



**PENGEMBANGAN SISTEM INFORMASI PENDUKUNG KEPUTUSAN
SISWA ASUH SEBAYA DENGAN MENGGUNAKAN METODE SIMPLE
MULTY-ATRIBUT RATING TECHNIQUE DI BANYUWANGI**

(Studi Kasus SMAN 1 Giri Banyuwangi)

SKRIPSI

oleh

Zul Fahmi Akmal

NIM 112410101003

**PROGRAM STUDI SISTEM INFORMASI
UNIVERSITAS JEMBER**

2015



**PENGEMBANGAN SISTEM INFORMASI PENDUKUNG KEPUTUSAN
SISWA ASUH SEBAYA DENGAN MENGGUNAKAN METODE SIMPLE
MULTY-ATRIBUT RATING TECHNIQUE DI BANYUWANGI**

(Studi Kasus SMAN 1 Giri Banyuwangi)

SKRIPSI

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat
untuk menyelesaikan Program Studi Sistem Informasi (S1)
dan mencapai gelar Sarjana Komputer

oleh

Zul Fahmi Akmal
NIM 112410101003

**PROGRAM STUDI SISTEM INFORMASI
UNIVERSITAS JEMBER**

2015

PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

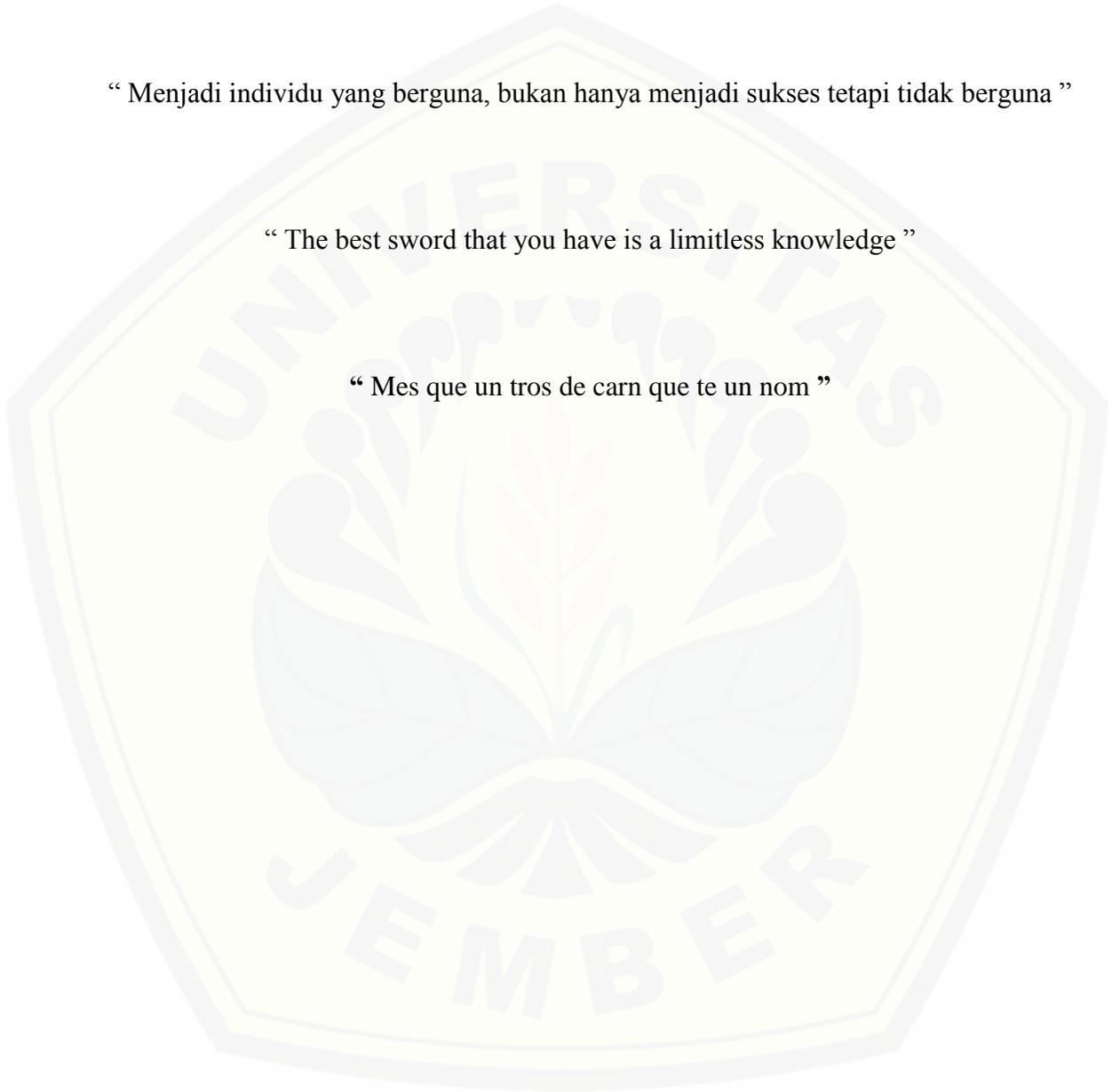
1. Allah SWT atas kuasaNya yang senantiasa memberikan rahmat, kemudahan, dan hidayahNya sehingga saya dapat terus berkarya hingga saat ini;
2. Alm. Ayah, ibu, adik, dan uti saya yang selalu menjadi inspirasi dan motifasi dalam menyelesaikan tugas akhir skripsi ini;
3. Keluarga besar Heru Santoso yang selalu memberikan dukungan dan motifasi untuk melanjutkan kuliah dan hingga lulus sekarang;
4. Bening Putri Jatningsih dan keluarga yang sudah menjadi keluarga kedua dan memberikan motifasi untuk menyelesaikan kuliah dengan cepat;
5. Bapak dan ibu dosen Program Studi Sistem Informasi terutama bapak Drs. Antonius Cahya P, M.App., Sc., Ph.D dan ibu Windi Eka Yulia Retnani, S.Kom., MT sebagai dosen pembimbing selama melakukan penelitian dan pengerjaan skripsi ini;
6. SMAN 1 Giri Banyuwangi selaku pemberi data dan masukan kepada pengerjaan skripsi hingga terselesaikan;
7. Keluarga besar Sistem Informasi tahun angkatan 2011;
8. Almamater Program Studi Sistem Informasi Universitas Jember;

MOTTO

“ Menjadi individu yang berguna, bukan hanya menjadi sukses tetapi tidak berguna ”

“ The best sword that you have is a limitless knowledge ”

“ Mes que un tros de carn que te un nom ”



PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Zul Fahmi Akmal

NIM : 112410101003

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Pengembangan Sistem Informasi Pendukung Keputusan Siswa Asuh Sebaya Dengan Menggunakan Metode Simple Multy-Atribut Rating Technique Di Banyuwangi ” adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada instansi mana pun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa ada tekanan dan paksaan dari pihak mana pun serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, Desember 2015

Yang menyatakan,

Zul Fahmi Akmal

NIM 112410101003

SKRIPSI

**PENGEMBANGAN SISTEM INFORMASI PENDUKUNG KEPUTUSAN
SISWA ASUH SEBAYA DENGAN MENGGUNAKAN METODE SIMPLE
MULTY-ATRIBUT RATING TECHNIQUE DI BANYUWANGI
(STUDI KASUS : SMAN 1 GIRI BANYUWANGI)**

Oleh

Zul Fahmi Akmal

NIM 112410101003

Pembimbing

Dosen Pembimbing Utama : Drs. Antonius Cahya P, M.App., Sc., Ph.D

NIP. 1969092819930121001

Dosen Pembimbing Anggota : Windi Eka Yulia Retnani, S.Kom., MT

NIP. 198403052010122002

PENGESAHAN PEMBIMBING

Skripsi berjudul Pengembangan Sistem Informasi Pendukung Keputusan Siswa Asuh Sebaya Dengan Menggunakan Metode Simple Multy-Atribut Rating Technique Di Banyuwangi telah diuji dan disahkan pada:

Hari, tanggal : 3 Desember 2015

Tempat : Program Studi Sistem Infrmasi Universitas Jember

Pembimbing I,

Pembimbing II,

Drs. Antonius Cahya P, M.App., Sc., Ph.D
NIP. 1969092819930121001

Windi Eka Yulia R, S.Kom., MT
NIP. 198403052010122002

PENGESAHAN PENGUJI

Skripsi berjudul Pengembangan Sistem Informasi Pendukung Keputusan Siswa Asuh Sebaya Dengan Menggunakan Metode Simple Multy-Atribut Rating Technique Di Banyuwangi telah diuji dan disahkan pada:

Hari, tanggal : 3 Desember 2015

Tempat : Program Studi Sistem Infrmasi Universitas Jember

Tim Penguji:

Penguji I,

Penguji II,

Anang Andriyanto, S.T., M.T.

NIP. 196906151997021002

Muhammad Arief, S.Kom., M.Komp.

NIP. 198101232010121003

Mengesahkan

Ketua Program Studi,

Prof. Drs. Slamın, M.Comp.Sc, Ph.D

NIP 196704201992011001

Pengembangan Sistem Informasi Pendukung Keputusan Siswa Asuh Sebaya Dengan Menggunakan Metode Simple Multy-Atribut Rating Technique Di Banyuwangi (Studi Kasus : SMAN 1 GIRI BANYUWANGI)

Zul Fahmi Akmal

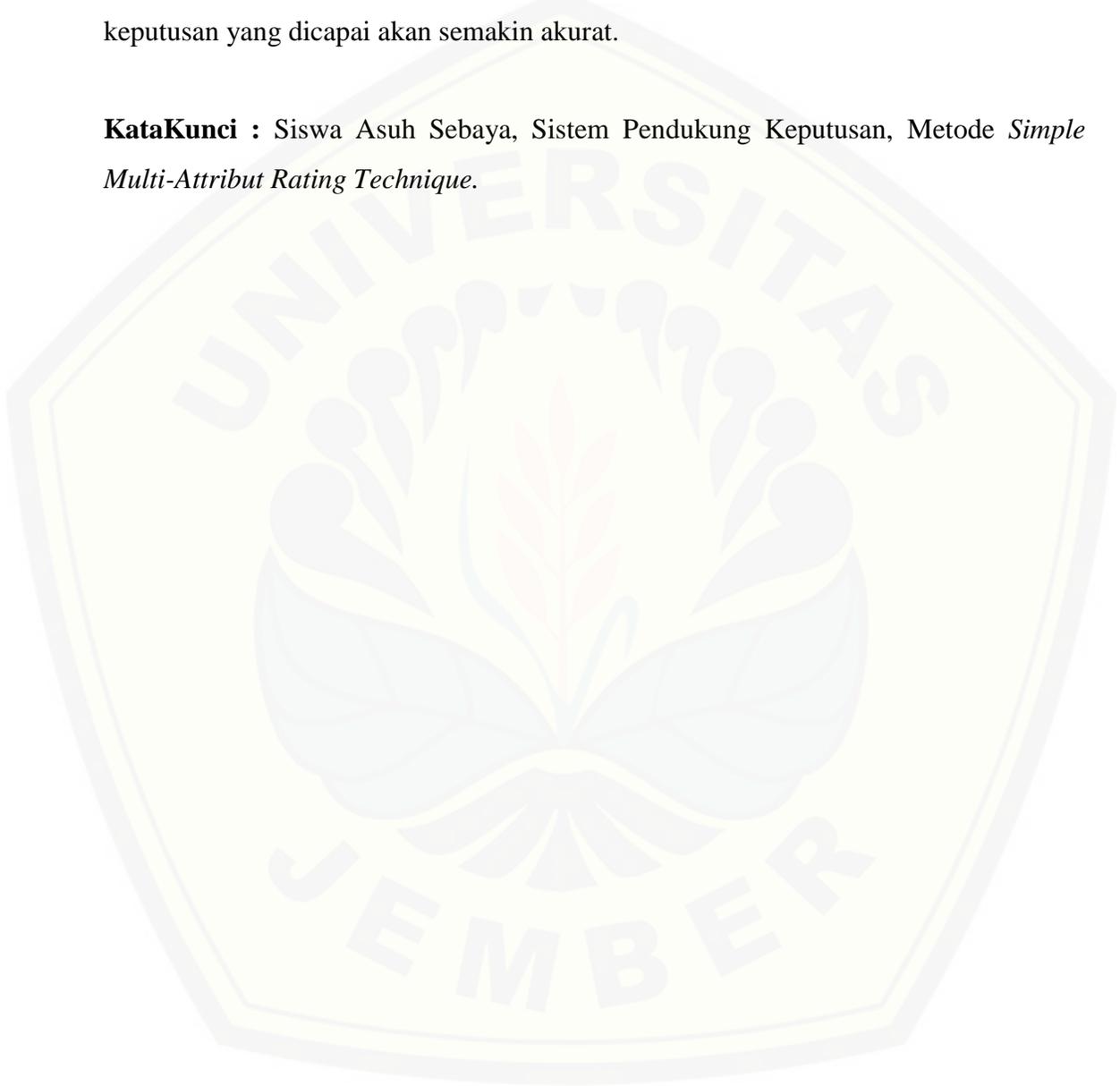
Jurusan Sistem Informasi, Program Studi Sistem informasi, Universitas Jember

ABSTRAK

Banyuwangi merupakan salah satu kabupaten paling ujung timur pulau jawa yang paling luas wilayah kabupatennya. Jumlah penduduk Banyuwangi mencapai 1,56 juta jiwa. Pemerintah Banyuwangi menghadapi banyak permasalahan perkembangan jumlah penduduknya terutama pada bidang pendidikan. Pemerintahan kabupaten Banyuwangi mencanangkan banyak program yang dilaksanakan guna meningkatkan wilayah Banyuwangi dengan memperkuat dalam semua pilar yang ada di Banyuwangi mulai dari ekonomi, lingkungan, kesehatan, pendidikan, pelayanan umum, pariwisata dan lainnya. Salah satu pilar yang didukung penuh oleh Pemkab Banyuwangi adalah pada pilar pendidikan. Salah satu masalah pada pendidikan di Banyuwangi dikarenakan kurangnya biaya untuk mendapat pendidikan. Banyuwangi mencanangkan program pendidikan yang bernama Siswa Asuh Sebaya (SAS) pada sekolah tingkat dasar (SD) sampai tingkat atas (SMA). Gerakan SAS adalah gerakan siswa yang berasal dari keluarga mampu secara ekonomi membantu siswa dari keluarga kurang mampu. Di setiap sekolah siswa menggalang dana secara sukarela untuk membantu biaya pendidikan temannya yang kurang mampu secara sukarela. Cara pemilihan seperti ini sering kurang tepat karena hanya melakukan pengamatan secara umum dengan melihat kondisi orang tua serta lingkungannya. Hal ini menyebabkan ketidak-akuratan pemberian dana SAS, Oleh karena itu diperlukan suatu sistem informasi pendukung keputusan untuk menentukan siswa yang berhak mendapatkan bantuan SAS secara akurat. Dalam permasalahan pengambilan keputusan pemberian bantuan SAS ini, salah satu metode yang tepat digunakan

adalah metode *Simple Multi-Attribute Rating Technique*(SMART). Karena metode ini dapat memaksimalkan banyaknya *criteria* yang ada dalam pendukung keputusan SAS. Dengan semakin banyak *criteria* yang dipakai, maka pengambilan hasil keputusan yang dicapai akan semakin akurat.

KataKunci : Siswa Asuh Sebaya, Sistem Pendukung Keputusan, Metode *Simple Multi-Attribut Rating Technique*.



RINGKASAN

Pengembangan Sistem Informasi Pendukung Keputusan Siswa Asuh Sebaya Dengan Menggunakan Metode Simple Multy-Atribut Rating Technique Di Banyuwangi (Studi Kasus : SMAN 1 GIRI BANYUWANGI); Zul Fahmi Akmal, 112410101003; 2015; halaman; Program Studi Sistem Informasi Universitas Jember.

Sistem informasi pendukung keputusan di SMAN 1 Giri yang dibangun dalam penelitian ini merupakan sistem yang mampu melakukan rekomendasi siswa yang akan di usulkan untuk dipilih menjadi siswa asuh sebaya dengan mengimplementasikan metode SMART didalamnya. Adapun criteria yang dipakai dalam penelitian ini meliputi total pendapatan orang tua, jumlah uang jajan, kendaraan kesekolah, jumlah saudara, ranking kelas, status anak, dan peralatan sekolah yang dipakai. Kriteria-kriteria tersebut akan dianalisis dan digunakan dalam sistem dengan mengimplementasikan pada metode SMART, sehingga dalam sistem yang dibangun ini memiliki menu proses perhitungan metode SMART. Selain itu sistem ini menyediakan fitur untuk guru dan osis dalam mengolah data calon pendaftar siswa asuh sebaya. Guru diberikan menu dan tanggung jawab dalam mengakses sistem dengan menginputkan siapa saja calon penerima siswa asuh sebaya yang akan mendaftar. Sedangkan osis diberi akses penuh pada proses pemilihan dan penilaian yang diataranya dapat mengedit bobot criteria yang dipakai dan menambah atribut yang digunakan pada setiap criteria. Sesuai dengan tujuan dibangunnya sistem ini, sistem ini dapat menampilkan sepuluh nama calon siswa yang di rekomendasikan oleh sistem untuk mendapatkan dana bantuan siswa asuh sebaya.

PRAKATA

Puji syukur kehadirat Allah SWT atas segala rahmat dan karuniaNya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Pengembangan Sistem Informasi Pendukung Keputusan Siswa Asuh Sebaya Dengan Menggunakan Metode Simple Multy-Atribut Rating Technique Di Banyuwangi (Studi Kasus : SMAN 1 GIRI BANYUWANGI)”. Skripsi ini bertujuan untuk memenuhi salah satu syarat menyelesaikan pendidikan strata satu (S1) pada Program Studi Sistem Informasi Universitas Jember.

Penyusunan skripsi ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis menyampaikan terimakasih kepada :

1. Allah SWT atas kuasaNya yang senantiasa memberikan rahmat, kemudahan, dan hidayahNya hingga dapat terselesaikan sebuah tugas akhir ini.
2. Alm. Ayah, ibu, adik, dan uti saya yang selalu menjadi inspirasi dan motivasi dalam menyelesaikan tugas akhir skripsi ini;
3. Keluarga besar Heru Santoso yang selalu memberikan dukungan dan motivasi untuk melanjutkan kuliah dan hingga lulus sekarang;
4. Bening Putri Jatningsih dan keluarga yang sudah menjadi keluarga kedua dan memberikan motivasi untuk menyelesaikan kuliah dengan cepat;
5. Bapak dan ibu dosen Program Studi Sistem Informasi terutama bapak Drs. Antonius Cahya P, M.App., Sc., Ph.D dan ibu Winda Eka Yulia Retnani, S.Kom., MT sebagai dosen pembimbing selama melakukan penelitian dan pengerjaan skripsi ini;
6. SMAN 1 Giri Banyuwangi selaku pemberi data dan masukan kepada pengerjaan skripsi hingga terselesaikan;
7. Keluarga besar Sistem Informasi tahun angkatan 2011;
8. Semua teman kos d'jokolers yang selalu hadir disaat boring mengerjakan skripsi.

DAFTAR ISI

Contents

PERSEMBAHAN.....	iii
MOTTO	iv
PERNYATAAN.....	v
SKRIPSI.....	vi
PENGESAHAN PEMBIMBING.....	vii
PENGESAHAN PENGUJI.....	viii
ABSTRAK.....	ix
RINGKASAN	xi
PRAKATA.....	xii
DAFTAR ISI.....	xiii
DAFTAR GAMBAR	xvii
DAFTAR TABEL.....	xix
BAB 1. PENDAHULUAN	193
1.1 Latar Belakang	193
1.2 Rumusan Masalah	195
1.3 Tujuan	195
1.4 Batasan Masalah.....	195
1.5 Sistematika Penulisan	196
BAB 2. TINJAUAN PUSTAKA	198
2.1 Hasil Penelitian Terdahulu.....	198

2.2 Relevansi Metode <i>Simple Multi-Attribute Rating Technique</i> terhadap pendukung keputusan penerimaan SAS.....	199
2.3 Sistem Pendukung Keputusan (<i>Decision Support System</i>).....	200
2.4 Simple Multy-Attribute Rating Technique (SMART).....	200
BAB 3. METODE PENELITIAN.....	205
3.1 Jenis Penelitian.....	205
3.2 Pengembangan Sistem	205
3.2.1 Analisis Kebutuhan	206
3.2.2 Desain.....	208
3.2.3 Implementasi	209
3.2.4 Pengujian.....	209
3.2.5 Pemeliharaan	211
BAB 4. PENGEMBANGAN SISTEM.....	212
4.1 Deskripsi Umum Sistem	212
4.1.1 SOP (Statement Of Perpose).....	212
4.1.2 Fungsi Sistem.....	213
4.2 Analisa Kebutuhan Sistem	214
4.2.1 Kebutuhan Fungsional	214
4.2.2 Kebutuhan Non-Fungsional	215
4.3 Desain Sistem.....	215
4.3.1 Business Process	215
4.3.2 Use Case Diagram.....	216
4.3.3 Class Diagram	230

4.3.4 Entity Relationship Diagram.....	230
4.4 Pengkodean Sistem	234
4.5 Pengujian Sistem.....	234
4.5.1 Metode <i>White Box</i>	234
4.5.2 Metode <i>Black Box</i>	261
BAB 5 HASIL DAN PEMBAHASAN.....	262
5.1 Hasil Implementasi Coding Pada Sistem Informasi Siswa Asuh Sebaya	262
5.1.1 Halaman Login User	262
5.1.2 Halaman User Guru.....	263
5.1.3 Halaman input SAS (guru).....	264
5.1.4 Halaman Penerima SAS (Guru).....	265
5.1.5 Halaman Home OSIS	265
5.1.6 Halaman Update Kriteria	266
5.1.7 Halaman Update Bobot.....	270
5.1.8 Halaman Penghitungan Metode Siswa Asuh Sebaya	271
5.1.9 Halaman Penerima Siswa Asuh Sebaya.....	272
5.2 Implementasi Metode SMART Pada Sistem Informasi Siswa Asuh Sebaya .	272
5.3 Pengujian Sistem Informasi Siswa Asuh Sebaya.....	276
5.3.1 Perhitungan Manual Metode SMART	277
5.4 Perbandingan Perhitungan SMART Dengan Data OSIS	296
5.4.1 Data Pemilihan Siswa Asuh Sebaya Oleh OSIS	296
5.4.2 Data Pemilihan Siswa Asuh Sebaya Oleh OSIS	297
5.5 Pembahasan.....	304

5.5.1 Pembahasan Hasil Implementasi Metode SMART Pada Sistem Informasi Siswa Asuh Sebaya	305
5.5.2 Pembahasan Hasil Perbandingan Hasil Sistem Dengan Menggunakan Tiga Perbedaan Nilai Metode Yang Dipakai.....	307
5.4.3 Pembahasan Hasil Pengujian Sistem	309
5.4.4 Pembahasan Cara Menyesuaikan Bobot Penilaian Yang Akan Digunakan	311
BAB 6. PENUTUP	312
6.1 Kesimpulan	312
6.2 Saran.....	314
DAFTAR PUSTAKA	316
LAMPIRAN.....	317

