



**PENERAPAN METODE AHP DALAM SISTEM INFORMASI PEMETAAN
SEBARAN STRATA DESA SIAGA AKTIF
DINAS KESEHATAN KABUPATEN JEMBER**

SKRIPSI

Oleh
Rizka Rohmatul Fitriah
NIM 112410101001

**PROGRAM STUDI SISTEM INFORMASI
UNIVERSITAS JEMBER
2015**



**PENERAPAN METODE AHP DALAM SISTEM INFORMASI PEMETAAN
SEBARAN STRATA DESA SIAGA AKTIF
DINAS KESEHATAN KABUPATEN JEMBER**

SKRIPSI

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

Oleh

Rizka Rohmatul Fitriah

NIM 112410101001

**PROGRAM STUDI SISTEM INFORMASI
UNIVERSITAS JEMBER**

2015

PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

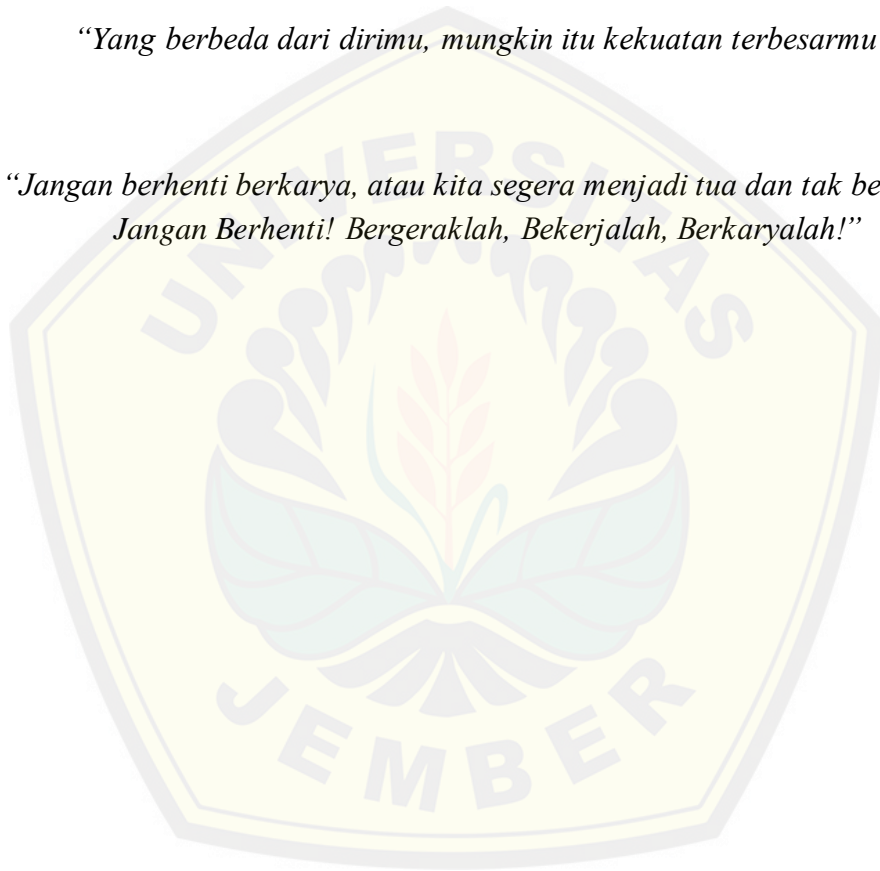
1. Allah SWT yang senantiasa memberikan nikmat, rahmat, dan karunia-Nya sehingga saya dapat terus berkarya hingga saat ini;
2. Ibunda Siti Latifah, S.Pd serta Ayahanda Drs. Miswanto atas segala kesabaran, keikhlasan, limpahan kasih sayang, doa, dan motivasi serta dukungan yang luar biasa selama hidup saya;
3. Adik-adik saya, Ahmad Zaki Abdurrahman dan Sularso;
4. Zulkarnenri Ponso, Alm. Abdul Qodir serta seluruh guru dalam hidup saya hingga saat ini, baik secara formal maupun non formal;
5. Keluarga serta seluruh pihak yang telah memberikan kontribusi dalam penulisan skripsi ini;
6. Ike Fatmawati, Keluarga Cendana Junior (Pakdhe Kuslan, Budi, Hiqmi, Yefi), Rurin, Laili Zakia, Novi Dama, Wahyu, dan M. Binawan, sosok-sosok inspiratif yang telah menjadi partner luar biasa di setiap bagian dalam hidup saya;
7. Almamater Program Studi Sistem Informasi Universitas Jember.

MOTTO

“Semangat terus belajar, jangan sampai berhenti. Kalau berhenti, kamu mati”

“Yang berbeda dari dirimu, mungkin itu kekuatan terbesarmu”

*“Jangan berhenti berkarya, atau kita segera menjadi tua dan tak berguna.
Jangan Berhenti! Bergeraklah, Bekerjalah, Berkaryalah!”*



PERNYATAAN

Saya yang bertanda tangan di bawah ini:

nama : Rizka Rohmatul Fitriah

NIM : 112410101001

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Penerapan Metode Ahp Dalam Sistem Informasi Pemetaan Sebaran Trata Desa Siaga Aktif Dinas Kesehatan Kabupaten Jember” adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada instansi mana pun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

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

Jember, Juni 2015

Yang menyatakan,

Rizka Rohmatul Fitriah

NIM 112410101001

SKRIPSI

**PENERAPAN METODE AHP DALAM SISTEM INFORMASI PEMETAAN
SEBARAN STRATA DESA SIAGA AKTIF
DINAS KESEHATAN KABUPATEN JEMBER**

Oleh

Rizka Rohmatul Fitriah

NIM 112410101001

Pembimbing

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

NIP. 196909281993021001

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

NIP. 198201012010121004

PENGESAHAN PEMBIMBING

Skripsi berjudul Penerapan Metode AHP dalam Sistem Informasi Pemetaan Sebaran Strata Desa Siaga Aktif Dinas Kesehatan Kabupaten Jember telah diuji dan disahkan pada:

hari, tanggal : 29 Juni 2015

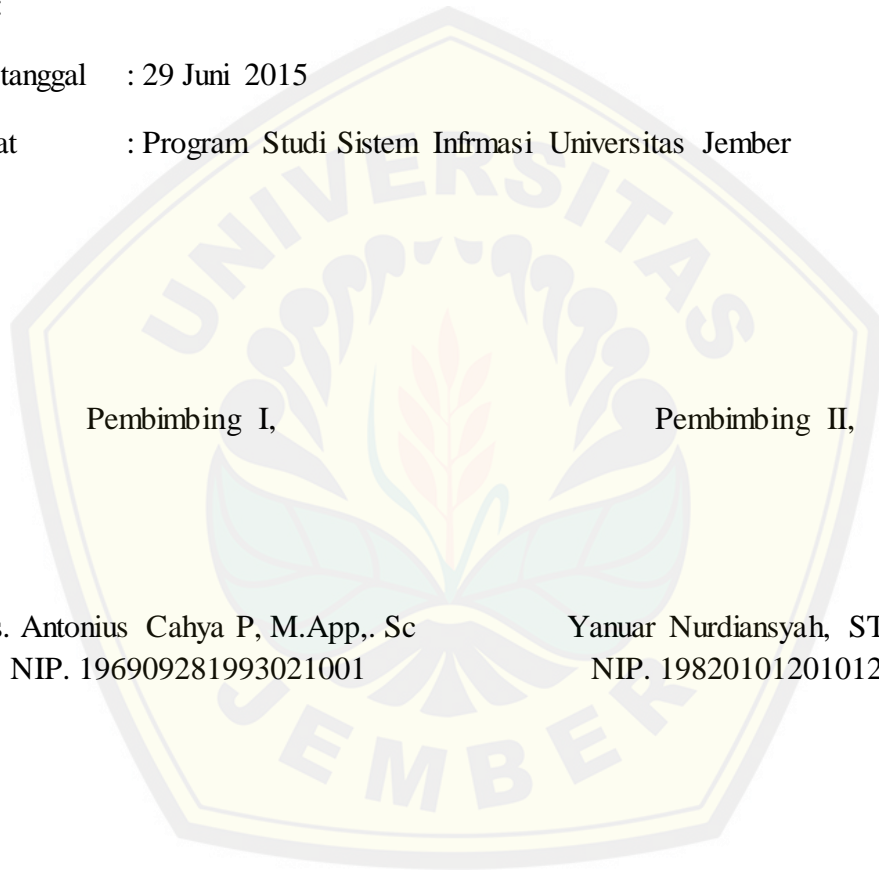
tempat : Program Studi Sistem Infrmasi Universitas Jember

Pembimbing I,

Pembimbing II,

Drs. Antonius Cahya P, M.App., Sc
NIP. 196909281993021001

Yanuar Nurdiansyah, ST, M.Cs.
NIP. 198201012010121004



PENGESAHAN PENGUJI

Skripsi berjudul Penerapan Metode AHP Dalam Sistem Informasi Pemetaan Sebaran Trata Desa Siaga Aktif Dinas Kesehatan Kabupaten Jember telah diuji dan disahkan pada:

hari, tanggal : 29 Juni 2015

tempat : Program Studi Sistem Infrmasi Universitas Jember

Tim Penguji:

Penguji I,

Penguji II,

Prof. Drs. Slamir, M.Comp.Sc, Ph.D
NIP. 196704201992011001

M. Arief Hidayat, S.Kom., M.Kom
NIP. 198101232010121003

Mengesahkan
Ketua Program Studi,

Prof. Drs. Slamir, M.Comp.Sc, Ph.D
NIP 196704201992011001

Penerapan Metode AHP Dalam Sistem Informasi Pemetaan Sebaran Trata Desa Siaga Aktif Dinas Kesehatan Kabupaten Jember

Rizka Rohmatul Fitriah

Jurusan Sistem Informasi, Program Studi Sistem Informasi, Universitas Jember

ABSTRAK

Selama ini proses pentahapan strata desa siaga aktif di wilayah kerja Dinas Kesehatan Kabupaten Jember masih dilakukan secara manual dengan mencocokkan data desa yang ada dengan kriteria pentahapan strata desa siaga aktif. Sehingga pentahapan yang dilakukan membutuhkan waktu yang cukup lama mengingat jumlah desa dalam 31 kecamatan di Kabupaten Jember bukanlah jumlah yang sedikit. Selain itu dalam proses pentahapan terdapat 8 kriteria yang harus dicocokkan dengan pencapaian masing-masing desa. Pembuatan sistem informasi pemetaan strata desa siaga aktif pada penelitian ini bertujuan untuk merancang dan membangun sistem informasi dengan mengimplementasikan metode AHP (*Analytical Hierarchy Process*) yang mampu menganalisis data capaian desa terhadap kriteria strata desa siaga sehingga dapat melakukan pentahapan terhadap data capaian desa tersebut. Proses analisis pentahapan strata desa siaga aktif menggunakan metode AHP didasarkan pada tingkat prioritas antar kriteria dan subkriteria pentahapan strata desa siaga aktif untuk dapat menentukan nilai dari masing-masing kriteria dan subkriteria. Selanjutnya terdapat proses penentuan batas nilai interval untuk mengecek nilai yang diperoleh masing-masing desa terhadap strata desa siaga aktif. Terdapat 4 strata desa siaga aktif yaitu pratama, madya, purna, dan mandiri. Hasil dari penelitian ini menghasilkan laporan pentahapan strata desa siaga yang otomatis tersimpan sesuai tanggal dilakukannya pentahapan. Selain itu sistem ini mampu menampilkan hasil pentahapan strata desa siaga dalam bentuk peta digital yang ditampilkan dengan perbedaan warna sesuai dengan strata yang dicapai masing-masing desa.

Kata Kunci: Kriteria dan Pentahapan Strata Desa Siaga Aktif, metode *Analytical Hierarchy Process*.

RINGKASAN

Penerapan Metode AHP Dalam Sistem Informasi Pemetaan Sebaran Trata Desa Siaga Aktif Dinas Kesehatan Kabupaten Jember; Rizka Rohmatul Fitriah, 112410101001; 2015: halaman; Program Studi Sistem Informasi Universitas Jember

Sistem pemetaan strata desa siaga aktif Dinas Kesehatan Kab. Jember yang dibangun dalam penelitian ini merupakan sistem yang mampu melakukan pentahapan strata desa siaga aktif serta menampilkan sebuah peta persebaran strata desa siaga berdasarkan analisis data kriteria strata desa siaga yang dilakukan dengan mengimplementasikan metode AHP. Adapun kriteria pentahapan desa siaga antara lain forum masyarakat desa/kelurahan, KPM/kader kesehatan, kemudahan akses pelayanan kesehatan, posyandu dan UKBM lainnya, dukungan dana untuk kegiatan kesehatan di desa dan kelurahan, peran masyarakat dan organisasi kemasyarakatan, peraturan kepala desa, dan pembinaan PHBS rumah tangga. Kriteria-kriteria tersebut yang dianalisis menggunakan metode AHP, sehingga dalam sistem ini terdapat menu proses AHP. Selain itu, sistem ini menyediakan fitur laporan yang disajikan setiap kali melakukan proses pentahapan strata desa siaga untuk dapat dijadikan sebagai bukti dan laporan bagi Dinas Kesehatan Kab. Jember. Sesuai dengan tujuan dibangunnya sistem pemetaan strata desa siaga ini menampilkan peta strata desa siaga dengan memanfaatkan fasilitas *Google Maps API* selain menampilkan peta dengan warna-warna yang sesuai dengan strata desa. *Info windows* yang dapat diakses dengan meng-klik pada peta disajikan dengan maksud agar pengguna lebih mudah dan mendapatkan informasi lebih dari peta yang ditampilkan selain hanya memperoleh informasi berupa persebaran desa siaga di Kabupaten Jember.

PRAKATA

Puji Syukur kehadiran Allah SWT. atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Penerapan Metode AHP Dalam Sistem Informasi Pemetaan Sebaran Trata Desa Siaga Aktif Dinas Kesehatan Kabupaten Jember”. Skripsi ini bertujuan untuk memenuhi salah satu syarat menyelesaikan pendidikan strata satu (S1) pada Program Studi Sistem Informasi Universitas Jember.

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

1. Ayahanda Drs.Miswanto dan Ibunda Siti Latifah, S.Pd., serta seluruh keluarga yang telah mencurahkan perhatian, dukungan, motivasi, dan limpahan doa demi terselesaikannya skripsi ini;
2. Prof. Drs. Slamini, M.Comp.Sc, Ph.D., selaku Dosen Pembimbing Akademik yang telah membimbing selama penulis menjadi mahasiswa;
3. Drs. Antonius Cahya P, M.App. Sc., selaku Dosen Pembimbing Utama, Yanuar Nurdiansyah, ST, M.Cs., selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi ini;
4. Seluruh dosen PS. Sistem Informasi yang telah membimbing selama penulis menjadi mahasiswa;
5. Dinas Kesehatan Kabupaten Jember yang telah membantu dan mengizinkan penulis untuk melakukan penelitian;
6. NEFOTION yang telah menjadi rekan berjuang selama penulis menjadi mahasiswa serta keluarga besar PS.Sistem Informasi karyawan dan mahasiswa seluruh angkatan (SI09, Zerone, Formation, Intention, Sensation) atas segala dukungan dan doa selama penulis menjadi mahasiswa hingga menyelesaikan skripsi ini;

7. Pengurus HIMASIF periode 2011/2012 dan periode 2012/2013, Panitia *Study Excursie*, Panitia HOKI Internet 2012, Panitia BITS, Ketua dan Asisten Laboratorium Rekayasa Perangkat Lunak periode 2013/2014, Panitia FORTRAN 2013 untuk semua ilmu, pelajaran, dan pengalaman yang tak ternilai;
8. Rizqa Farihatul Jannah, saudara se-nama namun tak serupa yang telah berkenan untuk berjuang bersama dalam segala kesempatan sejak menjadi mahasiswa hingga terselesaikannya skripsi ini;
9. Resha, Gibran, Amris, Eka, Selamat, Qilba, Kakak Nyai Rastra dan Ophie, Markov Adji, Mas Erick yang telah meluangkan waktu dan pikiran dalam penulisan skripsi ini;
10. Keluarga besar YASPPI AL-BAKRY, dewan guru beserta karyawan SMK 08 Ma'arif NU Wuluhan, seluruh siswa-siswi SMK 08 Ma'arif NU Wuluhan yang telah memberikan semangat serta dukungan moral hingga skripsi ini dapat terselesaikan;
11. Semua pihak yang tidak dapat disebutkan satu per satu.

Penulis juga menerima segala kritik dan saran dari semua pihak demi kesempurnaan skripsi ini. Akhirnya penulis berharap, semoga skripsi ini dapat bermanfaat.

Jember, Juni 2015

Penulis

DAFTAR ISI

BAB I PENDAHULUAN	13
1.1 Latar Belakang	20
1.2 Rumusan Masalah	3
1.3 Tujuan dan Manfaat	3
1.3.1 Tujuan	3
1.3.2 Manfaat	4
1.4 Batasan Masalah	5
1.4.1 Lingkup Materi	5
1.4.2 Lingkup Wilayah	5
1.5 Sistematika Penulisan	6
BAB II TINJAUAN PUSTAKA	7
2.1 Kajian Penelitian Terdahulu	7
2.2 Desa dan Kelurahan Siaga Aktif	8
2.3 Kriteria dan Pentahapan	9
2.3.1. Kriteria Desa Siaga Aktif	9
2.3.2. Pentahapan Desa Siaga Aktif	10
2.4 <i>Analytical Hierarchy Process (AHP)</i>	12
2.4.1. Penyusunan Hirarki (<i>Decomposition</i>)	12
2.4.2. Penyusunan dan Penetapan Prioritas (<i>Synthesis of Priority</i>)	13
2.5 Konsistensi Logis (<i>Logical Consistency</i>)	14
BAB III METODE PENELITIAN	17
3.1 Tempat dan Waktu Penelitian	17
3.1.1 Tempat Penelitian	17
3.1.2 Waktu Penelitian	17
3.2 Metode Penelitian	17
3.3 Metode Pengumpulan Data	19

3.3.1	Data Primer	19
3.3.2	Data Sekunder	19
3.4	Metode Analisis Data	19
3.5	Teknik Pengembangan Sistem	20
BAB IV PENGEMBANGAN SISTEM.....		24
4.1	Deskripsi Umum Sistem.....	24
4.1.1	SOP (<i>statement of perpose</i>) Sistem.....	24
4.1.2	Fungsi Sistem.....	25
4.2	Analisis Kebutuhan Sistem.....	26
4.2.1	Kebutuhan Fungsional.....	26
4.2.2	Kebutuhan Non-Fungsional	27
4.3	Desain Sistem.....	27
4.3.1	<i>Business Process</i>	28
4.3.2	<i>Use Case Diagram</i>	29
4.3.3	<i>Increment</i> Proses Penentuan Strata Desa Siaga	34
4.3.3.1	<i>Use Case</i> Menambah Data Puskesmas.....	34
4.3.3.2	<i>Use Case</i> Mengedit Data Puskesmas	34
4.3.3.3	<i>Use Case</i> Menghapus Data Puskesmas.....	34
4.3.3.4	<i>Use Case</i> Melihat Data Puskesmas	35
4.3.3.5	<i>Use Case</i> Menambah Data Desa	35
4.3.3.6	<i>Use Case</i> Mengedit Data Desa	35
4.3.3.7	<i>Use Case</i> Melihat Data Desa.....	36
4.3.3.8	<i>Use Case</i> Melihat Data Kriteria	36
4.3.3.9	<i>Use Case</i> Pembobotan Kriteria	36
4.3.3.10	<i>Use Case</i> Mengubah Bobot Kriteria.....	36
4.3.3.11	<i>Use Case</i> Melihat Data Subkriteria	37
4.3.3.12	<i>Use Case</i> Pembobotan Subkriteria	37
4.3.3.13	<i>Use Case</i> Mengubah Bobot Subkriteria	37

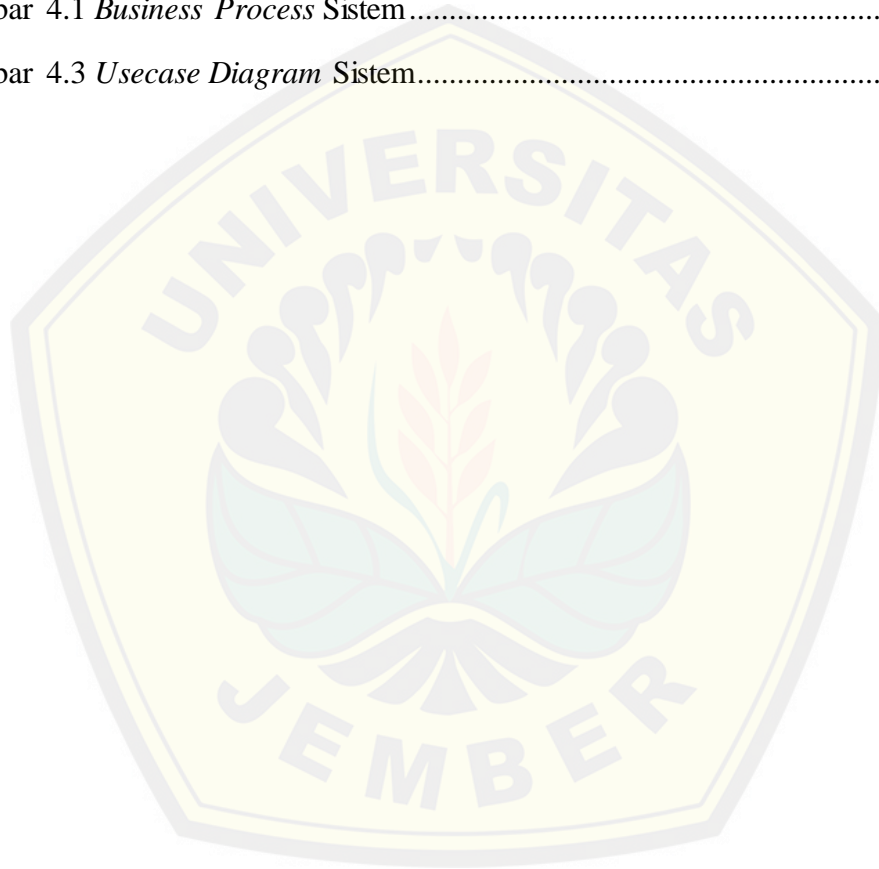
4.3.3.14	<i>Use Case</i> Pembobotan Alternatif Desa Siaga	37
4.3.3.15	<i>Use Case</i> Mengubah Nilai Pembobotan Alternatif	38
4.3.3.16	<i>Use Case</i> Melihat Hasil Pembobotan Alternatif	38
4.3.4	<i>Increment</i> Laporan	38
4.3.5	<i>Increment</i> Pemetaan	39
4.3.6	<i>Increment</i> Manajemen User	40
4.3.7	<i>Use Case</i> Login	41
4.4	Integrasi Sistem	42
4.5	Pengkodean Sistem	42
4.6	Pengujian Sistem	43
4.6.1.	Metode <i>White Box</i>	43
4.6.2.	Metode <i>Black Box</i>	44
BAB V HASIL DAN PEMBAHASAN		45
5.1	Hasil Analisis Data Kriteria	45
5.1.1.	Pentahapan Strata Desa Siaga Aktif	45
5.1.2.	Penyusunan Hirarki	46
5.1.3.	Penentuan Prioritas Kriteria	46
5.1.4.	Penentuan Prioritas Subkriteria	49
5.1.5.	Analisis Pentahapan Strata Desa Siaga Aktif	50
5.2	Implementasi Sistem	53
5.2.1.	<i>Increment</i> Proses Penentuan Strata Desa Siaga	53
5.2.2.	<i>Increment</i> Pemetaan	56
5.2.3.	<i>Increment</i> Laporan	57
5.2.4.	<i>Increment</i> Manajemen User	57
5.2.5.	<i>Use Case</i> Login	58
5.3	Pembahasan	58
5.3.1.	Pembahasan Hasil Implementasi Sistem Pemetaan Strata Desa Siaga Aktif dengan Metode AHP	58
5.3.2.	Pembahasan Hasil Pengujian Sistem	60

BAB VI PENUTUP	63
6.1. Kesimpulan	63
6.2. Saran	64
DAFTAR PUSTAKA	63
LAMPIRAN	65
LAMPIRAN A <i>Use Case Scenario</i>	65
A.1. <i>Increment</i> Proses Penentuan Strata Desa Siaga	65
A.2. <i>Increment</i> Pemetaan	81
A.3. <i>Increment</i> Laporan	82
A.4. <i>Increment</i> Manajemen <i>User</i>	83
LAMPIRAN B <i>Activity Diagram</i>	87
B.1. <i>Increment</i> Proses Penentuan Strata Desa Siaga	87
B.2. <i>Increment</i> Laporan	101
B.3. <i>Increment</i> Pemetaan	102
B.4. <i>Increment</i> Manajemen <i>User</i>	103
LAMPIRAN C <i>Sequence Diagram</i>	107
C.1. <i>Increment</i> Proses Penentuan Strata Desa Siaga	107
C.2. <i>Increment</i> Laporan	116
C.3. <i>Increment</i> Pemetaan	116
C.4. <i>Increment</i> Manajemen <i>User</i>	117
LAMPIRAN D <i>Class Diagram</i>	121
D.1. <i>Increment</i> Proses Penentuan Strata Desa Siaga	121
D.2. <i>Increment</i> Laporan	122
D.3. <i>Increment</i> Pemetaan	123
D.4. <i>Increment</i> Manajemen <i>User</i>	124
D.5. Integrasi Sistem	125
LAMPIRAN E <i>Entity Relationship Diagram</i>	126
E.1. <i>Increment</i> Proses Penentuan Strata Desa Siaga	126

E.2. <i>Increment</i> Laporan	126
E.3. <i>Increment</i> Pemetaan.....	127
E.4. <i>Increment</i> Manajemen <i>User</i>	127
E.5. Integrasi Sistem.....	128
LAMPIRAN F Implementasi Metode AHP	129
LAMPIRAN G Pengujian <i>White Box</i>	153
G.1 <i>Increment</i> Proses Penentuan Strata Desa Siaga	153
G.2 <i>Increment</i> Manajemen <i>User</i>	164
LAMPIRAN H Pengujian <i>Black Box</i>	169
LAMPIRAN I Implementasi Sistem	176
I.1. <i>Increment</i> Proses Penentuan Strata Desa Siaga	176
I.2. <i>Increment</i> Laporan	181
I.3. <i>Increment</i> Pemetaan.....	181
I.4. <i>Increment</i> Manajemen <i>User</i>	182
I.5. <i>Login</i>	183

DAFTAR GAMBAR

Gambar 3.1 Diagram Alir Metode Penelitian.....	18
Gambar 3.2 Diagram Alir Metode Analisis Data.....	20
<i>Gambar 3.3 Diagram alir model increment.....</i>	<i>21</i>
Gambar 4.1 <i>Business Process</i> Sistem.....	29
Gambar 4.3 <i>Usecase Diagram</i> Sistem.....	30



DAFTAR TABEL

Table 2.1 Tabel indikator pentahapan strata desa siaga aktif	10
Tabel 2.2 Tahapan Penyusunan Hirarki.....	12
Tabel 2.3 Skala Perbandingan Tingkat Kepentingan	13
Tabel 2.4 Daftar indeks random konsistensi (IR).....	15
Tabel 4.1 Deskripsi Pembagian Aktor Sistem.....	31
Tabel 4.2 Deskripsi <i>Usecase</i> Sistem.....	31
Tabel 5.1 Matrik Perbandingan Berpasangan	47
Tabel 5.2 Matrik Perhitungan Nilai Eigen Vektor Normalisasi.....	48
Tabel 5.3 Penentuan Batas Interval Strata Desa Siaga Aktif.....	51
Tabel 5.4 Tabel Penentuan Nilai Interval.....	52

BAB I PENDAHULUAN

1.1 Latar Belakang

Sejalan dengan visi pembangunan nasional tahun 2005-2025 sebagaimana ditetapkan dalam Undang-Undang Republik Indonesia Nomor 17 Tahun 2007 tentang Rencana Pembangunan Jangka Panjang Nasional Tahun 2005-2025 adalah “INDONESIA YANG MANDIRI, MAJU, ADIL, DAN MAKMUR”, pemerintah menetapkan delapan arah pembangunan jangka panjang yang salah satunya adalah mewujudkan bangsa yang berdaya saing. Pembangunan dibidang kesehatan merupakan salah satu pendukung terwujudnya arah pembangunan bangsa Indonesia menjadi bangsa yang berdaya saing. Oleh sebab itu, Undang-Undang Republik Indonesia Nomor 36 Tahun 2009 tentang Kesehatan mengamanatkan bahwa pembangunan kesehatan harus ditunjukan untuk meningkatkan kesadaran, kemauan, dan kemampuan hidup sehat masyarakat yang setinggi-tingginya sebagai investasi bagi pembangunan sumber daya manusia yang produktif secara sosial dan ekonomis. Hal tersebut menegaskan bahwa meski negara menjamin hak atas kesehatan setiap penduduknya, namun masyarakat juga harus memiliki kesadaran dan kemauan untuk melakukan kewajiban-kewajiban dibidang kesehatan disamping menuntut hak-haknya.

Strategi pemberdayaan masyarakat, swasta, dan masyarakat madani dalam pembangunan kesehatan merupakan salah satu langkah yang diambil oleh Kementerian Kesehatan RI. Menerapkan promosi kesehatan yang efektif memanfaatkan *agent of change* atau agen perubahan yang berasal dari masyarakat setempat dapat dilaksanakan oleh dinas kesehatan salah satunya dengan melaksanakan upaya memfasilitasi percepatan dan pencapaian peningkatan derajat kesehatan bagi seluruh penduduk dengan mengembangkan kesiap-siagaan di tingkat desa yang disebut Desa Siaga. Sesuai dengan keputusan Menteri Kesehatan Nomor 564/Menkes/SK/VIII/2006 tentang Pedoman Pelaksanaan Pengembangan Dese

Siaga, pelaksanaan Desa Siaga sendiri telah berkembang dan masih terus perlu dilakukan pembinaan. Kepala Pusat Promosi Kesehatan Kementerian Kesehatan RI dalam Pedoman Desa Siaga (2010) memaparkan bahwa target yang harus dicapai pada Tahun 2015 adalah 80% desa dan kelurahan yang ada di Indonesia telah menjadi Desa dan Kelurahan Siaga Aktif. Sehingga diperlukan akselerasi yang diselenggarakan dalam Pengembangan Desa dan Kelurahan Siaga Aktif demi mencapai target tersebut.

Dinas Kesehatan Kabupaten Jember sebagai instansi pemerintahan memiliki peran utama dalam usaha mewujudkan visi pembangunan kesehatan maupun pembangunan nasional yang telah diamanatkan dalam Undang-Undang di tingkat kabupaten. Pelaksanaan usaha pencapaian target Desa Siaga Tahun 2015 juga menjadi tanggung jawab bagi Dinas Kesehatan Kabupaten Jember dapat dilakukan dengan melakukan pendampingan serta pengawasan. Meski pelaksanaan Desa Siaga diserahkan kepada pemerintah desa atau kelurahan, namun Dinas Kesehatan Kabupaten harus tetap melakukan pengawasan melalui PUSKESMAS yang tersebar di seluruh wilayah Kabupaten Jember. Hingga saat ini pelaksanaan pengawasan Desa Siaga yang dilakukan oleh Dinas Kesehatan Kabupaten Jember hanya sebatas pengumpulan data yang telah diperoleh setiap PUSKESMAS sesuai dengan wilayah masing-masing desa atau kelurahan. Sehingga untuk dapat mengetahui kondisi persebaran Desa Siaga di Kabupaten Jember serta mengambil kebijakan terkait kondisi Desa Siaga, Dinas Kesehatan Kabupaten Jember harus menganalisis data yang telah dikumpulkan oleh masing-masing PUSKESMAS secara manual. Kondisi tersebut tentu saja dinilai kurang efektif mengingat target Desa Siaga 2015 yang harus dicapai.

Metode *Analytical Hierarchy Process* (AHP) merupakan metode yang dinilai mampu membantu dalam manajemen data kondisi Desa atau Kelurahan di wilayah kerja Dinas Kabupaten Jember yang diperoleh dari seluruh PUSKESMAS yang ada. Khususnya dalam proses penentuan strata desa siaga aktif sesuai dengan kondisi masing-masing desa yang ada di Kabupaten Jember. Kriteria-kriteria yang digunakan

sebagai acuan penentuan strata desa siaga diolah menggunakan analisis dari sebuah hirarki sehingga menghasilkan suatu pendukung keputusan yang lebih baik.

Sistem Informasi Persebaran Strata Desa Siaga Aktif Dinas Kesehatan Kabupaten Jember dengan Menerapkan Metode AHP diharapkan mampu membantu kinerja Dinas Kesehatan Kabupaten Jember dalam mewujudkan target Desa Siaga 2015. Sebab sistem ini nantinya akan membantu dalam mengelola data desa atau kelurahan sehingga dapat diketahui kondisi atau strata Desa Siaga yang ada pada masing-masing wilayah kerja PUSKESMAS. Selanjutnya data tersebut akan ditampilkan dalam bentuk peta persebaran strata Desa Siaga Aktif. Sehingga Dinas Kesehatan Kabupaten Jember dan seluruh pihak yang terkait dapat dengan mudah mengetahui kondisi capaian Desa Siaga yang ada di Kabupaten Jember untuk kemudian dapat segera mengambil kebijakan dan langkah yang efektif untuk meningkatkan kondisi tersebut.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah dijelaskan di atas, perumusan permasalahan yang muncul dalam penelitian ini yaitu:

1. Bagaimana pentahapan strata desa siaga aktif dapat dianalisis dengan metode AHP?
2. Bagaimana merancang dan membangun Sistem Informasi Persebaran Strata Desa Siaga Aktif Dinas Kesehatan Kabupaten Jember dengan Menerapkan Metode AHP?

