



**RANCANG BANGUN SISTEM PENDATAAN DAN SELEKSI CALON
PESERTA OLIMPIADE SAINS DENGAN MENGGUNAKAN METODE
PROFILE MATCHING (STUDI KASUS: SMAN 1 GEDANGAN,
SIDOARJO)**

SKRIPSI

Oleh

Emha Dinar Samudro

NIM 122410101063

FAKULTAS ILMU KOMPUTER

UNIVERSITAS JEMBER

2018



**RANCANG BANGUN SISTEM PENDATAAN DAN SELEKSI CALON
PESERTA OLIMPIADE SAINS DENGAN MENGGUNAKAN METODE
PROFILE MATCHING (STUDI KASUS: SMAN 1 GEDANGAN,
SIDOARJO)**

SKRIPSI

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

Oleh

Emha Dinar Samudro

NIM 122410101063

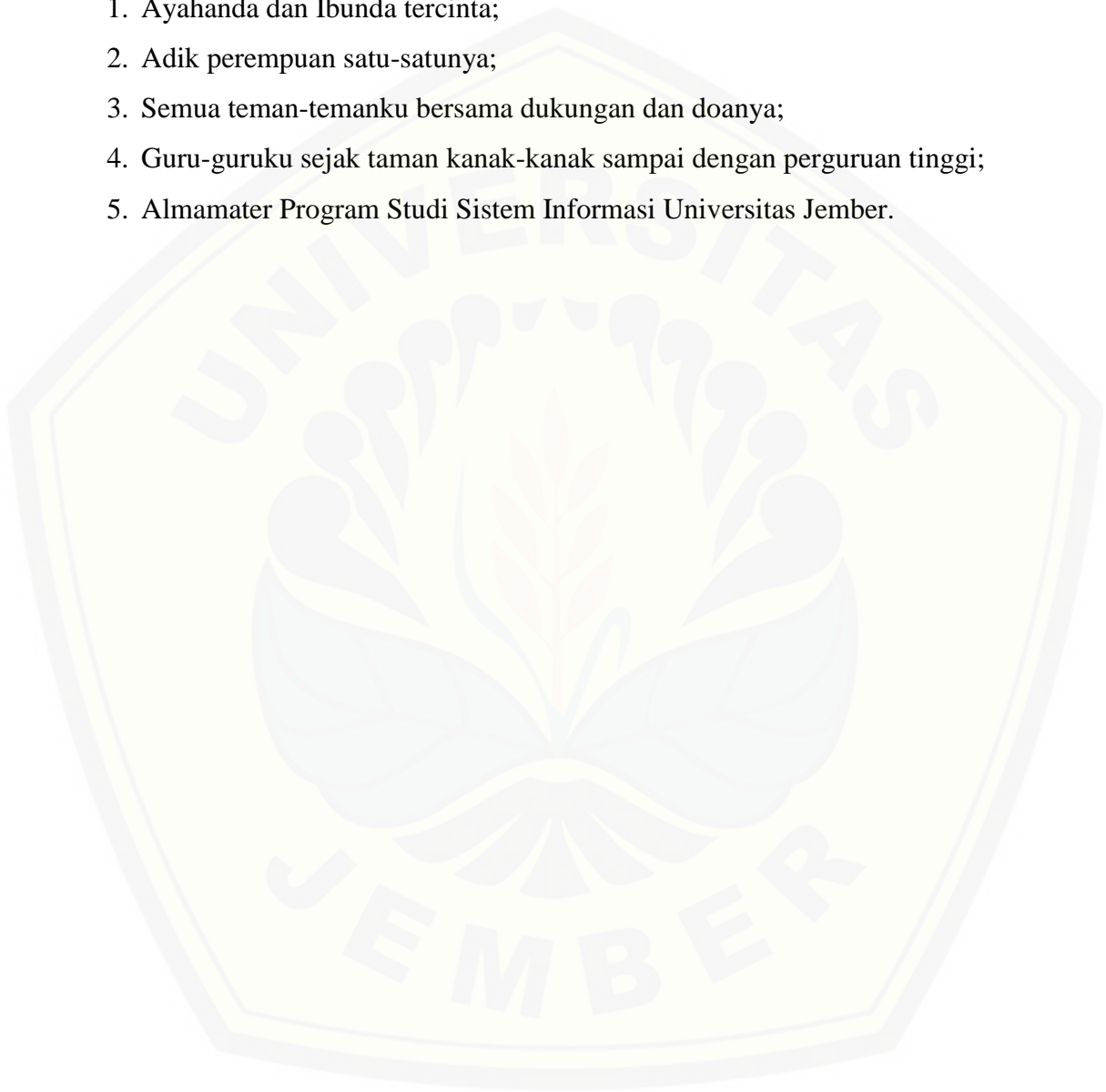
**FAKULTAS ILMU KOMPUTER
UNIVERSITAS JEMBER**

2018

PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

1. Ayahanda dan Ibunda tercinta;
2. Adik perempuan satu-satunya;
3. Semua teman-temanku bersama dukungan dan doanya;
4. Guru-guruku sejak taman kanak-kanak sampai dengan perguruan tinggi;
5. Almamater Program Studi Sistem Informasi Universitas Jember.



MOTTO

“If you born poor, it’s not your mistake. If you die poor, it’s your mistake.”

(William Henry Gates III)



PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Emha Dinar Samudro

NIM : 122410101063

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “RANCANG BANGUN SISTEM PENDATAAN DAN SELEKSI CALON PESERTA OLIMPIADE SAINS DENGAN MENGGUNAKAN METODE *PROFILE MATCHING* (STUDI KASUS: SMAN 1 GEDANGAN, SIDOARJO)”, adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya, belum pernah diajukan pada institusi 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 adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika di kemudian hari pernyataan ini tidak benar.

Jember, 27 Desember 2018

Yang menyatakan,

Emha Dinar S.

NIM 122410101063

SKRIPSI

**RANCANG BANGUN SISTEM PENDATAAN DAN SELEKSI CALON
PESERTA OLIMPIADE SAINS DENGAN MENGGUNAKAN METODE
PROFILE MATCHING (STUDI KASUS: SMAN 1 GEDANGAN,
SIDOARJO)**

Oleh

Emha Dinar Samudro

NIM 122410101063

Pembimbing :

Dosen Pembimbing Utama : Yanuar Nurdiansyah, ST, M.Cs

Dosen Pembimbing Pemdamping : M. Arief Hidayat, S. Kom, M. Kom.

PENGESAHAN PEMBIMBING

Skripsi berjudul “Rancang Bangun Sistem Pendataan Dan Seleksi Calon Peserta Olimpiade Sains Dengan Menggunakan Metode *Profile Matching* (Studi Kasus: SMAN 1 Gedangan, Sidoarjo)”, telah diuji dan disahkan pada:

hari, tanggal : Jum’at, 21 Desember 2018

tempat : Fakultas Ilmu Komputer Universitas Jember

Disetujui oleh:

Pembimbing I,

Pembimbing II,

Yanuar Nurdiansyah, ST, M.Cs

NIP 19820101 2010121004

M. Arief Hidayat, S. Kom, M. Kom.

NIP 198101232010121003

PENGESAHAN PENGUJI

Skripsi berjudul “Rancang Bangun Sistem Pendataan Dan Seleksi Calon Peserta Olimpiade Sains Dengan Menggunakan Metode *Profile Matching* (Studi Kasus: SMAN 1 Gedangan, Sidoarjo)”, telah diuji dan disahkan pada:

hari, tanggal : Jum’at, 21 Desember 2018

tempat : Fakultas Ilmu Komputer Universitas Jember

Tim penguji:

Penguji I,

Penguji II,

Anang Andrianto, S.T., M.T.

NIP. 196906151997021002

Priza Pandunata, S.Kom., M.Sc

NIP. 19830131201504001

Mengesahkan

Ketua Program Studi,

Prof. Dr. Saiful Bukhori, ST., M.Kom

NIP 196811131994121001

RINGKASAN

Rancang Bangun Sistem Pendataan Dan Seleksi Calon Peserta Olimpiade Sains Dengan Menggunakan Metode *Profile Matching* (Studi Kasus: SMAN 1 Gedangan, Sidoarjo)

Sistem pendataan dan seleksi calon peserta olimpiade sains bertujuan untuk membantu memudahkan pihak sekolah khususnya di SMAN 1 Gedangan, Sidoarjo untuk mendata dan menyaring calon siswa-siswi SMAN 1 Gedangan yang akan maju menjadi peserta OSN tingkat kabupaten dan sekaligus mewakili nama SMAN 1 Gedangan Sidoarjo dengan menerapkan Sistem Pendukung Keputusan yakni metode *Profile Matching*. Biasanya dalam pemilihan calon peserta OSN di SMAN 1 Gedangan, seleksi yang dilakukan adalah menggunakan cara konvensional yakni dengan tidak adanya data tertulis yang dipegang oleh wali kelas saat atau setelah penyeleksian dan juga perhitungan sistematis terhadap setiap nilai siswa yang seharusnya menjadi bobot-bobot perhitungan tersendiri sering kali terabaikan. Dan seiring minimnya prestasi pada bidang akademik terutama olimpiade sains, hal ini tentu saja dapat diantisipasi dengan cara melakukan seleksi ketat terhadap siswa calon peserta olimpiade demi mendapatkan kandidat yang terbaik. Fokus dari sistem ini adalah seleksi siswa calon peserta OSN berdasarkan berbagai spesifikasi siswa dan masing-masing bobotnya. Adapun parameter penilaian yang dipakai adalah diambil dari nilai rapor masing-masing calon peserta. Digunakannya metode *Profile Matching* diharapkan mampu memberikan saran keputusan yang lebih tepat, cepat, dan akurat. Uji coba keberhasilan sistem yang kemudian akan diukur dengan membandingkan kecocokan hasil rekomendasi sistem terhadap siswa yang berhasil lulus tes pada seleksi tahap pertama yakni tingkat kabupaten.

PRAKATA

Puji syukur kehadiran Allah SWT atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul “Rancang Bangun Sistem Pendataan Dan Seleksi Calon Peserta Olimpiade Sains Dengan Menggunakan Metode *Profile Matching* (Studi Kasus: SMAN 1 Gedangan, Sidoarjo)”. Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan Strata Satu (S1) pada Program Studi Sistem Informasi Universitas Jember.

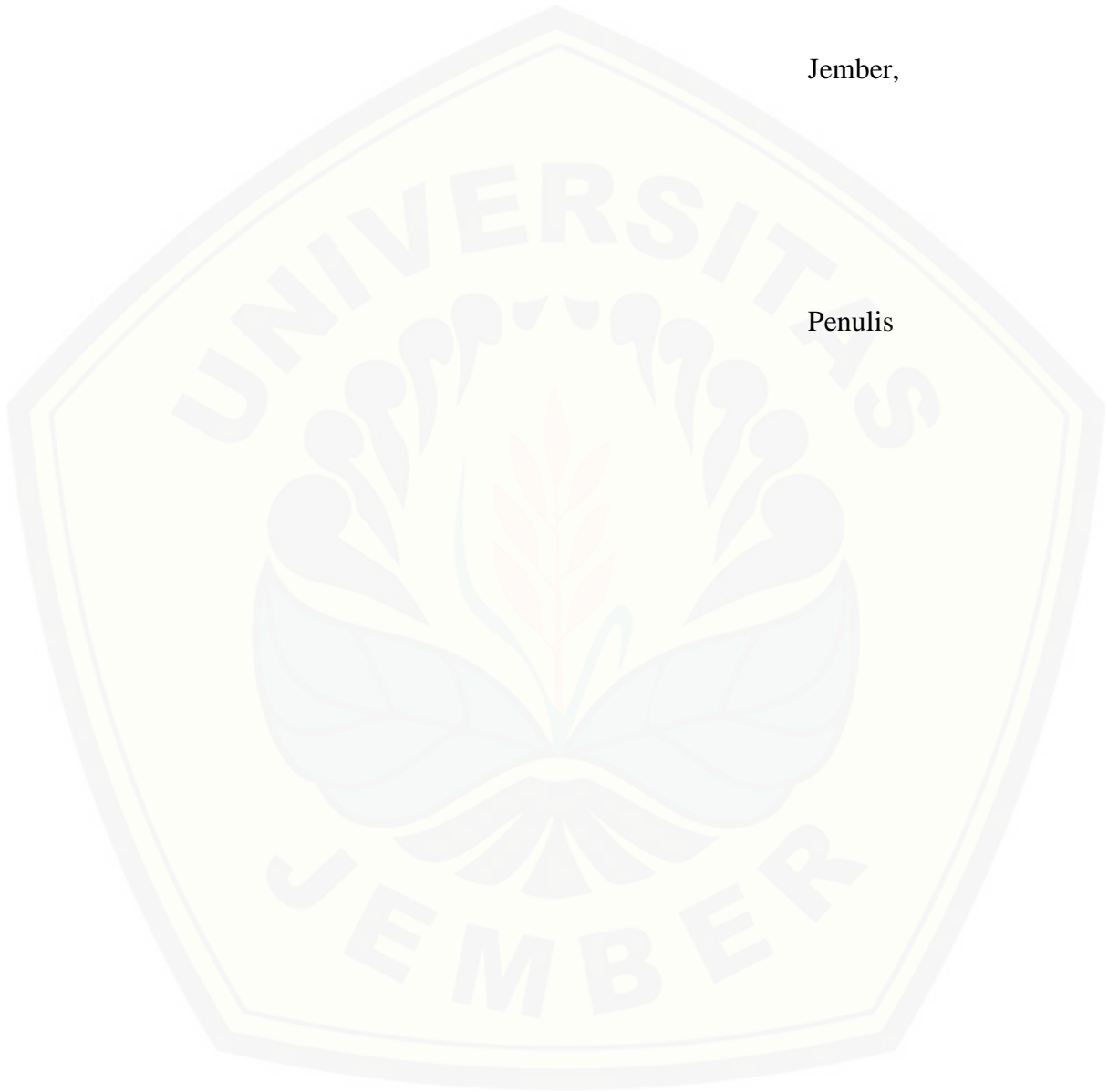
Penyusunan skripsi ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Prof. Drs. Slamir, M.Comp.Sc., Ph.D., selaku Ketua Program Studi Sistem Informasi Universitas Jember;
2. Yanuar Nurdiansyah, ST, M.Cs, selaku Dosen Pembimbing Utama dan M. Arief Hidayat, S. Kom, M. Kom., selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi;
3. Nelly Oktavia A, S.Si, MT., sebagai dosen pembimbing akademik, yang telah mendampingi penulis sebagai mahasiswa.
4. Seluruh Bapak dan Ibu dosen beserta staf karyawan di Program Studi Sistem Informasi Fakultas Ilmu Komputer Universitas Jember;
5. Ayahanda dan Ibunda yang selalu mendukung dan mendoakan;
6. Adik
7. Keluarga Besar
8. Sahabat
9. Teman
10. Laptop “ASUS N46VM” Saya yang senantiasa fasilitasi penulis dari hari ke hari
11. Semua pihak yang tidak dapat disebutkan satu-persatu.

Penulis menyadari bahwa laporan ini masih jauh dari sempurna, oleh sebab itu penulis mengharapkan adanya masukan yang bersifat membangun dari semua pihak. Penulis berharap skripsi ini dapat bermanfaat bagi semua pihak.

Jember,

Penulis



DAFTAR ISI

	Halaman
SKRIPSI.....	i
SKRIPSI.....	ii
PERSEMBAHAN.....	iii
MOTTO	iv
PERNYATAAN.....	v
SKRIPSI.....	vi
PENGESAHAN PEMBIMBING.....	vii
PENGESAHAN PENGUJI.....	viii
RINGKASAN	ix
PRAKATA.....	x
DAFTAR ISI.....	xii
DAFTAR TABEL.....	xvii
DAFTAR GAMBAR	xix
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan dan Manfaat.....	3
1.3.1 Tujuan	3
1.3.2 Manfaat	3
1.4 Batasan Masalah.....	4
1.5 Sistematika Penulisan.....	4
BAB 2. TINJAUAN PUSTAKA	6

2.1	Penelitian Terdahulu.....	6
2.2	Olimpiade Sains Nasional	8
2.3	Sistem Pendukung Keputusan	8
2.4	Metode <i>Profile Matching</i>	9
BAB 3.	METODOLOGI PENELITIAN	12
3.1.	Jenis Penelitian	12
3.2.	Tempat dan Waktu Penelitian	12
3.3.	Tahapan Penelitian	12
3.3.1	Tahapan Analisis Kebutuhan	13
3.3.2.	Tahapan Desain Sistem	14
3.3.3.	Tahapan Implementasi Sistem	15
3.3.4.	Tahapan Pengujian Sistem	15
3.3.5.	Tahapan Pemeliharaan Sistem	16
BAB 4.	PENGEMBANGAN SISTEM.....	17
4.1.	Analisis Kebutuhan Sistem	17
4.1.1.	SOP (<i>Statement of Purpose</i>)	18
4.1.2.	Kebutuhan Fungsional	18
4.1.3.	Kebutuhan Non-Fungsional	19
4.1.4.	Gambaran Umum Sistem	19
4.2.	Desain Sistem	20
4.2.1.	Elemen Data <i>Input</i> dan Data <i>Output</i> Sistem	21
4.2.2.	<i>Use Case Diagram</i>	22
4.2.3.	<i>Use Case Scenario</i>	25
4.2.4	<i>Activity Diagram</i>	29
4.2.5	<i>Sequence Diagram</i>	31

4.2.6.	<i>Entity Relationship Diagram</i>	34
4.3.	Pengkodean Sistem.....	36
4.4.	Pengujian Sistem	40
4.4.1	Pengujian <i>White Box</i>	40
4.4.2	Pengujian <i>Black Box</i>	54
BAB 5. HASIL DAN PEMBAHASAN.....		64
5.1.	Hasil Implementasi <i>Coding</i> Pada Sistem	64
5.1.1.	Tampilan Halaman <i>Sign In</i> dan <i>Home</i>	64
5.1.2.	Tampilan Menu Ubah <i>Password</i>	67
5.1.3.	Tampilan Menu Kuota	69
5.1.4.	Tampilan Menu Data Siswa	71
5.1.5.	Tampilan Menu Kriteria.....	73
5.1.6.	Tampilan Menu Subkriteria	74
5.1.7.	Tampilan Menu Bobot <i>gap</i>	77
5.1.8.	Tampilan Menu Data Nilai Siswa	79
5.1.9.	Tampilan Menu Hasil Perhitungan	80
5.1.10.	Tampilan Menu Hak Akses Pengguna	82
5.2.	Hasil Implementasi Metode <i>Profile Matching</i>	84
5.2.1	Perhitungan Manual Metode <i>Profile Matching</i>	84
5.2.2	Perhitungan Metode <i>Profile Matching</i> Menggunakan Sistem	97
5.3.	Perbandingan Hasil Pengambilan Keputusan Konvensional dan Sistem... 110	
5.4.	Pembahasan Sistem.....	117
5.4.1	Kelebihan Sistem	117
5.4.2	Kekurangan Sistem	117
BAB 6. PENUTUP		118

6.1. Kesimpulan.....	118
6.2. Saran.....	118
DAFTAR PUSTAKA	119
LAMPIRAN.....	121
LAMPIRAN A. <i>USE CASE</i> SKENARIO.....	121
A1. Skenario <i>Sign In</i>	121
A2. Skenario Ubah <i>Password</i>	122
A3. Skenario Input Kuota Lomba	124
A4. Skenario Pengelolaan Data Siswa	126
A5. Skenario Pengelolaan Data Nilai Siswa.....	129
A6. Skenario Pengelolaan Data Kriteria	130
A7. Skenario Pengelolaan Data Subkriteria.....	133
A8. Skenario Pengelolaan Bobot <i>gap</i>	136
A9. Skenario Pengelolaan Data Pengguna.....	139
A10. Skenario <i>View</i> Hasil Perhitungan.....	141
A11. Skenario <i>Sign Out</i>	142
LAMPIRAN B. <i>ACTIVITY DIAGRAM</i>	143
B1. <i>Activity Diagram Sign In</i>	143
B2. <i>Activity Diagram Ubah Password</i>	144
B3. <i>Activity Diagram</i> Input Kuota Lomba.....	145
B4. <i>Activity Diagram</i> Pengelolaan Data Siswa.....	147
B5. <i>Activity Diagram</i> Pengelolaan Data Nilai Siswa.....	150
B6. <i>Activity Diagram</i> Pengelolaan Data Kriteria.....	151
B7. <i>Activity Diagram</i> Pengelolaan Data Subkriteria.....	154
B8. <i>Activity Diagram</i> Pengelolaan Bobot <i>gap</i>	157

B9. <i>Activity Diagram</i> Pengelolaan Data Pengguna.....	160
B10. <i>Activity Diagram</i> View Hasil Perhitungan.....	163
B11. <i>Activity Diagram</i> Sign Out	163
LAMPIRAN C. <i>SEQUENCE DIAGRAM</i>	164
C1. <i>Sequence Diagram</i> Sign In	164
C2. <i>Sequence Diagram</i> Ubah Password	165
C3. <i>Sequence Diagram</i> Input Kuota Lomba	166
C4. <i>Sequence Diagram</i> Pengelolaan Data Siswa.....	168
C5. <i>Sequence Diagram</i> Pengelolaan Data Nilai Siswa.....	171
C6. <i>Sequence Diagram</i> Pengelolaan Data Kriteria.....	172
C7. <i>Sequence Diagram</i> Pengelolaan Data Subkriteria.....	175
C8. <i>Sequence Diagram</i> Pengelolaan Bobot gap	178
C9. <i>Sequence Diagram</i> Pengelolaan Data Pengguna.....	181
C10. <i>Sequence Diagram</i> View Hasil Perhitungan.....	184
C11. <i>Sequence Diagram</i> Sign Out	184
LAMPIRAN D. TRANSKRIP WAWANCARA	185

DAFTAR TABEL

Tabel 2.1. Bobot <i>gap</i>	10
Tabel 4.1. Definisi Aktor.....	23
Tabel 4.2. Definisi <i>Use Case</i>	23
Tabel 4.3. Deskripsi Entitas Pada Sistem.....	35
Tabel 4.4. Kode program pada perhitungan.php untuk melakukan perhitungan metode <i>Profile Matching</i>	36
Tabel 4.5. Listing Program Perhitungan Metode <i>Profile Matching</i>	41
Tabel 4.6. Test Case Pengelolaan Nilai Awal dan Nilai Selisih <i>gap</i>	52
Tabel 4.7. Test Case Pemetaan Nilai Bobot <i>gap</i>	52
Tabel 4.8. Test Case Perhitungan Nilai CF, SF, dan Nilai Total.....	53
Tabel 4.9. Test Case Perankingan Siswa (Nilai Akhir).....	53
Tabel 4.10. Test Case Daftar Siswa Terpilih.....	54
Tabel 4.11. Pengujian Metode <i>Black Box</i>	54
Tabel 5.1. Penentuan Bobot Kriteria.....	84
Tabel 5.2. Penentuan Subkriteria dan Nilai Ideal.....	84
Tabel 5.3. Perolehan Nilai siswa dan pemetaan <i>Gap</i>	85
Tabel 5.4. Daftar bobot <i>gap</i>	86
Tabel 5.5. Pembobotan <i>gap</i>	87
Tabel 5.6. Nilai CF dan SF Nilai Utama.....	88
Tabel 5.7. Nilai CF dan SF Nilai Tambah.....	88
Tabel 5.8. Nilai Total Nilai Utama.....	89
Tabel 5.9. Nilai Total Nilai Tambah.....	90
Tabel 5.10. Penentuan Bobot Kriteria.....	91
Tabel 5.11. Penentuan Subkriteria dan Nilai Ideal.....	91
Tabel 5.12. Perolehan Nilai siswa dan pemetaan <i>Gap</i>	92
Tabel 5.13. Daftar bobot <i>gap</i>	92
Tabel 5.14. Pembobotan <i>gap</i>	93
Tabel 5.15. Nilai CF dan SF Nilai Utama.....	94

Tabel 5.16. Nilai CF dan SF Nilai Tambah.....	95
Tabel 5.17. Nilai Total Nilai Utama.....	96
Tabel 5.18. Nilai Total Nilai Tambah.....	96
Tabel 5.19. Perbandingan Perhitungan Manual dan Sistem SPK OSN TIK.....	109
Tabel 5.20. Perbandingan Perhitungan Manual dan Sistem SPK OSN Fisika.....	109
Tabel 5.21. Kandidat dan Siswa Terpilih OSN TIK Tahun 2016.....	110
Tabel 5.22. Perbandingan Hasil Keputusan OSN TIK 2016.....	110
Tabel 5.23. Kandidat dan Siswa Terpilih OSN TIK Tahun 2017.....	111
Tabel 5.24. Perbandingan Hasil Keputusan OSN TIK 2017.....	111
Tabel 5.25. Kandidat dan Siswa Terpilih OSN TIK Tahun 2018.....	112
Tabel 5.26. Perbandingan Hasil Keputusan OSN TIK 2018.....	112
Tabel 5.27. Kandidat dan Siswa Terpilih OSN Fisika Tahun 2016.....	113
Tabel 5.28. Perbandingan Hasil Keputusan OSN Fisika 2016.....	113
Tabel 5.29. Kandidat dan Siswa Terpilih OSN Fisika Tahun 2017.....	114
Tabel 5.30. Perbandingan Hasil Keputusan OSN Fisika 2017.....	114
Tabel 5.31. Kandidat dan Siswa Terpilih OSN Fisika Tahun 2018.....	115
Tabel 5.32. Perbandingan Hasil Keputusan OSN Fisika 2018.....	115
Tabel 5.33. Uji Coba Keberhasilan Sistem.....	116

DAFTAR GAMBAR

Gambar 2.1. Perbandingan Algoritma Metode SAW dan *Profile Matching*.....7

Gambar 3.1. Waterfall Model (Pressman, 2010).....13

Gambar 4.1. Gambaran Umum Sistem.....20

Gambar 4.2. Elemen Data *Input* dan Data *Output* Sistem.....21

Gambar 4.3. *Use Case Diagram*.....22

Gambar 4.4. *Entity Relationship Diagram* (ERD).....34

Gambar 4.5. *Diagram* Alir olah Nilai Awal dan Nilai selisih *gap*.....49

Gambar 4.6. *Diagram* Alir Pemetaan Nilai Bobot *gap*.....49

Gambar 4.7. *Diagram* Alir Perhitungan Nilai CF, SF, dan Nilai Total.....50

Gambar 4.8. *Diagram* Alir Perankingan Alternatif.....50

Gambar 4.9. *Diagram* Alir Daftar Siswa Terpilih.....51

Gambar 5.1. Halaman *Sign In* SICALIM TIK.....64

Gambar 5.2. Halaman *Sign In* SICALIM Fisika.....65

Gambar 5.3. *Sign In* gagal.....65

Gambar 5.4. Halaman *Home* SICALIM TIK (akses admin).....66

Gambar 5.5. Halaman *Home* SICALIM TIK (akses wali kelas).....67

Gambar 5.6. Tampilan Menu Ubah *Password* SICALIM TIK (akses admin).....68

Gambar 5.7. Tampilan peringatan error Ubah *Password* 1.....68

Gambar 5.8. Tampilan peringatan error Ubah *Password* 2.....69

Gambar 5.9. Tampilan pesan berhasil Ubah *Password*.....69

Gambar 5.10. Tampilan Menu Kuota SICALIM TIK (akses admin).....70

Gambar 5.11. Tampilan sesi “Tambah Data” pada menu Kuota.....70

Gambar 5.12. Tampilan konfirmasi “Hapus Data” pada menu Kuota.....71

Gambar 5.13. Tampilan Menu Data Siswa SICALIM TIK (akses admin).....71

Gambar 5.14. Tampilan sesi “Tambah Data” pada menu Data Siswa.....72

Gambar 5.15. Tampilan sesi “Ubah Data” pada menu Data Siswa.....72

Gambar 5.16. Tampilan konfirmasi “Hapus Data” pada menu Data Siswa.....72

Gambar 5.17. Tampilan Menu Kriteria SICALIM TIK (akses admin).....73

Gambar 5.18. Tampilan sesi “Tambah Data” pada menu Kriteria.....	73
Gambar 5.19. Tampilan sesi “Ubah Data” pada menu Kriteria.....	74
Gambar 5.20. Tampilan konfirmasi “Hapus Data” pada menu Kriteria.....	74
Gambar 5.21. Tampilan Menu Subkriteria SICALIM TIK (akses admin).....	75
Gambar 5.22. Tampilan sesi “Tambah Data” pada menu Subkriteria.....	76
Gambar 5.23. Tampilan sesi “Ubah Data” pada menu Subkriteria.....	76
Gambar 5.24. Tampilan konfirmasi “Hapus Data” pada menu Subkriteria.....	76
Gambar 5.25. Tampilan Menu Bobot <i>gap</i> SICALIM TIK (akses admin).....	77
Gambar 5.26. Tampilan sesi “Tambah Data” pada menu Bobot <i>gap</i>	78
Gambar 5.27. Tampilan sesi “Ubah Data” pada menu Bobot <i>gap</i>	78
Gambar 5.28. Tampilan konfirmasi “Hapus Data” pada menu Bobot <i>gap</i>	78
Gambar 5.29. Tampilan Menu Data Nilai Siswa SICALIM TIK (akses admin)....	79
Gambar 5.30. Tampilan Menu Data Nilai Siswa SICALIM TIK (akses admin) 2..	79
Gambar 5.31. Tampilan awal menu Hasil Perhitungan.....	80
Gambar 5.32. Tabel Nilai Awal dan Nilai Selisih <i>gap</i> pada menu Hasil Perhitungan.....	81
Gambar 5.33. Tabel Pemetaan Nilai Bobot <i>gap</i> pada menu Hasil Perhitungan.....	81
Gambar 5.34. Tabel Nilai CF, SF, dan Nilai Total pada menu Hasil Perhitungan..	81
Gambar 5.35. Tabel Nilai Akhir dan Siswa Terpilih pada menu Hasil Perhitungan.....	82
Gambar 5.36. Tampilan Menu Hak Akses Pengguna SICALIM TIK (akses admin).....	82
Gambar 5.37. Tampilan sesi “Tambah Data” pada menu Hak Akses Pengguna..	83
Gambar 5.38. Tampilan sesi “Ubah Data” pada menu Hak Akses Pengguna.....	83
Gambar 5.39. Tampilan konfirmasi “Hapus Data” pada menu Hak Akses Pengguna.....	83
Gambar 5.40. Tampilan awal menu Kriteria.....	97
Gambar 5.41. Menambahkan Kriteria Penilaian.....	98
Gambar 5.42. Daftar kriteria sesuai hasil analisis.....	98
Gambar 5.43. Tampilan awal menu Subkriteria.....	99
Gambar 5.44. Menambahkan Subkriteria Penilaian.....	99

Gambar 5.45. Daftar subkriteria sesuai hasil analisis.....	100
Gambar 5.46. Tampilan awal menu Bobot <i>Gap</i>	100
Gambar 5.47. Menambahkan Bobot <i>Gap</i> Penilaian.....	101
Gambar 5.48. Daftar bobot <i>gap</i> sesuai hasil analisis.....	101
Gambar 5.49. Daftar bobot <i>gap</i> sesuai hasil analisis (2).....	102
Gambar 5.50. Tampilan awal menu Data Siswa.....	103
Gambar 5.51. Daftar Data Siswa sesuai hasil analisis.....	103
Gambar 5.52. Tampilan Awal Data Nilai Siswa.....	104
Gambar 5.53. Input Data Nilai Siswa per Periode Lomba.....	104
Gambar 5.54. Input Data Nilai Siswa per Periode Lomba (2).....	105
Gambar 5.55. Input Data Nilai Berhasil.....	105
Gambar 5.56. Tampilan Awal Menu Hasil Perhitungan.....	106
Gambar 5.57. Nilai Awal dan Selisih <i>Gap</i>	106
Gambar 5.58. Nilai Bobot <i>Gap</i>	107
Gambar 5.59. Nilai CF dan SF masing-masing Kriteria.....	107
Gambar 5.60. Nilai Akhir SPK OSN TIK.....	108
Gambar 5.61. Nilai Akhir SPK OSN Fisika.....	108

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

Pada era globalisasi ini perkembangan teknologi semakin pesat, sehingga kita harus meningkatkan kualitas sumber daya manusia di berbagai bidang. Kualitas pendidikan adalah salah satu aspek yang dapat mempengaruhi kualitas sumber daya manusia. Pendidikan dengan mutu yang baik sejak dini diperlukan guna mempersiapkan era global. Salah satu cara untuk mengetahui peningkatan mutu pendidikan sejak dini, adalah dengan diadakannya Olimpiade Sains Nasional (OSN).

Mutu sumber daya manusia suatu bangsa tergantung pada mutu pendidikan. Sebagai bagian dari masyarakat internasional, masa depan Indonesia sangat membutuhkan kemampuan kompetitif di kalangan pelajar untuk bersaing secara sehat dalam penguasaan ilmu pengetahuan dan teknologi. Agar dapat mengantisipasi hal tersebut, Departemen Pendidikan Nasional melalui Direktorat Jenderal Pendidikan Dasar dan Menengah telah memfasilitasi kegiatan kegiatan yang mengarah pada kreativitas siswa dalam bidang ilmu pengetahuan dan teknologi yaitu dengan lomba-lomba baik yang berskala nasional seperti Olimpiade Sains Nasional (OSN) maupun tingkat internasional.

SMAN 1 Gedangan merupakan salah satu Sekolah Menengah Atas favorit di kabupaten Sidoarjo yang setiap tahunnya dipilih calon peserta untuk mengikuti OSN tingkat kabupaten. Bidang lomba yang ditawarkan biasanya adalah TIK (Komputer) dan Fisika. Dari masing-masing bidang lomba tersebut, biasanya akan diambil 1 sampai 3 peserta terbaik dari sekian banyak siswa yang mengajukan diri untuk mengikuti seleksi tingkat kabupaten. Demi mewujudkan kualitas dari kandidat terpilih dari SMAN 1 Gedangan, maka perlu diadakan seleksi ketat yang dilakukan oleh pihak sekolah untuk mendapatkan calon peserta yang berkualitas dari masing-masing bidang lomba tersebut.

Biasanya dalam pemilihan calon peserta OSN di SMAN 1 Gedangan, seleksi yang dilakukan adalah menggunakan cara konvensional, yakni siswa yang mengajukan diri akan diseleksi yang sebelumnya telah menentukan posisi yang diinginkan di dalam bidang lomba OSN. Seleksi dilakukan oleh wali kelas. Namun, kekurangan seleksi dengan menggunakan cara ini adalah tidak adanya data tertulis yang dipegang oleh wali kelas. Perhitungan sistematis terhadap setiap nilai siswa yang seharusnya menjadi bobot-bobot nilai tersendiri juga sering kali terabaikan dan cenderung subjektif. Perbandingan siswa antar kelas juga merupakan kendala pada sistem konvensional ini karena untuk membandingkan siswa unggulan dari setiap kelas, otomatis setiap wali kelas harus mengadakan rapat yang mana cara tersebut tidaklah praktis dan membuang waktu.

Dari permasalahan tersebut, maka diambil kesimpulan bahwa seleksi calon peserta olimpiade pada SMAN 1 Gedangan memerlukan sarana atau alat bantu berupa wadah yang dapat menampung data siswa sekaligus mengolahnya menjadi hasil seleksi calon peserta olimpiade sains. Sistem pendukung keputusan seleksi calon peserta olimpiade ini bertujuan untuk membantu wali kelas maupun guru dalam hal melakukan seleksi siswa yang akan menjadi calon peserta OSN. Adapun parameter penilaian yang dapat dimanfaatkan untuk kriteria penilaian adalah nilai rapor siswa terkait dengan bidang lomba masing-masing. Pada tahap pertama, siswa yang mengajukan diri akan didata spesifikasinya. Tahap kedua yakni pemrosesan data yang dilakukan oleh sistem dimana nanti akan dihasilkan *output* berupa hasil *ranking* dari setiap bidang lomba dan terpilihnya kandidat sesuai jumlah kuota per bidang lomba..

1.2 Rumusan Masalah

Berdasarkan dari beberapa permasalahan yang telah diuraikan, maka dapat diambil rumusan masalah sebagai berikut:

1. Bagaimana mengimplementasikan metode *Profile Matching* dalam seleksi siswa calon peserta OSN?
2. Bagaimana membangun sistem penunjang keputusan seleksi siswa calon peserta OSN dengan metode *Profile Matching*?

1.3 Tujuan dan Manfaat

Berikut merupakan tujuan yang ingin dicapai dan manfaat yang ingin diperoleh dalam penelitian ini.

1.3.1 Tujuan

Berdasarkan rumusan masalah yang telah dipaparkan, maka dapat ditetapkan tujuan penelitian, yaitu sebagai berikut:

1. Mengimplementasikan metode *Profile Matching* dalam membantu *me-ranking* dan menyeleksi siswa yang menjadi calon peserta OSN berdasarkan kriteria-kriteria yang sesuai.
2. Membangun sistem penunjang keputusan seleksi calon peserta OSN dengan metode *Profile Matching*.

1.3.2 Manfaat

Berdasarkan rumusan masalah yang telah dipaparkan, maka dapat ditetapkan manfaat dari penelitian ini yaitu:

1. Bagi Akademis

Penelitian yang dilakukan diharapkan memberikan hasil yang mampu memberikan masukan informasi yang terkait dengan judul penelitian kepada pembaca pada umumnya dan pada Program Studi Sistem Informasi maupun semua civitas akademika Fakultas Ilmu Komputer Universitas Jember pada khususnya.

2. Bagi Peneliti

2.1 Mengetahui proses penerapan pengembangan sistem penunjang keputusan seleksi calon peserta OSN di SMAN 1 Gedangan Sidoarjo Menggunakan Metode *Profile Matching*.

2.2 Mengetahui lebih jauh cara kerja metode *Profile Matching* dalam sebuah sistem penunjang keputusan.

3. Bagi Objek Penelitian

Membantu pihak terkait dalam menyeleksi atau menentukan siswa yang terbaik untuk dipilih maju sebagai wakil peserta OSN di SMAN 1 Gedangan Sidoarjo.

1.4 Batasan Masalah

Adapun batasan masalah dalam penelitian ini dapat dijelaskan secara rinci yakni sebagai berikut:

1. Objek yang ditunjukkan dalam implementasi sistem penunjang keputusan ini adalah siswa-siswi SMAN 1 Gedangan Sidoarjo.
2. Berdasarkan yang telah dipaparkan pada latar belakang penelitian, bidang lomba OSN yang dicakup oleh sistem ialah TIK (Komputer) dan Fisika.
3. Sistem penunjang keputusan ini hanya sebagai alat bantu bagi pihak terkait dalam memilih atau menyeleksi siswa terbaik pada kategorinya.
4. Siswa yang ditunjukkan untuk ditangani oleh sistem adalah semua kelas pada kelas X dan XI jurusan IPA.
5. Hasil dari sistem penunjang keputusan ini adalah perankingan yang dihitung berdasarkan setiap kriteria dan bobot dari setiap bidang lomba yang dibutuhkan melalui sistem ini.
6. Sistem penunjang keputusan menggunakan metode *Profile Matching*.
7. Sistem ini berbasis *website*.

1.5 Sistematika Penulisan

Adapun sistematika penulisan skripsi ini adalah sebagai berikut:

1. Pendahuluan

Bab kesatu ini memuat uraian tentang latar belakang, perumusan masalah, tujuan, manfaat, batasan masalah, dan sistematika penulisan skripsi yang masing-masing tertuang secara eksplisit dalam subbab tersendiri.

2. Tinjauan Pustaka

Bab ini memaparkan tinjauan terhadap hasil-hasil penelitian terdahulu berkaitan dengan masalah yang dibahas, landasan materi, dan

kajian teori metode analisis data yang berkaitan dengan masalah dalam penelitian.

3. Metodologi Penelitian

Bab ini menguraikan tentang tempat dan waktu penelitian, metode penelitian, metode pengumpulan data, metode analisis data, dan teknik pengembangan sistem yang digunakan dalam penelitian.

4. Pengembangan Sistem

Bab ini berisi uraian tentang langkah-langkah yang ditempuh dalam proses menganalisis dan merancang sistem yang hendak dibangun meliputi desain, pengkodean, dan pengujian sistem.

5. Hasil dan Pembahasan

Bab ini memaparkan secara rinci pemecahan masalah melalui analisis yang disajikan dalam bentuk deskripsi dibantu dengan ilustrasi berupa tabel dan gambar untuk memperjelas hasil penelitian.

6. Penutup

Bab ini terdiri atas kesimpulan atas penelitian yang telah dilakukan dan saran untuk penelitian selanjutnya.

BAB 2. TINJAUAN PUSTAKA

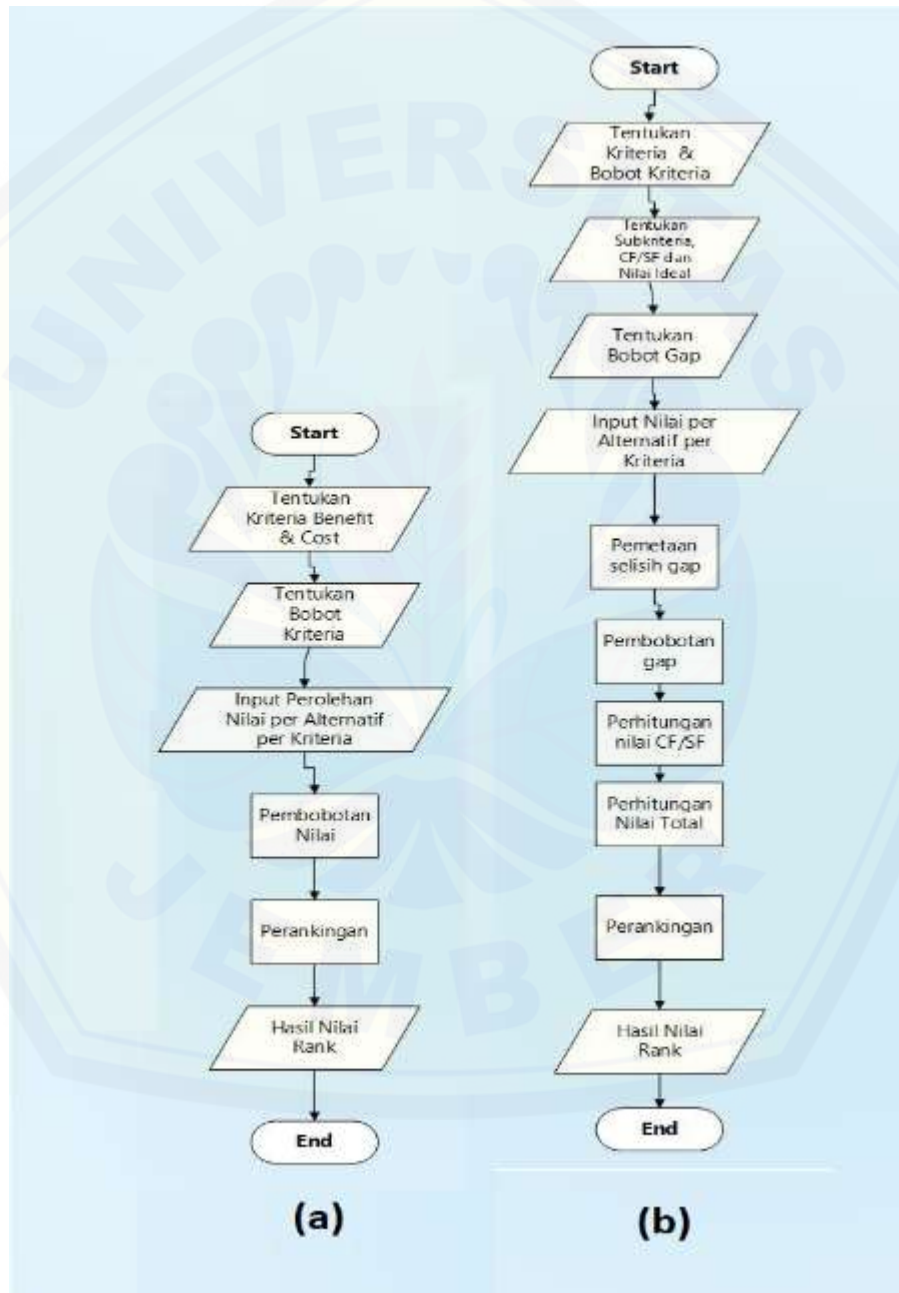
Pada bagian ini dipaparkan tinjauan yang berkaitan dengan masalah yang dibahas, kajian teori yang berkaitan dengan masalah, dan juga penelitian-penelitian terdahulu.

2.1 Penelitian Terdahulu

Penelitian sebelumnya yang berjudul "Perancangan Sistem Pendukung Keputusan Untuk Pemilihan Siswa Peserta Olimpiade Sains Nasional (OSN) Dengan Metode *Simple Additive Weighting* (SAW) Di SMA Negeri 2 Bondowoso" oleh Panji Setio (2013). Penelitian ini menjelaskan bahwa dengan menggunakan metode SAW didapatkan penentuan calon peserta Olimpiade Sains dengan melakukan penilaian pada masing-masing individu dengan cara menentukan kriteria-kriteria yang ada dengan masing-masing bobotnya sebelum dilakukan perankingan. Penunjang keputusan dengan menggunakan Perhitungan metode SAW ini dilakukan dengan beberapa tahap yakni menentukan kriteria cost dan benefit, penentuan bobot kriteria, input perolehan nilai alternatif, pembobotan nilai berdasarkan bobot kriteria, dan perankingan.

Penelitian kedua berjudul "Sistem Pendukung Keputusan Kenaikan Jabatan Menggunakan Metode *Profile Matching* (Studi kasus di PT. Industri Kemasan Semen Gresik)" yang dilakukan oleh Asfan Muqtadir, Irwan Purdianto (2013). Penelitian tersebut berlandaskan pada proses kenaikan jabatan untuk mengisi jabatan yang kosong, yang sering kali mengalami kesulitan karena pengajuan calon kandidat yang bisa menempati jabatan tersebut dengan cara pencocokan profil karyawan dan profil jabatan yang kurang terdefinisi dengan baik dan sering kali terabaikan. Untuk meminimalisir kendala tersebut diperlukan suatu sistem pendukung keputusan yang dapat menganalisa beberapa karyawan yang sesuai dengan profil jabatan yang ada. Hasil dari proses seleksi berupa skor akhir karyawan sebagai rekomendasi bagi pengambil keputusan untuk memilih karyawan yang cocok pada jabatan yang kosong tersebut. Penunjang keputusan dengan menggunakan Perhitungan metode *Profile Matching* ini dilakukan dengan beberapa

tahap yakni penentuan kriteria dan bobot, tentukan kriteria CF & SF beserta nilai ideal, penentuan bobot *gap*, input perolehan nilai alternatif, pemetaan selisih *gap*, pembobotan *gap*, perhitungan nilai CF & SF, perhitungan nilai total, dan yang terakhir yaitu perankingan. Perbandingan alur dari metode SAW dibandingkan dengan metode *Profile Matching* adalah seperti gambar 2.1 berikut ini.



Gambar 2.1. Perbandingan algoritma metode SAW (a) dan *Profile Matching* (b)

Dari perbandingan algoritma di atas, terlihat bahwa metode perhitungan *Profile Matching* memiliki lebih banyak variabel perhitungan, salah satunya adalah dapat menentukan bobot *gap* dari setiap kemungkinan perolehan nilai selisih *gap* yang mana perlakuan khusus pada setiap kemungkinan perolehan nilai ini tidak terdapat pada perhitungan metode SAW.

Metode serupa juga dapat diimplementasikan pada penyeleksian calon peserta OSN pada tingkat sekolah, Adapun kriteria seleksi bersifat dinamis, yaitu ditentukan jumlah, jenis, dan bobotnya oleh pengguna agar sistem yang dibuat dapat beradaptasi seiring pergantian *requirement* lomba pada setiap tahunnya.

2.2 Olimpiade Sains Nasional

Olimpiade Sains Nasional adalah ajang kompetisi dalam bidang sains bagi para siswa pada jenjang SD, SMP, dan SMA di Indonesia yang diselenggarakan oleh Direktorat Jenderal Pendidikan Dasar dan Menengah. Siswa yang mengikuti Olimpiade Sains Nasional adalah siswa yang telah lolos seleksi tingkat kabupaten dan provinsi dan karenanya adalah siswa-siswa terbaik dari propinsinya masing-masing. Olimpiade Sains Nasional diadakan setiap tahun di kota yang berbeda-beda. Kegiatan ini merupakan salah satu bagian dari rangkaian seleksi untuk mendapatkan siswa-siswi terbaik dari seluruh Indonesia yang akan dibimbing lebih lanjut oleh tim bidang kompetisi masing-masing dan akan diikutsertakan pada olimpiade-olimpiade tingkat internasional. (Galuh, 2010).

2.3 Sistem Pendukung Keputusan

Sistem Pendukung Keputusan (SPK) adalah salah satu cara mengorganisir informasi yang melibatkan penggunaan basisdata yang bertujuan untuk digunakan dalam membuat keputusan. SPK dirancang untuk pendekatan dalam menyelesaikan masalah para pembuat keputusan dan kebutuhan aplikasi, namun tidak berfungsi untuk menggantikan keputusan maupun suatu keputusan untuk pengguna sistem itu sendiri (Syaiful Luthvi A., 2016).

