingkat *precision* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-22) maka didapatkan hasil sebagai berikut.

$$Precision\ (P)Seleksi = \frac{19}{30} = 63.33\%$$

$$Precision\ (P)IPA = \frac{16}{23} = 69.56\%$$

$$Precision\ (P)IPS = \frac{1}{7} = 14.28\%$$

- Tingkat *recall* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-23) maka didapatkan hasil sebagai berikut.

$$Recall\ (R)Seleksi = \frac{19}{20} = 95\%$$

$$Recall\ (R)IPA = \frac{16}{19} = 84.21\%$$

$$Recall\ (R)IPS = \frac{1}{1} = 100\%$$

- *F1 Measure* dapat dihitung dengan Persamaan (2-24), maka didapatkan hasil sebagi berikut.

$$F1\ Measure\ Seleksi = \frac{2PR}{P+R} = \frac{2x63.33x95}{63.33+95} = 75.99\%$$

$$F1\ MeasureIPA = \frac{2PR}{P+R} = \frac{2x69.56x84.21}{69.56+84.21} = 76.18\%$$

$$F1\ Measure\ IPS = \frac{2PR}{P+R} = \frac{2x14.28x100}{14.28+100} = 24.99\%$$

b. Pengujian Precision and Recall dilakukan untuk kasus pengujian 32 data dengan nilai batas > 3 berdasarkan hasil perhitungan pada Tabel 6.38

Tabel 6.53 Hasil *Precision and Recall* 32 Data dengan Nilai Batas > 3

Penerimaan		Nilai Sebenarnya (Seleksi)		Jumlah	Nilai Sebenarnya (IPA)			Jumlah	Nilai Sebenarnya (IPS)			Jumlah
		T	F			T	F			T	F	
Hasil Metode	T	19	6	25	T	15	4	19	T	1	4	5
	F	1	6	7	F	4	0	4	F	0	0	0
Jumlah		20	12	32	J	19	4	23	J	1	4	5

Hasil perhitungan *precision and recall* dengan menggnakan nilai batas > 3 dari 32 data kasus didapatkan *True Positive* (TP) sebanyak 19, *False Positive* (FP) sebanyak 6, *False Negative* (FN) sebanyak 1, dan *True Negative* (TN) sebanyak 6.

- Tingkat *precision* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-22) maka didapatkan hasil sebagai berikut.

$$Precision\ (P)Seleksi = \frac{19}{25} = 76\%$$

$$Precision\ (P)IPA = \frac{15}{19} = 78.94\%$$

$$Precision\ (P)IPS = \frac{1}{5} = 20\%$$

- Tingkat *recall* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-23) maka didapatkan hasil sebagai berikut.

$$Recall\ (R)\ Seleksi = \frac{19}{20} = 95\%$$

$$Recall\ (R)IPA = \frac{15}{19} = 78.94\%$$

$$Recall\ (R)IPS = \frac{1}{1} = 100\%$$

- *F1 Measure* dapat dihitung dengan Persamaan (2-24), maka didapatkan hasil sebagi berikut.

$$F1\ Measure\ Seleksi = \frac{2PR}{P+R} = \frac{2x76x95}{76+95} = 84.44\%$$

$$F1\ Measure\ IPA = \frac{2PR}{P+R} = \frac{2x78.94x78.94}{78.94+78.94} = 78.94\%$$

$$F1\ Measure\ IPS = \frac{2PR}{P+R} = \frac{2x20x100}{20+100} = 33.33\%$$

c. Pengujian Precision and Recall dilakukan untuk kasus pengujian 32 data dengan nilai batas > 5 berdasarkan hasil perhitungan pada Tabel 6.39

Tabel 6.54 Hasil *Precision and Recall* 32 Data dengan Nilai Batas > 5

Penerimaan		Nilai Sebenarnya (Seleksi)		Jumlah	Nilai Sebenarnya (IPA)			Jumlah	Nilai Sebenarnya (IPS)			Jumlah
		T	F			T	F			T	F	
Hasil Metode	T	19	4	23	T	16	3	19	T	1	3	4
	F	1	8	9	F	3	0	3	F	0	0	0
Jumlah		20	12	32	J	19	3	22	J	1	3	4

Hasil perhitungan *precision and recall* dengan menggnakan nilai batas >5 dari 32 data kasus didapatkan *True Positive* (TP) sebanyak 19, *False Positive* (FP) sebanyak 4, *False Negative* (FN) sebanyak 1, dan *True Negative* (TN) sebanyak 8.