DAFTAR GAMBAR

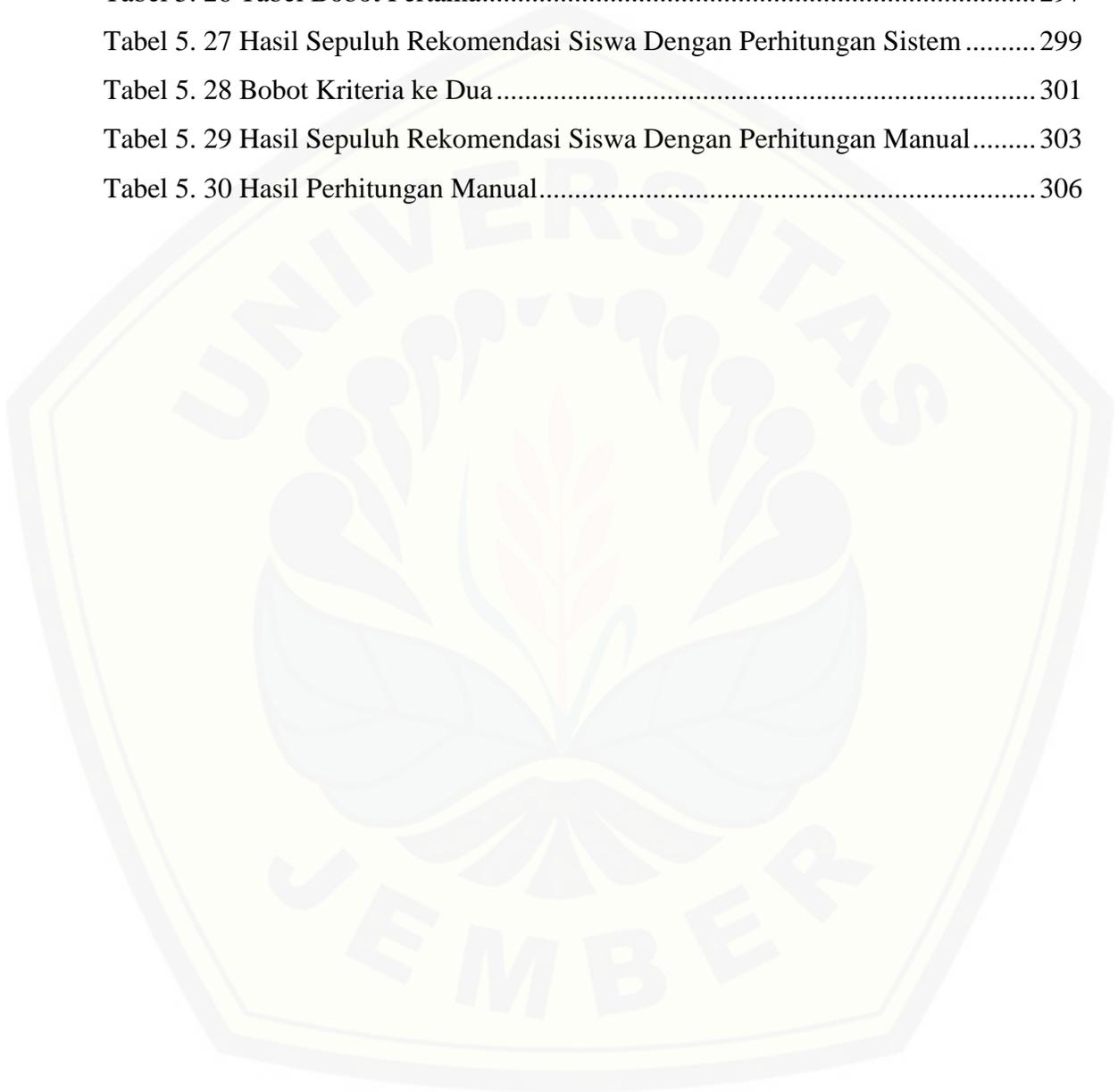
Gambar 2. 1 Diagram Alir Metode SMART	202
Gambar 3. 1 Metode Waterfall.....	206
Gambar 3. 2 Sitem kerja dari Teknik Pengujian <i>White Box</i>	210
Gambar 3. 3 Sitem kerja dari Teknik Pengujian <i>Black Box</i>	210
Gambar 4. 1 Bussiness Procces Sistem Informasi Siswa Asuh Sebaya.....	216
Gambar 4. 2 Use Case Diagram Sistem Informasi Siswa Asuh Sebaya.....	218
Gambar 4. 3 Activity Diagram.....	228
Gambar 4. 4 Sequence Diagram	231
Gambar 4. 5 Entity Relationship Diagram.....	232
Gambar 4. 6 Class Diagram	233
Gambar 5. 1 Osis Login	263
Gambar 5. 2 Guru Login	263
Gambar 5. 3 Home Guru.....	264
Gambar 5. 4 Input Calon Siswa Asuh Sebaya	264
Gambar 5. 5 Penerima SAS	265
Gambar 5. 6 Home OSIS	266
Gambar 5. 7 Kriteria Peralatan Sekolah.....	267
Gambar 5. 8 Kriteria Pendapatan Orang Tua.....	267
Gambar 5. 9 Kriteria Kendaraan Ke Sekolah.....	268
Gambar 5. 10 Kriteria Jumlah Saudara.....	268
Gambar 5. 11 Kriteria Status Anak.....	269
Gambar 5. 12 Kriteria Jumlah Uang Jajan.....	269
Gambar 5. 13 Kriteria Rangking Kelas.....	270
Gambar 5. 14 Update Bobot	271
Gambar 5. 15 Perhitungan Smart.....	271
Gambar 5. 16 Penerima Siswa Asuh Sebaya	272
Gambar 5. 17 Mengambil Nama Siswa	273
Gambar 5. 18 Menampilkan Nama Pada Form.....	273

Gambar 5. 19 Nilai Utility	274
Gambar 5. 20 Normalisasi	274
Gambar 5. 21 Nilai Sementara	275
Gambar 5. 22 Hitung Hasil	275
Gambar 5. 23 Contoh Data SAS	276
Gambar 5. 24 User Guru Menginputkan Peserta Siswa Asuh Sebaya.....	289
Gambar 5. 25 User OSIS Memberi Bobot Pada Kriteria	290
Gambar 5. 26 User OSIS Menginputkan Atribut Pada Kriteria Uang Jajan.....	290
Gambar 5. 27 User OSIS Menginputkan Atribut Pada Kriteria Status Anak	291
Gambar 5. 28 User OSIS Menginputkan Atribut Pada Kriteria Rangking Kelas.....	291
Gambar 5. 29 User OSIS Menginputkan Atribut Pada Kriteria Peralatan Sekolah..	292
Gambar 5. 30 User OSIS Menginputkan Atribut Pada Kriteria Kendaraan Sekolah	292
Gambar 5. 31 User OSIS Menginputkan Atribut Pada Kriteria Jumlah Saudara	293
Gambar 5. 32 User OSIS Menginputkan Atribut Pada Kriteria Total Pendapatan...	293
Gambar 5. 33 User OSIS Melakukan Perhitungan Metode SMART	294
Gambar 5. 34 Melihat 10 Rekomendasi Siswa Penerima SAS.....	294
Gambar 5. 35 Hasil Perhitungan Dengan Sistem.....	295
Gambar 5. 36 User OSIS Mengubah Nilai Bobot Perhitungan Kedua	298
Gambar 5. 37 User OSIS Melakukan Perhitungan Metode SMART	299
Gambar 5. 38 Melihat 10 Rekomendasi Siswa Penerima SAS Dari Sistem.....	299
Gambar 5. 39 User OSIS Mengubah Nilai Bobot Perhitungan Kedua	302
Gambar 5. 40 User OSIS Melakukan Perhitungan Metode SMART	302
Gambar 5. 41 Melihat 10 Rekomendasi Siswa Penerima SAS Dari Sistem.....	303
Gambar 5. 42 Hasil Perhitungan Dengan Sistem SAS	307

DAFTAR TABEL

Tabel 2. 1 Kriteria Penilaian	203
Tabel 4. 1 Diskripsi pembagian sistem	219
Tabel 4. 2 Deskripsi Use Case Sistem	219
Tabel 4. 3 Skenario Penghitungan Smart	225
Tabel 4. 4 Penerima SAS	229
Tabel 5. 1 Bobot Kriteria	278
Tabel 5. 2 Atribut Total Pendapatan	278
Tabel 5. 3 Atribut Jumlah Saudara	279
Tabel 5. 4 Atribut Kendaraan Ke Sekolah	279
Tabel 5. 5 Atribut Peralatan Sekolah	279
Tabel 5. 6 Atribut Rangking Kelas	280
Tabel 5. 7 Atribut Status Anak	280
Tabel 5. 8 Atribut Uang Jajan Sekolah	280
Tabel 5. 9 Siswa Pertama	281
Tabel 5. 10 Siswa Kedua	281
Tabel 5. 11 Siswa Ketiga	282
Tabel 5. 12 Siswa Keempat	282
Tabel 5. 13 Siswa Kelima	283
Tabel 5. 14 Siswa Keenam	283
Tabel 5. 15 Siswa Ketujuh	284
Tabel 5. 16 Siswa Kedelapan	284
Tabel 5. 17 Siswa Kesembilan	285
Tabel 5. 18 Siswa Kesepuluh	285
Tabel 5. 19 Siswa Kesebelas	286
Tabel 5. 20 Siswa Keduabelas	286
Tabel 5. 21 Siswa Ketigabelas	287
Tabel 5. 22 Siswa Keempatbelas	287
Tabel 5. 23 Siswa Kelimabelas	288

Tabel 5. 24 Hasil Sepuluh Rekomendasi Siswa Dengan Perhitungan Manual.....	288
Tabel 5. 25 Hasil Perbandingan Perhitungan Manual Dengan Sistem	295
Tabel 5. 26 Tabel Bobot Pertama.....	297
Tabel 5. 27 Hasil Sepuluh Rekomendasi Siswa Dengan Perhitungan Sistem	299
Tabel 5. 28 Bobot Kriteria ke Dua	301
Tabel 5. 29 Hasil Sepuluh Rekomendasi Siswa Dengan Perhitungan Manual.....	303
Tabel 5. 30 Hasil Perhitungan Manual.....	306



BAB 1. PENDAHULUAN

Bab ini merupakan bab awal dari laporan tugas akhir. Pada bab ini akan dibahas tentang latar belakang, perumusan masalah, tujuan dan manfaat, batasan masalah, dan sistematika penulisan.

1.1 Latar Belakang

Banyuwangi merupakan salah satu kabupaten paling ujung timur pulau Jawa yang paling luas wilayah kabupatennya. Jumlah penduduk Banyuwangi mencapai 1,56 juta jiwa. Pemerintah Banyuwangi menghadapi banyak permasalahan perkembangan jumlah penduduknya terutama pada bidang pendidikan. Angka putus sekolah di Banyuwangi masih besar di setiap tingkat sekolah. Untuk tingkat SD/MI, angka putus sekolah 0,03 persen pada 2013. Pada tingkat SMP/MTs, angka putus sekolah 0,42 persen pada 2013. Pada SMA/SMK/MA, angka putus sekolah 0,83 persen pada 2013 (Pekab Banyuwangi, 2014).

Pemerintahan kabupaten Banyuwangi mencanangkan banyak program yang dilaksanakan guna meningkatkan wilayah Banyuwangi dengan memperkuat dalam semua pilar yang ada di Banyuwangi mulai dari ekonomi, lingkungan, kesehatan, pendidikan, pelayanan umum, pariwisata dan lainnya. Salah satu pilar yang didukung penuh oleh Pekab Banyuwangi adalah pada pilar pendidikan. Salah satu masalah pada pendidikan di Banyuwangi dikarenakan kurangnya biaya untuk mendapat pendidikan. Banyuwangi mencanangkan program pendidikan yang bernama Siswa Asuh Sebaya (SAS) pada sekolah tingkat dasar (SD) sampai tingkat atas (SMA). Hal ini digunakan untuk menekan angka putus sekolah yang ada di Banyuwangi.

Gerakan SAS adalah gerakan siswa yang berasal dari keluarga mampu secara ekonomi membantu siswa dari keluarga kurang mampu. Di setiap sekolah siswa menggalang dana secara sukarela untuk membantu biaya pendidikan temannya yang

kurang mampu secara sukarela. Gerakan SAS untuk membangun kepedulian dan modal sosial diantara generasi muda di Banyuwangi. Jumlah rupiah untuk SAS diberikan secara ikhlas, tidak diberikan jumlah minimal untuk ikut membantu. Pada awal diluncurkan ditahun 2011, total dana SAS yang terkumpul Rp 293 juta. Pada tahun 2013 dana yang terkumpul seluruhnya melonjak menjadi Rp 1,6 miliar dan bertambah pada tahun ketahun. Total penerima manfaat mencapai lebih dari 6.000 siswa dari 309 sekolah (Pekab Banyuwangi, 2014).

Pengelolaan dana SAS dilakukan dari siswa, oleh siswa dan untuk siswa. Pemerintah Banyuwangi hanya bertindak sebagai pengawas jalannya program. Pemilihan siswa yang mendapatkan bantuan SAS dipilih sepenuhnya oleh pihak Organisasi Siswa Intra Sekolah atau biasa disebut OSIS yang ada di sekolah masing-masing. Sekolah melakukan pemilihan dengan cara pengamatan pada siswa yang dianggap kurang mampu untuk memenuhi biaya sekolahnya. Cara pemilihan seperti ini sering kurang tepat karena hanya melakukan pengamatan secara umum dengan melihat kondisi orang tua serta lingkungannya. Hal ini menyebabkan ketidakakuratan pemberian dana SAS, Oleh karena itu diperlukan suatu sistem informasi pendukung keputusan untuk menentukan siswa yang berhak mendapatkan bantuan SAS secara akurat.

Ada banyak metode pendukung keputusan untuk digunakan dalam pendukung keputusan. Dalam permasalahan pendukung keputusan pemberian bantuan SAS ini, salah satu metode yang tepat digunakan adalah metode *Simple Multi-Attribute Rating Technique* (SMART). Karena metode ini dapat memaksimalkan banyaknya *criteria* yang ada dalam pendukung keputusan SAS. Dengan semakin banyak *criteria* yang dipakai, maka pendukunghasil keputusan yang dicapai akan semakin akurat. Dalam permasalahan pemilihan keputusan untuk SAS banyak permasalahan yang sifatnya *multicriteria* dan *multiobjecty* yang sangat cocok di selesaikan dengan metode SMART.

1.2 Rumusan Masalah

Dengan mempertimbangkan latar belakang masalah diatas, dapat dirumuskan masalah yaitu :

1. Bagaimana mengimplementasikan metode *Simple Multi-Attribute Rating Technique* untuk penentuan penerima bantuan Siswa Asuh Sebaya?
2. Bagaimana merancang dan membangun sebuah sistem pendukung keputusan penentuan penerima bantuan Siswa Asuh Sebaya dengan metode SMART?

1.3 Tujuan

Tujuan yang ingin dicapai dalam penelitian ini adalah :

1. Mengimplementasikan metode Simple Multi-Attribute Rating Technique pada sistem informasi pendukung keputusan penerima bantuan Siswa Asuh Sebaya.
2. Merancang dan Membangun sebuah sistem informasi pendukung keputusan penerima bantuan Siswa Asuh Sebaya.
3. Menghasilkan pemilihan yang baik dan pendukung keputusan yang cepat dalam memilih siswa asuh sebaya (SAS).

1.4 Batasan Masalah

Agar tidak terjadi penyimpangan dalam prosen penelitian dan pengembangan Sistem Informasi Siswa Asuh Sebaya, maka ditetapkan beberapa batasan permasalahan. Adapun batasan masalah dalam penelitian dan pembangunan Sistem Informasi Siswa Asuh Sebaya ini yaitu :

1. Sistem hanya dapat digunakan untuk membantu pendukung keputusan pada SAS.
2. Sistem dibangun berbasis desktop dengan bahasa pemrograman JAVA
3. Menggunakan database MySQL
4. Sistem hanya dapat di kelolah oleh OSIS di sekolah
5. Jenis kreteria di tentukan oleh sekolah tanpa dapat dirubah.

6. Data yang di gunakan adalah data siswa yang ada di SMAN 1 Giri Banyuwangi
7. Dalam melakukan penilaian hanya ketua OSIS yang dapat memegang hak akses pada Sistem Informasi Siswa Asuh Sebaya
8. Guru hanya dapat melakukan *input* calon siswa yang akan diseleksi untuk mendapatkan data calon Siswa Asuh Sebaya.

1.5 Sistematika Penulisan

Adapun sistematika penulisan skripsi ini adalah sebagai berikut :

1. Pendahuluan

Bab kesatu ini membahas serta menguraikan tentang latar belakan, perumusan masalah, tujuan, manfaat, batasan masalah, dan sistematika penulisan skripsi yang dalam bab dan subbab tersendiri secara eksplisit.

2. Tinjauan Pustaka

Bab ini merupakan tinjauan terhadap hasil penelitian yang telah dilakukan peneliti terdahulu yang berkaitan dengan masalah yang dibahas, kajian metode, landasan materi dan konsep yang berkaitan dengan penelitian.

3. Metode Penelitian

Bab ini menjabarkan tentang metode penelitian, metode pengumpulan data, metode analisis data, pengembangan sistem yang dibangun, serta waktu dan tempat yang digunakan dalam proses penelitian.

4. Pengembangan Sistem

Bab ini berisikan tentang langkah – langkah yang di tempuh untuk proses perancangan sistem yang hendak dibangun yang meliputi desain sistem, pengkodean sampai dengan pengujian sistem.

5. Hasil dan Pembahasan

Bab ini menguraikan tentang pemecahan masalah yang ada dalam penelitian melalui suatu analisi yang dituangkan dalam bentuk deskripsi yang berupa tabel serta gambar untuk membermudahkan pemahaman hasil dari penelitian.

6. Penutup

Bab ini terdiri dari kesimpulan yang didapat dalam proses penelitian dan saran untuk peneliti selanjutnya.



BAB 2. TINJAUAN PUSTAKA

Pada bagian bab ini dijelaskan tinjauan yang berkaitan dengan masalah yang dibahas dalam proses penelitian, kajian teori yang berkaitan dengan masalah, dan hasil penelitian-penelitian terdahulu.

2.1 Hasil Penelitian Terdahulu

Pada artikel yang berjudul ‘Metode SMART Pada Sistem Pendukung Keputusan Pemilihan Pembelian Mobil Keluarga’ (Atiqah, 2013) menjelaskan bahwa metode SMART cocok untuk menyelesaikan masalah yang sifatnya *multicriteria* dan *multiobject*. Adapun kriteria yang digunakan dalam pemilihan mobil keluarga mulai dari harga, fasilitas, kebutuhan dan kualitasnya. Tahapan untuk penilaiannya pertama ditentukan beberapa mobil yang akan dipilih, kemudian dipilih empat mobil dari beberapa mobil tersebut. Penilaian pada setiap kriteria dan pada setiap atributnya diberikan nilai pembobotan sendiri sehingga hasil yang didapat maksimal. Setiap kriteria memiliki nilai sendiri dan setiap atribut di berikan bobot sendiri, sehingga perbandingan nilai yang diperoleh bukan hanya dari bobot atributnya saja tetapi juga dihitung dengan nilai kriteria yang di berikan.

Penelitian pada artikel tersebut terdapat kekurangan dalam memaksimalkan hasil yang diperoleh. Kriteria yang digunakan dalam mengambil keputusan sangat kurang banyak dan kurang berfariatif, sehingga perbandingan antar atribut sangat dekat dengan nilai atribut lainnya. Pemberian nilai bobot yang terlalu dekat tersebut menyebabkan kurang maksimalnya hasil akhir penilaian keputusan yang didapat.

Pada kajian artikel tersebut metode SMART digunakan untuk menghadapi sebuah permasalahan yang memiliki perbandingan banyak kriteria seperti pemilihan mobil keluarga. Pada penelitian sistem pendukung keputusan untuk bantuan SAS ini juga memiliki persamaan banyak kriteria yang akan di gunakan (Tabel 1). Banyaknya kriteria tersebut penggunaan metode SMART untuk digunakan dalam mendukung keputusan bantuan dana SAS sangat cocok, serta dapat diterapkan

dengan baik dalam Sistem informasi pendukung keputusan bantuan SAS di Banyuwangi.

2.2 Relevansi Metode *Simple Multi-Attribute Rating Technique* terhadap pendukung keputusan penerimaan SAS

SMART merupakan metode pendukung keputusan yang fleksibel. SMART lebih banyak digunakan karena kesederhanaanya dalam merespon kebutuhan pembuat keputusan dancaranya menganalisa sebuah permasalahan pengambil keputusan. Analisa yang terlibat adalah transparan sehingga metode ini memberikan pemahaman masalah yang tinggi dan dapat diterima oleh pembuat keputusan. Pada hakekatnya SMART merupakan model pendukung keputusan yang komprehensif dengan memperhitungkan hal-hal yang bersifat kualitatif dan kuantitatif. Dalam model pendukung keputusan dengan SMART pada dasarnya berusaha menutupi setiap kekurangan dari model-model tanpa komputerisasi sebelumnya. SMART juga memungkinkan suatu sistem dan lingkungan kedalam komponen dapat saling berinteraksi, kemudian menyatukan mereka dengan mengukur dan mengatur dampak dari komponen kesalahan sistem. SMART dapat persentase dari setiap pembobotan langsung yang ditentukan oleh hasil analisa permasalahan yang dihadapi.

Pada permasalahan pengambilan keputusan penerimaan bantuan SAS, metode SMART sangatlah tepat digunakan, mengingat pada permasalahan tersebut banyaknya atribut yang dipakai untuk menunjang keputusan yang diambil. Semakin banyak atribut yang digunakan dalam metode SMART semakin baik juga mengambil nilai keputusan akhir yang dihasilkan. Metode SMART diharapkan dapat memaksimalkan pengambilan keputusan untuk siswa yang berhak mendapatkan bantuan SAS dengan memperhitungkan banyak atribut yang digunakan secara detail dan menyeluruh dalam pemberian bobot penilaiannya.

2.3 Sistem Pendukung Keputusan (*Decision Support System*)

Simple Multi-Attribute Rating Technique merupakan salah satu metode dalam Sistem pendukung keputusan (*Decision Support System*) didefinisikan sebagai sistem *computer* yang mampu memberikan kemampuan baik kemampuan pemecahan masalah maupun kemampuan pengkomunikasian untuk masalah semi terstruktur. Secara khusus, DSS didefinisikan sebagai sebuah sistem yang mendukung kerja seorang manager maupun sekelompok manager dalam memecahkan masalah semi terstruktur dengan cara memberikan informasi ataupun usulan menuju pada keputusan tertentu (Sugiono, 2008).