Karakteristik utama SPK (Sprague, 1993), yaitu:

1. Sistem berbasis komputer

2. Sistem memudahkan user dalam mengambil keputusan
3. Sistem menggunakan kalkulasi manual untuk memecahkan masalah yang sulit
4. Sistem menggunakan simulasi yang mudah dimengerti
5. Sistem mempunyai data dan model analisis sebagai komponen utama

Sistem Penunjang Keputusan merupakan sistem yang mampu memberikan kemampuan pemecahan masalah maupun kemampuan pengkomunikasian untuk masalah dengan kondisi semi terstruktur dan tak terstruktur. Sistem Penunjang Keputusan bertujuan untuk menyediakan informasi, membimbing, memberikan prediksi, serta mengarahkan kepada pengguna sistem agar dapat melakukan pengambilan keputusan dengan lebih baik di kemudian hari.

Sistem Penunjang Keputusan merupakan implementasi teori-teori pengambilan keputusan yang telah diperkenalkan oleh ilmu-ilmu seperti *operation research* dan *management science*, hanya bedanya adalah bahwa jika dahulu untuk mencari penyelesaian masalah yang dihadapi harus dilakukan perhitungan iterasi secara manual (biasanya untuk mencari nilai minimum-maksimum, rata-rata, dan sebagainya). Namun saat ini, komputer atau PC telah menawarkan kemampuannya untuk menyelesaikan komputasi tersebut dengan cepat dan singkat.

2.4 Metode Profile Matching

Menurut Rachma (2003:101), *Profile Matching* merupakan suatu proses dalam manajemen SDM dimana terlebih dahulu ditentukan kompetensi (kemampuan) yang diperlukan oleh suatu jabatan. Kompetensi/kemampuan tersebut haruslah dapat dipenuhi oleh pemegang jabatan.

Penelitian ini menggunakan *Profile Matching* dengan tahap awal yaitu menentukan *gap* yaitu kriteria awal dalam perhitungan. Contoh pembobotan nilai *gap* dapat dilihat pada Tabel 2.1.

Tabel 2.1. Bobot *gap*

No.	Selisih <i>Gap</i>	Bobot Nilai	Keterangan
1.	0	6	Kompetensi sesuai dengan yang dibutuhkan
2.	1	5,5	Kompetensi individu kelebihan 1 tingkat
3.	-1	5	Kompetensi individu kurang 1 tingkat
4.	2	4,5	Kompetensi individu kelebihan 2 tingkat
5.	-2	4	Kompetensi individu kurang 2 tingkat
6.	3	3,5	Kompetensi individu kelebihan 3 tingkat
7.	-3	3	Kompetensi individu kekurangan 3 tingkat
8.	4	2,5	Kompetensi individu kelebihan 4 tingkat
9.	-4	2	Kompetensi individu kekurangan 4 tingkat
10.	5	1,5	Kompetensi individu kelebihan 5 tingkat
11.	-5	1	Kompetensi individu kekurangan 5 tingkat

Setelah tahap penentuan *gap*, ditentukan perhitungan *Core Factor* (ncf) dan *Secondary Factor* (scf). Perhitungan selanjutnya dipaparkan dengan Persamaan 1 sampai 4.

$$NCI = \sum NC / \sum IC \dots\dots\dots \text{Persamaan (1)}$$

Keterangan :

NCI = Nilai rata-rata *core factor* kriteria

NC = Jumlah nilai *core factor* kriteria

IC = Jumlah item *core factor*

$$NSI = \sum NS / \sum IS \dots\dots\dots \text{Persamaan (2)}$$

Keterangan :

NSI = Nilai rata-rata *secondary factor* kriteria

NS = Jumlah nilai *secondary factor* kriteria

IS = Jumlah item *secondary factor*

Lalu menghitung nilai total tiap kompetensi kriteria

$$N = 60\% NC + 40\% NS \dots\dots\dots \text{Persamaan (3)}$$

Keterangan :

N = Nilai total tiap aspek kriteria

NC = Nilai *core factor*

NS = Nilai *secondary factor*

Perhitungan *ranking*

$Ranking = 60\% \text{ nilai kompetensi 1} + 40\% \text{ nilai kompetensi 2}$

.....*Persamaan (4)*

Menurut Kusri (2007:53) berpendapat bahwa maksud dari pencocokan *Profile Matching* adalah sebuah mekanisme pengambilan keputusan dengan mengasumsikan bahwa terdapat tingkat indikator ideal yang harus dimiliki oleh pelamar, bukannya tingkat minimal yang harus dipenuhi atau dilewati.

Hasil akhir dari proses *Profile Matching* adalah *ranking* dari kandidat. Penentuan *ranking* mengacu pada hasil perhitungan tertentu dengan aspek yang dicontohkan (Kusri, 2007:66).

Pada penelitian tersebut, langkah dalam melakukan metode ini terdiri dari lima langkah. Langkah pertama menentukan Bobot Nilai *gap*. Pada tahap ini, akan ditentukan bobot nilai masing-masing aspek dengan menggunakan bobot nilai yang telah ditentukan bagi masing-masing aspek itu sendiri. Langkah kedua menghitung nilai *gap* tiap aspek. Lalu mencocokkan nilai *gap* untuk dihitung nilai *core factor* dan *secondary factor*. Sebelum masuk ke hasil akhir perankingan, perlu diketahui terlebih dahulu nilai total dari perhitungan *core factor* dan *secondary factor*.

Profile Matching pada penelitian tersebut secara garis besar merupakan proses membandingkan antara kompetensi individu kedalam kompetensi jabatan sehingga dapat diketahui perbedaan kompetensinya (disebut juga *gap*), semakin kecil *gap* yang dihasilkan maka bobot nilainya semakin besar yang berarti memiliki nilai yang lebih besar berdasarkan kriteria tersebut.

BAB 3. METODOLOGI PENELITIAN

Bab ini menjelaskan tentang gambaran tahapan yang sistematis yang dilakukan untuk menganalisa data untuk menjawab perumusan masalah sehingga dapat mencapai tujuan sebenarnya dari penelitian. Pada metodologi penelitian akan dijelaskan tentang jenis, tempat dan waktu, serta tahapan dari penelitian.

3.1. Jenis Penelitian

Pada penelitian ini, jenis penelitian yang digunakan ialah penelitian pengembangan, dikarenakan penelitian ini membuat dan mengembangkan suatu sistem yang telah ada, dan penelitian ini bukan penelitian yang dimaksudkan untuk menemukan teori atau mengujikan kebenaran suatu teori atau konsep dalam bentuk eksperimentasi.

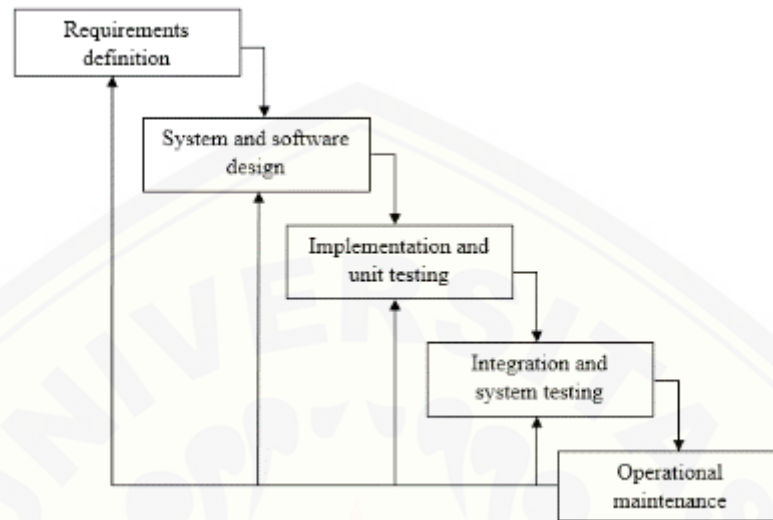
3.2. Tempat dan Waktu Penelitian

Penelitian dilakukan dengan menggunakan data dari kriteria ketuntasan minimal yang ada di SMAN 1 Gedangan Sidoarjo beserta hasil wawancara dengan guru atau tenaga pengajar yang ahli dalam bidang lomba OSN terkait. Sedangkan waktu penelitian dilaksanakan selama 4 bulan, yaitu Mei 2018 hingga Agustus 2018.

3.3. Tahapan Penelitian

Penelitian mengenai sistem pendataan dan seleksi calon peserta olimpiade sains ini menggunakan *Sistem Development Life Cycle (SDLC) Waterfall*. Pengembangan aplikasi ini menggunakan model waterfall karena disesuaikan dengan pengefisienan waktu bagi peneliti dalam mengerjakan penelitian ini. Model waterfall dipilih juga karena tahapan pada model waterfall lebih konsisten alur kerjanya. Selain itu, aplikasi yang akan dibangun merupakan aplikasi dalam skala yang kecil. Tahap pengembangan software dalam model waterfall dikerjakan bertahap mulai dari tahap tertinggi yaitu *Requirement Analysis*, *Sistem Design*, *Implementation*, *Unit Testing*, hingga *Maintenance*. Setiap proses pengerjaan dalam model waterfall tidak dapat dikerjakan sebelum tahap sebelumnya selesai. Sebagai contoh tahap *Sistem Design* tidak dapat dikerjakan sebelum *Requirement*

Analysis selesai. Berikut gambaran umum model *waterfall* dapat dilihat pada Gambar 3.1.



Gambar 3.1. Waterfall Model (Pressman, 2010).

3.3.1 Tahapan Analisis Kebutuhan

Pada tahap ini, hal yang dilakukan adalah merumuskan solusi dari permasalahan yang muncul. Data dan permasalahan diperoleh dari wawancara, studi sistem yang telah ada, dan menganalisis dokumen-dokumen yang terkait. Selanjutnya menganalisis kebutuhan yang harus dipenuhi oleh sistem yang akan dibangun, menentukan kebutuhan fungsional dan non fungsional serta menentukan fungsi dan fasilitas apa saja yang dibutuhkan.

a. Pengumpulan Data

Pengumpulan data menggunakan data hasil wawancara dengan guru dan staff pengajar SMAN 1 Gedangan yang bertempat di Jl. Raya Sedati KM 2, Wedi, Gedangan, Sidoarjo, Jawa Timur.

b. Analisis Data

Proses analisis data dilakukan setelah data diperoleh dari hasil wawancara pada tahap pengumpulan data yang kemudian akan digunakan untuk menjawab masalah dalam penelitian. Bukti pengumpulan data menggunakan catatan tertulis dari hasil wawancara.

c. Analisis Kebutuhan Fungsional dan Nonfungsional

Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan agar sistem dapat berjalan dengan baik. Kebutuhan fungsional dari sistem yaitu sistem dapat mendata dan menyeleksi calon siswa yang akan maju mewakili SMAN 1 Gedangan untuk ikut serta OSN tingkat kabupaten.

Analisis kebutuhan non-fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem untuk menunjang kebutuhan fungsional. Kebutuhan non-fungsional untuk mengoperasikan sistem ini meliputi kebutuhan perangkat keras, kebutuhan perangkat lunak dan pengguna yang menggunakan sistem yang akan dibangun.

3.3.2. Tahapan Desain Sistem

Pada penelitian ini, pembuatan desain sistem pada penelitian ini menggunakan *Unified Modeling Language* (UML) yang dirancang dengan konsep *Procedural Programming*. Pemodelan UML yang digunakan adalah sebagai berikut:

1. *Business Process*

Business Process adalah *Diagram* yang digunakan untuk mendefinisikan aktivitas dan proses. *Business Process* merupakan *Diagram* yang menggambarkan kebutuhan data antara user dan sistem.

2. *Use Case Diagram*

Use Case Diagram adalah model yang menggambarkan apa saja aktifitas yang dilakukan oleh suatu sistem dari sudut pandang pengamatan luar. Yang ditekankan dalam *Diagram Use Case* adalah “apa” yang dilakukan sistem, bukan “bagaimana” sistem bekerja. Sebuah *Use Case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Diagram Use Case* dapat sangat membantu dalam penyusunan *requirements* sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang *test case* untuk semua fitur yang ada pada sistem.

3. *Use Case Scenario*

Scenario berfungsi untuk menjelaskan alur sistem dari fitur yang ada di *job specification* dan *job description* yang ada pada *Diagram Use Case*. *Scenario* menjelaskan alur sistem dan keadaan yang terjadi pada *event* tertentu.

4. *Activity Diagram*

Activity Diagram menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing aktivitas berawal, keputusan yang mungkin terjadi, dan bagaimana aktivitas berakhir. *Activity Diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada eksekusi.

5. *Sequence Diagram*

Sequence Diagram menggambarkan aliran logika interaksi antar objek yang mengindikasikan komunikasi antar objek di dalam sistem yang disusun pada suatu urutan (*timeline*).

6. *Entity Relationship Diagram (ERD)*

Entity Relationship Diagram (ERD) merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan data yang mempunyai hubungan antar relasi.

3.3.3. Tahapan Implementasi Sistem

Pada tahap ini desain yang telah dibuat akan diimplementasikan ke dalam kode program. Beberapa hal yang dilakukan dalam tahap implementasi antara lain:

1. Penulisan kode program menggunakan bahasa *Page Hypertext Preprocessor (PHP)*, *Cascading Style Sheet (CSS)*, dan *Hyper Text Markup Language (HTML)* yang dibantu oleh *tool* Sublime Text 3.
2. *Database Management Sistem (DBMS)* yang digunakan adalah MySQL.

3.3.4. Tahapan Pengujian Sistem

Pada tahap ini dilakukan *testing* terhadap aplikasi yang telah dibuat, dan proses *testing* akan dilakukan secara langsung oleh *developer*. Tujuan dari *testing* adalah untuk menguji apakah aplikasi sudah sesuai dengan kebutuhan dan keinginan, serta mencari *bug* atau kesalahan pada aplikasi yang dibangun. Pengujian menggunakan dua metode yakni :

1. *White Box Testing*

Pengujian *White Box* dilakukan untuk mengecek *output* dan *input* serta kesesuaian algoritma yang telah dibuat. Pengujian ini dilakukan dengan menghitung *independent path* menggunakan *cyclomatic complexity*. Rumus dari *cyclomatic complexity* adalah sebagai berikut :

$$V(G) = E - N + 2$$

Keterangan :

E : Jumlah *edge* grafik alir

N : Jumlah *node* grafik alir

2. *Black Box Testing*

Testing ini dilakukan setelah *White Box testing*. Pengujian dilakukan berdasarkan apa yang dilihat, hanya fokus terhadap fungsionalitas dan *output*. Pengujian lebih ditujukan pada desain *software* sesuai standar dan reaksi apabila terdapat celah-celah *bug/vulnerabilitas* pada program aplikasi tersebut setelah dilakukan *White Box testing*.

3.3.5. Tahapan Pemeliharaan Sistem

Setelah melewati tahap pengujian. Maka tahap selanjutnya adalah *Deployment* yaitu tahap rilis sistem yang telah dibangun kepada *client*. Setelah *Deployment*, masuk ke tahap *Maintenance*. Pemeliharaan yang dimaksud adalah untuk menjaga sistem supaya tetap mampu beroperasi secara benar seperti pemeliharaan data, pembaharuan sistem sesuai kebutuhan baru, serta evaluasi keamanan data.

BAB 4. PENGEMBANGAN SISTEM

Bab ini menjelaskan pengembangan sistem dari hasil analisis yang telah dilakukan. Pada bab ini dijelaskan secara detail mulai dari tahap analisis, desain, pengkodean sistem, dan pengujian sistem yang dimulai dari deskripsi umum sistem.

4.1. Analisis Kebutuhan Sistem

Berdasarkan metode pengembangan sistem model *waterfall*, tahapan awal yang dilakukan adalah tahapan analisis. Tahapan analisis ini dilakukan terhadap objek penelitian untuk memperoleh kebutuhan-kebutuhan dari sistem yang dikembangkan, baik berupa kebutuhan fungsional maupun kebutuhan non-fungsional.

Dalam mengembangkan sistem ini, penulis menggunakan beberapa peralatan berupa *hardware* maupun *software* yang dapat diuraikan seperti berikut:

1. Laptop ASUS tipe N46VM dengan spesifikasi:
 - a. Processor intel core i7 3610qm
 - b. RAM 4GB DDR3
 - c. SSD 250GB
 - d. GPU Nvidia GT630M 2GB
2. Sistem operasi Windows 10
3. Microsoft Visio Professional untuk pemodelan UML
4. Visual Paradigm Community Edition untuk pemodelan UML
5. XAMPP Web Server dengan PHP dan MySQL
6. Sublime Text untuk melakukan rekayasa perangkat lunak pada aplikasi
7. Browser Google Chrome untuk testing aplikasi

Sedangkan untuk teknik pengumpulan data yang ditujukan untuk kebutuhan sistem itu sendiri, dilakukan dengan cara wawancara terhadap pihak sekolah. Informasi yang didapat dari wawancara tersebut adalah masalah tentang seleksi calon peserta. Penyeleksian calon peserta masih menggunakan cara lama yang pemecahannya dilakukan tanpa ada berdasarkan dokumen tertulis dan tanpa perhitungan yang sistematis. Dari hasil wawancara tersebut didapat bahwa semua

manajemen yang ada pada pihak sekolah untuk menyeleksi calon peserta OSN perlu disistemasi menggunakan sistem komputer. Sistem informasi pendataan dan seleksi calon peserta OSN mampu menjadi solusi yang tepat untuk diterapkan oleh pihak sekolah agar mampu menghasilkan luaran berupa kandidat yang terbaik. Adapun parameter yang digunakan sebagai tolak ukur dan penentuan kriteria penilaian adalah :

1. Nilai dan ketentuan minimal yang terdapat pada rapor siswa SMAN 1 Gedangan.
2. Hasil wawancara terhadap guru dan karyawan yang ahli dalam bidang studi OSN terkait.
3. Pedoman pelaksanaan OSN SMA berdasarkan Direktorat Jenderal Pendidikan Dasar dan Menengah, Kementerian Pendidikan dan Kebudayaan tahun 2018.

4.1.1. SOP (*Statement of Purpose*)

Sistem Pendataan dan Seleksi Calon Peserta Olimpiade Sains di SMAN 1 Gedangan merupakan sistem yang mendata sekaligus menyeleksi siswa-siswi SMAN 1 Gedangan yang ingin ikut serta dalam kegiatan Olimpiade Sains untuk maju ke seleksi tingkat kabupaten yang mengimplementasikan metode *Profile Matching* untuk algoritma perankingannya. Tujuan pengembangan sistem ini adalah untuk membantu pihak sekolah dalam mendata dan menyeleksi calon peserta OSN. Terdapat dua hak akses sistem yaitu admin dan wali kelas. Sistem dapat menambahkan data siswa, menentukan jumlah kriteria dan subkriteria, penentuan bobot *gap*, dan memunculkan hasil perhitungan. Sistem dibangun berbasis *web*.

4.1.2. Kebutuhan Fungsional

Kebutuhan fungsional sistem berisi fitur-fitur inti yang harus dipenuhi dalam sistem agar sistem mampu difungsikan sesuai dengan tujuan dan kebutuhan pengguna terhadap sistem itu sendiri. Kebutuhan fungsional dari sistem ini yaitu:

1. Sistem mampu melakukan *Sign In / Sign Out* untuk *user* yang akan masuk ataupun keluar dari sistem

2. Sistem mampu mengolah data *user* atau hak akses sistem (*View, insert, update, delete*)
3. Sistem mampu mengubah *password* dari *user* terautentikasi yang sedang *Sign In* ke sistem (*update*)
4. Sistem mampu mengelola kuota lomba per tahun / periode (*View, insert, delete*)
5. Sistem mampu mengelola data siswa / data alternatif (*View, insert, update, delete*)
6. Sistem mampu mengelola data nilai siswa / data nilai alternatif (*View, update*)
7. Sistem mampu mengelola data kriteria (*View, insert, update, delete*)
8. Sistem mampu mengelola data subkriteria (*View, insert, update, delete*)
9. Sistem mampu mengelola bobot *gap* (*View, insert, update, delete*)
10. Sistem mampu menampilkan hasil perhitungan akhir (*View*)

4.1.3. Kebutuhan Non-Fungsional

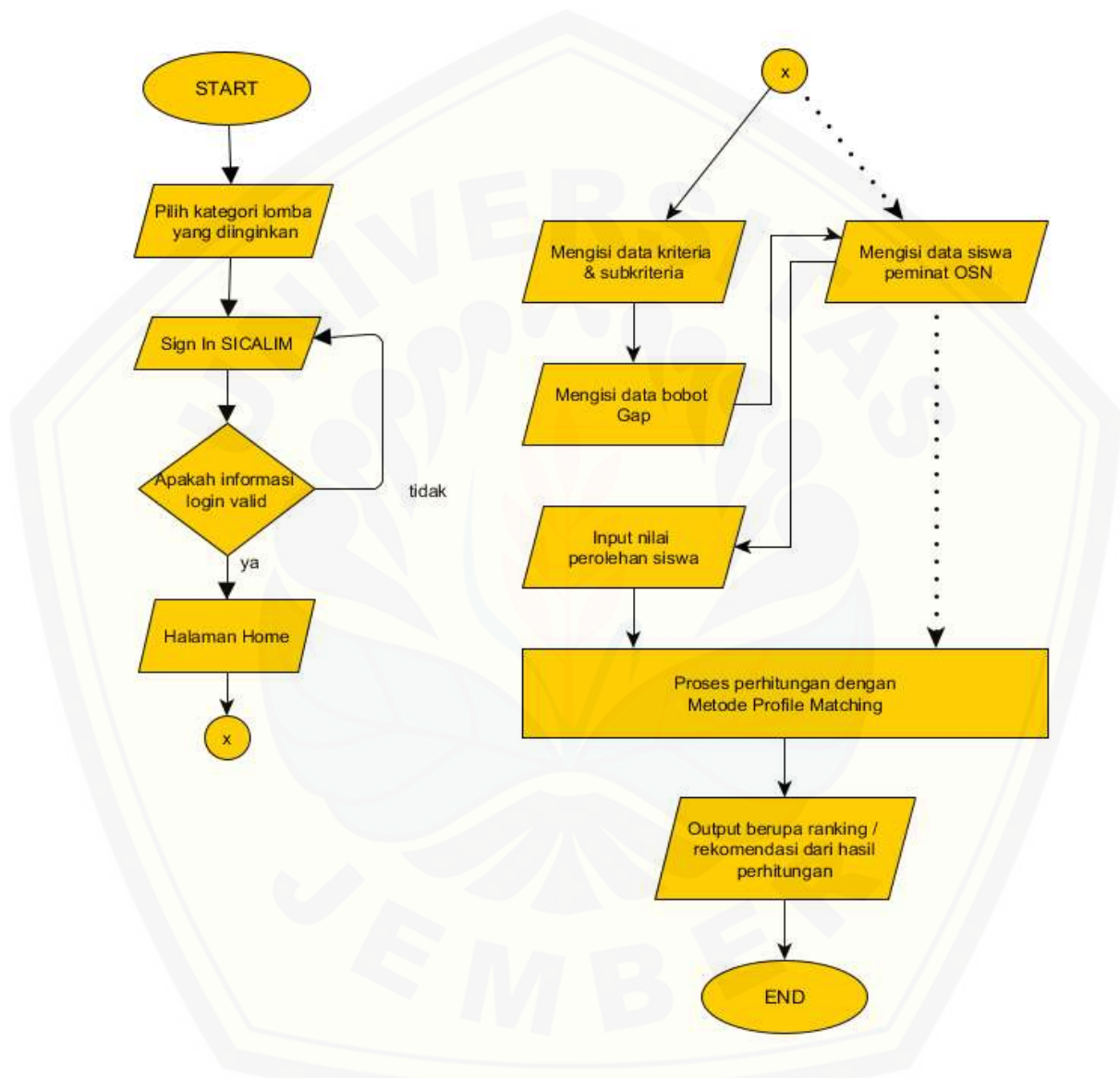
Kebutuhan non-fungsional merupakan fitur-fitur yang dimiliki sistem untuk mendukung sistem dalam memenuhi fungsionalitasnya untuk dapat memenuhi kebutuhan dari pengguna. Kebutuhan non-fungsional dari sistem ini yaitu:

1. Sistem bekerja sesuai dengan fungsinya dan dapat dijalankan pada semua komputer maupun *smartphone* pada *browser* yang berbeda-beda.
2. Tampilan dan bahasa komunikasi sistem mudah dimengerti oleh *user* untuk memberikan kenyamanan pemakaian dan memudahkan pengoperasian.
3. *Username* dan *password* sistem yang digunakan untuk *Sign In* pada *database* menggunakan enkripsi sehingga meminimalisir terjadinya peretasan sistem.

4.1.4. Gambaran Umum Sistem

Sistem pendukung keputusan seleksi calon peserta olimpiade yang akan dibangun adalah sistem informasi perankingan siswa dengan menggunakan algoritma *Profile Matching*. Sistem yang dirancang nantinya dapat mengelola data siswa yang mengajukan diri dan mengurutkan kandidat terbaik untuk ikut serta dalam pelaksanaan Olimpiade Sains pada bidang lomba yang telah ditentukan. Proses *input* akan dilakukan oleh admin maupun wali kelas yang nanti akan dapat

menambahkan alternatif (calon peserta) ke dalam sistem. Admin memiliki hak akses yang lebih banyak jika dibandingkan dengan wali kelas, yaitu dapat menentukan jumlah maupun bobot kriteria maupun sub-kriteria, serta bobot *gap*. Gambaran umum sistem dapat dilihat pada Gambar 4.1.



Gambar 4.1. Gambaran Umum Sistem

4.2. Desain Sistem

Tahapan yang dilakukan setelah melakukan analisis kebutuhan sistem yaitu tahap pengembangan sistem yang dapat digambarkan dengan desain sistem. Desain “Rancang Bangun Sistem Pendataan dan Seleksi Calon Peserta Olimpiade Sains Dengan Menggunakan Metode *Profile Matching*” ini meliputi *Business Process*,

Use Case Diagram, Use Case Scenario, Activity Diagram, Sequence Diagram, dan Entity Relationship Diagram.

4.2.1. Elemen Data *Input* dan Data *Output* Sistem

Elemen Data *Input* dan Data *Output* Sistem merupakan *Diagram* yang menggambarkan kebutuhan data antara *user* dan sistem tersebut. Elemen ini terdiri atas beberapa poin diantaranya:

1. *Input* : Data yang dimasukkan ke dalam sistem
2. *Output* : Data yang dihasilkan oleh sistem
3. *Goal* : Tujuan yang dihasilkan oleh suatu sistem
4. *Uses* : *Platform* yang menjadi basis sistem
5. *Supply* : Metode atau algoritma yang dipakai pada sistem
6. *Process* : Sistem yang bekerja yakni sistem pendataan dan seleksi

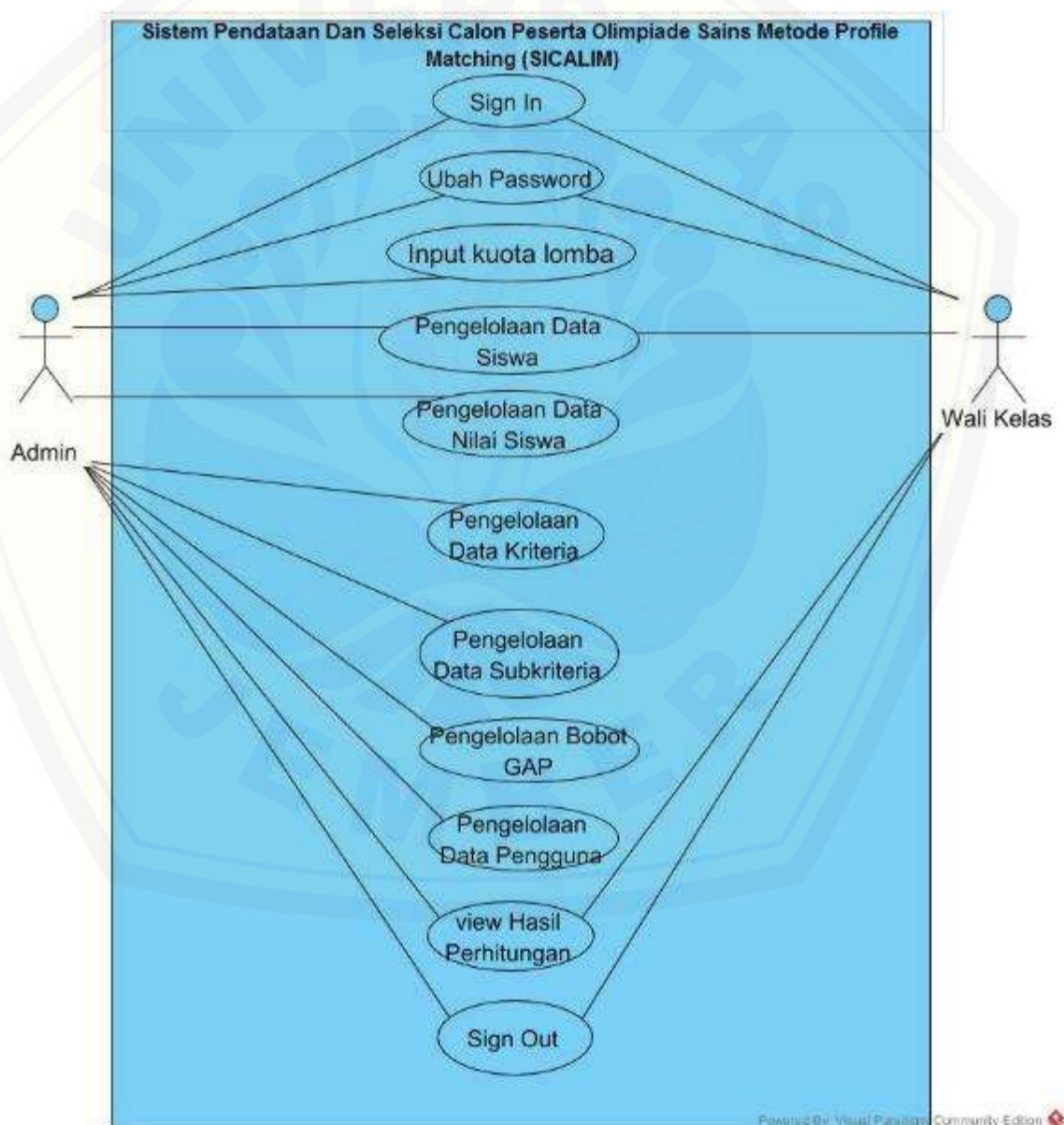
Pada Elemen Data *Input* dan Data *Output* sistem ini memiliki 6 buah *input* yaitu data siswa, data nilai siswa, kriteria & bobot, subkriteria & nilai ideal, kuota lomba, serta bobot *gap*. *Output* yang dihasilkan yaitu hasil perhitungan, *ranking*, dan tabel siswa terpilih. *Goal* dari sistem ini yaitu menentukan alternatif terbaik dari semua alternatif yang ada pada sistem. *Uses* dari sistem ini berbasis *website* dan *supply* menggunakan metode *Profile Matching*. Elemen Data *Input* dan Data *Output* ditunjukkan pada Gambar 4.2.



Gambar 4.2. Elemen Data *Input* dan Data *Output* Sistem

4.2.2. Use Case Diagram

Use Case Diagram merupakan pemodelan yang dibuat untuk menggambarkan interaksi antara aktor dengan sistem. Melalui *Use Case Diagram* dapat diketahui hubungan atau relasi aktor terhadap sistem sesuai dengan hak akses yang dimiliki oleh masing-masing aktor. *Use Case Diagram* ditunjukkan pada Gambar 4.3.



Gambar 4.3. Use Case Diagram

Gambar 4.3 menunjukkan *Use Case Diagram* sistem yang terdiri dari 2 aktor dengan 11 *Use Case*. *Use Case* ini mempunyai penjelasan berupa tabel definisi tugas aktor yang menggambarkan siapa saja dan apa saja tugas yang dimiliki oleh setiap aktor yang dapat dilihat pada tabel 4.1. Penjelasan lainnya yaitu disebut definisi *Use Case* yang menggambarkan fungsionalitas dari setiap *Use Case* yang dapat dilihat pada tabel 4.2.

Tabel 4.1. Definisi Aktor

NO.	AKTOR	DEFINISI TUGAS
1.	Admin	Seorang admin adalah aktor yang memiliki hak akses penuh terhadap sistem, yakni meliputi semua aspek dari sistem seperti kuota lomba, data siswa, data nilai siswa, kriteria, subkriteria, bobot <i>gap</i> , data pengguna (<i>Sign In</i>), serta <i>View</i> hasil perhitungan.
2.	Wali Kelas	Seorang wali kelas adalah aktor yang memiliki akses terhadap beberapa bagian dari sistem, seperti data siswa, dan <i>View</i> hasil perhitungan.

Tabel 4.2. Definisi *Use Case*

NO.	USE CASE	DESKRIPSI
1.	<i>Sign In</i>	<i>Use Case Sign In</i> merupakan <i>Use Case</i> yang menggambarkan proses masuk ke dalam hak akses masing-masing aktor.
2.	Ubah <i>Password</i>	<i>Use Case Ubah Password</i> merupakan <i>Use Case</i> yang menggambarkan penyediaan layanan untuk masing-masing aktor terautentikasi untuk dapat mengubah <i>password</i> sendiri yang digunakan untuk <i>Sign In</i> sesuai dengan yang diinginkan.
3.	Input Kuota Lomba	<i>Use Case Input Kuota Lomba</i> merupakan <i>Use Case</i> yang menggambarkan pengelolaan data kuota lomba per tahun. Yang akan berpengaruh pada menu hasil perhitungan.

4.	Pengelolaan Data Siswa	<i>Use Case</i> Pengelolaan Data Siswa menggambarkan sebuah hak akses oleh sistem kepada aktor terpilih yang meliputi <i>View</i> tabel data siswa, tambah data siswa, <i>update</i> data siswa, dan hapus data siswa.
5.	Pengelolaan Data Nilai Siswa	<i>Use Case</i> Pengelolaan Data Nilai Siswa menggambarkan sebuah hak akses oleh sistem kepada aktor terpilih yang meliputi <i>View</i> tabel data nilai siswa dan <i>update</i> data nilai siswa.
6.	Pengelolaan Data Kriteria	<i>Use Case</i> Penngelolaan Data Kriteria menggambarkan sebuah hak akses oleh sistem kepada aktor terpilih yang meliputi <i>View</i> tabel data kriteria, tambah data kriteria, <i>update</i> data kriteria, dan hapus data kriteria.
7.	Pengelolaan Data Subkriteria	<i>Use Case</i> Pengelolaan Data Subkriteria menggambarkan sebuah hak akses oleh sistem kepada aktor terpilih yang meliputi <i>View</i> tabel data subkriteria, tambah data subkriteria, <i>update</i> data subkriteria, dan hapus data subkriteria.
8.	Pengelolaan Bobot <i>gap</i>	<i>Use Case</i> Pengelolaan Bobot <i>gap</i> menggambarkan sebuah hak akses oleh sistem kepada aktor terpilih yang meliputi <i>View</i> tabel bobot <i>gap</i> , tambah bobot <i>gap</i> , <i>update</i> bobot <i>gap</i> , dan hapus bobot <i>gap</i> .
9.	Pengelolaan Data Pengguna	<i>Use Case</i> Pengelolaan Data Pengguna menggambarkan sebuah hak akses oleh sistem kepada admin untuk mengelola data <i>user</i> yang digunakan untuk <i>Sign In</i> seperti <i>View</i> data pengguna, tambah pengguna, <i>update</i> data pengguna, dan hapus pengguna.
10.	<i>View</i> Hasil Perhitungan	<i>Use Case View</i> Hasil Perhitungan menggambarkan tampilan mengenai perurutan <i>ranking</i> siswa dari yang terbaik sampai yang terburuk. Perhitungan ini dilandaskan berdasarkan metode <i>Profile Matching</i> .
11.	<i>Sign Out</i>	<i>Use Case Sign Out</i> merupakan <i>Use Case</i> yang menggambarkan proses bagi aktor untuk keluar dari sistem.

4.2.3. Use Case Scenario

Use Case Scenario berfungsi untuk menjelaskan alur utama dari sebuah sistem serta alur alternatif yang dilakukan oleh para aktor dalam menggunakan sistem. Jumlah *Use Case Scenario* sesuai dengan yang ada pada *Use Case Diagram* seperti pada gambar 4.3 yaitu berjumlah 11 bagian dan masing-masing bagian tersebut memiliki kondisi atau skenarionya sendiri-sendiri.

1. Skenario *Sign In*

Skenario *Sign In* merupakan penjelasan urutan reaksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif *Use Case Scenario Sign In*. Pada skenario tersebut dapat dilihat bagaimana alur perjalanan suatu fitur *Sign In* antara aksi yang dilakukan oleh aktor yaitu admin dan wali kelas ketika akan melakukan *Sign In*, kemudian bagaimana reaksi dari sistem untuk merespon aksi tersebut sehingga aktor telah melakukan *Sign In* pada sistem. Proses lengkap skenario *Sign In* dapat dilihat pada Lampiran A1.

2. Skenario Ubah *Password*

Skenario ubah *password* merupakan penjelasan urutan reaksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif *Use Case Scenario* ubah *password*. Skenario ini menggambarkan salah satu fitur kelebihan yang dimiliki oleh sistem yakni setiap pengguna terautentikasi yang berhasil masuk ke sistem dapat mengganti *password* sesuai yang diinginkan pengguna tersebut tanpa melapor ke admin. Pada skenario tersebut dapat dilihat bagaimana alur perjalanan suatu fitur ubah *password* antara aksi yang dilakukan oleh aktor yaitu admin dan wali kelas ketika akan melakukan ubah *password*, kemudian bagaimana reaksi dari sistem untuk merespon aksi tersebut sehingga aktor telah melakukan perubahan *password* pada sistem. Proses lengkap skenario ubah *password* dapat dilihat pada Lampiran A2.

3. Skenario Input Kuota Lomba

Skenario input kuota lomba merupakan penjelasan urutan reaksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif *Use Case Scenario* input

kuota lomba. Skenario ini menggambarkan tentang salah satu fitur yang dimiliki oleh sistem yaitu dapat menyimpan jumlah kuota per periode lomba yang kemudian akan berpengaruh pada menu hasil perhitungan pada tabel siswa terpilih. Pada skenario tersebut dapat dilihat bagaimana alur perjalanan suatu fitur kelola data kuota lomba antara aksi yang dilakukan oleh aktor yaitu admin ketika melakukan pengelolaan data kuota lomba, kemudian bagaimana reaksi dari sistem untuk merespon aksi tersebut sehingga aktor telah melakukan pengelolaan data yang berhubungan dengan kuota lomba pada sistem. Proses lengkap skenario input kuota lomba dapat dilihat pada Lampiran A3.

4. Skenario Pengelolaan Data Siswa

Skenario pengelolaan data siswa merupakan penjelasan urutan reaksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif *Use Case Scenario* pengelolaan data siswa. Skenario ini menggambarkan tentang salah satu fitur yang dimiliki oleh sistem yaitu dapat melihat, menambah, mengubah dan menghapus siswa yang ada. Pada skenario tersebut dapat dilihat bagaimana alur perjalanan suatu fitur kelola data siswa antara aksi yang dilakukan oleh aktor yaitu admin dan wali kelas ketika melakukan pengelolaan data siswa, kemudian bagaimana reaksi dari sistem untuk merespon aksi tersebut sehingga aktor telah melakukan pengelolaan data yang berhubungan dengan siswa pada sistem. Proses lengkap skenario pengelolaan data siswa dapat dilihat pada Lampiran A4.

5. Skenario Pengelolaan Data Nilai Siswa

Skenario pengelolaan data nilai siswa merupakan penjelasan urutan reaksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif *Use Case Scenario* pengelolaan data nilai siswa. Skenario ini menggambarkan tentang salah satu fitur yang dimiliki oleh sistem yaitu dapat melihat dan mengubah nilai dari masing-masing kriteria dan subkriteria dari setiap siswa yang ada. Data umum siswa dan data nilai siswa sengaja dipisah agar mempermudah pengguna dalam menginputkan data. Pada skenario tersebut dapat dilihat bagaimana alur perjalanan suatu fitur kelola data nilai siswa antara aksi yang dilakukan oleh aktor yaitu admin dan wali kelas ketika melakukan pengelolaan data nilai siswa, kemudian bagaimana

reaksi dari sistem untuk merespon aksi tersebut sehingga aktor telah melakukan pengelolaan data yang berhubungan dengan nilai masing-masing siswa pada sistem. Proses lengkap skenario pengelolaan data nilai siswa dapat dilihat pada Lampiran A5.

6. Skenario Pengelolaan Data Kriteria

Skenario pengelolaan data kriteria merupakan penjelasan urutan reaksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif *Use Case Scenario* pengelolaan data kriteria. Skenario ini menggambarkan tentang salah satu fitur yang dimiliki oleh sistem yaitu dapat melihat, menambah, mengubah, dan menghapus kriteria yang ada. Pada skenario tersebut dapat dilihat bagaimana alur perjalanan suatu fitur kelola data kriteria antara aksi yang dilakukan oleh aktor yaitu admin ketika melakukan pengelolaan data kriteria, kemudian bagaimana reaksi dari sistem untuk merespon aksi tersebut sehingga aktor telah melakukan pengelolaan data yang berhubungan dengan kriteria pada sistem. Proses lengkap skenario pengelolaan data kriteria dapat dilihat pada Lampiran A6.

7. Skenario Pengelolaan Data Subkriteria

Skenario pengelolaan data subkriteria merupakan penjelasan urutan reaksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif *Use Case Scenario* pengelolaan data subkriteria. Skenario ini menggambarkan tentang salah satu fitur yang dimiliki oleh sistem yaitu dapat melihat, menambah, mengubah, dan menghapus subkriteria yang ada. Pada skenario tersebut dapat dilihat bagaimana alur perjalanan suatu fitur kelola data subkriteria antara aksi yang dilakukan oleh aktor yaitu admin ketika melakukan pengelolaan data subkriteria, kemudian bagaimana reaksi dari sistem untuk merespon aksi tersebut sehingga aktor telah melakukan pengelolaan data yang berhubungan dengan subkriteria pada sistem. Proses lengkap skenario pengelolaan data subkriteria dapat dilihat pada Lampiran A7.

8. Skenario Pengelolaan Bobot *gap*

Skenario pengelolaan bobot *gap* merupakan penjelasan urutan reaksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif *Use Case Scenario* pengelolaan bobot *gap*. Skenario ini menggambarkan tentang salah satu fitur yang dimiliki oleh sistem yaitu dapat melihat, menambah, mengubah, dan menghapus daftar nilai bobot *gap* yang ada. Pada skenario tersebut dapat dilihat bagaimana alur perjalanan suatu fitur kelola bobot *gap* antara aksi yang dilakukan oleh aktor yaitu admin ketika melakukan pengelolaan bobot *gap*, kemudian bagaimana reaksi dari sistem untuk merespon aksi tersebut sehingga aktor telah melakukan pengelolaan data yang berhubungan dengan bobot *gap* pada sistem. Proses lengkap skenario pengelolaan bobot *gap* dapat dilihat pada Lampiran A8.

9. Skenario Pengelolaan Data Pengguna

Skenario pengelolaan data pengguna merupakan penjelasan urutan reaksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif *Use Case Scenario* pengelolaan data pengguna. Skenario ini menggambarkan salah satu fitur yang dimiliki oleh sistem yakni admin dapat mengelola data *user* atau pengguna (melihat, menambah, mengubah, dan menghapus) yang akan atau sedang berkepentingan untuk mengakses sistem, seperti wali kelas maupun menambahkan admin lainnya bila perlu. Pada skenario tersebut dapat dilihat bagaimana alur perjalanan suatu fitur kelola data pengguna antara aksi yang dilakukan oleh aktor yaitu admin ketika akan melakukan kelola data pengguna, kemudian bagaimana reaksi dari sistem untuk merespon aksi tersebut sehingga aktor telah melakukan kelola data pengguna pada sistem. Proses lengkap skenario pengelolaan data pengguna dapat dilihat pada Lampiran A9.

10. Skenario *View* Hasil Perhitungan

Skenario ini menggambarkan secara singkat aksi aktor dan reaksi sistem terkait pengamatan dari hasil olah sistem oleh algoritma. Proses lengkap skenario *View* Hasil Perhitungan dapat dilihat pada Lampiran A10.

11. Skenario *Sign Out*

Skenario *Sign Out* merupakan penjelasan urutan reaksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif *Use Case Scenario Sign Out*. Pada skenario tersebut dapat dilihat bagaimana alur perjalanan suatu fitur *Sign Out* antara aksi yang dilakukan oleh aktor yaitu admin dan wali kelas ketika akan melakukan *Sign Out*, kemudian bagaimana reaksi dari sistem untuk merespon aksi tersebut sehingga aktor telah melakukan *Sign Out* pada sistem. Proses lengkap skenario *Sign Out* dapat dilihat pada Lampiran A11.

4.2.4 Activity Diagram

Activity Diagram merupakan desain visualisasi yang menggambarkan aliran aktivitas dalam sistem yang akan dibangun. Secara umum, *Activity Diagram* merupakan *Use Case Scenario* yang divisualisasi kedalam bentuk *Diagram* alir, sehingga terlihat lebih mudah dipahami. Jumlah *Activity Diagram* sesuai dengan yang ada pada *Use Case Diagram* seperti pada gambar 4.3 yaitu berjumlah 11 bagian dan masing-masing bagian tersebut memiliki kondisinya sendiri-sendiri.

1. *Activity Diagram Sign In*

Activity Diagram Sign In menjelaskan bagaimana alur antara aktor yaitu admin dan wali kelas dengan sistem untuk menjalankan fungsi *Sign In*. Proses lengkapnya dapat dilihat pada Lampiran B1.

2. *Activity Diagram Ubah Password*

Activity Diagram Ubah Password menjelaskan bagaimana alur antara aktor yaitu admin dan wali kelas dengan sistem untuk menjalankan fungsi *Ubah Password*. Setelah mendapatkan hak akses dengan proses *Sign In*, tiap user yang terautentikasi berhak mengganti *password*nya jika diinginkan. Proses lengkapnya dapat dilihat pada Lampiran B2.

3. *Activity Diagram Input Kuota Lomba*

Activity Diagram Input Kuota Lomba menjelaskan bagaimana alur antara aktor yaitu admin dengan sistem untuk menjalankan fungsi *Input Kuota Lomba*. Pada *Activity Diagram* ini, pengelolaan kuota lomba dibagi menjadi 2 bagian, yaitu : tambah data, dan hapus data. Proses lengkapnya dapat dilihat pada Lampiran B3.

4. *Activity Diagram* Pengelolaan Data Siswa

Activity Diagram Pengelolaan Data Siswa menjelaskan bagaimana alur antara aktor yaitu admin dan wali kelas dengan sistem untuk menjalankan fungsi kelola data siswa. Pada *Activity Diagram* ini, pengelolaan data siswa dibagi menjadi 3 bagian yaitu : tambah data, ubah data, dan hapus data. Proses lengkapnya dapat dilihat pada Lampiran B4.

5. *Activity Diagram* Pengelolaan Data Nilai Siswa

Activity Diagram Pengelolaan Data Nilai Siswa menjelaskan bagaimana alur antara aktor yaitu admin dan wali kelas dengan sistem untuk menjalankan fungsi kelola data nilai siswa. Proses lengkapnya dapat dilihat pada Lampiran B5..

6. *Activity Diagram* Pengelolaan Data Kriteria

Activity Diagram Pengelolaan Data Kriteria menjelaskan bagaimana alur antara aktor yaitu admin dengan sistem untuk menjalankan fungsi kelola data kriteria. Pada *Activity Diagram* ini, pengelolaan data kriteria dibagi menjadi 3 bagian yaitu : tambah data, ubah data, dan hapus data. Proses lengkapnya dapat dilihat pada Lampiran B6.

7. *Activity Diagram* Pengelolaan Data Subkriteria

Activity Diagram Pengelolaan Data Subkriteria menjelaskan bagaimana alur antara aktor yaitu admin dengan sistem untuk menjalankan fungsi kelola data subkriteria. Pada *Activity Diagram* ini, pengelolaan data subkriteria dibagi menjadi 3 bagian yaitu : tambah data, ubah data, dan hapus data. Proses lengkapnya dapat dilihat pada Lampiran B7.

8. *Activity Diagram* Pengelolaan Bobot *gap*

Activity Diagram Pengelolaan Bobot *gap* menjelaskan bagaimana alur antara aktor yaitu admin dengan sistem untuk menjalankan fungsi kelola data bobot *gap*. Pada *Activity Diagram* ini, pengelolaan data bobot *gap* dibagi menjadi 3 bagian yaitu : tambah data, ubah data, dan hapus data. Proses lengkapnya dapat dilihat pada Lampiran B8.