- Tingkat *precision* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-22) maka didapatkan hasil sebagai berikut.

$$Precision\ (P) = \frac{19}{23} = 82.60\%$$

$$Precision\ (P)IPA = \frac{16}{19} = 84.21\%$$

$$Precision\ (P)IPS = \frac{1}{4} = 25\%$$

- Tingkat *recall* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-23) maka didapatkan hasil sebagai berikut.

$$Recall\ (R) = \frac{19}{20} = 95\%$$

$$Recall\ (R)IPA = \frac{16}{19} = 84.21\%$$

$$Recall\ (R)IPS = \frac{1}{1} = 100\%$$

- *F1 Measure* dapat dihitung dengan Persamaan (2-24), maka didapatkan hasil sebagi berikut.

$$F1\ Measure = \frac{2PR}{P+R} = \frac{2x82.60x95}{82.60+95} = 88.36\%$$

$$F1\ Measure = \frac{2PR}{P+R} = \frac{2x84.21x84.21}{84.21+84.21} = 84.21\%$$

$$F1\ Measure = \frac{2PR}{P+R} = \frac{2x25x100}{25+100} = 40\%$$

d. Pengujian Precision and Recall dilakukan untuk kasus pengujian 32 data dengan nilai batas > 7 berdasarkan hasil perhitungan pada Tabel 6.40

Tabel 6.55 Hasil *Precision and Recall* 32 Data dengan Nilai Batas > 7

Penerimaan		Nilai Sebenarnya (Seleksi)		Jumlah	Nilai Sebenarnya (IPA)			Jumlah	Nilai Sebenarnya (IPS)			Jumlah
		T	F			T	F			T	F	
	T	18	1	19	T	15	1	16	T	1	2	3

Hasil Metode	F	2	11	13	F	4	0	4	F	0	0	0
Jumlah		20	12	32	J	19	1	20	J	1	2	3

Hasil perhitungan *precision and recall* dengan menggnakan nilai batas > 7 dari 32 data kasus didapatkan *True Positive* (TP) sebanyak 18, *False Positive* (FP) sebanyak 1, *False Negative* (FN) sebanyak 2, dan *True Negative* (TN) sebanyak 11.

- Tingkat *precision* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-22) maka didapatkan hasil sebagai berikut.

$$Precision\ (P) = \frac{18}{19} = 94.33\%$$

$$Precision\ (P)IPA = \frac{15}{16} = 93.75\%$$

$$Precision\ (P)IPS = \frac{1}{3} = 33.33\%$$

- Tingkat *recall* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-23) maka didapatkan hasil sebagai berikut.

$$Recall\ (R) = \frac{18}{20} = 90\%$$

$$Recall\ (R)\ IPA = \frac{15}{19} = 78.94\%$$

$$Recall\ (R)IPS = \frac{1}{1} = 100\%$$

- *F1 Measure* dapat dihitung dengan Persamaan (2-24), maka didapatkan hasil sebagi berikut.

$$F1\ Measure = \frac{2PR}{P+R} = \frac{2x94.33x90}{94.33+90} = 92.30\%$$

$$F1\ Measure\ IPA = \frac{2PR}{P+R} = \frac{2x93.75x78.94}{93.75+78.94} = 85.70\%$$

$$F1\ Measure\ IPS = \frac{2PR}{P+R} = \frac{2x33.33x100}{33.33+100} = 49.99\%$$

e. Pengujian Precision and Recall dilakukan untuk kasus pengujian 32 data dengan nilai batas > 8 berdasarkan hasil perhitungan pada Tabel 6.41

Tabel 6.56 Hasil *Precision and Recall* 32 Data dengan Nilai Batas > 8

Penerimaan		Nilai Sebenarnya (Seleksi)		Jumlah	Nilai Sebenarnya (IPA)			Jumlah	Nilai Sebenarnya (IPS)			Jumlah
		T	F			T	F			T	F	
Hasil Metode	T	16	0	16	T	15	1	16	T	1	2	3
	F	4	12	16	F	4	0	4	F	0	0	0
Jumlah		20	12	32	J	19	1	20	J	1	2	3

Hasil perhitungan *precision and recall* dengan menggnakan nilai batas > 8 dari 32 data kasus didapatkan *True Positive* (TP) sebanyak 16, *False Positive* (FP) sebanyak 0, *False Negative* (FN) sebanyak 4, dan *True Negative* (TN) sebanyak 12.