2.4 Simple Multy-Attribute Rating Technique (SMART)

Simple Multy-Attribute Rating Technique (SMART) merupakan metode pendukung keputusan yang multiatribut. Teknik pembuatan keputusan multiatribut ini digunakan untuk membantu stakeholder dalam memilih antara beberapa alternatif. Setiap alternatif terdiri dari sekumpulan atribut dan setiap atribut mempunyai nilai-nilai, nilai ini dirata-rata dengan skala tertentu. Setiap atribut mempunyai bobot yang menggambarkan seberapa penting ia dibandingkan dengan atribut lain (Yeni Kustiyahningsih, Devie Rosa Anamisa, Nikmatus Syafa'ah, 2013).

Dengan SMART pembobotan atribut dilakukan dengan dua langkah yaitu:

1. Mengurutkan kepentingan suatu atribut dari level terburuk ke level terbaik.
2. Membuat perbandingan rasio kepentingan setiap atribut dengan atribut lain dibawahnya.

Langkah metode *Simple Multi-Attribute Rating Technique*

Langkah 1 : Menentukan jumlah kriteria

Langkah 2 : Sistem secara default memberikan skala 0-100 berdasarkan prioritas yang telah diinputkan kemudian dilakukan normalisasi, rumus normalisasin dapat dilihat pada rumus (1).

$$\frac{w_j}{\sum w_j} \quad (1)$$

Normalisasi :

Keterangan : w_j : bobot suatu kriteria

$\sum w_j$:total bobot semua kriteria

Langkah 3 : Memberikan nilai kriteria untuk setiap alternatif.

Hitung nilai akhir masing-masing.

Penghitungan nilai akhir didapat menggunakan perhitungan dengan rumus (2).

$$u(a_i) = \sum_{j=1}^m w_j u_j(a_i), \quad i = 1, 2, \dots, m \quad (2)$$

Keterangan : $u_i(a_i)$: nilai

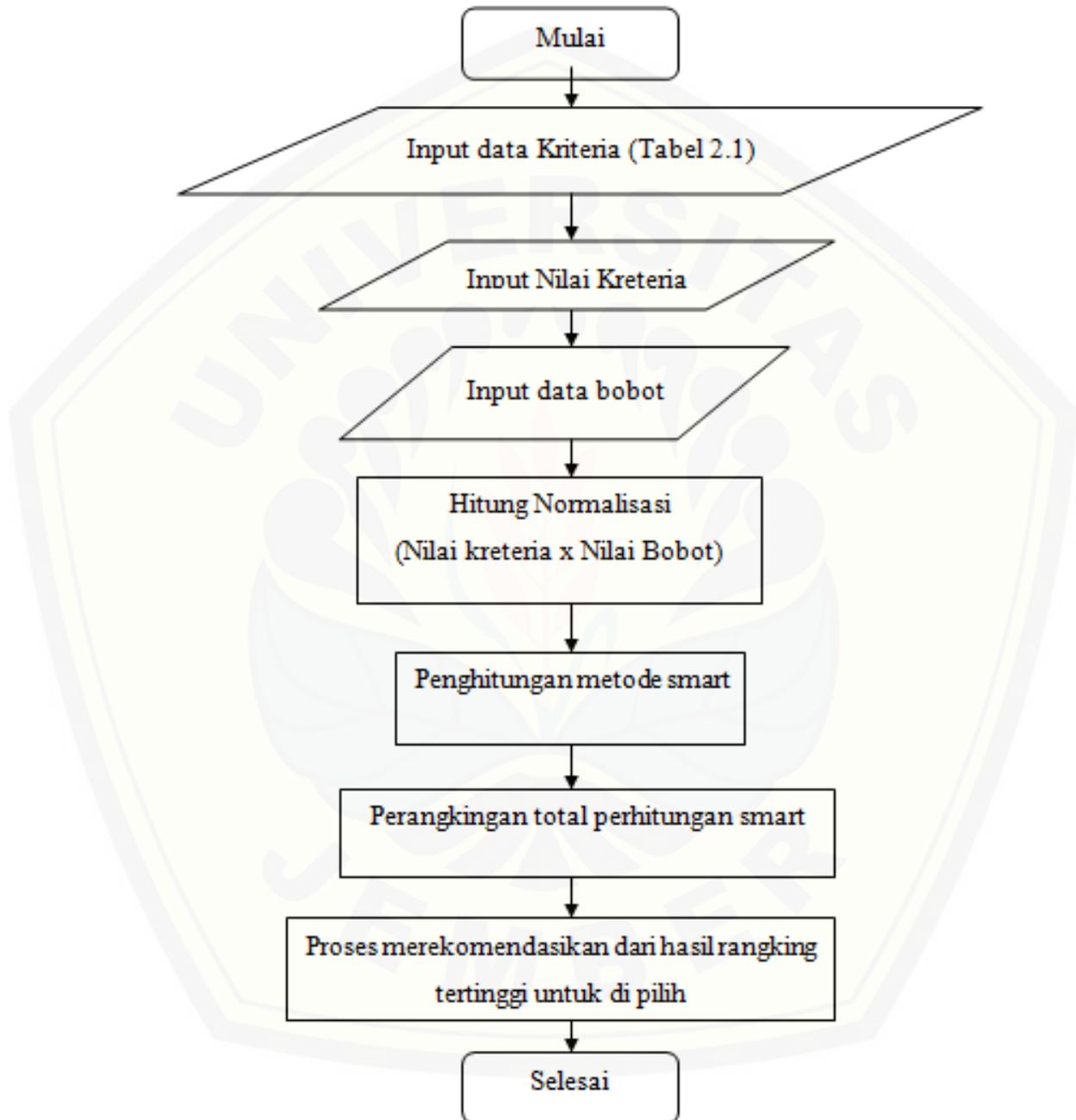
Utility kriteria ke-1 untuk kriteria ke-i

C_{max} : nilai kriteria maksimal

C_{min} : nilai kriteria minimal

$C_{out i}$: nilai kriteria ke-i

Kempat langkah tersebut dapat kita lihat diagram alir penyelesaiannya pada gambar 2.1.



Gambar 2. 1 Diagram Alir Metode SMART

Tabel 2. 1 Kriteria Penilaian

No	Kriteria Penilaian
1	Total pendapatan orang tua ibu dan ayah
	< Rp. 500.000
	Rp. 500.000 - Rp. 1.000.000
	Rp. 1.000.000 - Rp. 1.500.000
	Rp. 1.500.000 - Rp. 2.000.000
> Rp. 2.000.000	
2	Jumlah Saudara
	4
	3
	2
	1
	Tidak punya
3	Status Anak
	Tidak mempunyai ayah ibu
	Hanya memiliki 1 orang tua
Mempunyai dua orang tua	
4	Uang Jajan Sekolah
	Tanpa Uang jajan
	Rp. 3.000 - Rp. 5000
	Rp. 5000 - Rp. 10.000
> Rp.10.000	
5	Kendaraan Transportasi ke sekolah
	Jalan
	Angkutan Umum
	Sepeda
Motor	

6 Peralatan Sekolah

Tidak memiliki

Hansphone Biasa

Smartphone

Laptop

Laptop dan Smartphone

7 Rangkaing Kelas Sebelumnya

1

2

3

4

5

6

7

8

9

≤ 10

BAB 3. METODE PENELITIAN

Bab ini menjelaskan tentang jenis penelitian, tempat dan waktu penelitian, metode pengumpulan data, analisis data, dan teknik pengembangan sistem yang digunakan dalam membangun Sistem Informasi Pendukung Keputusan Siswa Asuh Sebaya

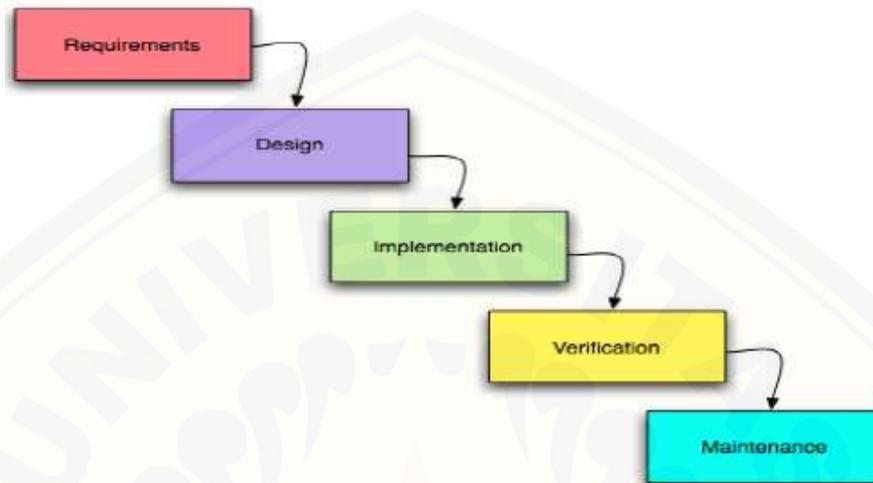
3.1 Jenis Penelitian

Jenis penelitian yang dilakukan untuk membangun Sistem Informasi SMART merupakan penelitian pengembangan yang dilakukan untuk mempermudah proses yang telah berjalan secara manual. Pengembangan dari proses manual dengan mengembangkan proses yang lebih cepat dan akurat dengan menggunakan sistem yang disesuaikan dengan proses manual serta mengimplementasikan metode *Simple Multi Atribut Rating Technique* atau metode SMART dalam proses penilaian yang dilakukan, sehingga dapat menghasilkan keluaran yang akurat dan sesuai dengan hasil yang dikehendaki.

3.2 Pengembangan Sistem

Pengembangan sistem informasi SAS dengan metode *Simple Multi-Attribute Rating Technique* dikembangkan menggunakan metode Waterfall. metode waterfall adalah proses pengembangan perangkat lunak tradisional yang umum digunakan dalam proyek-proyek perangkat lunak. Metode ini adalah model sekuensial, sehingga penyelesaian satu set kegiatan menyebabkan dimulainya aktivitas berikutnya. Metode ini disebut waterfall karena proses mengalir "secara sistematis dari satu tahap ke tahap lainnya dalam mode ke bawah. Membentuk kerangka kerja untuk pengembangan perangkat lunak (Imam Fahrurrozi, 2005). Beberapa varian dari model ada pada setiap label yang berbeda yang digunakan untuk setiap tahap. Secara umum, model ini dianggap memiliki lima tahap yang berbeda

seperti yang ditunjukkan pada Gambar 3.1 yaitu: analisis Kebutuhan, desain, implementasi, verifikasi, instalasi dan pemeliharaan.



Gambar 3. 1 Metode Waterfall

3.2.1 Analisis Kebutuhan

Tahap pertama pada proses perancangan perangkat lunak ini adalah analisis kebutuhan untuk sistem yang akan dibangun. Langkah pertama dan paling penting dari model waterfall. Ini melibatkan pengumpulan informasi mengenai solusi akhir dari kebutuhan pelanggan dan pemahaman. Ini melibatkan definisi yang jelas tentang tujuan pelanggan, harapan terhadap proyek dan masalah produk akhir diharapkan untuk memecahkan. Untuk memperoleh informasi dan data dilakukan dengan pendekatan kualitatif dan kuantitatif. Pendekatan kuantitatif adalah pendekatan yang bertujuan menggambarkan benar tidaknya fakta – fakta yang ada serta menjelaskan tentang hubungan antar variabel yang diteliti dengan cara mengumpulkan data, mengolah, menganalisis, dan menginterpretasi data dalam pengujian hipotesis statistik, sehingga dapat teruji kebenarannya (Sugiyono, 2010). Dalam penggunaannya, metode penelitian kuantitatif digunakan dalam penulisan ini untuk pengumpulan data yang di perlukan yang bersifat angka. Sedangkan pendekatan kuantitatif adalah suatu pendekatan yang juga disebut pendekatan investigasi karena biasanya peneliti

mengumpulkan data dengan cara bertatap muka langsung dan berinteraksi dengan orang-orang di tempat penelitian (McMillan, James H., Sally Scumacher, 2006).

Pada proses analisis meliputi pemahaman konteks bisnis pelanggan dan kendala, fungsi produk harus melakukan, tingkatkinerja itu harus mematuhi dan sistem eksternal itu harus sesuai dengan apa yang diminta oleh user (Imam Fahrurrozi, 2005). Peneliti mencari permasalahan yang ada untuk dapat dianalisis kebutuhan yang diperlukan, sebagai solusi dari permasalahan yang muncul. Data dan permasalahan dapat diperoleh dengan cara wawancara, studi sistem yang telah ada, dan menganalisis dokumen-dokumen yang terkait dengan penelitian. Adapun proses analisis kebutuhan lainnya yaitu sebagai berikut

3.2.1.1 Tempat dan Waktu Penelitian

Tempat yang dilaksanakan untuk penelitian adalah kabupaten Banyuwangi tepatnya pada SMAN 1 Giri Banyuwangi. Waktu penelitian dilakukan selama 3 (tiga) bulan dimulai pada bulan maret sampai selesai.

3.2.1.2 Metode Pengumpulan Data

Teknik pengumpulan data yang dilakukan pada penelitian ini menggunakan studi literature dan wawancara. Pengumpulan data yang dilakukan untuk mengumpulkan data primer dan data sekunder. Data primer diperoleh dari wawancara langsung dengan OSIS SMAN 1 Giri Banyuwangi mengenai data dan atribut yang digunakan dalam pertimbangan pemberian bantuan dana SAS. Data sekunder merupakan data yang diperoleh dari studi literature pada jurnal, buku, internet dan dokumen yang terkait dengan pendukung keputusan, metode SMART, dan SAS

3.2.1.3 Objek Penelitian

Objek penelitian ini dilakukan di daerah kabupaten Banyuwangi, dalam penelitian ini studi kasus yang diambil pada sekolah SMAN 1 Giri Banyuwangi.

3.2.2 Desain

Pembuatan desain sistem pada penelitian ini menggunakan *Unified Modeling Language (UML)* yang dirancang dengan konsep *Object-Oriented Programming (OOP)*. Berikut pemodelan *Unified Modeling Language (UML)* yang digunakan antara lain :

1. *Business Process*

Business Process digunakan dalam pembangunan sistem ini untuk membuat model atau diagram yang menggambarkan sebuah proses lengkap dengan *resources* dan *information* yang dibutuhkan hingga pada proses goal yang dikehendaki.

2. *Use Case Diagram*

Use case digunakan dalam pembangunan sistem ini untuk menggambarkan fungsi dan tugas yang dikerjakan oleh *user* dalam sistem yang mencakup hak akses yang diberikan sistem untuk user.

3. *Scenario*

Scenario diagram digunakan dalam pembangunan sistem ini untuk menceritakan secara detail fitur dan alur proses sistem dalam melakukan event.

4. *Activity Diagram*

Activity diagram digunakan untuk menceritakan secara detail alur dari sistem sama halnya dengan *scenario* tetapi *Activity Diagram* cara menceritakannya dengan menggunakan diagram alir

5. *Sequence Diagram*

Sequence diagram digunakan untuk menunjukkan rangkaian pesan yang dikirim antar object juga interaksi antar object.

6. *Class Diagram*

Class diagram digunakan untuk menggambarkan struktur statis class dalam sistem. *Class Diagram* dibuat untuk memudahkan dalam proses pengkodean karena dapat terlihat jelas alur data pada *Class Diagram*.

3.2.3 Implementasi

Pada tahap ini desain yang telah dibuat akan diimplementasikan kedalam sistem yang akan dibangun. Beberapa hal yang dilakukan dalam tahap implementasi antara lain :

1. Penulisan kode untuk membangun sistem informasi SAS menggunakan bahasa pemrograman *Java* yang berupa tampilan desktop.
2. DBMSMySQL sebagai manajemen basisdatanya.

3.2.4 Pengujian

Pada tahap ini dilakukan pengujian sistem yang telah dibangun menggunakan dua metode pengujian *white box* dan *black box*.

3.2.4.1 White Box

Menurut (Abdul Rouf. 2012) *White box* adalah Metode pengujian dengan menggunakan struktur kontrol program untuk memperoleh kasus uji. Pengujian *White box* menggunakan metode pengujian *Cyclometric Complexity* yang berfungsi untuk mengukur tingkat kompleksitas suatu method seperti proses pada gambar 3.2. Dengan menggunakan *white box* akan didapatkan kasus uji yaitu:

1. Menjamin seluruh jalur independen didalam modul yang dieksekusi sekurang-kurangnya sekali
2. menguji semua keputusan logikal
3. menguji seluruh Loop yang sesuai dengan batasannya
4. menguji seluruh struktur data internal yang menjamin validitas
5. Basis Path adalah teknik uji coba *white box* (Tom Mc Cabe).

6. Basis Path : untuk mendapatkan kompleksitas logik dari suatu prosedur dan menggunakan ukuran ini sebagai petunjuk untuk mendefinisikan himpunan jalur yang akan diuji
7. Basis Path menggunakan notasi graph untuk menggambarkan aliran kontrolnya.



Gambar 3. 2 Sitem kerja dari Teknik Pengujian *White Box*

3.2.4.2 Black Box

Black Box adalah pengujian untuk mengetahui apakah semua fungsi perangkat lunak telah berjalan semestinya sesuai dengan kebutuhan fungsional yang telah didefinsikan. Metode *Black Box* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program (Abdul Rouf, 2012). Sistem kerja *blackbox* dapat dilihat pada gambar 3.3. *Black Box* dapat menemukan kesalahan dalam kategori berikut :

1. Fungsi-fungsi yang tidak benar atau hilang
2. Kesalahan *interface*
3. Kesalahan dalam strutur data atau akses basisdata eksternal
4. Inisialisasi dan kesalahan terminasi
5. validitas fungsional
6. kesensitifan sistem terhadap nilai input tertentu
7. batasan dari suatu data



Gambar 3. 3 Sitem kerja dari Teknik Pengujian *Black Box*

3.2.5 Pemeliharaan

Pemeliharaan pada system diperlukan ketika sistem yang telah di gunakan oleh pengguna. Ketika sistem di jalankan dan digunakan selama pemakaian mungkin masih ada terjadi eror pada kode yang tidak ditemukan pada sebelumnya. Sehingga butuh pemeliharaan dan perbaikan untuk menanggulangi error yang terjadi pada sistem



BAB 4. PENGEMBANGAN SISTEM

Bab ini menguraikan mengenai analisis kebutuhan dan perancangan hingga tahap pengkodean dan pengujian sistem informasi SMART menggunakan Metode Double Moving Average. Dimana tahapan analisis hingga pengujian yang dilakukan sesuai dengan metode pengembangan *waterfall*.

4.1 Deskripsi Umum Sistem

Sistem pendukung keputusan Siswa Asuh Sebaya (SAS) merupakan sistem yang dibangun untuk membantu menunjang keputusan pemberian bantuan agar lebih akurat dan terarah pada siswa yang benar – benar membutuhkan. Sistem ini membantu mengambil keputusan menggunakan metode *Simple Multi-Attribute Rating Technique* yang sangat cocok digunakan dalam pendukung keputusan dalam kasus pemberian bantuan dana SAS. mengingat program SAS untuk sekolah di Banyuwangi sangatlah baik, tetapi dalam mengambil keputusan untuk menentukan siswa mana yang pantas untuk mendapatkannya sangat kurang jadi dapat didukung oleh sistem pendukung keputusan yang menggunakan metode SMART.

Deskripsi umum dari Sistem Informasi Siswa Asuh Sebaya yang dibangun dalam penelitian ini akan dijelaskan lebih detail pada SOP (*statement of perpose*) sistem dan fungsi sistem.

4.1.1 SOP (Statement Of Perpose)

Sistem Informasi Siswa Asuh Sebaya menggunakan metode *Simple Multi-Attribute Rating Technique* ini merupakan sebuah sistem yang mampu memberikan pendukung keputusan untuk memilih siswa siapa saja yang layak mendapatkan bantuan Siswa Asuh Sebaya. Ada beberapa data yang dibutuhkan untuk menggunakan sistem ini agar dapat memberi pendukung keputusan yaitu data setiap siswa yang mendafta untuk mendapatkan bantuan siswa asuh sebaya mulai dari data

diri sampai data yang sifatnya umum yang dapat menunjang kerja sistem. Berdasarkan data tersebut sistem akan mengolah dan menjadikan sebuah data tersebut kedalam angka dan mengakumulasi dengan menggunakan metode SMART sehingga akan didapat nilai akhir dari perbandingan semua data siswa tersebut. Data tersebut kemudian dibandingkan dengan nilai siswa lainnya untuk diambil 10 siswa yang layak mendapatkan bantuan SAS. Selain itu sistem ini dapat mengolah data siswa dan mencari siswa yang layak mendapatkan dana SAS dengan cepat dan tentunya akurat dengan batasan penggunaan metode SMART.

4.1.2 Fungsi Sistem

Fungsi utama dari Sistem Informasi Siswa Asuh Sebaya ini adalah terletak pada fitur penghitung dengan menggunakan metode SMART yang datanya hanya dapat dikelola oleh ketua OSIS dipercaya dari sekolah untuk menentukan siswa yang layak mendapatkan bantuan SAS dengan mengelolanya dengan menggunakan sistem ini. Sistem ini memiliki fitur login yang membedakan dua hak akses yang meliputi :

a. Organisasi Siswa Intar Sekolah / OSIS

OSIS merupakan pengguna sistem yang telah terdaftar sebagai pemegang hak akses untuk mengelola penuh data siswa yang akan mendaftar SAS dan dapat penuh menjalankan fitur penghitungan dengan menggunakan metode SMART. Selain itu OSIS juga dapat merubah semua perhitungan yang ada dan disesuaikan dengan yang diinginkan dengan syarat persetujuan sekolah.

b. Guru

Guru merupakan pengguna sistem yang hanya mendapatkan akses untuk menginputkan siswa siapa saja yang mendaftar untuk mendapatkan bantuan SAS, karena semua siswa yang ingin mendaftarkan diri untuk mendapatkan bantuan SAS harus lewat guru. Dengan demikian data siswa yang ikut terdaftar disistem tidak

sembarangan dan sudah melalui persetujuan sekolah yang diwakili oleh guru sebagai peinput data siswa yang akan mendaftar.

4.2 Analisa Kebutuhan Sistem

Berdasarkan metode pengembangan sistem yang menggunakan model waterfall. Tahap awal yang dilakukan adalah tahap analisa yang di tujukan kepada analisa objek yang diteliti. Analisa tersebut bertujuan untu memperoleh kebutuhan-kebutuhan dari sistem yang dibangun, baik berupa kebutuhan fungsional maupun kebutuhan nonfungsional. Dimana hasil analisa tersebut sangat mempengaruhi fungsionalitas system yang dibangun untuk dapat digunakan sesuai dengan fungsi dan kebutuhan pengguna. Adapun penjelasn tentang kebutuhan - kebutuahan tersebut yaitu :

4.2.1 Kebutuhan Fungsional

Kebutuhan fungsional yang dimaksud yaitu kebutuhan yang berisikan fitur – fitur inti yang harus ada dan dipenuhi dalam sistem agar dapat difungsikan sesuai dengan tujuan sistem itu sendiri. Adapun kebutuhan fungsional dalam Sistem Informasi Siswa Asuh Sebaya yang menggunakan metode SMART yaitu :

- a. Sistem dapat mengolah data siswa yang akan mendaftar sebagai penerimas Siswa Asuh Sebaya
- b. Sistem mampu mengelolah data siswa yang sudah terdaftar menjadi sebuah pertimbangan keputusan untuk user dengan menggunakan metode SMART
- c. Sistem dapat mengolah perhitungan metode SMART dengan menyesuaikan nilai yang akan digunakan dalam pendukung keputusan.
- d. Sistem dapat mengelolah kembali data siapa saja yang menerima SAS jika terjadi kesalahan dalam data yang dimasukkan atau diterima oleh user.
- e. Sistem dapat menyesuaikan penilaian user dengan merubah nilai perhitungan pada metode smart untuk mencari nilai yang lebih baik dan seimbang.