9. *Activity Diagram* Pengelolaan Data Pengguna

Activity Diagram Pengelolaan Data Pengguna menjelaskan bagaimana alur antara aktor yaitu admin dengan sistem untuk menjalankan fungsi kelola data pengguna. Pada *Activity Diagram* ini, pengelolaan data pengguna dibagi menjadi 3 bagian yaitu : tambah data, ubah data, dan hapus data. Proses lengkapnya dapat dilihat pada Lampiran B9.

10. *Activity Diagram View Hasil Perhitungan*

Activity Diagram View Hasil Perhitungan menjelaskan secara singkat aksi aktor yaitu admin dan wali kelas dalam mengakses hasil perhitungan akhir. Proses lengkapnya dapat dilihat pada Lampiran B10.

11. *Activity Diagram Sign Out*

Activity Diagram Sign Out menjelaskan bagaimana alur antara aktor yaitu admin dan wali kelas dengan sistem untuk menjalankan fungsi *Sign In*. Proses lengkapnya dapat dilihat pada Lampiran B11.

4.2.5 *Sequence Diagram*

Sequence Diagram adalah *Diagram* visualisasi yang menampilkan urutan interaksi-interaksi antar objek di dalam sistem. *Sequence Diagram* digunakan untuk menggambarkan aliran logika dalam sistem. Jumlah *Sequence Diagram* sesuai dengan yang ada pada *Use Case Diagram* seperti pada gambar 4.3 yaitu berjumlah 11 bagian dan masing-masing bagian tersebut memiliki kondisinya sendiri-sendiri.

1. *Sequence Diagram Sign In*

Sequence Diagram Sign In digunakan untuk menjelaskan fungsi yang akan dibuat pada *Use Case Sign In* yang menjelaskan aktor yaitu admin dan wali kelas untuk masuk ke sistem. Masing-masing fungsi dari objek yang ada pada komponen sistem yang terlibat ditampilkan secara visual dengan gambar. Proses lengkapnya dapat dilihat pada Lampiran C1.

2. *Sequence Diagram Ubah Password*

Sequence Diagram Ubah Password digunakan untuk menjelaskan fungsi yang akan dibuat pada *Use Case ubah password* yang menjelaskan aktor yaitu admin dan wali kelas untuk merubah *password* setelah masuk ke sistem. Masing-

masing fungsi dari objek yang ada pada komponen sistem yang terlibat ditampilkan secara visual dengan gambar. Proses lengkapnya dapat dilihat pada Lampiran C2.

3. *Sequence Diagram* Input Kuota Lomba

Sequence Diagram Input Kuota Lomba digunakan untuk menjelaskan fungsi yang akan dibuat pada *Use Case* input kuota lomba yang menjelaskan aktor yaitu admin untuk menginputkan kuota lomba per periode atau tahun lomba. Pada *Sequence Diagram* ini, pengelolaan data kuota lomba dibagi menjadi 2 bagian yaitu : tambah data, dan hapus data. Masing-masing fungsi dari objek yang ada pada komponen sistem yang terlihat ditampilkan secara visual dengan gambar. Proses lengkapnya dapat dilihat pada Lampiran C3.

4. *Sequence Diagram* Pengelolaan Data Siswa

Sequence Diagram Pengelolaan Data Siswa digunakan untuk menjelaskan fungsi yang akan dibuat pada *Use Case* pengelolaan data siswa yang menjelaskan aktor yaitu admin dan wali kelas untuk mengelola data siswa. Pada *Sequence Diagram* ini, pengelolaan data siswa dibagi menjadi 3 bagian yaitu : tambah data, ubah data, dan hapus data. Masing-masing fungsi dari objek yang ada pada komponen sistem yang terlibat ditampilkan secara visual dengan gambar. Proses lengkapnya dapat dilihat pada Lampiran C4.

5. *Sequence Diagram* Pengelolaan Data Nilai Siswa

Sequence Diagram Pengelolaan Data Nilai Siswa digunakan untuk menjelaskan fungsi yang akan dibuat pada *Use Case* pengelolaan data nilai siswa yang menjelaskan aktor yaitu admin dan wali kelas untuk mengelola data nilai siswa. Masing-masing fungsi dari objek yang ada pada komponen sistem yang terlibat ditampilkan secara visual dengan gambar. Proses lengkapnya dapat dilihat pada Lampiran C5.

6. *Sequence Diagram* Pengelolaan Data Kriteria

Sequence Diagram Pengelolaan Data Kriteria digunakan untuk menjelaskan fungsi yang akan dibuat pada *Use Case* pengelolaan data kriteria yang menjelaskan aktor yaitu admin untuk mengelola data kriteria. Pada *Sequence Diagram* ini,

pengelolaan data kriteria dibagi menjadi 3 bagian yaitu : tambah data, ubah data, dan hapus data. Masing-masing fungsi dari objek yang ada pada komponen sistem yang terlibat ditampilkan secara visual dengan gambar. Proses lengkapnya dapat dilihat pada Lampiran C6.

7. *Sequence Diagram* Pengelolaan Data Subkriteria

Sequence Diagram Pengelolaan Data Subkriteria digunakan untuk menjelaskan fungsi yang akan dibuat pada *Use Case* pengelolaan data subkriteria yang menjelaskan aktor yaitu admin untuk mengelola data subkriteria. Pada *Sequence Diagram* ini, pengelolaan data subkriteria dibagi menjadi 3 bagian yaitu : tambah data, ubah data, dan hapus data. Masing-masing fungsi dari objek yang ada pada komponen sistem yang terlibat ditampilkan secara visual dengan gambar. Proses lengkapnya dapat dilihat pada Lampiran C7.

8. *Sequence Diagram* Pengelolaan Bobot *gap*

Sequence Diagram Pengelolaan Bobot *gap* digunakan untuk menjelaskan fungsi yang akan dibuat pada *Use Case* pengelolaan bobot *gap* yang menjelaskan aktor yaitu admin untuk mengelola data bobot *gap*. Pada *Sequence Diagram* ini, pengelolaan data subkriteria dibagi menjadi 3 bagian yaitu : tambah data, ubah data, dan hapus data. Masing-masing fungsi dari objek yang ada pada komponen sistem yang terlibat ditampilkan secara visual dengan gambar. Proses lengkapnya dapat dilihat pada Lampiran C8.

9. *Sequence Diagram* Pengelolaan Data Pengguna

Sequence Diagram Pengelolaan Data Pengguna digunakan untuk menjelaskan fungsi yang akan dibuat pada *Use Case* pengelolaan data pengguna yang menjelaskan aktor yaitu admin untuk mengelola data pengguna. Pada *Sequence Diagram* ini, pengelolaan data pengguna dibagi menjadi 3 bagian yaitu : tambah data, ubah data, dan hapus data. Masing-masing fungsi dari objek yang ada pada komponen sistem yang terlibat ditampilkan secara visual dengan gambar. Proses lengkapnya dapat dilihat pada Lampiran C9.

10. *Sequence Diagram View* Hasil Perhitungan

Sequence Diagram View Hasil Perhitungan digunakan untuk menjelaskan fungsi yang akan dibuat pada *Use Case View* hasil perhitungan yang menjelaskan aktor yaitu admin dan wali kelas untuk melihat hasil perhitungan akhir oleh sistem.

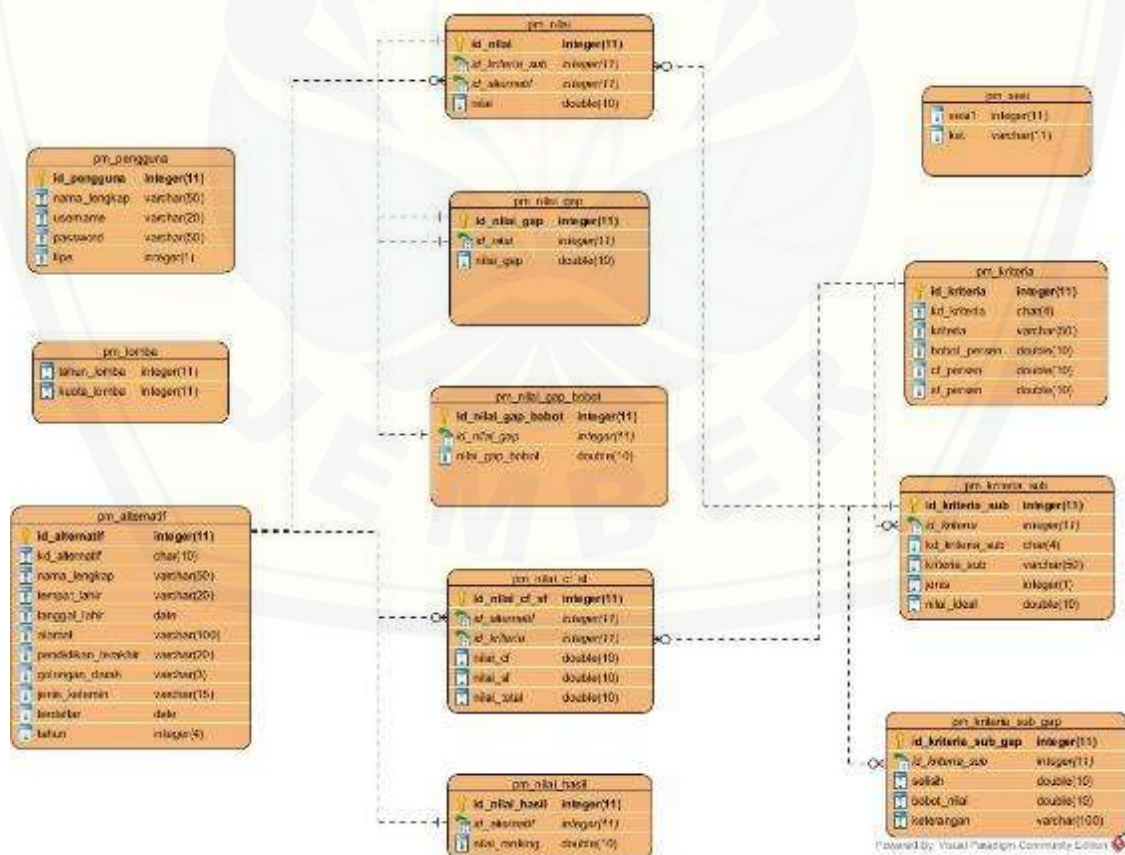
Masing-masing fungsi dari objek yang ada pada komponen sistem yang terlibat ditampilkan secara visual dengan gambar. Proses lengkapnya dapat dilihat pada Lampiran C10.

11. *Sequence Diagram Sign Out*

Sequence Diagram Sign Out digunakan untuk menjelaskan fungsi yang akan dibuat pada *Use Case Sign Out* yang menjelaskan aktor yaitu admin dan wali kelas untuk keluar dari sistem. Masing-masing fungsi dari objek yang ada pada komponen sistem yang terlibat ditampilkan secara visual dengan gambar. Proses lengkapnya dapat dilihat pada Lampiran C11.

4.2.6. *Entity Relationship Diagram*

Entity Relationship Diagram (ERD) sistem ini merupakan gambaran komponen dan struktur database untuk digunakan dalam pembuatan sistem ERD yang diimplementasikan pada sistem ini terdiri dari 12 entitas yang dapat dilihat pada Gambar 4.6.



Gambar 4.4. *Entity Relationship Diagram (ERD)*

Sistem yang dirancang secara keseluruhan memiliki 12 entitas atau yang biasa disebut dengan tabel database. Setiap entitas memiliki fungsinya tersendiri yang akan diuraikan pada deskripsi entitas pada tabel 4.3 berikut ini.

Tabel 4.3. Deskripsi Entitas Pada Sistem

No.	Nama Entitas	Fungsi
1.	pm_pengguna	Menyimpan data user / pengguna sistem yang digunakan untuk <i>Sign In</i> ke dalam sistem
2.	pm_lomba	Menyimpan data kuota yang ada pada per periode lomba
3.	pm_alternatif	Menyimpan data siswa
4.	pm_sesi	Menyimpan sesi sementara submit data
5.	pm_kriteria	Menyimpan data kriteria
6.	pm_kriteria_sub	Menyimpan data subkriteria
7.	pm_kriteria_sub_gap	Menyimpan data bobot <i>gap</i>
8.	pm_nilai	Menyimpan data nilai siswa
9.	pm_nilai_gap	Menyimpan data nilai hasil pemetaan selisih <i>gap</i>
10.	pm_nilai_gap_bobot	Menyimpan data nilai hasil pemetaan bobot <i>gap</i>
11.	pm_nilai_cf_sf	Menyimpan data nilai hasil perhitungan nilai CF dan SF
12.	pm_nilai_hasil	Menyimpan data nilai hasil perhitungan nilai total

4.3. Pengkodean Sistem

Desain yang telah dibuat akan diimplementasikan ke dalam kode program. Beberapa hal yang dilakukan dalam tahap implementasi antara lain:

- Penulisan kode program (*Coding*) menggunakan bahasa pemrograman *Page Hyper Text Pre-Processor (PHP)*, *HyperText Markup Language (HTML)*, *Cascading Style Sheet (CSS)*.
- Manajemen basisdata menggunakan *DBMS MySQL*. Kode program perhitungan metode algoritma *Profile Matching* terdapat pada file perhitungan.php. Penulisan kode program perhitungan metode *Profile Matching* tersebut terdapat pada tabel 4.5 berikut :

Tabel 4.4. Kode program pada perhitungan.php untuk melakukan perhitungan metode *Profile Matching*

No.	Kode
14	<code><h4 style="color:#F30; font-weight:bold; margin-bottom:5px; font-size:18px;">Nilai Awal dan Nilai gap (Selisih)</h4></code>
15	<code><table width="100%" border="0" cellspacing="0" cellpadding="4"></code>
16	<code><tr></code>
17	<code><td width="24" rowspan="2" style="vertical-align:middle;">No.</td></code>
18	<code><td width="35" rowspan="2" style="vertical-align:middle;">Kode</td></code>
19	<code><td width="699" rowspan="2" style="vertical-align:middle;">Alternatif [<?php echo \$kasus_objek; ?>]</td></code>
20	<code><?php</code>
21	<code> while(\$d_krit=mysql_fetch_array(\$h_krit)){</code>
22	<code> ?></code>
23	<code><td colspan="<?php echo "\$d_krit[jml]"; ?>" style="background:#CCC; border-top:1px solid #999;"><div style="text-align:center; font-weight:bold; color:#333; background-color:#CCC;"><?php echo "\$d_krit[kriteria]"; ?></div></td></code>
24	<code><?php } ?></code>
25	<code></tr></code>
26	<code><tr></code>
27	<code><?php</code>
28	<code> while(\$d_krit_sub=mysql_fetch_array(\$h_krit_sub)){</code>
29	<code> ?></code>
30	<code><td width="283"><div style="text-align:center;"><?php echo "\$d_krit_sub[kriteria_sub]"; ?></div></td></code>
31	<code><?php } ?></code>
32	<code></tr></code>
33	<code><?php</code>
34	<code> \$no=0;</code>
35	
36	<code> \$q_alter="SELECT id_alternatif, kd_alternatif, nama_lengkap FROM pm_alternatif ORDER BY kd_alternatif ASC";</code>
37	<code> \$h_alter=querydb(\$q_alter);</code>
38	<code> while (\$d_alter=mysql_fetch_array(\$h_alter)){</code>
39	<code> \$no=\$no+1;</code>
40	<code> ?></code>
41	<code><tr></code>
42	<code><td><?php echo"\$no"; ?></td></code>
43	<code><td><?php echo"\$d_alter[kd_alternatif]"; ?></td></code>
44	<code><td><?php echo"\$d_alter[nama_lengkap]"; ?></td></code>

```

45 <?php
46 $q_krit_sub="SELECT a.*, b.kriteria FROM pm kriteria_sub as a,
    pm kriteria as b WHERE a.id_kriteria=b.id_kriteria ORDER BY b.kd_kriteria
    ASC, a.kd_kriteria_sub ASC";
47 $h_krit_sub=querydb($q_krit_sub);
48 while($d_krit_sub=mysql_fetch_array($h_krit_sub)){
49     //Ambil Nilai yang sudah disimpan (lalu tampilkan)
50     $q_nilai="SELECT id_nilai, nilai FROM pm_nilai WHERE
id_kriteria_sub='".$d_krit_sub['id_kriteria_sub']."' and
id_alternatif='".$d_alter['id_alternatif']."'";
51     $h_nilai=querydb($q_nilai);
52     $d_nilai=mysql_fetch_array($h_nilai);
53
54     //1. HITUNG SELISIH (gap)
55
56     //=====
57     $n_gap=0;
58     $n_gap=$d_nilai['nilai']-$d_krit_sub['nilai_ideal'];
59     $cek=mysql_fetch_row(querydb("SELECT COUNT(*) FROM pm_nilai_gap
WHERE id_nilai='".$d_nilai['id_nilai']."'"));
60     if($cek[0]==0){
61         querydb("INSERT INTO pm_nilai_gap (id_nilai, nilai_gap)
VALUES ('".$d_nilai['id_nilai']."', '".$n_gap."'");
62     } else {
63         querydb("UPDATE pm_nilai_gap SET nilai_gap='".$n_gap.'"
WHERE id_nilai='".$d_nilai['id_nilai']."'");
64     }
65
66     //2. HITUNG BOBOT gap
67
68     //=====
69     $q_nilai_gap="SELECT a.id_nilai_gap, a.nilai_gap FROM
pm_nilai_gap as a, pm_nilai as b WHERE a.id_nilai=b.id_nilai AND
a.id_kriteria_sub='".$d_krit_sub['id_kriteria_sub']."'";
70     $h_nilai_gap=querydb($q_nilai_gap);
71     $d_nilai_gap=mysql_fetch_array($h_nilai_gap);
72
73     $q_bobot_gap="SELECT bobot_nilai FROM pm_kriteria_sub_gap WHERE
id_kriteria_sub='".$d_krit_sub['id_kriteria_sub']."' AND
selisih='".$n_gap."'";
74     $h_bobot_gap=querydb($q_bobot_gap);
75     $d_bobot_gap=mysql_fetch_assoc($h_bobot_gap);
76
77     $bobot_gap=0;
78     $bobot_gap=$d_bobot_gap['bobot_nilai'];
79
80     $cek=mysql_fetch_row(querydb("SELECT COUNT(*) FROM
pm_nilai_gap_bobot WHERE
id_nilai_gap='".$d_nilai_gap['id_nilai_gap']."'"));
81     if($cek[0]==0){
82         querydb("INSERT INTO pm_nilai_gap_bobot (id_nilai_gap,
nilai_gap_bobot) VALUES ('".$d_nilai_gap['id_nilai_gap']."',
'".$bobot_gap."'");
83     } else {
84         querydb("UPDATE pm_nilai_gap_bobot SET
nilai_gap_bobot='".$bobot_gap.'" WHERE
id_nilai_gap='".$d_nilai_gap['id_nilai_gap']."'");
85     }
86     ?>
87     <td>
88     <div style="text-align:center;">
89     <?php echo $d_nilai['nilai']; ?>
90     </div>
91     </td>
92 <?php } ?>
93 </tr>
94 <?php } ?>

```

```

94 <tr style="background:#6FB7FF; color:#333; font-weight:bold;">
95 <td colspan="3" style="text-align:center;">Nilai Ideal</td>
96 <?php
97 $q_krit_sub="SELECT a.*, b.kriteria FROM pm_kriteria_sub as a,
pm_kriteria as b WHERE a.id_kriteria=b.id_kriteria ORDER BY b.kd_kriteria
ASC, a.kd_kriteria_sub ASC";
98 $h_krit_sub=querydb($q_krit_sub);
99 while($d_krit_sub=mysql_fetch_array($h_krit_sub)){
100     ?>
101     <td>
102     <div style="text-align:center;">
103     <?php echo $d_krit_sub['nilai_ideal']; ?>
104     </div>
105     </td>
106 <?php } ?>
107 </tr>
108<?php
109$q_alter="SELECT id_alternatif, kd_alternatif, nama_lengkap FROM
pm_alternatif ORDER BY kd_alternatif ASC";
110$h_alter=querydb($q_alter);
111while ($d_alter=mysql_fetch_array($h_alter)){
112$no=$no+1;
113?>
114 <tr>
115 <td><?php echo"$no"; ?></td>
116 <td><?php echo"$d_alter[kd_alternatif]"; ?></td>
117 <td><?php echo"$d_alter[nama_lengkap]"; ?></td>
118 <?php
119 $q_krit_sub="SELECT a.*, b.kriteria FROM pm_kriteria_sub as a,
pm_kriteria as b WHERE a.id_kriteria=b.id_kriteria ORDER BY b.kd_kriteria
ASC, a.kd_kriteria_sub ASC";
120 $h_krit_sub=querydb($q_krit_sub);
121 while($d_krit_sub=mysql_fetch_array($h_krit_sub)){
122     //Ambil Nilai yang sudah disimpan (lalu tampilkan)
123     $q_nilai_gap="SELECT a.id_nilai_gap, a.nilai_gap FROM
pm_nilai_gap as a, pm_nilai as b WHERE a.id_nilai=b.id_nilai AND
b.id_kriteria_sub='$d_krit_sub[id_kriteria_sub]' and
b.id_alternatif='$d_alter[id_alternatif]'";
124     $h_nilai_gap=querydb($q_nilai_gap);
125     $d_nilai_gap=mysql_fetch_array($h_nilai_gap);
126     ?>
127     <td>
128     <div style="text-align:center;">
129     <?php echo $d_nilai_gap['nilai_gap']; ?>
130     </div>
131     </td>
132 <?php } ?>
133 </tr>
134<?php } ?>
135</table>

```

Pertama-tama, untuk menyajikan data nilai asli dari alternatif-alternatif yang ada, maka dilakukan *query* `SELECT` yang ada pada baris no. 50, diikuti dengan perintah *fetch* hasil *query* tersebut agar dapat direpresentasikan pada tabel yang ada pada perintah `mysql_fetch_array($h_nilai)` pada baris 52.

Langkah berikutnya yaitu melakukan operasi pengurangan pada nilai perolehan masing-masing alternatif, dengan nilai ideal yang telah ditentukan per

subkriteria untuk menghitung selisih nilai *gap* yang dinyatakan dalam baris kode 57:

```
$n_gap=$d_nilai['nilai']- $d_krit_sub['nilai_ideal'];.
```

Kode tersebut menggambarkan sekumpulan perolehan nilai setiap alternatif yang telah dikemas dalam *array* dan yang kemudian dikurangi oleh *array* dari nilai ideal untuk mendapatkan pemetaan selisih *gap*.

Langkah berikut yakni menentukan perolehan bobot *gap* dari masing-masing alternatif berdasarkan pemetaan selisih *gap* yang telah didapatkan sebelumnya yang dinyatakan pada kode

```
$q_bobot_gap="SELECT bobot_nilai FROM
pm_kriteria_sub_gap WHERE
id_kriteria_sub='". $d_krit_sub['id_kriteria_sub']. "'
AND selisih='". $n_gap. "'";
```

Pada kode tersebut, dicari variabel *bobot_nilai* yang memiliki nilai selisih sama dengan pemetaan selisih sebelumnya yaitu *n_gap*

Lalu dilakukan perhitungan prosentase *core factor* dan *secondary factor*, dan dilakukan perankingan atas alternatif

```
$nilai_rank=$nilai_rank+($d_nilai_total['nilai_total']*
($d_kriter['bobot_persen']/100));
```

yang kemudian diurutkan secara *descending* (pengurutan nilai dari yang terbesar) pada kode berikut

```
$q_alter="SELECT a.id_alternatif, a.kd_alternatif,
a.nama_lengkap, b.nilai_ranking FROM pm_alternatif as
a, pm_nilai_hasil as b WHERE
a.id_alternatif=b.id_alternatif ORDER BY
b.nilai_ranking DESC";
```

4.4. Pengujian Sistem

Pengujian dilakukan untuk mengevaluasi aplikasi yang telah dibuat. Proses pengujian dilakukan dengan pengujian *White Box* terlebih dahulu kemudian akan dilanjutkan dengan pengujian *Black Box*.

4.4.1 Pengujian *White Box*

Pengujian *White Box* pada sistem ini dengan cara menggambar *Diagram* alir, menghitung kompleksitas siklomatiknya, dan membuat tabel pengujian *test case*.

1. *Listing* program

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

2. *Diagram* alir

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

3. Kompleksitas siklomatik (*cyclomatic complexity*)

Kompleksitas siklomatik merupakan metrik perangkat lunak yang menyediakan ukuran kuantitatif dari kompleksitas logis suatu program. Kompleksitas siklomatik mendefinisikan jumlah jalur independen dalam suatu program. Perhitungan kompleksitas siklomatik menggunakan persamaan berikut:

$$V(G) = E - N + 2$$

Keterangan:

$V(G)$ = Kompleksitas siklomatik

E = Jumlah *edge* (garis)

N = Jumlah *node* (simpul)

4. Pengujian Basis Tes (*Test Case*)

Pada bagian ini diberikan contoh data yang menggambarkan pelaksanaan jalur di basis tes. Data yang dieksekusi dimasukkan ke dalam grafik alir apakah

sudah melewati basis tes yang tersedia. Sistem telah memenuhi syarat kelayakan perangkat lunak jika salah satu jalur yang dieksekusi setidaknya satu kali.

4.4.1.1 Listing Program

Listing Program dari perhitungan metode *Profile Matching* terhadap alternatif. *Listing* program yang akan diuji dapat dilihat pada Tabel 4.6 dibawah ini.

Tabel 4.5. Listing Program Perhitungan Metode *Profile Matching*

No.	Kode
1	<header>
2	<h2>Hasil Perhitungan (Seleksi Metode <i>Profile Matching</i>)</h2>
3	</header>
4	
5	<?php
6	\$q_krit="SELECT COUNT(a.id_kriteria_sub) as jml, b.kriteria, b.id_kriteria, b.cf_persen, b.sf_persen FROM pm_kriteria_sub as a, pm_kriteria as b WHERE a.id_kriteria=b.id_kriteria GROUP BY b.kd_kriteria ORDER BY b.kd_kriteria ASC";
7	\$h_krit=querydb(\$q_krit);
8	\$q_krit_sub="SELECT a.*, b.kriteria FROM pm_kriteria_sub as a, pm_kriteria as b WHERE a.id_kriteria=b.id_kriteria ORDER BY b.kd_kriteria ASC, a.kd_kriteria_sub ASC";
9	\$h_krit_sub=querydb(\$q_krit_sub);
10	\$jmlkkolom=mysql_num_rows(\$h_krit_sub);
11	?>
12	<div class="csstable" >
13	
14	<h4 style="color:#F30; font-weight:bold; margin-bottom:5px; font-size:18px;">Nilai Awal dan Nilai gap (Selisih)</h4>
15	<table width="100%" border="0" cellspacing="0" cellpadding="4">
16	<tr>
17	<td width="24" rowspan="2" style="vertical-align:middle;">No.</td>
18	<td width="35" rowspan="2" style="vertical-align:middle;">Kode</td>
19	<td width="699" rowspan="2" style="vertical-align:middle;">Alternatif [<?php echo \$kases_objek; ?>]</td>
20	<?php
21	while(\$d_krit=mysql_fetch_array(\$h_krit)){
22	?>
23	<td colspan="<?php echo "\$d_krit[jml]"; ?>" style="background:#CCC; border-top:1px solid #999;"><div style="text-align:center; font-weight:bold; color:#333; background-color:#CCC;"><?php echo "\$d_krit[kriteria]"; ?></div></td>
24	<?php } ?>
25	</tr>
26	<tr>
27	<?php
28	while(\$d_krit_sub=mysql_fetch_array(\$h_krit_sub)){
29	?>
30	<td width="283"><div style="text-align:center;"><?php echo "\$d_krit_sub[kriteria_sub]"; ?></div></td>
31	<?php } ?>
32	</tr>
33	<?php
34	\$no=0;
35	
36	\$q_alter="SELECT id_alternatif, kd_alternatif, nama_lengkap FROM pm_alternatif ORDER BY kd_alternatif ASC";
37	\$h_alter=querydb(\$q_alter);
38	while (\$d_alter=mysql_fetch_array(\$h_alter)){
39	\$no=\$no+1;
40	?>

```

41 <tr>
42 <td><?php echo"$no"; ?></td>
43 <td><?php echo"$d_alter[kd_alternatif]"; ?></td>
44 <td><?php echo"$d_alter[nama_lengkap]"; ?></td>
45 <?php
46 $q_krit_sub="SELECT a.*, b.kriteria FROM pm kriteria_sub as a,
pm kriteria as b WHERE a.id_kriteria=b.id_kriteria ORDER BY b.kd_kriteria
ASC, a.kd_kriteria_sub ASC";
47 $h_krit_sub=querydb($q_krit_sub);
48 while($d_krit_sub=mysql_fetch_array($h_krit_sub)){
49 //Ambil Nilai yang sudah disimpan (lalu tampilkan)
50 $q_nilai="SELECT id_nilai, nilai FROM pm_nilai WHERE
id_kriteria_sub='".$d_krit_sub['id_kriteria_sub']."' and
id_alternatif='".$d_alter['id_alternatif']."'";
51 $h_nilai=querydb($q_nilai);
52 $d_nilai=mysql_fetch_array($h_nilai);
53
54 //1. HITUNG SELISIH (gap)
55
56 //=====
57 $n_gap=0;
58 $n_gap=$d_nilai['nilai']-$d_krit_sub['nilai_ideal'];
59 $cek=mysql_fetch_row(querydb("SELECT COUNT(*) FROM pm_nilai_gap
WHERE id_nilai='".$d_nilai['id_nilai']."'"));
60 if($cek[0]==0){
61 querydb("INSERT INTO pm_nilai_gap (id_nilai, nilai_gap)
VALUES ('".$d_nilai['id_nilai']."', '".$n_gap."'");
62 } else {
63 querydb("UPDATE pm_nilai_gap SET nilai_gap='".$n_gap.'"
WHERE id_nilai='".$d_nilai['id_nilai']."'");
64 }
65
66 //2. HITUNG BOBOT gap
67
68 //=====
69 $q_nilai_gap="SELECT a.id_nilai_gap, a.nilai_gap FROM
pm_nilai_gap as a, pm_nilai as b WHERE a.id_nilai=b.id_nilai AND
a.id_kriteria_sub='".$d_krit_sub['id_kriteria_sub']."'";
70 $h_nilai_gap=querydb($q_nilai_gap);
71 $d_nilai_gap=mysql_fetch_array($h_nilai_gap);
72
73 $q_bobot_gap="SELECT bobot_nilai FROM pm_kriteria_sub_gap WHERE
id_kriteria_sub='".$d_krit_sub['id_kriteria_sub']."' AND
selisih='".$n_gap."'";
74 $h_bobot_gap=querydb($q_bobot_gap);
75 $d_bobot_gap=mysql_fetch_assoc($h_bobot_gap);
76
77 $bobot_gap=0;
78 $bobot_gap=$d_bobot_gap['bobot_nilai'];
79
80 $cek=mysql_fetch_row(querydb("SELECT COUNT(*) FROM
pm_nilai_gap_bobot WHERE
id_nilai_gap='".$d_nilai_gap['id_nilai_gap']."'"));
81 if($cek[0]==0){
82 querydb("INSERT INTO pm_nilai_gap_bobot (id_nilai_gap,
nilai_gap_bobot) VALUES ('".$d_nilai_gap['id_nilai_gap']."',
'".$bobot_gap."'");
83 } else {
84 querydb("UPDATE pm_nilai_gap_bobot SET
nilai_gap_bobot='".$bobot_gap.'" WHERE
id_nilai_gap='".$d_nilai_gap['id_nilai_gap']."'");
85 }
86 ?>
87 <td>
88 <div style="text-align:center;">
89 <?php echo $d_nilai['nilai']; ?>
</div>

```

```

90         </td>
91     <?php } ??
92 </tr>
93 <?php } ??
94 <tr style="background:#6FB7FF; color:#333; font-weight:bold;">
95     <td colspan="3" style="text-align:center;">Nilai Ideal</td>
96     <?php
97     $q_krit_sub="SELECT a.*, b.kriteria FROM pm_kriteria_sub as a,
pm_kriteria as b WHERE a.id_kriteria=b.id_kriteria ORDER BY b.kd_kriteria
ASC, a.kd_kriteria_sub ASC";
98     $h_krit_sub=querydb($q_krit_sub);
99     while($d_krit_sub=mysql_fetch_array($h_krit_sub)){
100         ?>
101         <td>
102         <div style="text-align:center;">
103         <?php echo $d_krit_sub['nilai_ideal']; ?>
104         </div>
105         </td>
106     <?php } ??
107 </tr>
108<?php
109$q_alter="SELECT id_alternatif, kd_alternatif, nama_lengkap FROM
pm_alternatif ORDER BY kd_alternatif ASC";
110$h_alter=querydb($q_alter);
111while($d_alter=mysql_fetch_array($h_alter)){
112$no=$no+1;
113?>
114 <tr>
115     <td><?php echo"$no"; ?></td>
116     <td><?php echo"$d_alter[kd_alternatif]"; ?></td>
117     <td><?php echo"$d_alter[nama_lengkap]"; ?></td>
118     <?php
119     $q_krit_sub="SELECT a.*, b.kriteria FROM pm_kriteria_sub as a,
pm_kriteria as b WHERE a.id_kriteria=b.id_kriteria ORDER BY b.kd_kriteria
ASC, a.kd_kriteria_sub ASC";
120     $h_krit_sub=querydb($q_krit_sub);
121     while($d_krit_sub=mysql_fetch_array($h_krit_sub)){
122         //Ambil Nilai yang sudah disimpan (lalu tampilkan)
123         $q_nilai_gap="SELECT a.id_nilai_gap, a.nilai_gap FROM
pm_nilai_gap as a, pm_nilai as b WHERE a.id_nilai=b.id_nilai AND
b.id_kriteria_sub='$d_krit_sub[id_kriteria_sub]' and
b.id_alternatif='$d_alter[id_alternatif]'";
124         $h_nilai_gap=querydb($q_nilai_gap);
125         $d_nilai_gap=mysql_fetch_array($h_nilai_gap);
126         ?>
127         <td>
128         <div style="text-align:center;">
129         <?php echo $d_nilai_gap['nilai_gap']; ?>
130         </div>
131         </td>
132     <?php } ??
133 </tr>
134<?php } ??
135</table>
136
137<?php
138$h_krit=querydb($q_krit);
139$h_krit_sub=querydb($q_krit_sub);
140?>
141<h4 style="color:#F30; font-weight:bold; margin-bottom:5px; font-size:18px;
margin-top:15px;">Nilai Bobot gap</h4>
142<table width="100%" border="0" cellpadding="4">
143 <tr>
144     <td width="24" rowspan="2" style="vertical-align:middle;">No.</td>
145     <td width="35" rowspan="2" style="vertical-align:middle;">Kode</td>
146     <td width="699" rowspan="2" style="vertical-align:middle;">Alternatif
[<?php echo $kasus_objek; ?>]</td>
147     <?php
148         while($d_krit=mysql_fetch_array($h_krit)){
149         ?>

```

```

150 <td colspan="<?php echo "$d_krit[jml]"; ?>" style="background:#CCC; border-
top:1px solid #999;"><div style="text-align:center; font-weight:bold;
color:#333; background-color:#CCC;"><?php echo "$d_krit[kriteria]";
?></div></td>
151 <?php } ?>
152 </tr>
153 <tr>
154 <?php
155     while($d_krit_sub=mysql_fetch_array($h_krit_sub)){
156     ?>
157         <td width="283"><div style="text-align:center;"><?php echo
"$d_krit_sub[kriteria_sub]"; ?></div></td>
158 <?php } ?>
159 </tr>
160<?php
161$no=0;
162$q_alter="SELECT id_alternatif, kd_alternatif, nama_lengkap FROM pm_alternatif
ORDER BY kd_alternatif ASC";
163$h_alter=querydb($q_alter);
164while ($d_alter=mysql_fetch_array($h_alter)){
165$no=$no+1;
166?>
167 <tr>
168 <td><?php echo"$no"; ?></td>
169 <td><?php echo"$d_alter[kd_alternatif]"; ?></td>
170 <td><?php echo"$d_alter[nama_lengkap]"; ?></td>
171 <?php
172     $q_krit_sub="SELECT a.*, b.kriteria FROM pm_kriteria_sub as a, pm_kriteria
as b WHERE a.id_kriteria=b.id_kriteria ORDER BY b.kd_kriteria ASC,
a.kd_kriteria_sub ASC";
173     $h_krit_sub=querydb($q_krit_sub);
174     while($d_krit_sub=mysql_fetch_array($h_krit_sub)){
175         //Ambil Nilai yang sudah disimpan (lalu tampilkan)
176         $q_nilai_gap_bobot="SELECT a.id_nilai_gap_bobot,
a.nilai_gap_bobot FROM pm_nilai_gap_bobot as a, pm_nilai_gap as b, pm_nilai as
c WHERE a.id_nilai_gap=b.id_nilai_gap AND b.id_nilai=c.id_nilai AND
c.id_kriteria_sub='$d_krit_sub[id_kriteria_sub]' and
c.id_alternatif='$d_alter[id_alternatif]";
177         $h_nilai_gap_bobot=querydb($q_nilai_gap_bobot);
178         $d_nilai_gap_bobot=mysql_fetch_array($h_nilai_gap_bobot);
179         ?>
180         <td>
181         <div style="text-align:center;">
182         <?php echo $d_nilai_gap_bobot['nilai_gap_bobot']; ?>
183         </div>
184         </td>
185     <?php } ?>
186 </tr>
187<?php } ?>
188</table>
189
190<?php
191$h_krit=querydb($q_krit);
192while($d_krit=mysql_fetch_array($h_krit)){
193$q_krit_sub="SELECT * FROM pm_kriteria_sub WHERE
id_kriteria='.$d_krit['id_kriteria'].' ORDER BY kd_kriteria_sub ASC";
194$h_krit_sub=querydb($q_krit_sub);
195     $q_alter="SELECT id_alternatif, kd_alternatif, nama_lengkap FROM
pm_alternatif ORDER BY kd_alternatif ASC";
196     $h_alter=querydb($q_alter);
197     while ($d_alter=mysql_fetch_array($h_alter)){
198         $no=$no+1;
199         $jml_core=0; $jml_seco=0; //Untuk menyimpan berapa sub kriteria
core dan secondary
200         $tot_core=0; $tot_seco=0; //Untuk menyimpan total nilai bobot core
dan secondary
201         $NCF=0; $NSF=0; $N_TOT=0; //Untuk menyimpan nilai NCF, NSF dan
Nilai Total
202

```

```

203             $q_krit_sub="SELECT * FROM pm_kriteria_sub WHERE
id_kriteria=".$d_krit['id_kriteria']."' ORDER BY kd_kriteria_sub ASC";
204             $h_krit_sub=querydb($q_krit_sub);
205             while($d_krit_sub=mysql_fetch_array($h_krit_sub)){
206
207                 //Ambil Nilai yang sudah disimpan (lalu tampilkan)
208                 $q_nilai_gap_bobot="SELECT a.id_nilai_gap_bobot,
a.nilai_gap_bobot FROM pm_nilai_gap_bobot as a, pm_nilai_gap as b, pm_nilai as c
WHERE a.id_nilai_gap=b.id_nilai_gap AND b.id_nilai=c.id_nilai AND
c.id_kriteria_sub='$d_krit_sub[id_kriteria_sub]' and
c.id_alternatif='$d_alter[id_alternatif]'";
209                 $h_nilai_gap_bobot=querydb($q_nilai_gap_bobot);
210                 $d_nilai_gap_bobot=mysql_fetch_array($h_nilai_gap_bobot);
211
212                 //3. HITUNG NILAI CF dan SF
213
214                 //=====
=====
214                 if($d_krit_sub['jenis']==1) { $jml_core=$jml_core+1;
$tot_core=$tot_core+$d_nilai_gap_bobot['nilai_gap_bobot']; }
215                 elseif($d_krit_sub['jenis']==2) { $jml_seco=$jml_seco+1;
$tot_seco=$tot_seco+$d_nilai_gap_bobot['nilai_gap_bobot']; }
216                 }
217
218                 //=====
=====
219                 //Hasil NCF, NSF dan Nilai Total
220                 if($jml_core==0 && $jml_seco==0) { $NCF=0; $NSF=0; $N_TOT=0; }
221                 elseif($jml_core==1 && $jml_seco==0) { $NCF=0; $NSF=0;
$N_TOT=$tot_core; }
222                 elseif($jml_core==0 && $jml_seco==1) { $NCF=0; $NSF=0;
$N_TOT=$tot_seco; }
223                 elseif($jml_core>1 && $jml_seco==0) { $NCF=$tot_core/$jml_core;
$NSF=0; $N_TOT=$NCF; }
224                 elseif($jml_core==0 && $jml_seco>1) { $NCF=0;
$NSF=$tot_seco/$jml_seco; $N_TOT=$NSF; }
225                 elseif($jml_core>0 && $jml_seco>0) { $NCF=$tot_core/$jml_core;
$NSF=$tot_seco/$jml_seco;
$N_TOT=((($d_krit['cf_persen']/100)*$NCF)+(($d_krit['sf_persen']/100)*$NSF));
}
226
227                 //Simpan hasilnya
228                 $cek=mysql_fetch_row(querydb("SELECT COUNT(*) FROM pm_nilai_cf_sf
WHERE id_kriteria=".$d_krit['id_kriteria']."' AND
id_alternatif=".$d_alter['id_alternatif']."'"));
229                 if($cek[0]==0) {
230                     querydb("INSERT INTO pm_nilai_cf_sf (id_alternatif,
id_kriteria, nilai_cf, nilai_sf, nilai_total) VALUES
('.$d_alter['id_alternatif']",'.$d_krit['id_kriteria']",'.$NSF',
'$NSF', '$N_TOT')");
231                 } else {
232                     querydb("UPDATE pm_nilai_cf_sf SET nilai_cf='$NCF',
nilai_sf='$NSF', nilai_total='$N_TOT'
WHERE id_kriteria=".$d_krit['id_kriteria']."' AND
id_alternatif=".$d_alter['id_alternatif']."'"));
233                 }
234             }
235         }
236?>
237
238<h4 style="color:#F30; font-weight:bold; margin-bottom:5px; font-size:18px;
margin-top:15px;">Nilai CF dan SF dan Nilai Total</h4>
239<div style="border:1px solid #F60; padding:15px;">
240<?php
241$h_krit=querydb($q_krit);
242while($d_krit=mysql_fetch_array($h_krit)){
243$q_krit_sub="SELECT * FROM pm_kriteria_sub WHERE
id_kriteria=".$d_krit['id_kriteria']."' ORDER BY kd_kriteria_sub ASC";
244$h_krit_sub=querydb($q_krit_sub);

```

```

245??
246 <table width="100%" border="0" cellspacing="0" cellpadding="4">
247 <tr>
248 <td width="24" rowspan="2" style="vertical-align:middle;">No.</td>
249 <td width="35" rowspan="2" style="vertical-align:middle;">Kode</td>
250 <td width="675" rowspan="2" style="vertical-align:middle;">Alternatif
[<?php echo $kasus_objek; ?>]</td>
251 <td colspan="<?php echo "$d_krit[jml]"; ?>" style="background:#CCC;
border-top:1px solid #999;"><div style="text-align:center; font-weight:bold;
color:#333; background-color:#CCC;"><?php echo "$d_krit[kriteria]";
?></div></td>
252 <td rowspan="2" style="vertical-align:middle;">NCF</td>
253 <td rowspan="2" style="vertical-align:middle;">NSF</td>
254 <td rowspan="2" style="vertical-align:middle;">Ntot</td>
255 </tr>
256 <tr>
257 <?php
258 while($d_krit_sub=mysql_fetch_array($h_krit_sub)) {
259 ?>
260 <td width="275"><div style="text-align:center;"><?php echo
"$d_krit_sub[kriteria_sub]"; ?></div></td>
261 <?php } ?>
262 </tr>
263 <?php
264 $no=0;
265 $q_alter="SELECT a.id_alternatif, a.kd_alternatif, a.nama_lengkap,
b.nilai_cf, b.nilai_sf, b.nilai_total
266 FROM pm_alternatif as a, pm_nilai_cf_sf as b
267 WHERE a.id_alternatif=b.id_alternatif AND
b.id_kriteria=".$d_krit['id_kriteria'].'"
268 ORDER BY a.kd_alternatif ASC";
269 $h_alter=querydb($q_alter);
270 while ($d_alter=mysql_fetch_array($h_alter)){
271 $no=$no+1;
272 ?>
273 <tr>
274 <td><?php echo"$no"; ?></td>
275 <td><?php echo"$d_alter[kd_alternatif]"; ?></td>
276 <td><?php echo"$d_alter[nama_lengkap]"; ?></td>
277 <?php
278 $q_krit_sub="SELECT * FROM pm_kriteria_sub WHERE
id_kriteria=".$d_krit['id_kriteria'].'" ORDER BY kd_kriteria_sub ASC";
279 $h_krit_sub=querydb($q_krit_sub);
280 while($d_krit_sub=mysql_fetch_array($h_krit_sub)) {
281
282 //Ambil Nilai yang sudah disimpan (lalu tampilkan)
283 $q_nilai_gap_bobot="SELECT a.id_nilai_gap_bobot,
a.nilai_gap_bobot FROM pm_nilai_gap_bobot as a, pm_nilai_gap as b, pm_nilai as
c WHERE a.id_nilai_gap=b.id_nilai_gap AND b.id_nilai=c.id_nilai AND
c.id_kriteria_sub='$d_krit_sub[id_kriteria_sub]' and
c.id_alternatif='$d_alter[id_alternatif]'" ;
284 $h_nilai_gap_bobot=querydb($q_nilai_gap_bobot);
285 $d_nilai_gap_bobot=mysql_fetch_array($h_nilai_gap_bobot);
286
287 ?>
288 <td>
289 <div style="text-align:center;">
290 <?php echo $d_nilai_gap_bobot['nilai_gap_bobot']; ?>
291 </div>
292 </td>
293 <?php } ?>
294 <td>
295 <div style="text-align:center;">
296 <?php if($d_alter['nilai_cf']==0) { echo "-"; } else { echo
number_format($d_alter['nilai_cf'],2,',','.' ); } ?>
297 </div>
298 </td>
299 <td>
300 <div style="text-align:center;">

```

```

301         <?php if($d_alter['nilai_sf']==0) { echo "-"; } else { echo
number_format($d_alter['nilai_sf'],2,',','.'); } ?>
302     </div>
303 </td>
304 <td>
305     <div style="text-align:center; font-weight:bold; color:#06F;">
306     <?php echo number_format($d_alter['nilai_total'],2,',','.'); ?>
307     </div>
308 </td>
309 </tr>
310 <?php } ?>
311 </table>
312<?php } ?>
313</div>
314
315
316<?php
317//4. HITUNG NILAI RANKING
318//=====
319$no=0;
320$q_alter="SELECT a.id_alternatif, a.kd_alternatif, a.nama_lengkap FROM
pm_alternatif as a
321 WHERE a.id_alternatif IN (SELECT DISTINCT(id_alternatif) FROM
pm_nilai_cf_sf) ORDER BY a.kd_alternatif ASC";
322$h_alter=querydb($q_alter);
323while ($d_alter=mysql_fetch_array($h_alter)){
324
325     $nilai_rank=0; //Untuk menyimpan nilai hasil perhitungan rank
326     $q_kriter="SELECT id_kriteria, bobot_persen FROM pm_kriteria
327 WHERE (id_kriteria IN (SELECT DISTINCT(id_kriteria) FROM
pm_alternatif as a, pm_nilai_cf_sf as b
328 WHERE a.id_alternatif=b.id_alternatif))
329 ORDER BY kd_kriteria ASC";
330     $h_kriter=querydb($q_kriter);
331     while($d_kriter=mysql_fetch_array($h_kriter)){
332
333         //Ambil Nilai yang sudah disimpan (lalu tampilkan)
334         $q_nilai_total="SELECT nilai_total FROM pm_nilai_cf_sf WHERE
id_alternatif='".$d_alter['id_alternatif']."' AND
id_kriteria='".$d_kriter['id_kriteria']."'";
335         $h_nilai_total=querydb($q_nilai_total);
336         $d_nilai_total=mysql_fetch_array($h_nilai_total);
337
338$nilai_rank=$nilai_rank+($d_nilai_total['nilai_total']*( $d_kriter['bobot_pers
en']/100));
339     }
340     //Simpan hasilnya
341 $cek=mysql_fetch_row(querydb("SELECT COUNT(*) FROM pm_nilai_hasil WHERE
id_alternatif='".$d_alter['id_alternatif']."'"));
342     if($cek[0]==0) {
343         querydb("INSERT INTO pm_nilai_hasil (id_alternatif, nilai_ranking) VALUES
('".$d_alter['id_alternatif']."', '$nilai_rank')");
344     } else {
345         querydb("UPDATE pm_nilai_hasil SET nilai_ranking='$nilai_rank' WHERE
id_alternatif='".$d_alter['id_alternatif']."'");
346     }
347
348}
349?>
350
351<h4 style="color:#F30; font-weight:bold; margin-bottom:5px; font-size:18px;
margin-top:15px;">Nilai Ranking</h4>
352<table width="100%" border="0" cellspacing="0" cellpadding="4">
353 <tr>
354     <td width="24">No.</td>
355     <td width="35">Kode</td>
356     <td width="474">Alternatif [<?php echo $kasus_objek; ?>]</td>
357 <?php
358         $h_krit=querydb($q_krit);
359         while($d_krit=mysql_fetch_array($h_krit)){