- Tingkat *precision* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-22) maka didapatkan hasil sebagai berikut.

$$Precision\ (P) = \frac{16}{16} = 100\%$$

$$Precision\ (P)IPA = \frac{15}{16} = 93.75\%$$

$$Precision\ (P)IPS = \frac{1}{3} = 33.33\%$$

- Tingkat *recall* yang didapatkan dari pengujian tersebut dihitung dengan persamaan (2-23) maka didapatkan hasil sebagai berikut.

$$Recall\ (R) = \frac{16}{20} = 80\%$$

$$Recall\ (R)\ IPA = \frac{15}{19} = 78.94\%$$

$$Recall\ (R)IPS = \frac{1}{1} = 100\%$$

- *F1 Measure* dapat dihitung dengan Persamaan (2-24), maka didapatkan hasil sebagi berikut.

$$F1\ Measure = \frac{2PR}{P+R} = \frac{2x100x80}{100+80} = 88.88\%$$

$$F1\ Measure\ IPA = \frac{2PR}{P+R} = \frac{2x93.75x78.94}{93.75+78.94} = 72.69\%$$

$$F1\ Measure\ IPS = \frac{2PR}{P+R} = \frac{2x33x100}{33+100} = 50.12\%$$

f. Pengujian Precision and Recall dilakukan untuk kasus pengujian 32 data dengan nilai batas > 9 berdasarkan hasil perhitungan pada Tabel 6.42

Tabel 6.57 Hasil *Precision and Recall* 32 Data dengan Nilai Batas > 9

Penerimaan		Nilai Sebenarnya (Seleksi)		Jumlah	Nilai Sebenarnya (IPA)			Jumlah	Nilai Sebenarnya (IPS)			Jumlah
		T	F			T	F			T	F	
Hasil Metode	T	15	0	15	T	15	1	16	T	1	2	3
	F	5	12	17	F	2	0	2	F	0	0	0
Jumlah		20	12	32	J	17	1	18	J	1	2	3

Hasil perhitungan *precision and recall* dengan menggnakan nilai batas > 9 dari 32 data kasus didapatkan *True Positive* (TP) sebanyak 15, *False Positive* (FP) sebanyak 0, *False Negative* (FN) sebanyak 5, dan *True Negative* (TN) sebanyak 12.

- Tingkat *precision* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-22) maka didapatkan hasil sebagai berikut.

$$Precision\ (P) = \frac{15}{15} = 100\%$$

$$Precision\ (P)IPA = \frac{15}{16} = 93.57\%$$

$$Precision\ (P)IPS = \frac{1}{3} = 33.33\%$$

- Tingkat *recall* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-23) maka didapatkan hasil sebagai berikut.

$$Recall\ (R) = \frac{15}{20} = 75\%$$

$$Recall\ (R)\ IPA = \frac{15}{17} = 88.23\%$$

$$Recall\ (R)\ IPS = \frac{1}{1} = 100\%$$

- *F1 Measure* dapat dihitung dengan Persamaan (2-24), maka didapatkan hasil sebagi berikut.

$$F1\ Measure = \frac{2PR}{P+R} = \frac{2x100x75}{100+75} = 85.71\%$$

$$F1\ Measure\ IPA = \frac{2PR}{P+R} = \frac{2x100x75}{100+75} = 81.8\%$$

$$F1\ Measure\ IPS = \frac{2PR}{P+R} = \frac{2x33.33x100}{33.33+100} = 49.99\%$$

g. Pengujian Precision and Recall dilakukan untuk kasus pengujian 32 data dengan nilai batas > 11 berdasarkan hasil perhitungan pada Tabel 6.43

Tabel 6.58 Hasil *Precision and Recall* 32 Data dengan Nilai Batas >11

Penerimaan		Nilai Sebenarnya (Seleksi)		Jumlah	Nilai Sebenarnya (IPA)			Jumlah	Nilai Sebenarnya (IPS)			Jumlah
		T	F			T	F			T	F	
Hasil Metode	T	13	0	13	T	14	1	15	T	1	2	3
	F	7	12	19	F	2	0	2	F	0	0	0
Jumlah		20	12	32	J	16	1	17	J	1	2	3

Hasil perhitungan *precision and recall* dengan menggnakan nilai batas > 11 dari 32 data kasus didapatkan *True Positive* (TP) sebanyak 13, *False Positive* (FP) sebanyak 0, *False Negative* (FN) sebanyak 7, dan *True Negative* (TN) sebanyak 12.