- f. Sistem dapat menampilkan siswa yang mendapatkan bantuan SAS kedalam dua hak akses yang ada, yaitu guru dan OSIS.
- g. Sistem dapat mencetak laporan siswa beserta nilai hasil keputusan per periode

4.2.2 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan fitur-fitur yang dimiliki untuk mendukung sistem dalam memenuhi fungsionalitasnya untuk dapat memenuhi kebutuhan dari pengguna. Kebutuhan non-fungsional dari sistem informasi siswa asuh sebaya yaitu :

- a. Sistem memiliki batasan hak akses dengan menggunakan username dan password untuk menggunakan fitur dalam Sistem Informasi Siswa Asuh Sebaya.
- b. Sistem dibangun berbasis desktop.
- c. Sistem menggunakan bahasa pemrograman Java.

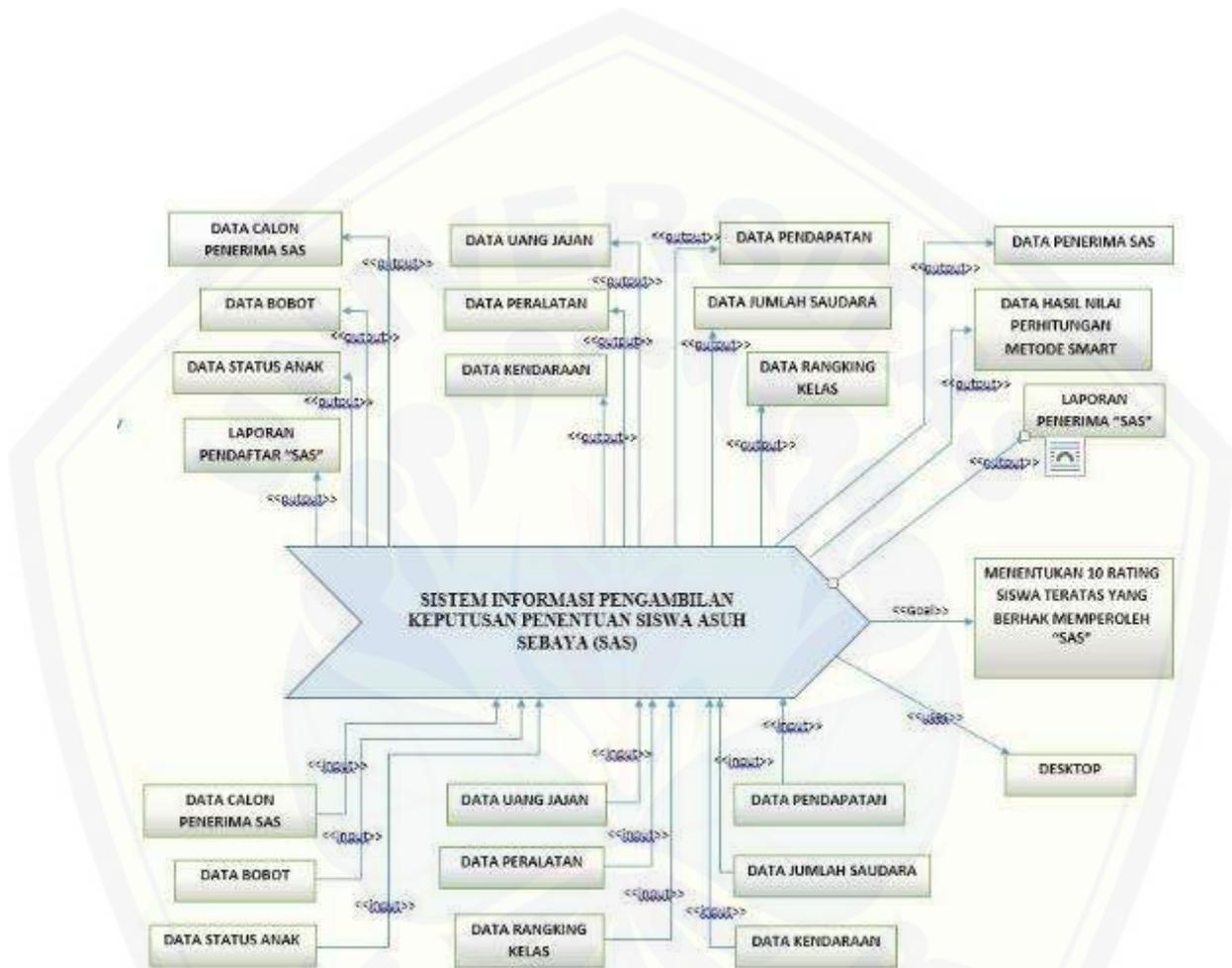
4.3 Desain Sistem

Tahap yang dilakukan setelah menentukan dan menganalisa kebutuhan sistem yaitu perencanaan pembangunan sistem yang dapat digambarkan dengan desain sistem. Desain Sistem Informasi Siswa Asuh Sebaya ini meliputi Business process, usecase, diagram, use case skenario, activity diagram, sequence diagram, class diagram, dan entity relationship diagram.

4.3.1 Business Process

Selain dapat dideskripsikan dalam sebuah Statement of Purpose gambar umum Sistem Informasi Siswa Asuh Sebaya ini dapat digambarkan melalui business process. Proses tersebut dapat kita lihat pada gambar 4.1. Pada gambar tersebut dapat kita lihat data – data yang digunakan sebagai masukan dan data yang dapat jadi data

keluaran termasuk hasil dari sistem itu sendiri, hingga sampai goal dari dibangunnya sistem pendukung keputusan itu sendiri.



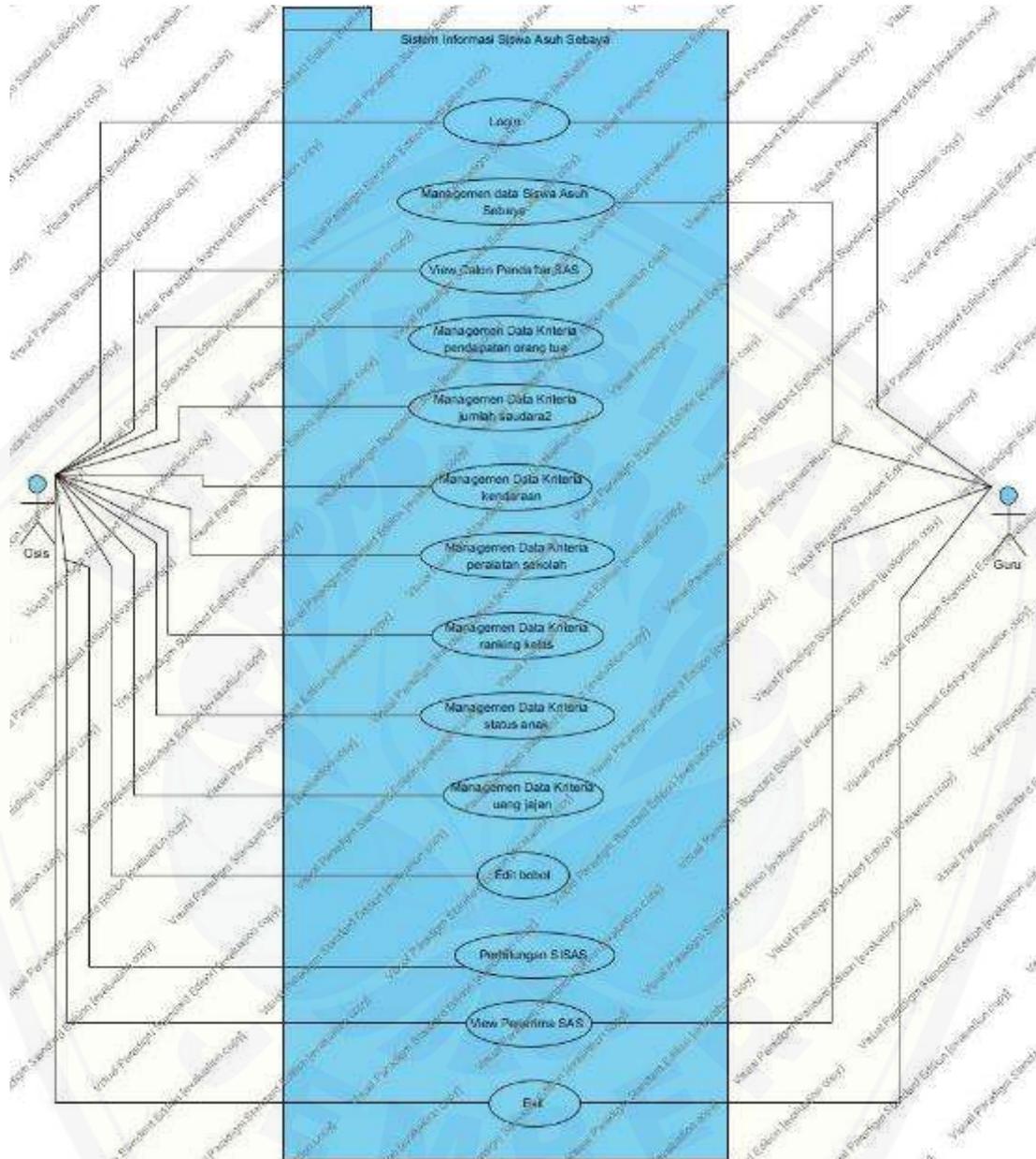
Gambar 4. 1 Bussiness Procces Sistem Informasi Siswa Asuh Sebaya

4.3.2 Use Case Diagram

Use case diagram yaitu pemodelan yang dibuat dengan bertujuan untuk menggambarkan interaksi antara actor dengan sistem informasi prediksi yang akan dibangun. Melalui use case pembaca dapat mengetahui interaksi actor terhadap sistem yang kita bangun yang dibedakan dengan menu dari setiap hak akses actor

tersebut. Pada gambar 4.2 use case diagram Sistem Informasi Siswa Asuh Sebaya menggunakan dua actor dengan empat belas use case.





Gambar 4. 2 Use Case Diagram Sistem Informasi Siswa Asuh Sebaya

Berdasarkan use case diagram pada gambar 4.2 tersebut terdapat dua actor yaitu guru dan OSIS. Kedua actor tersebut dapat melihat hak akses apa saja yang diperoleh dari use case tersebut. Adapun diskripsi dari masing masing actor dapat dilihat pada tabel 4.1

Tabel 4. 1 Diskripsi pembagian sistem

Aktor	Deskripsi
Guru	Aktor yang memiliki hak akses untuk menginputkan data siswa yang akan mendaftar, serta dapat mengakses dan melihat siapa saja yang menerima SAS dan dapat mencari siswa penerima SAS yang gagal
OSIS	Aktor yang memiliki hak akses dan tanggung jawa penuh terhadap sistem mulai dari menginputkan atribut pada kriteria, menentukan bobot penilaian, menghitung metode SMART dan melihat siswa yang lolos Siswa Asuh Sebaya

Selain Memiliki dua actor guru dan OSIS, dalam use case diagram juga terdapat empat belas use case. Deskripsi dari usecase tersebut pada Gambar 4.2 dapat dilihat pada tabel 4.2

Tabel 4. 2 Deskripsi Use Case Sistem

NO	Use Case	Deskripsi
1	Login	Fitur yang berfungsi sebagai form autentifikasi hak akses antar user OSIS dan guru dalam penggunaan sistem.

2	CRUD SAS	Fitur yang dapat diakses oleh pengguna level 2 yaitu guru yang berfungsi untuk menambah, mengedit, serta menghapus data ke dalam sistem
3	View Calon Penerima SAS	Fitur ini dapat di akses oleh pengguna level 1 yaitu OSIS untuk melihat calon penerima SAS yang sudah terdaftar lewat guru.
4	CRUD Total Pendapatan Orang Tua	Fitur ini dapat di akses oleh pengguna level 1 , yaitu OSIS untuk menambah, mengedit serta meng-menghapus nama atribut total pendapatan orang tua dan nilai atribut ke dalam sistem
5	CRUD Jumlah Saudara	Fitur ini dapat di akses oleh pengguna level 1 , yaitu OSIS untuk menambah, mengedit serta meng-menghapus nama atribut jumlah saudara dan nilai atribut ke dalam sistem
6	CRUD Status Anak	Fitur ini dapat di akses oleh pengguna level 1 , yaitu OSIS untuk menambah, mengedit serta meng-menghapus nama atribut status anak dan nilai atribut ke dalam sistem
7	CRUD Uang Jajan	Fitur ini dapat di akses oleh pengguna level 1 , yaitu OSIS untuk menambah, mengedit serta meng-menghapus nama atribut uang jajan dan nilai atribut ke dalam sistem
8	CRUD Kendaraan ke Sekolah	Fitur ini dapat di akses oleh pengguna level 1 , yaitu OSIS untuk menambah, mengedit serta meng-menghapus nama atribut kendaraan ke sekolah dan nilai atribut ke dalam sistem

Lanjutan...

9	CRUD Peralatan Sekolah	Fitur ini dapat di akses oleh pengguna level 1 , yaitu OSIS untuk menambah, mengedit serta meng-menghapus nama atribut Peralatan Sekolah dan nilai atribut ke dalam sistem
10	CRUD Rangkaing Kelas	Fitur ini dapat di akses oleh pengguna level 1 , yaitu OSIS untuk menambah, mengedit serta meng-menghapus nama atribut rangkaing kelas dan nilai atribut ke dalam sistem
11	Edit Bobot	Fitur ini dapat di akses oleh pengguna level 1 , yaitu OSIS untuk mengedit nilai bobot ke dalam sistem
12	CRUD Penghitungan SISAS	Fitur ini dapat di akses oleh pengguna level 1 , yaitu OSIS untuk mengolah siswa yang akan di daftarkan SAS dengan menggunakan metode SMART dan dapat menambah, mengedit serta menghapus siswa yang terdaftar SAS
13	View Penerima SAS	Fitur ini dapat di akses oleh pengguna level 1 dan level 2, yaitu OSIS dan guru untuk melihat siapa saja yang menerima bantuan Siswa Asuh Sebaya per periode dan dapat mencari nama siswa yang tidak lolos menerima bantuan Siswa Asuh Sebaya

Berikut ini merupakan penjelasan dari desain sistem yang akan dibangun berdasarkan masing-masing usecase sebagai berikut :

4.3.2.1 Use Case CRUD SAS

Use case CRUD SAS merupakan fitur yang memungkinkan pengguna level 2 atau guru untuk dapat menambahkan, mengedit serta mengupdate data dari calon siswa pendaftar SAS. Gambaran dari desai sistem untuk use case secara detail dapat dilihat pada lampiran A.2, Activity diagram seperti pada lampiran B.2, dan sequence diagram seperti pada lampiran C.

4.3.2.2 Use Case View Calon Penerima SAS

Use case view calon penerima SAS merupakan fitur yang memungkinkan pengguna level 1 yaitu OSIS untuk dapat melihat calon siswa yang akan mengikuti seleksi penerimaan SAS yang sudah didaftarkan melalui guru. Gambaran dari desai sistem untuk use case secara detail dapat dilihat pada lampiran A.3, Activity diagram seperti pada B.3, dan sequence diagram seperti pada lampiran C.

4.3.2.3 CRUD Kriteria Pendapatan Orang Tua

Use case CRUD kriteria pendapatan orang tua merupakan fitur yang memungkinkan pengguna level 1 yaitu OSIS menambah, mengedit, dan menghapus atribut yang akan dimasukkan pada kriteria. Gambaran dari desai sistem untuk use case secara detail dapat dilihat pada lampiran A.4, Activity diagram seperti pada lampiran B.6, dan sequence diagram seperti pada lampiran C.

4.3.2.4 CRUD Kriteria Jumlah Saudara

Use case CRUD kriteria jumlah saudara merupakan fitur yang memungkinkan pengguna level 1 yaitu OSIS menambah, mengedit, dan menghapus atribut yang akan dimasukkan pada kriteria. Gambaran dari desai sistem untuk use case secara detail dapat dilihat pada lampiran A5, Activity diagram seperti pada lampiran B.9, dan sequence diagram seperti pada lampiran C.

4.3.2.5 CRUD Kriteria Status Anak

Use case CRUD kriteria status anak merupakan fitur yang memungkinkan pengguna level 1 yaitu OSIS menambah, mengedit, dan menghapus atribut yang akan dimasukkan pada kriteria. Gambaran dari desain sistem untuk use case secara detail dapat dilihat pada lampiran A.6, Activity diagram seperti pada lampiran B.7, dan sequence diagram seperti pada lampiran C.

4.3.2.6 CRUD Kriteria Uang Jajan

Use case CRUD kriteria uang jajan merupakan fitur yang memungkinkan pengguna level 1 yaitu OSIS menambah, mengedit, dan menghapus atribut yang akan dimasukkan pada kriteria. Gambaran dari desain sistem untuk use case secara detail dapat dilihat pada lampiran A.7, Activity diagram seperti pada lampiran B.8, dan sequence diagram seperti pada lampiran C.

4.3.2.7 CRUD Kriteria Kendaraan ke Sekolah

Use case CRUD Kriteria kendaraan ke sekolah merupakan fitur yang memungkinkan pengguna level 1 yaitu OSIS menambah, mengedit, dan menghapus atribut yang akan dimasukkan pada kriteria. Gambaran dari desain sistem untuk use case secara detail dapat dilihat pada lampiran A.8, Activity diagram seperti pada lampiran B.10, dan sequence diagram seperti pada lampiran C.

4.3.2.8 CRUD Kriteria Peralatan Sekolah

Use case CRUD Kriteria peralatan sekolah merupakan fitur yang memungkinkan pengguna level 1 yaitu OSIS menambah, mengedit, dan menghapus atribut yang akan dimasukkan pada kriteria. Gambaran dari desain sistem untuk use case secara detail dapat dilihat pada lampiran A.9, Activity diagram seperti pada B.11, dan sequence diagram seperti pada lampiran C.

4.3.2.9 CRUD Kriteria Rangkaing Kelas

Use case CRUD Kriteria rangkaing kelas merupakan fitur yang memungkinkan pengguna level 1 yaitu OSIS menambah, mengedit, dan menghapus atribut yang akan dimasukkan pada kriteria. Gambaran dari desai sistem untuk use case secara detail dapat dilihat pada lampiran A.10, Activity diagram seperti pada lampiran B.12, dan sequence diagram seperti pada lampiran C.

4.3.2.10 Edit Bobot

Edit bobot merupakan fitur yang memungkinkan pengguna level 1 yaitu OSIS untuk dapat mengganti data nilai bobot. Gambaran dari desai sistem untuk use case secara detail dapat dilihat pada A.11, Activity diagram seperti pada lampiran B.5, dan sequence diagram seperti pada lampiran C.

4.3.2.11 CRUD Penghitungan Smart

CRUD penghitungan Smart merupakan fitur utama dari sistem yang hanya pengguna level 1 yaitu OSIS yang dapat mengaksesnya. Fitur ini berupa proses pemilihan Siswa Asuh Sebaya dengan menggunakan metode SMART dengan cara menginputkan nama serta data keadaan siswa untuk menjadi inputan pada sistem. Pada fitur ini juga dapat mengedit serta menghapus siswa yang akan mengikuti pencalonan Siswa Asuh Sebaya. Gambaran dari desai sistem untuk use case secara detail dapat dilihat pada Skenario Penghitungan SMART, *Activity* Penghitungan SMART, dan *Sequence diagram* Penghitungan SMART

Skenario penghitungan SMART merupakan interaksi antara actor (OSIS) dan sistem pada fitur penghitungan smart. Dalam scenario penghitungan SMART terdapat scenario normal dan scenario alternative. Skenario normal meliputi hasil dari penghitungan SMART, scenario alternative terjadi saat terdapat data yang kosong

pada form. Skenario Penghitungan SMART lebih lengkapnyadapat dilihat pada tabelTabel 4.3

Tabel 4. 3 Skenario Penghitungan Smart

ID	US-12
Name	Penghitungan Smart
Participating Actor	OSIS
Entry Condition	Manager akan melakukan Penghitungan pada Penghitungan SMART
Exit Condition	Manager telah melakukan Penghitungan pada Penghitungan SMART
SKENARIO NORMAL “Metode SMART”	
Manager	Sistem
1. Klik menu Penghitungan Smart	
	2. Menampilkan halaman menu Penghitungan Smart
3. Mengisi Tahun Periode	
4. Memilih Nama Siswa	
5. Memilih Atribut pada Kriteria Total pendapatan orang tua	
	6. Menampilkan nilai sementara dari atribut pada Kriteria Total pendapatan orang tua

7. Memilih Atribut pada Kriteria
Jumlah Saudara

8. Menampilkan nilai sementara dari
Atribut pada Kriteria Jumlah
Saudara

9. Memilih Atribut pada Kriteria
Status anak

10. Menampilkan nilai sementara dari
Atribut pada Kriteria Status anak

11. Memilih Atribut pada Kriteria
Uang Jajan

12. Menampilkan nilai sementara dari
Atribut pada Kriteria Uang Jajan

13. Memilih Atribut pada Kriteria
Kendaraan ke sekolah

14. Menampilkan nilai sementara dari
Atribut pada Kriteria Kendaraan ke
sekolah

15. Memilih Atribut pada Kriteria
Peralatan sekolah

16. Menampilkan nilai sementara dari
Atribut pada Kriteria Peralatan
sekolah

17. Memilih Atribut pada Kriteria
Rangking Kelas

18. Menampilkan nilai sementara dari
Atribut pada Kriteria Rangking

Kelas

	19. Menampilkan total nilai dari penghitungan metode
20. klik “simpan”	
	21. Menampilkan pesan “SISWA CALON PENERIMA SAS BERHASIL MASUK SELEKSI PENERIMAAN”

SKENARIO ALTERNATIF**“Belum Mengisi Field”**

Manager	Sistem
20a. Klik simpan	
	21a. Menampilkan pesan “Kolom Tahun Belum di isi atau ada form yang belum terisi”

Activity Diagram menggambarkan alir aktifitas dalam sistem yang sedang di rancang, mengetahui bagaimana masing-masing alir berawal, decision yang mungkin terjadi dan bagaimana sistem itu berakhir. Activity diagram harus sesuai dengan sistem yang dibangun. Activity diagram menggambarkan aktivitas aktor dan sistem yang saling berhubungan dalam suatu aktivitas atau *event*. Activity diagram penghitungan SMART lebih lengkapnya dapat dilihat pada gambar 4.3 dan *Sequence Diagram* melihat Perhitungan SMART lebih lengkapnya dapat dilihat pada gambar 4.4

4.3.2.12 View Penerima SAS

View penerima SAS merupakan fitur yang memungkinkan pengguna level 1 dan 2 yaitu OSIS dan guru untuk dapat melihat siswa yang terpilih menerima bantuan SAS dan mencari nama siswa untuk melihat hasil dari proses penghitungan dengan menggunakan metode SMART. Gambaran dari desain sistem untuk use case view penerima dapat dilihat pada tabel 4.4 dibawah ini