```

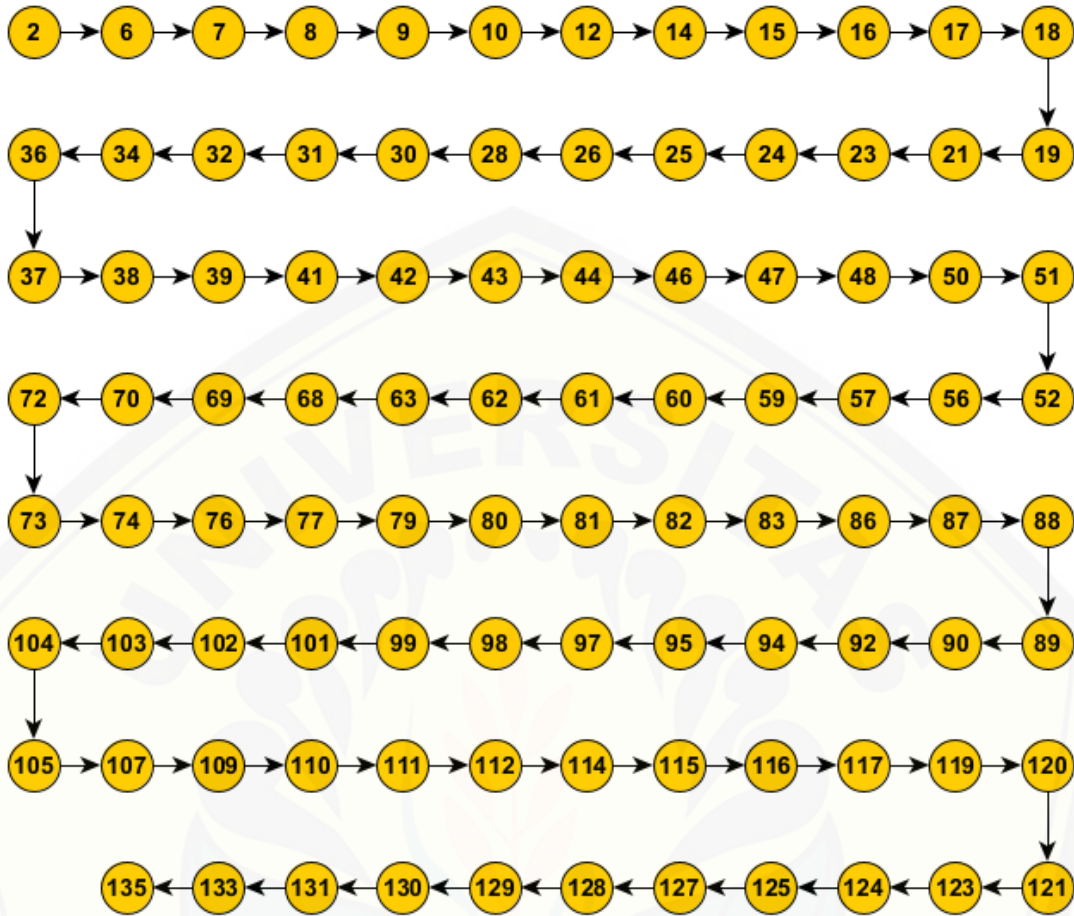
```

360     ?>
361     <td width="280" style="background:#CCC; border-top:1px solid #999;"><div
        style="text-align:center; font-weight:bold; color:#333; background-
        color:#CCC;">Ntot<sub><?php echo "$d_krit[kriteria]"; ?></sub></div></td>
362     <?php } ?>
363     <td width="220">Nilai Ranking</td>
364 </tr>
365<?php
366$no=0;
367$q_alter="SELECT      a.id_alternatif,      a.kd_alternatif,      a.nama_lengkap,
        b.nilai_ranking FROM pm_alternatif as a, pm_nilai_hasil as b WHERE
        a.id_alternatif=b.id_alternatif ORDER BY b.nilai_ranking DESC";
368$h_alter=querydb($q_alter);
369while ($d_alter=mysql_fetch_array($h_alter)){
370$no=$no+1;
371?>
372 <tr>
373     <td><?php echo"$no"; ?></td>
374     <td><?php echo"$d_alter[kd_alternatif]"; ?></td>
375     <td><?php echo"$d_alter[nama_lengkap]"; ?></td>
376     <?php
377         $q_krit="SELECT id_kriteria, kriteria FROM pm_kriteria
378                     WHERE          (id_kriteria      IN          (SELECT
DISTINCT(id_kriteria) FROM pm_alternatif as a, pm_nilai_cf_sf as b
379                     WHERE
        a.id_alternatif=b.id_alternatif))
380                     ORDER BY kd_kriteria ASC";
381         $h_krit=querydb($q_krit);
382         while($d_krit=mysql_fetch_array($h_krit)){
383             //Ambil Nilai yang sudah disimpan (lalu tampilkan)
384             $q_nilai_total="SELECT nilai_total FROM pm_nilai_cf_sf WHERE
        id_kriteria='".$d_krit['id_kriteria']."' AND
        id_alternatif='".$d_alter['id_alternatif']."'";
385             $h_nilai_total=querydb($q_nilai_total);
386             $d_nilai_total=mysql_fetch_array($h_nilai_total);
387             ?>
388             <td>
389                 <div style="text-align:center;">
390                     <?php
391                         number_format($d_nilai_total['nilai_total'],2,',','.') ?>
392                     </div>
393                 </td>
394             <td><div style="text-align:center; font-size:16px; color:#06F; font-
        weight:bold;"><?php echo number_format($d_alter['nilai_ranking'],3,',','.') ?>
        ?></div></td>
395     </tr>
396<?php } ?>
397</table>
398
399</div>

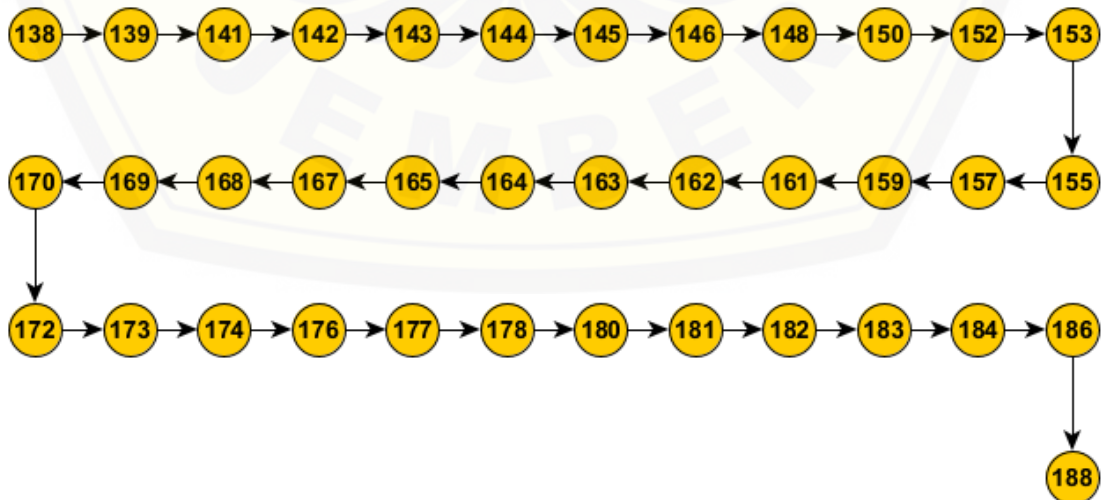
```

4.4.1.2 Diagram Alir

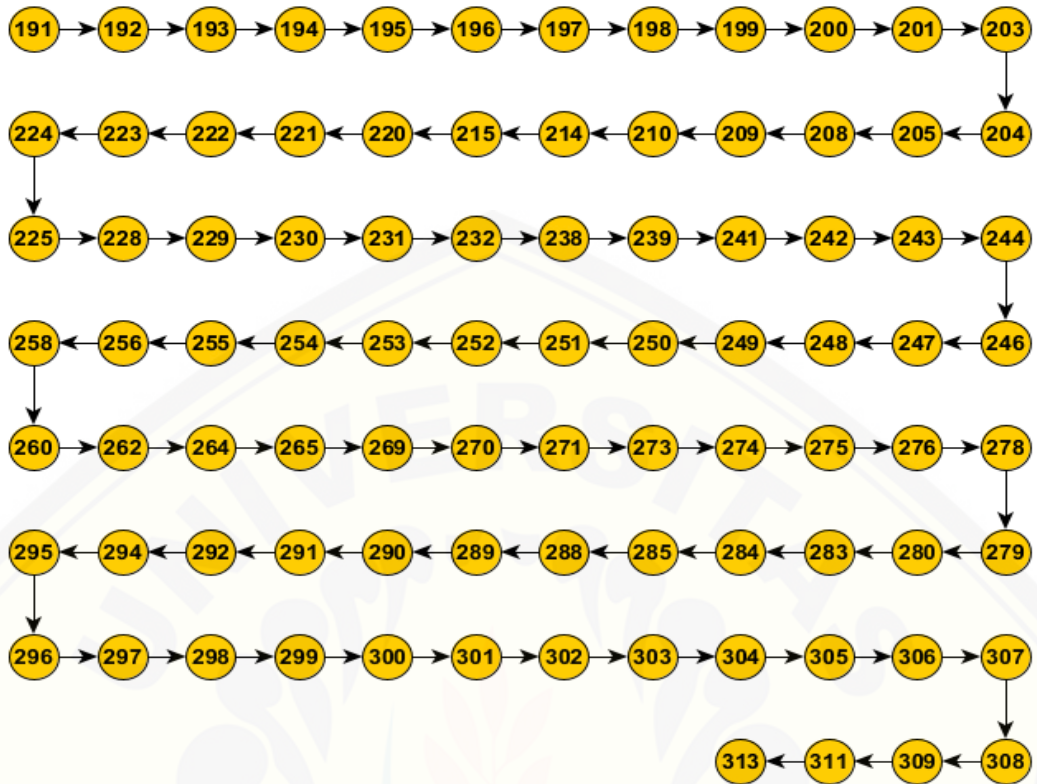
Pengujian *Diagram Alir* pada perhitungan metode *Profile Matching* ini dibagi menjadi 4 (empat) bagian, yakni *Diagram Alir* pengolahan Nilai Awal dan Nilai selisih *gap*, Pemetaan Nilai Bobot *gap*, Perhitungan Nilai CF, SF, dan Nilai Total, Perankingan Siswa, serta yang terakhir adalah Daftar Siswa Terpilih berdasarkan ranking dan kuota. Yang kemudian masing-masing *Diagram Alir* tersebut disajikan pada gambar 4.7, 4.8, 4.9, 4.10, dan 4.11.



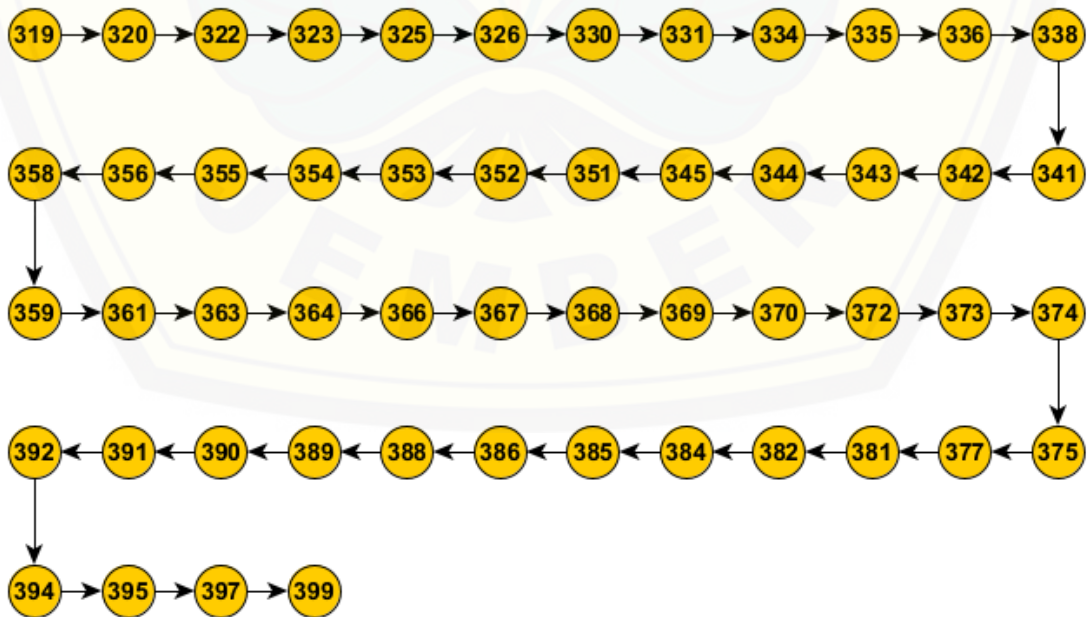
Gambar 4.5. Diagram Alir olah Nilai Awal dan Nilai selisih gap



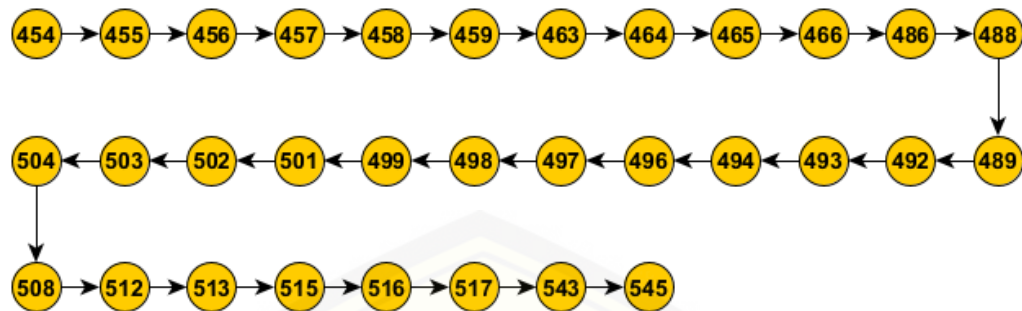
Gambar 4.6. Diagram Alir Pemetaan Nilai Bobot gap



Gambar 4.7. Diagram Alir Perhitungan Nilai CF, SF, dan Nilai Total



Gambar 4.8. Diagram Alir Perankingan Alternatif



Gambar 4.9. Diagram Alir Daftar Siswa Terpilih

4.4.1.3 Kompleksitas Siklomatik

Kompleksitas Siklomatik pada pengujian ini dapat dilihat pada persamaan berikut.

1. Pengolahan Nilai Awal dan Nilai Selisih *gap*

$$V(G) = E - N + 2$$

$$V(G) = 94 - 95 + 2 = 1$$

2. Pemetaan Nilai Bobot *gap*

$$V(G) = E - N + 2$$

$$V(G) = 36 - 37 + 2 = 1$$

3. Perhitungan Nilai CF, SF, dan Nilai Total

$$V(G) = E - N + 2$$

$$V(G) = 87 - 88 + 2 = 1$$

4. Perankingan Alternatif

$$V(G) = E - N + 2$$

$$V(G) = 51 - 52 + 2 = 1$$

5. Daftar Siswa Terpilih

$$V(G) = E - N + 2$$

$$V(G) = 31 - 32 + 2 = 1$$

4.4.1.4 Test Case

Test Case untuk masing-masing *Diagram Alir* pada pengujian ini dapat dilihat pada Tabel 4.7, 4.8, 4.9, 4.10, dan 4.11. Pada test case ini, dilakukan pengujian terhadap salah satu alternatif yakni yang bernama Azhar Syaikhul, salah satu peminat OSN TIK Tahun 2016.

Tabel 4.6. Test Case Pengelolaan Nilai Awal dan Nilai Selisih *gap*

Test Case Pengelolaan Nilai Awal dan Nilai Selisih <i>gap</i>	
Kasus	Pengelolaan nilai awal dan nilai selisih <i>gap</i>
Input	- Nilai awal alternatif : [94, 97, 95, 91, 89, 90] - Nilai ideal : [90, 100, 95, 95, 90, 85]
Target Yang Diharapkan	Sistem berhasil melakukan perhitungan dan menampilkan tabel Nilai Awal dan Nilai Selisih <i>gap</i>
Output	Hasil selisih <i>gap</i> alternatif : [4, -3, 0, -4, -1, 5]
Kebenaran Pengujian	Benar
Path/Jalur 1	2 – 6 – 7 – 8 – 9 – 10 – 12 – 14 – 15 – 16 – 17 – 18 – 19 – 21 – 23 – 24 – 25 – 26 – 28 – 30 – 31 – 32 – 34 – 36 – 37 – 38 – 39 – 41 – 42 – 43 – 44 – 46 – 47 – 48 – 50 – 51 – 52 – 56 – 57 – 59 – 60 – 61 – 62 – 63 – 68 – 69 – 70 – 72 – 73 – 74 – 76 – 77 – 79 – 80 – 81 – 82 – 83 – 86 – 87 – 88 – 89 – 90 – 92 – 94 – 95 – 97 – 98 – 99 – 101 – 102 – 103 – 104 – 105 – 107 – 109 – 110 – 111 – 112 – 114 – 115 – 116 – 117 – 119 – 120 – 121 – 123 – 124 – 125 – 127 – 128 – 129 – 130 – 131 – 133 – 135.

Tabel 4.7. Test Case Pemetaan Nilai Bobot *gap*

Test Case Pemetaan Nilai Bobot <i>gap</i>	
Kasus	Pemetaan nilai bobot <i>gap</i>
Input	- Hasil selisih <i>gap</i> alternatif : [4, -3, 0, -4, -1, 5] - Bobot <i>gap</i> : (sesuai hasil analisis)
Target Yang Diharapkan	Sistem berhasil melakukan pemetaan terhadap nilai bobot <i>gap</i> dan menampilkan tabel Nilai Bobot <i>gap</i>
Output	Hasil pembobotan <i>gap</i> alternatif : [100, 97, 100, 96, 99, 100]
Kebenaran Pengujian	Benar
Path/Jalur 1	138 – 139 – 141 – 142 – 143 – 144 – 145 – 146 – 148 – 150 – 152 – 153 – 155 – 157 – 159 – 161 – 162 – 163 – 164 – 165 – 167 – 168 – 169 – 170 – 172 – 173 – 174 – 176 – 177 – 178 – 180 – 181 – 182 – 183 – 184 – 186 – 188.

Tabel 4.8. Test Case Perhitungan Nilai CF, SF, dan Nilai Total

Test Case Perhitungan Nilai CF, SF, dan Nilai Total	
Kasus	Perhitungan nilai CF, SF, dan nilai total.
Input	Hasil pembobotan <i>gap</i> alternatif : [100, 97, 100, 96, 99, 100] Bobot subkriteria. : [CF = 60%, SF = 40%]
Target Yang Diharapkan	Sistem berhasil melakukan perhitungan nilai Core Factor dan Secondary Factor dan nilai total serta menampilkan table hasil perhitungannya
Output	NCF, NSF, dan NTotal alternatif kriteria 1 dan 2 : [100, 97, 98.8], dan [98, 99.5, 98.6].
Kebenaran Pengujian	Benar
Path/Jalur 1	191 – 192 – 193 – 194 – 195 – 196 – 197 – 198 – 199 – 200 – 201 – 203 – 204 – 205 – 208 – 209 – 210 – 214 – 215 – 220 – 221 – 222 – 223 – 224 – 225 – 228 – 229 – 230 – 231 – 232 – 238 – 239 – 241 – 242 – 243 – 244 – 246 – 247 – 248 – 249 – 250 – 251 – 252 – 253 – 254 – 255 – 256 – 258 – 260 – 262 – 264 – 265 – 269 – 270 – 271 – 273 – 274 – 275 – 276 – 278 – 279 – 280 – 283 – 284 – 285 – 288 – 289 – 290 – 291 – 292 – 294 – 295 – 296 – 297 – 298 – 299 – 300 – 301 – 302 – 303 – 304 – 305 – 306 – 307 – 308 – 309 – 311 – 313.

Tabel 4.9. Test Case Perankingan Siswa (Nilai Akhir)

Test Case Perankingan Siswa (Nilai Akhir)	
Kasus	Perankingan siswa
Input	Hasil Ntotal alternatif : [$N_{Utama} = 98.8$, $N_{Tambah} = 98.6$] Bobot kriteria.: [$N_{Utama} = 60\%$, $N_{Tambah} = 40\%$]
Target Yang Diharapkan	Sistem berhasil melakukan perankingan terhadap siswa dan membuat output berupa tabel rekomendasi siswa terbaik
Output	Hasil nilai akhir alternatif : [98.720]
Kebenaran Pengujian	Benar
Path/Jalur 1	319 – 320 – 322 – 323 – 325 – 326 – 330 – 331 – 334 – 335 – 336 – 338 – 341 – 342 – 343 – 344 – 345 – 351 – 352 – 353 – 354 – 355 – 356 – 358 – 359 – 361 – 363 – 364 – 366 – 367 – 368 – 369 – 370 – 372 – 373 – 374 – 375 – 377 – 381 – 382 – 384 – 385 – 386 – 388 – 389 – 390 – 391 – 392 – 394 – 395 – 397 – 399

Tabel 4.10 Test Case Daftar Siswa Terpilih

Test Case Daftar Siswa Terpilih	
Kasus	Daftar siswa terpilih
Input	Hasil nilai akhir alternatif : [98.720] Jumlah Kuota : [3]
Target Yang Diharapkan	Sistem berhasil menampilkan siswa terpilih berdasarkan ranking dan jumlah kuota
Output	Alternatif 1 peringkat 1
Kebenaran Pengujian	Benar
Path/Jalur 1	454 – 455 – 456 – 457 – 458 – 459 – 463 – 464 – 465 – 466 – 486 – 488 – 489 – 492 – 493 – 494 – 496 – 497 – 498 – 499 – 501 – 502 – 503 – 504 – 508 – 512 – 513 – 515 – 516 – 517 – 543 – 545.

4.4.2 Pengujian *Black Box*

Pengujian *Black Box* merupakan metode pengujian perangkat lunak yang memeriksa fungsionalitas dari aplikasi yang berkaitan dengan struktur internal atau kerja. Metode pengujian ini memfokuskan pada keperluan fungsionalitas dari *software*. Pengujian *Black Box* dilakukan untuk mengetahui apakah masukan dan keluaran dari sistem sesuai dengan kebutuhan fungsional atau tidak. Hasil pengujian *Black Box* dapat dilihat pada tabel 4.11.

Tabel 4.11. Pengujian Metode *Black Box*

NO	MENU	FUNGSI	KASUS	HASIL	KET
1	<i>Sign In</i>	Menu ini digunakan user untuk masuk ke sistem	Ketika user menginputkan username dan <i>password</i> yang valid	User berhasil masuk dan sistem merujuk user kepada halaman <i>Home</i>	OK
			Ketika user menginputkan kombinasi username dan <i>password</i> yang tidak valid	Menampilkan pesan “ <i>Username atau Password yang Anda masukkan salah/tidak cocok.</i> ”	OK

2	Kuota	Menu ini digunakan user untuk mengelola data kuota lomba seperti menambah data lomba dan menghapus data lomba	Ketika user menginputkan data kuota lomba dengan benar dan sesuai dengan yang dibutuhkan oleh sistem pada sesi tambah data	Data kuota lomba berhasil ditambahkan ke database dan menampilkan pesan " <i>Data berhasil disimpan</i> "	OK
			Ketika user menghapus data salah satu periode lomba	Sistem menghapus data lomba yang dimaksud dari database dan menampilkan pesan " <i>data berhasil dihapus</i> "	OK
3	Data Siswa	Menu ini digunakan user untuk mengelola data siswa seperti menambah siswa, mengubah data siswa, dan menghapus data siswa	Ketika user menginputkan data siswa dengan benar dan sesuai dengan yg dibutuhkan oleh sistem pada sesi tambah data.	Data siswa berhasil ditambahkan ke database dan menampilkan pesan " <i>Data berhasil disimpan.</i> "	OK
			Ketika user menginputkan data siswa dengan benar dan sesuai dengan yg dibutuhkan oleh sistem pada sesi ubah data.	Data siswa pada database berhasil diubah dan menampilkan pesan " <i>Data berhasil diubah</i> "	OK

			Ketika user menginputkan data siswa yang satu atau lebih field nya belum terisi/kosong	Sistem memberi peringatan langsung pada field yang bersangkutan " <i>data harus diisi?</i> " dan input data siswa tidak dapat diproses	OK
			Ketika user menginputkan data siswa yang satu atau lebih field nya mengandung format pengisian yang salah	Sistem memberi peringatan langsung pada field yang bersangkutan " <i>input salah?</i> " dan input data siswa tidak dapat diproses	OK
			Ketika user menghapus data salah satu siswa	Sistem menghapus data siswa yang dimaksud dari database dan menampilkan pesan " <i>data berhasil dihapus?</i> "	OK
4	Data Nilai Siswa	Menu ini digunakan user untuk mengelola data nilai dari siswa yang dilakukan secara kolektif yakni dapat dilakukan secara bersamaan per siswa dan per subkriteria	Ketika user menginputkan semua data nilai dengan benar dan sesuai dengan yg dibutuhkan oleh sistem	Data pada database berhasil diubah dan menampilkan pesan " <i>data berhasil disimpan?</i> "	OK

			Ketika user menginputkan data nilai dan terdapat field yang kosong / tidak diisi	Data pada field yang kosong tersebut direpresentasikan sebagai nol (0) dan tetap di eksekusi ke database	OK
5	Kriteria	Menu ini digunakan user (terkhusus untuk admin) yang berwenang untuk mengelola data kriteria seperti menambah kriteria, mengubah data kriteria, dan menghapus data kriteria	Ketika user menginputkan data kriteria dengan benar dan sesuai dengan yg dibutuhkan oleh sistem pada sesi tambah data	Data kriteria berhasil ditambahkan ke database dan menampilkan pesan “ <i>Data berhasil disimpan.</i> ”	OK
			Ketika user menginputkan data kriteria dengan benar dan sesuai dengan yg dibutuhkan oleh sistem pada sesi ubah data	Data kriteria pada database berhasil diubah dan menampilkan pesan “ <i>Data berhasil diubah.</i> ”	OK
			Ketika user menginputkan data kriteria yang satu atau lebih field nya belum terisi/kosong	Sistem memberi peringatan langsung pada field yang bersangkutan “ <i>data harus diisi</i> ” dan input data kriteria tidak dapat diproses	OK

			Ketika user menginputkan data kriteria yang satu atau lebih field nya mengandung format pengisian yang salah	Sistem memberi peringatan langsung pada field yang bersangkutan “ <i>input salah</i> ” dan input data kriteria tidak dapat diproses	OK
			Ketika user menginputkan data kriteria yang kode kriterianya telah terpakai	Sistem memberi peringatan “ <i>kode xxx sudah ada</i> ” dan input data kriteria tidak dapat diproses	OK
			Ketika user menghapus data salah satu kriteria	Sistem menghapus data kriteria yang dimaksud dari database dan menampilkan pesan “ <i>data berhasil dihapus</i> ”	OK
6	Subkriteria	Menu ini digunakan user (terkhusus untuk admin) yang berwenang untuk mengelola data subkriteria seperti menambah subkriteria, mengubah data subkriteria, dan menghapus data subkriteria	Ketika user menginputkan data subkriteria dengan benar dan sesuai dengan yg dibutuhkan oleh sistem pada sesi tambah data	Data subkriteria berhasil ditambahkan ke database dan menampilkan pesan “ <i>Data berhasil disimpan.</i> ”	OK

Ketika user menginputkan data subkriteria dengan benar dan sesuai dengan yg dibutuhkan oleh sistem pada sesi ubah data	Data subkriteria pada database berhasil diubah dan menampilkan pesan “ <i>Data berhasil diubah.</i> ”	OK
Ketika user menginputkan data subkriteria yang satu atau lebih field nya belum terisi/kosong	Sistem memberi peringatan langsung pada field yang bersangkutan “ <i>data harus diisi</i> ” dan input data subkriteria tidak dapat diproses	OK
Ketika user menginputkan data subkriteria yang satu atau lebih field nya mengandung format pengisian yang salah	Sistem memberi peringatan langsung pada field yang bersangkutan “ <i>input salah</i> ” dan input data subkriteria tidak dapat diproses	OK
Ketika user menginputkan data subkriteria yang kode subkriterianya telah terpakai	Sistem memberi peringatan “ <i>kode xxx sudah ada</i> ” dan input data subkriteria tidak dapat diproses	OK

			Ketika user menghapus data salah satu subkriteria	Sistem menghapus data subkriteria yang dimaksud dari database dan menampilkan pesan “ <i>data berhasil dihapus</i> ”	OK
7	Bobot <i>gap</i>	Menu ini digunakan user (terkhusus untuk admin) yang berwenang untuk mengelola data selisih nilai bobot <i>gap</i> seperti menambah data bobot <i>gap</i> , mengubah data bobot <i>gap</i> , dan menghapus data bobot <i>gap</i> .	Ketika user menginputkan data bobot <i>gap</i> baru dengan benar dan sesuai dengan yg dibutuhkan oleh sistem pada sesi tambah data	Data bobot <i>gap</i> berhasil ditambahkan ke database dan menampilkan pesan “ <i>Data berhasil disimpan.</i> ”	OK
			Ketika user menginputkan data bobot <i>gap</i> dengan benar dan sesuai dengan yg dibutuhkan oleh sistem pada sesi ubah data	Data bobot <i>gap</i> pada database berhasil diubah dan menampilkan pesan “ <i>Data berhasil diubah.</i> ”	OK
			Ketika user menginputkan data bobot <i>gap</i> yang satu atau lebih field nya belum terisi/kosong	Sistem memberi peringatan langsung pada field yang bersangkutan “ <i>data harus diisi</i> ” dan input data bobot <i>gap</i> tidak dapat diproses	OK

			Ketika user menginputkan data bobot <i>gap</i> yang satu atau lebih field nya mengandung format pengisian yang salah	Sistem memberi peringatan langsung pada field yang bersangkutan “ <i>input salah</i> ” dan input data bobot <i>gap</i> tidak dapat diproses	OK
			Ketika user menghapus data salah satu bobot <i>gap</i>	Sistem menghapus data bobot <i>gap</i> yang dimaksud dari database dan menampilkan pesan “ <i>data berhasil dihapus</i> ”	OK
8	Hasil Perhitungan	Menu ini digunakan user untuk melihat perolehan nilai akhir dari semua siswa yang terdaftar untuk kemudian ditentukan kandidat yang akan dipilih.	Ketika user meng-klik menu Hasil Perhitungan	Sistem menampilkan hasil perhitungan berupa ranking siswa dari yang terbaik sampai yang terburuk	OK
9	Hak Akses Pengguna	Menu ini digunakan user (terkhusus untuk admin) yang berwenang untuk mengelola data pengguna seperti menambah, mengubah, dan menghapus data pengguna.	Ketika user menginputkan data pengguna baru dengan benar dan sesuai dengan yg dibutuhkan oleh sistem pada sesi tambah data	Data pengguna berhasil ditambahkan ke database dan menampilkan pesan “ <i>Data berhasil disimpan.</i> ”	OK

Ketika user menginputkan data pengguna dengan benar dan sesuai dengan yg dibutuhkan oleh sistem pada sesi ubah data	Data pengguna pada database berhasil diubah dan menampilkan pesan “ <i>Data berhasil diubah.</i> ”	OK
Ketika user menginputkan data pengguna yang satu atau lebih field nya belum terisi/kosong	Sistem memberi peringatan langsung pada field yang bersangkutan “ <i>data harus diisi</i> ” dan input data pengguna tidak dapat diproses	OK
Ketika user menginputkan data pengguna yang satu atau lebih field nya mengandung format pengisian yang salah	Sistem memberi peringatan langsung pada field yang bersangkutan “ <i>input salah</i> ” dan input data pengguna tidak dapat diproses	OK
Ketika user menginputkan data pengguna yang username nya telah terpakai	Sistem memberi peringatan “ <i>pengguna xxx sudah ada</i> ” dan input data pengguna tidak dapat diproses	OK
Ketika user menghapus data salah satu pengguna	Sistem menghapus data pengguna yang dimaksud dari database dan menampilkan pesan “ <i>data berhasil dihapus</i> ”	OK

10	Ubah <i>Password</i>	Menu ini digunakan user terautentikasi untuk dapat mengubah <i>password</i> sesuai yang diinginkan	Ketika user menginputkan <i>password</i> lama yang valid, dan <i>password</i> baru dan konfirmasi <i>password</i> baru yang cocok	Sistem merubah <i>password</i> dari user terkait dan menampilkan pesan “ <i>Perubahan berhasil disimpan. Sistem logout.</i> ”. Lalu user otomatis dirujuk pada halaman Login	OK
			Ketika user menginputkan <i>password</i> lama yang tidak valid	Sistem menampilkan pesan “ <i>password lama salah</i> ” dan perubahan <i>password</i> tidak dapat diproses	OK
			Ketika user menginputkan <i>password</i> baru dan konfirmasi <i>password</i> baru yang tidak cocok	Sistem menampilkan pesan “ <i>konfirmasi password baru tidak cocok</i> ” dan perubahan <i>password</i> tidak dapat diproses	OK
11	<i>Sign Out</i>	Menu ini digunakan user untuk keluar dari sistem	Ketika user meng-klik menu <i>Sign Out</i>	Sistem mengakhiri sesi login dan otomatis mengarahkan pada halaman <i>Sign In</i>	OK

BAB 6. PENUTUP

Bab ini berisi kesimpulan dan saran bagi peneliti tentang penelitian yang telah dilakukan. Kesimpulan dan saran tersebut diharapkan dapat digunakan sebagai acuan pada penelitian selanjutnya.

6.1. Kesimpulan

Berdasarkan analisis yang telah dilakukan oleh peneliti, dapat diambil kesimpulan sebagai berikut:

1. Perhitungan metode *Profile Matching* adalah merupakan algoritma yang digunakan untuk mengambil siswa yang sesuai dengan nilai ideal dari setiap kriteria penilaian, bukan hanya pembobotan nilai pada kriteria saja.
2. Pembobotan nilai bersifat dinamis karena pada setiap subkriteria yang ada dapat ditentukan nilai ideal dan pada setiap kemungkinan perolehan nilai awal pada siswa dapat dipetakan kepada pembobotan Nilai *gap*.
3. Dari sampel data yang telah dilakukan uji coba, didapati kecocokan sebesar 52.67% oleh sistem penunjang keputusan yang telah dirancang terhadap seleksi konvensional yang dilakukan pada tahun-tahun lomba sebelumnya.

6.2. Saran

Pada hasil implementasi sistem ini, penginputan data nilai siswa yang dapat dilakukan secara kolektif oleh hak akses wali kelas masih belum terklasifikasi. Pada sistem ini penginputan data nilai oleh wali kelas, wali kelas dapat mengelola data nilai siswa lain yang bukan walinya. Pada penelitian selanjutnya, diharapkan untuk menangani kasus ini agar sistem penilaian dapat menjadi lebih baik lagi.

DAFTAR PUSTAKA

- Asfan Muqtadir, Irwan Purdianto. (2013). Sistem Pendukung Keputusan Kenaikan Jabatan Menggunakan Metode *Profile Matching* (Studi kasus di PT. Industri Kemasan Semen Gresik). Tuban: Universitas PGRI Ronggolawe.
- Galuh Pratiwi. (2010). Apa Itu Olimpiade Sains Nasional? Bagaimana Sejarah Pelaksanaannya? <http://www.galuhpratiwi.my.id/2010/11/apa-itu-olimpiade-sains-nasional.html>. Diakses 1 Mei 2018.
- Kusrini. (2007). Konsep dan Aplikasi Sistem Pendukung Keputusan. Yogyakarta: CV. Andi Offset.
- Panji Setio. (2013). Perancangan Sistem Pendukung Keputusan Untuk Pemilihan Siswa Peserta Olimpiade Sains Nasional (OSN) Dengan Metode Simple Additive Weighting (SAW) Di SMA Negeri 2 Bondowoso. Jember: Universitas Muhamadiyah Jember.
- Pressman, Roger S. (2010). Software Engineering : A Practitioner's Approach 7th ed. New York: McGraw-Hill.
- Rachma. (2003). Pembuatan Aplikasi Sistem Pendukung Keputusan Untuk Proses Kenaikan Jabatan dan Perencanaan Karir Pada PT.X. Jurnal Informatika, 4(2): 101 - 105.
- Rajali Rasyid. (2016). Harian Kompas Edisi Kamis 8 September 2016. Jakarta: Kompas. (8 September 2016).
- Sprague, R. H. (1993). Decision Support System : Putting Theory Into Practice, Englewood Clifts, N.J., Prentice Hall.
- Syaiful Luthvi Amar. (2016). Sistem Pendukung Keputusan Memilih Madrasah Ibtidayah Terbaik Di Wilayah Kementrian Agama Kota Salatiga Dengan Metode Simple Additive Weighting. Malang: Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Y. Muchsam, Falahah, dan G.I. Saputro. (2011). Penerapan *gap* Analysis Pada Pengembangan Sistem Pendukung Keputusan Penilaian Kinerja Pegawai. Seminar Nasional Aplikasi Teknologi Informasi, 96.



LAMPIRAN

LAMPIRAN A. USE CASE SKENARIO

A1. Skenario *Sign In*Tabel 1. Skenario *Sign In* (Admin)

Name	<i>Sign In</i>	
Participating Actor	Admin	
Entry Condition	Admin berhasil menampilkan halaman <i>Sign In</i>	
Exit Condition	Admin berhasil melakukan <i>Sign In</i> dan masuk ke sistem	
Event Flow	<ol style="list-style-type: none"> Admin membuka website Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> Admin klik tombol "<i>Sign In</i>" 	
Skenario Utama " <i>Sign In</i> "		
Aksi Aktor		Reaksi Sistem
1. Membuka website SICALIM		2. Menampilkan halaman <i>Sign In</i> untuk masuk ke sistem
3. Memasukkan username dan <i>password</i> dengan benar dan meng-klik tombol " <i>Sign In</i> "		4. Menampilkan halaman <i>Home</i>
Skenario Alternatif " <i>Username dan Password Salah</i> "		
Aksi Aktor		Reaksi Sistem
3.a Memasukkan username dan <i>password</i> yang salah dan meng-klik tombol " <i>Sign In</i> "		4.a Menampilkan pesan " <i>Username atau Password yang Anda masukkan salah/tidak cocok.</i> " Dan tidak dapat masuk ke sistem

Tabel 2. Skenario *Sign In* (Wali Kelas)

Name	<i>Sign In</i>	
Participating Actor	Wali Kelas	
Entry Condition	Wali Kelas berhasil menampilkan halaman <i>Sign In</i>	
Exit Condition	Wali Kelas berhasil melakukan <i>Sign In</i> dan masuk ke sistem	
Event Flow	<ol style="list-style-type: none"> Wali Kelas membuka website Wali Kelas memasukkan username dan <i>password</i> untuk <i>Sign In</i> Wali Kelas klik tombol "<i>Sign In</i>" 	
Skenario Utama " <i>Sign In</i> "		
Aksi Aktor		Reaksi Sistem
1. Membuka website SICALIM		2. Menampilkan halaman <i>Sign In</i> untuk masuk ke sistem
3. Memasukkan username dan <i>password</i> dengan benar dan meng-klik tombol " <i>Sign In</i> "		4. Menampilkan halaman <i>Home</i>
Skenario Alternatif " <i>Username dan Password Salah</i> "		
Aksi Aktor		Reaksi Sistem
3.a Memasukkan username dan <i>password</i> yang salah dan meng-klik tombol " <i>Sign In</i> "		4.a Menampilkan pesan " <i>Username atau Password yang Anda masukkan salah/tidak cocok.</i> " Dan tidak dapat masuk ke sistem

A2. Skenario Ubah *Password*Tabel 3. Skenario Ubah *Password* (Admin)

Name	Ubah <i>Password</i>
Participating Actor	Admin
Entry Condition	Admin berhasil melakukan <i>Sign In</i>
Exit Condition	Admin berhasil melakukan ubah <i>password</i>
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol “<i>Sign In</i>” 4. Admin menuju menu Ubah <i>Password</i>
Skenario Utama “Ubah <i>Password</i>”	
Aksi Aktor	Reaksi Sistem
1. Memilih menu “Ubah <i>Password</i> ”	
	2. Menampilkan form Ubah <i>Password</i> yang berisi field berupa <i>password</i> lama, <i>password</i> baru, dan konfirmasi <i>password</i> baru
3. Menginputkan <i>password</i> lama, <i>password</i> baru, dan konfirmasi <i>password</i> baru dengan benar dan meng-klik tombol “Simpan Data”	
	4. Menampilkan pesan bahwa perubahan <i>password</i> telah berhasil
	5. Mengarahkan secara otomatis user terautentikasi untuk <i>Sign Out</i>
	6. Menampilkan halaman <i>Sign In</i>
Skenario Alternatif “Menginputkan <i>Password</i> Lama Yang Salah”	
Aksi Aktor	Reaksi Sistem
3.a Menginputkan <i>password</i> lama yang salah, kemudian <i>password</i> baru dan konfirmasi <i>password</i> baru dan meng-klik tombol “Simpan Data”	
	4.a Menampilkan pesan bahwa <i>password</i> lama yang diinputkan salah dan perubahan <i>password</i> tidak berhasil dilakukan
	5.a Menampilkan kembali form ubah <i>password</i>
Skenario Alternatif “Menginputkan <i>Password</i> Baru dan Konfirmasinya Yang Tidak Cocok”	
Aksi Aktor	Reaksi Sistem
3.b Menginputkan <i>password</i> lama dengan benar kemudian menginputkan <i>password</i> baru dan konfirmasi <i>password</i> baru yang tidak sama dan meng-klik tombol “Simpan Data”	
	4.b Menampilkan pesan bahwa konfirmasi <i>password</i> baru tidak cocok dengan <i>password</i> baru yang diinputkan tepat sebelumnya dan perubahan <i>password</i> tidak berhasil dilakukan

Tabel 4. Skenario Ubah *Password* (Wali Kelas)

Name	Ubah <i>Password</i>	
Participating Actor	Wali Kelas	
Entry Condition	Wali Kelas berhasil melakukan <i>Sign In</i>	
Exit Condition	Wali Kelas berhasil melakukan ubah <i>password</i>	
Event Flow	<ol style="list-style-type: none"> 1. Wali Kelas membuka website 2. Wali Kelas memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Wali Kelas klik tombol “<i>Sign In</i>” 4. Wali Kelas menuju menu Ubah <i>Password</i> 	
Skenario Utama “Ubah <i>Password</i>”		
Aksi Aktor		Reaksi Sistem
1. Memilih menu “Ubah <i>Password</i> ”		
		2. Menampilkan form Ubah <i>Password</i> yang berisi field berupa <i>password</i> lama, <i>password</i> baru, dan konfirmasi <i>password</i> baru
3. Menginputkan <i>password</i> lama, <i>password</i> baru, dan konfirmasi <i>password</i> baru dengan benar dan meng-klik tombol “Simpan Data”		
		4. Menampilkan pesan bahwa perubahan <i>password</i> telah berhasil
		5. Mengarahkan secara otomatis user terautentikasi untuk <i>Sign Out</i>
		6. Menampilkan halaman <i>Sign In</i>
Skenario Alternatif “Menginputkan <i>Password</i> Lama Yang Salah”		
Aksi Aktor		Reaksi Sistem
3.a Menginputkan <i>password</i> lama yang salah, kemudian <i>password</i> baru dan konfirmasi <i>password</i> baru dan meng-klik tombol “Simpan Data”		
		4.a Menampilkan pesan bahwa <i>password</i> lama yang diinputkan salah dan perubahan <i>password</i> tidak berhasil dilakukan
		5.a Menampilkan kembali form ubah <i>password</i>
Skenario Alternatif “Menginputkan <i>Password</i> Baru dan Konfirmasinya Yang Tidak Cocok”		
Aksi Aktor		Reaksi Sistem
3.b Menginputkan <i>password</i> lama dengan benar kemudian menginputkan <i>password</i> baru dan konfirmasi <i>password</i> baru yang tidak sama dan meng-klik tombol “Simpan Data”		
		4.b Menampilkan pesan bahwa konfirmasi <i>password</i> baru tidak cocok dengan <i>password</i> baru yang diinputkan tepat sebelumnya dan perubahan <i>password</i> tidak berhasil dilakukan

A3. Skenario Input Kuota Lomba

Tabel 5. Skenario Tambah Data Kuota Lomba (Admin)

Name	Tambah Data Kuota Lomba
Participating Actor	Admin
Entry Condition	Admin berhasil menampilkan form Data Kuota OSN
Exit Condition	Admin berhasil menambahkan data kuota
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Kuota
Skenario Utama "Tambah Data Kuota Lomba"	
Aksi Aktor	Reaksi Sistem
1. Memilih menu "Kuota"	
	2. Menampilkan tabel Data Kuota yang berkolomkan Nomor, Tahun lomba, dan Jumlah Kuota
3. Meng-klik tombol "Tambah Data" (+)	
	4. Menampilkan form Tambah Data Kuota yang berisikan data field yang harus dipenuhi yaitu Tahun Lomba dan Jumlah Kuota
5. Mengisi semua keperluan data yang dibutuhkan oleh sistem dengan benar dan meng-klik tombol "Simpan Data"	
	6. Menampilkan pesan bahwa data kuota berhasil ditambahkan
Skenario Alternatif "Data Yang Diinputkan Tidak Lengkap"	
Aksi Aktor	Reaksi Sistem
5.a. Mengisi keperluan data, dan terdapat satu atau lebih field yang belum diisi lalu meng-klik tombol "Simpan Data"	
	6.a. Menampilkan pesan "data belum diisi" pada field yang masih kosong dan penambahan data kuota tidak dapat diproses
Skenario Alternatif "Data Yang Diinputkan Tidak Sesuai Format"	
Aksi Aktor	Reaksi Sistem
5.b Mengisi keperluan data, dan terdapat satu atau lebih field yang format penulisannya tidak sesuai dengan yang dibutuhkan oleh sistem lalu meng-klik tombol "Simpan Data"	
	6.b Menampilkan pesan "input salah" pada field yang format isinya tidak sesuai dan penambahan data kuota tidak dapat diproses
Skenario Alternatif "Periode Lomba Sudah Ada"	
Aksi Aktor	Reaksi Sistem
5.c Mengisi keperluan data, dan menginputkan periode lomba yang sama dengan periode lomba yang telah ada lalu meng-klik tombol "Simpan Data"	
	6.c Menampilkan pesan "Periode lomba xxxx sudah ada" dan penambahan data kuota lomba tidak dapat diproses

Tabel 6. Skenario Hapus Data Kuota Lomba (Admin)

Name	Hapus Data Kuota Lomba
Participating Actor	Admin
Entry Condition	Admin berhasil menampilkan form Data Kuota OSN
Exit Condition	Admin berhasil menghapus data kuota
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Kuota
Skenario Utama	
"Hapus Data Kuota Lomba"	
Aksi Aktor	Reaksi Sistem
1. Memilih menu "Kuota"	
	2. Menampilkan tabel Data Kuota yang berkolomkan Nomor, Tahun lomba, dan Jumlah Kuota
3. Meng-klik tombol "Hapus Data" pada salah satu data kuota yang hendak dihapus dengan benar dan meng-klik tombol "Simpan Data"	
	4. Menampilkan pertanyaan konfirmasi hapus data kuota
5. Melakukan konfirmasi penghapusan data kuota dengan meng-klik tombol "OK"	
	6. Menampilkan pesan bahwa data kuota telah berhasil dihapus

A4. Skenario Pengelolaan Data Siswa

Tabel 7. Skenario Tambah Data Siswa (Admin)

Name	Tambah Data Siswa	
Participating Actor	Admin	
Entry Condition	Admin berhasil menampilkan form Data Siswa	
Exit Condition	Admin berhasil menambahkan data siswa	
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Data Siswa 	
Skenario Utama "Tambah Data Siswa"		
Aksi Aktor		Reaksi Sistem
1. Memilih menu "Data Siswa"		
		2. Menampilkan tabel Data Siswa yang berkolomkan Nomor, Kelas/absen, Nama, Jenis Kelamin, Tempat/Tgl Lahir, Terdaftar, dan Tahun Lomba
3. Meng-klik tombol "Tambah Data" (+)		
		4. Menampilkan form Tambah Siswa yang berisikan data field yang harus dipenuhi seperti Tahun Lomba, Kelas/absen, Nama, Tempat Lahir, Tgl Lahir, Jenis Kelamin, dan Alamat
5. Mengisi semua keperluan data yang dibutuhkan oleh sistem dengan benar dan meng-klik tombol "Simpan Data"		
		6. Menampilkan pesan bahwa data siswa berhasil ditambahkan
Skenario Alternatif "Data Yang Diinputkan Tidak Lengkap"		
Aksi Aktor		Reaksi Sistem
5.b Mengisi keperluan data, dan terdapat satu atau lebih field yang belum diisi lalu meng-klik tombol "Simpan Data"		
		6.b Menampilkan pesan "data belum diisi" pada field yang masih kosong dan penambahan data siswa tidak dapat diproses
Skenario Alternatif "Data Yang Diinputkan Tidak Sesuai Format"		
Aksi Aktor		Reaksi Sistem
5.b Mengisi keperluan data, dan terdapat satu atau lebih field yang format penulisannya tidak sesuai dengan yang dibutuhkan oleh sistem lalu meng-klik tombol "Simpan Data"		
		6.b Menampilkan pesan "input salah" pada field yang format isian tidak sesuai dan penambahan data siswa tidak dapat diproses