- Tingkat *precision* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-22) maka didapatkan hasil sebagai berikut.

$$Precision\ (P) = \frac{13}{13} = 100\%$$

$$Precision\ (P)IPA = \frac{14}{15} = 93.33\%$$

$$Precision\ (P)IPS = \frac{1}{3} = 33.33\%$$

- Tingkat *recall* yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-23) maka didapatkan hasil sebagai berikut.

$$Recall\ (R) = \frac{13}{20} = 65\%$$

$$Recall\ (R) = \frac{14}{16} = 87.5\%$$

$$Recall\ (R) = \frac{1}{1} = 100\%$$

- *F1 Measure* dapat dihitung dengan Persamaan (2-24), maka didapatkan hasil sebagi berikut.

$$F1\ Measure = \frac{2PR}{P+R} = \frac{2x100x65}{100+65} = 78.88\%$$

$$F1\ Measure\ IPS = \frac{2PR}{P+R} = \frac{2x93.33x87.5}{93.33+87.5} = 90.32\%$$

$$F1\ Measure\ IPS = \frac{2PR}{P+R} = \frac{2x33.33x100}{33.33+100} = 49.99\%$$

6.4.3 Analisis Hasil Pengujian Precision & Recall

Pada analisis hasil pengujian *precision & recall* akan dijelaskan mengenai analisa hasil terhadap 2 skenario pengujian *precision & recall* dengan 7 percobaan yang dilakukan menggunakan algoritma ELECTRE - *Weighted Product*. Pengujian *precision*, *recall*, dan F-*Measure* dibagi menjadi tiga bagian, yakni:

1. *Precision*, *Recall,* dan F-*Measure* Seleksi
2. *Precision*, *Recall,* dan F-*Measure* IPA
3. *Precision*, *Recall,* dan F-*Measure* IPS

Berdasarkan hasil pengujian yang telah dilakukan pada skenario pengujian pertama, hasil *F-Measure* seleksi tertinggi ada pada percobaan 3 (dengan *threshold* > 5) dengan nilai *f-measure* sebesar 85.71% dan hasil *f-measure* seleksi terendah ada pada percobaan 1 (dengan *threshold* > 0) dengan nilai *f-measure* sebesar 52.16%. Untuk hasil *f-measure* IPA, nilai tertinggi ada pada percobaan 1 (dengan *threshold* > 0), percobaan 2 (dengan *threshold* > 3), dan percobaan 3 (dengan *threshold* > 5) dengan nilai *f-measure* sebesar 85.71%. Nilai terendah *f-measure* IPA ada pada percobaan 7 (dengan *threshold* > 11) dengan nilai *f-measure* sebesar 58.82%. Rendahnya nilai *f-measure* IPA pada percobaan 7 disebabkan karena banyaknya alternatif calon siswa baru yang tidak diterima berdasarkan hasil percobaan metode ELECTRE dan hasil tersebut berpengaruh pada perhitungan *Weighted Product*. Untuk hasil *f-measure* IPS, nilai tertinggi ada pada percobaan 1 (dengan *threshold* > 0) dengan nilai *f-measure* sebesar 85.71%. Nilai terendah *f-measure* IPS ada pada percobaan 1 (dengan *threshold* > 0), percobaan 2 (dengan *threshold* > 3) dan percobaan 3 (dengan *threshold* > 5) dengan nilai *f-measure* sebesar 49.99%. Rendahnya nilai *f-measure* IPS pada percobaan 1, 2, dan 3 disebabkan karena berdasarkan perhitungan *Weighted Product*, terdapat tiga

alternatif yang masuk ke dalam peminatan IPS berdasarakan kriteria alternatif pemiantan, namun hanya ada satu alternatif yang masuk kelas IPS dalam hasil akhir BSS. SMA BSS memiliki kewenangan dalam mengambil keputusan akhir dari kelas peminatan.