Tabel 4. 4 Penerima SAS

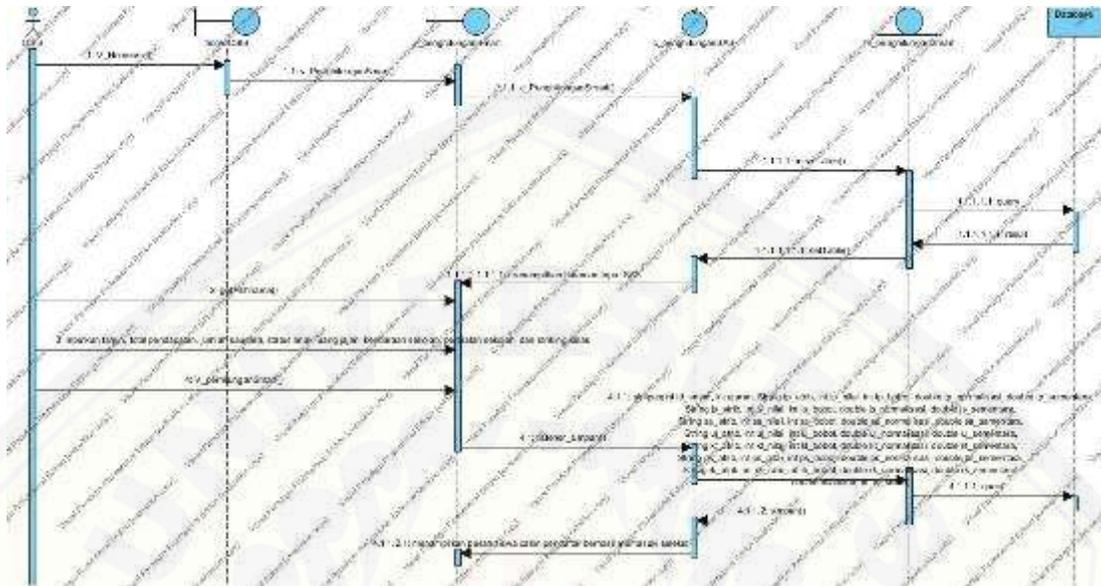
ID	US-12
Name	Penghitungan Smart
Participating Actor	OSIS dan Guru
Entry Condition	Manager akan melakukan Penghitungan pada Penghitungan SMART
Exit Condition	Manager telah melakukan Penghitungan pada Penghitungan SMART
SKENARIO NORMAL	
“Penghitungan SMART”	
Manager	Sistem
1. Klik menu Penerima SAS	
	2. Menampilkan halaman menu Penerima SAS
3. Memilih tahun periode	
4. Klik cari	
	5. Menampilkan hasil penerima SAS berdasarkan tahun periode

4.3.3 Class Diagram

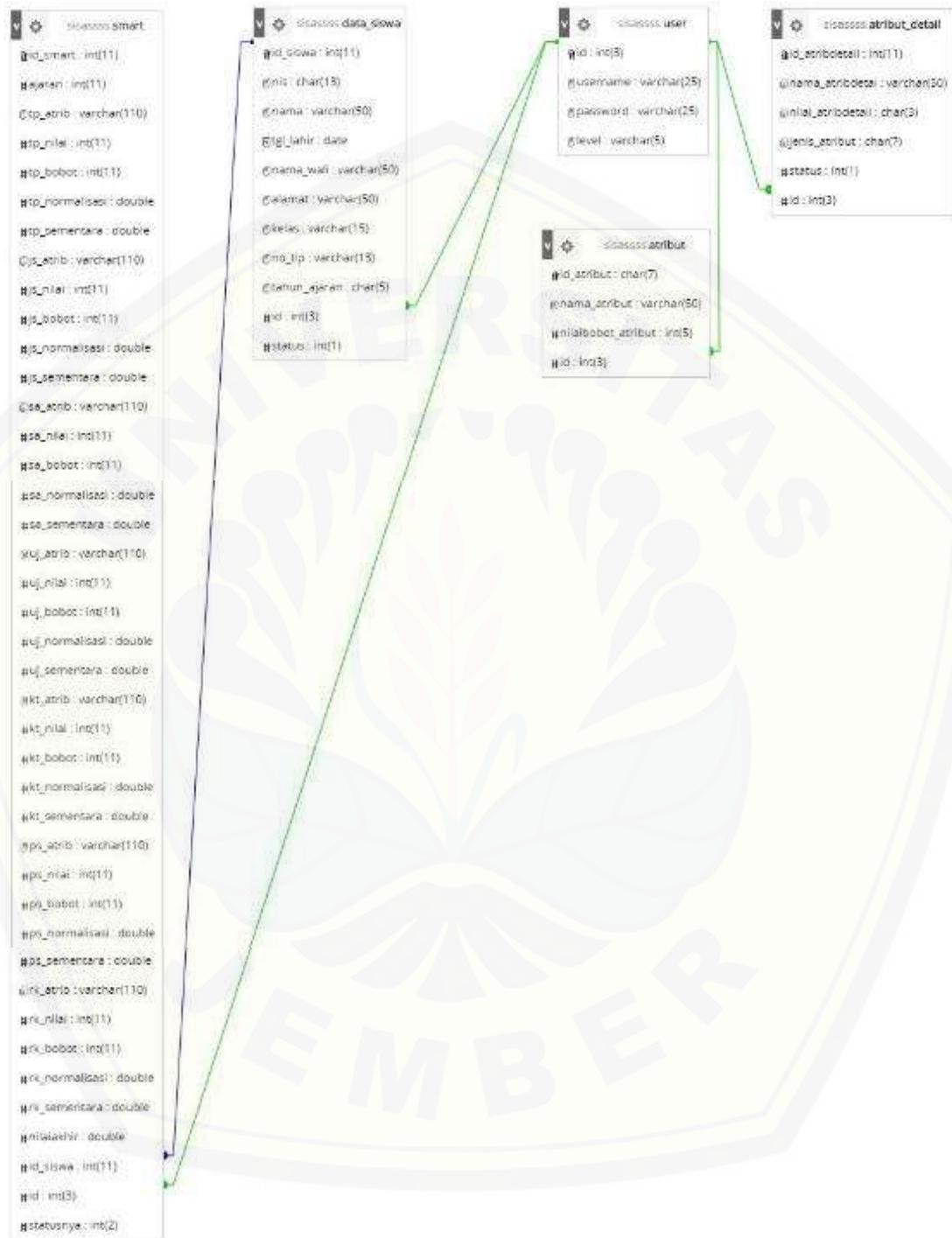
Setelah melalui pembuatan desai dengan sequence diagram, padah tahap ini dilanjutkan membuat desai perancangan class diagram. Class diagram terdiri dari model, view, dan controller yang masing-masing berisi method dan data yang berbeda namun memiliki hubungan dengan yang lainnya. Deskripsi lengkap class diagram dapat dilihat pada gambar 4.6

4.3.4 Entity Relationship Diagram

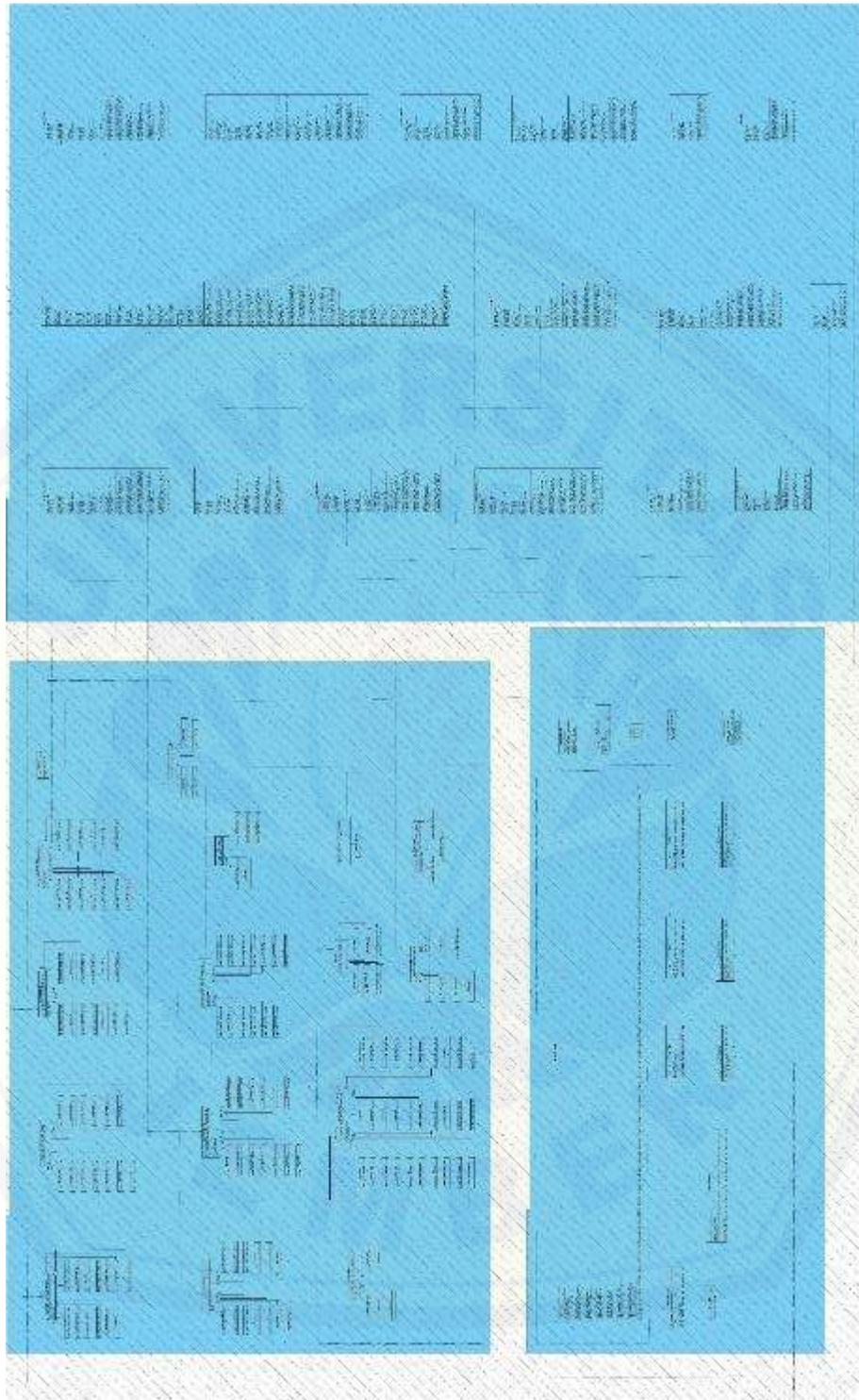
Setelah pembuatan class diagram, tahapan yang dilakukan selanjutnya yaitu membuat desain database sistem informasi siswa asuh sebaya. Pada tahapan ini membuat desain data base yang berisi tentang data data yang akan digunakan dalam pengembangan sistem. Data tersebut disimpan dalam tabel-tabel yang ada di database yang meliputi tabel atribut (bobot), tabel atribut detail, tabel smart, tabel data siswa dan tabel user. Tabel atribut berisi nama kriteria, nilai bobot perkriteria, id kriteria serta id user yang menginputkan. Tabel atribut detail berisi nama atribut, bobot nilai atribut, id atributdetail, jenis atribut, id user dan status. Tabel data siswa berisikan data siswa yang akan mendaftar menjadi peserta SAS yaitu tabel id_siswa, nis, nama, tanggal lahir, nama wali, alamat, kelas, nomor telepon, id_user dan tahun ajaran. Tabel smart berisikan semua data perhitungan yang dibutuhkan oleh metode seperti id_smart, ajaran, tp_atrib, tp_nilai, tp_bobot, tp_normalisasi, tp_sementara, js_atrib, js_nilai, js_bobot, js_normalisasi, js_sementara, sa_atrib, sa_nilai, sa_bobot, sa_normalisasi, sa_sementara, uj_atrib, uj_nilai, uj_bobot, uj_normalisasi, uj_sementara, kt_atrib, kt_nilai, kt_bobot, kt_normalisasi, kt_sementara, tp_atrib, tp_nilai, tp_bobot, tp_normalisasi, tp_sementara, ps_atrib, ps_nilai, ps_bobot, ps_normalisasi, ps_sementara, rk_atrib, rk_nilai, rk_bobot, rk_normalisasi, rk_sementara, nilai_akhir, id dan id_siswa. Tabel user berisi id, username, password, level. Gambaran desain database lebih lengkapnya dapat dilihat pada gambar 4.5.



Gambar 4. 4 Sequence Diagram



Gambar 4. 5 Entity Relationship Diagram



Gambar 4. 6 Class Diagram

4.4 Pengkodean Sistem

Setelah tahapan desai database yang dilakukan, selanjutnya masuk pada tahap pengimplementasian desain perancangan ke dalam bahasa pemrograman. Bahasa pemrograman yang digunakan adalah bahasa JAVA dan menggunakan database MySQL. Dalam tahapan implementasi perancangan ini menjelaskan tentang yang terdapat pada sistem informasi SAS. fitur yang ada di desain khusus dengan menyesuaikan desain perancangan yang ada dan fitur yang pasti ada yaitu fitur penghitungan SMART untuk mereting siswa yang masuk dalam pemilihan SAS.

4.5 Pengujian Sistem

Pada tahap ini sistem yang sudah kita bangun akan melakukan pengujian dan mengevaluasi sistem dengan menggunakan sebuah metode pengujian sistem. Hal tersebut bertujuan untuk mengevaluasi apakah sistem yang telah kita bangun sudah terpenuhi semua dan sistem layak untuk digunakan pengguna. Untuk pengujian sistem informasi SAS menggunakan dua pengujian yaitu *white box* dan *black box*.

4.5.1 Metode *White Box*

Pengujian sistem dengan menggunakan metode pengujian white box dilakukan untuk menguji sistem dari segi desai dan kode penulisan program. Pengujian tersebut bertujuan untuk mengevaluasi apakah sistem mampu menghasilkan fungsi, inputan dan keluaran yang sesuai dan diharapkan dengan spesifikasi dari kebutuhan sistem informasi SAS sendiri. Pengujian ini dilakukan dengan cara menghitung independent path, independent path yaitu dengan menggunakan suatu ukuran kuantitatif *Cyclomatic Complexity*, *Listener Program*, penentuan jalur independen, dan *test case*. Pada tahapan ini dilakukan pengujian pada fitur yang dinilai dapat mewakili sistem informasi SAS yaitu pada fitur penghitungan metode SMART. Tahap pengujian jalur dasar yang dilakukan meliputi :

a. *Listing Program*

Listing program merupakan baris-baris kode yang nantinya akan diuji. Setiap langkah dari kode-kode yang ada diberi nomor baik menjalankan *statement* biasa atau penggunaan kondisi dalam program.

b. Diagram Alir

Diagram alir merupakan notasi sederhana yang digunakan untuk merepresentasikan aliran kontrol. Aliran kontrol yang digambarkan merupakan hasil penomoran dari *listing* program yang digambarkan dengan *node-node* (simpul) yang dihubungkan dengan *edge-edge* (garis).

c. Kompleksitas Siklomatik

Kompleksitas siklomatik yaitu suatu metrik perangkat lunak yang menyediakan ukuran kuantitatif dari kompleksitas logis suatu program. Bila digunakan dalam konteks teknik pengujian jalur dasar, nilai yang dihitung untuk kompleksitas siklomatik mendefinisikan jumlah jalur independen dalam basis set suatu program.

d. Jalur Program Independen

Jalur independen adalah setiap jalur yang melalui program yang memperkenalkan setidaknya satu kumpulan pernyataan-pernyataan pemrosesan atau kondisi baru. Bila dinyatakan dalam grafik alir, jalur independen harus bergerak setidaknya sepanjang satu *edge* yang belum dilintasi sebelum jalur tersebut didefinisi.

e. Pengujian Basis Set

Pada bagian ini diberikan contoh data yang akan memaksa pelaksanaan jalur di basis set. Data yang dieksekusi dimasukkan ke dalam grafik alir apakah sudah melewati basis set yang tersedia. Sistem telah memenuhi syarat kelayakan perangkat lunak jika salah satu jalur yang dieksekusi setidaknya satu kali.

Listing Program

Program Normalisasi Pendapatan

```
444 public Object setNormalisasiPendapatan(String bobot) {
445     String a = "";
446
447     double dibagi = 100;
448     try {
449         Statement statement = kon.getKoneksi().createStatement();
450         ResultSet result = statement.executeQuery("select * from atribut whe
451
452         if (result.next()) {
453             double b = Integer.parseInt(result.getString("nilaibobot_atribut
454             a = b / dibagi + "";
455         }
456
457     } catch (SQLException E) {
458         System.out.println("gagal dicetak karena :" + E);
459     }
460     System.out.println(a);
461     return a;
462 }
```

Program Normalisasi Saudara

```
464 public Object setNormalisasiSaudara(String bobot) {
465     String a = "";
466
467     double dibagi = 100;
468     try {
469         Statement statement = kon.getKoneksi().createStatement();
470         ResultSet result = statement.executeQuery("select * from atribut whe
471
472         if (result.next()) {
473             double b = Integer.parseInt(result.getString("nilaibobot_atribut
474             a = b / dibagi + "";
475         }
476
477     } catch (SQLException E) {
478         System.out.println("gagal dicetak karena :" + E);
479     }
480     System.out.println(a);
481     return a;
482 }
```

Program Normalisasi Status

```
484 public Object setNormalisasiStatus(String bobot) {
485     String a = "";
486
487     double dibagi = 100;
488     try {
489         Statement statement = kon.getKoneksi().createStatement();
490         ResultSet result = statement.executeQuery("select * from atribut wh
491
492         if (result.next()) {
493             double b = Integer.parseInt(result.getString("nilaibobot_atribu
494             a = b / dibagi + "";
495         }
496
497     } catch (SQLException E) {
498         System.out.println("gagal dicetak karena :" + E);
499     }
500     System.out.println(a);
501     return a;
502 }
```

Program Normalisasi Jajan

```
504 public Object setNormalisasijajan(String bobot) {
505     String a = "";
506
507     double dibagi = 100;
508     try {
509         Statement statement = kon.getKoneksi().createStatement();
510         ResultSet result = statement.executeQuery("select * from atribut whe
511
512         if (result.next()) {
513             double b = Integer.parseInt(result.getString("nilaibobot_atribu
514             a = b / dibagi + "";
515         }
516
517     } catch (SQLException E) {
518         System.out.println("gagal dicetak karena :" + E);
519     }
520     System.out.println(a);
521     return a;
522 }
```

Program Normalisasi Kendaraan

```
524 public Object setNormalisasikendaraan(String bobot) {
525     String a = "";
526
527     double dibagi = 100;
528     try {
529         Statement statement = kon.getKoneksi().createStatement();
530         ResultSet result = statement.executeQuery("select * from atribut wha
531
532         if (result.next()) {
533             double b = Integer.parseInt(result.getString("nilaibobot_atribut
534             a = b / dibagi + "";
535         }
536
537     } catch (SQLException E) {
538         System.out.println("gagal dicetak karena :" + E);
539     }
540     System.out.println(a);
541     return a;
542 }
```

Program Normalisasi Peralatan

```
544 public Object setNormalisasiperalatan(String bobot) {
545     String a = "";
546
547     double dibagi = 100;
548     try {
549         Statement statement = kon.getKoneksi().createStatement();
550         ResultSet result = statement.executeQuery("select * from atribut wha
551
552         if (result.next()) {
553             double b = Integer.parseInt(result.getString("nilaibobot_atribut
554             a = b / dibagi + "";
555         }
556
557     } catch (SQLException E) {
558         System.out.println("gagal dicetak karena :" + E);
559     }
560     System.out.println(a);
561     return a;
562 }
```

Program Normalisasi Ranking

```
564 public Object setNormalisasiRanking(String bobot) {
565     String a = "";
566
567     double dibagi = 100;
568     try {
569         Statement statement = kon.getKoneksi().createStatement();
570         ResultSet result = statement.executeQuery("select * from atribut");
571
572         if (result.next()) {
573             double b = Integer.parseInt(result.getString("nilaibobot_atrib"));
574             a = b / dibagi + "";
575         }
576         System.out.println();
577     } catch (SQLException E) {
578         System.out.println("gagal dicetak karena : " + E);
579     }
580     System.out.println(a);
581     return a;
582 }
```

Program Nilai Sementara Pendapatan

```
265 class listener_tp_atrib implements ActionListener {
266
267     @Override
268     public void actionPerformed(ActionEvent e) {
269         if (!v.getTp_atrib().equals("-- Pilih Satu --")) {
270             v.setTp_nilai(m.pendapatan_atrib(v.getTp_atrib().toString()));
271             v.setTp_sementara((Double.parseDouble(v.getTp_nilai()) * Double
272         } else {
273             v.setTp_nilai("");
274             v.setTp_sementara("");
275         }
276         new hitung_hasil().hitung_hasil();
277     }
278
279 }
```

Program Nilai Sementara Saudara

```
249 class listener_js_atrib implements ActionListener {
250
251     @Override
252     public void actionPerformed(ActionEvent e) {
253         if (!v.getJs_atrib().equals("-- Pilih Satu --")) {
254             v.setJs_nilai(m.nilai_atrib(v.getJs_atrib().toString()));
255             v.setJs_sementara((Double.parseDouble(v.getJs_nilai()) * Double.
256         } else {
257             v.setJs_nilai("");
258             v.setJs_sementara("");
259         }
260         new hitung_hasil().hitung_hasil();
261     }
262 }
263 }
```

Program Nilai Sementara Status

```
281 class listener_sa_atrib implements ActionListener {
282
283     @Override
284     public void actionPerformed(ActionEvent e) {
285         if (!v.getSa_atrib().equals("-- Pilih Satu --")) {
286             v.setSa_nilai(m.status_atrib(v.getSa_atrib().toString()));
287             v.setSa_sementara((Double.parseDouble(v.getSa_nilai()) * Double.
288         } else {
289             v.setSa_nilai("");
290             v.setSa_sementara("");
291         }
292         new hitung_hasil().hitung_hasil();
293     }
294 }
```

Program Nilai Sementara Jajan

```
296 class listener_uj_atrib implements ActionListener {
297
298     @Override
299     public void actionPerformed(ActionEvent e) {
300         if (!v.getUj_atrib().equals("-- Pilih Satu --")) {
301             v.setUj_nilai(m.jajan_atrib(v.getUj_atrib().toString()));
302             v.setUj_sementara((Double.parseDouble(v.getUj_nilai()) * Double.
303         } else {
304             v.setSa_nilai("");
305             v.setSa_sementara("");
306         }
307         new hitung_hasil().hitung_hasil();
308     }
309 }
```