1.3 Tujuan dan Manfaat

Tujuan dan manfaat dalam penelitian ini merupakan arahan jawaban dari permasalahan serta menemukan hasil-hasil yang hendak dicapai.

1.3.1 Tujuan

Adapun tujuan dari dilakukannya penelitian ini adalah sebagai berikut:

1. Dapat melakukan pentahapan strata desa siaga aktif menggunakan metode AHP
2. Dapat merancang dan membangun Sistem Informasi Persebaran Strata Desa Siaga Aktif Dinas Kesehatan Kabupaten Jember dengan Menerapkan Metode AHP

1.3.2 Manfaat

Manfaat dari dibangunnya membangun Sistem Informasi Persebaran Strata Desa Siaga Aktif Dinas Kesehatan Kabupaten Jember adalah sebagai berikut:

1. Bagi Akademisi
Memberikan informasi dan bahan literatur bagi dunia pendidikan dalam bidang pentahapan strata desa siaga aktif maupun sistem informasi pemetaan.
2. Bagi Instansi
 - a. Lebih mudah dalam melakukan peran *controlling* terhadap pentahapan strata desa siaga aktif di wilayah kerja Dinas Kesehatan Kabupaten Jember.
 - b. Lebih menghemat waktu dalam upaya menentukan kebijakan terhadap kondisi persebaran strata desa siaga aktif di wilayah kerja Dinas Kesehatan Kabupaten Jember sesuai dengan visualisasi data dan informasi yang ditampilkan dalam membangun Sistem Informasi Persebaran Strata Desa Siaga Aktif.
3. Bagi Penulis
 - a. Penulis mampu menerapkan bidang keahlian ke dalam kebutuhan instansi, dalam hal ini adalah mampu merancang dan membangun Sistem Informasi Persebaran Strata Desa Siaga Aktif sesuai dengan kebutuhan Dinas Kesehatan Kabupaten Jember.

- b. Memenuhi salah satu syarat menyelesaikan pendidikan strata satu (S1) Program Studi Sistem Informasi Universitas Jember.

1.4 Batasan Masalah

Agar tidak terjadi penyimpangan dalam proses penelitian dan pembangunan Sistem Informasi Pemetaan Strata Desa Siaga Aktif Dinas Kesehatan Kabupaten Jember, maka ditetapkan beberapa batasan permasalahan. Adapun batasan masalah dalam penelitian ini dapat dibagi menjadi dua, yaitu batasan ruang lingkup materi dan ruang lingkup wilayah.

1.4.1. Lingkup Materi

Batasan ruang lingkup materi dimaksudkan untuk memberikan batasan terhadap pembahasan objek penelitian yang dilakukan dapat terarah dan sesuai tujuan yang hendak dicapai. Pembatasan materi tersebut meliputi:

- a. Objek penelitian yang digunakan adalah data desa atau kelurahan berdasarkan indikator-indikator Desa Siaga
- b. Parameter yang digunakan adalah kriteria strata desa siaga aktif yang telah ditetapkan oleh KEMENKES RI.
- c. Sistem Informasi yang dibangun dapat menentukan dan memetakan pentahapan strata desa siaga aktif menggunakan metode AHP
- d. Menggunakan *Google Maps API* untuk membantu dalam memvisualisasikan peta sebaran strata desa siaga aktif di Kabupaten Jember.

1.4.2. Lingkup Wilayah

Batasan ruang lingkup wilayah dalam penelitian ini adalah lima wilayah kecamatan di Kabupaten Jember, yaitu Tanggul, Wuluhan, Panti, Sumber Sari, dan Sumberjambe.

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, konsep, informasi, dan kajian teori berkaitan dengan masalah dalam penelitian.

3. Metode Penelitian

Bab ini menguraikan tentang jenis penelitian, tempat dan waktu penelitian, metode pengumpulan data, metode analisis data, metode perancangan sistem, implementasi, dan metode evaluasi sistem.

4. Analisis dan Perancangan Sistem

Bab ini berisi uraian tentang langkah-langkah yang ditempuh dalam proses menganalisis dan merancang sistem yang hendak dibangun.

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 II TINJAUAN PUSTAKA

2.1 Kajian Penelitian Terdahulu

Penelitian lain dilakukan oleh Tokoro,dkk.(2013) untuk menentukan tingkat kerentanan gizi buruk di Distrik Sentani Kabupaten Jayapura menggunakan metode AHP. Pada penelitian ini sistem dibangun untuk membantu pengambil kebijakan dalam menentukan wilayah rentan gizi buruk serta faktor apa saja yang berpotensi menimbulkan masalah dengan menekankan pada faktor-faktor di setiap wilayah. Penyajian data hasil analisis penentuan wilayah rentan gizi buruk dalam sistem ini ditampilkan dalam tabel data dan diagram batang.

Pemetaan potensi kemiskinan dengan metode AHP (*Analytical Hierarchy Process*) digunakan untuk melakukan pemetaan terhadap wilayah yang berpotensi memiliki masyarakat miskin cukup banyak sehingga dapat dijadikan pendukung tercapainya percepatan program pengentasan kemiskinan yang akan di berikan oleh pemerintah maupun pihak lainnya. Sistem ini dibangun berbasis website dengan menampilkan data dalam bentuk tabel dan tampilan peta digital dengan gradasi warna sesuai dengan jumlah kepala keluarga per kategori miskin (Sri Redjeki,dkk.2014).

Metode AHP juga digunakan oleh Heru Nurwarsito.dkk(2010) dalam penelitian pengembangan sistem pendukung keputusan untuk menentukan gizi ibu hamil sebab menentukan gizi pada ibu hamil merupakan salah satu contoh kasus multikriteria. Dimana dalam proses pengambilan keputusan dapat dipengaruhi oleh beberapa aspek. Sehingga metode AHP dengan analisis perbandingan berpasangan yang terdapat didalamnya dinilai mampu menghasilkan keputusan yang tepat dan fleksibel dengan aspek yang mempengaruhi keputusan tersebut.

Berdasarkan penelitian-penelitian yang telah dilakukan sebelumnya seperti yang telah dipaparkan di atas, maka penulis berusaha mengkombinasikan keunggulan-keunggulan dari setiap penelitian yang telah dilakukan tersebut untuk dapat membangun suatu sistem yang sesuai dengan kebutuhan. Analisis dalam

penelitian pemetaan persebaran strata desa siaga aktif ini dilakukan dengan metode AHP, sebab metode AHP menggunakan suatu susunan hirarki untuk dapat memecahkan suatu masalah kemudian memasukkan nilai numeric sebagai pengganti persepsi manusia dalam melakukan perbandingan relatif. Hal tersebut merupakan alasan mengapa analisis yang dilakukan dalam penelitian ini menggunakan metode AHP dibandingkan dengan metode pendukung keputusan yang lain.

Sistem yang dibangun akan menyajikan hasil pentahapan strata desa siaga aktif dalam bentuk peta digital dengan gradasi warna berdasarkan strata atau tingkatan desa siaga aktif. Selain dalam bentuk tabel, data hasil analisis disajikan dalam bentuk peta digital dimaksudkan agar pihak Dinas Kesehatan Kabupaten Jember serta seluruh pihak yang terait dapat lebih mudah dalam melihat kondisi persebaran desa siaga aktif sebagai sarana evaluasi untuk menentukan kebijakan yang akan diambil selanjutnya.

2.2 Desa dan Kelurahan Siaga Aktif

Desa dan Kelurahan Siaga Aktif adalah bentuk pengembangan dari Desa Siaga yang telah dimulai sejak tahun 2006. Desa atau Kelurahan Aktif adalah desa atau yang disebut dengan nama lain atau kelurahan, yang :

1. Penduduknya dapat mengakses dengan mudah pelayanan kesehatan dasar yang memberikan pelayanan setiap hari melalui Pos Kesehatan Desa (PosKesDes) atau sarana kesehatan yang ada di wilayah tersebut seperti, Pusat Kesehatan Masyarakat Pembantu (PusTu), Pusat Kesehatan Masyarakat (Puskesmas) atau sarana kesehatan lainnya;
2. Penduduknya mengembangkan UKBM dan melaksanakan survailans berbasis masyarakat (meliputi pemantauan penyakit, kesehatan ibu dan anak, gizi, lingkungan, dan perilaku), kedaruratan kesehatan dan penanggulangan bencana, serta penyehatan lingkungan sehingga masyarakatnya menerapkan Perilaku Hidup Bersih dan Sehat (PHBS). (Pedoman Umum Pengembangan Desa dan Kelurahan Siaga Aktif ,2010).

2.3 Kriteria dan Pentahapan

Berdasarkan pemahaman tentang desa siaga aktif seperti yang telah dipaparkan kementerian kesehatan RI dalam Pedoman Umum Pengembangan Desa dan Kelurahan Siaga Aktif Tahun 2010, selanjutnya dilakukan suatu pentahapan terhadap capaian masing-masing desa di seluruh wilayah Negara Indonesia. Untuk dapat mengetahui capaian tahapan dari suatu desa dan kelurahan, kementerian kesehatan RI juga telah menetapkan kriteria dan pentahapan strata desa siaga aktif.

2.3.1. Kriteria Desa Siaga Aktif

Berdasarkan Pedoman Umum Pengembangan Desa dan Kelurahan Siaga Aktif tahun 2010 terdapat 8 (delapan) kriteria yang harus dicapai dan dipenuhi oleh desa dan kelurahan, yaitu :

1. Kepedulian Pemerintah Desa atau Kelurahan dan pemuka masyarakat terhadap Desa dan Kelurahan Siaga Aktif yang tercermin dari keberadaan dan keaktifan Forum Desa dan Kelurahan.
2. Keberadaan kader pemberdayaan masyarakat/ kader teknis Desa dan Kelurahan Siaga Aktif
3. Kemudahan akses masyarakat terhadap pelayanan kesehatan dasar yang buka atau memberikan pelayanan setiap hari
4. Keberadaan UKBM yang dapat melaksanakan (a) survailans berbasis masyarakat, (b) penanggulangan bencana dan kedaruratan kesehatan, (c)penyehatan lingkungan
5. Tercakupnya (terakomodasinya) pendanaan untuk pembangunan Desa dan Kelurahan Siaga Aktif dalam Anggaran Pembangunan Desa atau Kelurahan serta masyarakat dari dunia usaha
6. Peran serta aktif masyarakat dan organisasi kemasyarakatan dalam kegiatan kesehatan di Desa dan Kelurahan Siaga Aktif

7. Peraturan di tingkat desa atau kelurahan yang melandasi dan mengatur tentang pengembangan Desa dan Kelurahan Siaga Aktif
8. Pembinaan PHBS di rumah tangga di desa atau kelurahan.

2.3.2. Pentahapan Desa Siaga Aktif

Pedoman Umum Pengembangan Desa dan Kelurahan Siaga Aktif tahun 2010 yang diterbitkan oleh kementerian kesehatan, terdapat empat tahapan desa siaga aktif yaitu:

- a. Desa dan kelurahan siaga aktif Pratama
- b. Desa dan kelurahan siaga aktif Madya
- c. Desa dan kelurahan siaga aktif Purnama
- d. Desa dan kelurahan siaga aktif Mandiri

Untuk dapat mencapai suatu tahapan tertentu, desa siaga aktif harus memenuhi indikator-indikator yang telah ditetapkan seperti yang dapat kita lihat dalam Tabel 2.1 berikut.

Table 2.2 Tabel indikator pentahapan strata desa siaga aktif

Kriteria	Pratama	Madya	Purnama	Mandiri
Forum Masyarakat Desa/Kelurahan	Ada, tetapi belum jalan	Berjalan, tetapi belum rutin setiap tri-wulan	Berjalan setiap tri-wulan,	Berjalan setiap Bulan
KPM / Kader Kesehatan/ Kader Tehnis	Sudah ada min. 2 orang	Sudah ada 3-5 orang	Sudah ada 6-8 orang	Sudah ada 9 orang atau lebih
Kemudahan Akses Pelayanan Kesehatan	Ya	Ya	Ya	Ya
Posyandu & UKBM lainnya Aktif	Posyandu ya, UKBM lainnya tidak aktif	Posyandu & 2 UKBM lainnya aktif	Posyandu & 3 UKBM lainnya aktif	Posyandu & 4 UKBM lainnya aktif

Dilanjutkan..

Kriteria	Pratama	Madya	Purnama	Mandiri
Dukungan dana untuk kegiatan kesehatan di desa dan kelurahan	Sudah ada dari pemerintah desa dan kelurahan serta belum ada sumber lainnya	Sudah ada dari pemerintah desa dan kelurahan serta satu sumber lainnya baik masyarakat ataupun dunia usaha	Sudah ada dari pemerintah desa dan kelurahan serta dua sumber lainnya yaitu masyarakat dan dunia usaha	Sudah ada dari pemerintah desa dan kelurahan serta dua sumber lainnya yaitu masyarakat dan dunia usaha
Peran Masyarakat dan Organisasi Kemasyarakatan	Ada peran aktif masyarakat namun tidak ada peran aktif ormas	Ada peran aktif masyarakat yang didukung peran aktif satu ormas	Ada peran aktif masyarakat yang didukung peran aktif dua ormas	Ada peran aktif masyarakat yang didukung peran aktif lebih dari dua ormas
Peraturan Kepala Desa atau peraturan Bupati/walikota	Belum ada	Ada, belum direalisasikan	Ada, sudah direalisasikan	Ada, sudah direalisasikan
Pembinaan PHBS Rumah Tangga	Pembinaan PHBS Kurang dari 20% rumah tangga yang ada	Pembinaan PHBS minimal 20% rumah tangga yang ada	Pembinaan PHBS minimal 40% rumah tangga yang ada	Pembinaan PHBS minimal 70% rumah tangga yang ada

2.4 Analytical Hierarchy Process (AHP)

AHP (*Analytical Hierarchy Process*) merupakan suatu model pendukung keputusan yang diperkenalkan dan dikembangkan oleh Thomas L. Saaty dengan menguraikan masalah multi kriteria menjadi suatu hirarki. Hirarki sendiri dapat diartikan sebagai suatu representasi dari suatu permasalahan yang kompleks (multi level) dimana level pertama merupakan *goal* atau tujuan yang hendak dicapai, kemudian terdapat level faktor, kriteria, sub kriteria, hingga level terakhir dari alternative yang ada.

Implementasi AHP untuk mendapatkan suatu alternatif kriteria yang tepat dapat dilakukan dalam beberapa tahap sesuai prinsip dasar dari metode AHP, yaitu (Saaty dalam Sukarnei,2011):

- a. *Decomposition* (prinsip menyusun hirarki)
- b. *Synthesis of Priority* (penyusunan dan penetapan prioritas)
- c. *Logical Consistency* (prinsip konsistensi logika)

2.4.1. Penyusunan Hirarki (*Decomposition*)

Proses penyusunan hirarki merupakan proses dimana permasalahan yang ada akan disusun menjadi model hirarki yaitu dengan memecah suatu persoalan menjadi elemen-elemen yang terpisah. Terdapat 3 tahapan dalam proses penyusunan hirarki seperti dalam Tabel 2.2.

Tabel 2.2 Tahapan Penyusunan Hirarki

Tahap Pertama	Identifikasi tujuan keseluruhan pembuatan hirarki yang biasa disebut dengan <i>goal</i> (tujuan), yaitu masalah yang akan dicari pemecahannya melalui model AHP
Tahap Kedua	Menentukan kriteria yang diperlukan untuk mendukung tujuan keseluruhan
Tahap Ketiga	Identifikasi strategi – alternatif yang akan dievaluasi dibawah kriteria

Brojonegoro (dalam Asmarani,2010)

2.4.2. Penyusunan dan Penetapan Prioritas (*Synthesis of Priority*)

Penyusunan dan penetapan prioritas merupakan tahapan yang dilakukan untuk memperoleh suatu nilai prioritas dari kriteria-kriteria yang digunakan dalam mendukung tercapainya tujuan, beserta nilai prioritas subkriteria dari masing-masing kriteria yang ada. Berikut merupakan tahapan-tahapan yang dilakukan dalam menyusun dan menetapkan nilai prioritas dari kriteria dan subkriteria.

2.4.1.1. Perbandingan Penilaian (*Comparative Judgement*)

Comparative Judgement berarti membuat suatu penilaian tentang kepentingan relatif dua elemen pada suatu tingkat tertentu dimana penilaian tersebut disajikan dalam bentuk matriks dengan menggunakan skala prioritas. Perbandingan penilaian ini lebih mudah disajikan dalam bentuk matriks perbandingan berpasangan (*pairwise comparison*) antar berbagai kriteria yang telah ditetapkan, yang dilakukan dengan tahapan sebagai berikut:

- a. Menentukan mana diantara dua yang dianggap penting/disukai/mungkin terjadi;
- b. Menentukan berapa kali lebih penting/disukai/mungkin terjadi.

Seluruh prioritas yang ada dibandingkan satu sama lain secara berpasangan dan diberi bobot berupa skala dari 1 sampai dengan 9. Skala perbandingan tingkat kepentingan pada metode AHP dapat dilihat pada Tabel 2.3.

Tabel 2.3 Skala Perbandingan Tingkat Kepentingan

Perbandingan nilai relative antar kriteria i dan kriteria j	Definisi Penilaian	Keterangan
1	Sama penting	Dua kriteria (i dan j) memiliki tingkat kepentingan yang sama dalam memenuhi tujuan
3	Sedikit lebih penting	Kriteria i sedikit lebih penting dibandingkan kriteria j dalam memenuhi tujuan
5	Lebih penting	Kriteria i memiliki tingkat kepentingan yang cukup besar dibandingkan kriteria j dalam memenuhi tujuan

Dilanjutkan..

Perbandingan nilai relative antar kriteria i dan kriteria j	Definisi Penilaian	Keterangan
7	Sangat penting	Kriteria i memiliki tingkat kepentingan sangat besar dibandingkan kriteria j dalam memenuhi tujuan
9	Jauh lebih penting	Kriteria i memiliki tingkat kepentingan jauh lebih besar dibandingkan kriteria j dalam memenuhi tujuan
2,4,6,8	Nilai antara	Penilaian diantara relatif yang lainnya
kebalikan	Jika aktivitas i mendapat satu angka dibanding aktivitas j, maka j memiliki nilai kebalikannya dibanding dengan i.	

Saaty (dalam Sukarmeri, 2011)

Selanjutnya proses yang dilakukan yaitu menetapkan kriteria dimana proses ini dilakukan dengan perbandingan berpasangan (*pairwise comparison*) antar berbagai kriteria yang telah ditetapkan. Tahapan yang dilakukan setelah matrik perbandingan berpasangan selesai diisi yaitu menetapkan prioritas yang dilakukan dengan metode *eigen vector* dan *eigen value*. Dari *eigen vector* yang diperoleh dapat ditentukan *local priority* (prioritas untuk satu level). Kemudian *global priority* diperoleh dengan mengalikan prioritas elemen pada level di atasnya sampai pada level terakhir.

2.5 Konsistensi Logis (*Logical Consistency*)

Metode AHP memiliki perbedaan asumsi dengan metode pengambilan keputusan yang lain yaitu tidak adanya syarat konsistensi mutlak dimana metode ini menggunakan persepsi manusia sebagai masukan. Keterbatasan manusia tentu saja dapat menyebabkan munculnya nilai yang tidak konsisten dalam menyatakan persepsinya, apalagi jika membandingkan banyak elemen. Konsistensi menunjukkan intensitas relasi antar elemen didasarkan pada suatu kriteria tertentu.

Index konsistensi (CI) dapat diperoleh dari

$$CI = \frac{\lambda_{maks} - \text{jumlah kriteria } (n)}{\text{jumlah kriteria } (n) - 1} \dots\dots\dots \text{(persamaan 1)}$$

λ maks : nilai eigen maksimum dari bobot prioritas yang diperoleh

Rasio konsistensi (CR) dapat diperoleh dari

$$CR = \frac{CI}{RI} , (RI = \text{Random Index}) \dots\dots\dots \text{(persamaan 2)}$$

Daftar indeks random konsistensi (IR) bisa dilihat pada tabel berikut:

Tabel 2.4 Daftar indeks random konsistensi (IR)

Ukuran Matriks	Nilai IR
1,2	0.00
3	0.58
4	0.90
5	1.12
6	1.24
7	1.32
8	1.41
9	1.45
10	1.49
11	1.51
12	1.48
13	1.56
14	1.57
15	1.59

Berdasarkan pemaparan di atas dapat kita ketahui bahwa untuk mendapatkan suatu keputusan maka tahapan yang dilakukan dalam proses analisis menggunakan metode AHP antara lain menyusun sebuah hirarki, penyusunan dan penetapan prioritas, serta konsistensi logis. Hirarki disusun untuk dapat menguraikan permasalahan yang akan diselesaikan dimana didalam hirarki terdapat goal atau tujuan yang hendak dicapai, penyusunan kriteria, dan identifikasi alternatif-alternatif yang akan dievaluasi. Tahap selanjutnya yaitu penyusunan dan penetapan prioritas dimana kriteria serta subkriteria yang ada diberikan skala dan dibandingkan

menggunakan matrik perbandingan berpasangan hingga diperoleh nilai prioritas antar kriteria dan antar subkriteria yang ada. Setelah nilai prioritas kriteria dan subkriteria diperoleh, maka tahapan selanjutnya yaitu melakukan uji konsistensi terhadap nilai tersebut untuk mendapatkan nilai yang valid. Apabila nilai konsistensi lebih dari 0,1 maka penilaian dari data perbandingan harus diperbaiki.



BAB III METODE PENELITIAN

Bab ini menguraikan mengenai jenis, tempat dan waktu penelitian, objek serta metode penelitian yang digunakan dalam membangun Sistem Informasi Persebaran Strata Desa Siaga Aktif Dinas Kesehatan Kabupaten Jember menggunakan metode AHP.

3.1 Tempat dan Waktu Penelitian

3.1.1 Tempat Penelitian

Penelitian ini dilakukan di lima wilayah kerja Dinas Kesehatan Kabupaten Jember, yaitu: Kecamatan Tanggul, Wuluhan, Panti, Sumpalsari, dan Sumberjambe.

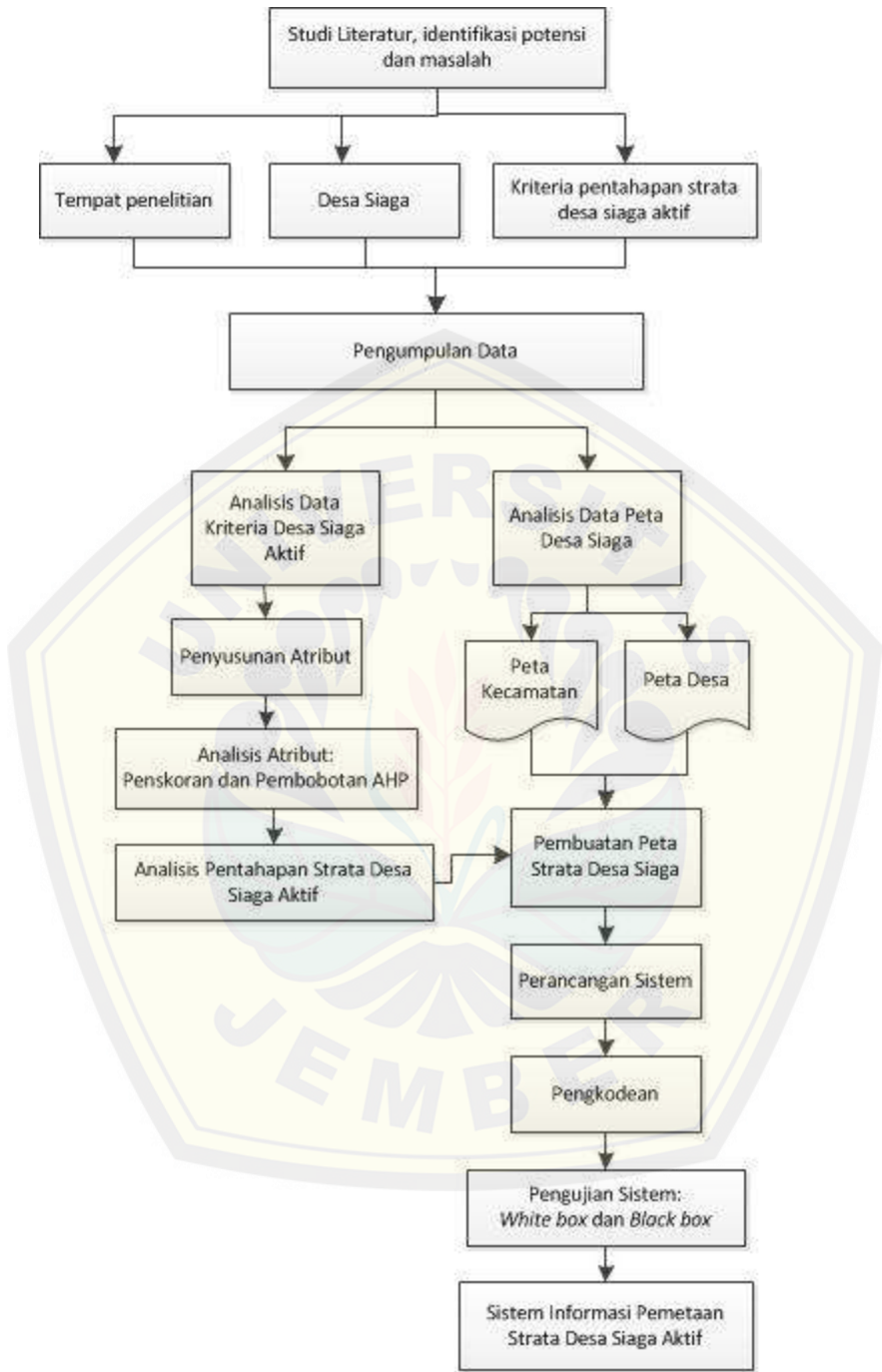
3.1.2 Waktu Penelitian

Penelitian ini dilakukan dalam jangka waktu 3 bulan.

3.2 Metode Penelitian

Metode penelitian merupakan suatu langkah-langkah yang digunakan dalam melakukan suatu penelitian. Untuk membangun atau mengembangkan suatu produk tentu saja dibutuhkan suatu tahap analisis kebutuhan untuk menguji keefektifan produk tersebut. Sesuai dengan penelitian yang akan dilakukan dalam skripsi ini, maka metode yang digunakan dalam penelitian ini yaitu metode penelitian dan pengembangan *Research and Development* (R&D). Metode R&D sendiri merupakan metode penelitian yang digunakan untuk menghasilkan produk tertentu, dan menguji keefektifan produk tersebut (Sugiyono, 2011:297).

Mengadopsi dari alur atau tahapan pengembangan sistem yang dilakukan sesuai dengan metode penelitian dan pengembangan *Research and Development* (R&D). Maka alur penelitian dalam skripsi ini dapat dilihat pada Gambar 2.1 berikut.



Gambar 3.5 Diagram Alir Metode Penelitian

3.3 Metode Pengumpulan Data

3.3.1 Data Primer

Pengumpulan data primer dilakukan dengan metode observasi dan wawancara langsung kepada bagian Promosi Kesehatan Dinas Kesehatan Kabupaten Jember. Data yang dibutuhkan dalam penelitian ini dibagi menjadi 2 kategori, yaitu kategori data spasial dan data nonspasial.

a. Data Spasial

Data spasial berisi tentang data-data keruangan seperti peta data administrasi. Data administrasi dalam penelitian ini adalah data kecamatan dan data desa di Kabupaten Jember.

b. Data Nonspasial

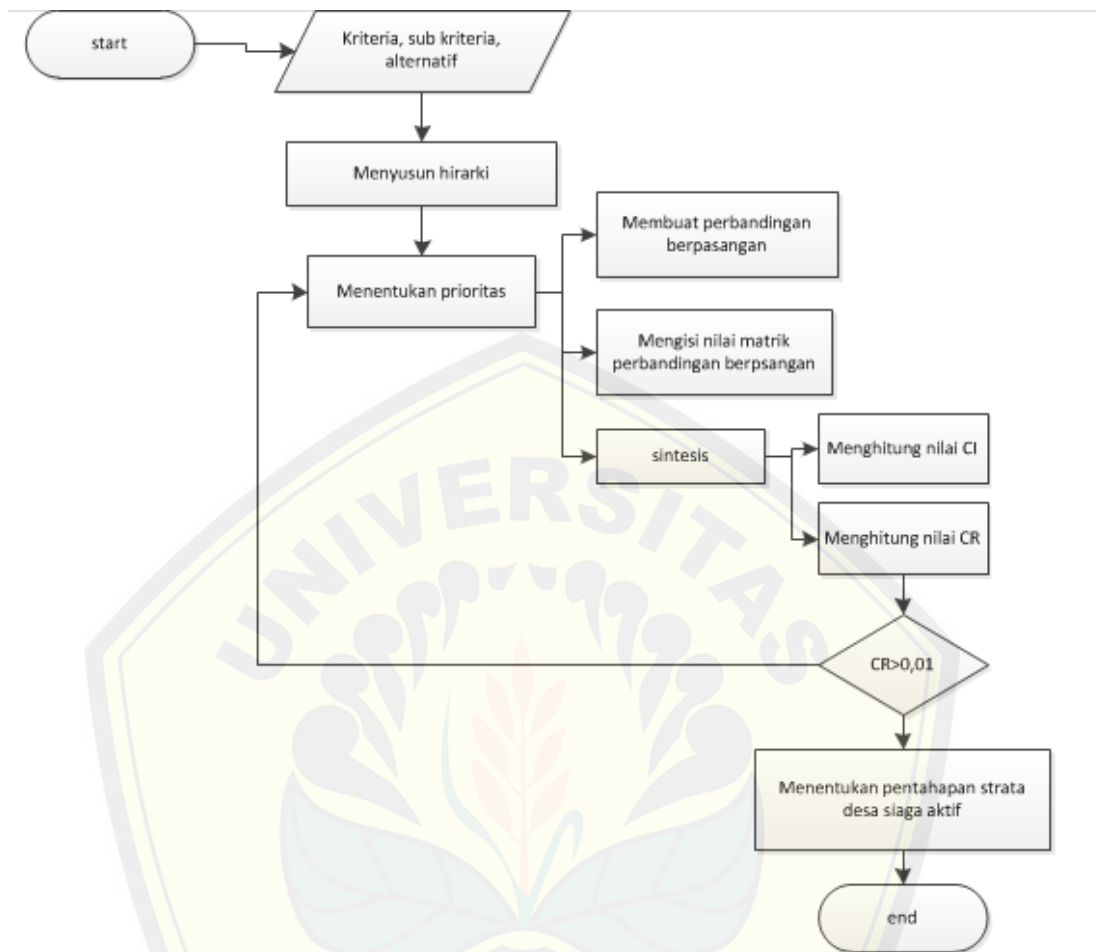
Data nonspasial merupakan data atribut yang digunakan sebagai pendukung data spasial.

3.3.2 Data Sekunder

Data sekunder digunakan sebagai pendukung data primer yang diperoleh dari data bagian Promosi Kesehatan Dinas Kesehatan Kabupaten Jember. Selain itu data juga diperoleh dengan membaca buku literatur, jurnal, penelitian terdahulu, dan internet.

3.4 Metode Analisis Data

Metode analisis data yang digunakan sesuai dengan tujuan dan rumusan masalah dari penelitian penentuan dan persebaran strata desa siaga aktif di Kabupaten Jember dengan menggunakan metode AHP. Berikut merupakan penjelasan dari analisis data yang digunakan dalam penelitian ini, seperti yang ditampilkan pada Gambar 3.2.



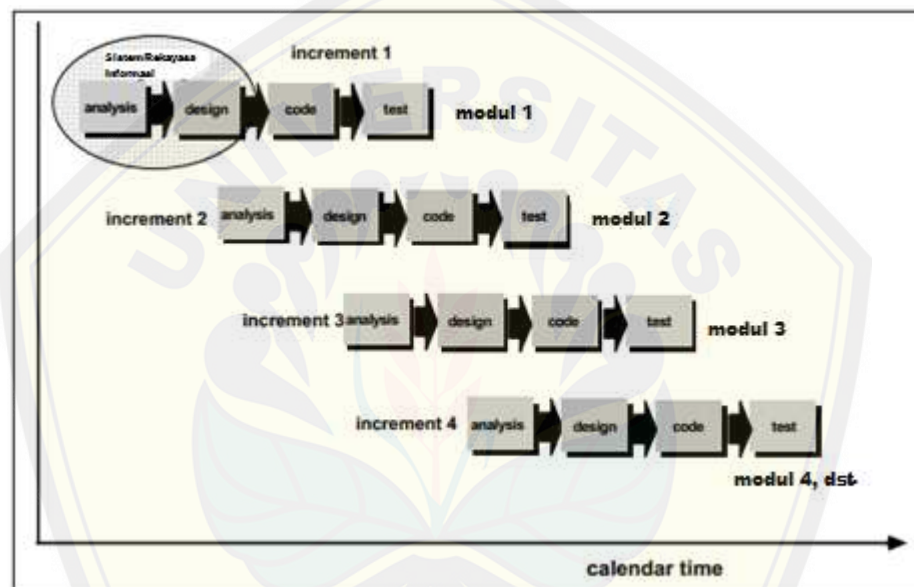
Gambar 3.6 Diagram Alir Metode Analisis Data

3.5 Teknik Pengembangan Sistem

Teknik pengembangan sistem dalam penelitian ini menggunakan *System Development Life Cycle (SDLC) model incremental*. Model pengembangan sistem ini merupakan penggabungan dari model *waterfall* dan *prototype*. Sehingga dalam pelaksanaannya sistem dibangun atau dikembangkan sesuai dengan pembagian-pembagian yang telah dilakukan sebelumnya. Sistem yang akan dibangun akan dibagi menjadi beberapa increment yang nantinya akan menentukan urutan dalam proses pembangunan atau pengembangan sistem itu sendiri. Pembagian increment-

increment tersebut dilakukan berdasarkan pada tingkat *critical* yang dimiliki serta berdasarkan pada keterkaitan antar increment dalam sistem tersebut.