Tabel 8. Skenario Tambah Data Siswa (Wali Kelas)

Name	Tambah Data Siswa	
Participating Actor	Wali Kelas	
Entry Condition	Wali Kelas berhasil menampilkan form Data Siswa	
Exit Condition	Wali Kelas berhasil menambahkan data siswa	
Event Flow	<ol style="list-style-type: none"> 1. Wali Kelas membuka website 2. Wali Kelas memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Wali Kelas klik tombol "<i>Sign In</i>" 4. Wali Kelas menuju menu Data Siswa 	
Skenario Utama "Tambah Data Siswa"		
Aksi Aktor	Reaksi Sistem	
1. Memilih menu "Data Siswa"	2. Menampilkan tabel Data Siswa yang berkolomkan Nomor, Kelas/absen, Nama, Jenis Kelamin, Tempat/Tgl Lahir, Terdaftar, dan Tahun Lomba	
3. Meng-klik tombol "Tambah Data" (+)	4. Menampilkan form Tambah Siswa yang berisikan data field yang harus dipenuhi seperti Tahun Lomba, Kelas/absen, Nama, Tempat Lahir, Tgl Lahir, Jenis Kelamin, dan Alamat	
5. Mengisi semua keperluan data yang dibutuhkan oleh sistem dengan benar dan meng-klik tombol "Simpan Data"	6. Menampilkan pesan bahwa data siswa berhasil ditambahkan	
Skenario Alternatif "Data Yang Diinputkan Tidak Lengkap"		
Aksi Aktor	Reaksi Sistem	
5.a Mengisi keperluan data, dan terdapat satu atau lebih field yang belum diisi lalu meng-klik tombol "Simpan Data"	6.a Menampilkan pesan "data belum diisi" pada field yang masih kosong dan penambahan data siswa tidak dapat diproses	
Skenario Alternatif "Data Yang Diinputkan Tidak Sesuai Format"		
Aksi Aktor	Reaksi Sistem	
5.b Mengisi keperluan data, dan terdapat satu atau lebih field yang format penulisannya tidak sesuai dengan yang dibutuhkan oleh sistem lalu meng-klik tombol "Simpan Data"	6.b Menampilkan pesan "input salah" pada field yang format isiannya tidak sesuai dan penambahan data siswa tidak dapat diproses	

Tabel 9. Skenario Ubah Data Siswa (Admin)

Name	Ubah Data Siswa	
Participating Actor	Admin	
Entry Condition	Admin berhasil menampilkan form Data Siswa	
Exit Condition	Admin berhasil mengubah data dari siswa yang diinginkan	
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Data Siswa 	
Skenario Utama "Ubah Data Siswa"		
Aksi Aktor		Reaksi Sistem
1. Memilih menu "Data Siswa"		
		2. Menampilkan tabel Data Siswa yang berkolomkan Nomor, Kelas/absen, Nama, Jenis Kelamin, Tempat/Tgl Lahir, Terdaftar, dan Tahun Lomba
3. Meng-klik tombol "Ubah Data" pada salah satu data siswa yang hendak diubah		
		4. Menampilkan form Ubah Siswa yang berisikan data field yang harus dipenuhi seperti Tahun Lomba, Kelas/absen, Nama, Tempat Lahir, Tgl Lahir, Jenis Kelamin, dan Alamat
5. Mengubah data siswa sesuai yang diinginkan kedalam sistem dengan benar dan meng-klik tombol "Simpan Data"		
		6. Menampilkan pesan bahwa data siswa berhasil diubah
Skenario Alternatif "Data Yang Diinputkan Tidak Lengkap"		
Aksi Aktor		Reaksi Sistem
5.a Mengubah data siswa sesuai yang diinginkan kedalam sistem, dan terdapat satu atau lebih field yang belum diisi dan meng-klik tombol "Simpan Data"		
		6.a Menampilkan pesan "data belum diisi" pada field yang masih kosong dan perubahan data siswa tidak dapat diproses
Skenario Alternatif "Data Yang Diinputkan Tidak Sesuai Format"		
Aksi Aktor		Reaksi Sistem
5.b Mengubah data siswa sesuai yang diinginkan kedalam sistem, dan terdapat satu atau lebih field yang format penulisannya tidak sesuai dengan yang dibutuhkan oleh sistem dan meng-klik tombol "Simpan Data"		
		6.b Menampilkan pesan "input salah" pada field yang format isinya tidak sesuai dan perubahan data siswa tidak dapat diproses

Tabel 10. Skenario Hapus Data Siswa (Admin)

Name	Hapus Data Siswa	
Participating Actor	Admin	
Entry Condition	Admin berhasil menampilkan form Data Siswa	
Exit Condition	Admin berhasil menghapus salah satu siswa	
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol “<i>Sign In</i>” 4. Admin menuju menu Data Siswa 	
Skenario Utama “Hapus Data Siswa”		
	Aksi Aktor	Reaksi Sistem
	1. Memilih menu “Data Siswa”	
		2. Menampilkan tabel Data Siswa yang berkolomkan Nomor, Kelas/absen, Nama, Jenis Kelamin, Tempat/Tgl Lahir, Terdaftar, dan Tahun Lomba
	3. Meng-klik tombol “Hapus Data” pada salah satu data siswa yang hendak dihapus	
		4. Menampilkan pertanyaan konfirmasi hapus data siswa
	5. Melakukan konfirmasi penghapusan data siswa dengan meng-klik tombol “OK”	
		6. Menampilkan pesan bahwa data siswa telah berhasil dihapus

A5. Skenario Pengelolaan Data Nilai Siswa

Tabel 11. Skenario Pengelolaan Data Nilai Siswa (Admin)

Name	Pengelolaan Data Nilai Siswa	
Participating Actor	Admin	
Entry Condition	Admin berhasil menampilkan form Data Nilai Siswa	
Exit Condition	Admin berhasil mengelola data nilai pada siswa yang diinginkan	
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol “<i>Sign In</i>” 4. Admin menuju menu Data Nilai Siswa 	
Skenario Utama “Pengelolaan Data Nilai Siswa”		
	Aksi Aktor	Reaksi Sistem
	1. Memilih menu “Data Nilai Siswa”	
		2. Menampilkan menu Data Nilai Siswa
	3. Memilih periode lomba yang diinginkan lalu klik “Submit”	
		4. Menampilkan form data nilai siswa per subkriteria penilaian sesuai periode lomba yang dipilih
	5. Menginputkan nilai pada masing-masing subkriteria pada siswa yang diinginkan dan meng-klik tombol “Simpan Data”	
		6. Menampilkan pesan bahwa data nilai siswa telah berhasil disimpan

A6. Skenario Pengelolaan Data Kriteria

Tabel 12. Skenario Tambah Data Kriteria (Admin)

Name	Tambah Data Kriteria	
Participating Actor	Admin	
Entry Condition	Admin berhasil menampilkan form Kriteria	
Exit Condition	Admin berhasil menambahkan kriteria baru kedalam sistem	
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Kriteria 	
Skenario Utama "Tambah Data Kriteria"		
Aksi Aktor		Reaksi Sistem
1. Memilih menu "Kriteria"		
		2. Menampilkan tabel Kriteria yang berkolomkan Nomor, Kode Kriteria, Nama Kriteria, Bobot Kriteria, Prosentase CF, dan Prosentase SF
3. Meng-klik tombol "Tambah Data" (+)		
		4. Menampilkan form Tambah Kriteria yang berisikan data field yang harus dipenuhi seperti Kode Kriteria, Nama Kriteria, Bobot, Persen CF, dan Persen SF.
5. Mengisi semua keperluan data yang dibutuhkan oleh sistem dengan benar dan meng-klik tombol "Simpan Data"		
		6. Menampilkan pesan bahwa data kriteria berhasil ditambahkan
Skenario Alternatif "Data Yang Diinputkan Tidak Lengkap"		
Aksi Aktor		Reaksi Sistem
5.a Mengisi keperluan data, dan terdapat satu atau lebih field yang belum diisi lalu meng-klik tombol "Simpan Data"		
		6.a Menampilkan pesan "data belum diisi" pada field yang masih kosong dan penambahan data kriteria tidak dapat diproses
Skenario Alternatif "Data Yang Diinputkan Tidak Sesuai Format"		
Aksi Aktor		Reaksi Sistem
5.b Mengisi keperluan data, dan terdapat satu atau lebih field yang format penulisannya tidak sesuai dengan yang dibutuhkan oleh sistem lalu meng-klik tombol "Simpan Data"		
		6.b Menampilkan pesan "input salah" pada field yang format isiannya tidak sesuai dan penambahan data kriteria tidak dapat diproses
Skenario Alternatif "Kode Kriteria Telah Terpakai"		
Aksi Aktor		Reaksi Sistem
5.c Mengisi keperluan data, dan menginputkan kode kriteria yang sama dengan kriteria yang telah ada lalu meng-klik tombol "Simpan Data"		
		6.c Menampilkan pesan "Kode xxx sudah ada" dan penambahan data kriteria tidak dapat diproses

Tabel 13. Skenario Ubah Data Kriteria (Admin)

Name	Ubah Data Kriteria	
Participating Actor	Admin	
Entry Condition	Admin berhasil menampilkan form Kriteria	
Exit Condition	Admin berhasil mengubah kriteria yang diinginkan pada sistem	
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Kriteria 	
Skenario Utama "Ubah Data Kriteria"		
	Aksi Aktor	Reaksi Sistem
	1. Memilih menu "Kriteria"	
		2. Menampilkan tabel Kriteria yang berkolomkan Nomor, Kode Kriteria, Nama Kriteria, Bobot Kriteria, Prosentase CF, dan Prosentase SF
	3. Meng-klik tombol "Ubah Data" pada salah satu data kriteria yang hendak diubah	
		4. Menampilkan form Ubah Kriteria yang berisikan data field yang harus dipenuhi seperti Kode Kriteria, Nama Kriteria, Bobot, Persen CF, dan Persen SF
	5. Mengubah data kriteria sesuai yang diinginkan kedalam sistem dengan benar dan meng-klik tombol "Simpan Data"	
		6. Menampilkan pesan bahwa data kriteria berhasil diubah
Skenario Alternatif "Data Yang Diinputkan Tidak Lengkap"		
	Aksi Aktor	Reaksi Sistem
	5.a Mengubah data kriteria sesuai yang diinginkan kedalam sistem, dan terdapat satu atau lebih field yang belum diisi dan meng-klik tombol "Simpan Data"	
		6.a Menampilkan pesan "data belum diisi" pada field yang masih kosong dan pengubahan data kriteria tidak dapat diproses
Skenario Alternatif "Data Yang Diinputkan Tidak Sesuai Format"		
	Aksi Aktor	Reaksi Sistem
	5.b Mengubah data kriteria sesuai yang diinginkan kedalam sistem, dan terdapat satu atau lebih field yang format penulisannya tidak sesuai dengan yang dibutuhkan oleh sistem dan meng-klik tombol "Simpan Data"	
		6.b Menampilkan pesan "input salah" pada field yang format isiannya tidak sesuai dan pengubahan data kriteria tidak dapat diproses
Skenario Alternatif "Kode Kriteria Telah Terpakai"		
	Aksi Aktor	Reaksi Sistem
	5.c Mengubah data kriteria sesuai yang diinginkan kedalam sistem, dan menginputkan kode kriteria yang sama dengan kriteria yang telah ada lalu meng-klik tombol "Simpan Data"	
		6.c Menampilkan pesan "Kode xxx sudah ada" dan pengubahan data kriteria tidak dapat diproses

Tabel 14. Skenario Hapus Data Kriteria (Admin)

Name	Hapus Data Kriteria	
Participating Actor	Admin	
Entry Condition	Admin berhasil menampilkan form Kriteria	
Exit Condition	Admin berhasil menghapus salah satu kriteria dari dalam sistem	
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Kriteria 	
Skenario Utama "Hapus Data Kriteria"		
	Aksi Aktor	Reaksi Sistem
	1. Memilih menu "Kriteria"	
		2. Menampilkan tabel Kriteria yang berkolomkan Nomor, Kode Kriteria, Nama Kriteria, Bobot Kriteria, Prosentase CF, dan Prosentase SF
	3. Meng-klik tombol "Hapus Data" pada salah satu data kriteria yang hendak dihapus	
		4. Menampilkan pertanyaan konfirmasi hapus data kriteria
	5. Melakukan konfirmasi penghapusan data kriteria dengan meng-klik tombol "OK"	
		6. Menampilkan pesan bahwa data kriteria telah berhasil dihapus

A7. Skenario Pengelolaan Data Subkriteria

Tabel 15. Skenario Tambah Data Subkriteria (Admin)

Name	Tambah Data Subkriteria	
Participating Actor	Admin	
Entry Condition	Admin berhasil menampilkan form Subkriteria	
Exit Condition	Admin berhasil menambahkan subkriteria baru kedalam sistem	
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Subkriteria 	
Skenario Utama "Tambah Data Subkriteria"		
Aksi Aktor		Reaksi Sistem
1. Memilih menu "Subkriteria"		
		2. Menampilkan tabel Subkriteria yang berkolomkan Nomor, Kode Subkriteria, Nama Subkriteria, Jenis (CF/SF), Nilai Ideal, dan Kriteria.
3. Meng-klik tombol "Tambah Data" (+)		
		4. Menampilkan form Tambah Subkriteria yang berisikan data field yang harus dipenuhi seperti Kriteria, Kode Subkriteria, Nama Subkriteria, Jenis, dan Nilai Ideal.
5. Mengisi semua keperluan data yang dibutuhkan oleh sistem dengan benar dan meng-klik tombol "Simpan Data"		
		6. Menampilkan pesan bahwa data subkriteria berhasil ditambahkan
Skenario Alternatif "Data Yang Diinputkan Tidak Lengkap"		
Aksi Aktor		Reaksi Sistem
5.a Mengisi keperluan data, dan terdapat satu atau lebih field yang belum diisi lalu meng-klik tombol "Simpan Data"		
		6.a Menampilkan pesan "data belum diisi" pada field yang masih kosong dan penambahan data subkriteria tidak dapat diproses
Skenario Alternatif "Data Yang Diinputkan Tidak Sesuai Format"		
Aksi Aktor		Reaksi Sistem
5.b Mengisi keperluan data, dan terdapat satu atau lebih field yang format penulisannya tidak sesuai dengan yang dibutuhkan oleh sistem lalu meng-klik tombol "Simpan Data"		
		6.b Menampilkan pesan "input salah" pada field yang format isiannya tidak sesuai dan penambahan data subkriteria tidak dapat diproses
Skenario Alternatif "Kode Subkriteria Telah Terpakai"		
Aksi Aktor		Reaksi Sistem
5.c Mengisi keperluan data, dan menginputkan kode subkriteria yang sama dengan subkriteria yang telah ada lalu meng-klik tombol "Simpan Data"		
		6.c Menampilkan pesan "Kode xxx sudah ada" dan penambahan data subkriteria tidak dapat diproses

Tabel 16. Skenario Ubah Data Subkriteria (Admin)

Name	Ubah Data Subkriteria	
Participating Actor	Admin	
Entry Condition	Admin berhasil menampilkan form Subkriteria	
Exit Condition	Admin berhasil mengubah subkriteria yang diinginkan pada sistem	
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Subkriteria 	
Skenario Utama		
"Ubah Data Subkriteria"		
	Aksi Aktor	Reaksi Sistem
1.	Memilih menu "Subkriteria"	
		2. Menampilkan tabel Subkriteria yang berkolomkan Nomor, Kode Subkriteria, Nama Subkriteria, Jenis (CF/SF), Nilai Ideal, dan Kriteria
3.	Meng-klik tombol "Ubah Data" pada salah satu data subkriteria yang hendak diubah	
		4. Menampilkan form Ubah Subkriteria yang berisikan data field yang harus dipenuhi seperti Kriteria, Kode Subkriteria, Nama Subkriteria, Jenis, dan Nilai Ideal
5.	Mengubah data subkriteria sesuai yang diinginkan kedalam sistem dengan benar dan meng-klik tombol "Simpan Data"	
		6. Menampilkan pesan bahwa data subkriteria berhasil diubah
Skenario Alternatif		
"Data Yang Diinputkan Tidak Lengkap"		
	Aksi Aktor	Reaksi Sistem
5.a	Mengubah data subkriteria sesuai yang diinginkan kedalam sistem, dan terdapat satu atau lebih field yang belum diisi dan meng-klik tombol "Simpan Data"	
		6.a Menampilkan pesan "data belum diisi" pada field yang masih kosong dan perubahan data subkriteria tidak dapat diproses
Skenario Alternatif		
"Data Yang Diinputkan Tidak Sesuai Format"		
	Aksi Aktor	Reaksi Sistem
5.b	Mengubah data subkriteria sesuai yang diinginkan kedalam sistem, dan terdapat satu atau lebih field yang format penulisannya tidak sesuai dengan yang dibutuhkan oleh sistem dan meng-klik tombol "Simpan Data"	
		6.b Menampilkan pesan "input salah" pada field yang format isiannya tidak sesuai dan perubahan data subkriteria tidak dapat diproses
Skenario Alternatif		
"Kode Subkriteria Telah Terpakai"		
	Aksi Aktor	Reaksi Sistem
5.c	Mengubah data subkriteria sesuai yang diinginkan kedalam sistem, dan menginputkan kode subkriteria yang sama dengan subkriteria yang telah ada lalu meng-klik tombol "Simpan Data"	
		6.c Menampilkan pesan "Kode xxx sudah ada" dan perubahan data subkriteria tidak dapat diproses

Tabel 17. Skenario Hapus Data Subkriteria (Admin)

Name	Hapus Data Subkriteria	
Participating Actor	Admin	
Entry Condition	Admin berhasil menampilkan form Subkriteria	
Exit Condition	Admin berhasil menghapus salah satu subkriteria dari dalam sistem	
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Subkriteria 	
Skenario Utama "Hapus Data Subkriteria"		
	Aksi Aktor	Reaksi Sistem
	1. Memilih menu "Subkriteria"	
		2. Menampilkan tabel Subkriteria yang berkolomkan Nomor, Kode Subkriteria, Nama Subkriteria, Jenis (CF/SF), Nilai Ideal, dan Kriteria
	3. Meng-klik tombol "Hapus Data" pada salah satu data subkriteria yang hendak dihapus	
		4. Menampilkan pertanyaan konfirmasi hapus data subkriteria
	5. Melakukan konfirmasi penghapusan data subkriteria dengan meng-klik tombol "OK"	
		6. Menampilkan pesan bahwa data subkriteria telah berhasil dihapus

A8. Skenario Pengelolaan Bobot *gap*Tabel 18. Skenario Tambah Bobot *gap* (Admin)

Name	Tambah Bobot <i>gap</i>	
Participating Actor	Admin	
Entry Condition	Admin berhasil menampilkan form Bobot <i>gap</i>	
Exit Condition	Admin berhasil menambahkan bobot <i>gap</i> baru kedalam sistem	
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Bobot <i>gap</i> 	
Skenario Utama "Tambah Bobot <i>gap</i>"		
Aksi Aktor		Reaksi Sistem
1. Memilih menu "Bobot <i>gap</i> "		2. Menampilkan tabel Bobot <i>gap</i> yang berkolomkan Nomor, Kriteria, Subkriteria, Selisih, Bobot, dan Keterangan.
3. Meng-klik tombol "Tambah Data" (+)		4. Menampilkan form Tambah Bobot <i>gap</i> yang berisikan data field yang harus dipenuhi seperti Subkriteria, Selisih, Bobot, dan Keterangan.
5. Mengisi semua keperluan data yang dibutuhkan oleh sistem dengan benar dan meng-klik tombol "Simpan Data"		6. Menampilkan pesan bahwa data bobot <i>gap</i> berhasil ditambahkan
Skenario Alternatif "Data Yang Diinputkan Tidak Lengkap"		
Aksi Aktor		Reaksi Sistem
5.a Mengisi keperluan data, dan terdapat satu atau lebih field yang belum diisi lalu meng-klik tombol "Simpan Data"		6.a Menampilkan pesan "data belum diiisi" pada field yang masih kosong dan penambahan data bobot <i>gap</i> tidak dapat diproses
Skenario Alternatif "Data Yang Diinputkan Tidak Sesuai Format"		
Aksi Aktor		Reaksi Sistem
5.b Mengisi keperluan data, dan terdapat satu atau lebih field yang format penulisannya tidak sesuai dengan yang dibutuhkan oleh sistem lalu meng-klik tombol "Simpan Data"		6.b Menampilkan pesan "input salah" pada field yang format isiannya tidak sesuai dan penambahan data bobot <i>gap</i> tidak dapat diproses

Tabel 19. Skenario Ubah Bobot *gap* (Admin)

Name	Ubah Bobot <i>gap</i>
Participating Actor	Admin
Entry Condition	Admin berhasil menampilkan form Bobot <i>gap</i>
Exit Condition	Admin berhasil mengubah bobot <i>gap</i> yang diinginkan pada sistem
Event Flow	<ol style="list-style-type: none"> Admin membuka website Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> Admin klik tombol "<i>Sign In</i>" Admin menuju menu Bobot <i>gap</i>
Skenario Utama "Ubah Bobot <i>gap</i>"	
Aksi Aktor	Reaksi Sistem
1. Memilih menu "Bobot <i>gap</i> "	
	2. Menampilkan tabel Bobot <i>gap</i> yang berkolomkan Nomor, Kriteria, Subkriteria, Selisih, Bobot, dan Keterangan
3. Meng-klik tombol "Ubah Data" pada salah satu data bobot <i>gap</i> yang hendak diubah	
	4. Menampilkan form Ubah Bobot <i>gap</i> yang berisikan data field yang harus dipenuhi seperti Subkriteria, Selisih, Bobot, dan Keterangan
5. Mengubah data bobot <i>gap</i> sesuai yang diinginkan kedalam sistem dengan benar dan meng-klik tombol "Simpan Data"	
	6. Menampilkan pesan bahwa data bobot <i>gap</i> berhasil diubah
Skenario Alternatif "Data Yang Diinputkan Tidak Lengkap"	
Aksi Aktor	Reaksi Sistem
5.a Mengubah data bobot <i>gap</i> sesuai yang diinginkan kedalam sistem, dan terdapat satu atau lebih field yang belum diisi dan meng-klik tombol "Simpan Data"	
	6.a Menampilkan pesan "data belum diisi" pada field yang masih kosong dan perubahan data bobot <i>gap</i> tidak dapat diproses
Skenario Alternatif "Data Yang Diinputkan Tidak Sesuai Format"	
Aksi Aktor	Reaksi Sistem
5.b Mengubah data bobot <i>gap</i> sesuai yang diinginkan kedalam sistem, dan terdapat satu atau lebih field yang format penulisannya tidak sesuai dengan yang dibutuhkan oleh sistem dan meng-klik tombol "Simpan Data"	
	6.b Menampilkan pesan "input salah" pada field yang format isinya tidak sesuai dan perubahan data bobot <i>gap</i> tidak dapat diproses

Tabel 20. Skenario Hapus Bobot *gap* (Admin)

Name	Hapus Bobot <i>gap</i>
Participating Actor	Admin
Entry Condition	Admin berhasil menampilkan form Bobot <i>gap</i>
Exit Condition	Admin berhasil menghapus salah satu bobot <i>gap</i> dari dalam sistem
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Bobot <i>gap</i>
Skenario Utama "Hapus Bobot <i>gap</i>"	
Aksi Aktor	Reaksi Sistem
1. Memilih menu "Bobot <i>gap</i> "	
	2. Menampilkan tabel Bobot <i>gap</i> yang berkolomkan Nomor, Kriteria, Subkriteria, Selisih, Bobot, dan Keterangan
3. Meng-klik tombol "Hapus Data" pada salah satu data bobot <i>gap</i> yang hendak dihapus	
	4. Menampilkan pertanyaan konfirmasi hapus data bobot <i>gap</i>
5. Melakukan konfirmasi penghapusan data bobot <i>gap</i> dengan meng-klik tombol "OK"	
	6. Menampilkan pesan bahwa data bobot <i>gap</i> telah berhasil dihapus

A9. Skenario Pengelolaan Data Pengguna

Tabel 21. Skenario Tambah Data Pengguna (Admin)

Name	Tambah Data Pengguna	
Participating Actor	Admin	
Entry Condition	Admin berhasil menampilkan form Hak Akses Pengguna	
Exit Condition	Admin berhasil menambahkan data pengguna	
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Hak Akses Pengguna 	
Skenario Utama		
"Tambah Data Pengguna"		
	Aksi Aktor	Reaksi Sistem
	1. Memilih menu "Hak Akses Pengguna"	
		2. Menampilkan tabel Hak Akses Pengguna yang berkolomkan Nomor, Username, Nama Lengkap, dan Hak Akses.
	3. Meng-klik tombol "Tambah Data" (+)	
		4. Menampilkan form Tambah Pengguna yang berisikan data field yang harus dipenuhi seperti Nama Lengkap, Username, Password, Ulangi Password, dan Hak Akses.
	5. Mengisi semua keperluan data yang dibutuhkan oleh sistem dengan benar dan meng-klik tombol "Simpan Data"	
		6. Menampilkan pesan bahwa data pengguna berhasil ditambahkan
Skenario Alternatif		
"Data Yang Diinputkan Tidak Lengkap"		
	Aksi Aktor	Reaksi Sistem
	5.a Mengisi keperluan data, dan terdapat satu atau lebih field yang belum diisi lalu meng-klik tombol "Simpan Data"	
		6.a Menampilkan pesan "data belum diisi" pada field yang masih kosong dan penambahan data pengguna tidak dapat diproses
Skenario Alternatif		
"Data Yang Diinputkan Tidak Sesuai Format"		
	Aksi Aktor	Reaksi Sistem
	5.b Mengisi keperluan data, dan terdapat satu atau lebih field yang format penulisannya tidak sesuai dengan yang dibutuhkan oleh sistem lalu meng-klik tombol "Simpan Data"	
		6.b Menampilkan pesan "input salah" pada field yang format isinya tidak sesuai dan penambahan data pengguna tidak dapat diproses
Skenario Alternatif		
"Username Telah Terpakai"		
	Aksi Aktor	Reaksi Sistem
	5.c Mengisi keperluan data, dan menginputkan username yang sama dengan username yang telah ada lalu meng-klik tombol "Simpan Data"	
		6.c Menampilkan pesan "Pengguna xxx sudah ada" dan penambahan data pengguna tidak dapat diproses

Tabel 22. Skenario Ubah Data Pengguna (Admin)

Name	Ubah Data Pengguna	
Participating Actor	Admin	
Entry Condition	Admin berhasil menampilkan form Hak Akses Pengguna	
Exit Condition	Admin berhasil mengubah data dari pengguna yang diinginkan	
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Hak Akses Pengguna 	
Skenario Utama "Ubah Data Pengguna"		
	Aksi Aktor	Reaksi Sistem
	1. Memilih menu "Hak Akses Pengguna"	
		2. Menampilkan tabel Hak Akses Pengguna yang berkolomkan Nomor, Username, Nama Lengkap, dan Hak Akses
	3. Meng-klik tombol "Ubah Data" pada salah satu data pengguna yang hendak diubah	
		4. Menampilkan form Ubah Pengguna yang berisikan data field yang harus dipenuhi seperti Nama Lengkap, Username, Password, Ulangi Password, dan Hak Akses
	5. Mengubah data pengguna sesuai yang diinginkan kedalam sistem dengan benar dan meng-klik tombol "Simpan Data"	
		6. Menampilkan pesan bahwa data pengguna berhasil diubah
Skenario Alternatif "Data Yang Diinputkan Tidak Lengkap"		
	Aksi Aktor	Reaksi Sistem
	5.a Mengubah data pengguna sesuai yang diinginkan kedalam sistem, dan terdapat satu atau lebih field yang belum diisi dan meng-klik tombol "Simpan Data"	
		6.a Menampilkan pesan "data belum diisi" pada field yang masih kosong dan perubahan data pengguna tidak dapat diproses
Skenario Alternatif "Data Yang Diinputkan Tidak Sesuai Format"		
	Aksi Aktor	Reaksi Sistem
	5.b Mengubah data pengguna sesuai yang diinginkan kedalam sistem, dan terdapat satu atau lebih field yang format penulisannya tidak sesuai dengan yang dibutuhkan oleh sistem dan meng-klik tombol "Simpan Data"	
		6.b Menampilkan pesan "input salah" pada field yang format isiannya tidak sesuai dan perubahan data pengguna tidak dapat diproses
Skenario Alternatif "Username Telah Terpakai"		
	Aksi Aktor	Reaksi Sistem
	5.c Mengubah data pengguna sesuai yang diinginkan kedalam sistem, dan menginputkan username yang sama dengan username yang telah ada lalu meng-klik tombol "Simpan Data"	
		6.c Menampilkan pesan "Pengguna xxx sudah ada" dan perubahan data pengguna tidak dapat diproses

Tabel 23. Skenario Hapus Data Pengguna (Admin)

Name	Hapus Data Pengguna	
Participating Actor	Admin	
Entry Condition	Admin berhasil menampilkan form Hak Akses Pengguna	
Exit Condition	Admin berhasil menghapus salah satu pengguna	
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Hak Akses Pengguna 	
Skenario Utama "Hapus Data Pengguna"		
	Aksi Aktor	Reaksi Sistem
	1. Memilih menu "Hak Akses Pengguna"	
		2. Menampilkan tabel Hak Akses Pengguna yang berkolomkan Nomor, Username, Nama Lengkap, dan Hak Akses
	3. Meng-klik tombol "Hapus Data" pada salah satu data pengguna yang hendak dihapus	
		4. Menampilkan pertanyaan konfirmasi hapus data pengguna
	5. Melakukan konfirmasi penghapusan data pengguna dengan meng-klik tombol "OK"	
		6. Menampilkan pesan bahwa data pengguna telah berhasil dihapus

A10. Skenario View Hasil Perhitungan

Tabel 24. Skenario View Hasil Perhitungan (Admin)

Name	View Hasil Perhitungan	
Participating Actor	Admin	
Entry Condition	Admin berhasil masuk ke sistem (<i>Sign In</i>)	
Exit Condition	Admin berhasil menampilkan form Hasil Perhitungan	
Event Flow	<ol style="list-style-type: none"> 1. Admin membuka website 2. Admin memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Admin klik tombol "<i>Sign In</i>" 4. Admin menuju menu Hasil Perhitungan 	
Skenario Utama "View Hasil Perhitungan"		
	Aksi Aktor	Reaksi Sistem
	1. Memilih menu "Hasil Perhitungan"	
		2. Menampilkan menu Hasil Perhitungan
	3. Memilih periode lomba yang diinginkan lalu klik "Submit"	
		4. Menyajikan data seleksi siswa terdaftar sesuai dengan periode lomba yang dipilih berisikan tabel langkah-langkah seleksi berdasarkan metode, ranking siswa, dan siswa terpilih berdasarkan ranking dan kuota.

Tabel 25. Skenario *View* Hasil Perhitungan (Wali Kelas)

Name	<i>View</i> Hasil Perhitungan	
Participating Actor	Wali Kelas	
Entry Condition	Wali Kelas berhasil masuk ke sistem (<i>Sign In</i>)	
Exit Condition	Wali Kelas berhasil menampilkan form Hasil Perhitungan	
Event Flow	<ol style="list-style-type: none"> 1. Wali Kelas membuka website 2. Wali Kelas memasukkan username dan <i>password</i> untuk <i>Sign In</i> 3. Wali Kelas klik tombol "<i>Sign In</i>" 4. Wali Kelas menuju menu Hasil Perhitungan 	
Skenario Utama "View Hasil Perhitungan"		
Aksi Aktor		Reaksi Sistem
1. Memilih menu "Hasil Perhitungan"		2. Menampilkan menu Hasil Perhitungan
3. Memilih periode lomba yang diinginkan lalu klik "Submit"		4. Menyajikan data seleksi siswa terdaftar sesuai dengan periode lomba yang dipilih berisikan tabel langkah-langkah seleksi berdasarkan metode, ranking siswa, dan siswa terpilih berdasarkan ranking dan kuota.

A11. Skenario *Sign Out*Tabel 26. Skenario *Sign Out* (Admin)

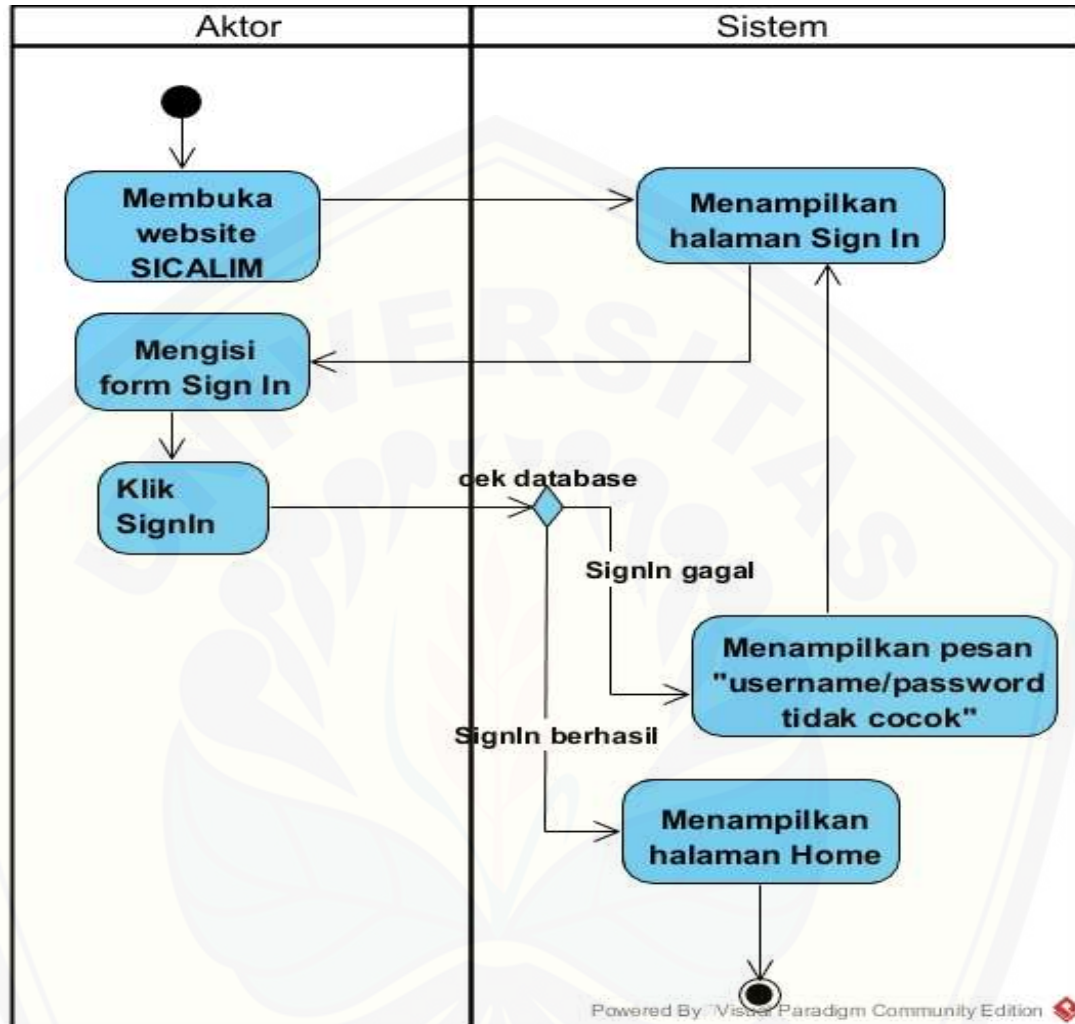
Name	<i>Sign Out</i>	
Participating Actor	Admin	
Entry Condition	Admin ingin keluar dari sistem	
Exit Condition	Admin berhasil melakukan <i>Sign Out</i> dan keluar dari sistem	
Event Flow	1. Admin klik tombol <i>Sign Out</i>	
Skenario Utama "Sign Out"		
Aksi Aktor		Reaksi Sistem
1. Meng-klik tombol " <i>Sign Out</i> "		2. Menampilkan halaman <i>Sign In</i>

Tabel 27. Skenario *Sign Out* (Wali Kelas)

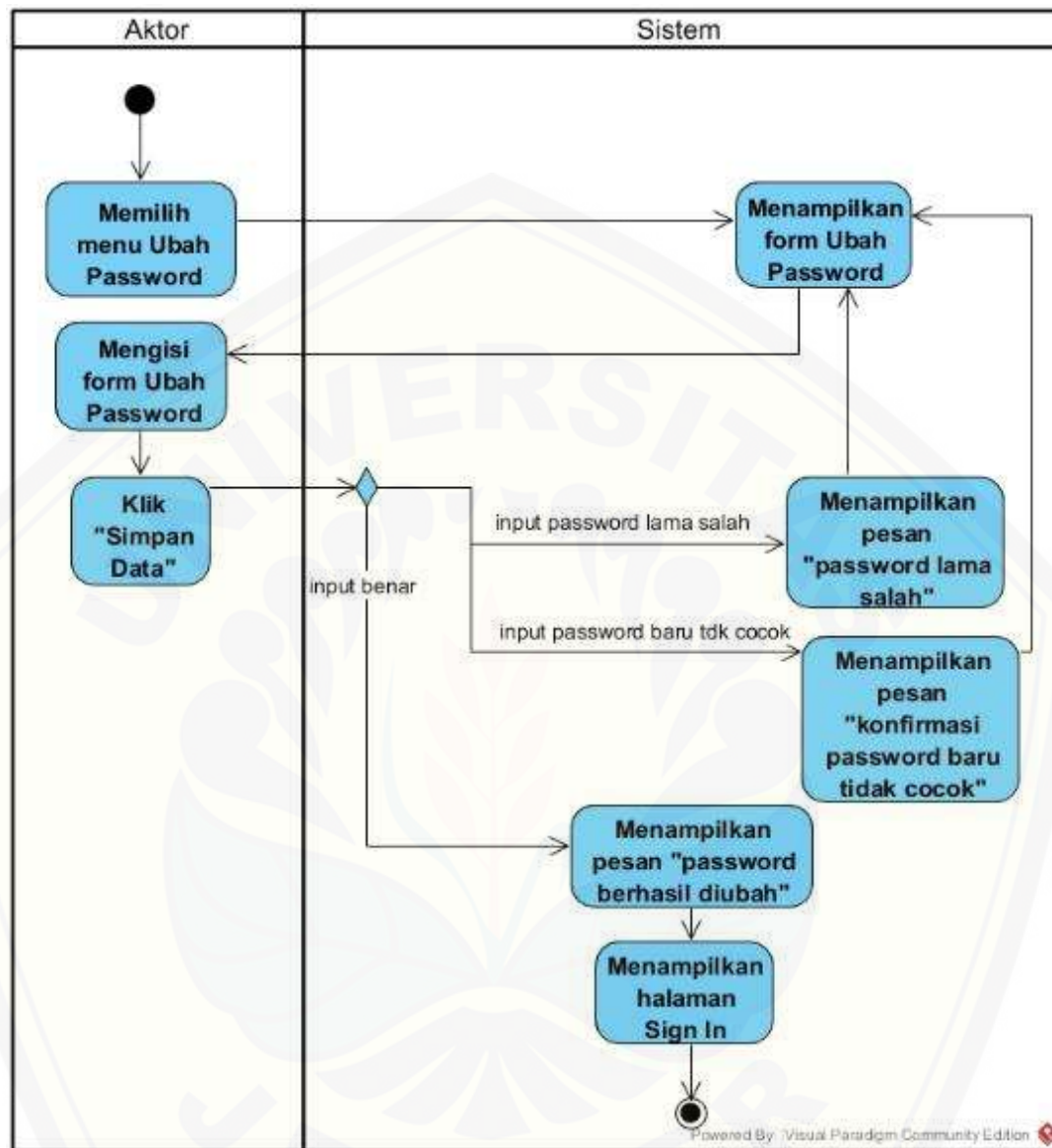
Name	<i>Sign Out</i>	
Participating Actor	Wali Kelas	
Entry Condition	Wali Kelas ingin keluar dari sistem	
Exit Condition	Wali Kelas berhasil melakukan <i>Sign Out</i> dan keluar dari sistem	
Event Flow	1. Admin klik tombol <i>Sign Out</i>	
Skenario Utama "Sign Out"		
Aksi Aktor		Reaksi Sistem
1. Meng-klik tombol " <i>Sign Out</i> "		2. Menampilkan halaman <i>Sign In</i>

LAMPIRAN B. ACTIVITY DIAGRAM

B1. Activity Diagram Sign In

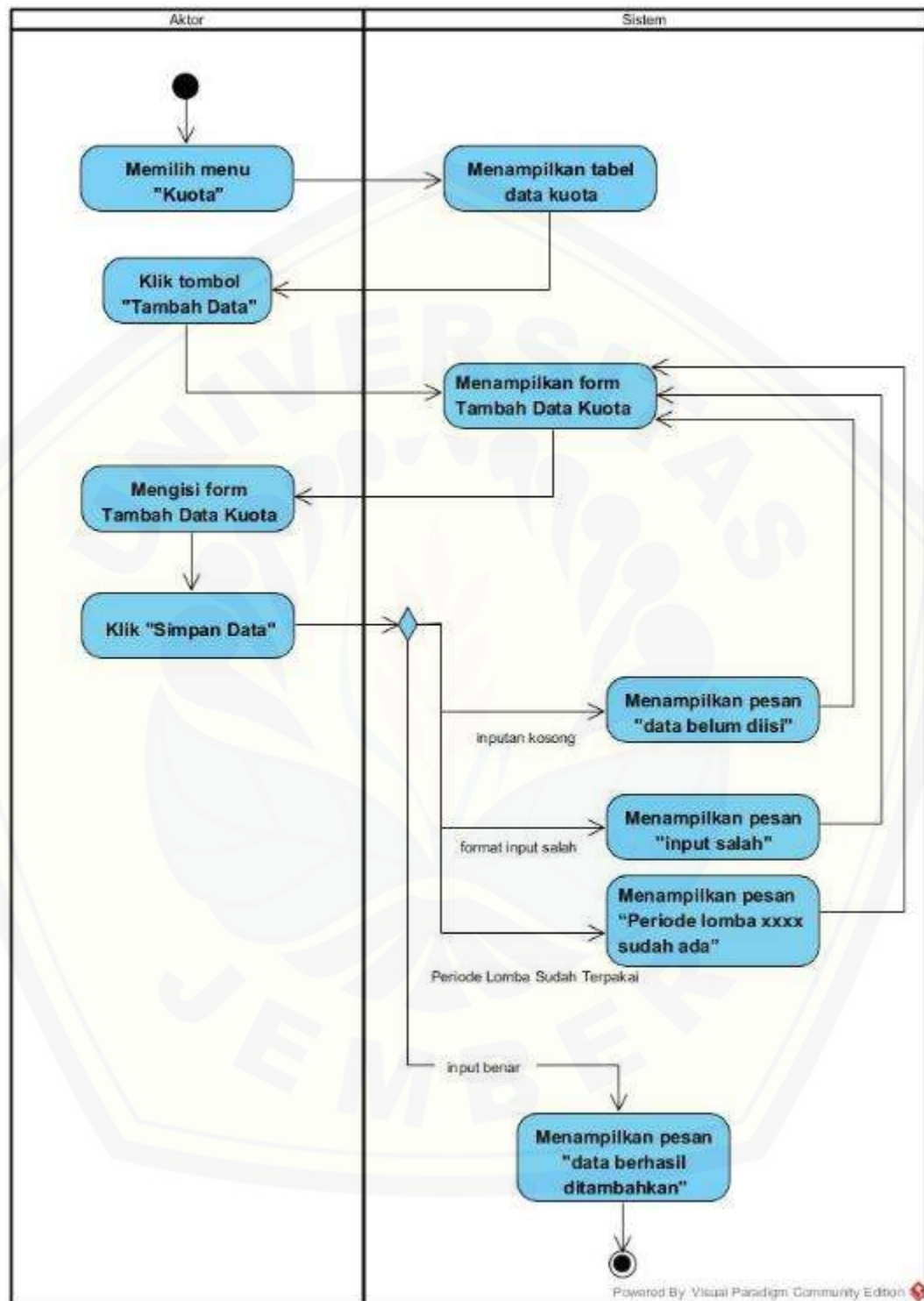


Gambar 1. Activity Diagram Sign In

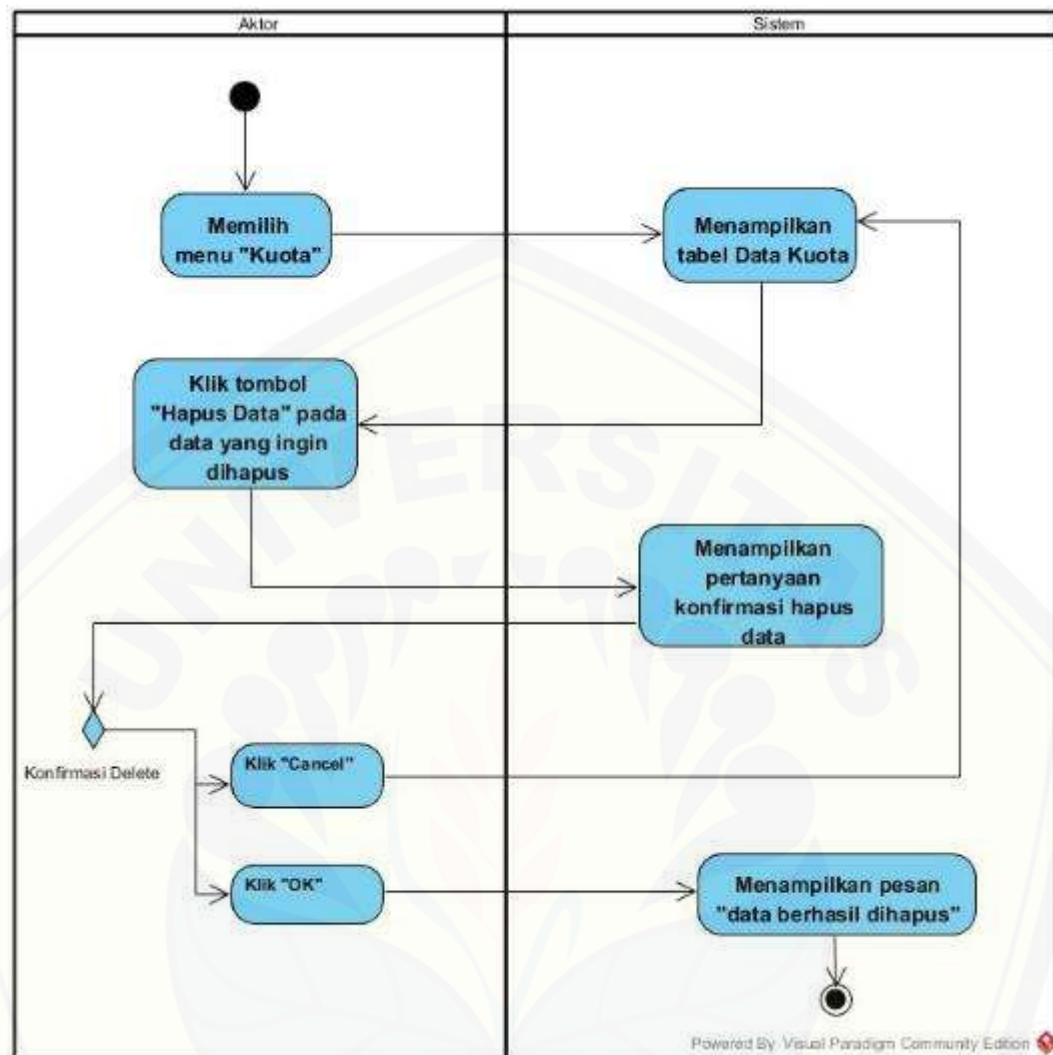
B2. Activity Diagram Ubah Password

Gambar 2. Activity Diagram Ubah Password

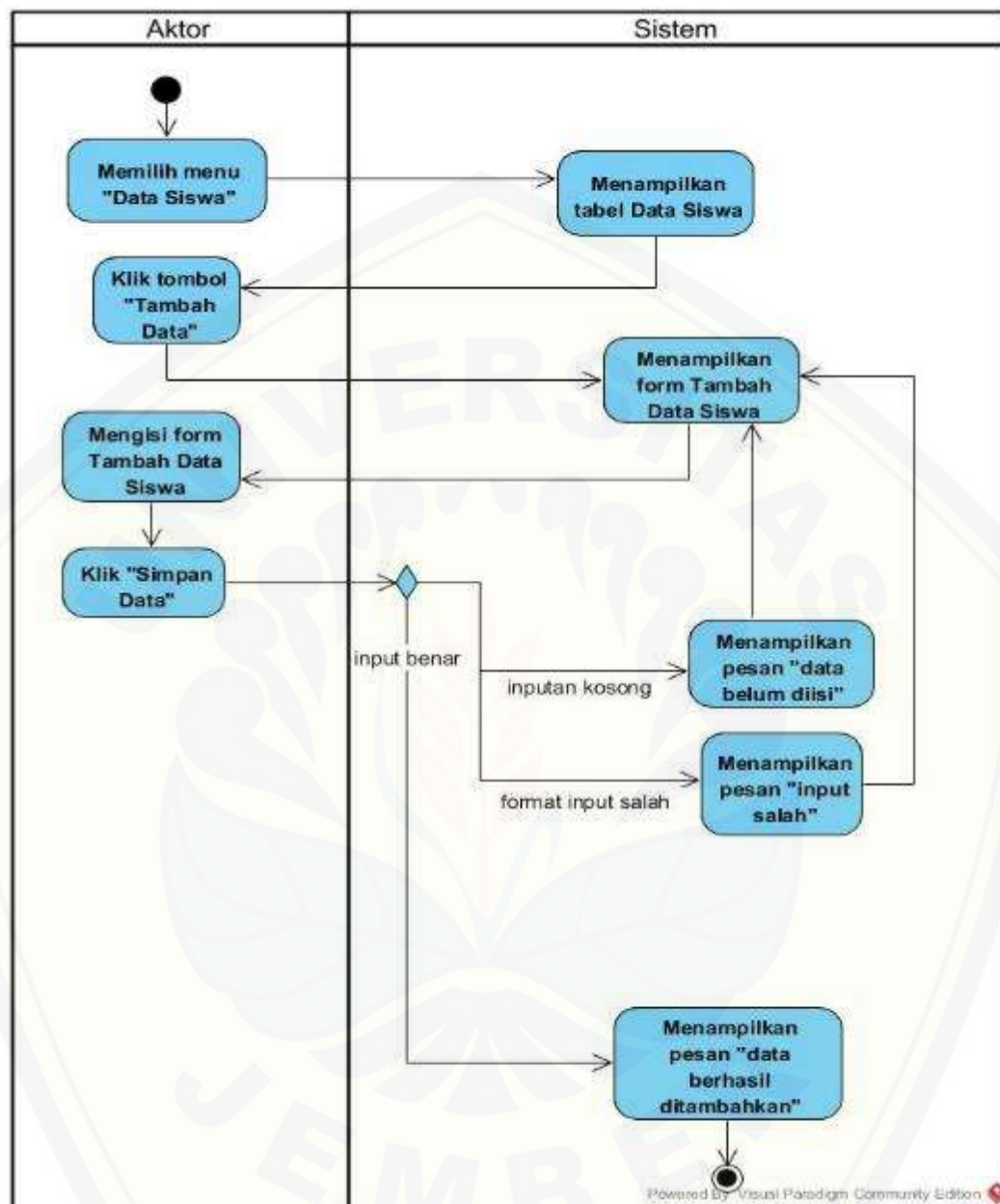
B3. Activity Diagram Input Kuota Lomba



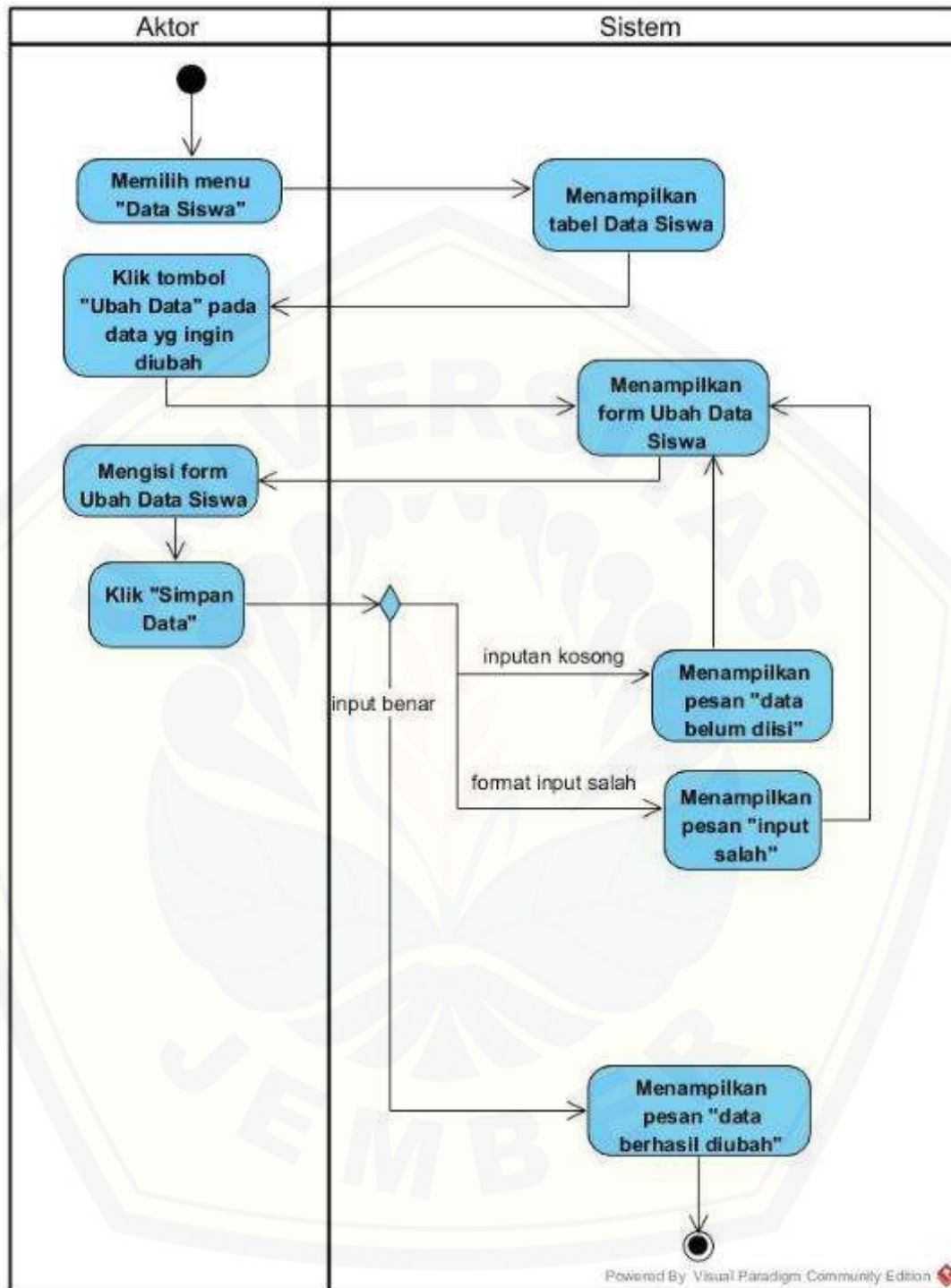
Gambar 3. Activity Diagram Input Kuota Lomba (Tambah Data)