Program Nilai Sementara Kendaraan

```
311 class listener_kt_atrib implements ActionListener {
312
313     @Override
314     public void actionPerformed(ActionEvent e) {
315         if (!v.getKt_atrib().equals("-- Pilih Satu --")) {
316             v.setKt_nilai(m.kendaraan_atrib(v.getKt_atrib().toString()));
317             v.setKt_sementara((Double.parseDouble(v.getKt_nilai()) * Double
318         } else {
319             v.setKt_nilai("");
320             v.setKt_sementara("");
321         }
322         new hitung_hasil().hitung_hasil();
323     }
324 }
```

Program Nilai Sementara Peralatan

```
326 class listener_ps_atrib implements ActionListener {
327
328     @Override
329     public void actionPerformed(ActionEvent e) {
330         if (!v.getPs_atrib().equals("-- Pilih Satu --")) {
331             v.setPs_nilai(m.peralatan_atrib(v.getPs_atrib().toString()));
332             v.setPs_sementara((Double.parseDouble(v.getPs_nilai()) * Double
333         } else {
334             v.setPs_nilai("");
335             v.setPs_sementara("");
336         }
337         new hitung_hasil().hitung_hasil();
338     }
339 }
340 }
```

Program Nilai Sementara Ranking

```
342 class listener_rk_atrib implements ActionListener {
343
344     @Override
345     public void actionPerformed(ActionEvent e) {
346         if (!v.getRk_atrib().equals("-- Pilih Satu --")) {
347             v.setRk_nilai(m.ranking_atrib(v.getRk_atrib().toString()));
348             v.setRk_sementara((Double.parseDouble(v.getRk_nilai()) * Double
349         } else {
350             v.setRk_nilai("");
351             v.setRk_sementara("");
352         }
353         new hitung_hasil().hitung_hasil();
354     }
355 }
```

Program Hitung Hasil

```

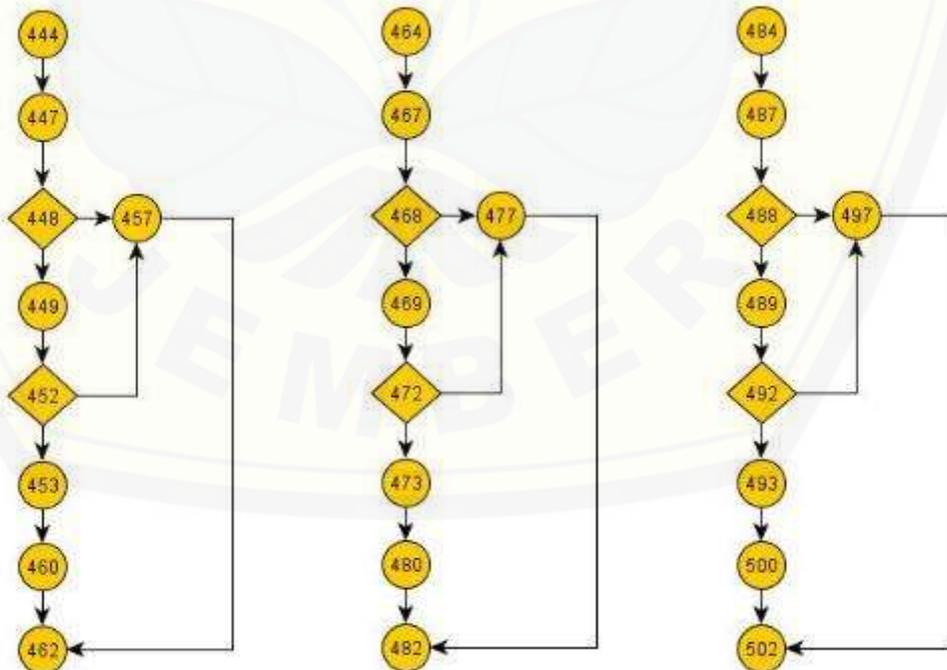
66 class hitung_hasil {
67
68     public void hitung_hasil() {
69         if (!v.getTp_sementara().equals(""))
70             && !v.getJs_sementara().equals("")
71             && !v.getSa_sementara().equals("")
72             && !v.getUj_sementara().equals("")
73             && !v.getKt_sementara().equals("")
74             && !v.getPs_sementara().equals("")
75             && !v.getRk_sementara().equals("")) {
76             System.out.println("masuk");
77             double hasil
78                 = Double.parseDouble(v.getTp_sementara()) + Double.parseDouble(v.getJs_sementara())
79                 + Double.parseDouble(v.getSa_sementara()) + Double.parseDouble(v.getUj_sementara())
80                 + Double.parseDouble(v.getKt_sementara()) + Double.parseDouble(v.getPs_sementara())
81                 + Double.parseDouble(v.getRk_sementara());
82             v.setHasil(hasil + "");
83         } else {
84             v.setHasil("");
85         }
86     }
87 }
    
```

A. Diagram Alir

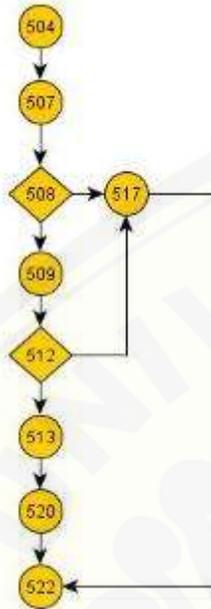
Normalisasi Pendapatan

Normalisasi Saudara

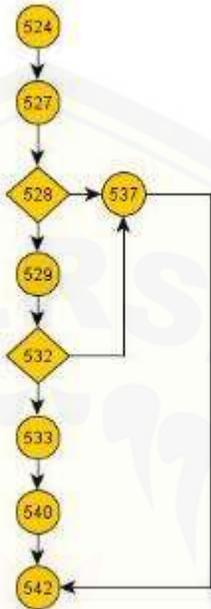
Normalisasi Status



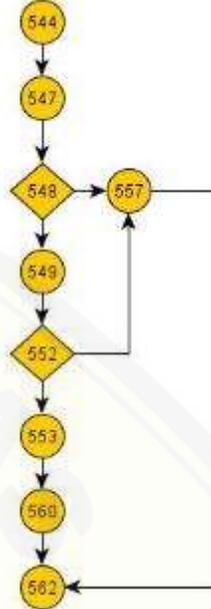
Normalisasi Jajan



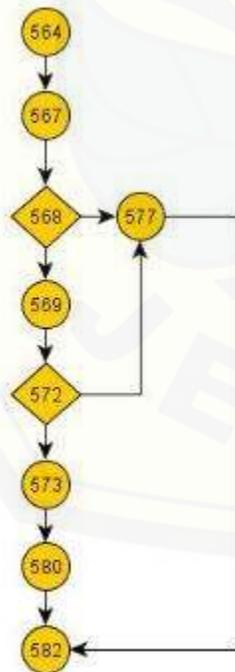
Normalisasi Kendaraan



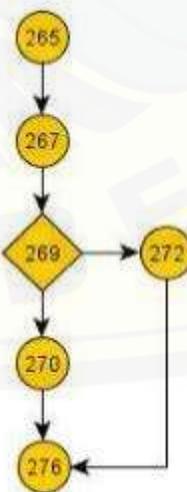
Normalisasi Peralatan



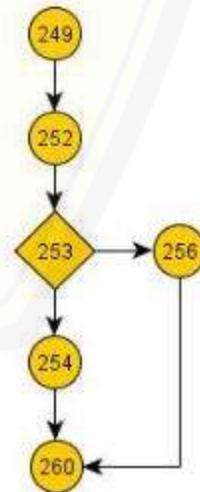
Normalisasi Ranking



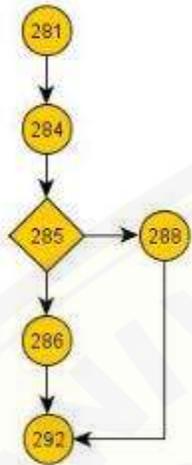
Nilai Pendapatan



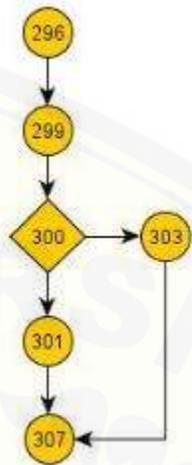
Nilai Saudara



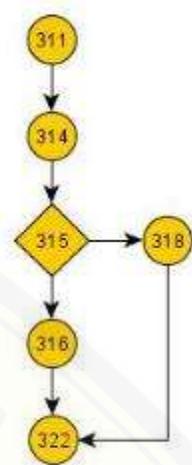
Nilai Status



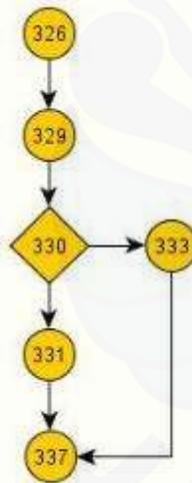
Nilai Uang Jajan



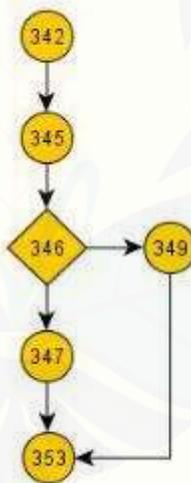
Nilai Kendaraan



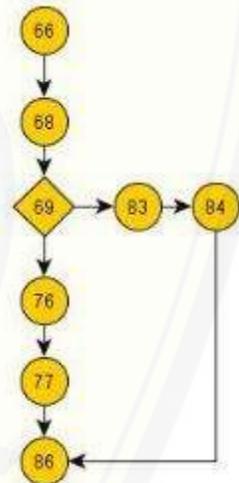
Nilai Peralatan



Nilai Ranking



Hitung Hasil



B. Cyclomatic Complexity

No.	Function	Cyclomatic Complexity
1.	Normalisasi Pendapatan	CC = Edge-Node+2 CC = 10-9+2 CC = 3
2.	Normalisasi Saudara	CC = Edge-Node+2 CC = 10-9+2 CC = 3
3.	Normalisasi Status	CC = Edge-Node+2 CC = 10-9+2 CC = 3
4.	Normalisasi Jajan	CC = Edge-Node+2 CC = 10-9+2 CC = 3
5.	Normalisasi Kendaraan	CC = Edge-Node+2 CC = 10-9+2 CC = 3
6.	Normalisasi Peralatan	CC = Edge-Node+2 CC = 10-9+2 CC = 3
7.	Normalisasi Ranking	CC = Edge-Node+2 CC = 10-9+2 CC = 3
8.	Nilai Pendapatan	CC = Edge-Node+2 CC = 6-6+2 CC = 2

9. Nilai Saudara	$CC = \text{Edge-Node}+2$ $CC = 6-6+2$ $CC = 2$
10. Nilai Status	$CC = \text{Edge-Node}+2$ $CC = 6-6+2$ $CC = 2$
11. Nilai Uang Jajan	$CC = \text{Edge-Node}+2$ $CC = 6-6+2$ $CC = 2$
12. Nilai Kendaraan	$CC = \text{Edge-Node}+2$ $CC = 6-6+2$ $CC = 2$
13. Nilai Peralatan	$CC = \text{Edge-Node}+2$ $CC = 6-6+2$ $CC = 2$
14. Nilai Ranking	$CC = \text{Edge-Node}+2$ $CC = 6-6+2$ $CC = 2$
15. Hitung Hasil	$CC = \text{Edge-Node}+2$ $CC = 8-8+2$ $CC = 2$

C. Jalur Independen

No.	Function	Jalur Independen
1.	Normalisasi Pendapatan	Jalur 1 = 1-2-3-4-6-7-8-9 Jalur 2 = 1-2-3-5-9 Jalur 3 = 1-2-3-4-5-6-9
2.	Normalisasi Saudara	Jalur 1 = 1-2-3-4-6-7-8-9 Jalur 2 = 1-2-3-5-9 Jalur 3 = 1-2-3-4-5-6-9
3.	Normalisasi Status	Jalur 1 = 1-2-3-4-6-7-8-9 Jalur 2 = 1-2-3-5-9 Jalur 3 = 1-2-3-4-5-6-9
4.	Normalisasi Jajan	Jalur 1 = 1-2-3-4-6-7-8-9 Jalur 2 = 1-2-3-5-9 Jalur 3 = 1-2-3-4-5-6-9
5.	Normalisasi Kendaraan	Jalur 1 = 1-2-3-4-6-7-8-9 Jalur 2 = 1-2-3-5-9 Jalur 3 = 1-2-3-4-5-6-9
6.	Normalisasi Peralatan	Jalur 1 = 1-2-3-4-6-7-8-9 Jalur 2 = 1-2-3-5-9 Jalur 3 = 1-2-3-4-5-6-9
7.	Normalisasi Ranking	Jalur 1 = 1-2-3-4-6-7-8-9 Jalur 2 = 1-2-3-5-9 Jalur 3 = 1-2-3-4-5-6-9
8.	Nilai Pendapatan	Jalur 1 = 1-2-3-4-6 Jalur 2 = 1-2-3-5-6
9.	Nilai Saudara	Jalur 1 = 1-2-3-4-6 Jalur 2 = 1-2-3-5-6

10. Nilai Status	Jalur 1 = 1-2-3-4-6 Jalur 2 = 1-2-3-5-6
11. Nilai Uang Jajan	Jalur 1 = 1-2-3-4-6 Jalur 2 = 1-2-3-5-6
12. Nilai Kendaraan	Jalur 1 = 1-2-3-4-6 Jalur 2 = 1-2-3-5-6
13. Nilai Peralatan	Jalur 1 = 1-2-3-4-6 Jalur 2 = 1-2-3-5-6
14. Nilai Ranking	Jalur 1 = 1-2-3-4-6 Jalur 2 = 1-2-3-5-6
15. Hitung Hasil	Jalur 1 = 1-2-3-4-5-8 Jalur 2 = 1-2-3-6-7-8

D. Test Case

Normalisasi Pendapatan

Jalur 1	
<i>Test Case</i>	Jika koneksi berhasil dan result query terus berjalan, maka menghitung nilai a dengan rumus b dibagi dengan 100
Target yang diharapkan	Menampilkan hasil a
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-6-7-8-9
Jalur 2	
<i>Test Case</i>	Jika koneksi gagal dan result query tidak berjalan maka mencetak warning gagal
Target yang diharapkan	Mencetak warning gagal
Hasil Pengujian	Benar
Path/Jalur	1-2-3-5-9
Jalur 3	
<i>Test Case</i>	Jika koneksi berhasil, namun result.next() tidak berjalan berulang maka menjalankan fungsi catch
Target yang diharapkan	Mencetak warning gagal
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-5-6-9

Normalisasi Saudara

Jalur 1	
<i>Test Case</i>	Jika koneksi berhasil dan result query terus berjalan, maka menghitung nilai a dengan rumus b dibagi dengan 100
Target yang diharapkan	Menampilkan hasil a
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-6-7-8-9
Jalur 2	
<i>Test Case</i>	Jika koneksi gagal dan result query tidak berjalan maka mencetak warning gagal
Target yang diharapkan	Mencetak warning gagal
Hasil Pengujian	Benar
Path/Jalur	1-2-3-5-9
Jalur 3	
<i>Test Case</i>	Jika koneksi berhasil, namun result.next() tidak berjalan berulang maka menjalankan fungsi catch
Target yang diharapkan	Mencetak warning gagal
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-5-6-9

Normalisasi Status

Jalur 1	
<i>Test Case</i>	Jika koneksi berhasil dan result query terus berjalan, maka menghitung nilai a dengan rumus b dibagi dengan 100
Target yang diharapkan	Menampilkan hasil a
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-6-7-8-9
Jalur 2	
<i>Test Case</i>	Jika koneksi gagal dan result query tidak berjalan maka mencetak warning gagal
Target yang diharapkan	Mencetak warning gagal
Hasil Pengujian	Benar
Path/Jalur	1-2-3-5-9
Jalur 3	
<i>Test Case</i>	Jika koneksi berhasil, namun result.next() tidak berjalan berulang maka menjalankan fungsi catch
Target yang diharapkan	Mencetak warning gagal
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-5-6-9

Normalisasi Uang Jajan

Jalur 1	
<i>Test Case</i>	Jika koneksi berhasil dan result query terus berjalan, maka menghitung nilai a dengan rumus b dibagi dengan 100
Target yang diharapkan	Menampilkan hasil a
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-6-7-8-9
Jalur 2	
<i>Test Case</i>	Jika koneksi gagal dan result query tidak berjalan maka mencetak warning gagal
Target yang diharapkan	Mencetak warning gagal
Hasil Pengujian	Benar
Path/Jalur	1-2-3-5-9
Jalur 3	
<i>Test Case</i>	Jika koneksi berhasil, namun result.next() tidak berjalan berulang maka menjalankan fungsi catch
Target yang diharapkan	Mencetak warning gagal
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-5-6-9

Normalisasi Kendaraan

Jalur 1	
<i>Test Case</i>	Jika koneksi berhasil dan result query terus berjalan, maka menghitung nilai a dengan rumus b dibagi dengan 100
Target yang diharapkan	Menampilkan hasil a
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-6-7-8-9
Jalur 2	
<i>Test Case</i>	Jika koneksi gagal dan result query tidak berjalan maka mencetak warning gagal
Target yang diharapkan	Mencetak warning gagal
Hasil Pengujian	Benar
Path/Jalur	1-2-3-5-9
Jalur 3	
<i>Test Case</i>	Jika koneksi berhasil, namun result.next() tidak berjalan berulang maka menjalankan fungsi catch
Target yang diharapkan	Mencetak warning gagal
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-5-6-9

Normalisasi Peralatan

Jalur 1	
<i>Test Case</i>	Jika koneksi berhasil dan result query terus berjalan, maka menghitung nilai a dengan rumus b dibagi dengan 100
Target yang diharapkan	Menampilkan hasil a
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-6-7-8-9
Jalur 2	
<i>Test Case</i>	Jika koneksi gagal dan result query tidak berjalan maka mencetak warning gagal
Target yang diharapkan	Mencetak warning gagal
Hasil Pengujian	Benar
Path/Jalur	1-2-3-5-9
Jalur 3	
<i>Test Case</i>	Jika koneksi berhasil, namun result.next() tidak berjalan berulang maka menjalankan fungsi catch
Target yang diharapkan	Mencetak warning gagal
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-5-6-9

Normalisasi Ranking

Jalur 1	
<i>Test Case</i>	Jika koneksi berhasil dan result query terus berjalan, maka menghitung nilai a dengan rumus b dibagi dengan 100
Target yang diharapkan	Menampilkan hasil a
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-6-7-8-9
Jalur 2	
<i>Test Case</i>	Jika koneksi gagal dan result query tidak berjalan maka mencetak warning gagal
Target yang diharapkan	Mencetak warning gagal
Hasil Pengujian	Benar
Path/Jalur	1-2-3-5-9
Jalur 3	
<i>Test Case</i>	Jika koneksi berhasil, namun result.next() tidak berjalan berulang maka menjalankan fungsi catch
Target yang diharapkan	Mencetak warning gagal
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-5-6-9

Nilai Sementara Pendapatan

Jalur 1	
<i>Test Case</i>	Jika atribut terpilih maka menghitung nilai sementara dengan rumus nilai utility * normalisasi, kemudian menjalankan fungsi hitung hasil
Target yang diharapkan	Nilai sementara pada fungsi hitung hasil yang dijalankan diisi dengan hasil fase ini
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-6
Jalur 2	
<i>Test Case</i>	Jika atribut tidak terpilih maka nilai sementara di set “ ”, dan menjalankan fungsi hitung hasil
Target yang diharapkan	Nilai sementara pada fungsi hitung hasil diisi dengan kosong “ ”
Hasil Pengujian	Benar
Path/Jalur	1-2-3-5-6

Nilai Sementara Saudara

Jalur 1	
<i>Test Case</i>	Jika atribut terpilih maka menghitung nilai sementara dengan rumus nilai utility * normalisasi, kemudian menjalankan fungsi hitung hasil
Target yang diharapkan	Nilai sementara pada fungsi hitung hasil yang dijalankan diisi dengan hasil fase ini
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-6

Jalur 2	
<i>Test Case</i>	Jika atribut tidak terpilih maka nilai sementara di set “ ”, dan menjalankan fungsi hitung hasil
Target yang diharapkan	Nilai sementara pada fungsi hitung hasil diisi dengan kosong “ ”
Hasil Pengujian	Benar
Path/Jalur	1-2-3-5-6

Nilai Sementara Status

Jalur 1	
<i>Test Case</i>	Jika atribut terpilih maka menghitung nilai sementara dengan rumus nilai utility * normalisasi, kemudian menjalankan fungsi hitung hasil
Target yang diharapkan	Nilai sementara pada fungsi hitung hasil yang dijalankan diisi dengan hasil fase ini
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-6

Jalur 2	
<i>Test Case</i>	Jika atribut tidak terpilih maka nilai sementara di set “ ”, dan menjalankan fungsi hitung hasil
Target yang diharapkan	Nilai sementara pada fungsi hitung hasil diisi dengan kosong “ ”
Hasil Pengujian	Benar
Path/Jalur	1-2-3-5-6

Nilai Sementara Uang Jajan

Jalur 1	
<i>Test Case</i>	Jika atribut terpilih maka menghitung nilai sementara dengan rumus nilai utility * normalisasi, kemudian menjalankan fungsi hitung hasil
Target yang diharapkan	Nilai sementara pada fungsi hitung hasil yang dijalankan diisi dengan hasil fase ini
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-6

Jalur 2	
<i>Test Case</i>	Jika atribut tidak terpilih maka nilai sementara di set “ ”, dan menjalankan fungsi hitung hasil
Target yang diharapkan	Nilai sementara pada fungsi hitung hasil diisi dengan kosong “ ”
Hasil Pengujian	Benar
Path/Jalur	1-2-3-5-6

Nilai Sementara Kendaraan

Jalur 1	
<i>Test Case</i>	Jika atribut terpilih maka menghitung nilai sementara dengan rumus nilai utility * normalisasi, kemudian menjalankan fungsi hitung hasil
Target yang diharapkan	Nilai sementara pada fungsi hitung hasil yang dijalankan diisi dengan hasil fase ini
Hasil Pengujian	Benar

Path/Jalur	1-2-3-4-6
Jalur 2	
<i>Test Case</i>	Jika atribut tidak terpilih maka nilai sementara di set “ ”, dan menjalankan fungsi hitung hasil
Target yang diharapkan	Nilai sementara pada fungsi hitung hasil diisi dengan kosong “ ”
Hasil Pengujian	Benar
Path/Jalur	1-2-3-5-6

Nilai Sementara Peralatan

Jalur 1	
<i>Test Case</i>	Jika atribut terpilih maka menghitung nilai sementara dengan rumus nilai utility * normalisasi, kemudian menjalankan fungsi hitung hasil
Target yang diharapkan	Nilai sementara pada fungsi hitung hasil yang dijalankan diisi dengan hasil fase ini
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-6
Jalur 2	
<i>Test Case</i>	Jika atribut tidak terpilih maka nilai sementara di set “ ”, dan menjalankan fungsi hitung hasil
Target yang diharapkan	Nilai sementara pada fungsi hitung hasil diisi dengan kosong “ ”
Hasil Pengujian	Benar
Path/Jalur	1-2-3-5-6

Nilai Sementara Ranking

Jalur 1	
<i>Test Case</i>	Jika atribut terpilih maka menghitung nilai sementara dengan rumus nilai utility * normalisasi, kemudian menjalankan fungsi hitung hasil
Target yang diharapkan	Nilai sementara pada fungsi hitung hasil yang dijalankan diisi dengan hasil fase ini
Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-6
Jalur 2	
<i>Test Case</i>	Jika atribut tidak terpilih maka nilai sementara di set “ ”, dan menjalankan fungsi hitung hasil
Target yang diharapkan	Nilai sementara pada fungsi hitung hasil diisi dengan kosong “ ”
Hasil Pengujian	Benar
Path/Jalur	1-2-3-5-6

Fungsi Hitung Hasil

Jalur 1	
<i>Test Case</i>	Jika semua nilai atribut tidak kosong (“ ”) maka dilakukan proses penghitungan hasil dengan cara menjumlahkan seluruh nilai atribut, kemudian mencetak nilai hasil
Target yang diharapkan	Mencetak nilai hasil sesuai perhitungan

Hasil Pengujian	Benar
Path/Jalur	1-2-3-4-5-8
Jalur 2	
<i>Test Case</i>	Jika semua nilai atribut kosong (“ ”) maka langsung mencetak nilai hasil dengan kosong (“”)
Target yang diharapkan	Mencetak nilai hasil dengan (“ ”)
Hasil Pengujian	Benar
Path/Jalur	1-2-3-6-7-8

4.5.2 Metode Black Box

Pengujian dengan menggunakan metode black box bertujuan untuk menguji sistem dari segi spesifikasi fungsional yang terdapat pada sistem yang bertujuan untuk mengetahui apakah fungsi yang terdapat pada sistem yang telah dibangun serta inputan dan keluaran pada sistem sesuai dengan spesifikasi yang diinginkan oleh pengguna. Pengujian ini dilakukan oleh pihak SMAN 1 GIRI banyuwangi yaitu pada OSIS dan pada guru. Hasil pengujian black box dapat dilihat pada lampiran G

BAB 6. PENUTUP

Bab ini berisi mengenai kesimpulan dan saran dari peneliti tentang pelunisan penelitian yang telah dilakukan. Kesimpulan dan saran tersebut diharapkan dapat digunakan sebagai acuan dalam penelitian selanjutnya ataupun pengembangan penelitian dengan topic yang sama.