Berdasarkan increment-increment yang telah ditentukan, maka proses selanjutnya yaitu tahapan pengembangan sistem. Dimana sistem akan dibangun pertahap sesuai dengan increment yang telah ditentukan. Tahapan pengembangan sistem model *incremental* sesuai dengan pembagian increment dapat dilihat pada Gambar 3.3.



Gambar 3.7 Diagram alir model *increment*
(Rosa dan Shalahuddin, 2013:38)

Berdasarkan gambar di atas, dapat kita ketahui bahwa dalam model inkremental terdapat beberapa tahapan proses yang harus dijalankan, yaitu :

1. Analisis Kebutuhan

Menganalisis kebutuhan yang akan digunakan dalam pembuatan aplikasi. Meliputi pengumpulan data kebutuhan fungsional dan non-fungsional dari aplikasi yang akan kita bangun. Setelah itu, menentukan fungsi dan fasilitas apa saja yang akan dibuat dalam aplikasi.

2. Desain Sistem

Jika proses analisis kebutuhan telah diketahui maka proses selanjutnya yaitu pada tahapan desain sistem. Proses pendesainan sistem dari aplikasi yang akan kita bangun yaitu dengan menggunakan *Unified Modeling Language (UML)*. Penggunaan UML karena sudah menggunakan konsep *Object Oriented Design* yang tentunya akan sangat memudahkan developer untuk membangun sebuah sistem. Dalam UML ada beberapa diagram yang akan dibuat antara lain:

- a. *Business Process*, digunakan untuk mendeskripsikan sistem secara keseluruhan (*goal, uses, input, output, dll.*)
- b. *Use Case Diagram*, digunakan untuk mendeskripsikan interaksi antara aktor dengan sistem yang akan dibuat
- c. *Activity Diagram*, digunakan untuk menggambarkan *workflow* atau aktivitas dari sebuah sistem atau menu yang ada pada perangkat lunak
- d. *Class Diagram*, digunakan untuk menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat dalam sebuah sistem yang akan dibangun
- e. *Entity Relationship Diagram*, digunakan untuk menggambarkan *database* sistem yang akan dibangun

3. Pengkodean (*Coding*)

Setelah proses desain system dikerjakan, proses selanjutnya adalah *coding* atau penulisan kode program. Bahasa pemrograman yang dipakai adalah php (*Hypertext Preprocessor*) dengan *codeigniter frameworok* karena sesuai dengan desain sistem yang telah dilakukan yaitu menggunakan konsep *Object Oriented Programming (OOP)*, sedangkan *tool* yang digunakan adalah Netbeans 7.2. Database yang digunakan yaitu PostgreSQL dengan *tool* yang digunakan yaitu PhpPgAdmin.

4. Pengujian (*Testing*)

Pada tahap ini dilakukan ujicoba terhadap sistem yang telah dibangun atau dikembangkan untuk menemukan kesalahan-kesalahan yang mungkin terjadi supaya dapat melakukan perbaikan dan lebih menyempurnakan sistem. Pengujian yang akan dilakukan dalam penelitian ini menggunakan dua metode pengujian sistem, yakni *white box testing method* dan *black box testing method*. Metode *White Box Testing* berfungsi untuk menguji perangkat lunak dari segi desain dan kode program apakah mampu menghasilkan fungsi-fungsi, masukan, dan keluaran yang sesuai dengan spesifikasi kebutuhan. Dimana pengujian ini dilakukan dengan memeriksa logika serta kesalahan-kesalahan yang mungkin terjadi pada kode program. Sedangkan metode *Black Box Testing* digunakan untuk menguji perangkat lunak dari segi spesifikasi fungsional yang dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

Pengujian dilakukan dengan metode *White Box* dilakukan oleh penulis menggunakan teknik pengujian *Cyclometric Complexity* (CC). Sedangkan untuk metode *Black Box* dilakukan oleh Bagian Promosi Kesehatan Dinas Kesehatan Kabupaten Jember.

BAB IV PENGEMBANGAN SISTEM

Bab ini menguraikan mengenai analisis kebutuhan dan perancangan hingga tahap pengkodean dan pengujian sistem yang digunakan dalam proses pengembangan atau pembangunan Sistem Informasi Persebaran Strata Desa Siaga Aktif Dinas Kesehatan Kabupaten Jember dengan Menerapkan Metode AHP. Dimana tahapan analisis hingga pengujian yang dilakukan sesuai dengan pembagian *increment* yang telah ditetapkan sebelumnya.

4.1 Deskripsi Umum Sistem

Deskripsi umum dari sistem pemetaan strata desa siaga aktif Dinas Kesehatan Kab. Jember dengan mengimplementasikan metode AHP yang dibangun dalam penelitian ini akan dijelaskan lebih detail pada SOP (*statement of perpose*) sistem dan fungsi sistem.

4.1.1 SOP (*statement of perpose*) Sistem

Sistem pemetaan strata desa siaga aktif Dinas Kesehatan Kab. Jember yang dibangun dalam penelitian ini merupakan sistem yang mampu menampilkan sebuah peta persebaran strata desa siaga berdasarkan analisis data kriteria strata desa siaga dengan metode AHP. Adapun kriteria pentahapan desa siaga antara lain forum masyarakat desa/kelurahan, KPM/kader kesehatan, kemudahan akses pelayanan kesehatan, posyandu dan UKBM lainnya, dukungan dana untuk kegiatan kesehatan di desa dan kelurahan, peran masyarakat dan organisasi kemasyarakatan, peraturan kepala desa, dan pembinaan PHBS rumah tangga. Kriteria-kriteria tersebut yang dianalisis menggunakan metode AHP, sehingga dalam sistem ini terdapat menu proses AHP. Selain itu, sistem ini menyediakan fitur laporan yang disajikan setiap kali melakukan proses pentahapan strata desa siaga untuk dapat dijadikan sebagai bukti dan laporan bagi Dinas Kesehatan Kab. Jember. sesuai dengan tujuan dibangunnya sistem pemetaan strata desa siaga ini menampilkan peta strata desa siaga dengan memanfaatkan fasilitas *Google Maps API* selain menampilkan peta dengan

warna-warna yang sesuai dengan strata desa. *Info windows* yang dapat diakses dengan meng-klik pada peta disajikan dengan maksud agar pengguna lebih mudah dan mendapatkan informasi lebih dari peta yang ditampilkan selain hanya memperoleh informasi berupa persebaran desa siaga di Kabupaten Jember.

4.1.2 Fungsi Sistem

Fungsi utama dari sistem yang dibangun dalam penelitian ini terletak pada fitur login yang dapat menentukan hak akses dari setiap pengguna dari sistem ini sendiri. Ketika pengguna melakukan login, maka sistem akan melakukan autentifikasi username dan password dari pengguna. Selanjutnya sistem akan menyajikan tampilan sistem yang sesuai dengan hak akses dari pengguna yang meliputi :

a. User umum

User umum merupakan pengguna sistem yang tidak terdaftar atau tidak memiliki username dan password. Hak akses dari user umum hanya sebatas melihat peta sebaran strata desa siaga.

b. Operator

Operator merupakan pengguna sistem yang terdaftar dan berhasil melakukan login. Selanjutnya operator akan masuk pada halaman utama sistem sesuai dengan level yaitu level dua. Selain dapat mengakses halaman peta sebaran strata desa siaga, operator juga dapat mengakses halaman menu data puskesmas dan data desa. Hak akses tersebut adalah dapat melakukan input, edit, dan view data puskesmas maupun data desa. Selain itu operator juga dapat melakukan pembobotan alternatif desa siaga aktif .

c. Super Admin

Super admin memiliki peran yang sangat kompleks dalam sistem. Dimana super admin memiliki hak akses penuh untuk melakukan manajemen data seperti menambah, mengupdate, dan menghapus data (user, puskesmas, desa), melihat history sistem, dan manajemen laporan setiap kali melakukan pentahapan strata desa siaga. Selain itu, super admin dapat melakukan manajemen data kriteria dan

subkriteria strata desa siaga untuk selanjutnya dilakukan perhitungan bobot menggunakan metode AHP hingga dapat tersaji peta sebaran strata desa siaga.

4.2 Analisis Kebutuhan Sistem

Berdasarkan metode pengembangan sistem model *incremental*, tahapan awal yang dilakukan pada setiap increment adalah tahapan analisis. Tahapan analisis ini dilakukan terhadap objek penelitian untuk memperoleh kebutuhan-kebutuhan dari sistem yang dibangun, baik berupa kebutuhan fungsional maupun kebutuhan non-fungsional. Dimana hasil analisa tersebut sangat mempengaruhi fungsionalitas sistem yang dibangun untuk dapat digunakan sesuai dengan fungsi dan kebutuhan pengguna.

4.2.1. 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 informasi pemetaan strata desa siaga yaitu:

- a. Sistem mampu mengelola data kriteria strata desa siaga menggunakan metode AHP
- b. Sistem mampu menentukan strata desa siaga berdasarkan hasil analisis data kriteria menggunakan metode AHP
- c. Sistem dapat menampilkan peta sebaran strata desa siaga dengan gradasi warna sesuai dengan strata yang telah ditentukan
- d. Sistem dapat mendukung proses manajemen data desa dan puskesmas (input, edit, delete)
- e. Sistem dapat menampilkan data desa dan puskesmas
- f. Sistem dapat menyajikan laporan berdasarkan waktu pentahapan strata desa siaga
- g. Peta sebaran strata ditampilkan dengan memanfaatkan fasilitas *Google Maps API* dan menampilkan *info window* pada peta

4.2.2. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan fitur-fitur yang dimiliki untuk mendukung sistem dalam memenuhi fungsionalitasnya untuk dapat memenuhi kebutuhan dari pengguna. Kebutuhan non-fungsional dari sistem informasi pemetaan strata desa siaga yaitu:

- a. Sistem memiliki batasan hak akses pengguna dengan menggunakan username dan password
- b. Sistem berbentuk website
- c. Sistem menggunakan framework Codeigniter

4.3 Desain Sistem

Tahapan yang dilakukan setelah melakukan analisis kebutuhan sistem yaitu tahap perencanaan pembangunan sistem yang dapat digambarkan dengan desain sistem. Desain Sistem Informasi Pemetaan Strata Desa Siaga Aktif Dinas Kesehatan Kab. Jember ini meliputi *use case diagram*, *use case* skenario, *activity diagram*, *sequence diagram*, *class diagram*, dan *entity relationship diagram*. Sesuai dengan metode pengembangan sistem yang digunakan penulis dalam penelitian ini, yaitu model *incremental* maka sistem akan dibangun sesuai dengan pembagian *increment* yang telah ditentukan sebelumnya. Adapun tahapan pembangunan sistem sesuai dengan pembagian *increment*, yaitu

- a. *Increment* proses penentuan strata desa siaga
- b. *Increment* pemetaan
- c. *Increment* laporan
- d. *Increment* manajemen user

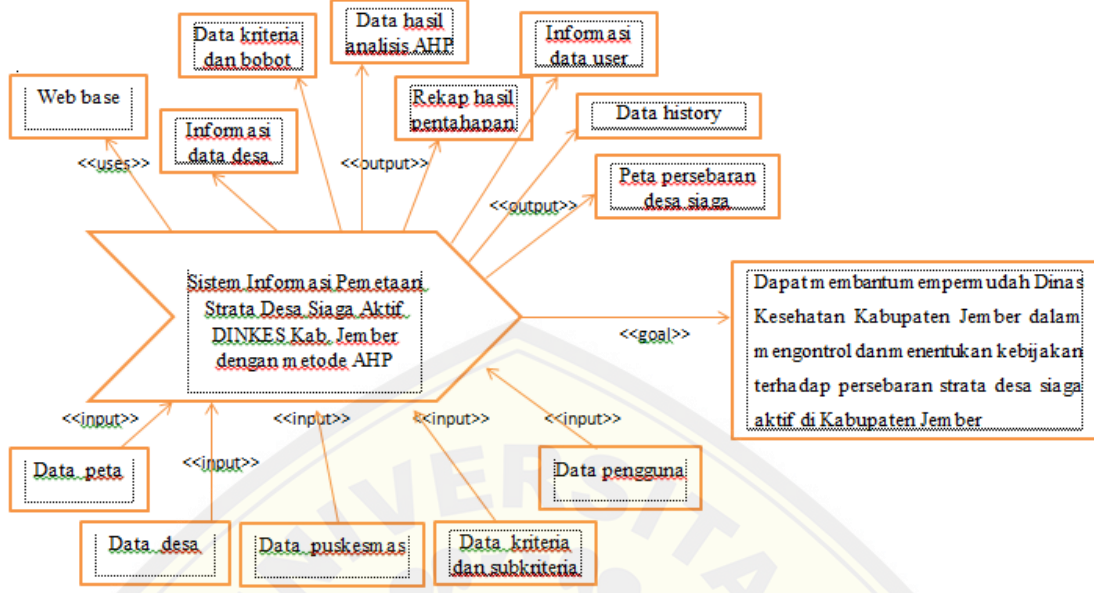
Seperti yang telah kita ketahui bahwa dasar pembagian *increment* adalah pada letak *critical modul* atau titik kritis sistem yang akan dibangun terlebih dahulu serta keterkaitan antar fitur pada tiap *increment*. Adapun fitur-fitur yang saling berkaitan berdasarkan masing-masing *increment* yaitu:

1. *Increment* proses penentuan strata desa siaga merupakan *increment* yang pertama dibangun karena terdapat titik kritis sistem dalam *increment* ini. Dimana proses penentuan strata desa siaga dengan mengimplementasikan metode AHP terdapat pada *increment* ini. Fitur-fitur yang terdapat dalam *increment* ini antara lain fitur manajemen data (tambah data, edit data, hapus data) administrasi desa dan kecamatan, fitur manajemen data kriteria dan sub kriteria, fitur proses AHP dan pentahapan strata desa siaga, dan fitur pembobotan alternatif.
2. *Increment* pemetaan merupakan *increment* yang berfungsi untuk menampilkan peta sebaran strata desa siaga berdasarkan hasil pembobotan alternatif yang dilakukan dalam *increment* proses penentuan strata desa siaga. Sehingga *increment* pemetaan ini dibangun setelah *increment* pertama selesai.
3. *Increment* laporan memiliki fungsi untuk melihat dan mencetak laporan setiap kali setelah melakukan proses pentahapan strata desa siaga.
4. *Increment* manajemen user memiliki fitur menambah user, mengubah data user, menghapus dan melihat data user. Selain itu dalam *increment* ini terdapat fitur yang memungkinkan super admin dapat melihat *record* aktivitas yang dilakukan oleh pengguna (level 1 dan level 2) terhadap sistem.

Selanjutnya tahap desain sistem dibuat menggunakan pemodelan *Unified Modeling Language (UML)*. Pemodelan ini digunakan sesuai dengan konsep *Object-Oriented Programming (OOP)* yang digunakan dalam tahap implementasi atau pengkodean.

4.3.1 *Business Process*

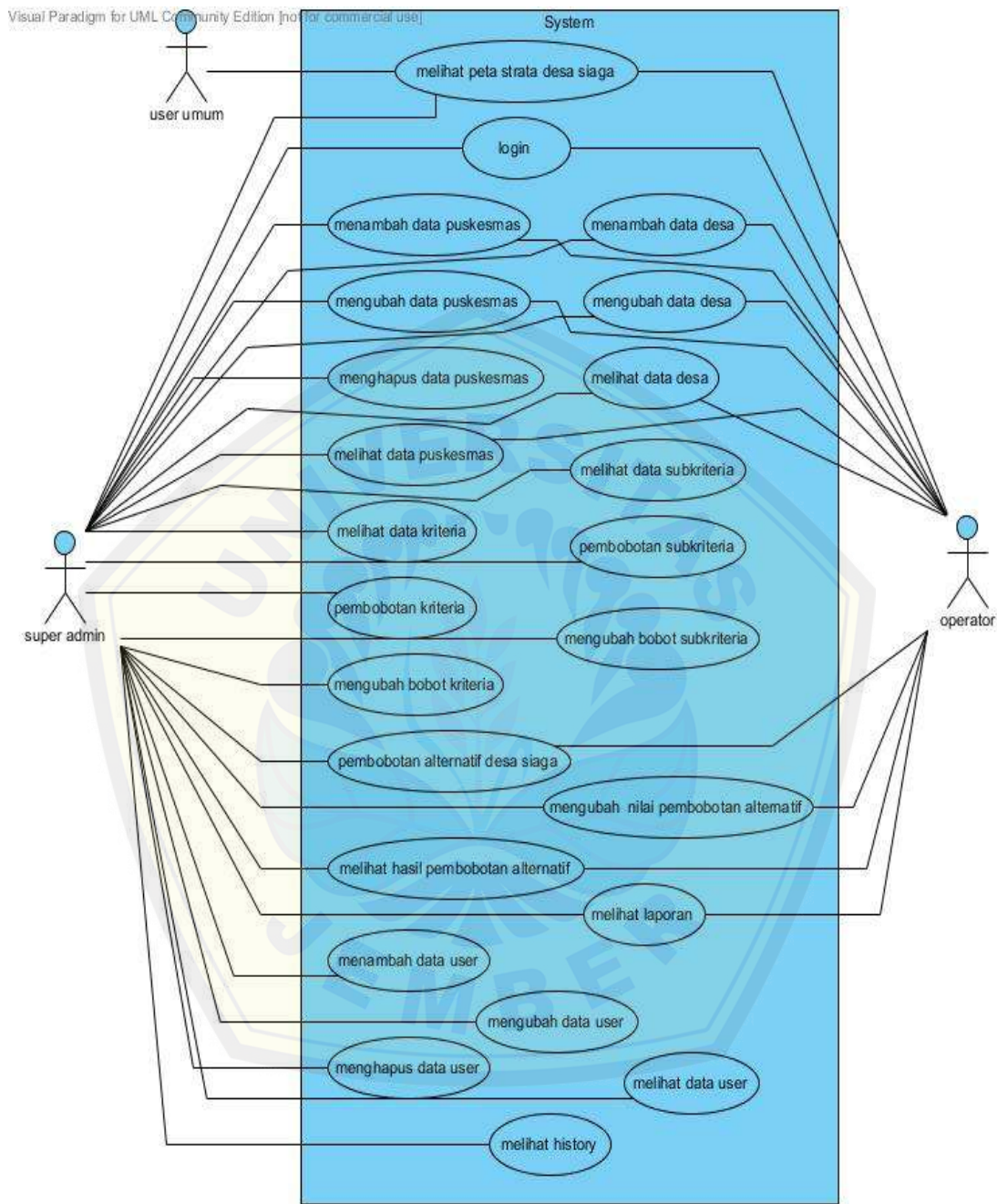
Selain dapat dideskripsikan dalam sebuah *SOP (Statement of Purpose)*, gambaran umum sistem informasi pemetaan strata desa siaga aktif dapat digambarkan melalui sebuah *business process*. Seperti yang dapat kita lihat pada Gambar 4.1. *Business process* menggambarkan data-data yang digunakan sebagai masukan, data keluaran, *uses* sistem yang dibangun, hingga *goal* dari dibangunnya sistem informasi pemetaan ini sendiri.



Gambar 8.1 Business Process System

4.3.2 Use Case Diagram

Use case diagram merupakan pemodelan yang dibuat untuk dapat menggambarkan interaksi antara aktor dengan sistem informasi pemetaan yang akan dibangun. Melalui use case diagram dapat diketahui interaksi yang dapat dilakukan aktor terhadap sistem sesuai dengan hak akses yang dimiliki oleh masing-masing aktor atau pengguna. Pada Gambar 4.2 digambarkan use case diagram sistem informasi pemetaan yang terdiri dari tiga aktor dengan dua puluh dua use case.



Gambar 4.2 Usecase Diagram Sistem

Berdasarkan *use case diagram* pada gambar 4.3 terdapat tiga aktor atau pengguna, yaitu user umum, super admin, dan operator. Adapun deskripsi dari masing-masing aktor dapat dilihat pada Tabel 4.1.

Tabel 4.3 Deskripsi Pembagian Aktor Sistem

Aktor	Deskripsi
User umum	Aktor yang tidak terdaftar sebagai user, tidak dapat melakukan login, dan hanya bisa melihat peta sebaran strata desa siaga
Operator	Aktor yang memiliki hak untuk dapat menginput, mengedit, dan melihat data puskesmas serta data desa, dapat menginputkan data alternatif, dan melihat laporan pentahapan
Super admin	Aktor ini memiliki <i>username</i> dan <i>password</i> untuk dapat melakukan login Aktor yang memiliki otoritas penuh dalam mengelola sistem

Selain memiliki tiga aktor, dalam *use case diagram* juga terdapat dua puluh empat *use case* yang dibagi menjadi empat bagian sesuai dengan pembagian increment yang telah dilakukan dalam tahap sebelumnya. Deskripsi dari *use case* tersebut dapat dilihat pada Tabel 4.2 di bawah ini.

Tabel 4.4 Deskripsi *Usecase* Sistem

No	Use Case	Deskripsi
1	login	Fitur yang berfungsi melakukan autentifikasi hak akses semua aktor dalam menggunakan sistem
Increment Proses Penentuan Strata Desa Siaga		
2	Menambah data puskesmas	Fitur yang dapat diakses oleh pengguna level 1 dan level 2 berfungsi untuk menambah data puskesmas
3	Mengubah data puskesmas	Fitur yang dapat diakses oleh pengguna level 1 dan level 2 berfungsi untuk mengubah data puskesmas apabila terdapat kesalahan atau perubahan terhadap data yang telah ditambahkan sebelumnya
4	Menghapus data puskesmas	Fitur untuk menghapus data hanya dapat diakses oleh pengguna level 1 atau super admin.

Dilanjutkan...

No	Use Case	Deskripsi
5	Melihat data puskesmas	Berfungsi sebagai fitur yang memungkinkan pengguna level 1 dan level 2 untuk dapat melihat data puskesmas yang telah ditambahkan sebelumnya
6	Menambah data desa	Fitur yang dapat diakses oleh pengguna level 1 dan level 2 berfungsi untuk menambah data desa sesuai dengan wilayah puskesmas yang menangani desa tersebut
7	Mengedit data desa	Fitur yang dapat diakses oleh pengguna level 1 dan level 2 berfungsi untuk mengubah data desa apabila terdapat kesalahan atau perubahan terhadap data yang telah ditambahkan sebelumnya
8	Melihat data desa	Berfungsi sebagai fitur yang memungkinkan pengguna level 1 dan level 2 untuk dapat melihat data desa yang telah ditambahkan sebelumnya
9	Melihat data kriteria	Fitur hanya dapat diakses oleh super admin yang memungkinkan super admin untuk dapat melihat kriteria-kriteria yang telah ditambahkan sebelumnya
10	Pembobotan kriteria	Berfungsi sebagai salah satu fitur yang berisi tahapan proses perhitungan dari metode ahp yaitu menghitung bobot dari setiap kriteria yang telah disimpan sebelumnya
11	Mengubah bobot kriteria	Fitur ini memungkinkan super admin untuk dapat mengubah hasil pembobotan kriteria yang telah diproses sebelumnya
12	Melihat data subkriteria	Fitur hanya dapat diakses oleh super admin yang memungkinkan super admin untuk dapat melihat subkriteria dari tiap-tiap kriteria yang telah ditambahkan sebelumnya
13	Pembobotan subkriteria	Berfungsi sebagai salah satu fitur yang berisi tahapan proses perhitungan dari metode ahp yaitu menghitung bobot dari setiap subkriteria dari masing-masing kriteria yang telah disimpan sebelumnya
14	Mengubah bobot subkriteria	Fitur ini memungkinkan super admin untuk dapat mengubah hasil pembobotan subkriteria yang telah diproses sebelumnya

Dilanjutkan...

No	Use Case	Deskripsi
15	Pembobotan alternatif desa siaga	Fitur ini berfungsi melakukan pembobotan terhadap alternatif desa siaga yang dapat dilakukan oleh pengguna level 1 dan level 2. Pembobotan didasarkan pada hasil pembobotan kriteria dan subkriteria yang telah diproses sebelumnya. Selanjutnya masing-masing alternatif akan dikelompokkan berdasarkan pembagian strata desa siaga yang telah ditentukan
16	Melihat hasil pembobotan alternatif	Super admin dapat melihat hasil pembobotan dari setiap alternatif yang diinputkan oleh pengguna level 2 maupun oleh super admin sendiri
Increment Pemetaan		
17	Melihat peta persebaran strata desa siaga	Fitur yang dapat memberikan informasi kepada pengguna berupa tampilan peta persebaran strata desa siaga sesuai dengan hasil perhitungan ahp dengan menggunakan fasilitas <i>google maps Api</i> serta tambahan detail informasi yang ditampilkan melalui <i>info window</i>
Increment Manajemen User		
18	Menambah data user	Super admin dapat melakukan penambahan user pada fitur ini
19	Mengedit data user	Fitur ini berfungsi untuk dapat mengubah data user yang telah ditambahkan sebelumnya oleh super admin
20	Menghapus data user	Fitur ini memungkinkan super admin untuk dapat menghapus user yang ada
21	Melihat data user	Berfungsi untuk melihat data user yang ada pada sistem
22	Melihat history	Berfungsi untuk melihat <i>record</i> aktivitas yang dilakukan oleh pengguna terhadap sistem
Increment Laporan		
23	Melihat laporan	Berfungsi untuk menampilkan laporan setiap kali dilakukan pentahapn strata desa siaga
Login		
24	Login	Berfungsi untuk melakukan cek terhadap <i>username</i> dan <i>password</i>

4.3.3 *Increment* Proses Penentuan Strata Desa Siaga

Increment proses penentuan strata desa siaga aktif merupakan *increment* utama dalam sistem informasi pemetaan strata desa siaga aktif dengan metode AHP. Sebagai *increment* utama, *increment* ini terdiri atas fitur-fitur yang mendukung tercapainya proses analisis dengan mengimplementasikan metode AHP seperti data puskesmas, data desa, kriteria, subkriteria, dan pentahapan desa. Berikut merupakan pemaparan dari desain sistem yang akan dibangun berdasarkan masing-masing *use case* yang ada. *class diagram increment* proses penentuan strata desa siaga dapat digambarkan seperti pada lampiran D.1, dan *entity relationship diagram* dapat digambarkan seperti dalam lampiran E.1

4.3.3.1 *Use Case* Menambah Data Puskesmas

Use Case Menambah Data Puskesmas merupakan fitur yang memungkinkan pengguna level 1 dan level 2 untuk dapat menambahkan data puskesmas. Gambaran dari desain sistem untuk *use case* menambah data puskesmas secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.1.1, *activity diagram* seperti pada lampiran B.1.1, dan *sequence diagram* seperti pada lampiran C.1.1.

4.3.3.2 *Use Case* Mengedit Data Puskesmas

Use Case mengedit data puskesmas merupakan fitur yang berfungsi untuk mengubah data puskesmas yang telah ditambahkan sebelumnya. Fitur mengedit data puskesmas dapat diakses oleh pengguna level 1 dan level 2. Penjelasan lebih lanjut mengenai *use case* ini dapat digambarkan dalam *use case scenario* seperti pada lampiran A.1.2, *activity diagram* seperti pada lampiran B.1.2, dan *sequence diagram* seperti pada lampiran C.1.2.

4.3.3.3 *Use Case* Menghapus Data Puskesmas

Use case menghapus data puskesmas merupakan salah satu fitur yang memungkinkan super admin untuk dapat menghapus data puskesmas yang telah ditambahkan

sebelumnya. Gambaran dari desain sistem untuk *use case* menghapus data puskesmas secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.1.3, *activity diagram* seperti pada lampiran B.1.3, dan *sequence diagram* seperti pada lampiran C.1.3.

4.3.3.4 *Use Case* Melihat Data Puskesmas

Use case melihat data puskesmas merupakan salah satu fitur yang memungkinkan super admin dan operator untuk dapat melihat data puskesmas yang telah ditambahkan sebelumnya. Gambaran dari desain sistem untuk *use case* melihat data puskesmas secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.1.4, *activity diagram* seperti pada lampiran B.1.4, dan *sequence diagram* seperti pada lampiran C.1.4.

4.3.3.5 *Use Case* Menambah Data Desa

Use case menambah data desa merupakan salah satu fitur yang memungkinkan super admin dan operator untuk dapat menambah data desa sesuai dengan wilayah kerja puskesmas yang ada. Gambaran dari desain sistem untuk *use case* menambah data desa secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.1.5, *activity diagram* seperti pada lampiran B.1.5, dan *sequence diagram* seperti pada lampiran C.1.5.

4.3.3.6 *Use Case* Mengedit Data Desa

Use case mengedit data desa merupakan salah satu fitur yang memungkinkan super admin dan operator untuk dapat mengedit data desa yang telah ditambahkan sebelumnya. Gambaran dari desain sistem untuk *use case* mengedit data desa secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.1.6, *activity diagram* seperti pada lampiran B.1.6, dan *sequence diagram* seperti pada lampiran C.1.6.

4.3.3.7 Use Case Melihat Data Desa

Use case melihat data desa merupakan salah satu fitur yang memungkinkan super admin dan operator untuk dapat melihat data desa yang telah ditambahkan sebelumnya. Gambaran dari desain sistem untuk *use case* melihat data desa secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.1.7, *activity diagram* seperti pada lampiran B.1.7, dan *sequence diagram* seperti pada lampiran C.1.7.

4.3.3.8 Use Case Melihat Data Kriteria

Use case melihat data kriteria merupakan fitur yang memungkinkan super admin untuk dapat melihat kriteria-kriteria desa siaga aktif. Gambaran dari desain sistem untuk *use case* melihat data kriteria secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.1.8, *activity diagram* seperti pada lampiran B.1.8, dan *sequence diagram* seperti pada lampiran C.1.8.

4.3.3.9 Use Case Pembobotan Kriteria

Use case pembobotan kriteria berfungsi sebagai salah satu fitur yang berisi tahapan proses perhitungan dari metode ahp yaitu menentukan nilai bobot dari setiap kriteria pentahapan strata desa siaga. Gambaran dari desain sistem untuk *use case* pembobotan secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.1.9, *activity diagram* seperti pada lampiran B.1.9, dan *sequence diagram* seperti pada lampiran C.1.9.

4.3.3.10 Use Case Mengubah Bobot Kriteria

Use case mengubah bobot kriteria merupakan fitur yang memungkinkan super admin untuk dapat mengubah nilai bobot dari setiap kriteria. Gambaran dari desain sistem untuk *use case* pembobotan secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.1.10, *activity diagram* seperti pada lampiran B.1.10, dan *sequence diagram* seperti pada lampiran C.1.10.

4.3.3.11 Use Case Melihat Data Subkriteria

Use case melihat data subkriteria merupakan fitur yang berfungsi untuk super admin melihat subkriteria dari masing-masing kriteria desa siaga yang telah ditambahkan sebelumnya. Gambaran dari desain sistem untuk *use case* melihat data subkriteria secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.1.11, *activity diagram* seperti pada lampiran B.1.11, dan *sequence diagram* seperti pada lampiran C.1.11.

4.3.3.12 Use Case Pembobotan Subkriteria

Use case pembobotan subkriteria berfungsi sebagai salah satu fitur yang berisi tahapan proses perhitungan dari metode ahp untuk menghitung bobot subkriteria dari setiap kriteria yang telah disimpan sebelumnya. Gambaran dari desain sistem untuk *use case* pembobotan subkriteria secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.1.12, *activity diagram* seperti pada lampiran B.1.12, dan *sequence diagram* seperti pada lampiran C.1.12.

4.3.3.13 Use Case Mengubah Bobot Subkriteria

Use case mengubah bobot subkriteria merupakan fitur yang memungkinkan super admin untuk dapat mengubah nilai bobot subkriteria menggunakan metode AHP. Gambaran dari desain sistem untuk *use case* pembobotan subkriteria secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.1.13, *activity diagram* seperti pada lampiran B.1.13, dan *sequence diagram* seperti pada lampiran C.1.13.