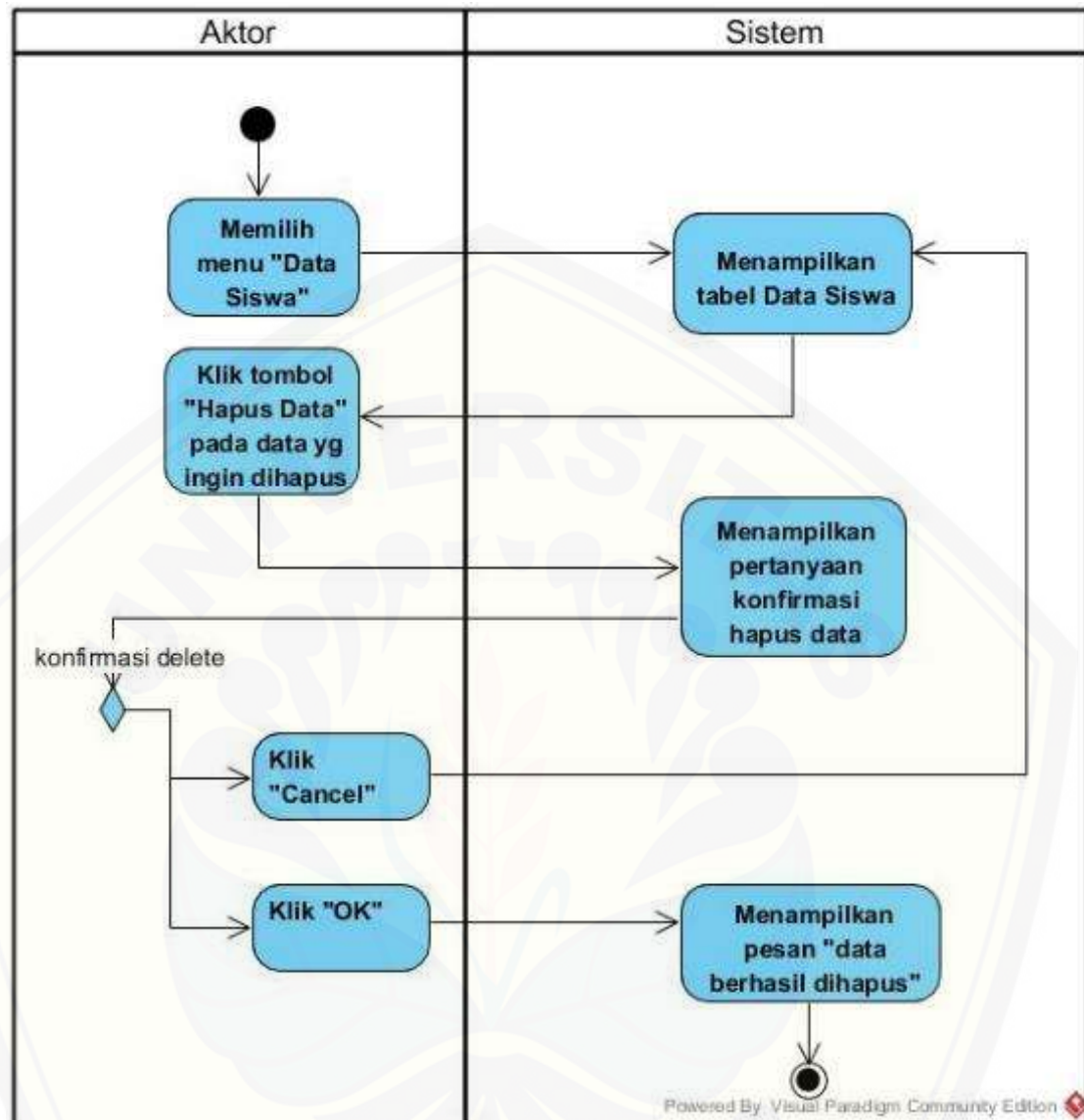
Gambar 4. Activity Diagram Input Kuota Lomba (Hapus Data)

B4. Activity Diagram Pengelolaan Data Siswa

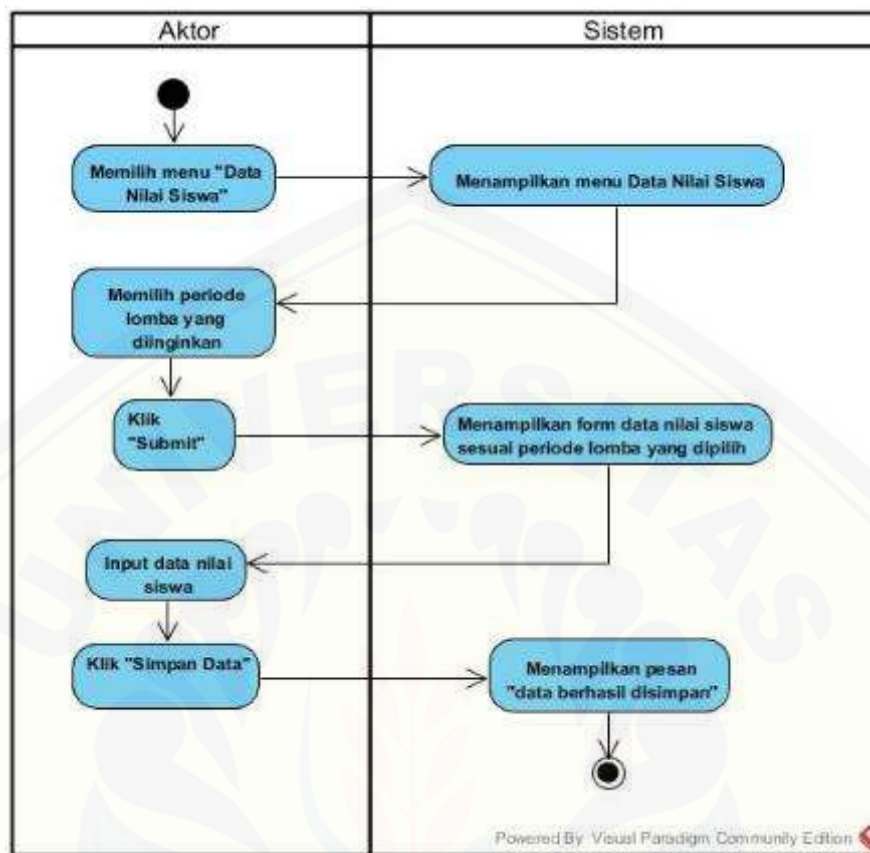
Gambar 5. Activity Diagram Pengelolaan Data Siswa (Tambah Data)



Gambar 6. Activity Diagram Pengelolaan Data Siswa (Ubah Data)

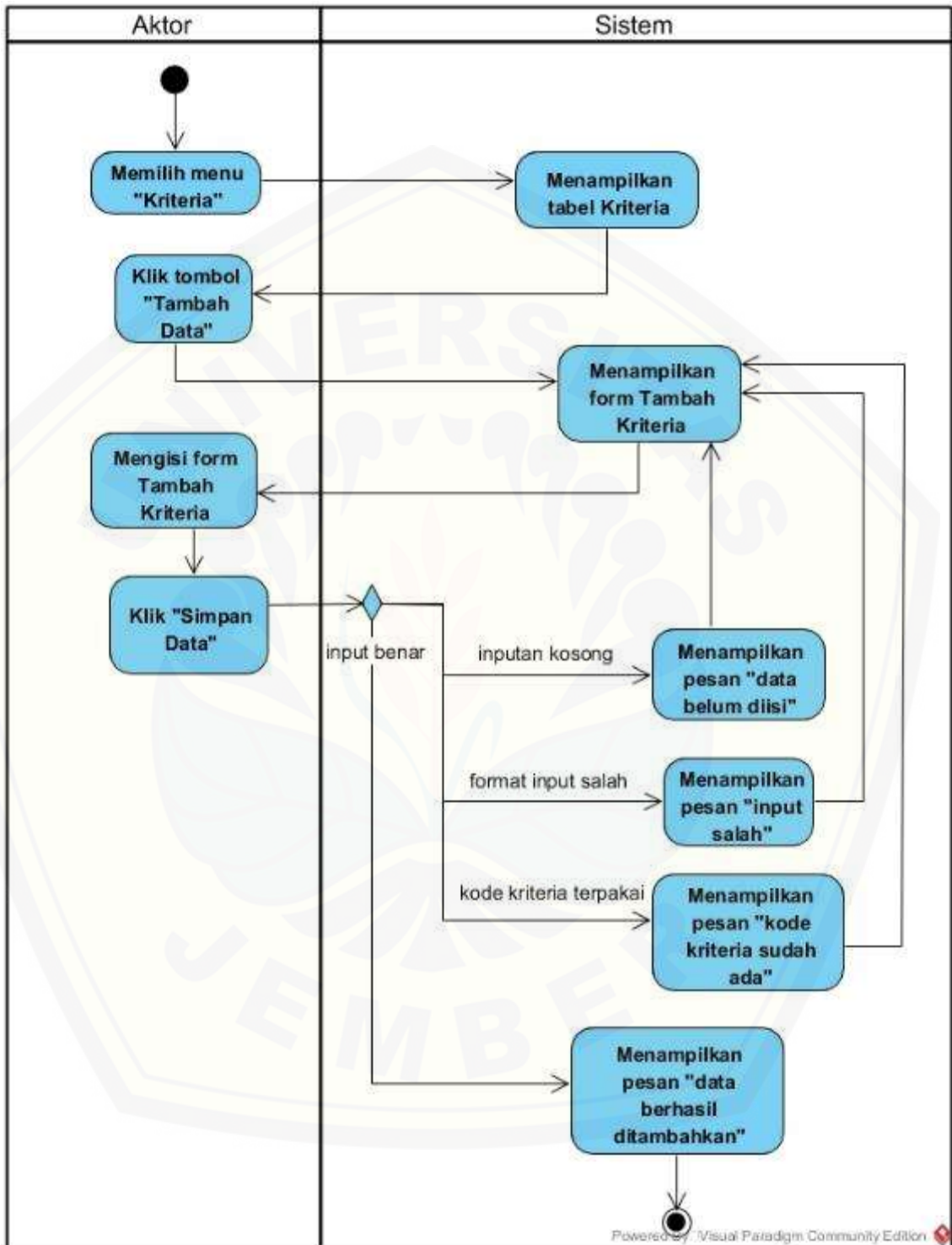


Gambar 7. Activity Diagram Pengelolaan Data Siswa (Hapus Data)

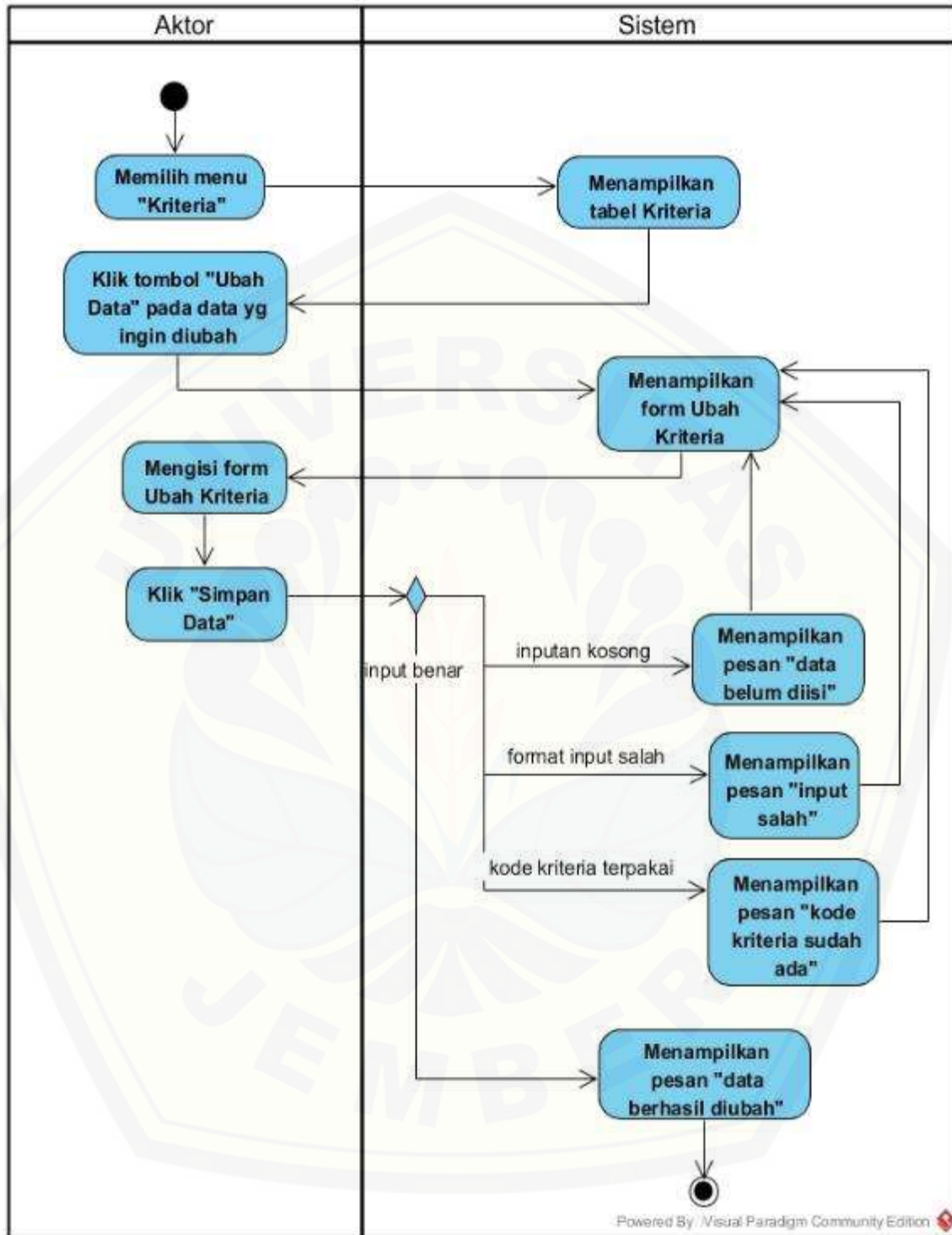
B5. Activity Diagram Pengelolaan Data Nilai Siswa

Gambar 8. Activity Diagram Pengelolaan Data Nilai Siswa

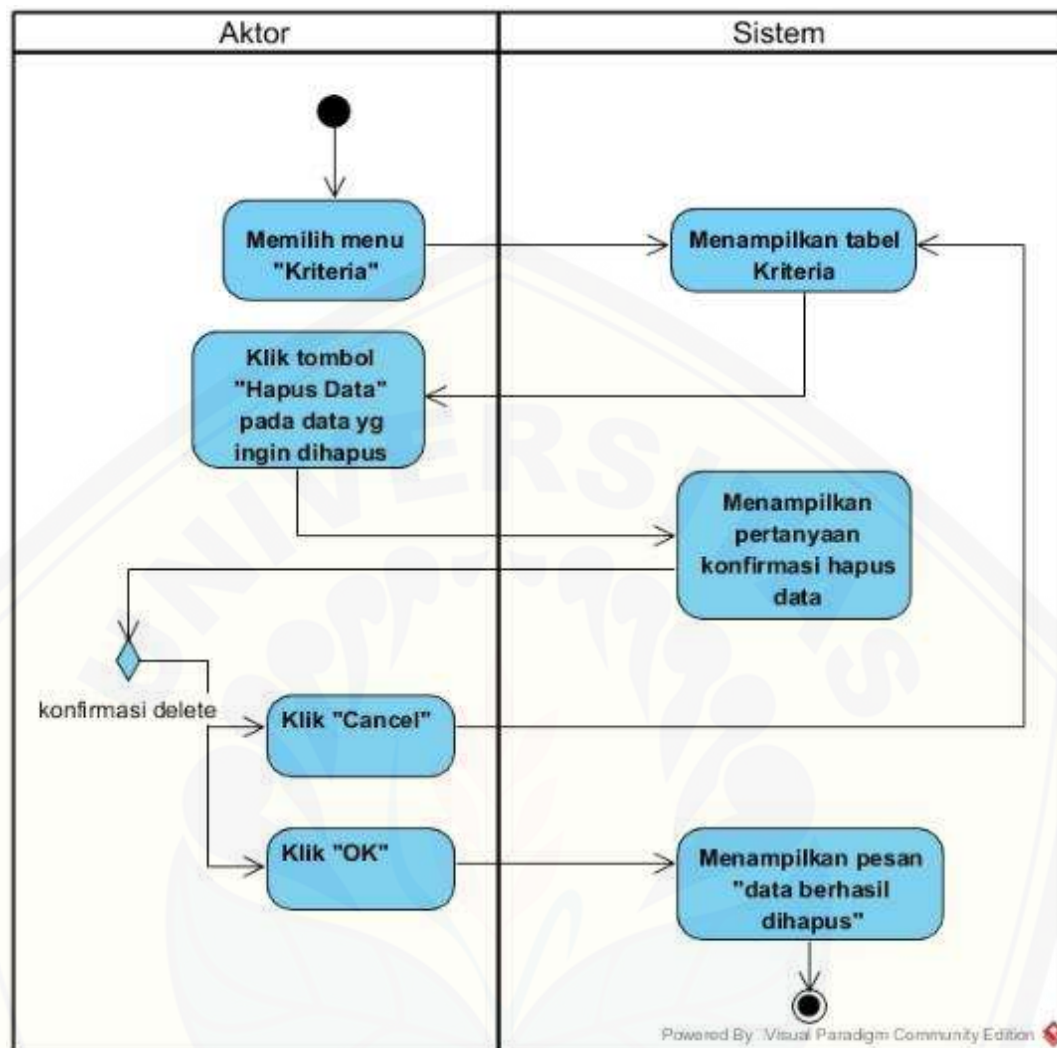
B6. Activity Diagram Pengelolaan Data Kriteria



Gambar 9. Activity Diagram Pengelolaan Data Kriteria (Tambah Data)

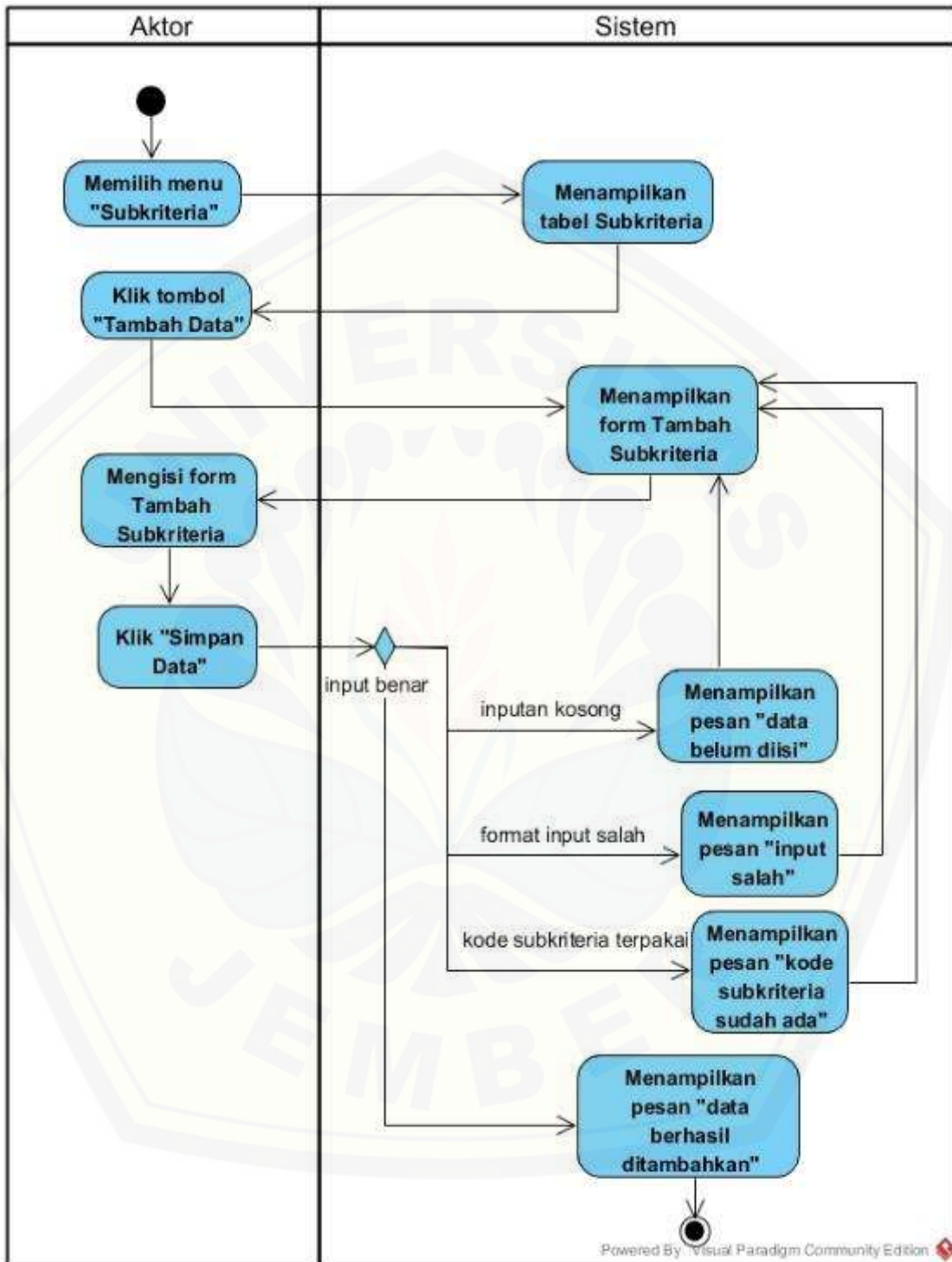


Gambar 10. Activity Diagram Pengelolaan Data Kriteria (Ubah Data)

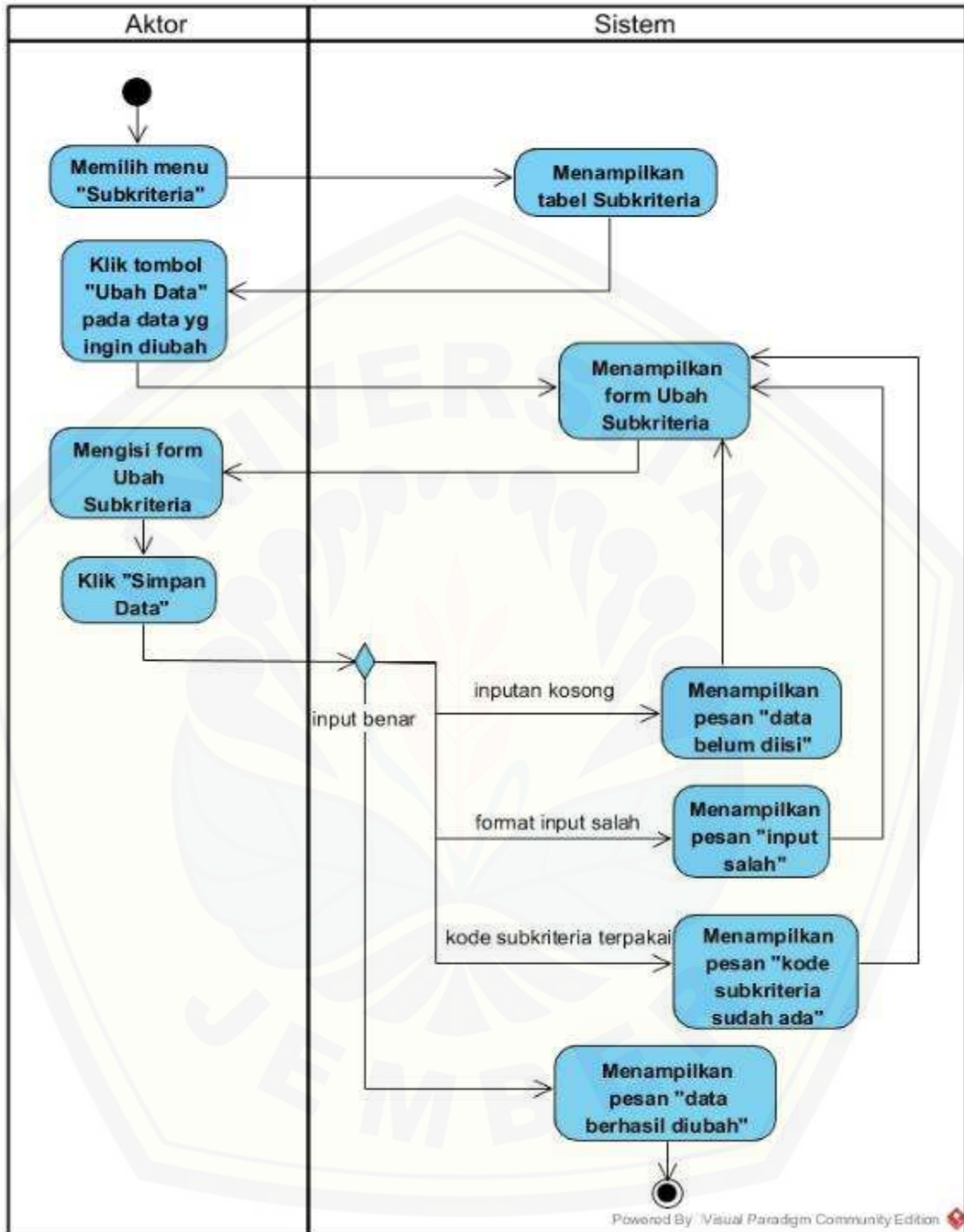


Gambar 11. Activity Diagram Pengelolaan Data Kriteria (Hapus Data)

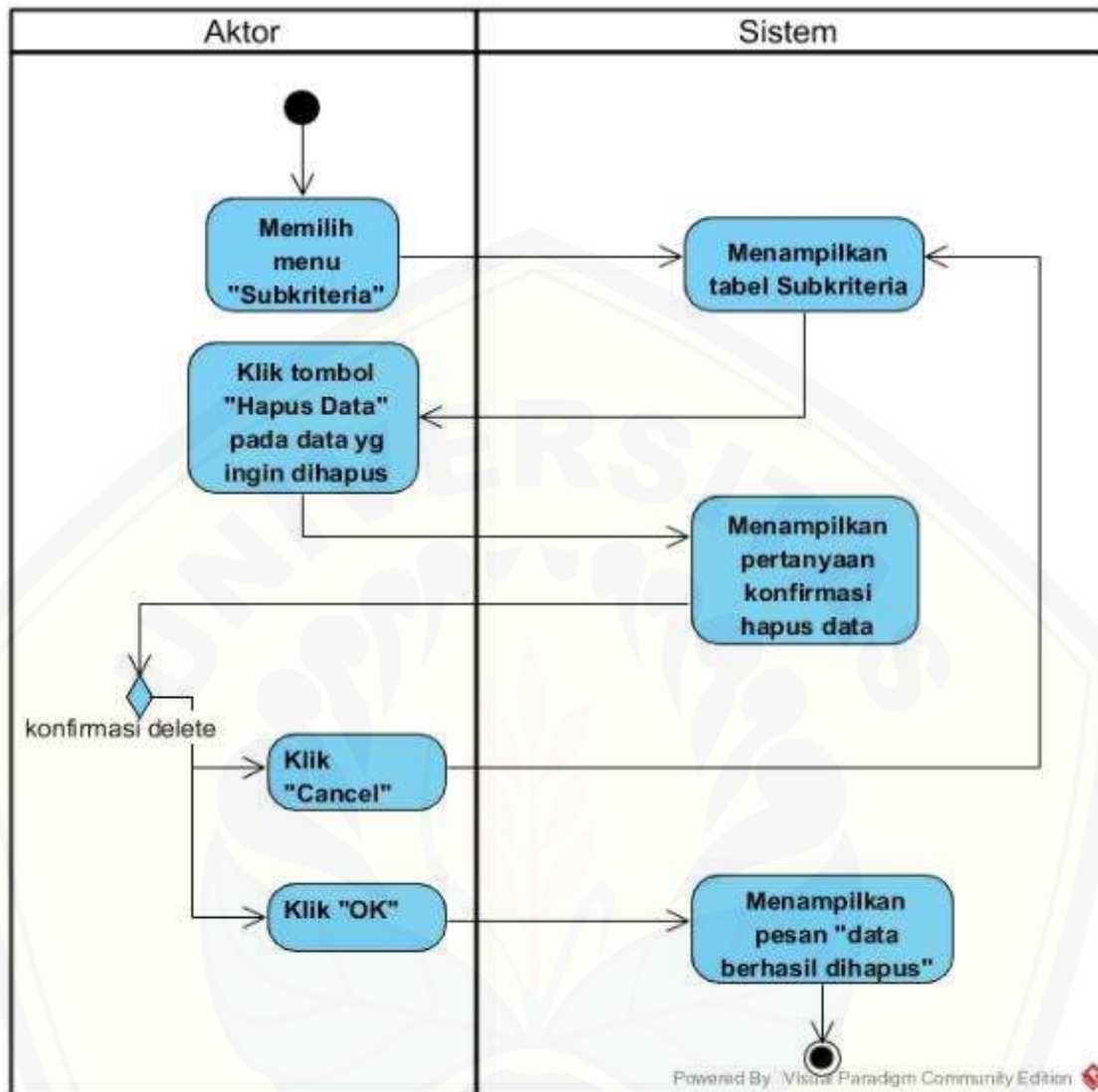
B7. Activity Diagram Pengelolaan Data Subkriteria



Gambar 12. Activity Diagram Pengelolaan Data Subkriteria (Tambah Data)

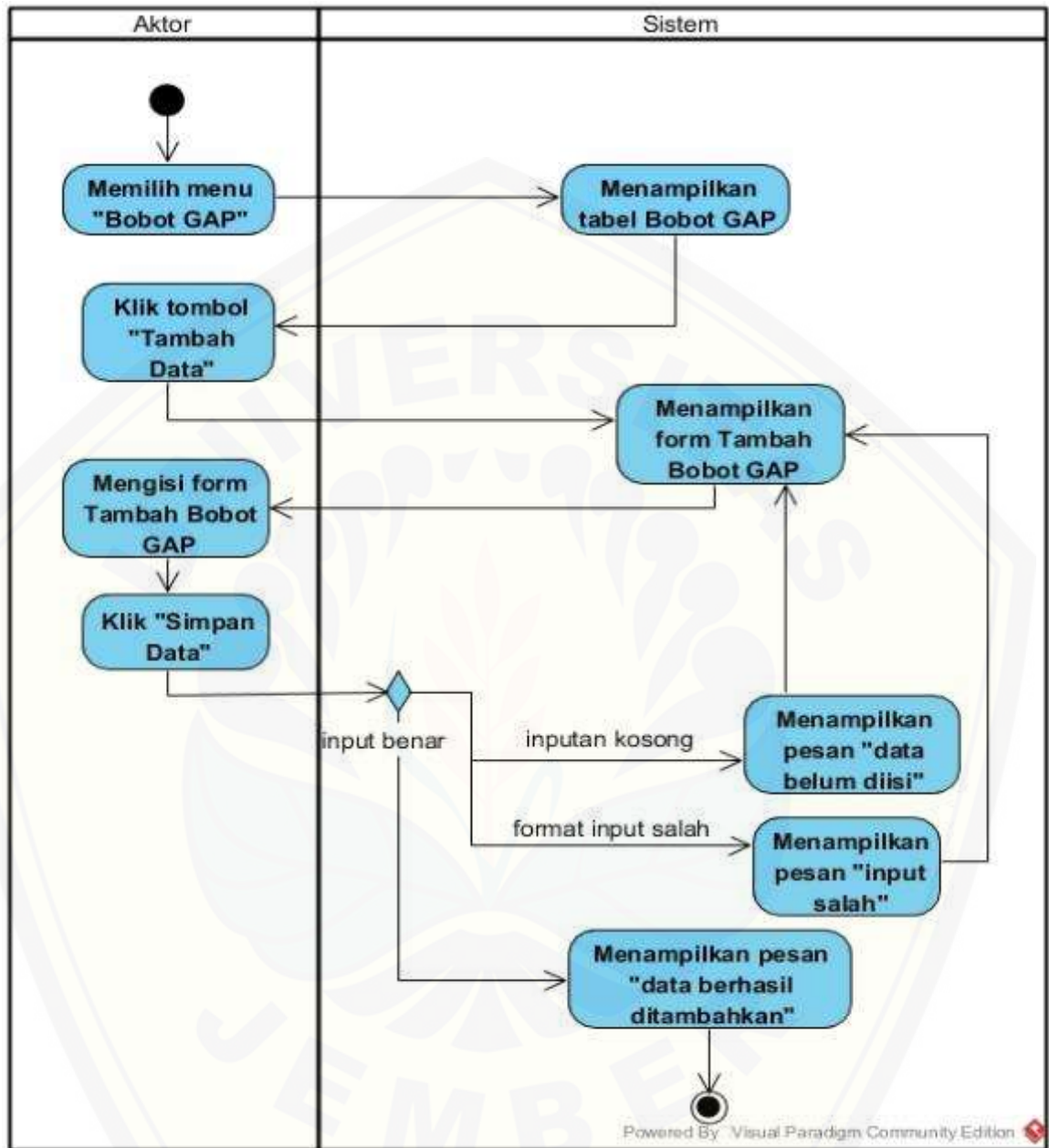


Gambar 13. Activity Diagram Pengelolaan Data Subkriteria (Ubah Data)

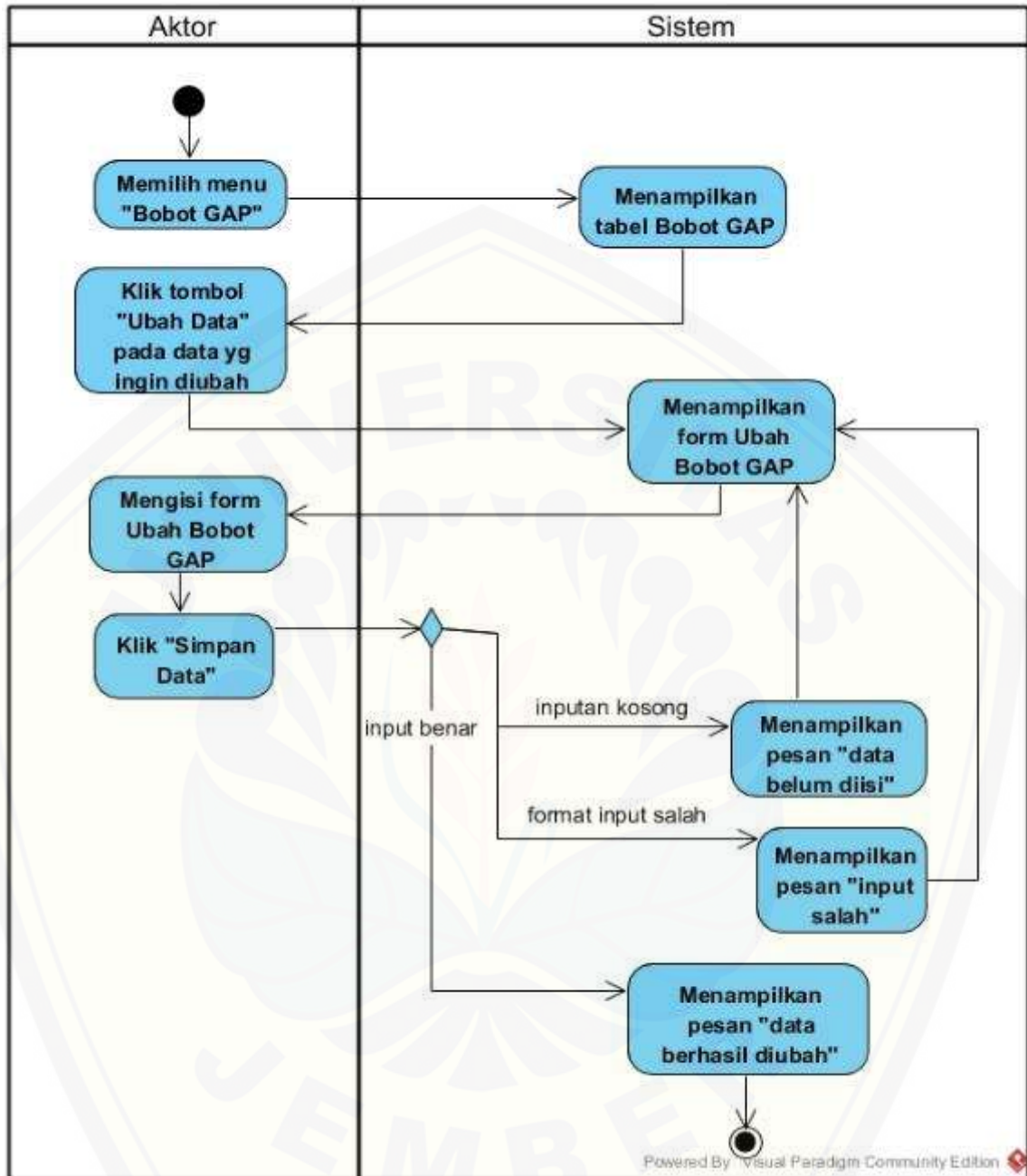


Gambar 14. Activity Diagram Pengelolaan Data Subkriteria (Hapus Data)

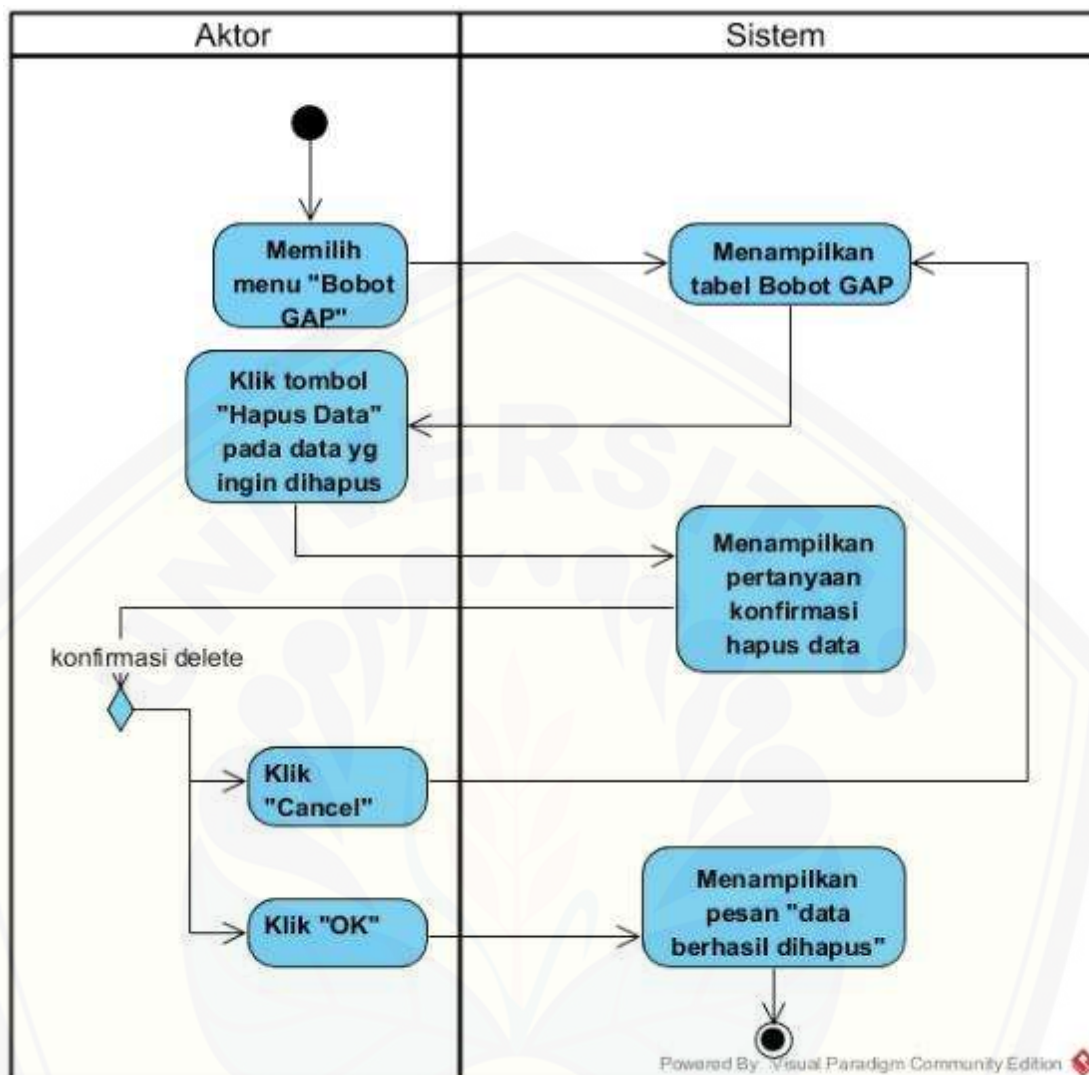
B8. Activity Diagram Pengelolaan Bobot gap