Berdasarkan hasil pengujian yang telah dilakukan pada skenario pengujian kedua, hasil *F-Measure* seleksi tertinggi ada pada percobaan 4 (dengan *threshold* > 7) dengan nilai *f-measure* sebesar 92.30% dan hasil *f-measure* seleksi terendah ada pada percobaan 1 (dengan *threshold* > 0) dengan nilai *f-measure* sebesar 75.99%. Untuk hasil *f-measure* IPA, nilai tertinggi ada pada percobaan 7 (dengan *threshold* > 11) dengan nilai *f-measure* sebesar 90.32%. Nilai terendah *f-measure* IPA ada pada percobaan 5 (dengan *threshold* > 8) dengan nilai *f-measure* sebesar 72.96%. Rendahnya nilai *f-measure* IPA pada percobaan 7 disebabkan karena banyaknya alternatif calon siswa baru yang tidak diterima berdasarkan hasil percobaan metode ELECTRE dan hasil tersebut berpengaruh pada perhitungan *Weighted Product*. Untuk hasil *f-measure* IPS, nilai tertinggi ada pada percobaan 5 (dengan *threshold* > 8) dengan nilai *f-measure* sebesar 50.12%. Nilai terendah *f-measure* IPS ada pada percobaan 1 (dengan *threshold* > 0) dengan nilai *f-measure* sebesar 24.99%. Rendahnya nilai *f-measure* IPS pada percobaan 1 disebabkan karena berdasarkan perhitungan ELECTRE - *Weighted Product*, terdapat tujuh alternatif yang diterima dan masuk ke dalam peminatan IPS berdasarkan kriteria alternatif pemiantan, namun hanya ada satu alternatif yang masuk kelas IPS dalam hasil akhir BSS. SMA BSS memiliki kewenangan dalam mengambil keputusan akhir dari kelas peminatan.

6.5 Pengujian Akurasi Menggunakan Kuota

a. Pengujian Akurasi

Pengujian akurasi dilakukan dengan menggunakan nilai batas > 0 berdasarkan kuota yang diterapkan oleh SMA BSS. Skenario pengujian yang menggunakan data calon siswa sebanyak 32 data, dengan menggunakan nilai pembobotan yang didapatkan dari hasil wawancara dengan pihak SMA BSS. Berikut ini skenario pengujian akurasi kuota dengan menggunakan 32 data calon siswa dan kuota sebesar 20.

Tabel 6.59 Hasil Seleksi Kuota

no urut	Perhitungan BSS	Perhitungan Metode	Hasil BSS	Hasil Metode	Hasil Akhir
31	7949	20	diterima	diterima	diterima
9	7887	17	diterima	diterima	diterima
30	8022	16	diterima	diterima	diterima
15	7907	16	diterima	diterima	diterima
28	7902	16	diterima	diterima	diterima
19	7898	16	diterima	diterima	diterima
4	8026	15	diterima	diterima	diterima
17	7714	15	diterima	diterima	diterima
26	8131	13	diterima	diterima	diterima
8	7979	13	diterima	diterima	diterima
14	7997	12	diterima	diterima	diterima
22	7877	10	diterima	diterima	diterima
11	7783	10	diterima	diterima	diterima
32	7631	9	diterima	diterima	diterima
27	7717	8	diterima	diterima	diterima
18	7694	8	diterima	diterima	diterima
12	7818	7	diterima	diterima	diterima
25	7595	8	diterima	diterima	tidak diterima
20	7908	0	diterima	tidak diterima	diterima
6	7699	4	diterima	tidak diterima	tidak diterima
13	6828	31	tidak diterima	diterima	diterima
29	7547	13	tidak diterima	diterima	diterima
24	7434	7	tidak diterima	tidak diterima	tidak diterima
5	7557	6	tidak diterima	tidak diterima	tidak diterima
1	6667	6	tidak diterima	tidak diterima	tidak diterima
21	7527	5	tidak diterima	tidak diterima	tidak diterima
10	7553	3	tidak diterima	tidak diterima	tidak diterima
16	7302	3	tidak diterima	tidak diterima	tidak diterima
23	7082	3	tidak diterima	tidak diterima	tidak diterima
7	7317	2	tidak diterima	tidak diterima	tidak diterima
3	6632	1	tidak diterima	tidak diterima	tidak diterima
2	6611	0	tidak diterima	tidak diterima	tidak diterima

Sumber: [Pengujian]

Hasil metode ELECTRE diurutkan dari nilai terbesar hingga nilai terkecil. Sistem akan menyeleksi 20 nilai tertinggi dari perhitungan ELECTRE yang telah dilakukan. Tingkat akurasi yang didapatkan dari pengujian tersebut dihitung dengan Persamaan (2-25) maka didapatkan hasil sebagai berikut.