4.3.3.14 Use Case Pembobotan Alternatif Desa Siaga

Use case pembobotan alternatif desa siaga merupakan fitur berfungsi melakukan perhitungan nilai bobot kriteria dan subkriteria terhadap alternatif desa siaga yang dapat dilakukan oleh pengguna level 1 dan level 2. Hasil dari fitur ini adalah pentahapan strata dari masing-masing alternatif atau desa berdasarkan nilai kriteria yang telah dicapai oleh masing-masing alternatif tersebut. Proses ini akan menyimpan

data hasil pentahapan sesuai dengan waktu dilakukannya pentahapan. Dimana dalam satu hari hanya dapat tersimpan satu kali proses pentahapan. Gambaran dari desain sistem untuk *use case* pembobotan alternatif desa siaga secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.1.14, *activity diagram* seperti pada lampiran B.1.14, dan *sequence diagram* seperti pada lampiran C.1.14.

4.3.3.15 Use Case Mengubah Nilai Pembobotan Alternatif

Use case mengubah nilai pembobotan alternatif merupakan fitur yang memungkinkan super admin dapat melakukan update terhadap nilai pembobotan dari setiap alternatif yang diinputkan oleh pengguna level 2 maupun oleh super admin sebelumnya. Namun fitur update hanya dapat dilakukan pada hari yang sama dengan pentahapan yang dilakukan sebelumnya. Gambaran dari desain sistem untuk *use case* mengubah nilai pembobotan alternatif secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.1.15, *activity diagram* seperti pada lampiran B.1.15, dan *sequence diagram* seperti pada lampiran C.1.15.

4.3.3.16 Use Case Melihat Hasil Pembobotan Alternatif

Use case melihat hasil pembobotan alternatif merupakan fitur yang memungkinkan super admin dapat melihat hasil pembobotan dari setiap alternatif yang diinputkan oleh pengguna level 2 maupun oleh super admin sendiri yang dilakukan paling akhir. Gambaran dari desain sistem untuk *use case* melihat hasil pembobotan alternatif secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.1.16, *activity diagram* seperti pada lampiran B.1.16, dan *sequence diagram* seperti pada lampiran C.1.16.

4.3.4 Increment Laporan

Increment laporan merupakan *increment* dalam sistem informasi pemetaan strata desa siaga aktif dengan metode AHP yang menampilkan data hasil pentahapan data alternatif desa siaga berdasarkan tanggal setiap kali dilakukan pentahapan. Berikut

merupakan pemaparan dari desain sistem yang akan dibangun berdasarkan masing-masing *use case* yang ada. *class diagram increment* proses penentuan strata desa siaga dapat digambarkan seperti pada lampiran D.2, dan *entity relationship diagram* dapat digambarkan seperti dalam lampiran E.2.

4.3.4.1 Use Case Melihat Laporan

Setiap kali dilakukan pentahapan terhadap data desa yang dilakukan oleh operator maupun super admin, sistem akan otomatis menyimpan data tersebut berdasarkan waktu dilakukannya proses tersebut. Jika terjadi proses yang sama pada tanggal yang sama, maka sistem akan otomatis meng-update data yang telah tersimpan sebelumnya. *Use case* melihat laporan merupakan fitur yang memungkinkan super admin dan operator untuk dapat melihat data rekap pentahapan yang telah dilakukan. Gambaran dari desain sistem untuk *use case* melihat laporan secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.3.1, *activity diagram* seperti pada lampiran B.2.1, dan *sequence diagram* seperti pada lampiran C.2.1.

4.3.5 Increment Pemetaan

Increment pemetaan merupakan *increment* dalam sistem informasi pemetaan pemetaan strata desa siaga aktif dengan metode AHP yang menampilkan data hasil pentahapan data alternatif desa siaga dalam bentuk peta digital. Berikut merupakan pemaparan dari desain sistem yang akan dibangun berdasarkan masing-masing *use case* yang ada. *class diagram increment* proses penentuan strata desa siaga dapat digambarkan seperti pada lampiran D.3, dan *entity relationship diagram* dapat digambarkan seperti dalam lampiran E.3.

4.3.5.1 Use Case Melihat Peta Persebaran Strata Desa Siaga

Sesuai dengan kebutuhan fungsional, sistem mampu menampilkan peta persebaran strata desa siaga aktif berdasarkan hasil pentahapan yang dilakukan terhadap data desa yang ada. Fitur ini dapat diakses oleh semua pengguna sistem tanpa harus

melakukan *login*. Gambaran dari desain sistem untuk *use case* melihat peta persebaran strata desa siaga secara lebih detail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.2, *activity diagram* seperti pada lampiran B.3, dan *sequence diagram* seperti pada lampiran C.3.

4.3.6 Increment Manajemen User

Increment manajemen *user* merupakan *increment* yang berfungsi untuk super admin dapat mengelola data *user*. Fungsi-fungsi tersebut yaitu menambah data *user*, mengedit data *user*, menghapus data *user*, dan melihat *history*. Berikut merupakan pemaparan dari desain sistem yang akan dibangun berdasarkan masing-masing *use case* yang ada. *class diagram increment* manajemen *user* dapat digambarkan seperti pada lampiran D.4 dan *entity relationship diagram* seperti dalam lampiran E.4.

4.3.6.1 Use Case Menambah Data User

Use case menambah data user merupakan sebuah fitur yang digunakan oleh pengguna level 1 atau super admin untuk dapat menambahkan data pengguna baru. Gambaran dari desain sistem untuk *use case* menambah data user secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.4.1, *activity diagram* seperti pada lampiran B.4.1, dan *sequence diagram* seperti pada lampiran C.4.1.

4.3.6.2 Use Case Mengedit Data User

Use case mengedit data user berfungsi untuk mengubah data user yang telah ditambahkan sebelumnya dan hanya dapat diakses oleh super admin. Gambaran dari desain sistem untuk *use case* mengedit data user secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.4.2, *activity diagram* seperti pada lampiran B.4.2, dan *sequence diagram* seperti pada lampiran C.4.2.

4.3.6.3 Use Case Menghapus Data User

Use case menghapus data user merupakan fitur yang berfungsi untuk super admin dapat menghapus data user yang telah ditambahkan sebelumnya. Gambaran dari

desain sistem untuk *use case* menghapus data user secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.4.3, *activity diagram* seperti pada lampiran B.4.3, dan *sequence diagram* seperti pada lampiran C.4.3.

4.3.6.4 Use Case Melihat Data User

Use case melihat data user merupakan fitur yang berfungsi untuk super admin dapat melihat data user yang telah ditambahkan sebelumnya. Gambaran dari desain sistem untuk *use case* melihat data user secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.4.4, *activity diagram* seperti pada lampiran B.4.4, dan *sequence diagram* seperti pada lampiran C.4.4.

4.3.6.5 Use Case Melihat History

Use case melihat *history* merupakan fitur berfungsi untuk super admin dapat melihat *record* aktivitas yang dilakukan oleh pengguna terhadap sistem. Gambaran dari desain sistem untuk *use case* melihat *history* secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.4.5, *activity diagram* seperti pada lampiran B.4.5, dan *sequence diagram* seperti pada lampiran C.4.5.

4.3.6.6 Use Case Login

Use case login merupakan sebuah fitur yang digunakan oleh semua pengguna untuk dapat masuk dan menggunakan sistem. Proses *login* pada sistem dibatasi hanya untuk pengguna yang memiliki *username* dan *password*. Kemudian setiap user memiliki hak akses untuk masuk ke sistem yang memberikan batasan dalam mengakses fitur-fitur yang ada pada sistem. Gambaran dari desain sistem untuk *use case* melihat *history* secara lebih mendetail dapat digambarkan melalui *use case scenario* seperti pada lampiran A.5, *activity diagram* seperti pada lampiran B.5, dan *sequence diagram* seperti pada lampiran C.5, *class diagram* seperti pada lampiran D.5, dan *entity relationship diagram* seperti dalam lampiran E.5.

4.4 Integrasi Sistem

Desain sistem yang telah dibuat sesuai dengan pembagian *increment* yang didalamnya terbagi menjadi beberapa *use case*, selanjutnya akan dipadukan menjadi sebuah desain sistem yang utuh. Sehingga desain tersebut dapat menggambarkan dan menjelaskan semua kebutuhan dari sistem. Desain sistem yang dapat dipadukan menjadi sebuah kesatuan gambaran desain sistem yang utuh adalah *class diagram* dan *entity relationship diagram*. *Class diagram* merupakan desain sistem yang menggambarkan struktur sistem dari segi pendeskripsian kelas-kelas yang dibuat untuk membangun sistem. Sedangkan *entity relationship diagram* merupakan sebuah diagram yang memodelkan relasi basis data yang akan diterapkan pada sistem. Integrasi kelas-kelas pada sistem yang akan dibangun berdasarkan *use case* yang ada dapat dilihat dalam lampiran D.6. Sedangkan pemodelan relasi basis data sistem informasi pemetaan strata desa siaga dapat dilihat dalam lampiran E.6.

4.5 Pengkodean Sistem

Tahapan yang dilakukan setelah desain sistem dapat digambarkan secara detail melalui *use case diagram*, *use case scenario*, *activity diagram*, *sequence diagram*, *class diagram*, dan *entity relationship diagram* adalah melakukan tahap pengkodean (*coding*). Tahap pengkodean dalam pembuatan sistem ini dilakukan dengan menggunakan bahasa pemrograman PHP (*Page Hypertext Pre-Processor*) dengan *Framework Code Igniter* yang terintegrasi dengan *database* menggunakan *PsotgreeSQL* dan *tool phpPgAdmin5.1*. Tahapan pengkodean pada sistem informasi pemetaan strata desa siaga aktif dilakukan berdasarkan pembagian *increment* yang telah ditentukan sebelumnya.

Implementasi metode AHP dalam sistem informasi pemetaan strata desa siaga dapat dipaparkan dalam baris kode yang dibangun menggunakan *Framework Code Igniter* terdiri dari model, *controller* dan *view*. Gambaran baris kode dari implementasi metode AHP dalam sistem informasi pemetaan strata desa siaga dapat dilihat pada lampiran F.

4.6 Pengujian Sistem

Tahapan pengujian sistem merupakan suatu tahapan yang dilakukan secara sistematis untuk menguji dan mengevaluasi sistem dengan menggunakan sebuah metode pengujian sistem. Hal tersebut dilakukan dengan tujuan untuk mengevaluasi apakah kebutuhan sistem telah terpenuhi dan sistem layak untuk digunakan oleh pengguna. Agar pengujian yang dilakukan lebih valid, maka tahap pengujian sistem informasi pemetaan strata desa siaga ini dilakukan dengan menggunakan dua metode, yaitu *white box* dan *black box*.

4.6.1. Metode *White Box*

Pengujian sistem dengan metode *white box* dilakukan untuk menguji sistem dari segi desain dan kode program. Hal tersebut bertujuan untuk mengevaluasi apakah sistem mampu menghasilkan fungsi-fungsi, inputan, dan keluaran yang sesuai dengan spesifikasi dari kebutuhan sistem itu sendiri. Pengujian dengan metode *white box* dilakukan oleh penulis dengan cara menghitung *independent path* yaitu dengan menggunakan suatu pengukuran kuantitatif *cyclomatic complexity*, *listing program*, penentuan jalur independen, dan *test case*.

Tahapan-tahapan pengujian dengan metode *white box* ini akan diterapkan pada beberapa fitur yang dinilai dapat mewakili sistem informasi pemetaan strata desa siaga. Beberapa fitur tersebut antara lain:

- a. Pembobotan Kriteria
- b. Pembobotan Subkriteria
- c. Pentahapan Strata Desa Siaga

Selanjutnya pemaparan lebih detail tentang pengujian sistem dengan metode *white box* dapat dilihat dalam lampiran G.

4.6.2. Metode *Black Box*

Pengujian *black box* berfungsi untuk menguji sistem dari segi spesifikasi fungsional sistem dengan tujuan mengetahui apakah fungsi-fungsi, inputan, dan keluaran sistem sesuai dengan spesifikasi yang dibutuhkan oleh pengguna. Pengujian dengan metode *black box* dilakukan oleh operator bagian promosi kesehatan Dinas Kesehatan Kab. Jember. Hasil pengujian dengan metode *black box* dapat dilihat pada lampiran H.



BAB V HASIL DAN PEMBAHASAN

Bab ini menjelaskan mengenai hasil penelitian yang telah dilakukan serta pembahasan sistem yang telah dibuat. Pembahasan dilakukan guna menjelaskan dan memaparkan bagaimana penelitian ini menjawab perumusan masalah serta tujuan dan manfaat dari penelitian ini seperti apa yang telah ditentukan pada awal penelitian.

5.1 Hasil Analisis Data Kriteria

Analisis data kriteria pentahapan strata desa siaga aktif dalam penelitian ini dilakukan dengan menggunakan metode AHP. Untuk dapat menganalisis data kriteria yang ada hingga dapat menghasilkan suatu hasil yang tepat, maka penulis harus melalui tahapan atau proses dalam metode AHP itu sendiri.

5.1.1. Pentahapan Strata Desa Siaga Aktif

Dinas Kesehatan Kabupaten Jember sebagai instansi pemerintahan memiliki tanggungjawab serta kewenangan untuk melakukan *controlling* terhadap kondisi masyarakat di Kabupaten Jember khususnya di bidang kesehatan. Sejalan dengan keputusan Kementerian Kesehatan RI tentang desa siaga, Dinas Kesehatan Kabupaten Jember juga melakukan fungsi *control* terhadap kondisi kesehatan di seluruh wilayah Kabupaten Jember dengan melakukan pendataan-pendataan yang termasuk didalamnya terdapat pendataan desa siaga. Namun selama ini kegiatan *controlling* terhadap desa siaga hanya dilakukan sebatas pengumpulan data yang diperoleh dari seluruh puskesmas di Kabupaten Jember, selanjutnya tidak dilakukan pentahapan secara berkala. Hal tersebut disebabkan pentahapan desa siaga harus dilakukan secara manual sesuai dengan pedoman pentahapan strata desa siaga aktif yang telah ditentukan oleh Kementerian Kesehatan RI.

Berdasarkan kendala tersebut penulis mencoba melakukan analisis pentahapan strata desa siaga aktif menggunakan metode AHP. Metode AHP memiliki proses perbandingan dari masing-masing kriteria dalam bentuk angka yaitu angka 1 hingga

angka 9 yang selanjutnya diproses pada sebuah matrik perbandingan berpasangan. Angka 1 hingga 9 menunjukkan perbandingan tingkat kepentingan dari masing-masing elemen. Penentuan nilai angka perbandingan yang diberikan terhadap masing-masing elemen dilakukan oleh pihak promosi kesehatan Dinas Kesehatan Kabupaten Jember. Tabel penentuan nilai perbandingan dapat dilihat dalam lampiran.

5.1.2. Penyusunan Hirarki

Tahap pertama yang dilakukan dalam proses analisis menggunakan metode AHP yaitu melakukan penyusunan hirarki dari permasalahan yang ada pada proses pentahapan strata desa siaga aktif. Proses penyusunan hirarki dibagi menjadi beberapa bagian, bagian yang pertama yaitu menentukan tujuan (*goal*) dari permasalahan yang akan dianalisis menggunakan metode AHP. Tujuan dari penelitian ini adalah melakukan pentahapan strata desa siaga untuk selanjutnya ditampilkan dalam bentuk peta digital. Tahap selanjutnya yaitu menentukan kriteria, subkriteria, dan alternatif. Pentahapan strata desa siaga aktif memiliki delapan kriteria. Kriteria tersebut yaitu adanya forum desa/kelurahan, KPM/kader kesehatan, kemudahan akses pelayanan kesehatan dasar, posyandu dan UKBM, adanya sumber dukungan dana untuk kegiatan kesehatan, adanya peran serta masyarakat dan organisasi kemasyarakatan, peraturan kepala desa atau bupati/walikota, serta tingkat pembinaan PHBS. Setiap kriteria tersebut memiliki subkriteria dimana nilai dari keduanya digunakan untuk menghasilkan penilaian terhadap alternatif desa siaga aktif.

5.1.3. Penentuan Prioritas Kriteria

Tahapan dasar yang dilakukan dalam proses analisis menggunakan metode AHP adalah proses perhitungan menggunakan matrik perbandingan berpasangan. Proses tersebut nantinya akan menghasilkan sebuah nilai prioritas kriteria. Nilai prioritas kriteria itu sendiri diperoleh dari perbandingan nilai antar kriteria dalam hal seberapa penting kriteria tersebut terhadap pencapaian tujuan analisis (pentahapan strata desa siaga aktif). Sehingga nilai tersebut dapat menunjukkan tingkat prioritas

antar kriteria yang ada dalam pentahapan strata desa siaga aktif. Nilai prioritas inilah yang kemudian digunakan sebagai nilai bobot kriteria untuk kemudian diolah kembali dengan bobot subkriteria yang ada.

5.1.3.1. Perhitungan Manual

Analisis data kriteria penentuan strata desa siaga menggunakan metode AHP sebelum diterapkan pada sistem, penulis menganalisis secara manual. Setelah menyusun hirarki maka tahapan selanjutnya yaitu menentukan bobot kriteria. Penentuan nilai bobot kriteria dilakukan dengan beberapa proses atau tahapan perhitungan, yaitu:






- a. Menentukan matrik perbandingan berpasangan antar kriteria.

Seperti yang telah kita ketahui bahwa kriteria yang digunakan dalam pentahapan strata desa siaga aktif terdapat delapan kriteria. Maka selanjutnya kita akan membandingkan kriteria-kriteria tersebut dalam sebuah matrik perbandingan berpasangan seperti pada Tabel 5.1

Tabel 5.5 Matrik Perbandingan Berpasangan

Kriteria	a	b	c	d	e	f	g	h
a	1	1	1	1	1	1	1	1
b	1	1	1	3	5	5	5	1
c	1	1	1	1	5	5	5	1
d	1	0.33	1	1	5	5	5	1
e	1	0.2	0.2	0.2	1	1	5	1
f	1	0.2	0.2	0.2	1	1	1	1
g	1	0.2	0.2	0.2	0.2	1	1	1
h	1	1	1	1	1	1	1	1
Jumlah	8	4.93	5.6	7.6	19.2	20	24	8

Keterangan Tabel 5.1:

-  Merupakan kriteria desa siaga aktif
-  Merupakan nilai perbandingan kriteria terhadap dirinya sendiri
-  Merupakan nilai skala perbandingan
-  Merupakan nilai perbandingan sehingga nilainya adalah 1/nilai kriteria
-  Merupakan jumlah dari nilai per kolom

b. Menentukan Rengking Kriteria

Setelah membuat matrik perbandingan berpasangan, selanjutnya yaitu menghitung atau menentukan rangking kriteria dalam bentuk vektor prioritas. Nilai vektor prioritas diperoleh dari hasil pembagian antara nilai baris terhadap nilai jumlah per kolom, yaitu 1/8, 1/4.93, dst.

c. Menghitung Eigen Vektor Normalisasi

Proses menghitung nilai eigen vektor normalisasi dilakukan dengan menjumlahkan nilai setiap baris, kemudian membagi dengan jumlah data yang dijumlahkan. Hasil dari perhitungan tersebut dapat dilihat pada Tabel 5.2

Tabel 5.6 Matrik Perhitungan Nilai Eigen Vektor Normalisasi

	a	b	c	d	e	f	g	h	Jumlah	eigen vec
a	0.125	0.20	0.18	0.13	0.05	0.05	0.04	0.13	0.91	0.11
b	0.125	0.20	0.18	0.39	0.26	0.25	0.21	0.13	1.74	0.22
c	0.125	0.20	0.18	0.13	0.26	0.25	0.21	0.13	1.48	0.19
d	0.125	0.07	0.18	0.13	0.26	0.25	0.21	0.13	1.35	0.17
e	0.125	0.04	0.04	0.03	0.05	0.05	0.21	0.13	0.66	0.08
f	0.125	0.04	0.04	0.03	0.05	0.05	0.04	0.13	0.50	0.06
g	0.125	0.04	0.04	0.03	0.01	0.05	0.04	0.13	0.45	0.06
h	0.125	0.20	0.18	0.13	0.05	0.05	0.04	0.13	0.91	0.11

d. Uji Konsistensi

Setelah diperoleh nilai eigen vektor normalisasi, maka tahap selanjutnya yaitu melakukan uji konsistensi. Pengujian ini dilakukan untuk mengetahui bahwa nilai perbandingan berpasangan yang dihasilkan bersifat konsisten.

a). Menghitung Nilai Eigen Maksimum (λ_{maks})

$$\begin{aligned}\lambda_{maks} &= (0.91/0.11)+(1.74/0.22)+(1.48/0.19)+(1.35/0.17)+(0.66/0.08)+ \\ &\quad (0.5/0.06)+(0.45/0.06)+(0.91/0.11) \\ &= 8.033\end{aligned}$$

b). Menghitung Nilai Index Consistency (CI)

$$\begin{aligned}CI &= (\lambda_{maks} - n)/n-1 \\ &= (8.033566-8)/7 \\ &= 0.004\end{aligned}$$

c). Menghitung Nilai Ratio Consistency (CR)

$$\begin{aligned}CR &= CI/RI \\ &= 0.004795/1.41 \\ &= 0.003\end{aligned}$$

Berdasarkan analisis yang telah dilakukan terhadap data kriteria pentahapan strata desa siaga aktif yang telah dianalisis secara manual tersebut, dapat kita ketahui nilai bobot prioritas dari masing-masing kriteria yaitu pada kolom *eigen value*. Nilai bobot prioritas tersebut telah diuji konsistensinya melalui beberapa tahapan uji konsistensi yaitu menghasilkan nilai $CR < 0,1$.

5.1.3.2. Perhitungan Sistem

Analisis secara manual dilakukan oleh penulis guna mengetahui alur logis analisis data kriteria pentahapan strata desa siaga serta sebagai media pengujian apakah metode AHP yang diimplementasikan dalam sistem dapat melakukan analisis sehingga menghasilkan nilai yang tepat. Maka setelah proses analisis secara manual telah diketahui hasilnya, selanjutnya penulis dapat mengimplementasikan alur logis proses analisis menggunakan metode AHP ke dalam sistem informasi pemetaan strata desa siaga aktif. Pemaparan secara detail proses penentuan nilai bobot kriteria akan dibahas pada sub bab implementasi sistem.

5.1.4. Penentuan Prioritas Subkriteria

Nilai bobot prioritas yang telah diperoleh dalam proses penentuan nilai bobot prioritas kriteria selanjutnya akan diproses dengan nilai bobot prioritas dari sub kriteria yang dimiliki oleh masing-masing kriteria strata desa siaga aktif. Tahapan-

tahapan analisis yang dilakukan untuk memperoleh nilai bobot prioritas sub kriteria sama dengan tahapan-tahapan yang dilakukan dalam menentukan nilai bobot prioritas AHP. Pemaparan secara detail proses penentuan nilai bobot sub kriteria akan dibahas pada sub bab implementasi sistem.

5.1.5. Analisis Pentahapan Strata Desa Siaga Aktif

Analisis pentahapan strata desa siaga aktif dilakukan setelah diperoleh bobot prioritas kriteria dan subkriteria. Seperti yang telah kita ketahui bahwa desa siaga aktif sendiri memiliki 4 (empat) tahapan strata yaitu :

- a. Desa dan kelurahan siaga aktif Pratama
- b. Desa dan kelurahan siaga aktif Madya
- c. Desa dan kelurahan siaga aktif Purnama
- d. Desa dan kelurahan siaga aktif Mandiri

Setelah proses penentuan prioritas kriteria dan subkriteria menggunakan metode AHP sebelumnya, selanjutnya nilai tersebut diolah dan digunakan untuk melakukan pentahapan strata desa siaga aktif terhadap desa-desa di Kabupaten Jember. selama ini Dinas Kesehatan Kabupaten Jember melakukan pentahapan terhadap desa siaga aktif secara manual dengan mencocokkan masing-masing data desa yang telah dihimpun dari seluruh puskesmas di Kabupaten Jember dengan kriteria pentahapan strata desa siaga aktif. Dalam sistem informasi pemetaan strata desa siaga aktif penulis mencoba mengimplementasikan metode AHP agar sistem dapat melakukan analisis terhadap data dari masing-masing desa yang diinputkan. Sesuai dengan pedoman pentahapan strata desa siaga aktif oleh KEMENKES RI penulis membuat aturan batasan interval dari masing-masing strata desa siaga aktif. Penentuan aturan tersebut dilakukan dengan cara mencari nilai dari masing-masing strata dari perkalian nilai bobot kriteria dan subkriteria. Berikut merupakan tahapan yang dilakukan untuk mendapatkan batasan interval masing-masing strata desa siaga aktif, yaitu:

- 1) Mengalikan dan menjumlahkan nilai bobot kriteria dan subkriteria yang sesuai dengan ketentuan tahapannya strata desa siaga seperti pada Tabel 5.3.

Tabel 5.7 Penentuan Batas Interval Strata Desa Siaga Aktif

No	Kriteria	Subkriteria	Nilai Bobot
1	Forum masyarakat desa/ kelurahan	Ada, tetapi belum jalan	a
2	KPM/kader kesehatan	Sudah ada minimal 2 orang	b
3	kemudahan akses pelayanan kesehatan dasar	Ya	c
4	posyandu dan UKBM	UKBM lainnya tidak aktif	d
5	adanya sumber dukungan dana untuk kegiatan kesehatan	Sudah ada dari pemerintah desa dan kelurahan serta belum ada sumber lainnya	e
6	adanya peran serta masyarakat dan organisasi kemasyarakatan	Ada peran aktif masyarakat namun tidak ada peran aktif ormas	f
7	peraturan kepala desa atau bupati/walikota	Belum ada	g
8	serta tingkat pembinaan PHBS	Pembinaan PHBS Kurang dari 20% rumah tangga yang ada	h
Nilai batas interval untuk desa siaga pratama yaitu $(\text{Bobot_kriteria} * a) + (\text{Bobot_kriteria} * b) + (\text{Bobot_kriteria} * c) +$ $(\text{Bobot_kriteria} * d) + (\text{Bobot_kriteria} * e) + (\text{Bobot_kriteria} * f) +$ $(\text{Bobot_kriteria} * g) + (\text{Bobot_kriteria} * h)$			

Selanjutnya dilakukan juga terhadap tiga strata desa siaga aktif yang lain, yaitu desa siaga aktif madya, purnama, dan mandiri. Selain itu juga dilakukan perhitungan untuk perkalian nilai bobot kriteria dan subkriteria maksimum

serta perkalian nilai bobot kriteria dan subkriteria minimum sebagai batas atas dan batas bawah interval pentahapan strata desa siaga aktif.

- 2) Membuat batasan interval dari masing-masing strata desa siaga aktif. Berdasarkan nilai-nilai yang telah diperoleh pada tahapan sebelumnya, maka selanjutnya menentukan batasan atau pembagian interval strata desa siaga aktif seperti dalam Tabel 5.4.

Tabel 5.8 Tabel Penentuan Nilai Interval

Strata	Batas Atas	Batas Bawah
Pratama	perkalian nilai bobot kriteria dan subkriteria maksimum	Nilai batas interval desa siaga mandiri
Madya	Nilai batas interval desa siaga mandiri - 1	Nilai batas interval desa siaga Purnama
Purnama	Nilai batas interval desa siaga Purnama - 1	Nilai batas interval desa siaga Madya
Mandiri	Nilai batas interval desa siaga Madya - 1	perkalian nilai bobot kriteria dan subkriteria minimum

- 3) Setiap data desa yang diinputkan ke dalam sistem sesuai dengan kriteria yang ada, maka sistem akan otomatis melakukan cek terhadap nilai yang dicapai desa tersebut terhadap batasan interval yang telah ditentukan.

5.2 Implementasi Sistem

Implementasi sistem informasi pemetaan strata desa siaga aktif Dinas Kesehatan Kabupaten Jember akan dipaparkan dalam bentuk deskripsi dan gambar tampilan dari sistem sesuai dengan fitur yang ada dalam setiap increment yang telah ditentukan.

5.2.1. *Increment* Proses Penentuan Strata Desa Siaga

Increment proses penentuan strata desa siaga terdiri atas enam belas fitur. Fitur-fitur tersebut saling berhubungan dalam proses menentukan strata desa siaga.

5.2.1.1. Menambah Data Puskesmas

Fitur menambah data puskesmas merupakan fitur yang berfungsi untuk menambah data puskesmas. Fitur ini dijalankan ketika user menekan tombol tambah data, kemudian sistem menampilkan form tambah data puskesmas. Setelah user mengisi form dan menekan tombol simpan, maka sistem akan menyimpan data kedalam database. Sedangkan jika user mengisi form kurang lengkap atau tidak sesuai, maka sistem akan menampilkan pemberitahuan. Tampilan dari fitur tambah data dapat dilihat pada lampiran I.1.1.

5.2.1.2. Mengubah Data Puskesmas

Fitur mengubah data puskesmas merupakan fitur yang berfungsi untuk mengubah data puskesmas yang telah ditambahkan sebelumnya. Fitur ini dijalankan ketika user menekan *icon* edit, kemudian sistem menampilkan form edit data puskesmas. Setelah user mengisi form dan menekan tombol simpan, maka sistem akan menyimpan data kedalam database. Sedangkan jika user mengisi form kurang lengkap atau tidak sesuai, maka sistem akan menampilkan pemberitahuan. Tampilan dari fitur mengubah data puskesmas dapat dilihat pada lampiran I.1.2.

5.2.1.3. Menghapus Data Puskesmas

Fitur menghapus data puskesmas berfungsi untuk menghapus data puskesmas yang dipilih dari database. Fungsi ini akan dijalankan ketika user menekan *icon delete* pada

tabel data puskesmas. Sebelum sistem menghapus data dari database, sistem akan menampilkan kotak dialog konfirmasi hapus. Tampilan dari fitur menghapus data puskesmas dapat dilihat pada lampiran I.1.3.

5.2.1.4. Melihat Data Puskesmas

Fungsi fitur melihat data puskesmas yang telah disimpan sebelumnya dapat dijalankan ketika user memilih submenu data puskesmas. Data puskesmas ditampilkan dalam bentuk tabel. Tampilan dari fitur melihat data puskesmas dapat dilihat pada lampiran I.1.4.

5.2.1.5. Menambah Data Desa

Fitur menambah data desa merupakan fitur yang berfungsi untuk menambah data desa ke dalam database. Fitur ini dijalankan ketika user menekan tombol tambah data, kemudian sistem menampilkan form tambah data desa. Setelah user mengisi form dan menekan tombol simpan, maka sistem akan menyimpan data ke dalam database. Sedangkan jika user mengisi form kurang lengkap atau tidak sesuai, maka sistem akan menampilkan pemberitahuan. Tampilan dari fitur tambah data desa dapat dilihat pada lampiran I.1.5.

5.2.1.6. Mengubah Data Desa

Fitur mengubah data desa merupakan fitur yang berfungsi untuk mengubah data desa yang telah ditambahkan sebelumnya. Fitur ini dijalankan ketika user menekan *icon* edit, kemudian sistem menampilkan form edit data desa. Setelah user mengisi form dan menekan tombol simpan, maka sistem akan menyimpan data ke dalam database. Sedangkan jika user mengisi form kurang lengkap atau tidak sesuai, maka sistem akan menampilkan pemberitahuan. Tampilan dari fitur mengubah data desa dapat dilihat pada lampiran I.1.6.

5.2.1.7. Melihat Data Desa

Fungsi fitur melihat data desa yang telah disimpan sebelumnya dapat dijalankan ketika user memilih submenu data desa. Data desa ditampilkan dalam bentuk tabel. Tampilan dari fitur melihat data puskesmas dapat dilihat pada lampiran I.1.7.

5.2.1.8. Melihat Data Kriteria

Fungsi fitur melihat data kriteria dapat dijalankan ketika user memilih menu kriteria. Data kriteria ditampilkan dalam bentuk tabel. Tampilan dari fitur melihat data kriteria dapat dilihat pada lampiran I.1.8.

5.2.1.9. Pembobotan Kriteria

Fitur pembobotan kriteria merupakan fitur yang terdapat implementasi metode AHP didalamnya. Tampilan dari fitur pembobotan kriteria dapat dilihat pada lampiran I.1.9.

5.2.1.10. Mengubah Bobot Kriteria

Bobot kriteria dapat diperbarui oleh super admin melalui fitur mengubah bobot kriteria. Tampilan dari fitur mengubah bobot kriteria dapat dilihat pada lampiran I.1.10.

5.2.1.11. Melihat Data Subkriteria

Fungsi fitur melihat data subkriteria dapat dijalankan ketika user memilih menu subkriteria. Data subkriteria ditampilkan dalam bentuk tabel. Tampilan dari fitur melihat data subkriteria dapat dilihat pada lampiran I.1.8

5.2.1.12. Pembobotan Subkriteria

Fitur pembobotan subkriteria merupakan fitur yang terdapat implementasi metode AHP untuk mendapatkan nilai bobot subkriteria. Tampilan dari fitur pembobotan subkriteria dapat dilihat pada lampiran I.1.12.

5.2.1.13. Mengubah Bobot Subkriteria

Bobot subkriteria dapat diperbarui oleh super admin melalui fitur mengubah bobot subkriteria. Tampilan dari fitur mengubah bobot subkriteria dapat dilihat pada lampiran I.1.13.

5.2.1.14. Pembobotan Alternatif Desa Siaga

Pembobotan alternatif desa siaga merupakan fitur yang digunakan untuk melakukan pembobotan capaian desa/kelurahan terhadap kriteria pentahapan strata desa siaga aktif. Tampilan dari fitur mengubah bobot subkriteria dapat dilihat pada lampiran I.1.14.

5.2.1.15. Mengubah Nilai Pembobotan Alternatif

Mengubah nilai pembobotan alternatif desa siaga merupakan fitur yang digunakan untuk melakukan pembaruan terhadap capaian desa/kelurahan. Tampilan dari fitur mengubah bobot subkriteria dapat dilihat pada lampiran I.1.15.