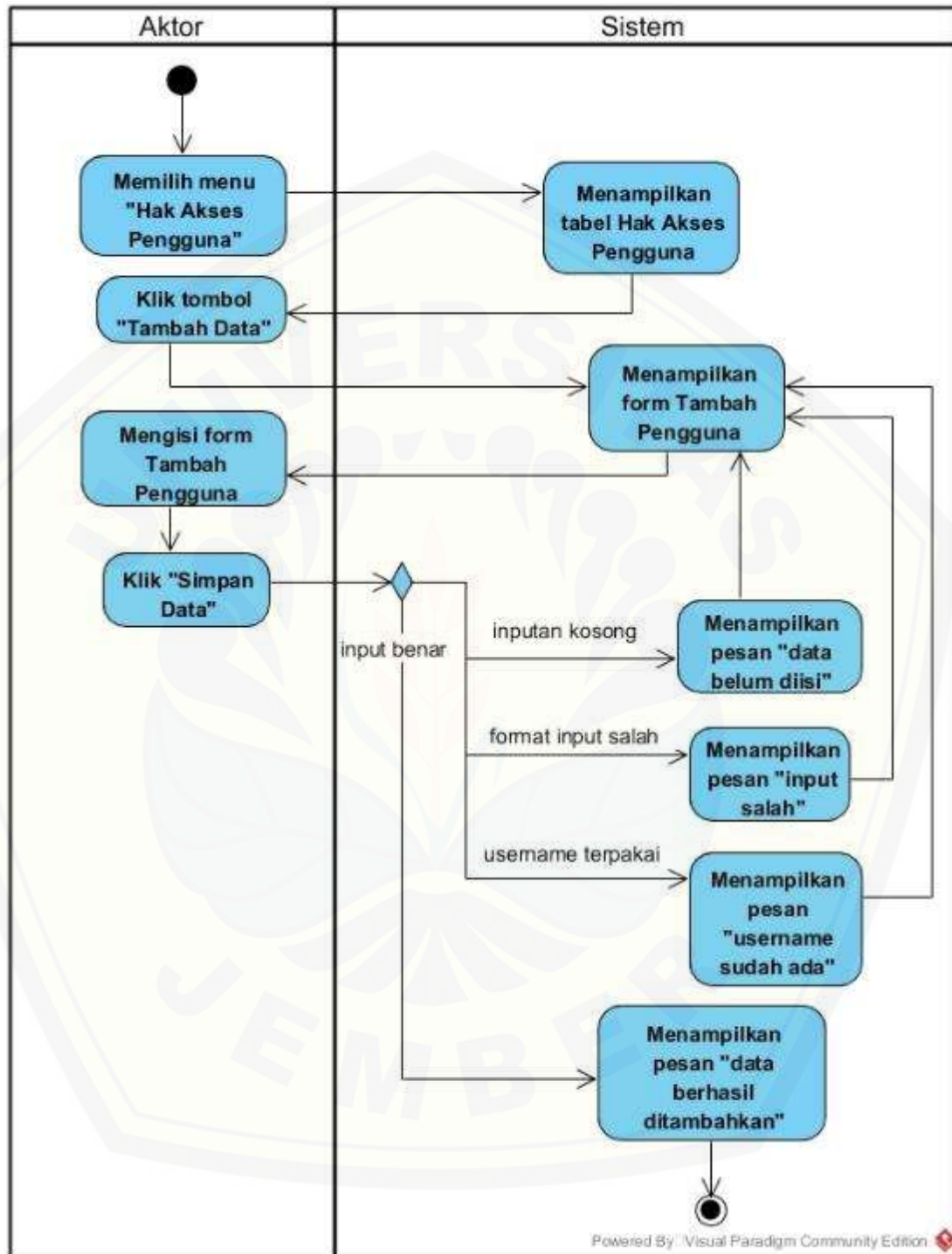
Gambar 15. Activity Diagram Pengelolaan Bobot gap (Tambah Data)



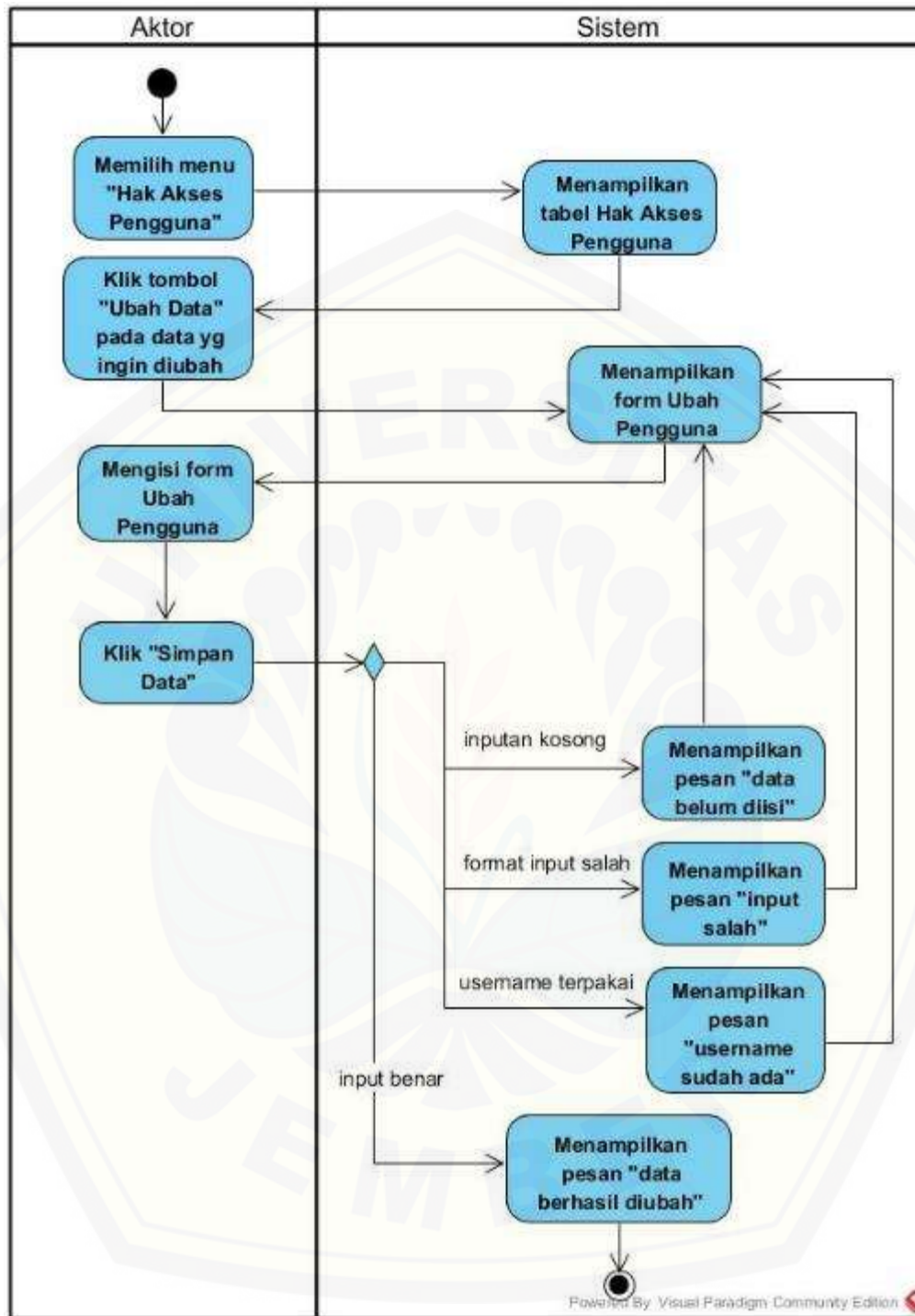
Gambar 16. Activity Diagram Pengelolaan Bobot gap (Ubah Data)



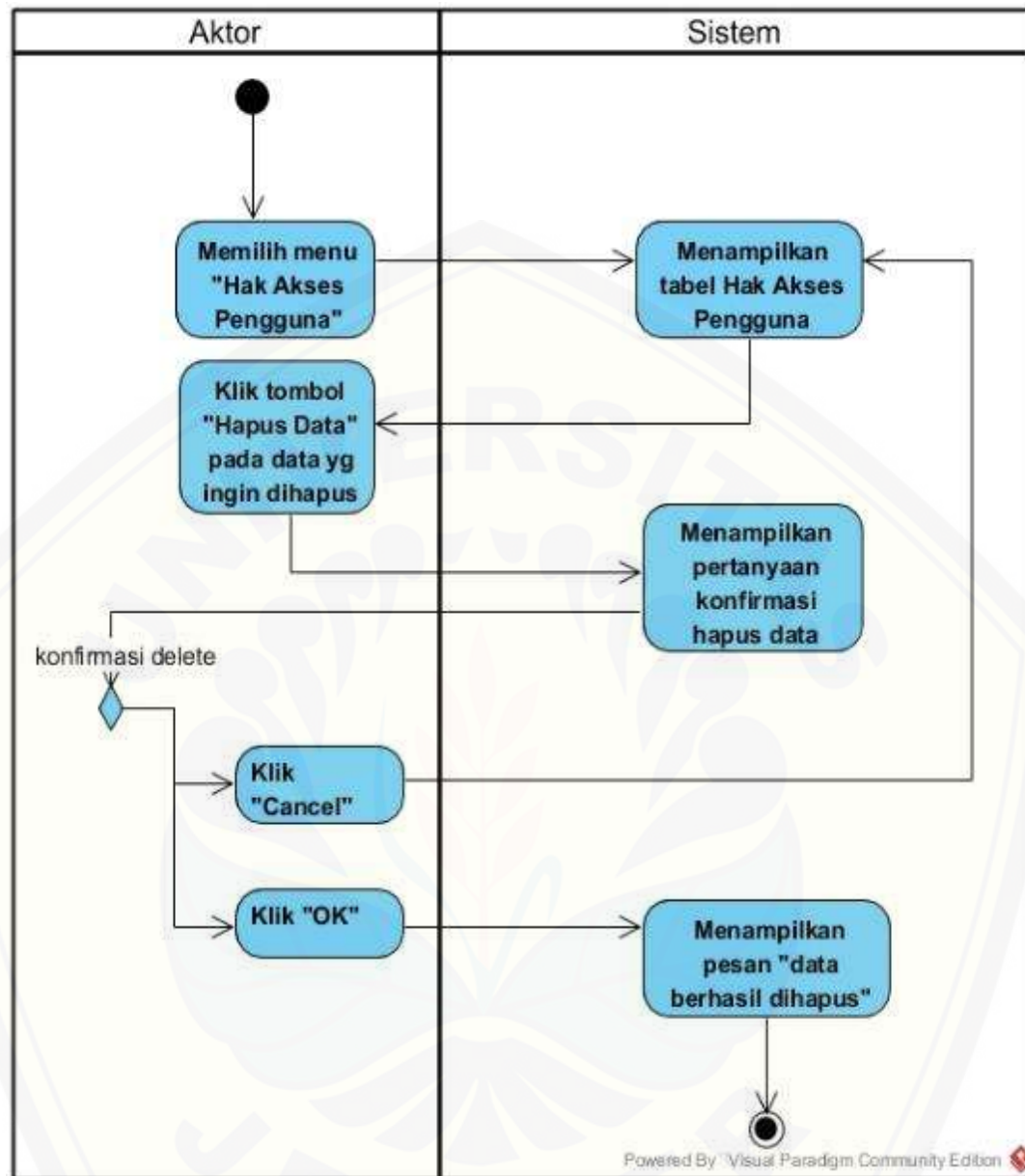
Gambar 17. Activity Diagram Pengelolaan Bobot gap (Hapus Data)

B9. Activity Diagram Pengelolaan Data Pengguna

Gambar 18. Activity Diagram Pengelolaan Data Pengguna (Tambah Data)

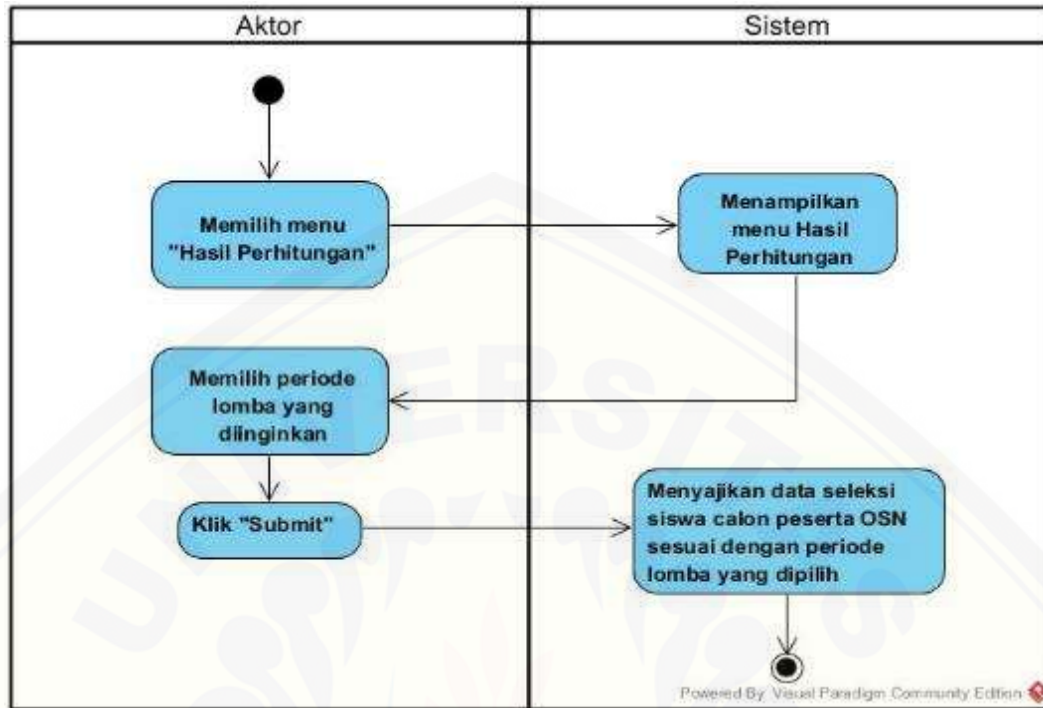


Gambar 19. Activity Diagram Pengelolaan Data Pengguna (Ubah Data)



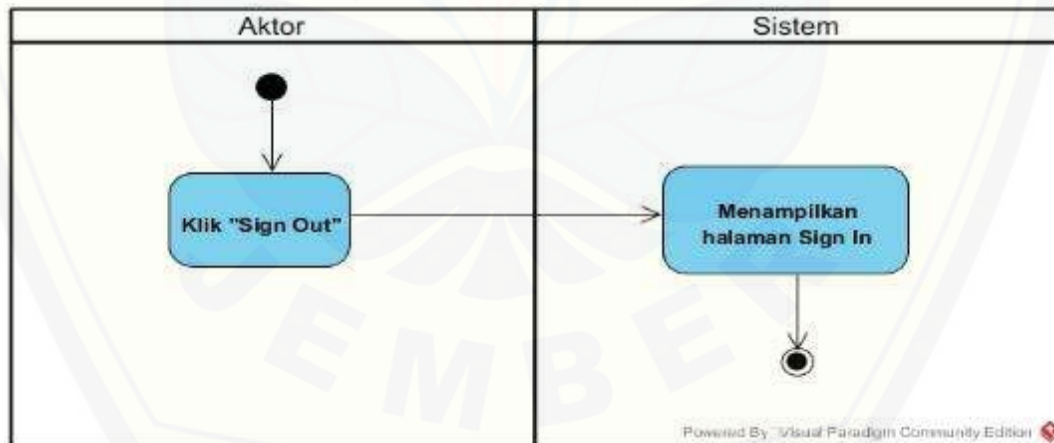
Gambar 20. Activity Diagram Pengelolaan Data Pengguna (Hapus Data)

B10. Activity Diagram View Hasil Perhitungan



Gambar 21. Activity Diagram View Hasil Perhitungan

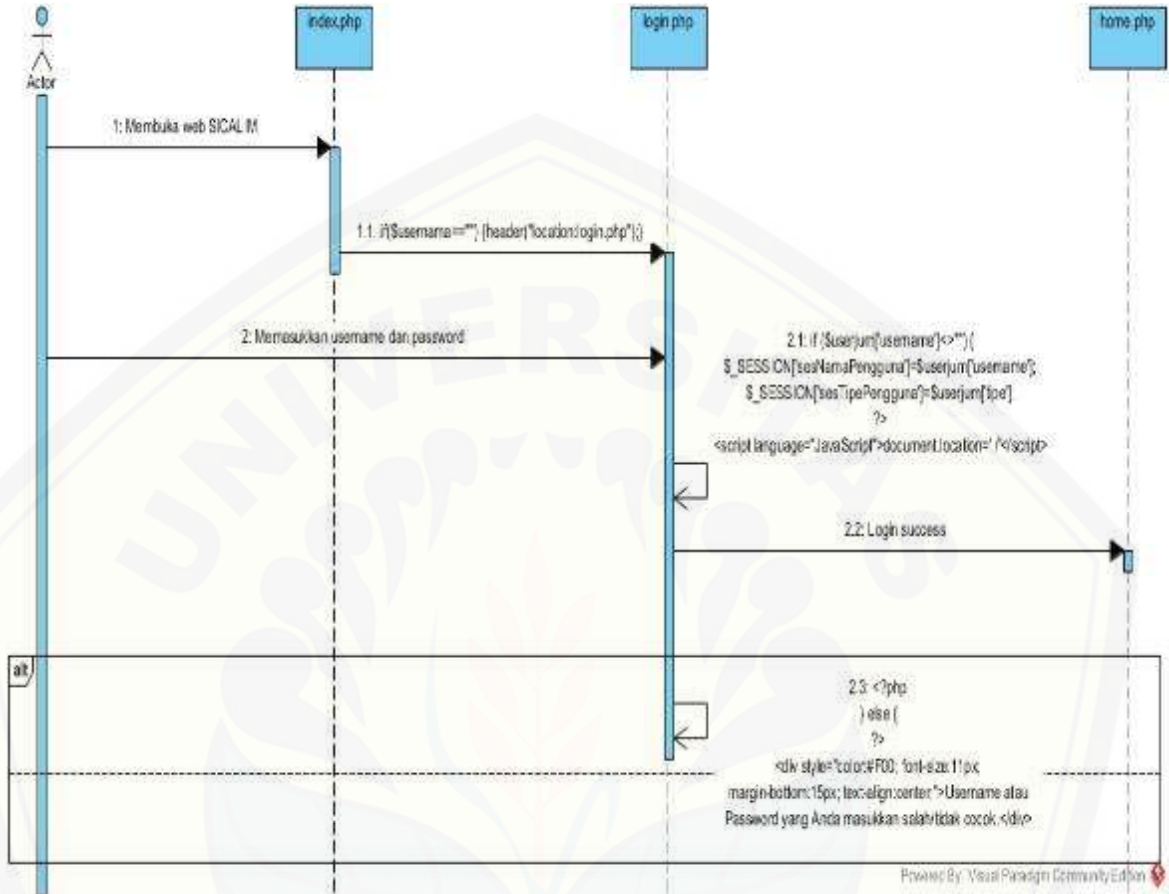
B11. Activity Diagram Sign Out



Gambar 22. Activity Diagram Sign Out

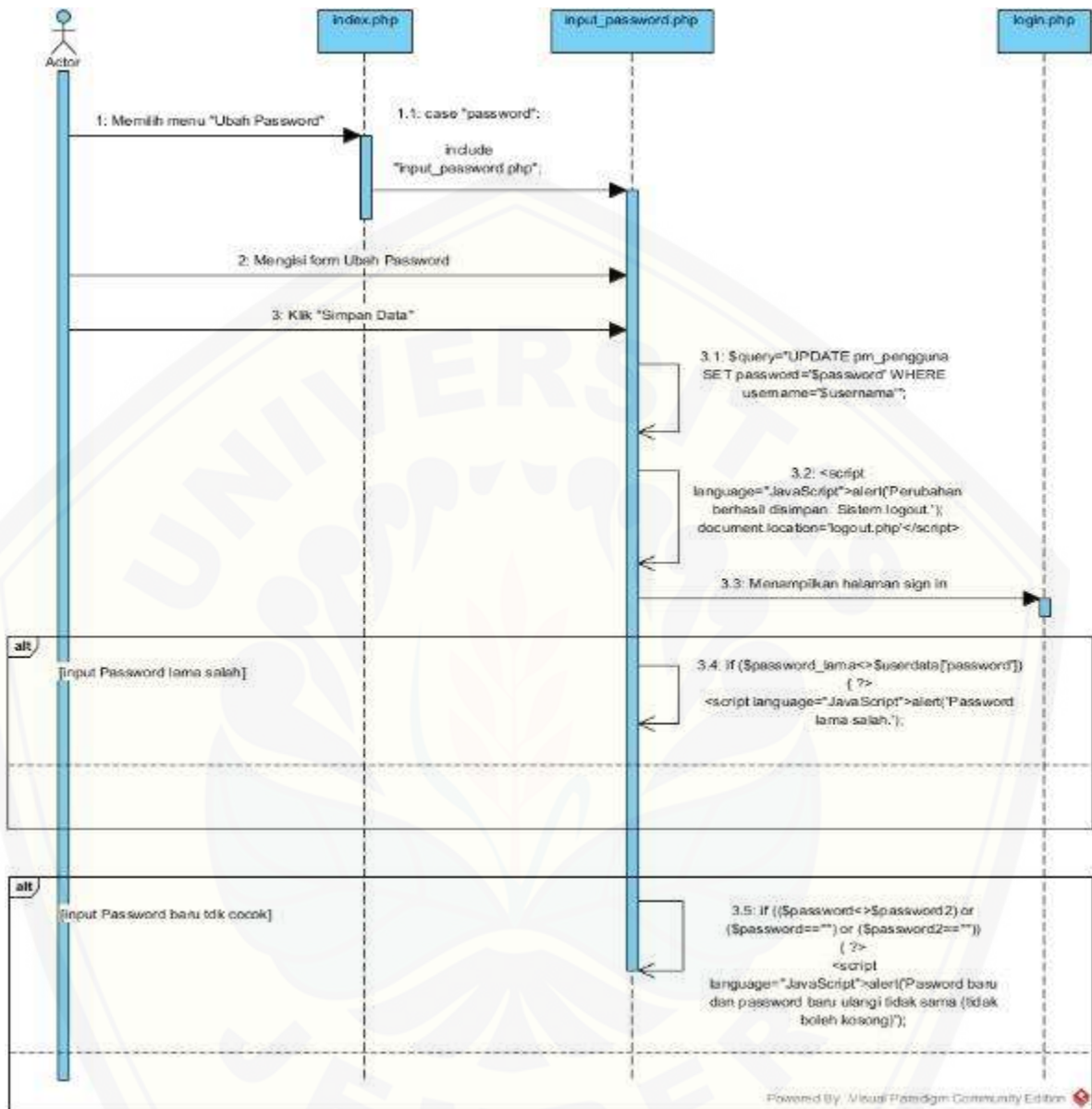
LAMPIRAN C. SEQUENCE DIAGRAM

C1. Sequence Diagram Sign In



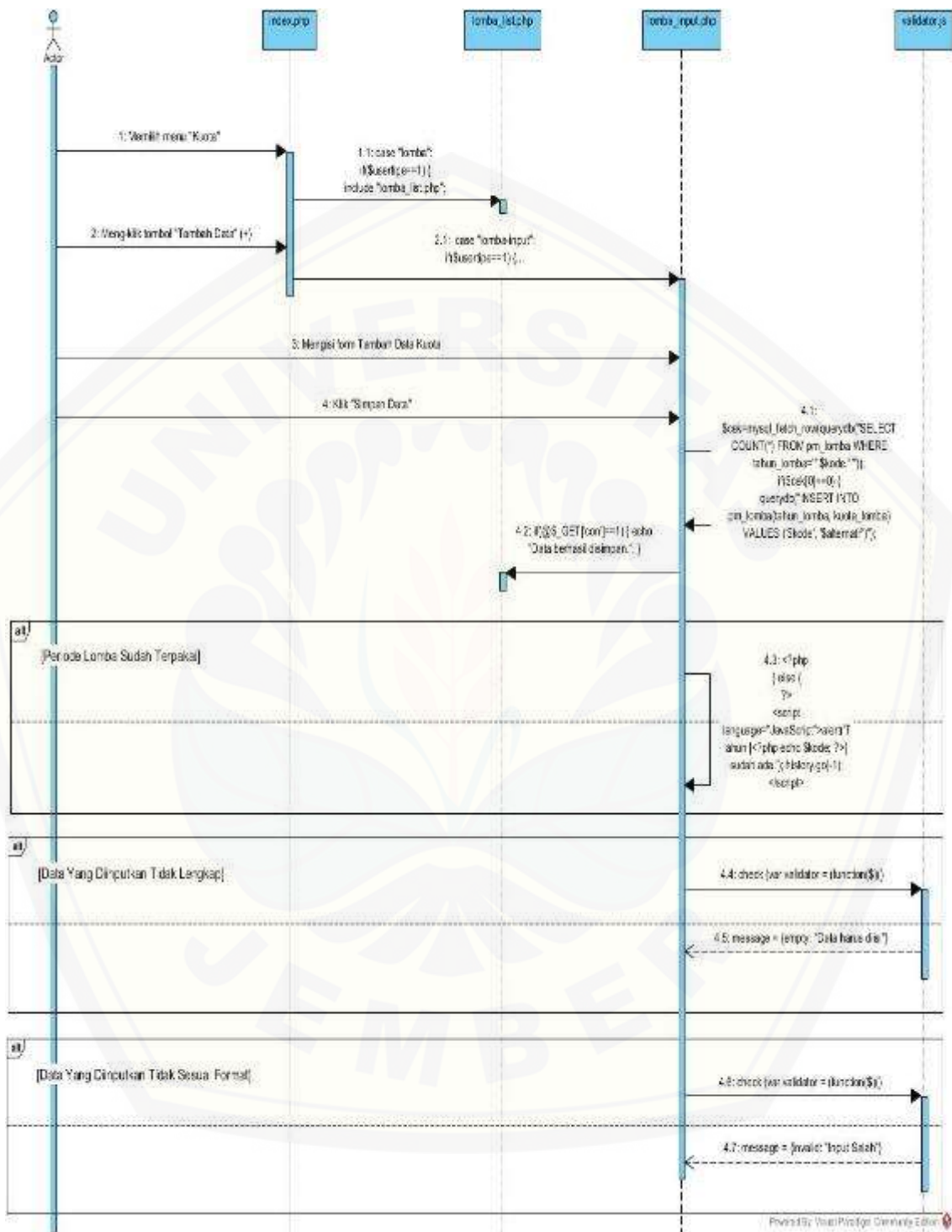
Gambar 23. Sequence Diagram Sign In

C2. Sequence Diagram Ubah Password

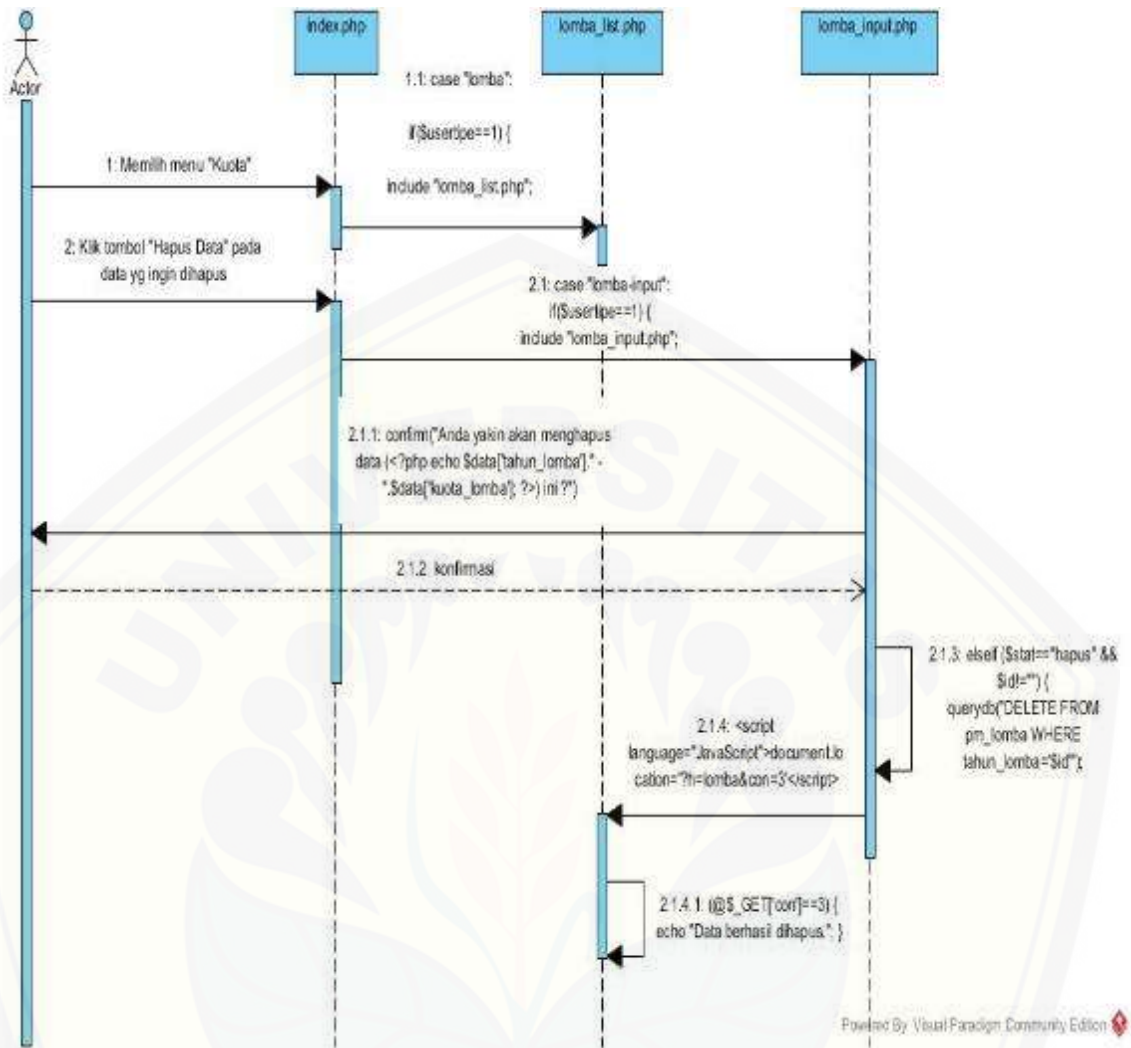


Gambar 24. Sequence Diagram Ubah Password

C3. Sequence Diagram Input Kuota Lomba

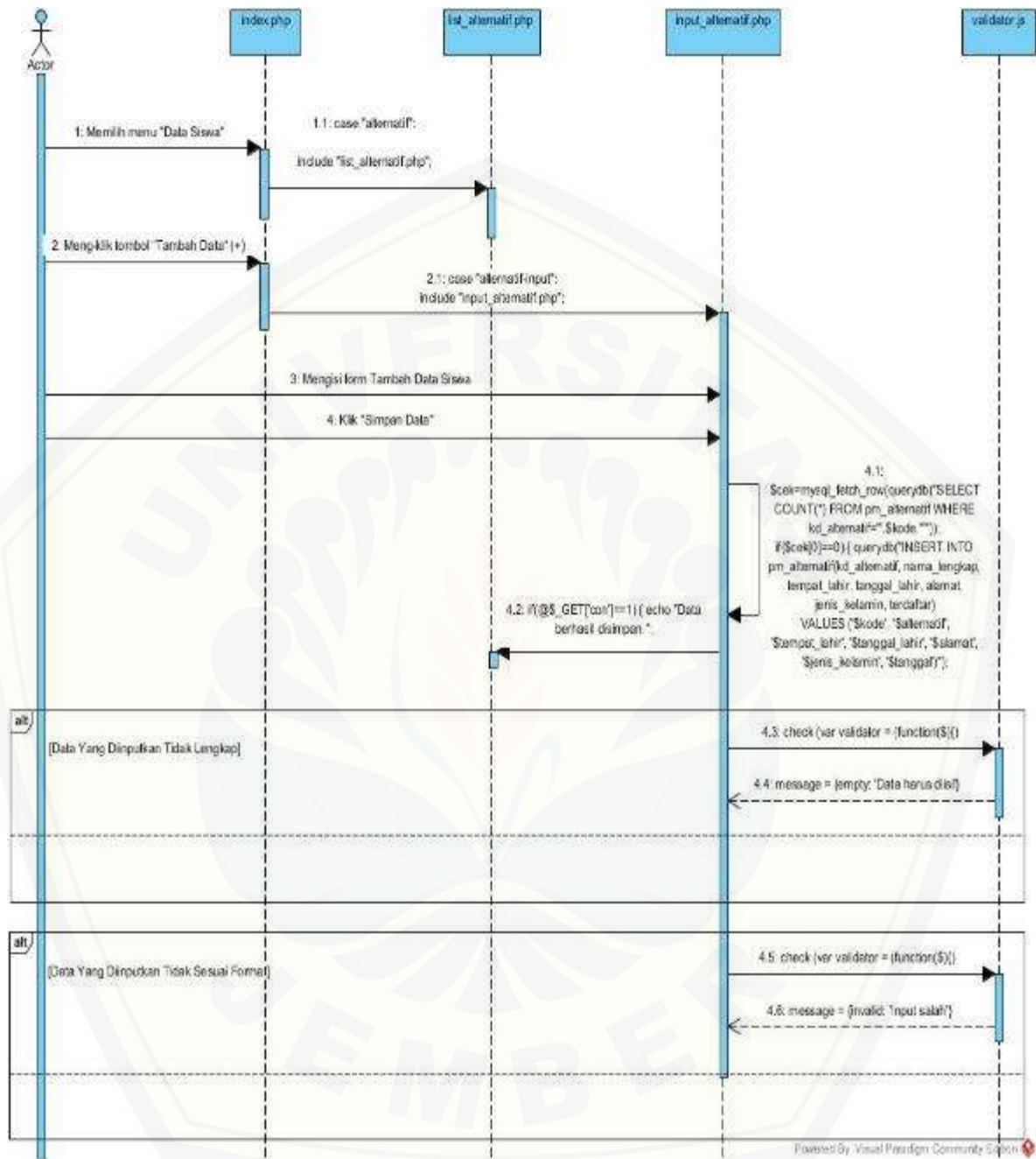


Gambar 25. Sequence Diagram Input Kuota Lomba (Tambah Data)

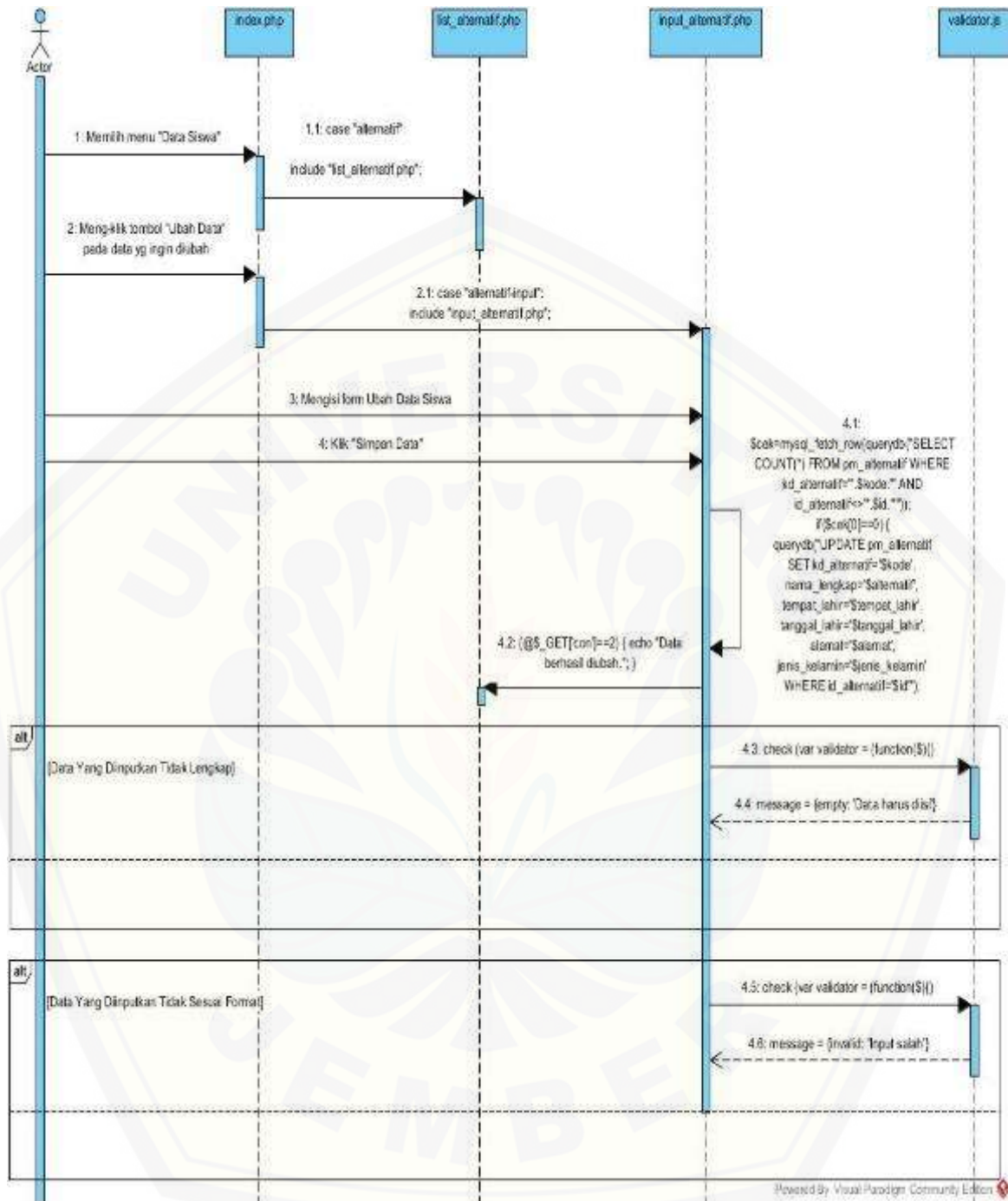


Gambar 26. Sequence Diagram Input Kuota Lomba (Hapus Data)

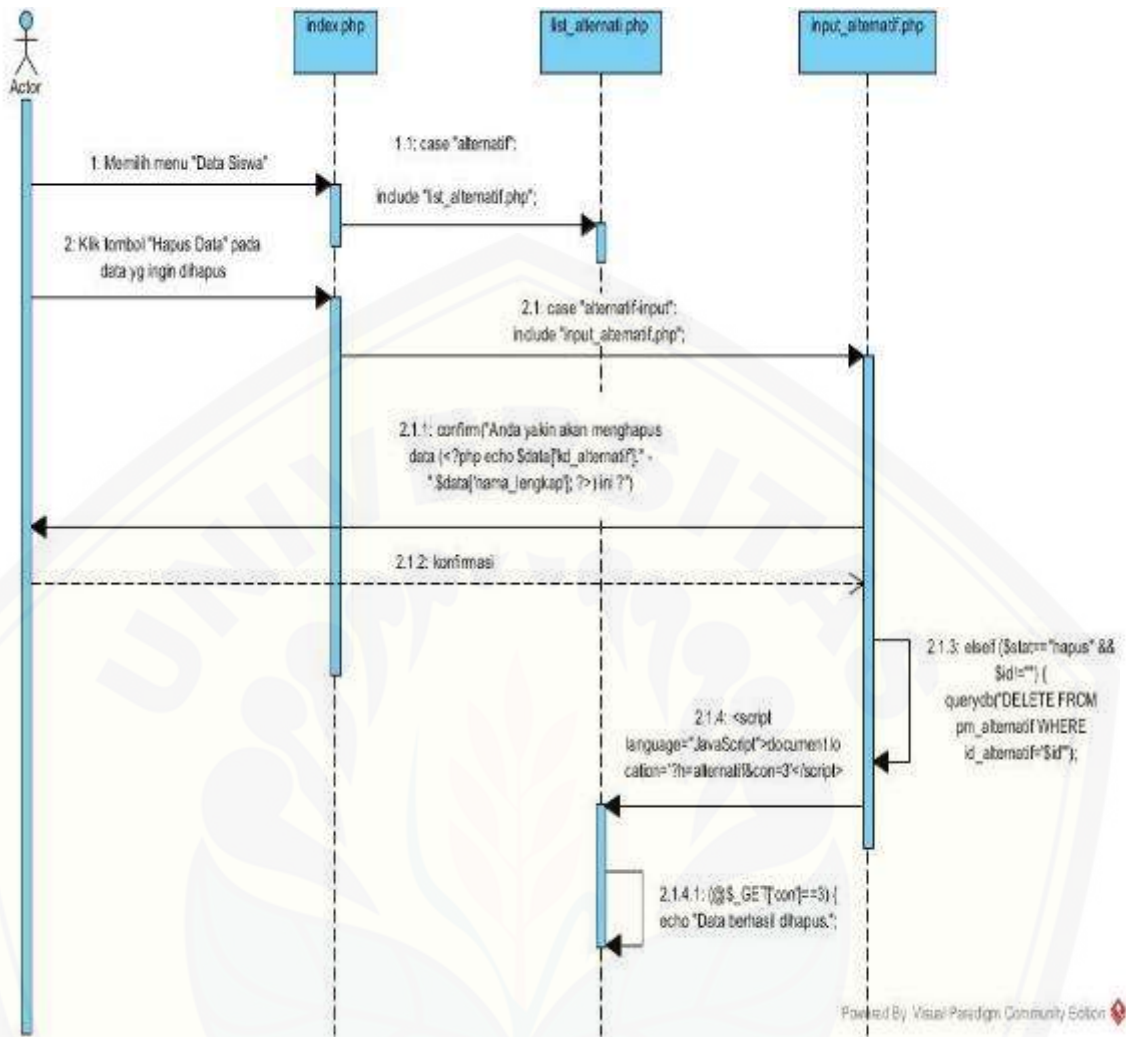
C4. Sequence Diagram Pengelolaan Data Siswa



Gambar 27. Sequence Diagram Pengelolaan Data Siswa (Tambah Data)

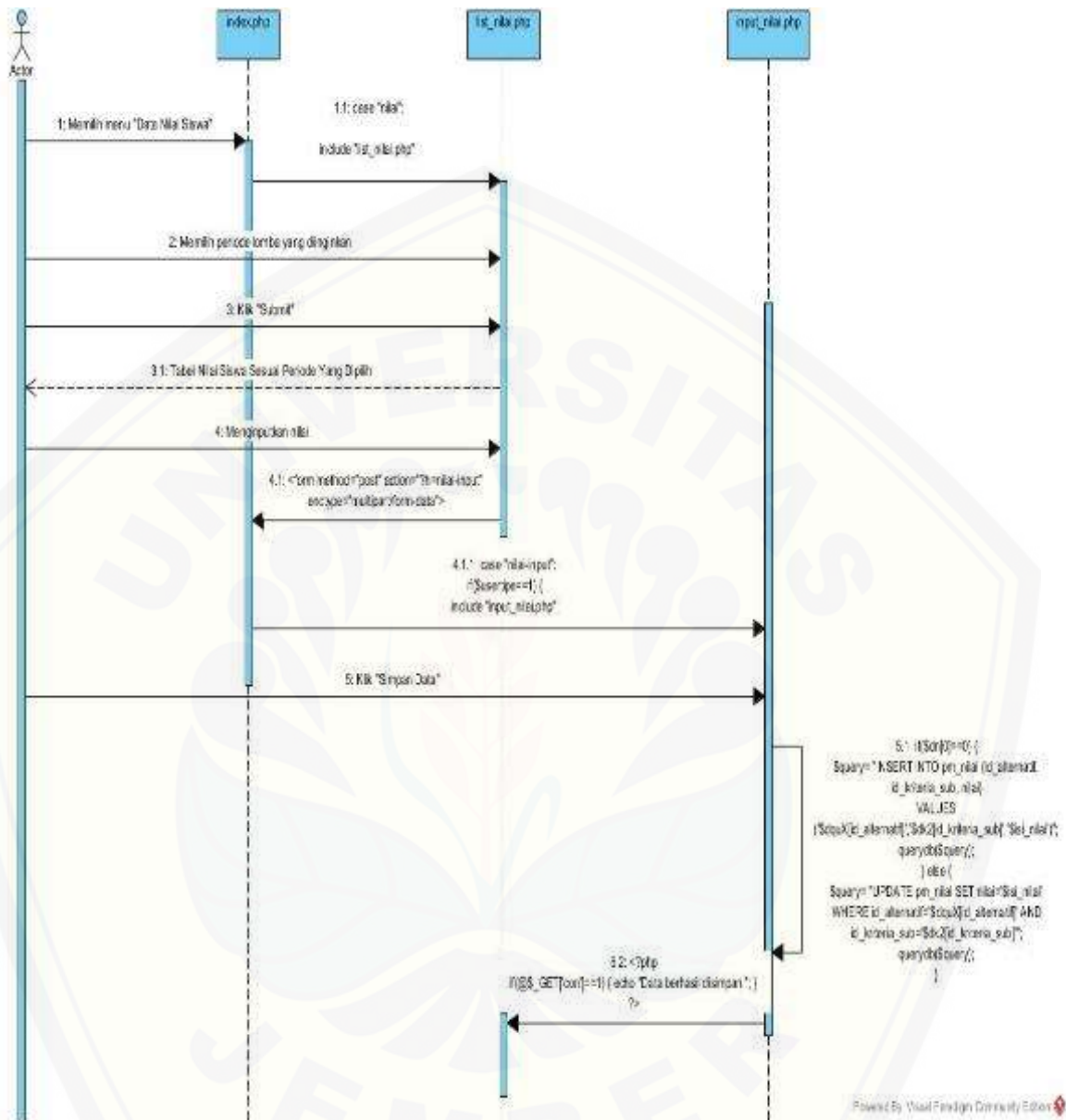


Gambar 28. Sequence Diagram Pengelolaan Data Siswa (Ubah Data)



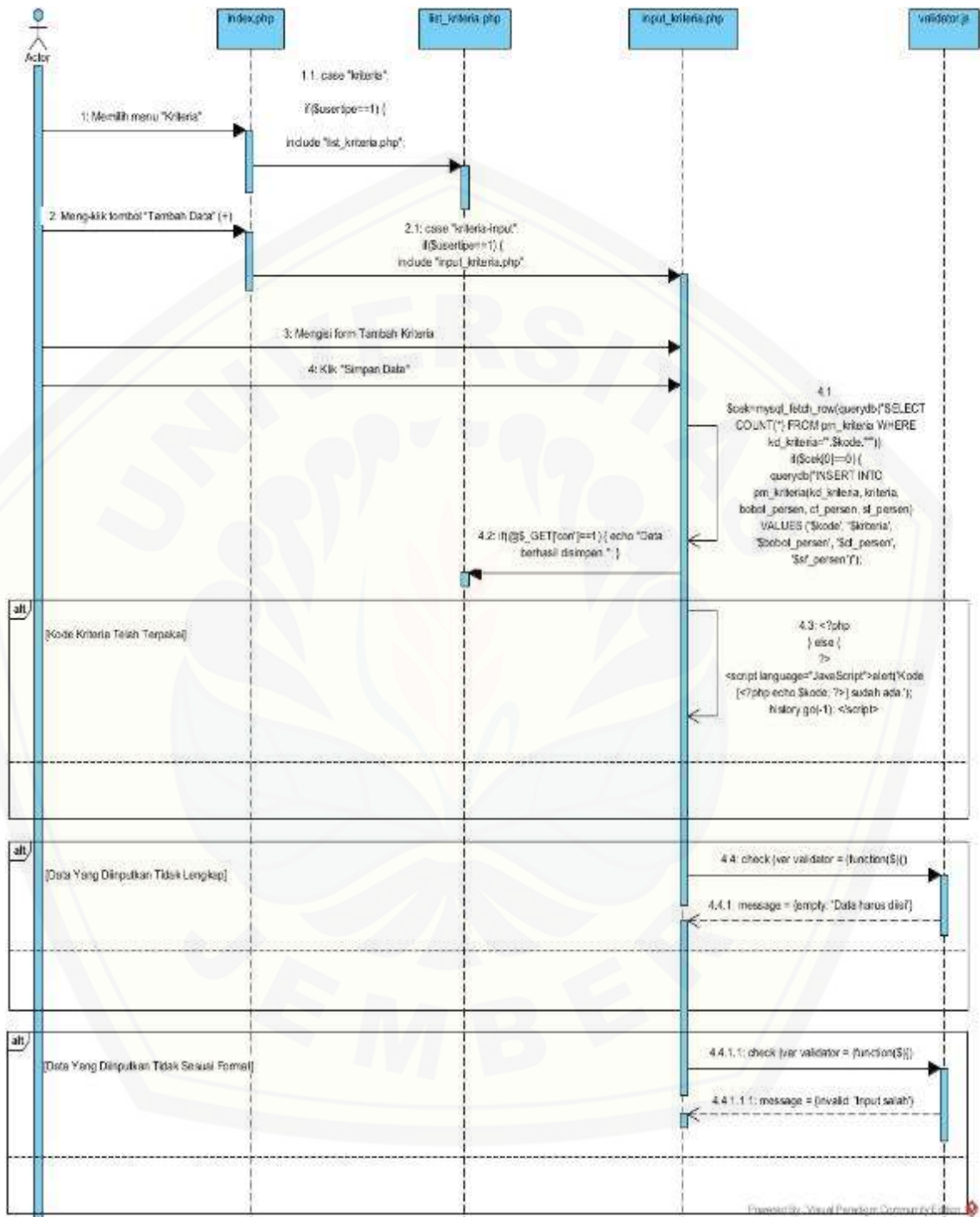
Gambar 29. Sequence Diagram Pengelolaan Data Siswa (Hapus Data)