$$\text{Akurasi Data Diterima (Metode ELECTRE)} = \frac{30}{32} \times 100\% = 93.75\%$$

$$\text{Akurasi Data Diterima (Metode BSS)} = \frac{28}{32} \times 100\% = 87.5\%$$

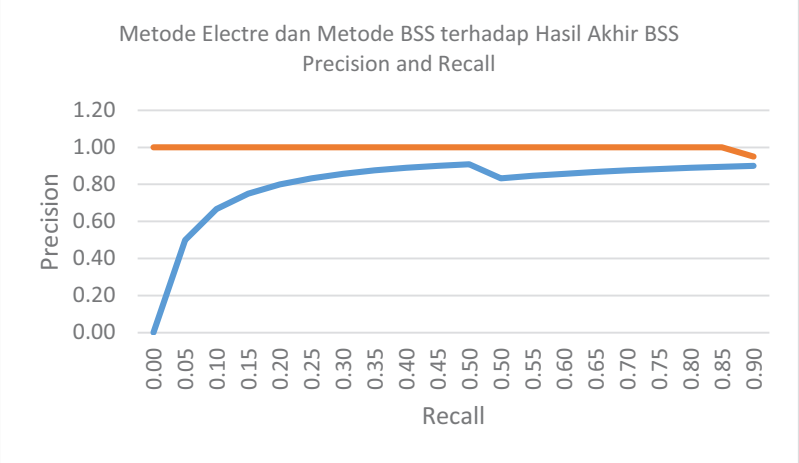
b. Precision dan Recall

Perhitungan *precision* dapat dihitung dengan Persamaan (2-22) dan *recall* dapat dihitung dengan Persamaan (2-23) terhadap pengujian 20 kuota. Berikut adalah hasil perhitungan *precision* dan *recall*.

Tabel 6.60 Hasil *Precision and Recall* Kuota

Metode BSS		Metode ELECTRE	
Precision	Recall	Precision	Recall
0.00	0.00	1	0.05
0.50	0.05	1	0.1
0.67	0.10	1	0.15
0.75	0.15	1	0.2
0.80	0.20	1	0.25
0.83	0.25	1	0.3
0.86	0.30	1	0.35
0.88	0.35	1	0.4
0.89	0.40	1	0.45
0.90	0.45	1	0.5
0.91	0.50	1	0.55
0.83	0.50	1	0.6
0.85	0.55	1	0.65
0.86	0.60	1	0.7
0.87	0.65	1	0.75
0.88	0.70	1	0.8
0.88	0.75	1	0.85
0.89	0.80	1	0.9
0.89	0.85	1	0.95
0.90	0.90	0.95	0.9

Sumber: [Pengujian]



Gambar 6.3 Grafik Tingkat Akruasi Hasil *Precision* & *Recall* Kuota
Sumber: [Pengujian]

Grafik oranye menunjukkan hasil *precision* dan *recall* metode ELECTRE dan grafik biru menunjukkan *precision* dan *recall* metode BSS. Grafik oranye cenderung stabil dan baru mengalami penurunan *precision* dan *recall* pada data ke-20. Berbeda dengan grafik oranye, grafik biru cenderung mengalami peningkatan dan penurunan. Pada data pertama, grafik biru berada dititik terendah *precision* dan *recall* karena ketidaksesuaian antara data perhitungan metode BSS dan hasil akhir BSS. Grafik biru cenderung mengalami peningkatan pada ke-2 hingga data ke-10, dan kembali mengalami penurunan pada data ke-11. Grafik biru kembali mengalami peningkatan dari data ke-11 hingga data ke-20.

Maka dapat disimpulkan bahwa metode ELECTRE memiliki tingkat akurasi yang lebih besar serta cenderung lebih stabil dibanding dengan metode BSS pada pengujian dengan menggunakan kuota .

BAB VII
KESIMPULAN DAN SARAN

Bab ini membahas tentang kesimpulan dan saran dari penelitian yang telah dilakukan berdasarkan hasil perancngan, implementasi dan pengujian sistem.