5.2.1.16. Melihat Hasil Pembobotan Alternatif

Hasil dari pembobotan capaian desa/kelurahan terhadap kriteria dapat ditampilkan pada fitur melihat hasil pembobotan alternatif. Tampilan dari fitur mengubah bobot subkriteria dapat dilihat pada lampiran I.1.16.

5.2.2. *Increment* Pemetaan

Increment pemetaan hanya terdiri dari satu fitur yaitu fitur melihat peta strata desa siaga. Fitur ini berada pada halaman utama sistem sehingga dapat diakses oleh semua pengguna sistem meskipun tidak memiliki *username* dan *password*. Selain melihat persebaran strata desa siaga, pengguna akan mendapatkan informasi lebih detail mengenai desa siaga yang ditampilkan seperti yang ditampilkan pada gambar dalam lampiran I.3.

5.2.3. *Increment* Laporan

Increment laporan terdiri dari dua fitur yaitu menampilkan data rekap hasil pentahapan strata desa siaga aktif dalam bentuk tabel sesuai dengan tanggal dilakukannya pentahapan. Gambar dari tampilan implementasi sistem terdapat pada lampiran I.2.

5.2.4. *Increment* Manajemen User

Increment manajemen user memiliki beberapa fitur yang mendukung fungsi manajemen user yang ada pada sistem. Fitur-fitur tersebut adalah menambah data user, mengedit data user, menghapus data user, dan melihat data user.

5.2.4.1. Menambah Data User

Fitur menambah data user berfungsi bagi super admin untuk dapat menambahkan data user baru sesuai dengan level yang akan diberikan. Gambar dari tampilan implementasi sistem terdapat pada lampiran I.4.1.

5.2.4.2. Mengedit Data User

Fitur mengedit data user merupakan fitur yang memungkinkan super admin untuk dapat mengubah data user yang telah ditambahkan sebelumnya. Gambar dari tampilan implementasi sistem terdapat pada lampiran I.4.2.

5.2.4.3. Menghapus Data User

Fitur menghapus data user merupakan fitur yang memungkinkan super admin untuk dapat menghapus data user. Gambar dari tampilan implementasi sistem terdapat pada lampiran I.4.3.

5.2.4.4. Melihat Data User

Fitur melihat data user merupakan fitur yang memungkinkan super admin untuk dapat menampilkan data user yang telah disimpan dalam *database*. Gambar dari tampilan implementasi sistem terdapat pada lampiran I.4.4.

5.2.5. Use Case Login

Use case login merupakan fitur yang berfungsi memverifikasi *username* dan *password* pengguna yang akan mengakses sistem. Gambar dari tampilan implementasi sistem terdapat pada lampiran I.4.5.

5.3 Pembahasan

5.3.1. Pembahasan Hasil Implementasi Sistem Pemetaan Strata Desa Siaga Aktif dengan Metode AHP

Implementasi sistem informasi pemetaan strata desa siaga aktif dengan metode AHP terhadap tujuan dari penelitian yaitu untuk membantu mempermudah Dinas Kesehatan Kabupaten Jember dalam mengontrol dan menentukan kebijakan terhadap persebaran strata desa siaga aktif di Kabupaten Jember terdapat beberapa kelebihan dan kekurangan yang masih harus disempurnakan kembali. Untuk mencapai tujuan tersebut sistem ini memiliki fitur utama yaitu fitur yang mengimplementasikan analisis metode AHP. Fitur tersebut memiliki alur analisis yang sesuai dengan tahapan-tahapan analisis dalam metode AHP yang hanya dapat diakses oleh pengguna dengan level 1 atau sebagai super admin. Pembatasan hak akses pada fitur-fitur dalam sistem ini dimaksudkan agar tidak sembarang orang dapat mengontrol jalannya sistem secara keseluruhan. Termasuk dalam proses menentukan pembobotan nilai prioritas kriteria dan nilai prioritas subkriteria. Sebab sistem ini memiliki fungsi otomatis *update* terhadap nilai interval strata desa siaga aktif ketika nilai pembobotan kriteria dan subkriteria dilakukan perubahan.

Fungsi *control* yang harus dilakukan oleh Dinas Kesehatan Kabupaten Jember serta pihak-pihak yang terkait dapat ditunjang dengan adanya fitur yang menampilkan hasil pentahapan strata desa siaga dengan metode AHP dalam bentuk peta digital. Peta digital tersebut menampilkan desa-desa yang telah dilakukan penilaian atau pentahapan dengan warna-warna yang berbeda sesuai dengan strata yang dicapai oleh masing-masing desa/kelurahan. Untuk dapat lebih menunjang, maka peta digital memberikan fungsi tambahan yaitu dapat menampilkan *info window* dari masing-

masing desa yang berisi detail hasil pentahapan. Agar tidak hanya dapat diketahui setiap kali dilakukan pentahapan, maka sistem memiliki menu laporan dengan fungsi menampilkan data laporan hasil pentahapan yang dilakukan dalam bentuk tabel yang dapat ditampilkan sesuai dengan tanggal dilakukannya pentahapan.

Mencegah terjadinya kecurangan dalam proses pentahapan, maka sistem memiliki fungsi otomatis menyimpan tanggal ketika dilakukan pentahapan terhadap data desa/kelurahan. Selanjutnya ketika dilakukan pentahapan ulang terhadap data desa/kelurahan di tanggal yang sama, maka sistem akan langsung meng-*update* data laporan dengan nilai hasil pentahapan yang terakhir kali dilakukan. Selain itu untuk dapat mengontrol dan mengawasi aktivitas yang dilakukan *user* pada sistem, sistem informasi pemetaan ini memiliki fitur history yang hanya dapat diakses oleh super admin. Sehingga super admin dapat mengetahui aktivitas apa saja yang dilakukan oleh *user* pada sistem.

Kemampuan sistem dalam mencapai tujuan yang telah ditentukan di awal masih terdapat banyak bagian yang harus diperbaiki dan dikembangkan lagi. Peta digital yang ditampilkan dalam sistem ini hanya sebatas pada hasil pentahapan yang dilakukan paling terakhir. Seharusnya sistem mampu menampilkan peta digital dengan hasil pentahapan berdasarkan tanggal. Selain itu, sistem yang dibangun tidak memungkinkan pengguna untuk mengubah, menambah atau mengurangi jumlah kriteria dan subkriteria yang digunakan dalam pentahapan strata desa siaga.

Batasan wilayah yang digunakan dalam penelitian memberikan pengaruh pada hasil analisis yang dilakukan. Berdasarkan data desa yang dianalisis oleh sistem dapat diketahui bahwa desa-desa yang menjadi objek penelitian tidak dapat mewakili 4 tahapan strata desa siaga aktif. Selain itu semakin banyak jumlah desa yang dianalisis, maka semakin terlihat bagaimana kondisi persebaran strata desa siaga aktif di Kabupaten Jember.

5.3.2. Pembahasan Hasil Pengujian Sistem

Pengujian program dalam penelitian ini dilakukan dengan menggunakan dua metode pengujian sistem, yaitu metode pengujian *white box* dan *black box*. Pada pengujian *white box* digunakan *cyclometric complexity* untuk menghitung kompleksitas suatu method. Kompleksitas suatu method dapat diuji dengan mengetahui berapa jumlah *test case* yang diperlukan untuk mengevaluasi method tersebut sehingga setiap percabangan didalam method tersebut pernah dilalui. Semakin banyak jumlah *test case* maka method dapat dikatakan semakin kompleks. Dalam sistem informasi pemetaan strata desa siaga ini jumlah *test case* yang diperlukan tidak lebih dari 5. Sehingga sistem ini tingkat kompleksitasnya dapat dikatakan cukup rendah.

Metode pengujian *black box* digunakan untuk mengetahui apakah fungsi-fungsi, inputan, dan keluaran sistem sesuai dengan spesifikasi yang dibutuhkan oleh pengguna. Maka berdasarkan pengujian *black box* yang telah dilakukan, dapat diketahui bahwa sistem informasi pemetaan strata desa siaga aktif dengan metode AHP telah berhasil dibangun sesuai dengan kebutuhan fungsional dari proses pentahapan strata desa siaga aktif.

BAB VI PENUTUP

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

6.1. Kesimpulan

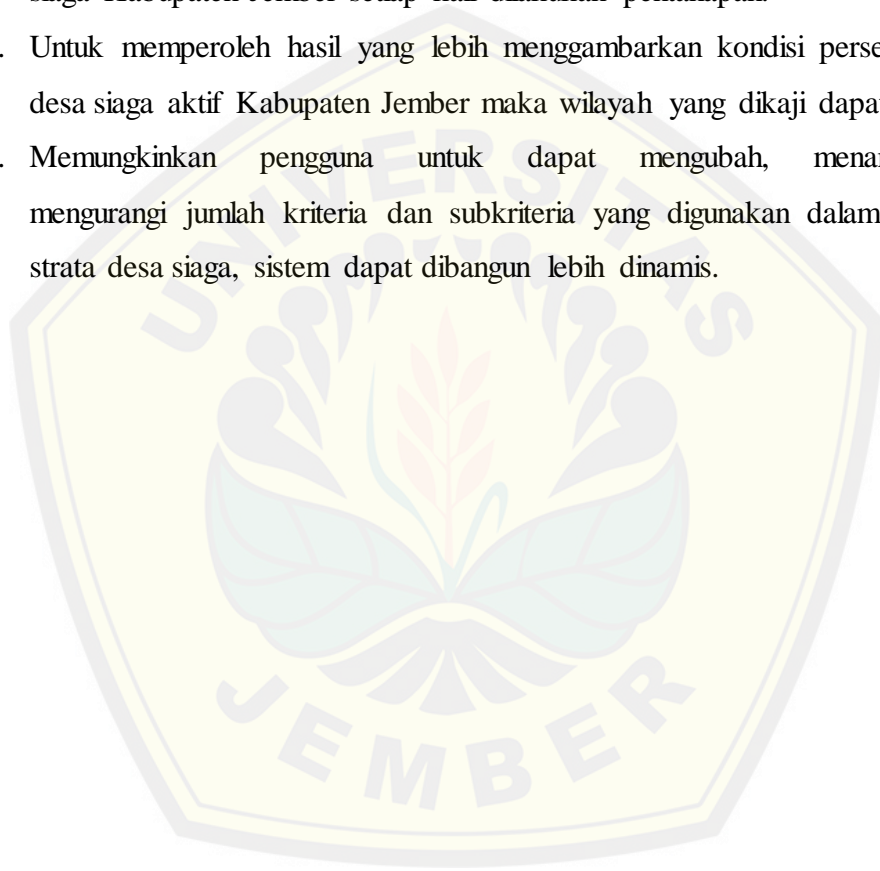
Kesimpulan dari hasil penelitian yang telah dilakukan oleh peneliti adalah sebagai berikut:

1. Pentahapan strata desa siaga aktif dapat dianalisis dengan metode AHP. Analisis pentahapan strata desa siaga aktif dapat dilakukan dengan menggunakan perbandingan berpasangan skala prioritas sesuai dengan tahapan dalam metode AHP. Maka hal yang paling mempengaruhi hasil analisis pentahapan strata desa siaga adalah penentuan nilai skala prioritas kriteria dan subkriteria yang digunakan.
2. Untuk dapat merancang dan membangun sistem informasi pemetaan strata desa siaga aktif DINKES Kab.Jember dengan menerapkan metode AHP diperlukan pemahaman yang baik terhadap proses pentahapan strata desa siaga aktif yang dilakukan secara manual. Sebab setelah proses analisis yang dilakukan dengan metode AHP untuk memperoleh nilai bobot prioritas kriteria dan subkriteria, terdapat proses penentuan batas nilai interval yang berfungsi untuk melakukan pengecekan terhadap nilai yang dicapai suatu desa/kelurahan terhadap strata desa siaga aktif yang ada. Sehingga hasil pentahapan yang dilakukan oleh sistem sesuai dengan ketentuan pentahapan desa siaga aktif yang telah ditetapkan oleh Kementerian Kesehatan RI.

6.2.Saran

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

1. Peta digital dapat menampilkan berdasarkan hasil pentahapan sesuai tanggal sehingga dapat diketahui bagaimana perkembangan persebaran strata desa siaga Kabupaten Jember setiap kali dilakukan pentahapan.
2. Untuk memperoleh hasil yang lebih menggambarkan kondisi persebaran strata desa siaga aktif Kabupaten Jember maka wilayah yang dikaji dapat diperluas.
3. Memungkinkan pengguna untuk dapat mengubah, menambah atau mengurangi jumlah kriteria dan subkriteria yang digunakan dalam pentahapan strata desa siaga, sistem dapat dibangun lebih dinamis.



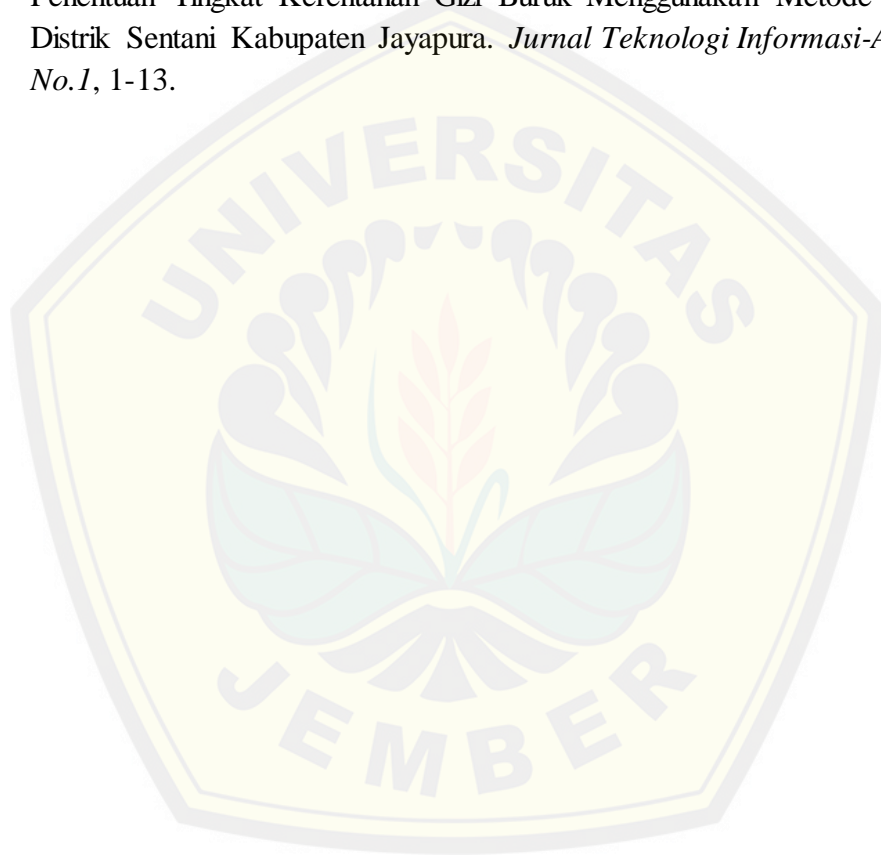
DAFTAR PUSTAKA

- Keputusan Menteri Kesehatan Nomor 564/Menkes/SK/VIII/2006.* (2006). Jakarta: Menteri Kesehatan Republik Indonesia.
- Undang-Undang Republik Indonesia Nomor 17 Tahun 2007.* (2007). Jakarta.
- Undang-Undang RI Nomor 36 Tahun 2009 tentang Kesehatan.* (2009). Jakarta.
- Pedoman Umum Pengembangan Desa dan Kelurahan Siaga Aktif.* (2010). Jakarta: Kementerian Kesehatan Republik Indonesia.
- Asmarani, A. D. (2010). *Strategi Kebijakan Pembangunan Daerah Kabupaten Klaten: Pendekatan Analisis SWOT dan AHP*. Jakarta: Universitas Indonesia.
- Heru Nurwarsito, dkk. (2010). Pengembangan Sistem Pendukung Keputusan untuk Menentukan Gizi Ibu Hamil Menggunakan Metode AHP. *DINAMICA DOTCOM Volume, 1 Nomor, 2 Juli 2010*, 116-128.
- Nazir, M. (2003). *Metode Penelitian*. Jakarta: Ghalia Indonesia.
- Niko Irijaya Desmonda dan Adjie Pamungkas. (2014). Penentuan Zona Kerentanan Bencana Gempa Bumi Tektonik di Kabupaten Malang Wilayah Selatan. *Jurnal Teknik POMITS Vol.3 No.2*, C107-C112.
- Nurman, R. M. (2007). *Sistem Informasi Pemetaan Profil Kriminalitas Berbasis Web (Studi Kasus : Kejahatan Konvensional Kota Bogor)*. Bogor: Institut Pertanian Bogor.
- R Fiati dan N Latifah. (2013). Sistem Pendukung Keputusan Untuk Menentukan Kelayakan Desa Mandiri Menggunakan FMADM. *Seminar Nasional Ilmu Komputer 2013*, 137-141.
- Rosa, S. A. (2013). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Sri Redjeki, M. Guntara, Pius Anggoro. (2014). Perancangan Sistem Identifikasi dan Pemetaan Potensi Kemiskinan untuk Optimalisasi Program Kemiskinan. *Jurnal Sistem Informasi(JSI) Vol.6 No.2*, 731-743.

Sugiyono, P. D. (2011). METODE PENELITIAN KUANTITATIF, KUALITATIF, DAN R&D. Bandung: ALVABETA, cv.

Sukarme, D. (2011). *Pengaruh Metode Evaluasi Penawaran Pengadaan Barang/Jasa Pemerintah terhadap Hasil Pekerjaan dengan Pendekatan AHP*. Semarang: Universitas Diponegoro .

Yokelin Tokoro, Dharmaputra Palekahelu, Andeka Rocky Tanaamah. (2013). SPK Penentuan Tingkat Kerentanan Gizi Buruk Menggunakan Metode AHP di Distrik Sentani Kabupaten Jayapura. *Jurnal Teknologi Informasi-Aiti Vol.10 No.1*, 1-13.



LAMPIRAN

LAMPIRAN A *Use Case Scenario*

A.1. *Increment* Proses Penentuan Strata Desa Siaga

A.1.1 Menambah Data Puskesmas

Nama usecase	menambah data puskesmas
Aktor	Admin dan operator
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password Aktor memiliki data puskesmas berupa nama puskesmas, alamat, jenis puskesmas, jumlah desa yang ditangani, jumlah dokter, jumlah bidan, jumlah perawat, jumlah POSKESDES, serta jumlah POSYANDU
Pasca kondisi	User berhasil melakukan input data puskesmas

Aksi Aktor

Reaksi Sistem

Skenario Normal

1. Klik menu administrasi	2. Menampilkan sub menu data puskesmas dan data desa
3. Klik submenu data puskesmas	4. Menampilkan halaman data puskesmas
5. Klik tombol tambah data	6. Menampilkan popup form input data puskesmas
7. Input data puskesmas sesuai form	9. Memeriksa data inputan
8. Klik tombol simpan	10. Menyimpan data puskesmas
	11. Menampilkan tabel data puskesmas

Skenario Alternatif

8b Klik tombol batal	9b Menampilkan halaman data puskesmas
8c input data tidak sesuai form	9c memeriksa data inputan
	10 menampilkan pemberitahuan
	11 Menampilkan popup form input data puskesmas

A.1.2 Mengubah Data Puskesmas

Nama usecase	mengubah data puskesmas
Aktor	Admin dan operator
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password Aktor memiliki data puskesmas berupa nama puskesmas, alamat, jenis puskesmas, jumlah desa yang ditangani, jumlah dokter, jumlah bidan, jumlah perawat, jumlah POSKESDES, serta jumlah POSYANDU
Pasca kondisi	User berhasil melakukan edit data puskesmas

Aksi Aktor

Reaksi Sistem

Skenario Normal

1. Klik menu administrasi	2. Menampilkan sub menu data puskesmas dan data desa
3. Klik submenu data puskesmas	4. Memeriksa data puskesmas
	5. Menampilkan data puskesmas
6. Klik ikon edit pada tabel	7. Menampilkan popup form edit data puskesmas
8. Input data puskesmas sesuai form	
9. Klik tombol simpan	10. Memeriksa data inputan
	11. Menyimpan data puskesmas
	12. Menampilkan tabel data puskesmas

Skenario Alternatif

	5b menampilkan pemberitahuan “data tidak ditemukan”
8b input data tidak sesuai form	9 menampilkan peringatan data tidak valid
	10 Menampilkan popup form edit data puskesmas
9b Klik tombol batal	10 Menampilkan halaman data puskesmas

A.1.3 Menghapus Data Puskesmas

Nama usecase	menghapus data puskesmas
Aktor	Admin
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	User memiliki username dan password
Pasca kondisi	User berhasil menghapus data puskesmas

Aksi Aktor

Reaksi Sistem

Skenario Normal

1. Klik menu adminidtrasi	2. Menampilkan sub menu data puskesmas dan data desa
3. Klik submenu data puskesmas	4. Memeriksa data puskesmas
	5. Menampilkan data puskesmas
6. Klik icon delete pada tabel	7. Menampilkan popup konfirmasi hapus data
8. Klik tombol OK	9. Melakukan delete data puskesmas
	10. Menampilkan data puskesmas

Skenario Alternatif

	5b menampilkan pemberitahuan “data tidak ditemukan”
8b klik tombol Cancel	9 menampilkan data puskesmas

A.1.4 Melihat Data Puskesmas

Nama usecase	melihat data puskesmas
Aktor	Admin dan operator
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	User memiliki username dan password
Pasca kondisi	User berhasil menampilkan data puskesmas
Aksi Aktor	
Reaksi Sistem	
Skenario Normal	
1. Klik menu adminidtrasi	2. Menampilkan sub menu data puskesmas dan data desa
3. Klik submenu data puskesmas	4. Memeriksa data puskesmas
	5. Menampilkan data puskesmas
Skenario Alternatif	
	5b menampilkan menampilkan pemberitahuan “data tidak ditemukan”

A.1.5 Menambah Data Desa

Nama usecase	menambah data desa
Aktor	Admin dan operator
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password Aktor memiliki data desa berupa nama desa, alamat, dan nama puskesmas
Pasca kondisi	User berhasil melakukan input data desa

Aksi Aktor

Reaksi Sistem

Skenario Normal

1. Klik menu adminidtrasi	2. Menampilkan sub menu data puskesmas dan data desa
3. Klik submenu data desa	4. Menampilkan halaman data desa
5. Klik tombol tambah data	6. Menampilkan popup form input data desa
7. Input data desa sesuai form	9. Memeriksa data inputan
8. Klik tombol simpan	10. Menyimpan data desa
	11. Menampilkan tabel data desa

Skenario Alternatif

8b Klik tombol batal	9 Menampilkan halaman data desa
7b input data tidak sesuai form	9 menampilkan peringatan "harus diisi"
8 klik tombol simpan	10 menampilkan peringatan "hanya boleh angka dan huruf"

A.1.6 Mengubah Data Desa

Nama usecase	mengubah data desa
Aktor	Admin dan operator
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password Aktor memiliki data desa berupa nama desa, alamat, dan nama puskesmas
Pasca kondisi	User berhasil melakukan edit data desa

Aksi Aktor

Reaksi Sistem

Skenario Normal

1. Klik menu adminidtrasi	2. Menampilkan sub menu data puskesmas dan data desa
3. Klik submenu data desa	4. Memeriksa data desa
	5. Menampilkan data desa
6. Klik icon edit pada tabel	7. Menampilkan popup form edit data desa
8. Input data desa sesuai form	
9. Klik tombol simpan	10. Memeriksa data inputan
	11. Menyimpan data desa
	12. Menampilkan tabel data desa

Skenario Alternatif

	5b menampilkan pemberitahuan “data tidak ditemukan”
9b Klik tombol batal	10 Menampilkan halaman data desa
8b input data tidak sesuai form	
9 klik tombol simpan	10 menampilkan peringatan “harus diisi”
	11 Menampilkan peringatan “hanya boleh angka dan huruf”

A.1.7 Melihat Data Desa

Nama usecase	melihat data desa
Aktor	Admin dan operator
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password
Pasca kondisi	User berhasil menampilkan data desa
Aksi Aktor	
Reaksi Sistem	
Skenario Normal	
1. Klik menu adminidtrasi	2. Menampilkan sub menu data puskesmas dan data desa
3. Klik submenu data desa	4. Memeriksa data desa
	5. Menampilkan data desa
Skenario Alternatif	
	5b menampilkan pemberitahuan “data tidak ditemukan”

A.1.8 Pembobotan Kriteria

Nama usecase	pembobotan kriteria
Aktor	Admin
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password Aktor memiliki data kriteria
Pasca kondisi	User berhasil melakukan pembobotan kriteria
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Klik menu kriteria	2. Menampilkan halaman kriteria
3. Klik tombol pembobotan	4. Menampilkan halaman matrik pembobotan kriteria
5. Mengisi nilai matrik	7. Memroses nilai matrik
6. Klik tombol proses	8. Menampilkan matrik hasil pembobotan dengan tombol proses (ubah, proses, batal, dan tabel)
9. Klik tombol tabel	10. Mengecek data kriteria
	11. Menampilkan tabel data kriteria
Skenario Alternatif	
5 Mengisi nilai matrik	7 memroses nilai matrik
6 klik tombol proses	8 menampilkan peringatan nilai CR tidak memenuhi
	9 Menampilkan matrik hasil pembobotan dengan tombol proses (ubah, proses, batal, dan tabel)

A.1.9 Mengubah Bobot Kriteria

Nama usecase	Mengubah bobot kriteria
Aktor	Admin
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password Aktor memiliki data kriteria
Pasca kondisi	User berhasil mengubah nilai bobot kriteria

Aksi Aktor

Reaksi Sistem

Skenario Normal

1. Klik menu kriteria	2. Melakukan cek data
	3. Menampilkan halaman tabel data kriteria dengan dua tombol proses pembobotan
4. Klik tombol pembobotan	5. Melakukan cek data
	6. Menampilkan matrik hasil pembobotan dengan tombol proses (ubah, proses, batal, dan tabel)
7. Klik tombol ubah	8. Menampilkan matrik ubah nilai pembobotan
9. Mengisi nilai matrik	11. Menghitung matrik pembobotan
10. Klik tombol proses	12. Menampilkan matrik hasil pembobotan dengan tombol proses (ubah, proses, batal, dan tabel)
13. Klik tombol tabel	14. Mengecek data kriteria
	15. Menampilkan tabel data kriteria

Skenario Alternatif

11 klik tombol batal	12 Menampilkan matrik hasil pembobotan dengan tombol proses (ubah, proses, batal, dan tabel)
11 klik tombol proses	14 Menampilkan matrik hasil pembobotan dengan tombol proses

A.1.10 Melihat Data Kriteria

Nama usecase	melihat data kriteria
Aktor	Admin
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password
Pasca kondisi	User berhasil menampilkan data subkriteria

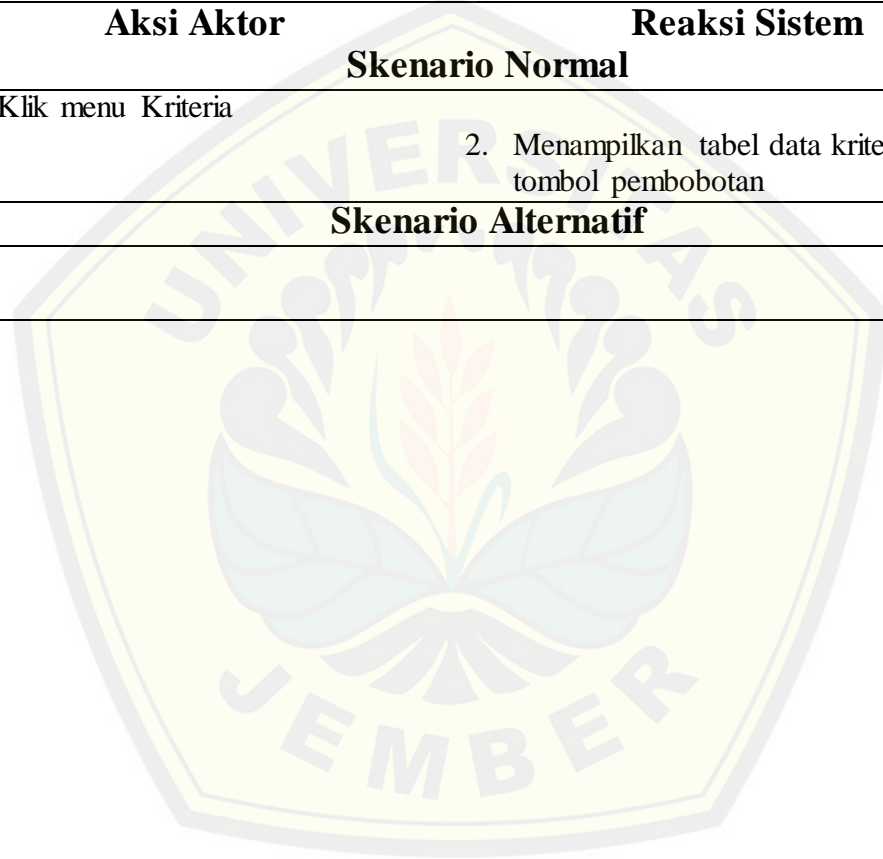
Aksi Aktor

Reaksi Sistem

Skenario Normal

- | | |
|-----------------------|---|
| 1. Klik menu Kriteria | 2. Menampilkan tabel data kriteria dengan tombol pembobotan |
|-----------------------|---|

Skenario Alternatif



A.1.11 Pembobotan Subkriteria

Nama usecase	pembobotan subkriteria
Aktor	Admin
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password Aktor memiliki data subkriteria
Pasca kondisi	User berhasil melakukan pembobotan kriteria

Aksi Aktor

Reaksi Sistem

Skenario Normal

1. Klik menu subkriteria	2. Menampilkan halaman subkriteria
3. Klik tombol subkriteria pada tabel	4. Menampilkan halaman tabel subkriteria dengan tombol proses pembobotan dan tombol kembali
5. Klik tombol pembobotan	6. Menampilkan matrik pembobotan
7. Mengisi nilai matrik	9. Memroses nilai matrik
8. Klik tombol proses	10. Menampilkan matrik hasil pembobotan dengan tombol proses (ubah, proses, batal, dan tabel)
11. Klik tombol tabel	12. Mengecek data kriteria
	13. Menampilkan tabel data subkriteria dengan tombol proses pembobotan dan tombol kembali

Skenario Alternatif

14 Mengisi nilai matrik	16 memroses nilai matrik
15 klik tombol proses	17 menampilkan peringatan nilai CR tidak memenuhi
	18 Menampilkan matrik hasil pembobotan dengan tombol proses (ubah, proses, batal, dan tabel)

A.1.12 Mengubah Bobot Subkriteria

Nama usecase	Mengubah bobot subkriteria
Aktor	Admin
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password Aktor memiliki data subkriteria
Pasca kondisi	User berhasil melakukan pembobotan kriteria

Aksi Aktor

Reaksi Sistem

Skenario Normal

1. Klik menu subkriteria	2. Menampilkan halaman subkriteria
3. Klik tombol subkriteria pada tabel	4. Menampilkan halaman tabel subkriteria dengan tombol proses pembobotan dan tombol kembali
5. Klik tombol pembobotan	6. Menampilkan matrik pembobotan dengan tombol proses (ubah, proses, batal, dan tabel)
7. Klik tombol ubah	8. Menampilkan matrik ubah nilai pembobotan
9. Mengisi nilai matrik	11. Memroses nilai matrik
10. Klik tombol proses	12. Menampilkan matrik hasil pembobotan dengan tombol proses (ubah, proses, batal, dan tabel)
13. Klik tombol tabel	14. Mengecek data kriteria
	15. Menampilkan tabel data subkriteria dengan tombol proses pembobotan

Skenario Alternatif

10 klik tombol proses	11 memroses nilai matrik
	13 Menampilkan matrik hasil pembobotan dengan tombol proses (ubah, proses, batal, dan tabel)

A.1.13 Melihat Data Subkriteria

Nama usecase	melihat data subkriteria
Aktor	Admin
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password
Pasca kondisi	User berhasil menampilkan data subkriteria