C5. Sequence Diagram Pengelolaan Data Nilai Siswa

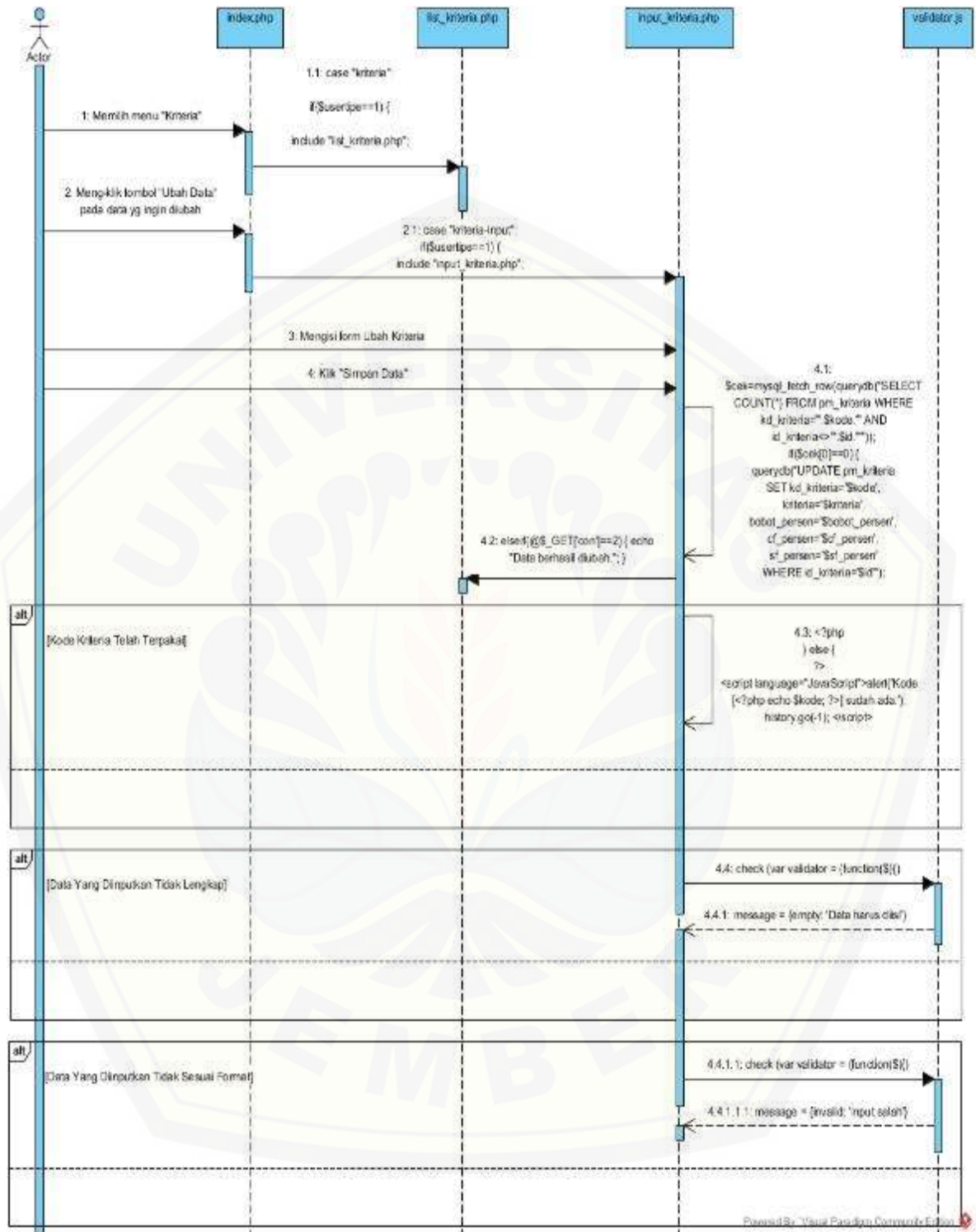


Gambar 30. Sequence Diagram Pengelolaan Data Nilai Siswa

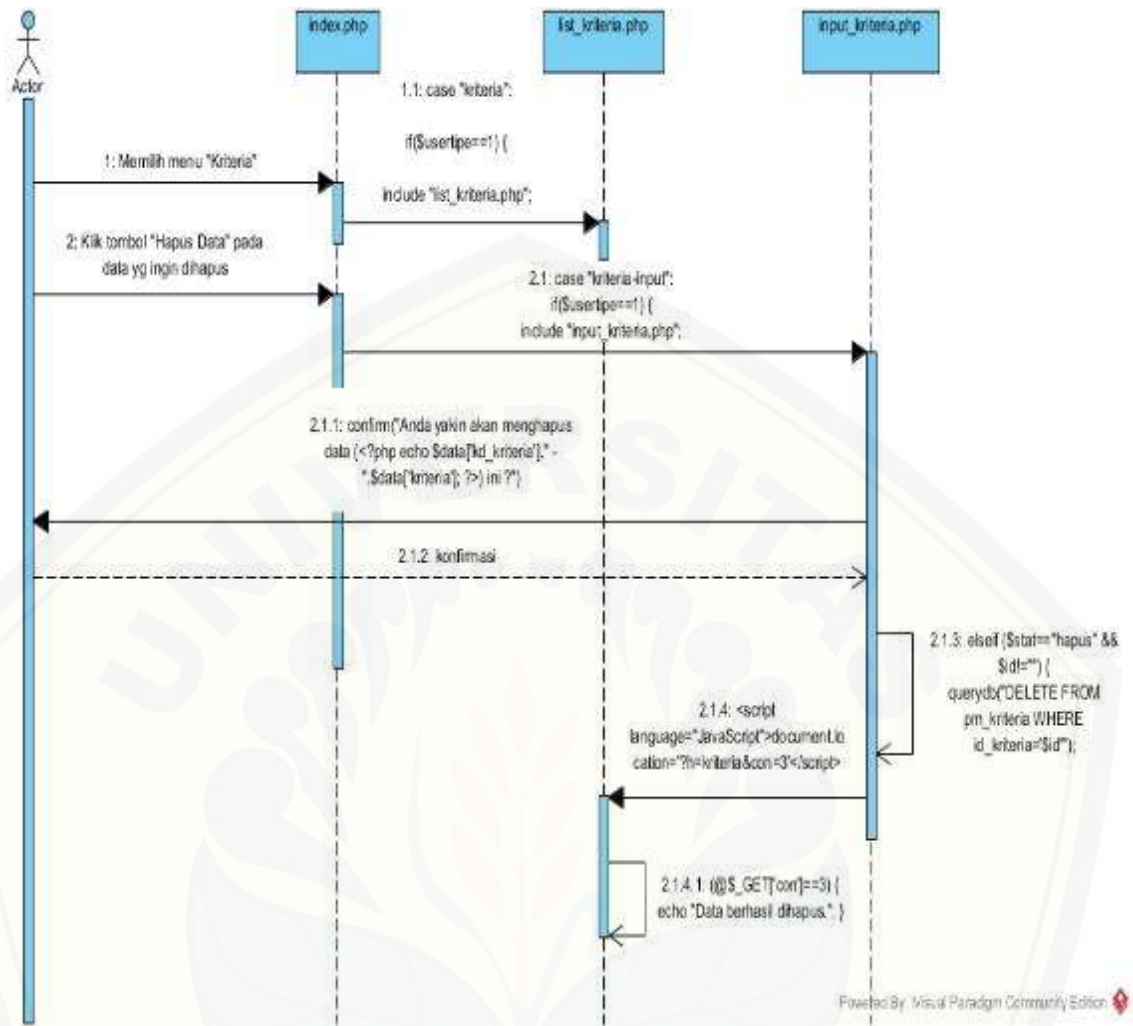
C6. Sequence Diagram Pengelolaan Data Kriteria



Gambar 31. Sequence Diagram Pengelolaan Data Kriteria (Tambah Data)

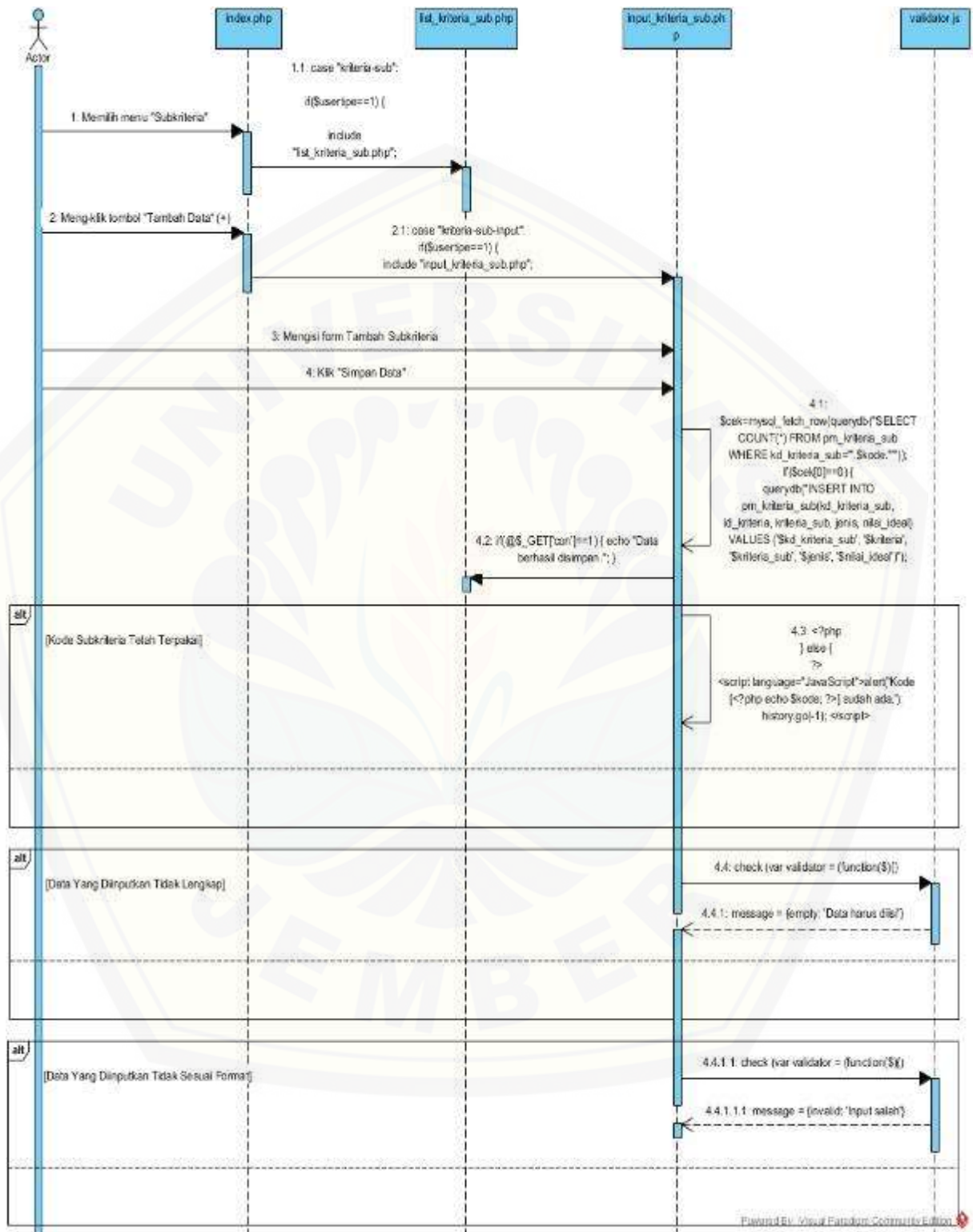


Gambar 32. Sequence Diagram Pengelolaan Data Kriteria (Ubah Data)

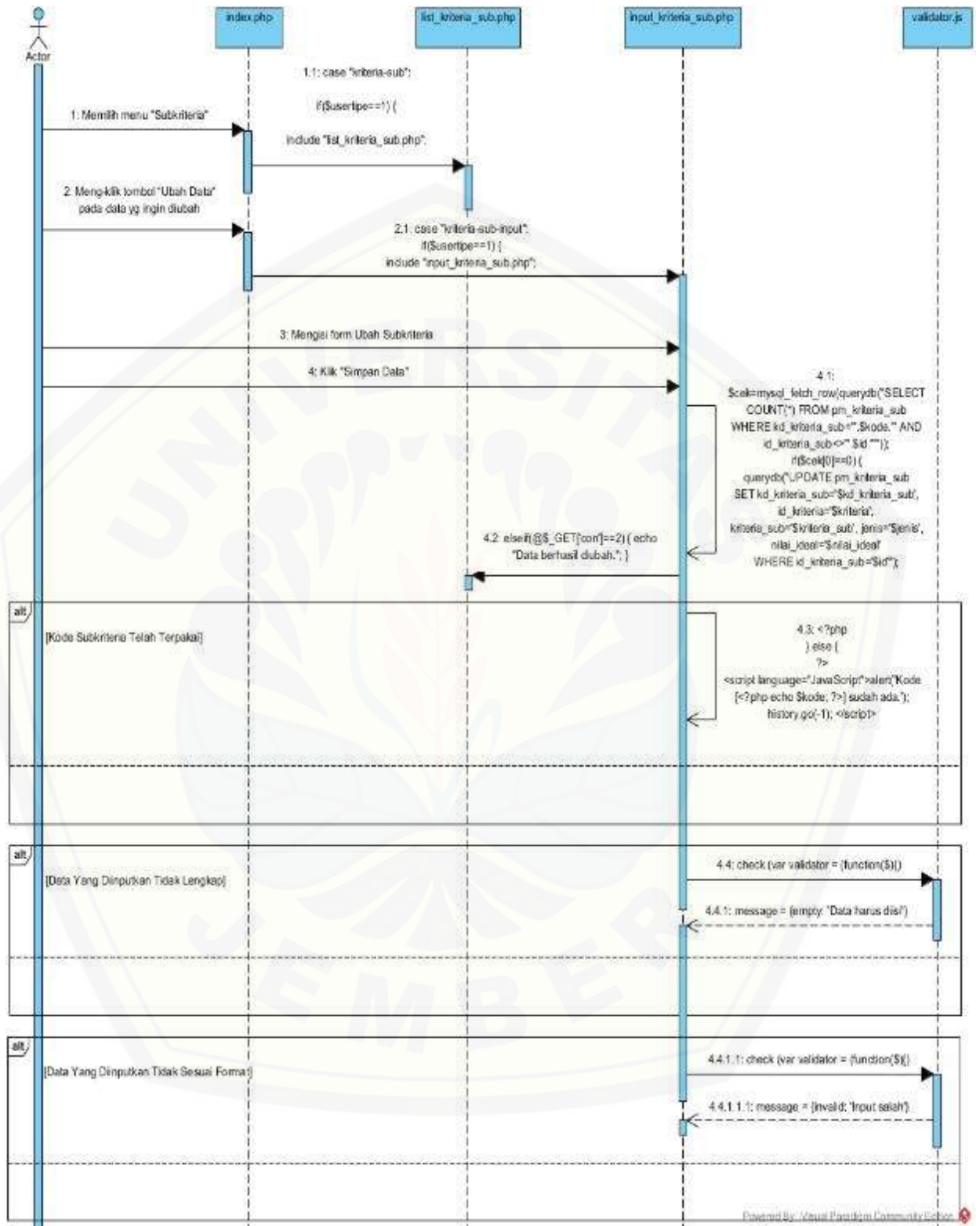


Gambar 33. Sequence Diagram Pengelolaan Data Kriteria (Hapus Data)

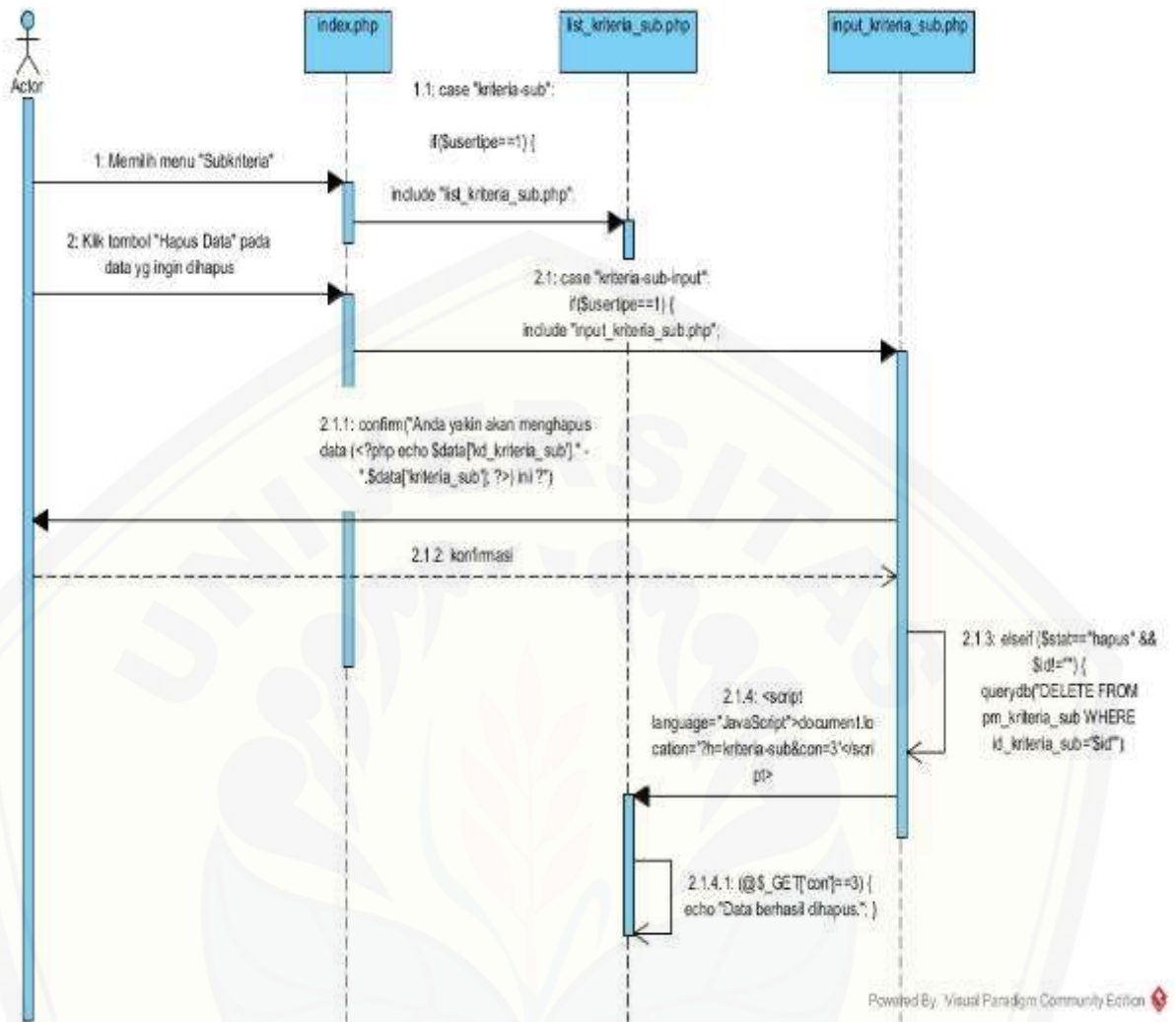
C7. Sequence Diagram Pengelolaan Data Subkriteria



Gambar 34. Sequence Diagram Pengelolaan Data Subkriteria (Tambah Data)

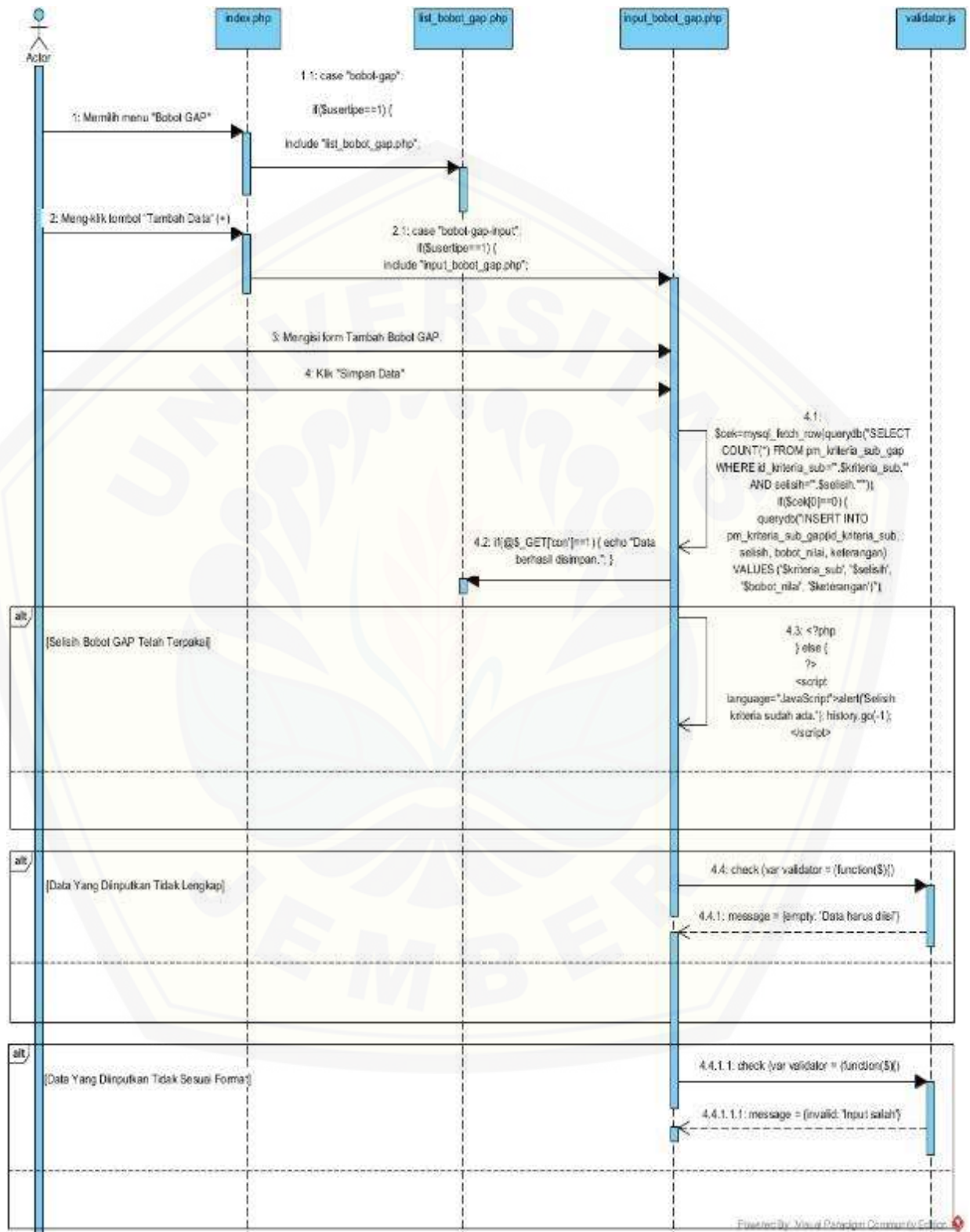


Gambar 35. Sequence Diagram Pengelolaan Data Subkriteria (Ubah Data)

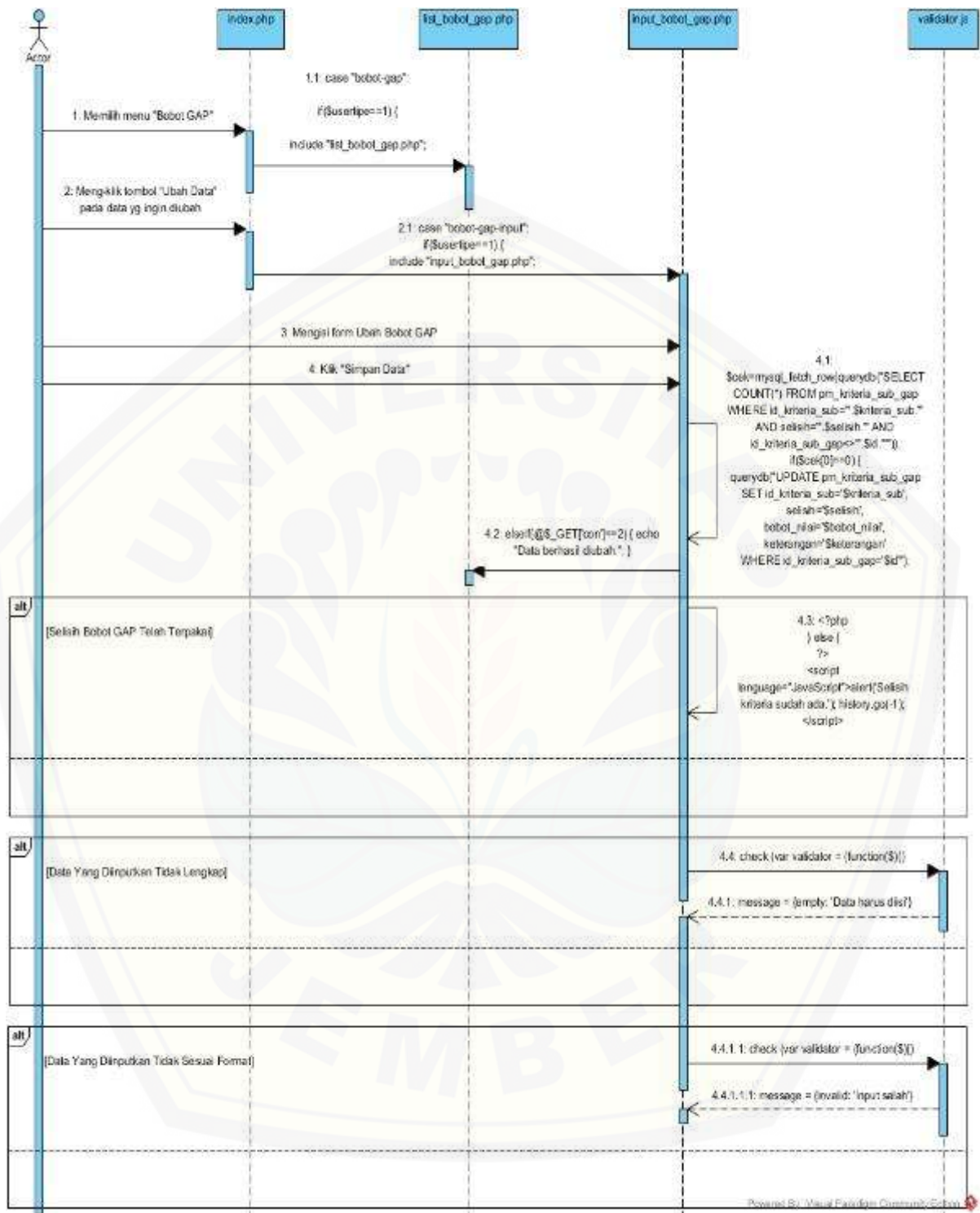


Gambar 36. Sequence Diagram Pengelolaan Data Subkriteria (Hapus Data)

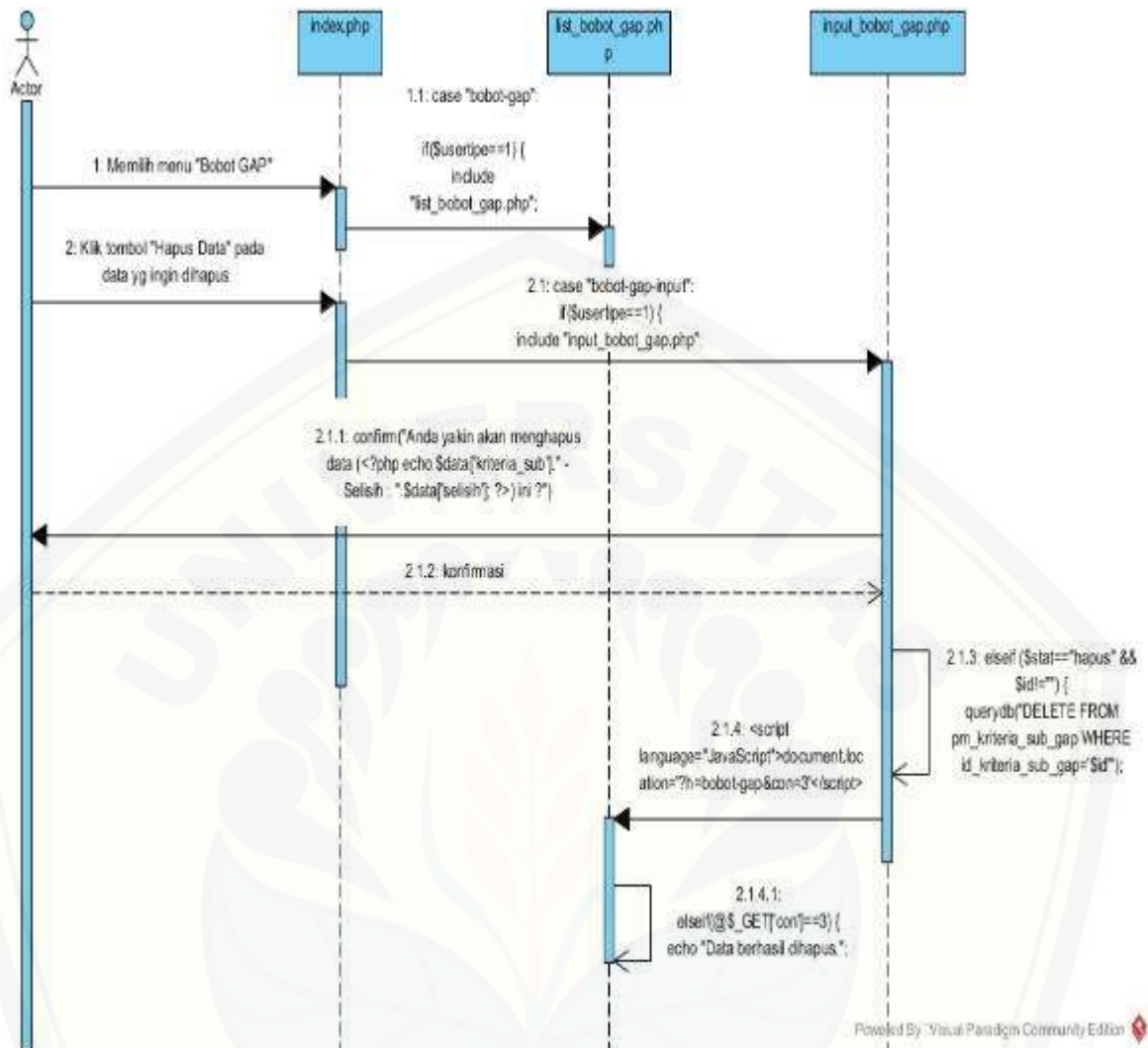
C8. Sequence Diagram Pengelolaan Bobot gap



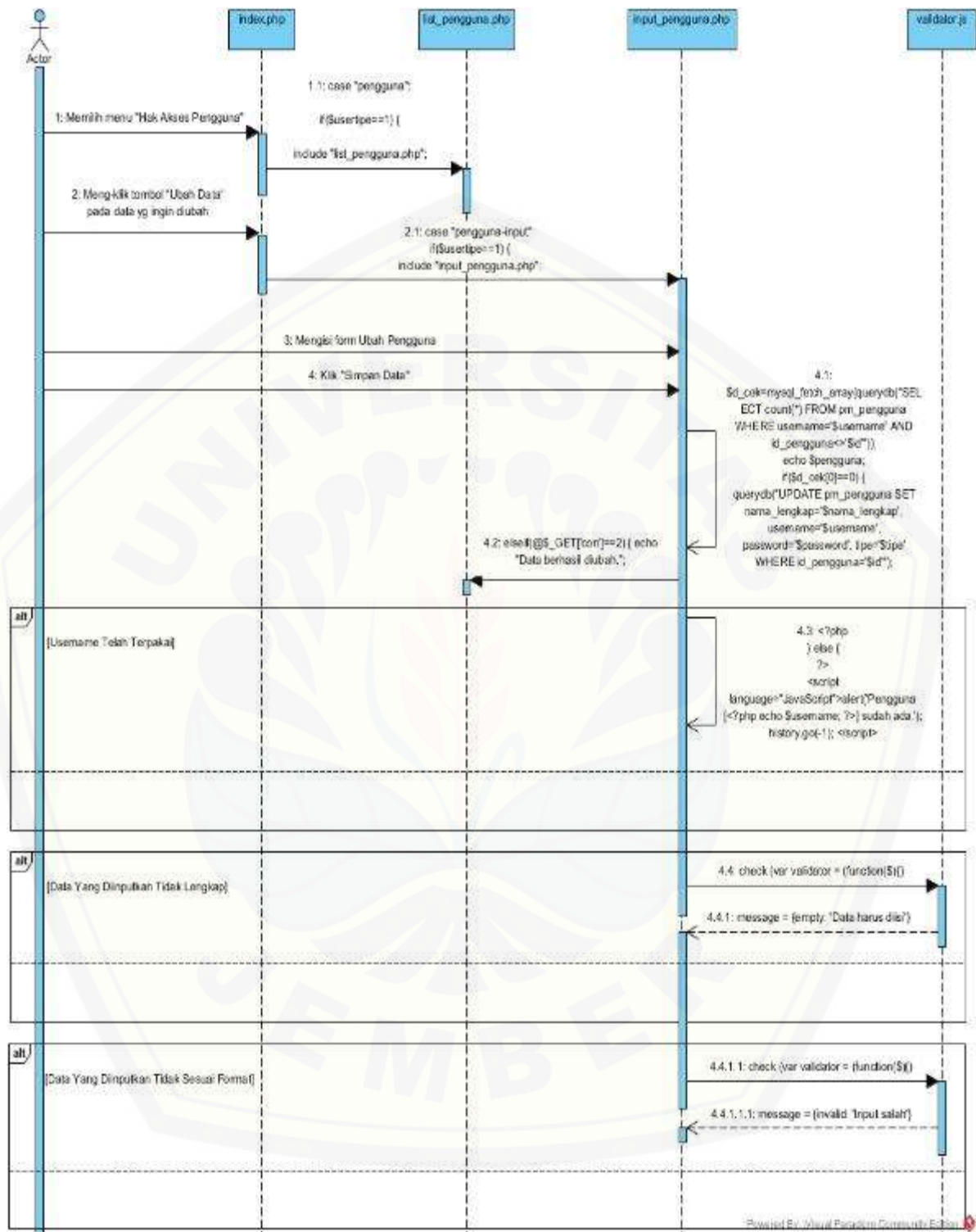
Gambar 37. Sequence Diagram Pengelolaan Bobot gap (Tambah Data)



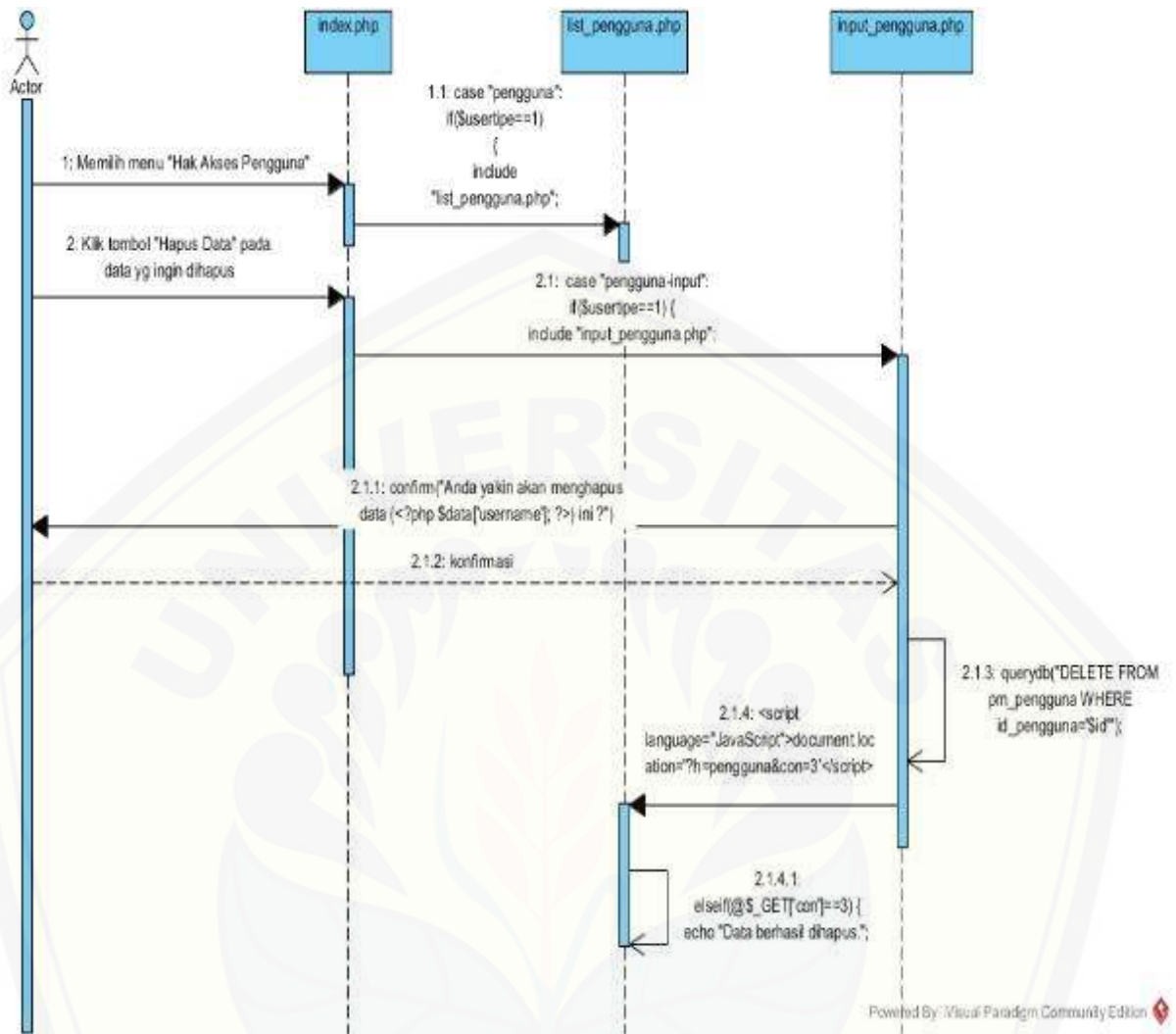
Gambar 38. Sequence Diagram Pengelolaan Bobot gap (Ubah Data)



Gambar 39. Sequence Diagram Pengelolaan Bobot gap (Hapus Data)

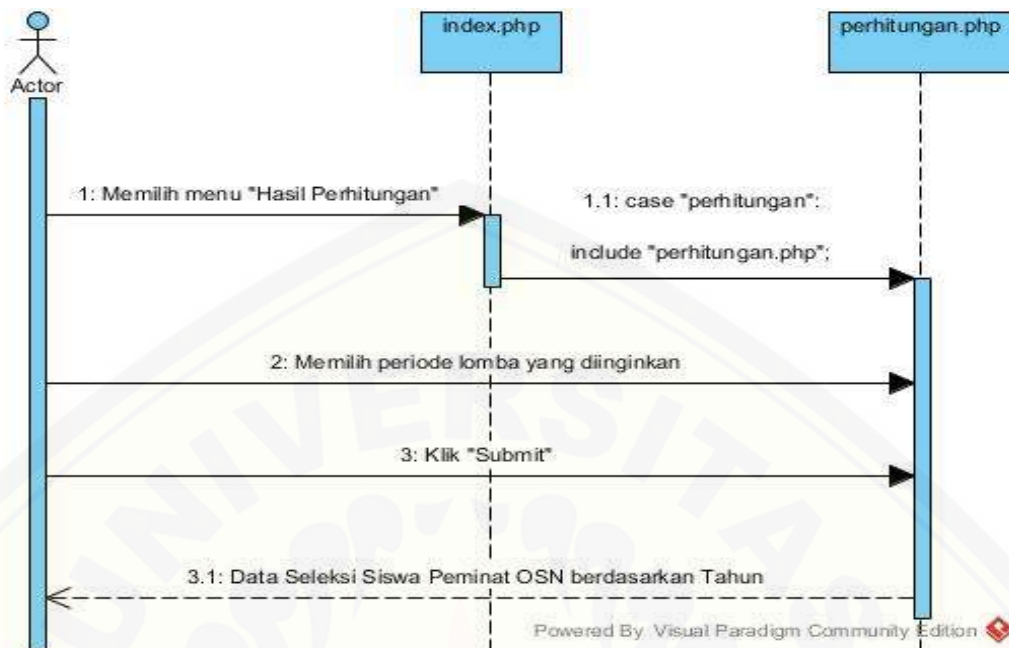


Gambar 41. Sequence Diagram Pengelolaan Data Pengguna (Ubah Data)



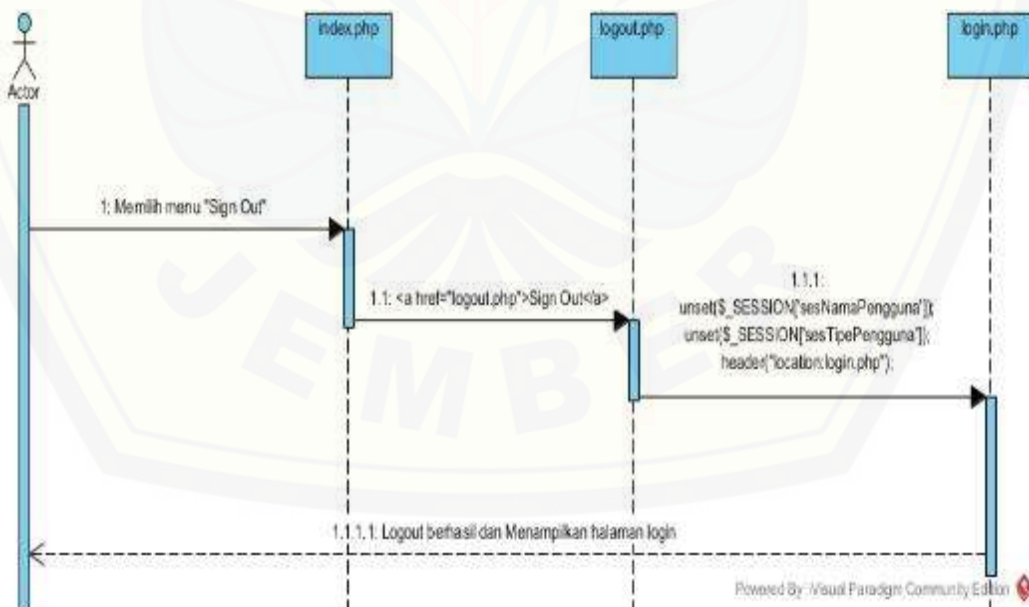
Gambar 42. Sequence Diagram Pengelolaan Data Pengguna (Hapus Data)

C10. Sequence Diagram View Hasil Perhitungan



Gambar 43. Sequence Diagram View Hasil Perhitungan

C11. Sequence Diagram Sign Out



Gambar 44. Sequence Diagram Sign Out

LAMPIRAN D. TRANSKRIP WAWANCARA

Transkrip Wawancara Pengumpulan Data Dengan Tenaga Pengajar TIK di SMAN 1 Gedangan

Nama : Ninis Herawati, M.Pd.

Jabatan : Guru TIK

Alamat : .Jl. Sedati Asri 02 No. 23, Gedangan, Sidoarjo.

Tabel 28. Transkrip Dialog Wawancara Guru TIK

Materi Wawancara	
Peneliti	Apakah di SMA ini ada metode khusus untuk menyaring calon peserta olimpiade TIK, bu?
Guru TIK	Tidak ada, biasanya kita langsung tunjuk saja siswa yang menonjol di bidang pelajaran TIK
Peneliti	Bolehkah saya meminta data nilai peminat olimpiade sains TIK tahun-tahun lalu?
Guru TIK	Ya, silahkan. Semua ada di rapor siswa.
Peneliti	Peminat tahun ini, dan tahun lalu mana saja, bu? Saya butuh untuk sampel data penelitian saya
Guru TIK	Peminat tahun ini (2018) Elistyawati Sukma C., Christian Agung S., Bryan Brajamusti, Ma'rifatul Islam, dan Agnes Dwi Aryanti, dengan 3 siswa tertunjuk berdasarkan kuota saat itu yaitu Christian Agung S., Bryan Brajamusti dan Elistyawati Sukma C.

Guru TIK	Peminat tahun lalu (2017) ada Eko Dhani R., Agus Tri S., Bambang Huda P., Budi Prasetyo, dan Ardhy Rakhmatul F., dengan 2 siswa tertunjuk berdasarkan kuota saat itu Eko Dhani dan Agus Tri.
-----------------	--

Guru TIK	Peminat tahun 2016 ada Azhar Syaikhul, Judhan Achmad N., Muhammad Jefri A., Claudya R. Kurnia, dan Santhy Setyaning R., dengan 3 siswa tertunjuk berdasarkan kuota saat itu adalah Azhar Syaikhul, Judhan Achmad, dan Claudya.
-----------------	--

Peneliti	Saya ada usul untuk menerapkan sistem penunjang keputusan berbasis komputer menggunakan metode <i>Profile Matching</i> untuk pemilihan calon peserta olimpiade, bu. Jadi untuk kedepannya, semua kriteria penilaian yang harusnya menjadi bobot-bobot tersendiri terkalkulasi dengan akurat, sehingga penentuan kandidat tidak lagi subjektif, bu.
-----------------	--

Guru TIK	Baiklah, sepertinya menarik.
-----------------	------------------------------

Peneliti	Jadi, Bu. Dari rapor siswa, parameter penilaian tentang TIK ada: Nilai Rapor TIK (Teori), dan Nilai Rapor TIK (praktik). Mungkin Ibu bisa menentukan mana yang termasuk Nilai Utama dan Nilai Tambah?
-----------------	---

Guru TIK	Kalau nilai utama (yang bobot paling besar).sebaiknya diberikan pada Nilai Rapor TIK Teori. Karena nilai ujian tulis harian maupun yang lain diperoleh pada nilai tsb. Juga tambahkan nilai kepercayaan diri (mental) pada nilai utama karena juga sangat penting.
-----------------	--

Peneliti	Berapa utk bobotnya bu?
-----------------	-------------------------

Guru TIK	Beri saja 60% untuk nilai utama
-----------------	---------------------------------

Peneliti	Baik, berarti sisanya (40%) untuk nilai tambah yakni Nilai Rapor TIK Praktik ya, bu?
Guru TIK	Ya, juga tambahkan Nilai Ekstrakurikuler TIK, Rapor Matematika Teori, dan Rapor Bahasa Inggris Teori kedalam nilai tambah, karena parameter tsb juga berkesinambungan dengan bidang lomba ini.
Peneliti	Baik, Bu. Untuk nilai ideal dari masing-masing parameter tsb berapa saja? Lalu berapa range nilai nya?
Guru TIK	Nilai Ideal sudah tentu semuanya adalah nilai maksimal (100). Range ada pada 75 hingga 100.
Peneliti	Baik, Bu. Terima kasih untuk waktunya.

Transkrip Wawancara Pengumpulan Data Dengan Tenaga Pengajar Fisika di SMAN 1 Gedangan

Nama : Soedarsono, S.Pd.
 Jabatan : Guru Fisika
 Alamat : Jl. Telaga Indah II, Ganting, Sidoarjo.

Tabel 29. Transkrip Dialog Wawancara Guru Fisika

Materi Wawancara	
Peneliti	Permisi pak, saya butuh data peserta OSN tahun ini dan tahun lalu. Mungkin bapak masih ingat siapa yang menjadi peminat dan juga terpilih waktu itu?
Guru Fisika	Peminat tahun ini (2018) adalah Danu Ega Arifin, Geovany Maulidia P., I Wayan Wahyu D., Dimas Putra Bagus, dan Eva Yulia P. Dengan 2 siswa tertunjuk berdasarkan kuota pada saat itu, yakni I Wayan dan Geovany .
Guru Fisika	Peminat tahun lalu (2017) ada Wendy Pratiwi, Asmaul Husna, Ganang Alfadin, dan Adya Wardhana dengan 2 siswa tertunjuk berdasarkan kuota pada saat itu yaitu Wendy Pratiwi dan Asmaul Husna.
Guru Fisika	Peminat tahun 2016 yaitu Reza Indra, Hendra Hidayatullah, Bimantara Hadi K., dan Galuh Putra dengan 2 siswa tertunjuk berdasarkan kuota pada saat itu yakni Hendra Hidayatullah dan Bimantara Hadi K.
Peneliti	Baik pak, jadi saya ingin menerapkan sistem penunjang keputusan berbasis komputer metode <i>Profile Matching</i> untuk menyeleksi calon peserta olimpiade fisika.

	Mungkin bapak bisa menentukan pembobotannya?
Guru Fisika	Ok.
Peneliti	Jadi pak, di rapor yang berkenaan dengan nilai Fisika ada Nilai Fisika Teori dan Praktik, kira-kira mana yang termasuk nilai utama dan nilai tambah?
Guru Fisika	Nilai Utama jelas tentu Teori, karena bahan ujian OSN fisika adalah teori, sedangkan untuk nilai tambah selain fisika praktik tambahkan juga Rapor Matematika Teori, ya. Karena matematika di OSN Fisika akan sangat membantu sekali.
Peneliti	Apakah nilai kepercayaan diri siswa tidak perlu ditambahkan?
Guru Fisika	Ya saya lupa, karena penting masukkan saja pada Nilai Utama
Peneliti	Baik, pak. Untuk pembobotannya bagaimana?
Guru Fisika	Nilai utama 60%, nilai tambah 40%
Peneliti	Lalu satu lagi pak, utk penentuan nilai ideal dan range nilainya pak
Guru Fisika	Nilai ideal semuanya 100. Range biasanya 75 tidak pernah kurang dari itu hingga maksimal yaitu 100
Peneliti	Baik pak, terima kasih.