7.1 Kesimpulan

Berdasarkan pada hasil perancangan, implementasi, dan pengujian yang telah dilakukan maka kesimpulan yang didapat adalah sebagai berikut:

a. Perancangan sistem pendukung keputusan seleksi penerimaan peserta didik baru menggunakan metode ELECTRE - *Weighted Product* dibuat berdasarkan hasil yang telah diperoleh dari tahap pengumpulan data analisa kebutuhan sistem berdasarkan hasil wawancara. Selanjutnya informasi tersebut digunakan untuk membuat desain sistem, baik dari segi model ataupun arsitektur.
b. Sistem pendukung keputusan seleksi penerimaan peserta didik baru dengan metode ELECTRE - *Weighted Product* diimplementasikan dengan menggunakan 4 kriteria masukan untuk proses perhitungan ELECTRE dan 21 kriteria masukan untuk proses perhitungan *Weighted Product*. Hasil keluaran sistem adalah rekomendasi siswa baru yang diterima berdasarkan kelompok peminatan tertentu.
c. Hasil pengujian fungsionalitas sistem pendukung keputusan seleksi penerimaan siswa baru dengan metode ELECTRE - *Weighted Product* memiliki tingkat kesesuaian presentase sebesar 100%. Sedangkan hasil pengujian akurasi sistem pendukung keputusan seleksi penerimaan siswa baru dengan metode ELECTRE dan *Weighted Prodoct* memiliki tingkat kesesuaian yang didasarkan pada data hasil seleksi penerimaan siswa baru SMA BSS pada tahun 2014/2015 dengan presentase tertinggi sebesar 84.37%. Hasil akurasi belum maksimal karena ada kebijakan-kebijakan SMA BSS dalam menentukan hasil akhir dari siswa yang diterima.

7.2 Saran

Saran yang diberikan untuk pengembangan sistem dalam penelitian selanjutnya, antara lain:

a. Dapat dikembangkan menjadi sebuah sistem yang lebih dinamis, sehingga dapat melakukan penambahan terhadap kriteria dan bobot, jalur masuk, dan kelas peminatan sehingga dapat menyesuaikan dengan kebutuhan SMA BSS selanjutnya.
b. Dapat dikembangan untuk proses seleksi penerimaan siswa baru pada jalur prestasi yang ada di SMA BSS beserta proses peminatannya.
c. Dapat dikembangkan dengan melakukan pengujian lebih lanjut menggunakan data perankingan akhir semester SMA BSS untuk membuktikan bahwa sistem telah benar-benar memilih siswa terbaik.

DAFTAR PUSTAKA

[1] Abidin, Taufik Buadi. *Accuracy Measure Precision, Recall & F-Measur*e (*PDF Document*). www.informatika.unsyiah.ac.id/tfa

[2] Adrianto, Joko. 2013. Sistem Pendukung Keputusan Pemilihan Jurusan Dengan Metode *Weighted Product* Untuk Siswa-Siswa SMA Negeri 1 SALE

[3] Akshareari, Syeril., Marwati, Rini., & Wijayanti, Utari. Sistem Pendukung Keputusan Pemilihan Sepatu dan Sandal dengan Metode Elimination Et Choix Traduisant La Realite (ELECTRE) (Studi Kasus pada Produsen Sepatu dan Sandal ”Obara Shoes” Cibaduyut Bandung). Indonesia.

[4] Alfita, Riza. 2011. Perancangan Sistem Pendukung Keputusan Penentuan Prioritas Produk Unggulan Daerah Menggunakan Metode Weighted Product (WP). Indonesia: Universitas Trunojoyo Madura

[5] Bohanec, Marko. *What is Decision Support?* Slovenia: Jozef Stefan Institute.

[6] *Brawijaya Smart School*. *Brawijaya Smart School* Universitas Brawijaya. http://bss.ub.ac.id/ diakses pada tanggal 9 Mei 2014.

[7] Buede, Dennis M. *The Engineering Design of Systems: Models and Method*. 2009. Wiley: 2nd Edition

[8] Caterino, Nicola, Iervolino, Iunio, Cosenza, Edoardo, & Manfredi, Gaetano. *Applicability and effectiveness of different decision making methods for seismic upgrading building structures*. Italy: Universita di Napoli Federico II.

[9] Chang, Yu-Hern, & Yeh, Chung-Hsing. *Evaluating airline competitiveness using multiattribute decision making. Omega The International Journal of Management Science*, Omega 29, 2001, pp 405-415

[10] Druzdzel, M. J., & Flynn, R. R. 2002. *Decision Support Systems*. Pittsburgh: University Of Pittsburgh.