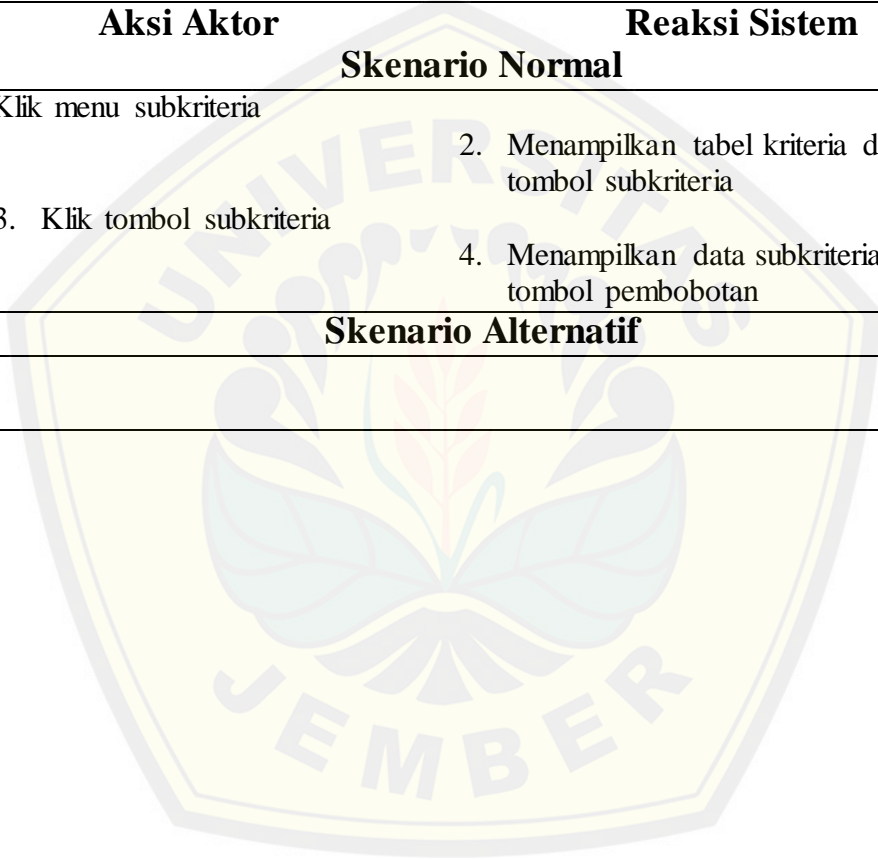
Aksi Aktor

Reaksi Sistem

Skenario Normal

- | | |
|----------------------------|--|
| 1. Klik menu subkriteria | 2. Menampilkan tabel kriteria dengan tombol subkriteria |
| 3. Klik tombol subkriteria | 4. Menampilkan data subkriteria dengan tombol pembobotan |

Skenario Alternatif



A.1.14 Pembobotan Alternatif Desa Siaga

Nama usecase	Pembobotan alternatif desa siaga
Aktor	Admin dan operator
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password Aktor memiliki capaian kriteria dari alternatif desa siaga
Pasca kondisi	User berhasil melakukan pentahapan strata desa siaga

Aksi Aktor

Reaksi Sistem

Skenario Normal

- | | |
|--|--|
| 1. Klik menu kriteria desa siaga | 2. Menampilkan tabel data strata desa siaga |
| 3. Klik tombol tambah data | 4. Menampilkan popup form pentahapan strata |
| 5. Memilih puskesmas | 6. Menampilkan desa wilayah kerja puskesmas yang dipilih |
| 7. Mengisi form sesuai capaian kriteria dari alternatif desa siaga | 9. Cek nilai terhadap batas interval |
| 8. Klik tombol simpan | 10. Menyimpan data hasil pentahapan |
| | 11. Menampilkan tabel data strata desa siaga |

Skenario Alternatif

- | | |
|----------------------|---|
| 8b klik tombol batal | 9b menampilkan tabel data strata desa siaga |
|----------------------|---|

A.1.15 Mengubah Nilai Alternatif Desa Siaga

Nama usecase	Mengubah nilai alternatif desa siaga
Aktor	Admin dan operator
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password Aktor memiliki capaian kriteria dari alternatif desa siaga
Pasca kondisi	User berhasil memperbarui nilai pentahapan strata desa siaga

Aksi Aktor

Reaksi Sistem

Skenario Normal

- | | |
|--|--|
| 1. Klik menu kriteria desa siaga | 2. Menampilkan tabel data strata desa siaga |
| 3. Klik tombol tambah data | 4. Menampilkan popup form pentahapan strata |
| 5. Memilih puskesmas | 6. Menampilkan desa wilayah kerja puskesmas yang dipilih |
| 7. Mengisi form sesuai capaian kriteria dari alternatif desa siaga | 9. Menyimpan data hasil pentahapan |
| 8. Klik tombol simpan | 10. Menampilkan tabel data strata desa siaga |

Skenario Alternatif

- | | |
|----------------------|---|
| 8b klik tombol batal | 9b menampilkan tabel data strata desa siaga |
|----------------------|---|

A.1.16 Melihat Hasil Pembobotan Alternatif

Nama usecase	Melihat hasil pembobotan alternatif
Aktor	Admin dan operator
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password
Pasca kondisi	User berhasil melihat data hasil pentahapan strata desa siaga

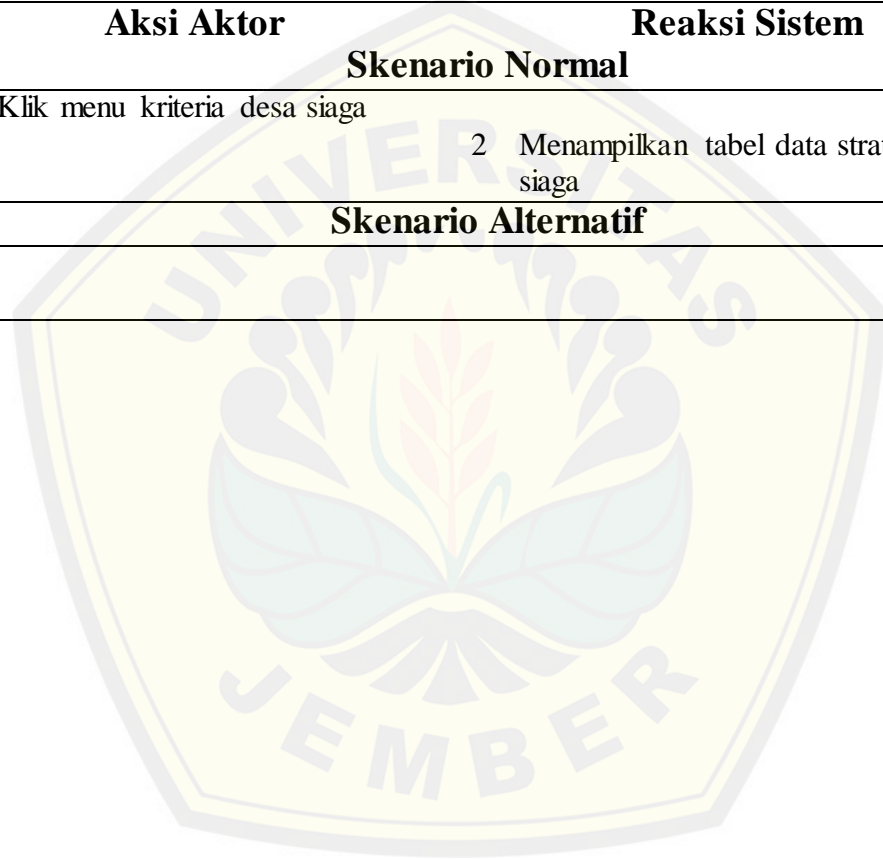
Aksi Aktor

Reaksi Sistem

Skenario Normal

1	Klik menu kriteria desa siaga	2	Menampilkan tabel data strata desa siaga
---	-------------------------------	---	--

Skenario Alternatif



A.2. *Increment* Pemetaan

Nama usecase	melihat peta sebaran strata desa siaga
Aktor	Semua user
Triger	-
Pra kondisi	User membuka alamat url sistem
Entri kondisi	-
Pasca kondisi	User berhasil menampilkan peta sebaran

Aksi Aktor

Reaksi Sistem

Skenario Normal

- | | |
|------------------------------|--|
| 1. Membuka alamat url sistem | 2. Menampilakan halaman frontend dengan menu home, peta sebaran, dan login |
| 3. Klik menu peta sebaran | 4. Mengakses data peta sebaran |
| | 5. Menampilkan peta sebaran |
| 6. Klik pada peta | 7. Menampilkan info window |

Skenario Alternatif

A.3. *Increment* Laporan

A.3.1. Melihat Laporan

Nama usecase	melihat laporan
Aktor	Admin dan operator
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password
Pasca kondisi	User berhasil melihat data laporan

Aksi Aktor

Reaksi Sistem

Skenario Normal

1. Klik menu laporan	2. Memeriksa data laporan
	3. Menampilkan tabel data laporan
4. Cari data sesuai tanggal	5. Menampilkan data laporan sesuai tanggal yang diinginkan

Skenario Alternatif

	5b menampilkan pemberitahuan “data tidak ditemukan”
--	---

A.4. *Increment Manajemen User*

A.4.1. *Menambah Data User*

Nama usecase	Menambah data user
Aktor	Admin
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password Aktor memiliki data user berupa username, password, dan keterangan level user
Pasca kondisi	User berhasil menambahkan user baru

Aksi Aktor

Reaksi Sistem

Skenario Normal

- | | |
|--------------------------------|---|
| 1. Klik menu manajemen user | 2. Menampilkan halaman data user |
| 3. Klik tombol tambah data | 4. Menampilkan popup form input data user |
| 5. Input data user sesuai form | 7. Memeriksa data inputan |
| 6. Klik tombol simpan | 8. Menyimpan data user |
| | 9. Menampilkan tabel data user |

Skenario Alternatif

- | | |
|---------------------------------|--|
| 6b Klik tombol batal | 7 Menampilkan halaman data user |
| 5b input data tidak sesuai form | 7 menampilkan peringatan "harus diisi" |
| 6 klik tombol simpan | 8 Menampilkan poringatan "hanya boleh angka dan huruf" |

A.4.2. Mengubah Data *User*

Nama usecase	mengubah data user
Aktor	Admin
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password Aktor memiliki data user berupa username, password, dan keterangan level user
Pasca kondisi	User berhasil melakukan edit data user

Aksi Aktor

Reaksi Sistem

Skenario Normal

1. Klik menu manajemen user	2. Memeriksa data user
	3. Menampilkan data user
4. Klik ikon edit pada tabel	5. Menampilkan popup form edit data user
6. Input data user sesuai form	8. Memeriksa data inputan
7. Klik tombol simpan	9. Menyimpan data user
	10. Menampilkan tabel data user

Skenario Alternatif

7b Klik tombol batal	8 Menampilkan halaman data user
6b input data tidak sesuai form	
7 klik tombol simpan	8 menampilkan peringatan "harus diisi"
	9 menampilkan peringatan "hanya boleh angka dan huruf"

A.4.3. Menghapus Data User

Nama usecase	menghapus data user	
Aktor	Admin	
Triger	-	
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard	
Entri kondisi	Aktor memiliki username dan password	
Pasca kondisi	User berhasil menghapus data user	
	Aksi Aktor	Reaksi Sistem
	Skenario Normal	
1. Klik menu manajemen user		2. Memeriksa data user
		3. Menampilkan data user
4. Klik ikon delete pada tabel		5. Menampilkan popup konfirmasi hapus data
6. Klik tombol OK		7. Melakukan delete data user
		8. Menampilkan data user
	Skenario Alternatif	
6b klik tombol Cancel		7 menampilkan data user

A.4.4. Melihat Data User

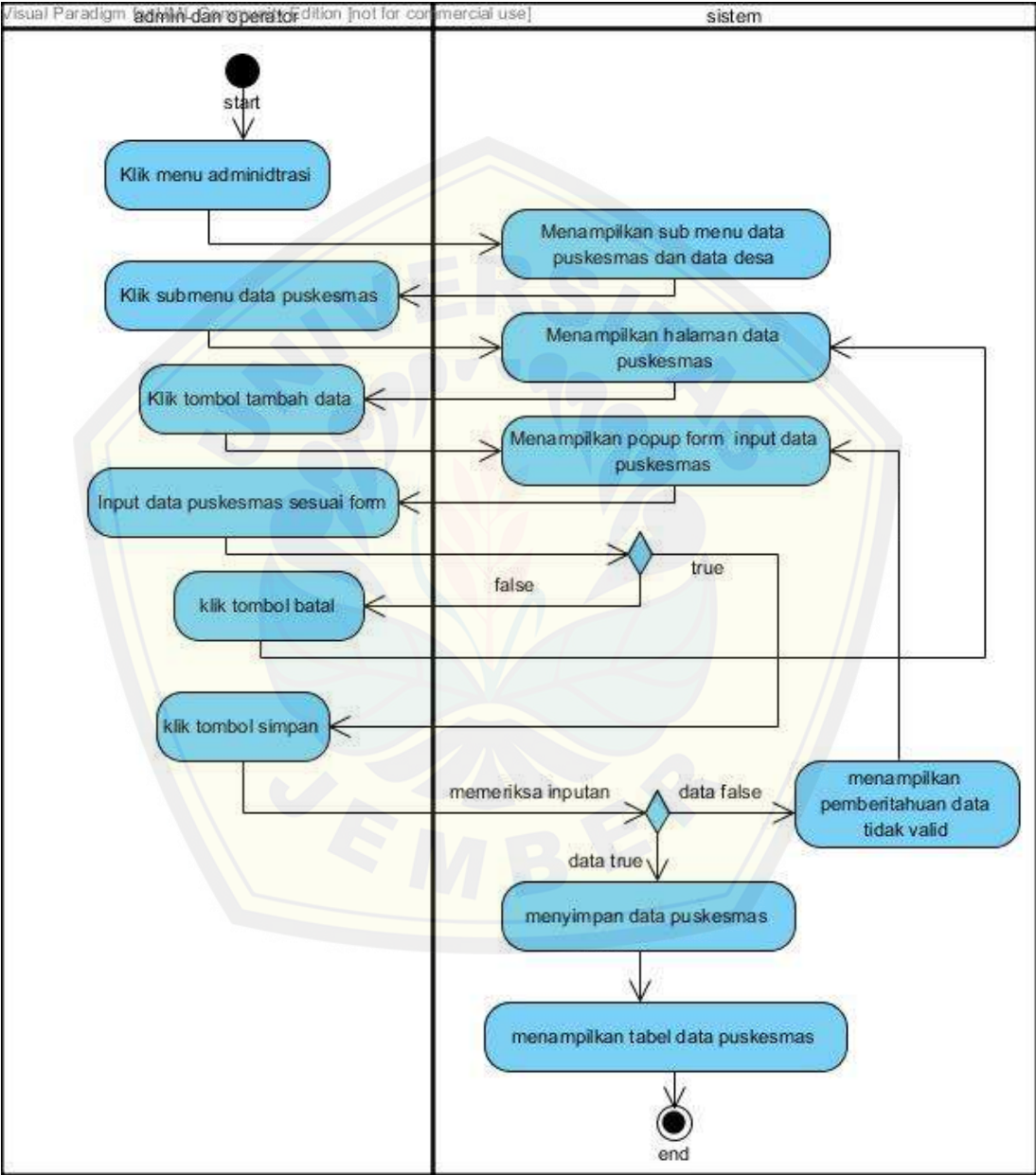
Nama usecase	melihat data user
Aktor	Admin
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password
Pasca kondisi	User berhasil melakukan edit data user
Aksi Aktor	
Reaksi Sistem	
Skenario Normal	
1. Klik menu manajemen user	2. Memeriksa data user 3. Menampilkan data user
Skenario Alternatif	
	3 menampilkan pemberitahuan “data tidak ditemukan”

A.4.5. Melihat History

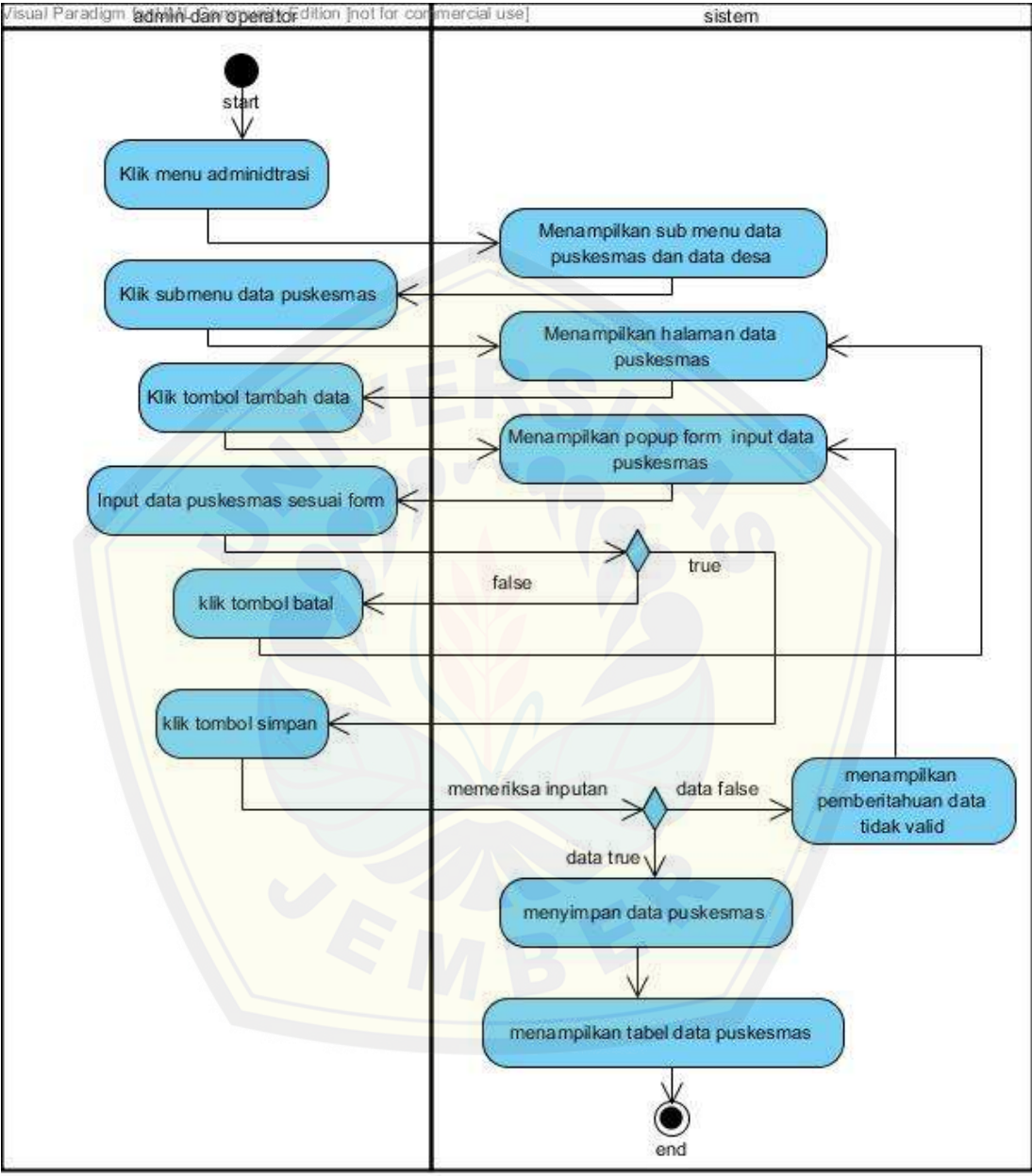
Nama usecase	melihat data history
Aktor	Admin
Triger	-
Pra kondisi	User berhasil melakukan login dan berada pada halaman dashboard
Entri kondisi	Aktor memiliki username dan password
Pasca kondisi	User berhasil menampilkan data history
Aksi Aktor	
Reaksi Sistem	
Skenario Normal	
1. Klik menu hostory	2. Memeriksa data history 3. Menampilkan data history
Skenario Alternatif	
	3 menampilkan pemberitahuan “data tidak ditemukan”

LAMPIRAN B Activity Diagram

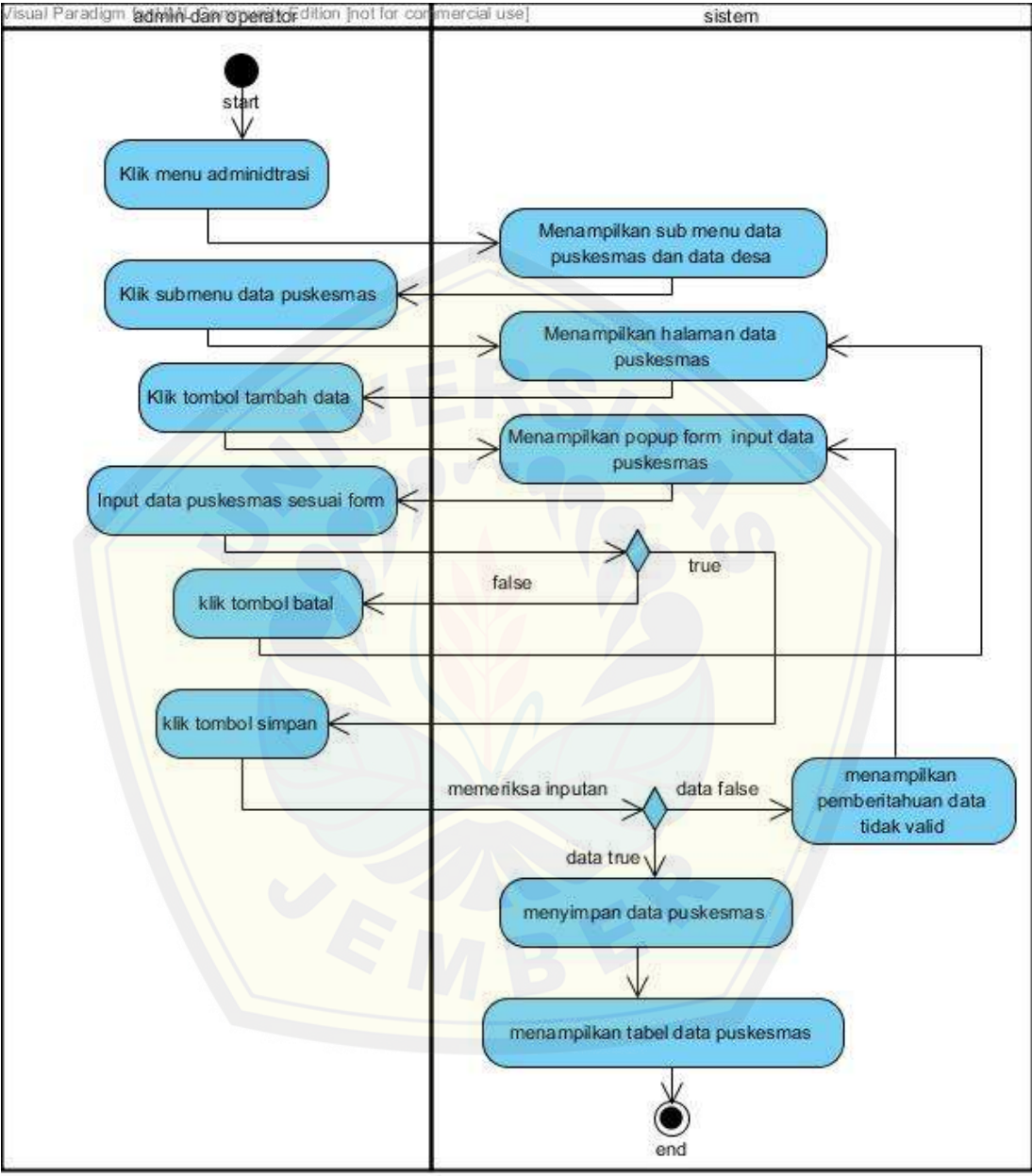
B.1. Increment Proses Penentuan Strata Desa Siaga
B.1.1. Menambah data puskesmas