6.1 Kesimpulan

Kesimpulan dari hasil penelitian yang ditelaah dilakukan oleh peneliti adalah sebagai berikut :

1. Proses pemilihan Siswa Asuh Sebaya dapat lebih diperkuat dengan menggunakan sistem informasi siswa asuh sebaya dengan mengkombinasikan metode Simple Multy Rating Technique (SMART) sebagai pembantu. SMART diimplementasikan pada pemilihan siswa asuh sebaya dengan membuat sebuah kriteria yang akan digunakan sebagai pembanding dalam penilaian, kriteria tersebut juga harus memiliki sebuah atribut untuk menunjang penilaian pada setiap kriteria. dalam penelitian ini kriteria yang dipakai berjumlah tujuh kriteria dengan masing masing atribut. Pada tahapan selanjutnya membuat bobot untuk setiap kriteria yang digunakan yang bertujuan untuk pembanding tingkat kepentingan antar bobot. Dengan adanya data tersebut metode SMART dapat mengolah data dengan menggunakan data yang ada dan diproses dalam metode sehingga dapat menghasilkan pendukung keputusan untuk siswa yang akan mendapatkan bantuan siswa asuh sebaya.

2. Pada merancang dan membangun sistem informasi siswa asuh sebaya diperlukan pemahaman dalam metode SMART. Pemahaman tentang pemanfaatan metode SMART yang digunakan dengan kasus permasalahan yang tepat seperti rekomendasi siswa asuh sebaya. Pada penelitian ini pemahaman yang dimaksud dimana mengharuskan pengguna memahami apa saja yang dibutuhkan seperti data kriteria, data atribut pada kriteria serta bobot setiap kriteria yang akan digunakan, sehingga dapat diimplementasikan kepada sistem. Dengan demikian metode SMART dapat digunakan dengan maksimal dengan mendapatkan hasil yang maksimal.
3. Sistem Informasi Siswa Asuh Sebaya yang dibangun dengan mengimplementasikan metode SMART didalamnya dapat lebih mudah dan lebih dapat menghasilkan hasil yang akurat, karena pada perhitungan metode SMART yang dilakukan secara manual dapat diimplementasikan secara benar pada sistem, jadi proses pemilihan siswa asuh sebaya yang untuk direkomendasikan dapat dengan cepat serta akurat sesuai penilaian yang ada. Disamping itu sistem informasi siswa asuh sebaya dapat menyesuaikan penilaian yang dipakai oleh sekolah, jika sewaktu waktu atau setiap tahun berubah maka bobot bisa disesuaikan dengan penilaian yang diberikan oleh pihak sekolah. Pada penelitian ini dibahas dengan membandingkan dengan dua perhitungan yang berbeda dengan nilai bobot yang berbeda dan hasil rekomendasinya juga berbeda. Pada penelitian ini dengan studi kasus yang dipakai di sekolah dapat di peroleh perbandingan bobot yang tepat untuk mendapatkan hasil yang sama dengan pihak sekolah yaitu dengan Total Pendapatan : 30, Jumlah saudara : 25, Status anak : 5, Uang jajan sekolah : 10, Kendaraan transportasi : 15, Peralatan sekolah : 10, Rangkaing kelas : 5. Jika sekolah merubah kebijakan untuk mendapatkan nilainya maka sistem akan menyesuaikan kembali bobot yang dipakai oleh sekolah untuk memperoleh hasil yang cocok untuk standar penilaian yang baru. Dengan

demikian user dapat mengetahui nilai bobot yang sesuai dengan yang dikehendaki dan menghasilkan rekomendasi keputusan yang sesuai yang dikehendaki oleh user.

6.2 Saran

Beberapa saran dan masukan berikut diharapkan dapat memberikan perbaikan sistem dalam penelitian selanjutnya, antara lain

1. Sistem informasi siswa asuh sebaya dapat digunakan disemua sekolah dibanyuwangi dengan persyaratan dan ketentuan standart nilai yang berbeda-beda.
2. Sistem informasi siswa asuh sebaya dapat digunakan secara fleksibel dengan mampu menambah kriteria yang ada.
3. Sistem informasi siswa asuh sebaya dapat dilengkapi dengan daftar semua siswa yang ada di sekolah, agar mampu menyimpan data pribadi semua murid pada sistem.
4. Pengguna yang akan mengoprasikan sistem ini harus menguji sistem dengan perbandingan bobot tertentu sampai mendapatkan perbandingan bobot yang sesuai dan benar untuk digunakan oleh sistem siswa asuh sebaya.



DAFTAR PUSTAKA

- Atiqah. 2013. *“Implementasi Metode Smart Pada Sistem Pendukung Keputusan Pemilihan Pembelian Mobil Keluarga”*, dalam Pelita Informatika Budi Darma. Program Studi Teknik Informatika STMIK Budi Darma Medan, Jl. Sisingamangaraja No. 338 Sp. Limun Medan
- Fahrurrozi. Imam. 2005. *“Proses Pemodelan Software Dengan Metode Waterfall Dan Extreme Programming: Studi Perbandingan”*. Program Studi Ilmu Komputer, Universitas Gadjah Mada
- Kustiyahningsih. Yeni, dkk . 2013. *“Sistem Pendukung Keputusan Untuk Menentukan Jurusan Pada Siswa Sma Menggunakan Metode Knn Dan Smart”*. Jurusan Teknik Informatika, Fakultas Teknik, Universitas Trunojoyo, Jl. Raya Telang PO. BOX 2 Kamal, Bangkalan, Madura.
- Pemkab Banyuwangi. 2014. *Program Siswa Asuh Sebaya Banyuwangi Jadi Nominator MDGs Award*. Diakses pada tanggal 18-maret-2015 <http://banyuwangikab.go.id/berita-daerah/program-siswa-asuh-sebaya-banyuwangi-jadi-nominator-mdgs-award.html>
- Rouf. Abdul. 2012. *“Pengujian Perangkat Lunak Dengan Menggunakan Metode White Box Dan Black Box”*. Sistem Informasi – STMIK HIMSYA Semarang
- Sugiyono , (2008), *“Metode Penelitian Kuantitatif kualitatif dan R&D”* Alfabeta : Bandung
- UPTDSD. 2012. Daftar Nama Sekolah Dasar Se-Kabupaten Banyuwangi. Diakses pada tanggal 18-maret-2015 https://docs.google.com/spreadsheet/ccc?key=0An5m_4toMHFDdEVadVh6Q29ES2wxN1UzcE9kRFRfWFE#gid=0
- Sugiyono. 2010. *Metode Penelitian Kuantitatif Kualitatif & RND*. Bandung : Alfabeta.
- McMillan, J.H., dan Schumacher, Sally. 2006. *Research in Education*. New Jersey : Pearson.

LAMPIRANA. Lampiran *Use Case Scenario*A.1. *Use Case Scenario Login*

ID	US-01
Name	Login
Participating Actor	OSIS, dan Guru
Entry Condition	Aktor akan melakukan login
Exit Condition	Aktor telah melakukan login

**SKENARIO UTAMA
"Login"**

Aksi Aktor	Reaksi Sistem
1. Buka halaman sistem informasi siswa asuh sebaya	
	2. Menampilkan halaman login
3. Inputkan Username dan Password sesuai dengan hak akses	
4. Klik sign in	
	5. Menampilkan halaman dashboard

**SKENARIO ALTERNATIF
"Salah Username dan Password"**

Reaksi Aktor	Sistem
3a. Inputkan Username dan Password sebagai manager	
4. a Klik sign in	

5a. Menampilkan Username atau Password SALAH!
Silahkan periksa dan coba kembali

A.2. Use Case Scenario CRUD SAS

ID	US-02
Name	Input SAS
Participating Actor	Guru
Entry Condition	Aktor akan Input, Update, Delete
Exit Condition	Aktor telah Input, Update, Delete

SKENARIO UTAMA “A CRUD SAS (Input)”

Aksi Aktor	Reaksi Sistem
1. Buka halaman sistem informasi siswa asuh sebaya	
2. Klik menu input SAS	
	3. Menampilkan halaman Input SAS
4. Mengisi NISN, Nama, Tanggal Lahir, Nama Orang Tua, Alamat, Kelas, No Telp, Tahun Ajaran	
5. Klik Button Simpan	
	6. Menyimpan data dan menampilkan data NISN, Nama, Tanggal Lahir, Nama Orang Tua,

Alamat, Kelas, No Telp, Tahun
Ajaran pada Tabel

“B CRUD SAS (Update)”

3. Klik data siswa pada tabel

4. Menampilkan data siswa yang
dipilih pada form

5. Merubah data pada Form

6. Klik Update

7. Menyimpan data dan menampilkan
data NISN, Nama, Tanggal
Lahir, Nama Orang Tua, Alamat,
Kelas, No Telp, Tahun Ajaran
pada Tabel

“ C CRUD SAS (Delete) “

3. Klik data siswa pada tabel

4. Menampilkan data siswa yang
dipilih pada form

5. Klik Delete

7. Menghapus data NISN, Nama,
Tanggal Lahir, Nama Orang Tua,
Alamat, Kelas, No Telp, Tahun
Ajaran pada Tabel

**SKENARIO ALTERNATIF
“Form Tidak Terisi Lengkap”**

Reaksi Aktor	Sistem
4 a. Mengisi NISN, Nama, Tanggal Lahir, Nama Orang Tua, Alamat, Kelas, No Telp, Tahun Ajaran	
5 a. Klik Button Simpan	
	6 a. Menampilkan Kolom (Nama Form Yang Kosong) masih kosong, tolong diisi!!!

A.3. *Use Case Scenario* VIEW CALON PENERIMA SAS

ID	US-03
Name	<i>VIEW CALON PENERIMA SAS</i>
Participating Actor	OSIS
Entry Condition	Aktor akan melihat calon penrima SAS
Exit Condition	Aktor telah melihat calon penrima SAS

**SKENARIO UTAMA
“View Calon penerima SAS”**

Aksi Aktor	Reaksi Sistem
1. Buka halaman sistem informasi siswa asuh sebaya	
2. Klik menu “Home”	
	3. Menampilkan data siswa calon penerima SAS

A.4. Use Case Scenario CRUD Total Pendapatan Orang Tua

ID	US-04
Name	Total Pendapatan Orang Tua
Participating Actor	OSIS
Entry Condition	Aktor akan Input, Update, Delete Pendapatan Orang Tua
Exit Condition	Aktor telah Input, Update, Delete Pendapatan Orang Tua

SKENARIO UTAMA
“A CRUD Total Pendapatan Orang Tua (Input)”

Aksi Aktor	Reaksi Sistem
1. Buka halaman sistem informasi siswa asuh sebaya	
2. Klik menu Update Kriteria	
3. Klik menu Total Pendapatan Orang Tua	
	4. Menampilkan halaman Total Pendapatan Orang Tua
5. Mengisi Nama Atribut, Nilai Atribut	
6. Klik Button Insert	
	7. Menyimpan data dan menampilkan data Id_Pendapatan, Nama Atribut, Dan Nilai Atribut pada Tabel

“B CRUD Total Pendapatan Orang Tua (Update)”

4. Klik data Total pendapatan orang tua

5. Menampilkan data total pendapatan orang tua

6. Merubah data pada Form

7. Klik Update

8. Menyimpan data dan menampilkan data Id_Pendapatan, Nama Atribut, Dan Nilai Atribut pada Tabel

“ C Total Pendapatan Orang Tua (Delete) “

3. Klik data total pendapatan pada tabel

4. Menampilkan data total pendapatan orang tua

5. Klik Delete

6. Menghapus data Id_Pendapatan, Nama Atribut, Dan Nilai Atribut pada Tabel

**SKENARIO ALTERNATIF
“Form Tidak Terisi Lengkap”****Reaksi Aktor****Sistem**

5 a. Klik Button Insert

6 a. Menampilkan AlertData pada form belum lengkap

7 a. Klik Ok

A.5. Use Case Scenario CRUD Jumlah Saudara

ID	US-05
Name	Jumlah Saudara
Participating Actor	OSIS
Entry Condition	Aktor akan Input, Update, Delete Jumlah Saudara
Exit Condition	Aktor telah Input, Update, Delete Jumlah Saudara

SKENARIO UTAMA
“A CRUD Jumlah Saudara (Input)”

Aksi Aktor	Reaksi Sistem
1. Buka halaman sistem informasi siswa asuh sebaya	
2. Klik menu Update Kriteria	
3. Klik menu Jumlah Saudara	
	4. Menampilkan halaman Jumlah Saudara
5. Mengisi Nama Atribut, Nilai Atribut	
6. Klik Button Insert	
	7. Menyimpan data dan menampilkan data Id_Saudara, Nama Atribut, Dan Nilai Atribut pada Tabel

“B CRUD Jumlah Saudara (Update)”

4. Klik data Jumlah Saudara

5. Menampilkan data Jumlah Saudara

6. Merubah data pada Form

7. Klik Update

8. Meyimpan data dan menampilkan data Id_Saudara, Nama Atribut, Dan Nilai Atribut pada Tabel

“ C Jumlah Sudara (Delete) “

3. Klik data saudara pada tabel

4. Menampilkan data Jumlah Saudara

5. Klik Delete

6. Mendelete data Id_Saudara, Nama Atribut, Dan Nilai Atribut pada Tabel

**SKENARIO ALTERNATIF
“Jumlah Saudara”****Reaksi Aktor****Sistem**

5 a. Klik Button Insert

6 a. Menampilkan AlertData pada form belum lengkap

7 a. Klik Ok

A.6. Use Case Scenario CRUD Status Anak

ID	US-06
Name	Status Anak
Participating Actor	OSIS
Entry Condition	Aktor akan Input, Update, Delete Status Anak
Exit Condition	Aktor telah Input, Update, Delete Status Anak

SKENARIO UTAMA
“A CRUD Status Anak (Input)”

Aksi Aktor	Reaksi Sistem
1. Buka halaman sistem informasi siswa asuh sebaya	
2. Klik menu Update Kriteria	
3. Klik menu Status Anak	
	4. Menampilkan halaman Status Anak
5. Mengisi Nama Atribut, Nilai Atribut	
6. Klik Button Insert	
	7. Menyimpan data dan menampilkan data Id_StatusAnak, Nama Atribut, Dan Nilai Atribut pada Tabel

“B CRUD Status Anak (Update)”

4. Klik data Status Anak	
	5. Menampilkan data Status Anak

6. Merubah data pada Form

7. Klik Update

8. Meyimpan data dan menampilkan data Id_StatusAnak, Nama Atribut, Dan Nilai Atribut pada Tabel

“ C Status Anak (Delete) “

3. Klik data status anak pada tabel

4. Menampilkan data Status anak

5. Klik Delete

6. Mendelete data Id_StatusAnak, Nama Atribut, Dan Nilai Atribut pada Tabel

SKENARIO ALTERNATIF “Status Anak”

Reaksi Aktor

Sistem

5 a. Klik Button Insert

6 a. Menampilkan AlertData pada form belum lengkap

7 a. Klik Ok

A.7. Use Case Scenario CRUD Uang Jajan

ID	US-07
Name	Uang Jajan
Participating Actor	OSIS
Entry Condition	Aktor akan Input, Update, Delete Uang Jajan
Exit Condition	Aktor telah Input, Update, Delete Uang Jajan

SKENARIO UTAMA
“A CRUD Uang Jajan (Input)”

Aksi Aktor	Reaksi Sistem
1. Buka halaman sistem informasi siswa asuh sebaya	
2. Klik menu Update Kriteria	
3. Klik menu Uang Jajan	
	4. Menampilkan halaman Uang Jajan
5. Mengisi Nama Atribut, Nilai Atribut	
6. Klik Button Insert	
	7. Menyimpan data dan menampilkan data Id_UangJajan, Nama Atribut, Dan Nilai Atribut pada Tabel

“B CRUD Uang Jajan (Update)”

4. Klik data Uang Jajan	
	5. Menampilkan data Uang Jajan

6. Merubah data pada Form

7. Klik Update

8. Meyimpan data dan menampilkan data Id_UangJajan, Nama Atribut, Dan Nilai Atribut pada Tabel

“ C Uang Jajan (Delete) “

3. Klik Uang Jajan anak pada tabel

4. Menampilkan data Uang Jajan

5. Klik Delete

6. Mendelete data Id_UangJajan, Nama Atribut, Dan Nilai Atribut pada Tabel

SKENARIO ALTERNATIF “Uang Jajan”

Reaksi Aktor	Sistem
5 a. Klik Button Insert	6 a. Menampilkan AlertData pada form belum lengkap
7 a. Klik Ok	

A.8. Use Case Scenario CRUD Kendaraan Ke Sekolah

ID	US-08
Name	Kendaraan Ke Sekolah
Participating Actor	OSIS
Entry Condition	Aktor akan Input, Update, Delete Kendaraan Sekolah
Exit Condition	Aktor telah Input, Update, Delete Kendaraan Sekolah

SKENARIO UTAMA
“A CRUD Kendaraan Sekolah (Input)”

Aksi Aktor	Reaksi Sistem
1. Buka halaman sistem informasi siswa asuh sebaya	
2. Klik menu Update Kriteria	
3. Klik menu Kendaraan Ke Sekolah	
	4. Menampilkan halaman Kendaraan Sekolah
5. Mengisi Nama Atribut, Nilai Atribut	
6. Klik Button Insert	
	7. Menyimpan data dan menampilkan data Id_KendaraanSekolah, Nama Atribut, Dan Nilai Atribut pada Tabel

“B CRUD Kendaraan ke Sekolah (Update)”

4. Klik data Uang Jajan

5. Menampilkan data Kendaraan Sekolah

6. Merubah data pada Form

7. Klik Update

8. Meyimpan data dan menampilkan data Id_KendaraanSekolah, Nama Atribut, Dan Nilai Atribut pada Tabel

“ C Kendaraan Sekolah (Delete) “

3. Klik data Kendaraan ke Sekolah pada tabel

4. Menampilkan Kendaraan Ke Sekolah

5. Klik Delete

6. Mendelete data Id_KendaraanSekolah, Nama Atribut, Dan Nilai Atribut pada Tabel

**SKENARIO ALTERNATIF
“Kendaraan Ke Sekolah”****Reaksi Aktor****Sistem**

5 a. Klik Button Insert

6 a. Menampilkan AlertData pada

form belum lengkap

7 a. Klik Ok

A.9. Use Case Scenario CRUD Peralatan Sekolah

ID	US-09
Name	Peralatan Sekolah
Participating Actor	OSIS
Entry Condition	Aktor akan Input, Update, Delete Peralatan Sekolah
Exit Condition	Aktor telah Input, Update, Delete Peralatan Sekolah

SKENARIO UTAMA
“A CRUD Peralatan Sekolah (Input)”

Aksi Aktor	Reaksi Sistem
1. Buka halaman sistem informasi siswa asuh sebaya	
2. Klik menu Update Kriteria	
3. Klik menu Peralatan Sekolah	
	4. Menampilkan halaman Peralatan Sekolah
5. Mengisi Nama Atribut, Nilai Atribut	
6. Klik Button Insert	
	7. Menyimpan data dan menampilkan data Id_PeralatanSekolah, Nama Atribut, Dan Nilai Atribut pada

Tabel

“B CRUD Peralatan Sekolah (Update)”

4. Klik data Peralatan Sekolah

5. Menampilkan data Peralatan Sekolah

6. Merubah data pada Form

7. Klik Update

8. Meyimpan data dan menampilkan data Id_PeralatanSekolah, Nama Atribut, Dan Nilai Atribut pada Tabel

“ C Peralatan Sekolah (Delete) “

3. Klik data Peralatan Sekolah pada tabel

4. Menampilkan Kendaraan Ke Sekolah

5. Klik Delete

7. Mendelete data Id_PeralatanSekolah, Nama Atribut, Dan Nilai Atribut pada Tabel

**SKENARIO ALTERNATIF
“Peralatan Sekolah”**

Reaksi Aktor	Sistem
5 a. Klik Button Insert	
	6 a. Menampilkan AlertData pada form belum lengkap
7 a. Klik Ok	

A.10. Use Case Scenario CRUD Rangking Kelas

ID	US-10
Name	Rangking Kelas
Participating Actor	OSIS
Entry Condition	Aktor akan Input, Update, Delete Rangking Kelas
Exit Condition	Aktor telah Input, Update, Delete Rangking Kelas

**SKENARIO UTAMA
“A CRUD Rangking Kelas (Input)”**

Aksi Aktor	Reaksi Sistem
1. Buka halaman sistem informasi siswa asuh sebaya	
2. Klik menu Update Kriteria	
3. Klik menu Rangking Kelas	
	4. Menampilkan halaman Rangking Rangking
5. Mengisi Nama Atribut, Nilai Atribut	

6. Klik Button Insert

7. Menyimpan data dan menampilkan data Id_RangkingKelas, Nama Atribut, Dan Nilai Atribut pada Tabel

“B CRUD Rangking Kelas (Update)”