[11] Dwijayanti, Yessy Febri. 2012. Sistem Pendukung Keputusan Untuk Menentukan Penjurusan di Sekolah Menengah Atas (Studi Kasus di SMA Negeri 1 Lebakwangi). Indonesia: Universitas Kuningan.

[12] E. Triantaphyllou, B. Shu, S. Nieto Sanchez,& T. Ray. *Multi-Criteria Decision Making: An Operations Research Approach. Encyclopedia of Electrical and Electronics Engineering*, (J.G. Webster, Ed.), John Wiley & Sons, New York, NY, Vol. 15, pp. 175-186, (1998).

[13] Exhadi, Baskoro Y.I. Sistem Pendukung Keputusan Penentuan Kelayakan Pengisian Bibit Ayam Broiler Dikandang Peternak menggunakan Metode AHP dan TOPSIS. 2013. Indonesia: Universitas Brawijaya.

[14] *Information Retrieval Lecture 7 - Evaluation in information Retrieval* (PDF Document). International Studies in Computational Linguistics

[15] Karnajaya, Anak Agung Alit. 2010. Rancang Bangun Sistem Pendukung Keputusan Pemilihan Sepeda Motor Honda Menggunakan Metode Electre Berbasis Web (Studi Kasus di PT. Lumenindo Gilang Cahaya). Indonesia: Stikom Surabaya.

[16] Li, Qing & Chen, Yu-Liu. *Modeling and Analysis of Enterprise and Information Systems From Requirements to Realization*. 2009. Higher Education Press, Beijing and Springer-Verlag GmbH Berlin Heidelberg.

[17] Matthews, M., Cole, J., & Gradecki, J. D. (2003). MySQL and Java developer's guide. John Wiley & Sons.

[18] Nugroho, A. (2009). Rekayasa Perangkat Lunak Menggunakan UML dan Java. Penerbit Andi.

[19] Parno, S.Kom., M.Msi. Lecture Notes: Sistem Inormasi Data Flow Diagram

[20] Powers, David M. W. *Evaluation*: *From Precision, Recall and F-Measure To ROC, Informedness, Markedness & Correlation*. *Journal of Machine Learning Technologies, Volume 2, Issue 1, 2011, pp-37-63*

[21] Pendidikan. Kamus Kamus Besar Bahasa Indonesia Online. http://kamusbahasaindonesia.org/pendidikan/mirip diakses pada tanggal 9 Mei 2014.

[22] Rizqi, A. 2009. Sistem Pendukung Keputusan Untuk Pemilihan Bahan Dasar Obat Alternatif Dengan Metode Electre Dan Topsis. Indonesia : Stikom Surabaya.

[23] Rudiansih, N., Suprapto, Regasari, R.M.P. Sistem Pendukung Keputusan untuk Proses Penentuan rumah Tangga Miskin menggunakan Metode Weighted Product. Indonesia: Universitas Brawijaya.

[24] Salinan Lampiran Peraturan Menteri Pendidikan dan Kebudayaan Nomor 69 Tahun 2013 tentang Kerangka Dasar dan Struktur Kurikulum Sekolah Menengah Atas/Madrasah Aliyah. 2013.

[25] Seban, Siddiq Fahriady. 2013. Sistem Pendukung Keputusan Penentuan Lokasi Kantor Bank Indonesia Gorontalo Menggunakan Metode Weighted Product. Gorontalo : Program Studi S1 Sistem Informasi Jurusan Teknik Informatika Fakultas Teknik Universitas Negeri Gorontalo.

[26] Sianturi, Ingot Seen. Sistem Pendukung Keputusan Untuk Menentukan Pmeilihan Jurusan Siswa Dengan Menggunakan Metode Weighted Product (Studi Kasus: SMA Swasta HKBP Doloksanggul). 2013. Indonesia: STMIK Budidarma Medan, Vol. 1, No. 1

[27] Turban, Efraim, Jay E Aronson dan Ting-Peng Liang. 2004, *Decision Support Systems and Intelligent Systems* – 7th edition. Prentice Hall PTR.

[28] Triantaphyllou, Evangelos. 2000. Multi-criteria Decision Making Methods: A Comparative Study (Applied Optimization). Springer.

[29] Dolan , James G. Multi-criteria clinical decision support: A primer on the use of multipple methods to promote evidence-based, patient-centered healthcare. 2010. National Institute of Health: 3(4): 229–248. ROC