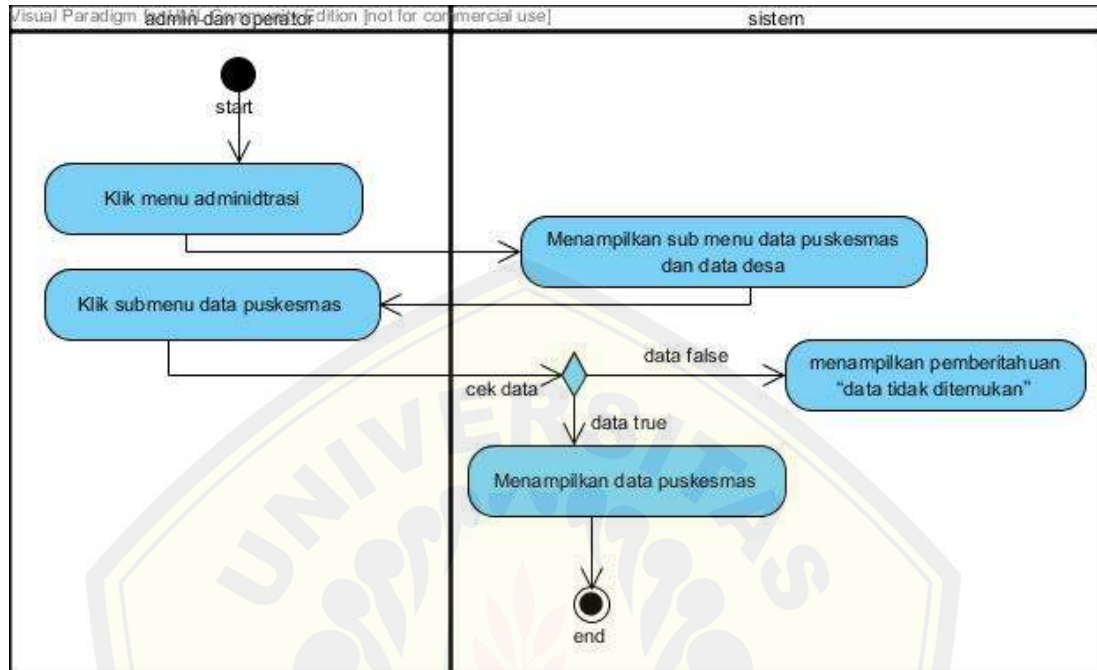
B.1.2. Mengubah data puskesmas



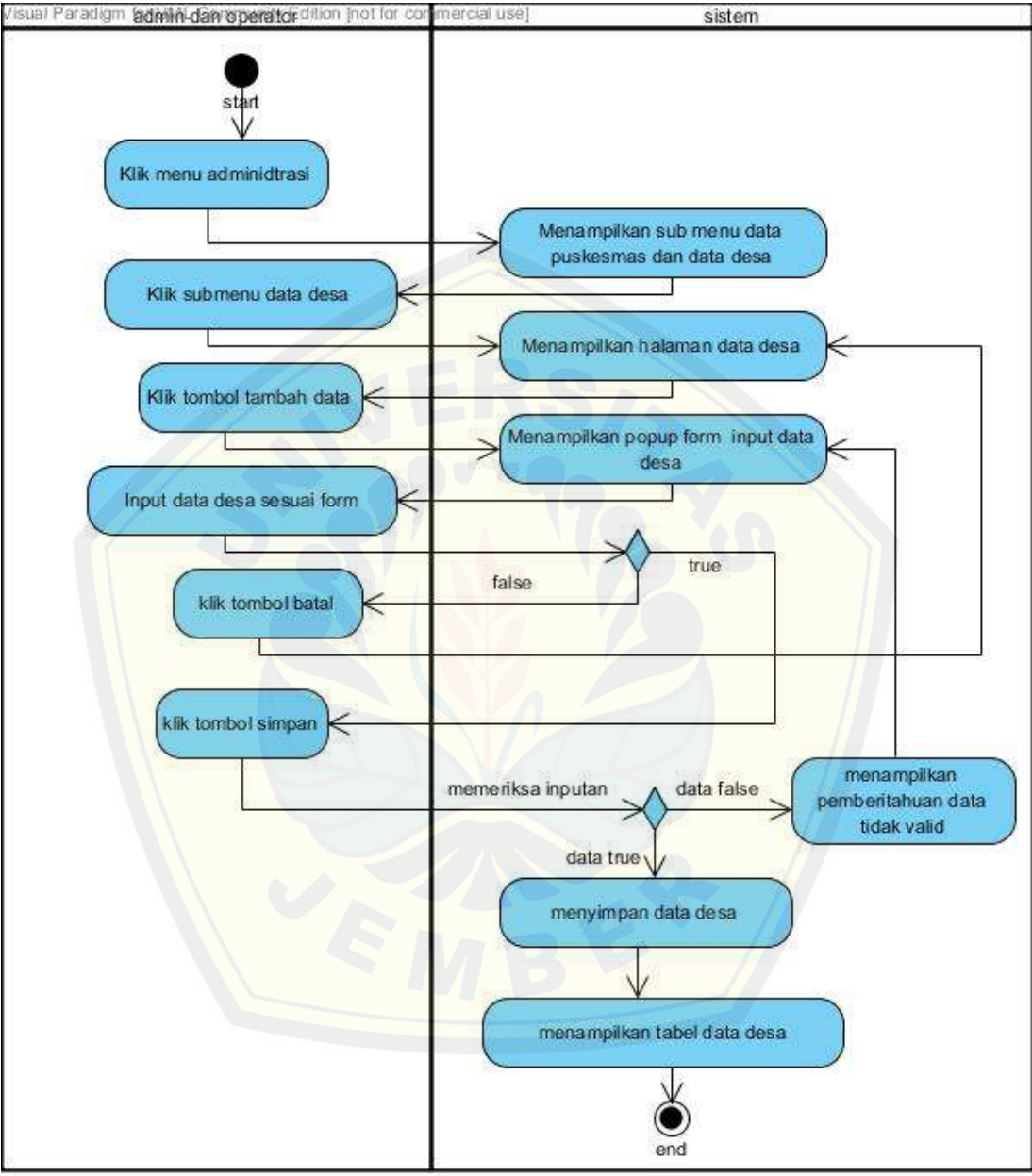
B.1.3. Menghapus data puskesmas



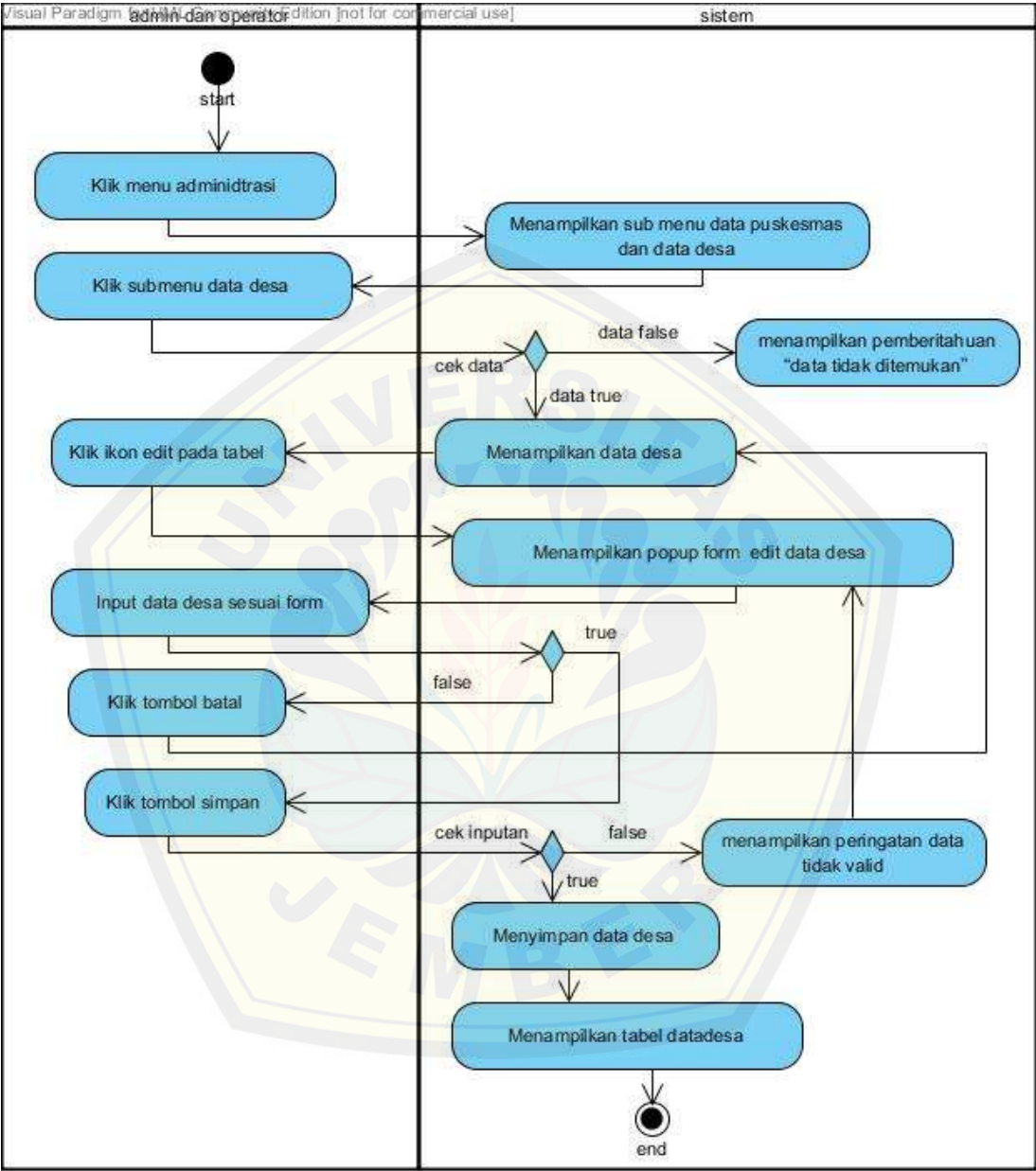
B.1.4. Melihat data puskesmas



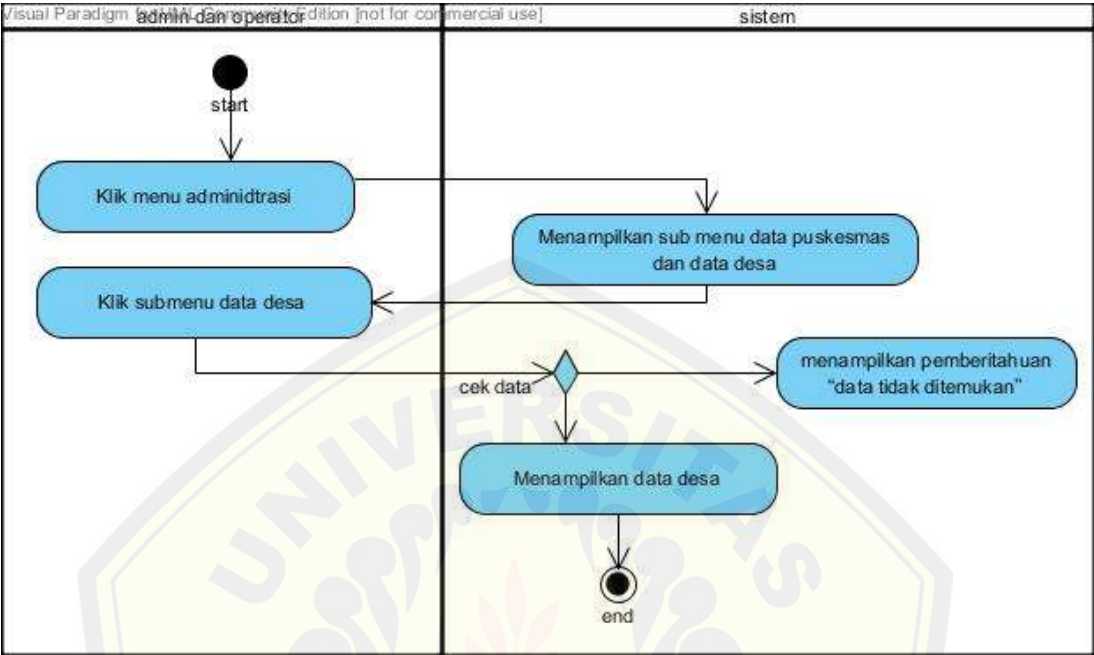
B.1.5. Menambah data desa



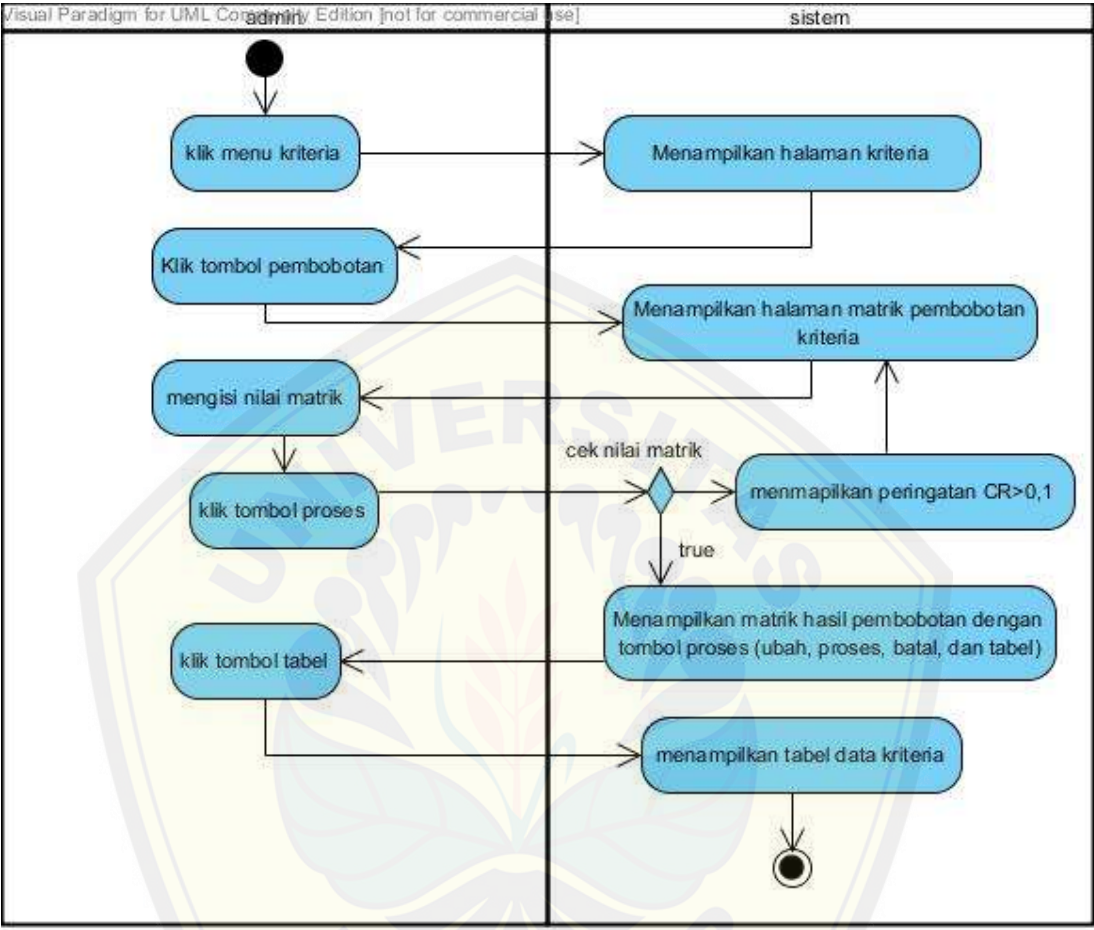
B.1.6. Mengubah data desa



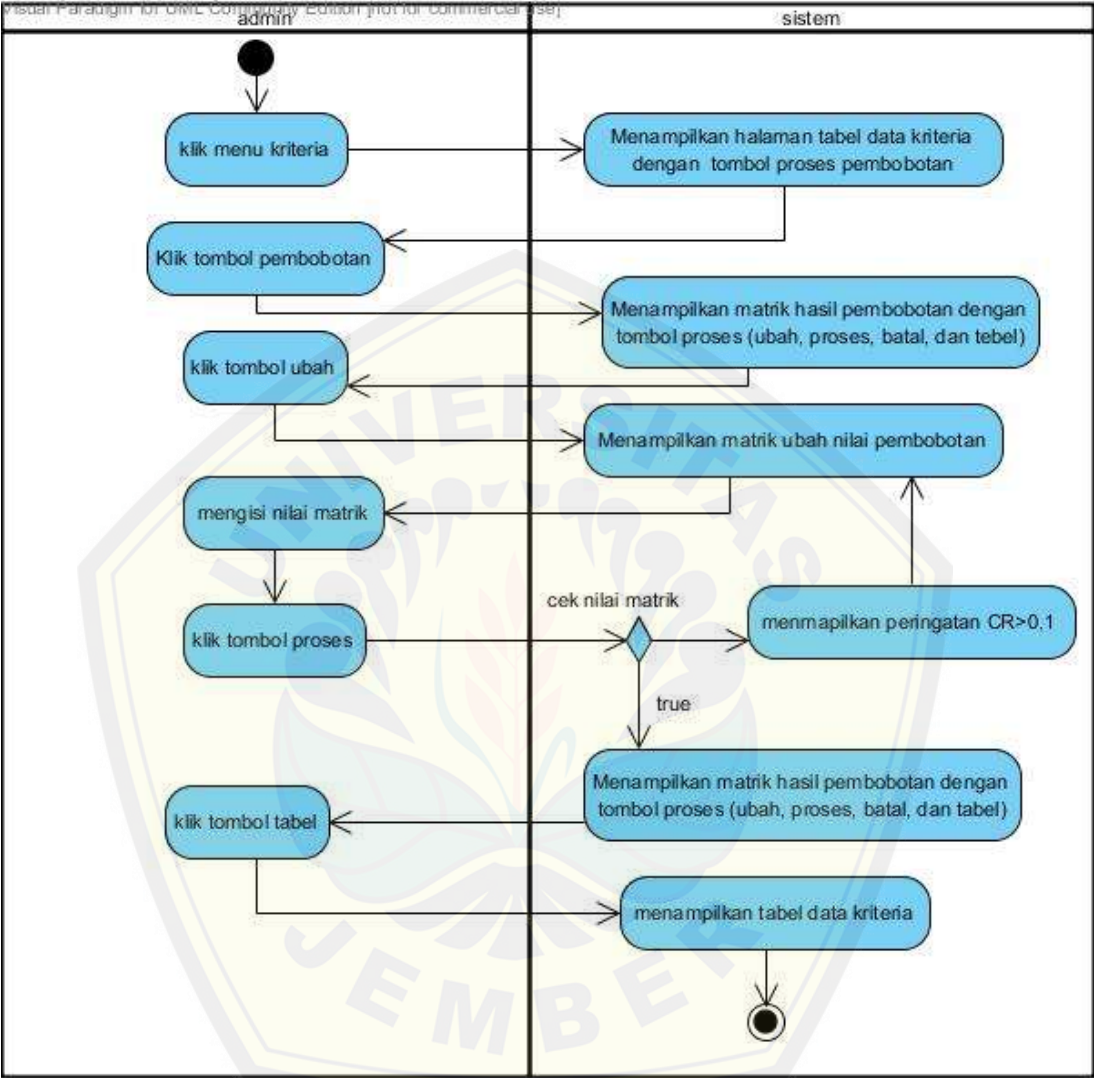
B.1.7. Melihat data desa



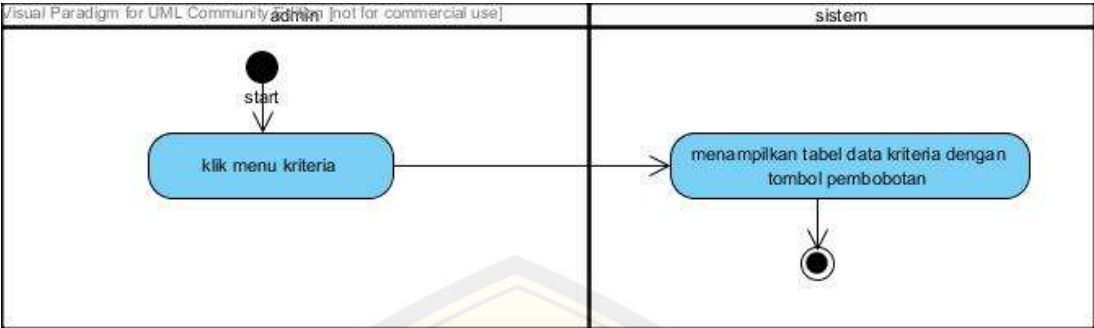
B.1.8. Pembobotan Kriteria



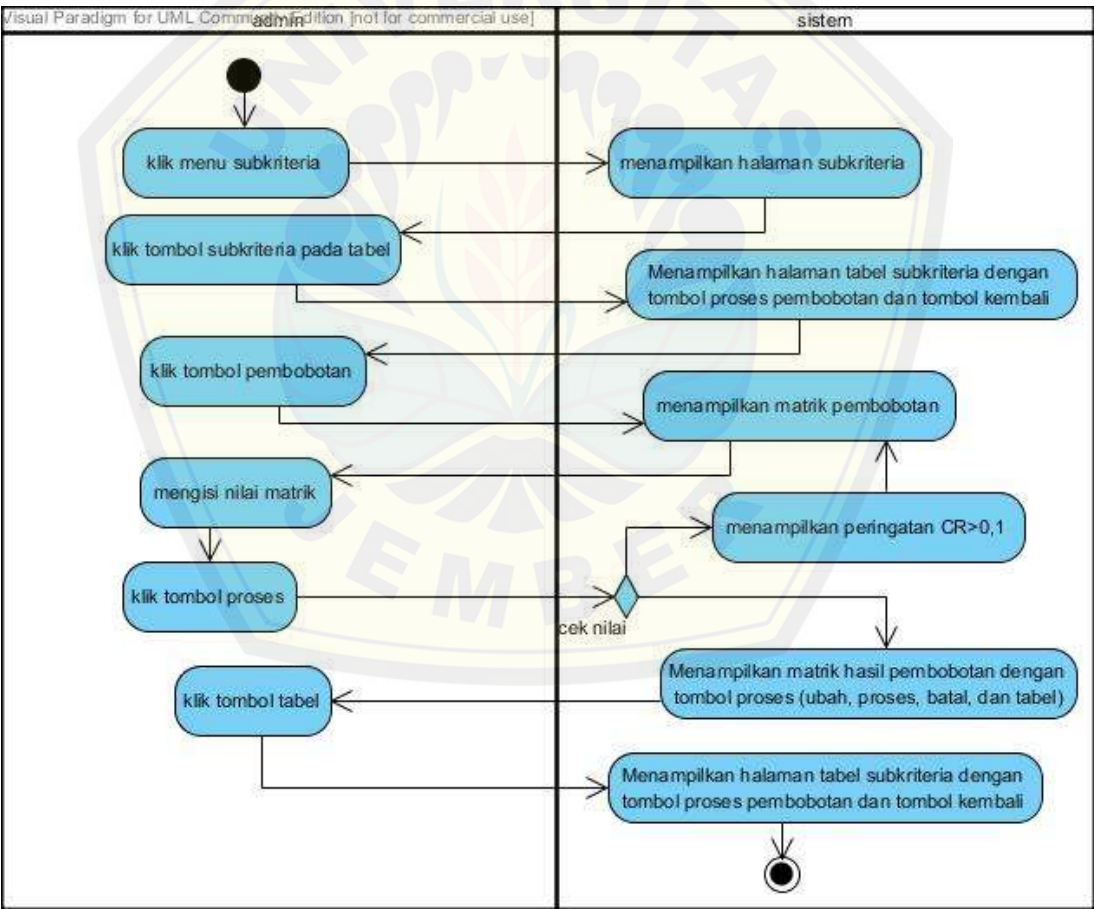
B.1.9. Mengubah bobot kriteria



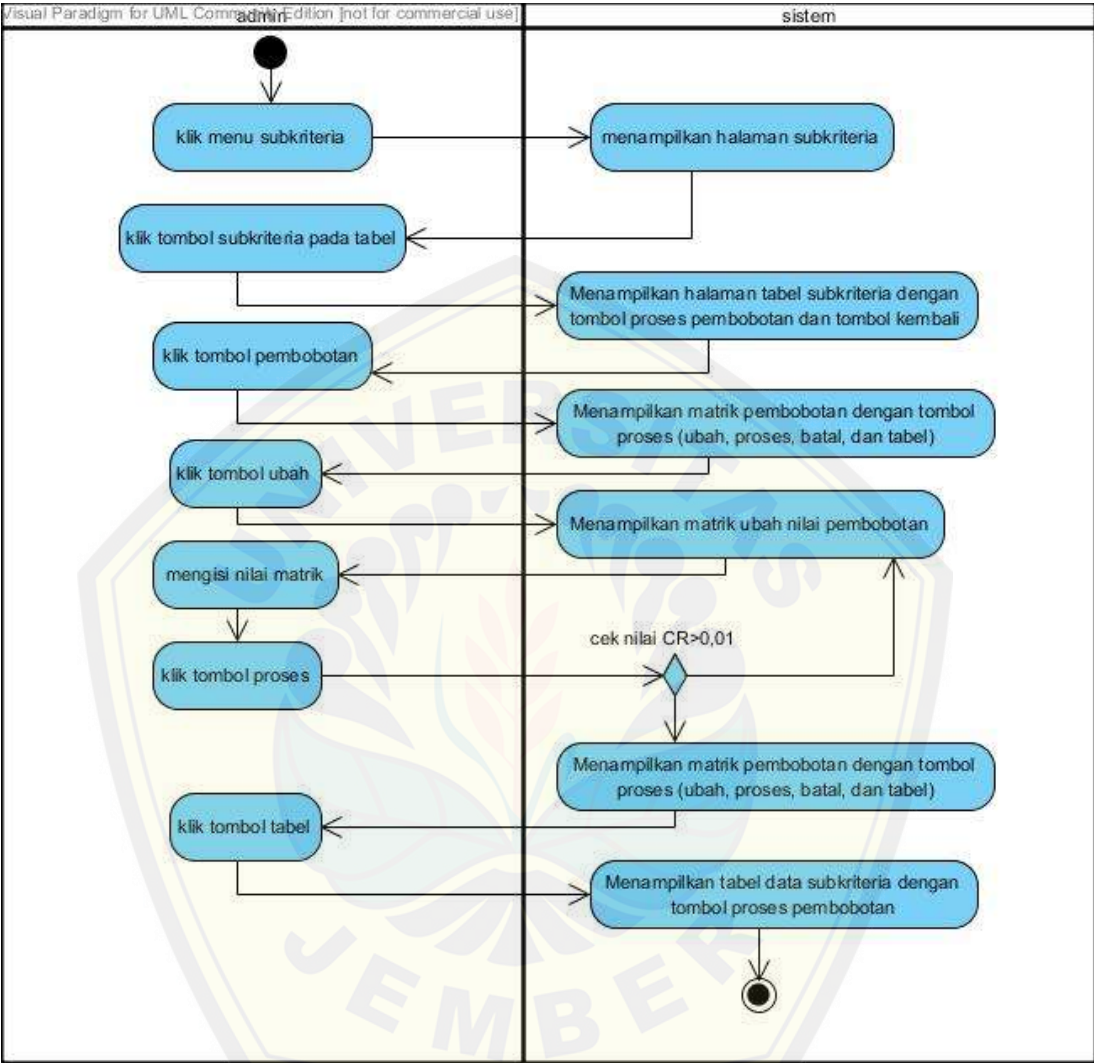
B.1.10. Melihat data kriteria



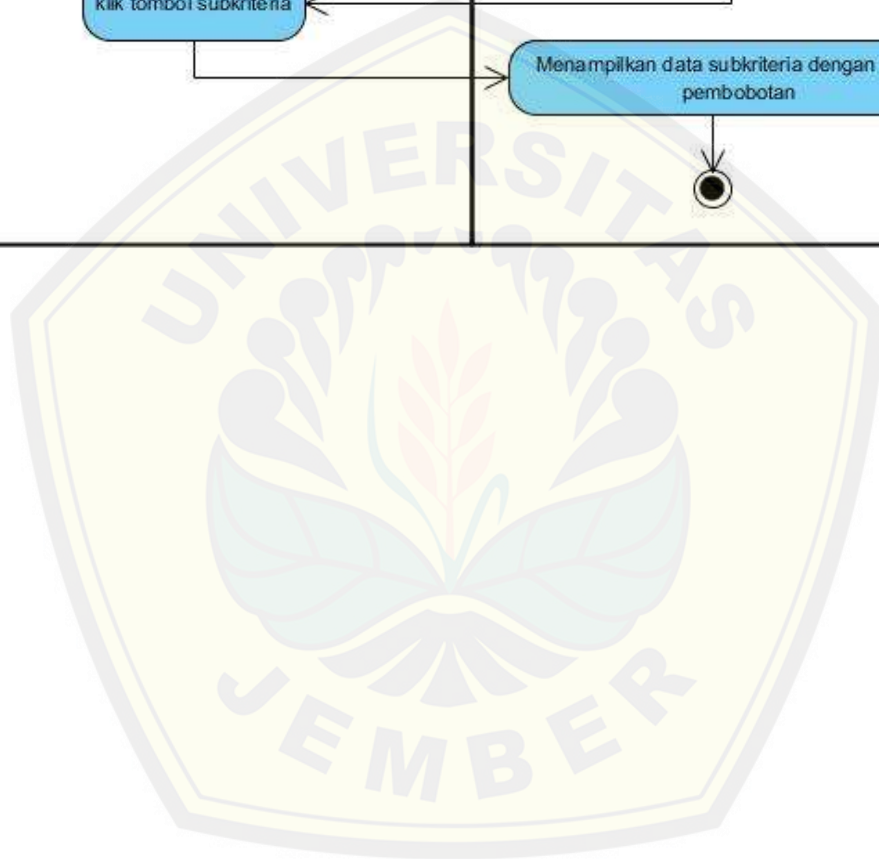
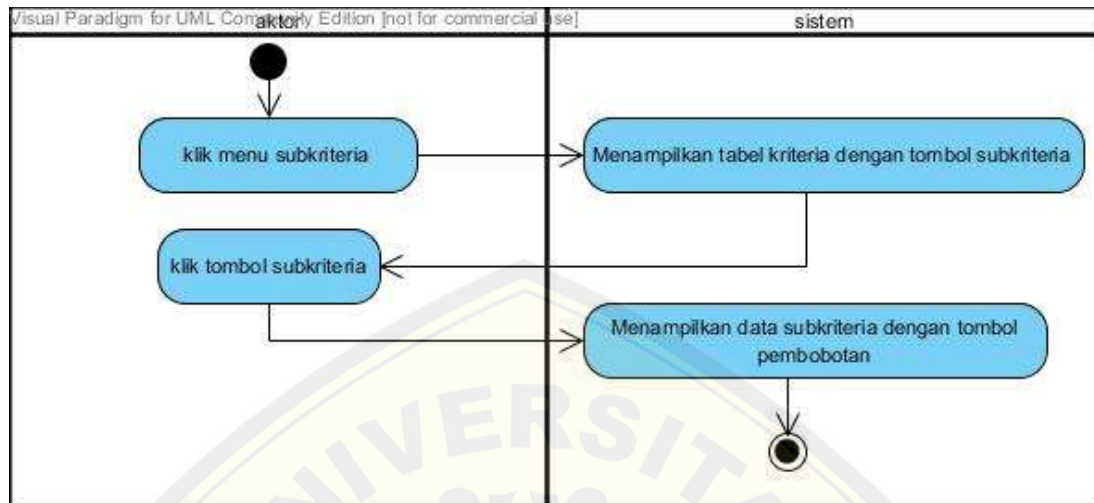
B.1.11. Pembobotan subkriteria



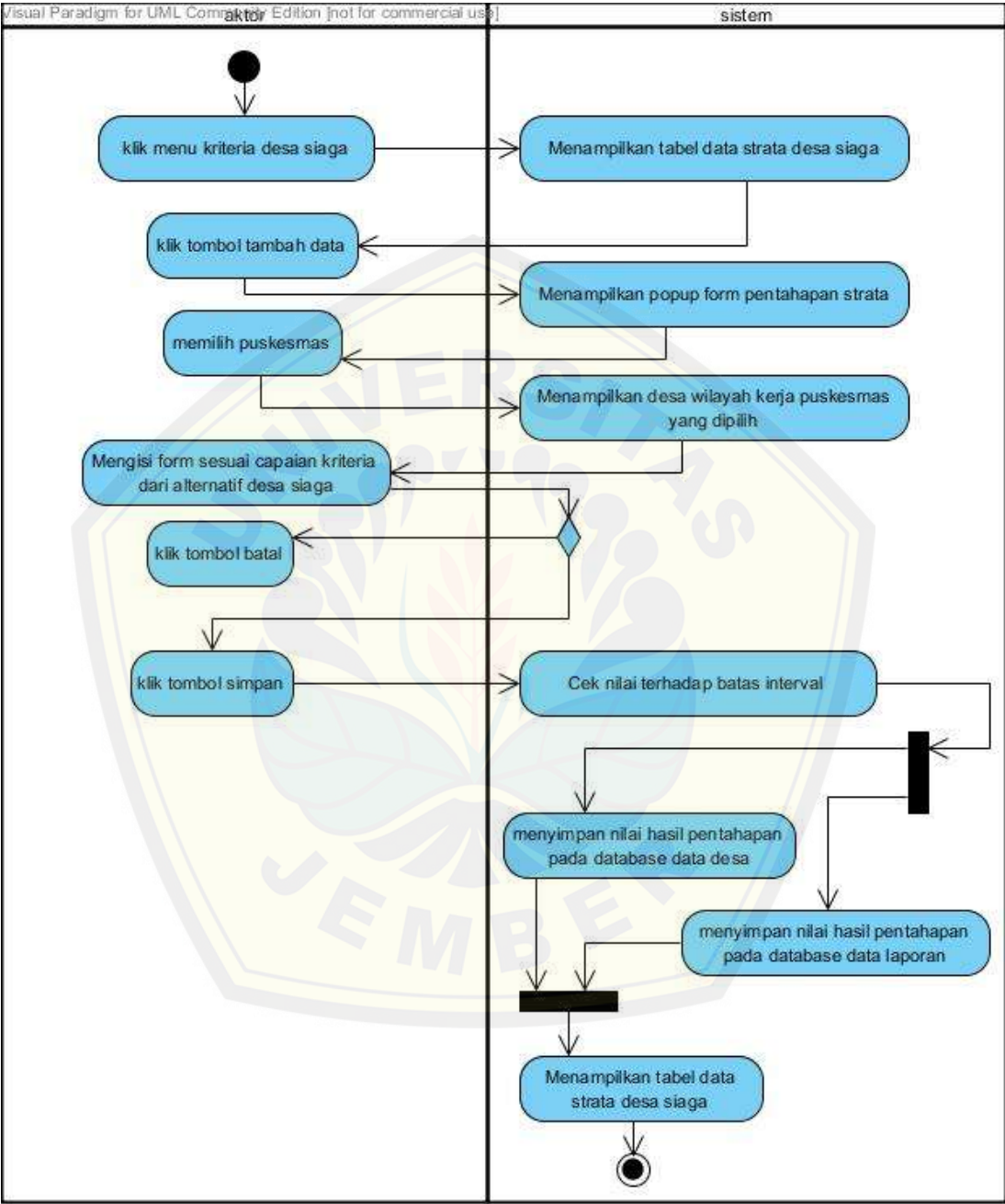
B.1.12. Mengubah bobot subkriteria



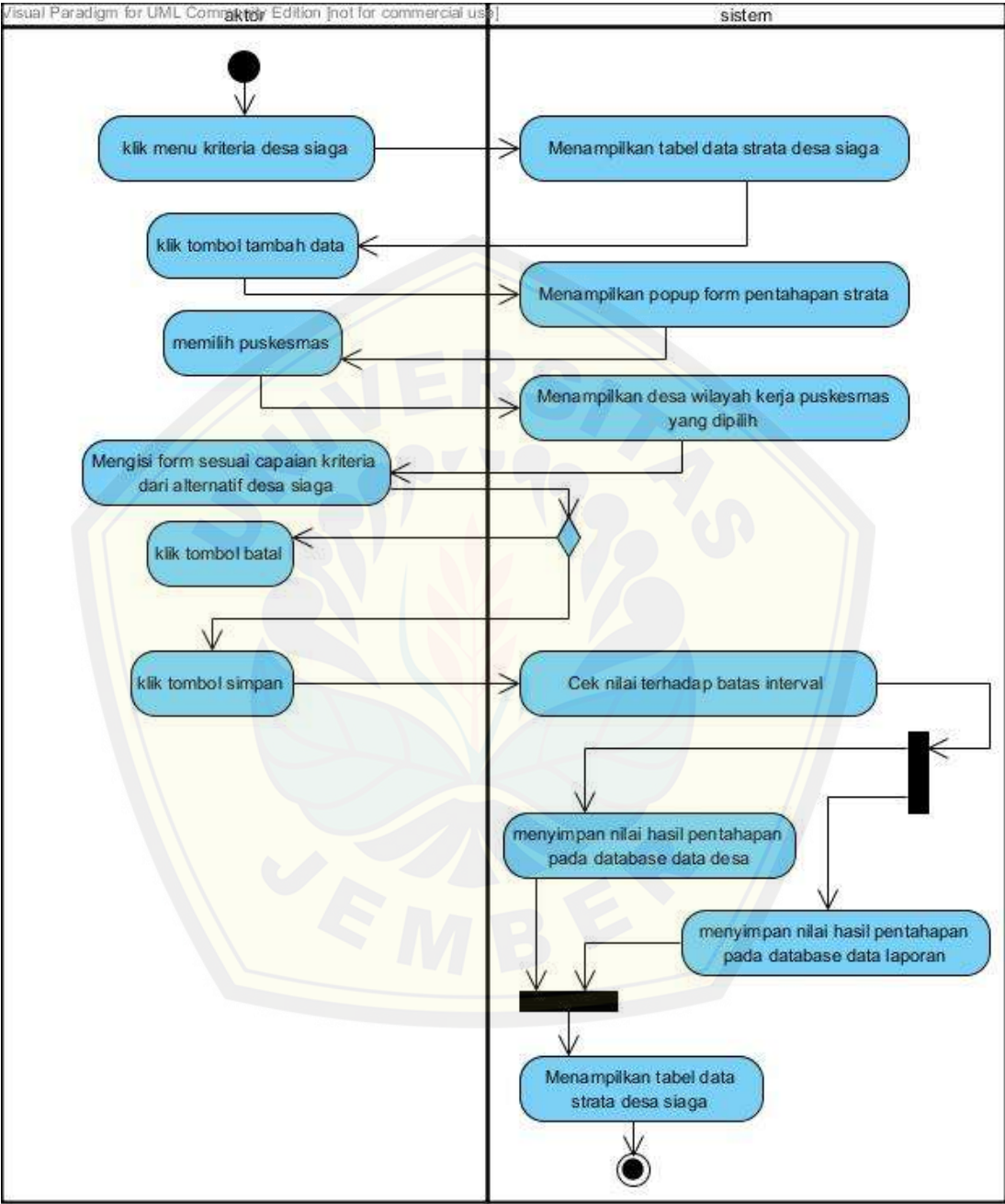
B.1.13. Melihat data subkriteria



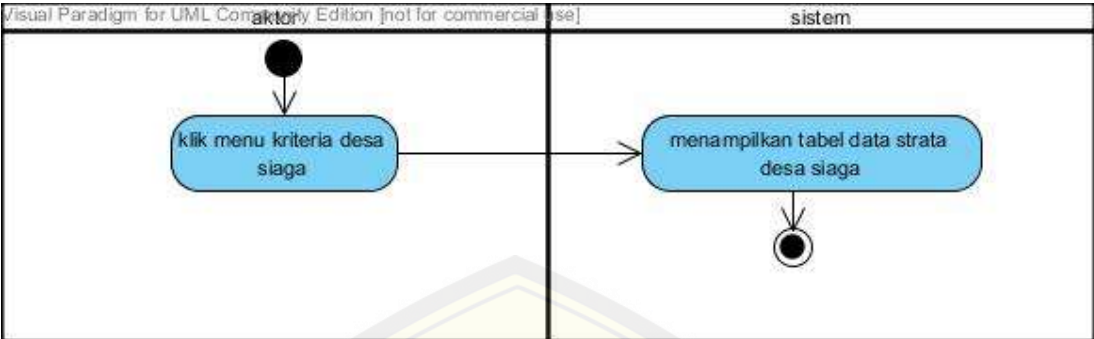
B.1.14. Pembobotan alternatif desa siaga