4. Klik data Rangking Kelas

5. Menampilkan data Rangking Kelas

6. Merubah data pada Form

7. Klik Update

8. Meyimpan data dan menampilkan data Id_Rangking Kelas, Nama Atribut, Dan Nilai Atribut pada Tabel

“ C Rangking Kelas (Delete) “

3. Klik data Rangking Kelas pada tabel

4. Menampilkan Rangking Kelas

5. Klik Delete

6. Mendelete data Id_RangkingKelas, Nama Atribut, Dan Nilai Atribut pada Tabel

**SKENARIO ALTERNATIF
“Rangking Kelas”**

Reaksi Aktor	Sistem
5 a. Klik Button Insert	
	6 a. Menampilkan AlertData pada form belum lengkap
7 a. Klik Ok	

A.11. Use Case Scenario EDIT BOBOT

ID	US-11
Name	Edit Bobot
Participating Actor	OSIS
Entry Condition	Aktor akan Update, Cek Nilai Bobot Kriteria
Exit Condition	Aktor telah Update, Cek Nilai Bobot Kriteria

**SKENARIO UTAMA
“A Rangking Kelas Update”**

Aksi Aktor	Reaksi Sistem
1. Buka halaman sistem informasi siswa asuh sebaya	
2. Klik menu Update Bobot	
	3. Menampilkan halaman Update Bobot
4. Pilih Data Atribut Bobot pada tabel	
	5. Menampilkan data id_bobot, nama kriteri dan Nilai bobot

6. Merubah Nilai bobot pada Form

7. Klik Update

8. Sistem menyimpan dan menampilkan data id_bobot, nama kriteria dan nilai bobot pada tabel

“B Cek Total Nilai Bobot”

4. Klik cek total nilai bobot

5. Menampilkan Alert Nilai anda sudah : 100

6. Klik Ok

7. Menampilkan halaman Update bobot

SKENARIO ALTERNATIF “Cek Total Nilai Bobot kurang dari 100”

Reaksi Aktor

Sistem

5 b. Klik cek total nilai bobot

6 b. Menampilkan AlertMaaf Nilai Anda Kurang dari 100, Nilai Anda Sekrang (Total Nilai)

7 b. Klik Ok

8 b. Menampilkan halaman Update bobot

“Cek Total Nilai Bobot Lebih dari 100”

5 b. Klik cek total nilai bobot

6 b. Menampilkan AlertMaaf Nilai
Anda Melebihi dari 100, Nilai
Anda Sekrang (Total Nilai)

7 b. Klik Ok

8 b. Menampilkan halaman Update
bobot

A.12. Use Case Scenario CRUD Perhitungan SMART

ID	US-12
Name	Perhitungan SMART
Participating Actor	OSIS
Entry Condition	Aktor akan Simpan, Update, Delete Perhitungan Smart Siswa Asuh Sebaya
Exit Condition	Simpan, Update, Delete Perhitungan Smart Siswa Asuh Sebaya

**SKENARIO UTAMA
“A Perhitungan Smart (Simpan)”**

Aksi Aktor

Reaksi Sistem

1. Buka halaman sistem informasi
siswa asuh sebaya

2. Klik menu Perhirungan SMART

3. Menampilkan halaman
Perhitungan SMART



	4. Menampilkan Nilai Bobot dan Normalisasi Bobot pada Form
5. Mengisi Tahun Periode sekarang	
6. Memilih Nama Siswa	
	7. Menampilkan NISN pada Form NISN Siswa
8. Memilih Atribut Total Pendapatan	
	9. Menampilkan Nilai Utility dari Pilihan bobot dan menampilkan Nilai Sementara pada Form
10. Memilih Atribut Jumlah Saudara	
	11. Menampilkan Nilai Utility dari Pilihan bobot dan menampilkan Nilai Sementara pada Form
12. Memilih Atribut Status Anak	
	13. Menampilkan Nilai Utility dari Pilihan bobot dan menampilkan Nilai Sementara pada Form
14. Memilih Atribut Uang Jajan Sekolah	
	15. Menampilkan Nilai Utility dari Pilihan bobot dan menampilkan Nilai Sementara pada Form
16. Memilih Atribut Kendaraan Sekolah	
	17. Menampilkan Nilai Utility dari

	Pilihan bobot dan menampilkan Nilai Sementara pada Form
18. Memilih Atribut Peralatan Sekolah	
	19. Menampilkan Nilai Utility dari Pilihan bobot dan menampilkan Nilai Sementara pada Form
	20. Menampilkan Hasil akhir pada Form Hasil Akhir
21. Klik Simpan	
	22. Menyimpan dan menampilkan data ID SAS, NIS. Tahun, Nama, TP_Atrib, JS_Atrib, SA_Atrib, UJ_Atrib, KT_Atrib, PS_Atrib, RK_Atrib, dan Hasil Akhir pada Tabel
“B Perhitungan SMART (Update)”	
5. Klik salah satu data siswa yang akan diupdate pada tabel	
	6. Menampilkan data ID SAS, NIS. Tahun, Nama, TP_Atrib, JS_Atrib, SA_Atrib, UJ_Atrib, KT_Atrib, PS_Atrib, RK_Atrib, dan Hasil Akhir pada form
7. Merubah data yang akan diganti	
8. Klik Button Update	
	9. Menyimpan dan menampilkan

data ID SAS, NIS. Tahun, Nama, TP_Atrib, JS_Atrib, SA_Atrib, UJ_Atrib, KT_Atrib, PS_Atrib, RK_Atrib, dan Hasil Akhir pada Tabel

“C Perhitungan SMART (Delete)”

5. Klik salah satu data siswa yang akan diupdate pada tabel

6. Menampilkan data ID SAS, NIS. Tahun, Nama, TP_Atrib, JS_Atrib, SA_Atrib, UJ_Atrib, KT_Atrib, PS_Atrib, RK_Atrib, dan Hasil Akhir pada form

7. Klik Button Delete

8. Menghapus data pada sistem

SKENARIO ALTERNATIF “A Perhitungan Smart Data Kosong”

Reaksi Aktor	Sistem
21 a. Klik Simpan	22 a. Menampilkan Alert Data Tahun belum diisi atau ada data form yang belum terisi
	23 a. Menmpilkan Alert Nilai Akhir belum tercapai, pilih semua kriteria yang ada
	24 a. Menampilkan Alert Anda Belum Memilih Calon Nama

Siswa SAS

24 a. klik Ok

A.13. Use Case Scenario VEW Penerima SAS

ID	US-13
Name	Penerima SAS
Participating Actor	OSIS dan guru
Entry Condition	Aktor akan Melihat Penerima dan Mencari siswa Penerima
Exit Condition	Aktor telah Penerima dan Mencari siswa Penerima

SKENARIO UTAMA “A Melihat Penerima SAS”

Aksi Aktor	Reaksi Sistem
1. Buka halaman sistem informasi siswa asuh sebaya	
2. Klik menu Penerima SAS	
	3. Menampilkan halaman Penerima SAS
4. Pilih Tahun Periode	
5. Klik Cari	
	6. Menampilkan sepuluh penerima SAS pada Periode yang dipilih

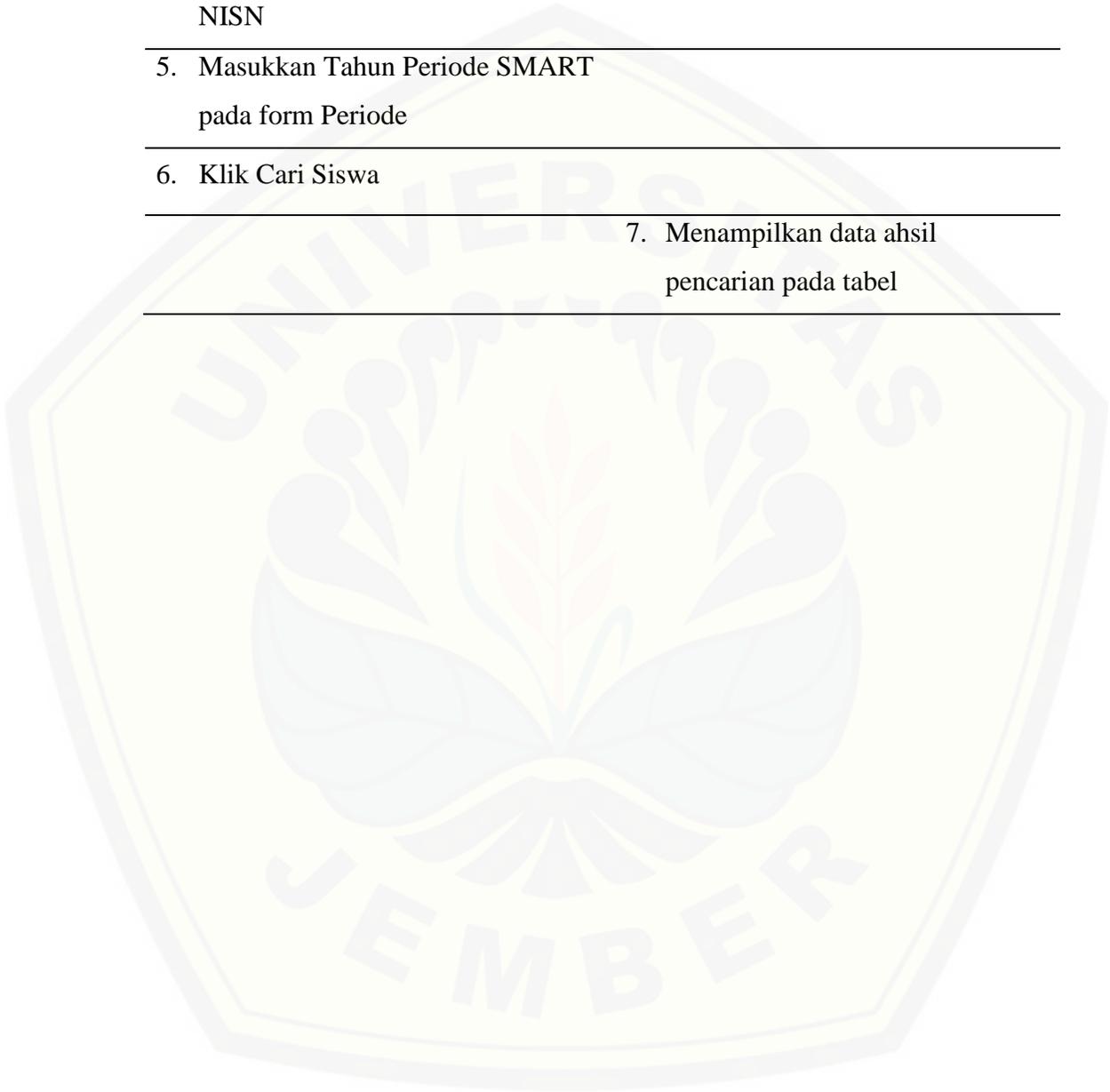
“B Mencari Siswa Penerima SAS”

4. Masukkan NISN siswa pada form
NISN

5. Masukkan Tahun Periode SMART
pada form Periode

6. Klik Cari Siswa

7. Menampilkan data hasil
pencarian pada tabel



F. Lampiran *Black Box Testing*

NO	Menu	Fungsi	Kasus	Hasil	Keterangan
1	Input Siswa Asuh Sebaya (SAS)	Menu ini digunakan oleh user guru yang memiliki fungsi untuk Insert Update, Delete serta untuk melihat data calon siswa asuh sebaya	Ketika user masuk pada menu Input Siswa asuh sebaya	Menampilkan data calon siswa asuh sebaya yang sebelumnya pada tabel	Berhasil
			Ketika user memilih atau klik data siswa pada tabel	Menampilkan data siswa pada form sistem	Berhasil
			Ketika user memasukkan nis, no tlp, thn ajaran dengan huruf	Sistem memblok inputan huruf pada form nis, no tlp, thn ajaran	Berhasil

		Ketika user memasukkan nama dengan angka	Sistem memblok inputan angka pada form nama	Berhasil
		Krtika user menekan button simpan	Sistem menyimpan data dan menampilkannya pada tabel	Berhasil
		Ketika user menekan button update	Sitem mengupdate perubahan pada data dan menyimpan serta menampilkan pada tabel	Berhasil
		Ketika user menekan button delete	Sistem akan menghapus data yang di pilih oleh user	Berhasil
2	Penerima SAS	Menu ini dapat digunakan oleh user guru dan OSIS yang memiliki Ketika user memilih tahun periode SAS	Sistem menampilkan pilihan tahun periode sesuai dengan data yang ada	Berhasil

<p>fungsi untuk melihat siswa yang mendapat SAS dan mencari nama siswa yang tidak diterima</p>	
<p>Ketika user nenikih tahun periode dan menekan button cari</p>	<p>Sistem menampilkan sepuluh siswa yang direkomendasikan mendapatkan bantuan siswa asuh sebaya sesuai peperiode yang dipilih dan ditampilkan pada tabel</p> <p>Berhasil</p>
<p>Ketika User Memasukkan NISN dan Periode sesuai dan menekan button Cari Siswa</p>	<p>Sistem menampilkan siswa dengan nispn dan periode sesuai dengan yang dicari pada form</p> <p>Berhasil</p>

			dan ditampilkan pada tabel		
			Sistem tidak menampilkan data yang di cari dan menampilkan data kosong pada tabel	Berhasil	
3	Update Bobot	Menu ini hanya diakses oleh user OSIS yang berfungsi untuk update dan mengecek total nilai bobot	Ketika user masuk menu Update Bobot	Sistem akan menampilkan data Kriteria dan nilai bobot kriteria yang ada pada tabel	Berhasil
			Ketika user memilih atau mengklik salah satu kriteria pada tabel	Sistem menampilkan data yang dipilih oleh user pada form	Berhasil
			Ketika user klik button cek total	Sistem menampilkan	Berhasil

	nilai	alert nilai anda adalah : 100 serta Alert nilai kurang dan nilai lebih	
	Ketika user klik button update	Sistem menyimpan perubahan data dan menampilkan pada tabel	Berhasil
	Ketika form nilai bobot diisi dengan huruf	Sistem memblok huruf pada innputan form dan todak menampilkan huruf	Berhasil
4	Kriteria Pendapatan Orang Tua	Menu ini hanya diakses oleh user OSIS yang berfungsi untuk insert, update, detele atribut pada kriteria Ketika user masuk menu Pendapatan Orang Tua	Sistem akan menampilkan data atribut dan nilai atribut yang telah ada Berhasil

Pendapatan Orangtua		
Ketika user memilih atribut / mengklik atribut pada tabel	Sistem akan menampilkan data yang sesuai yang dipilih oleh user pada form	Berhasil
Ketika user mengisi nilai atribut dengan huruf	Sistem memblokir huruf pada form nilai atribut	Berhasil
Ketika user menekan button Insert	Sistem menyimpan data yang baru dan menampilkan pada tabel	Berhasil
Ketika user menekan button Update	Sistem mengupdate perubahan data dan menyimpannya serta menampilkan pada tabel	Berhasil
Ketika user klik button Delete	Sistem menghapus data yang telah dipilih oleh	Berhasil

		user			
5	Kriteria Jumlah saudara	Menu ini hanya diakses oleh user OSIS yang berfungsi untuk insert, update, delete atribut pada kriteria Jumlah Saudara	Ketika user masuk menu Jumlah Saudara	Sistem akan menampilkan data atribut dan nilai atribut yang telah ada	Berhasil
			Ketika user memilih atribut / mengklik atribut pada tabel	Sistem akan menampilkan data yang sesuai yang dipilih oleh user pada form	Berhasil
			Ketika user mengisi nilai atribut dengan huruf	Sistem memblokir huruf pada form nilai atribut	Berhasil
			Ketika user menekan button Insert	Sistem menyimpan data yang baru dan menampilkan pada	Berhasil

			tabel		
		Ketika user menekan button Update	Sistem mengupdate perubahan data dan menyimpannya serta menampilkan pada tabel	Berhasil	
		Ketika user klik button Delete	Sistem menghapus data yang telah dipilih oleh user	Berhasil	
6	Kriteria Status Anak	Menu ini hanya diakses oleh user OSIS yang berfungsi untuk insert, update, delete atribut pada kriteria Status Anak	Ketika user masuk menu Status Anak	Sistem akan menampilkan data atribut dan nilai atribut yang telah ada	Berhasil
		Ketika user memilih atribut /	Sistem akan	Berhasil	

			mengklik atribut pada tabel	menampilkan data yang sesuai yang dipilih oleh user pada form	
			Ketika user mengisi nilai atribut dengan huruf	Sistem memblokir huuf pada form nilai atribut	Berhasil
			Ketika user menekan button Insert	Sistem menyimpan data yang baru dan menampilkan pada tabel	Berhasil
			Ketika user menekan button Update	Sistem mengupdate perubahan data dan menyimpannya serta menampilkan pada tabel	Berhasil
			Ketika user klik button Delete	Sistem menghapus data yang telah dipilih oleh user	Berhasil
7	Kriteria Uang Jajan	Menu ini hanya diakses	Ketika user masuk menu Uang Jajan	Sistem akan menampilkan data	Berhasil

oleh user OSIS yang berfungsi untuk insert, update, delete atribut pada kriteria Uang Jajan	atribut dan nilai atribut yang telah ada	
Ketika user memilih atribut / mengklik atribut pada tabel	Sistem akan menampilkan data yang sesuai yang dipilih oleh user pada form	Berhasil
Ketika user mengisi nilai atribut dengan huruf	Sistem memblokir huruf pada form nilai atribut	Berhasil
Ketika user menekan button Insert	Sistem menyimpan data yang baru dan menampilkan pada tabel	Berhasil
Ketika user menekan button Update	Sistem mengupdate perubahan data dan	Berhasil

			menyimpannya serta menampilkan pada tabel		
		Ketika user klik button Delete	Sistem menghapus data yang telah dipilih oleh user	Berhasil	
8	Kriteria Kendaraan	Menu ini hanya diakses oleh user OSIS yang berfungsi untuk insert, update, delete atribut pada kriteria Kendaraan	Ketika user masuk menu Kendaraan	Sistem akan menampilkan data atribut dan nilai atribut yang telah ada	Berhasil
		Ketika user memilih atribut / mengklik atribut pada tabel	Sistem akan menampilkan data yang sesuai yang dipilih oleh user pada form	Berhasil	

		Ketika user mengisi nilai atribut dengan huruf	Sistem memblokir huruf pada form nilai atribut	Berhasil	
		Ketika user menekan button Insert	Sistem menyimpan data yang baru dan menampilkan pada tabel	Berhasil	
		Ketika user menekan button Update	Sistem mengupdate perubahan data dan menyimpannya serta menampilkan pada tabel	Berhasil	
		Ketika user klik button Delete	Sistem menghapus data yang telah dipilih oleh user	Berhasil	
9	Kriteria Peralatan sekolah	Menu ini hanya diakses oleh user OSIS yang berfungsi untuk insert,	Ketika user masuk menu Peralatan sekolah	Sistem akan menampilkan data atribut dan nilai atribut yang telah ada	Berhasil

update, delete atribut pada kriteria Peralatan Sekolah			
	Ketika user memilih atribut / mengklik atribut pada tabel	Sistem akan menampilkan data yang sesuai yang dipilih oleh user pada form	Berhasil
	Ketika user mengisi nilai atribut dengan huruf	Sistem memblokir huruf pada form nilai atribut	Berhasil
	Ketika user menekan button Insert	Sistem menyimpan data yang baru dan menampilkan pada tabel	Berhasil
	Ketika user menekan button Update	Sistem mengupdate perubahan data dan menyimpannya serta menampilkan pada	Berhasil

			tabel	
		Ketika user klik button Delete	Sistem menghapus data yang telah dipilih oleh user	Berhasil
10	Kriteria Rangking Kelas	Menu ini hanya diakses oleh user OSIS yang berfungsi untuk insert, update, detele atribut pada kriteria Rangking Kelas	Ketika user masuk menu Rangking Kelas Sistem akan menampilkan data atribut dan nilai atribut yang telah ada	Berhasil
		Ketika user memilih atribut / mengklik atribut pada tabel	Sistem akan menampilkan data yang sesuai yang dipilih oleh user pada form	Berhasil
		Ketika user mengisi nilai atribut	Sistem memblok huuf	Berhasil

		dengan huruf	pada form nilai atribut		
		Ketika user menekan button Insert	Sistem menyimpan data yang baru dan menampilkan pada tabel	Berhasil	
		Ketika user menekan button Update	Sistem mengupdate perubahan data dan menyimpannya serta menampilkan pada tabel	Berhasil	
		Ketika user klik button Delete	Sistem menghapus data yang telah dipilih oleh user	Berhasil	
11	Perhitungan SMART	Menu ini hanya dapat diakses oleh user OSIS yang berfungsi untuk	Ketika user masuk menu Perhitungan SMART	Sistem akan menampilkan halaman perhitungan SMART dan sistem menampilkan bobot dan normalisasi bobot pada	Berhasil

menjalankan metode SMART dan mampu melakukan Insert Update Delete	form yang ada	
	Sistem akan menampilkan data siswa asuh sebaya yang telah dirangking pada	Berhasil
Ketika user mengisi tahun dengan huruf	Sistem memblok form jika di isi dengan huruf, harus angka	Berhasil
Ketika user memilih nama siswa	Sistem menampilkan nama calon siswa yang sudah terdaftar dari guru	Berhasil
Ketika user memilih drop down	Sistem menampilkan	Berhasil

total pendapatan	atribut dari total pendapatan yang sudah di buat pada menu pendapatan	
Ketika user memilih drop down Jumlah Saudara	Sistem menampilkan atribut dari Jumlah saudara yang sudah di buat pada menu jumlah saudara	Berhasil
Ketika user memilih drop down status anak	Sistem menampilkan atribut dari Status anak yang sudah di buat pada menu Status anak	Berhasil
Ketika user memilih drop down Uang jajan sekolah	Sistem menampilkan atribut dari Uang Jajan sekolah yang sudah di buat pada menu Uang Jajan Sekolah	Berhasil
Ketika user memilih drop down	Sistem menampilkan	Berhasil

Kendaraan Sekolah	atribut dari Kendaraan Sekolah yang sudah di buat pada menu Kendaraan sekolah	
Ketika user memilih drop down Peralatan sekolah	Sistem menampilkan atribut dari Peralatan Sekolah yang sudah di buat pada menu Peralatan sekolah	Berhasil
Ketika user memilih drop down Rangkaing Kelas	Sistem menampilkan atribut dari Rangkaing Kelas yang sudah di buat pada menu Rangkaing Kelas	Berhasil
Ketika user sudah memilih semua atribut, maka hasil akhir akan keluar	Sistem menampilkan nilai akhir pada form nilai akhir	Berhasil
Ketika user tidak mengisi tanggal, tidak memilih, atribut	Sistem menampilkan alert form yang belum	Berhasil

	terisi	
Ketika user menekan button simpan	Sistem menyimpan data dan menampilkan data pada tabel	Berhasil
Ketika user menekan button update	Sistem menyimpan perubahan data dan menampilkan data pada tabel	Berhasil
Ketika user menekan button delete	Sistem menghapus data yang dipilih oleh user	Berhasil
Ketika user memilih / mengklik data pada tabel	Sistem menampilkan data pada form	Berhasil
Ketika user tidak mengisi data atribut secara lengkap	Sistem tidak akan menampilkan hasil akhir pada form hasil akhir	Berhasil