B.1.15. Mengubah nilai alternatif desa siaga

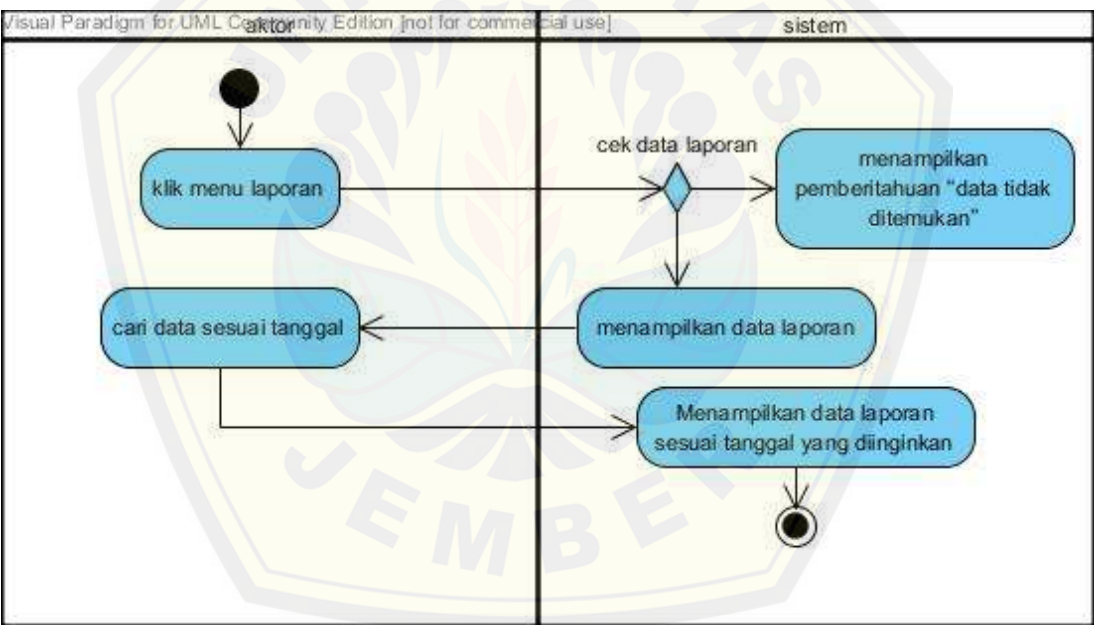


B.1.16. Melihat hasil pembobotan alternative

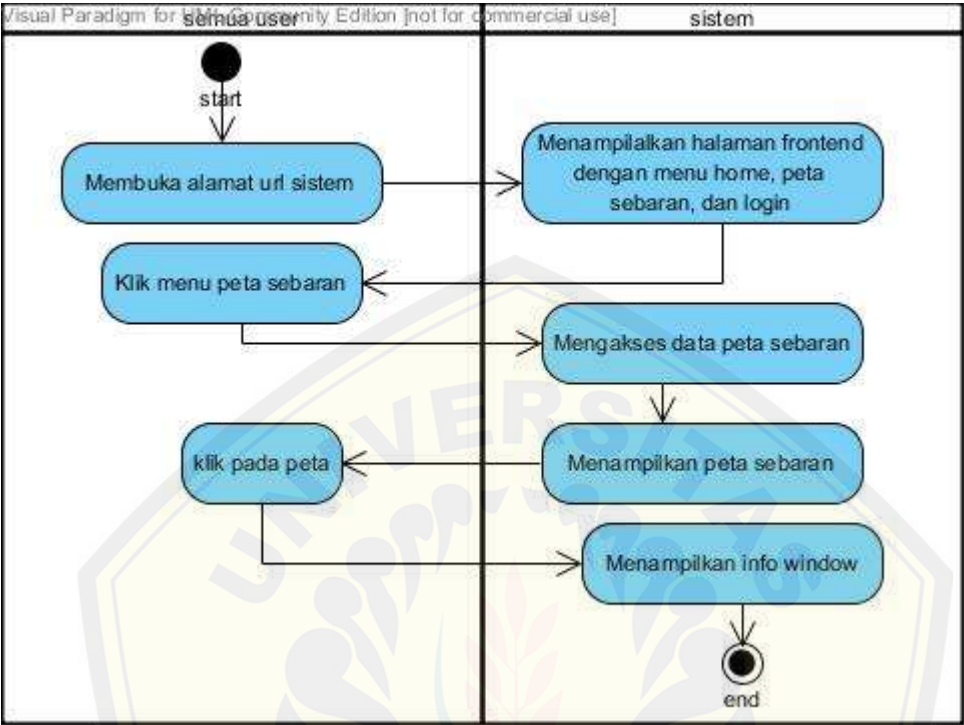


B.2. Increment Laporan

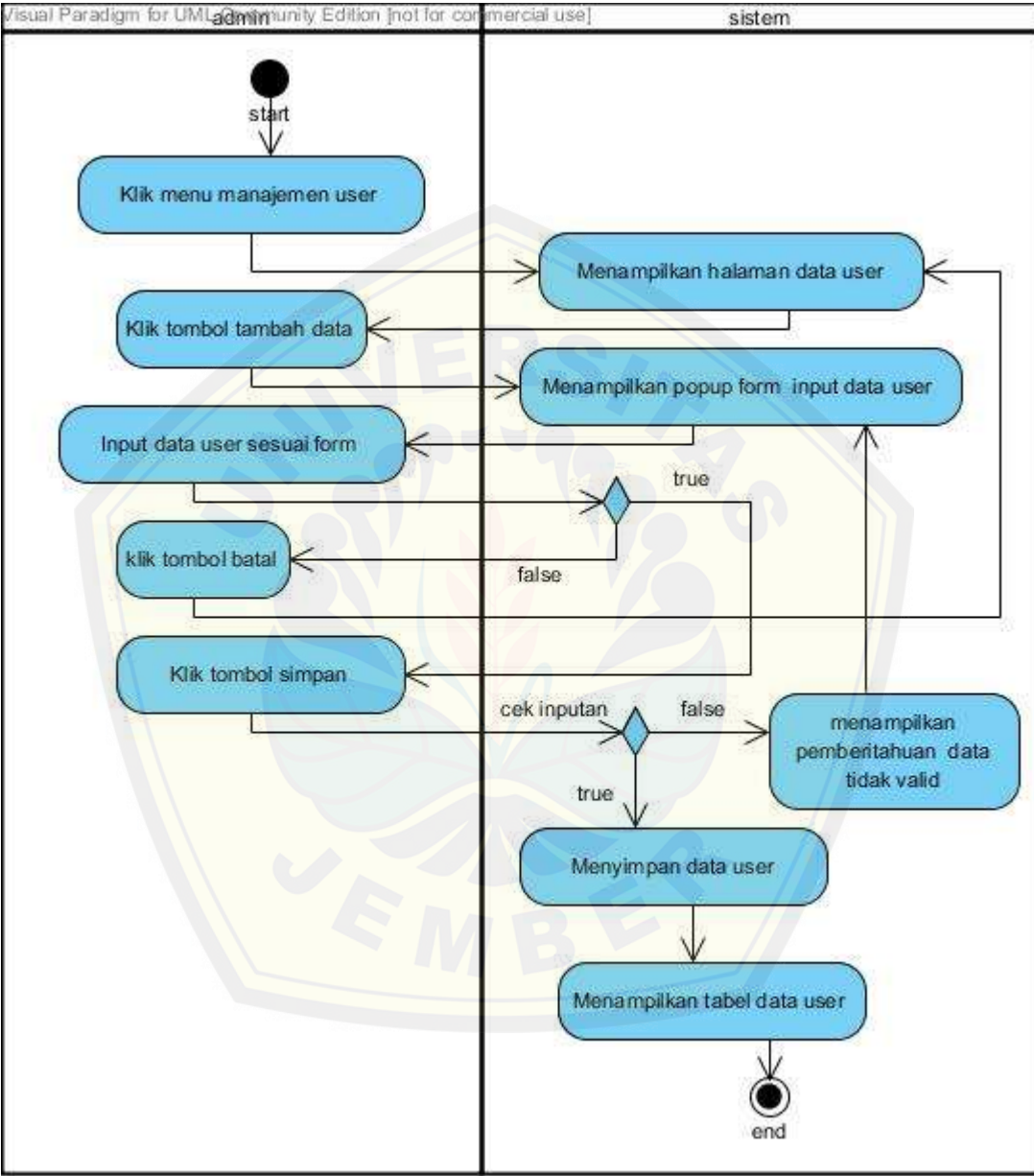
B.2.1. Melihat laporan



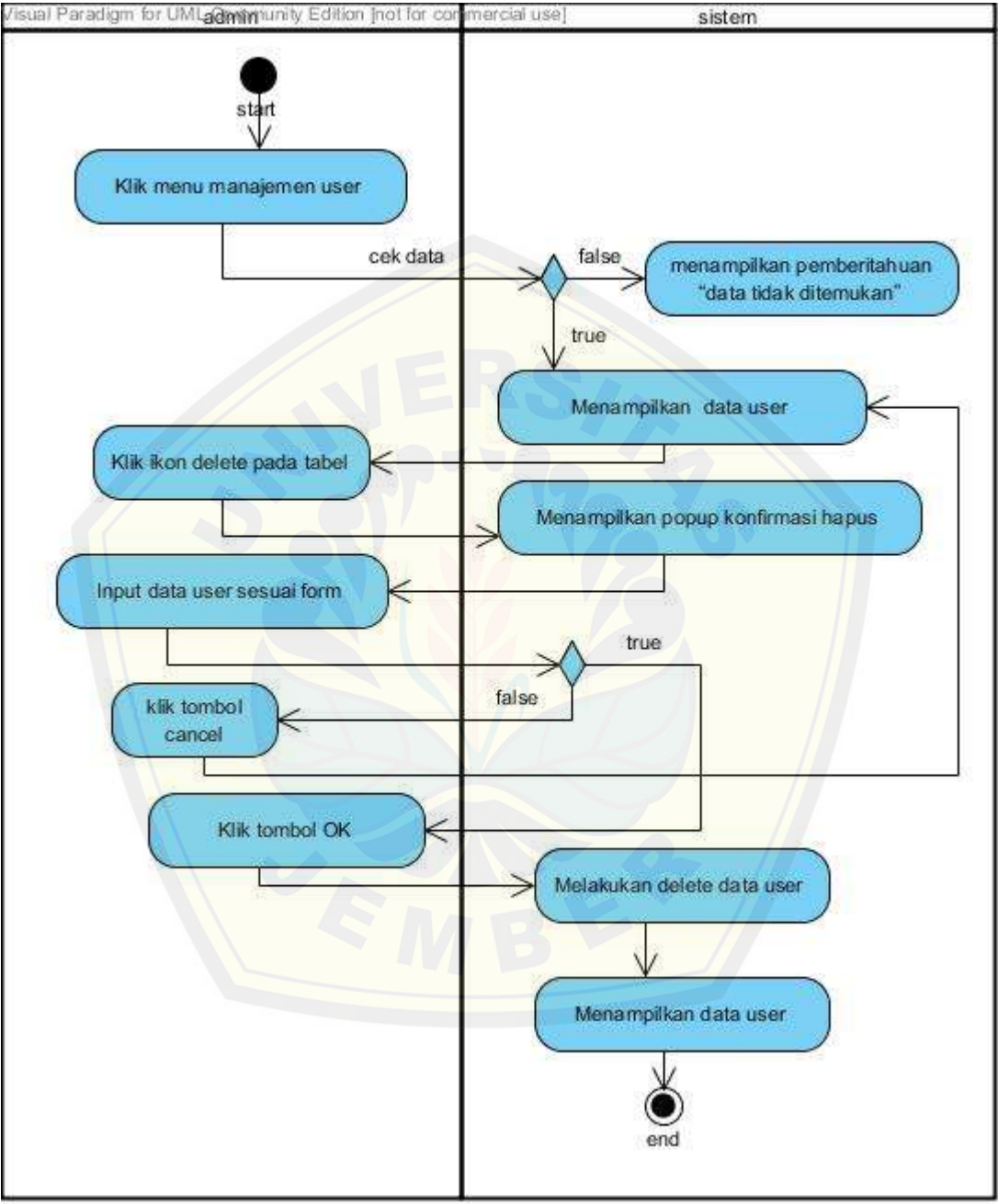
B.3. Increment Pemetaan



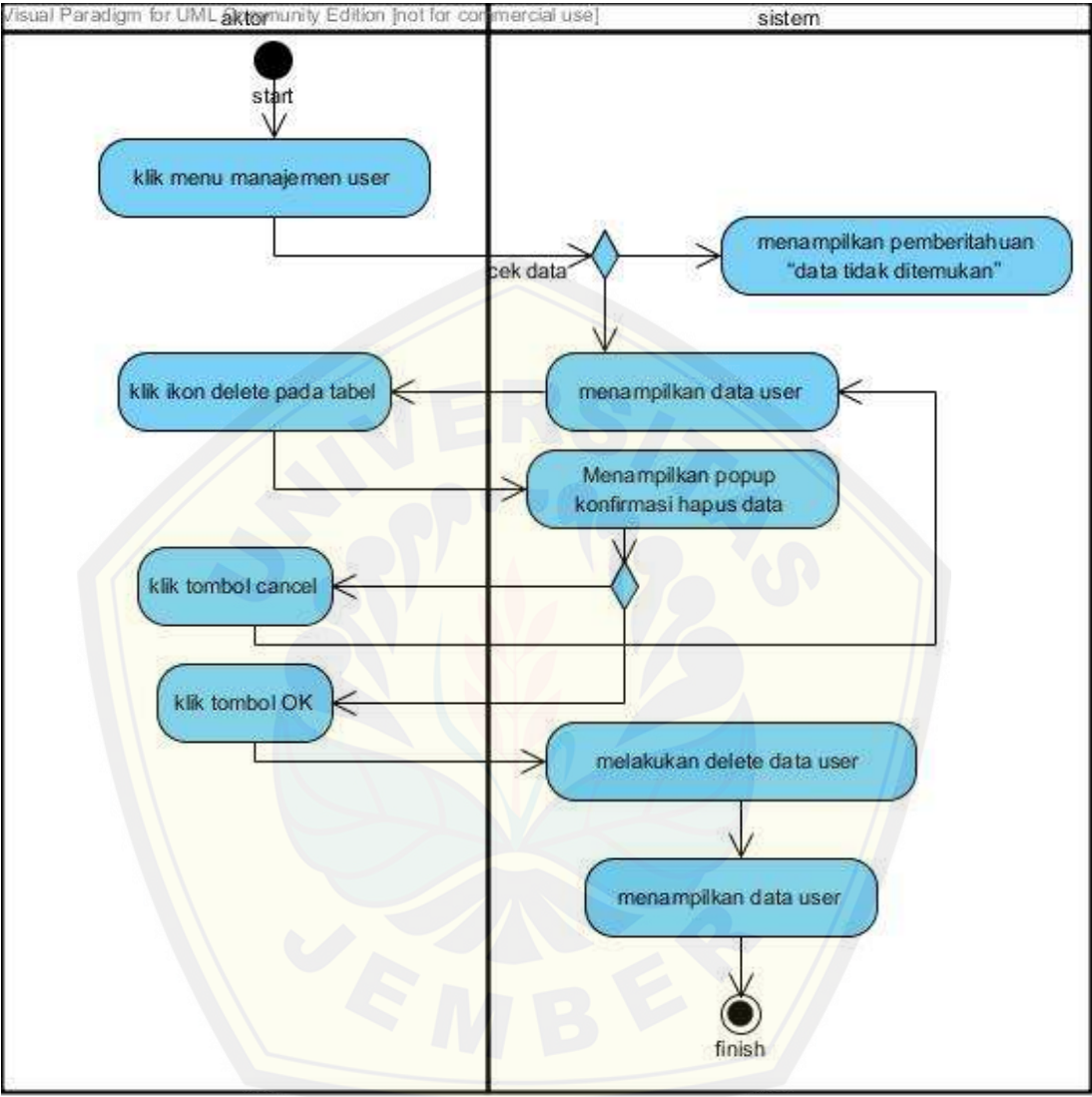
B.4. Increment Manajemen User
B.4.1. Menambah data user



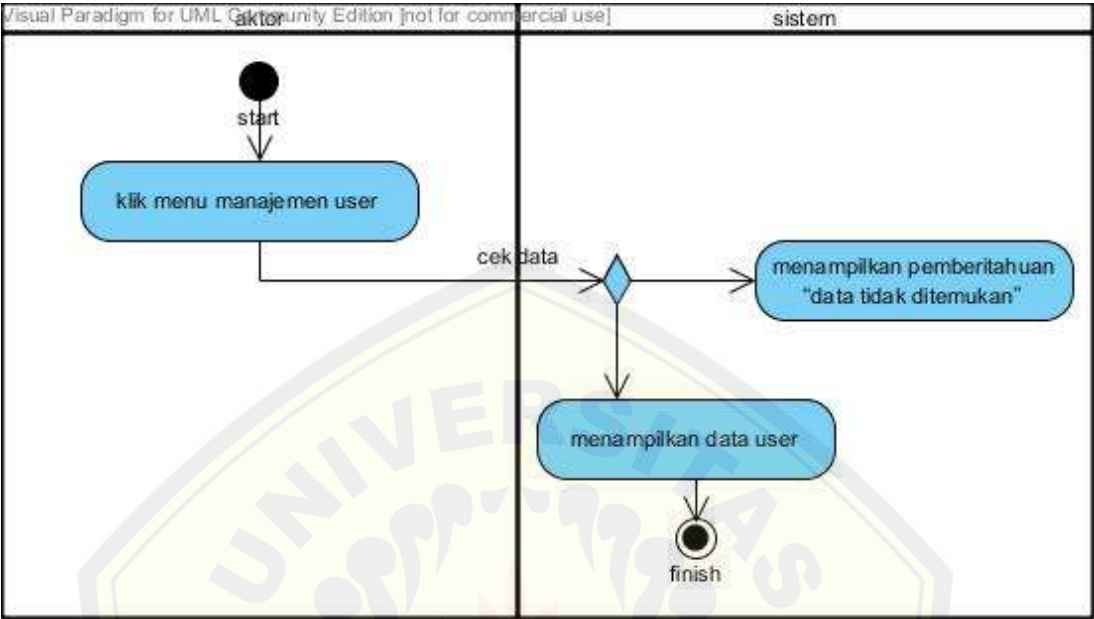
B.4.2. Mengubah data user



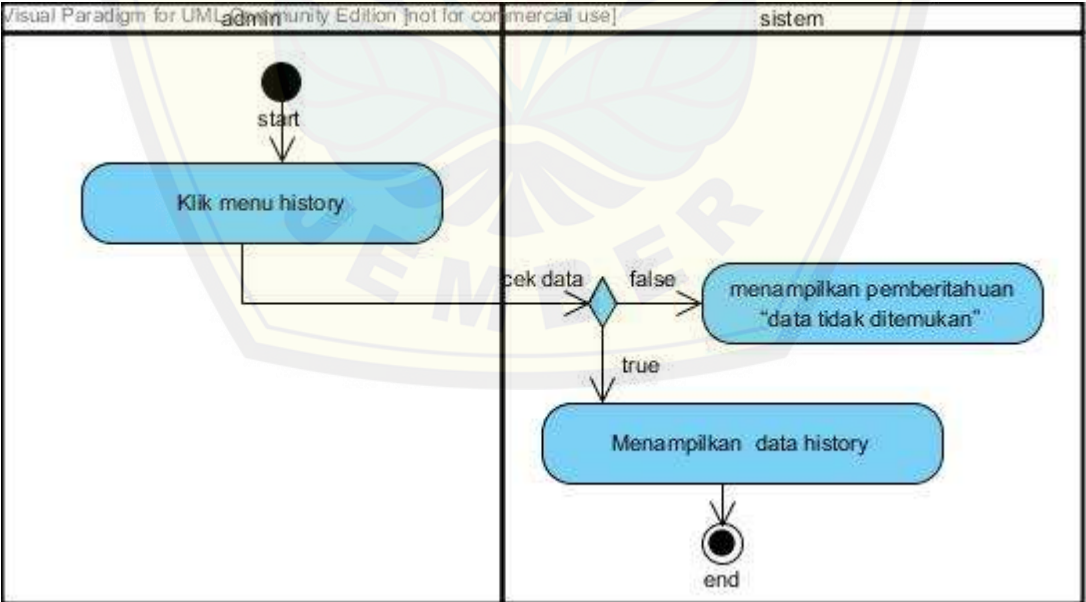
B.4.3. Menghapus data user



B.4.4. Melihat data user

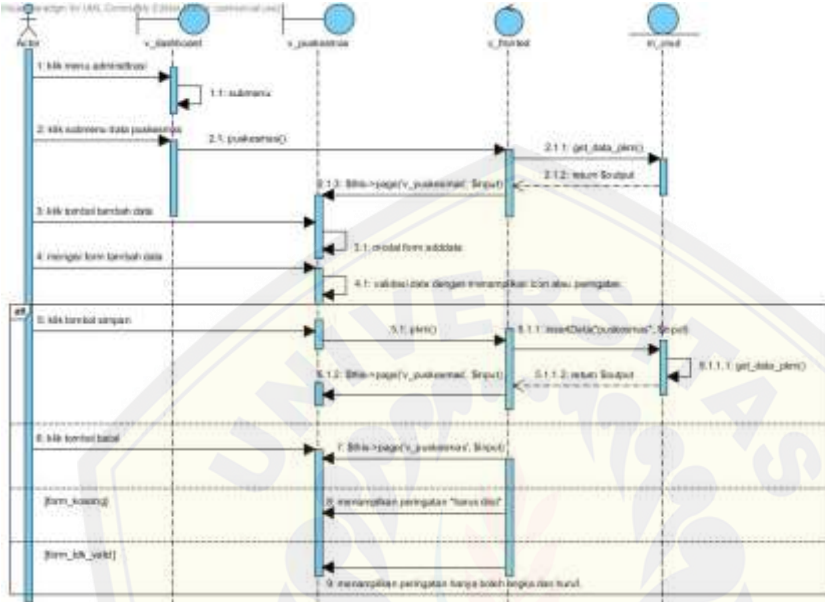


B.4.5. Melihat history

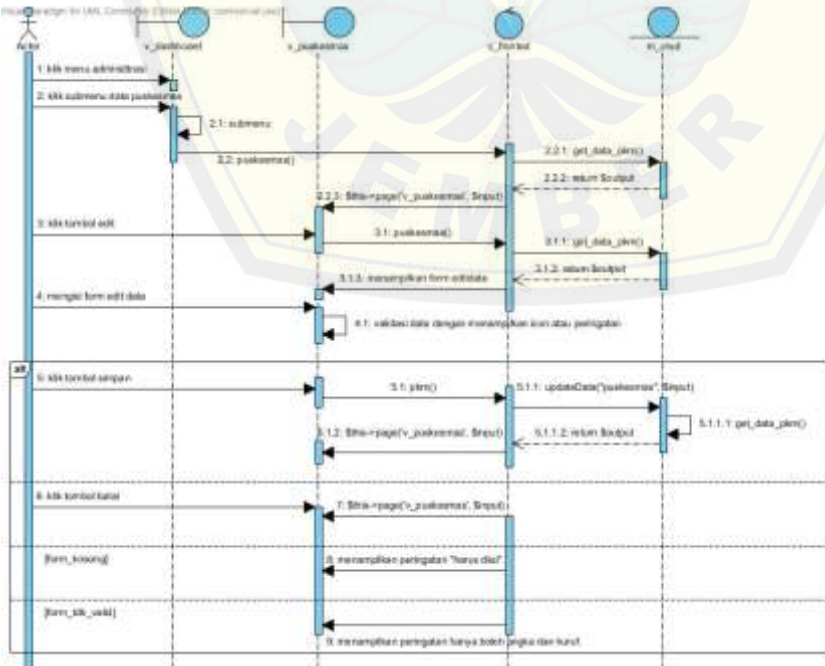


LAMPIRAN C Sequence Diagram

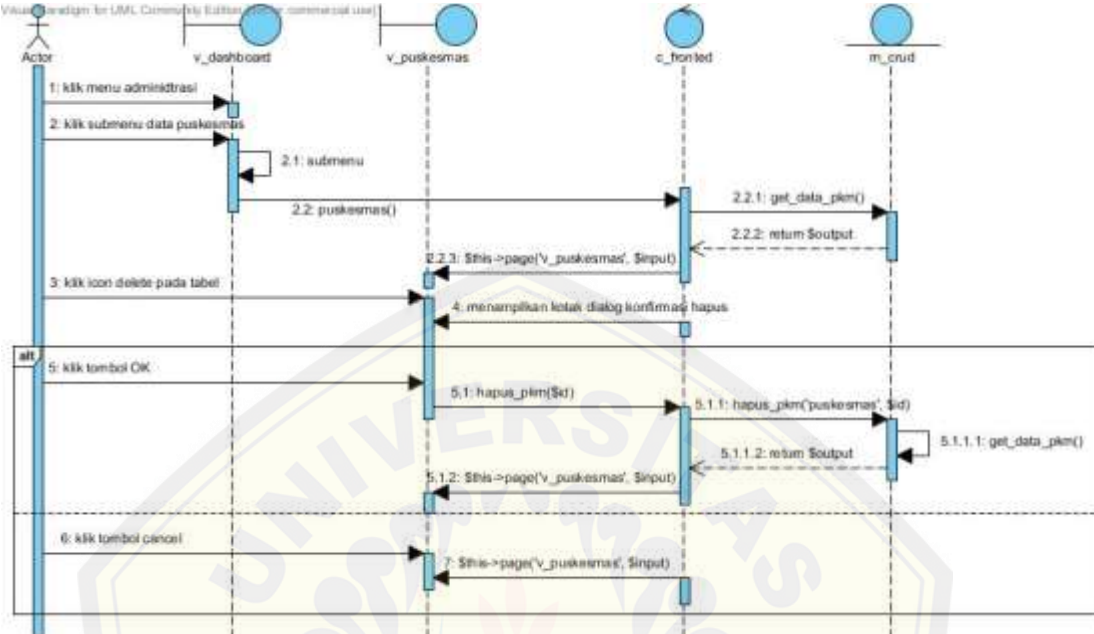
C.1. Increment Proses Penentuan Strata Desa Siaga
C.1.1. Menambah data puskesmas



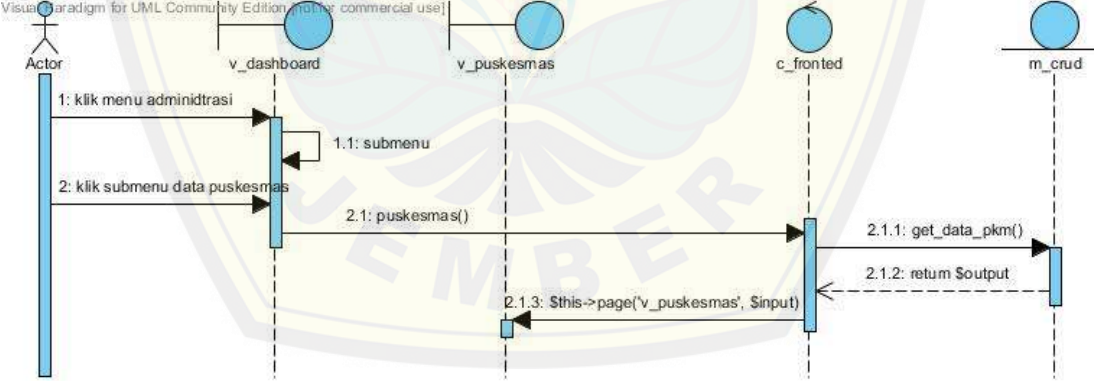
C.1.2. Mengubah data puskesmas



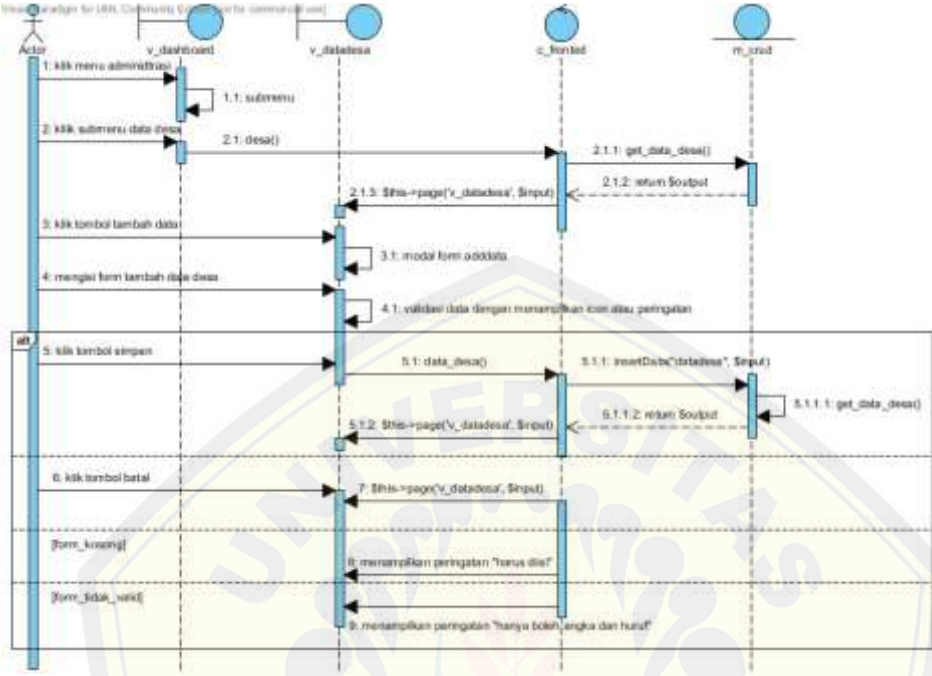
C.1.3. Menghapus data puskesmas



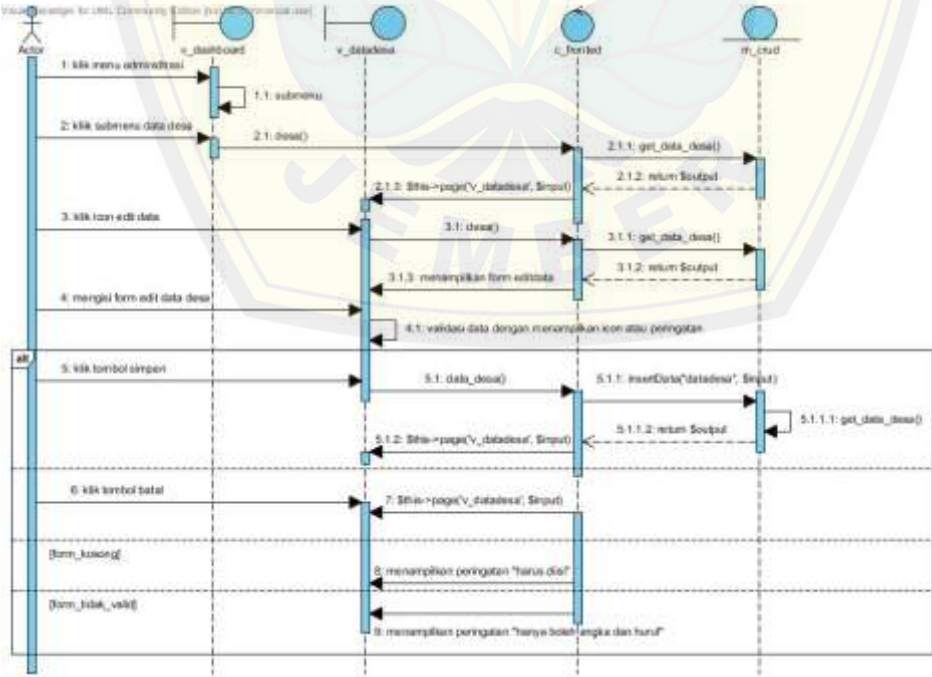
C.1.4. Melihat data puskesmas



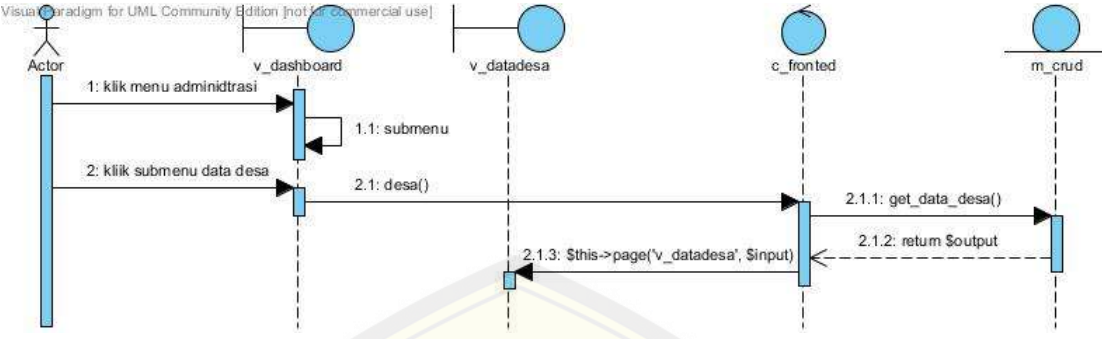
C.1.5. Menambah data desa



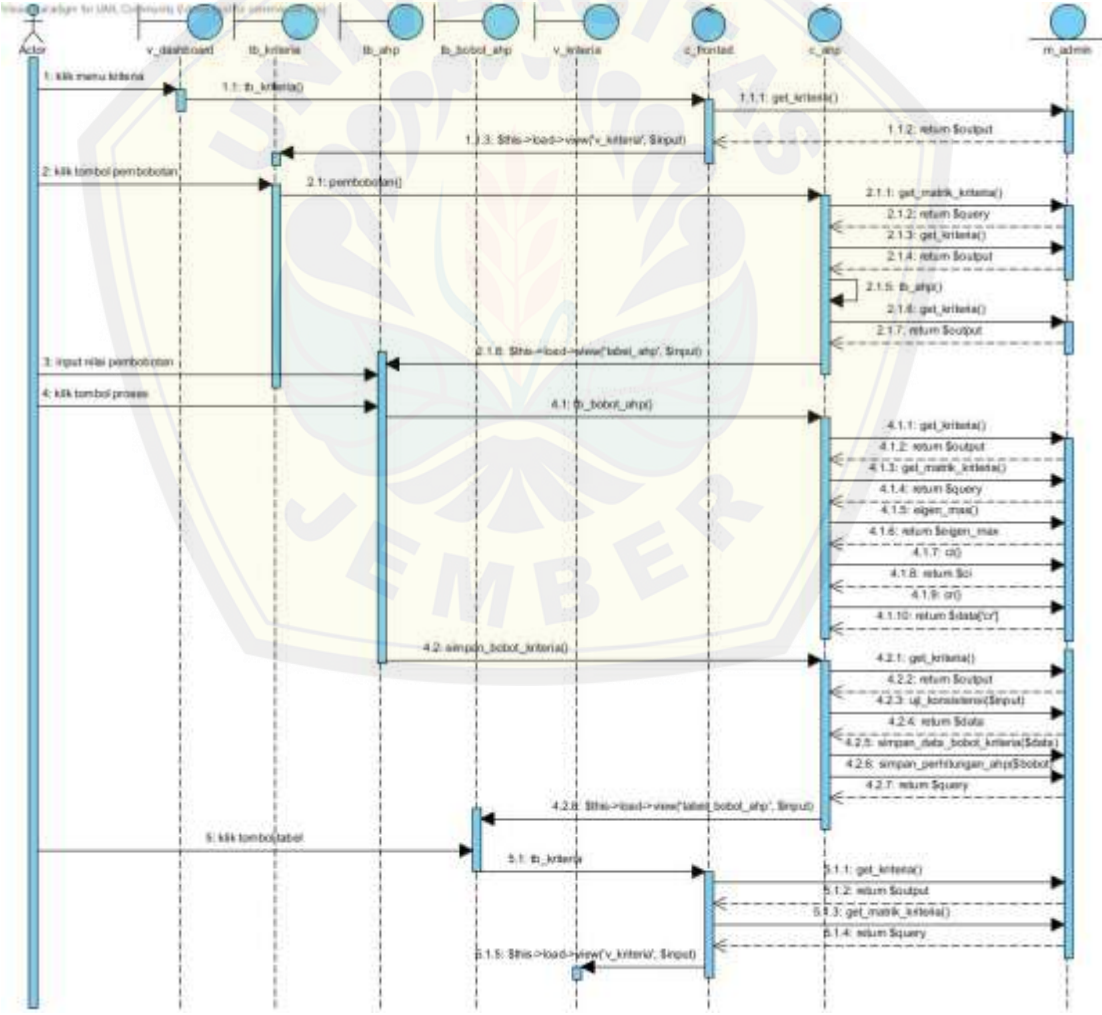
C.1.6. Mengubah data desa



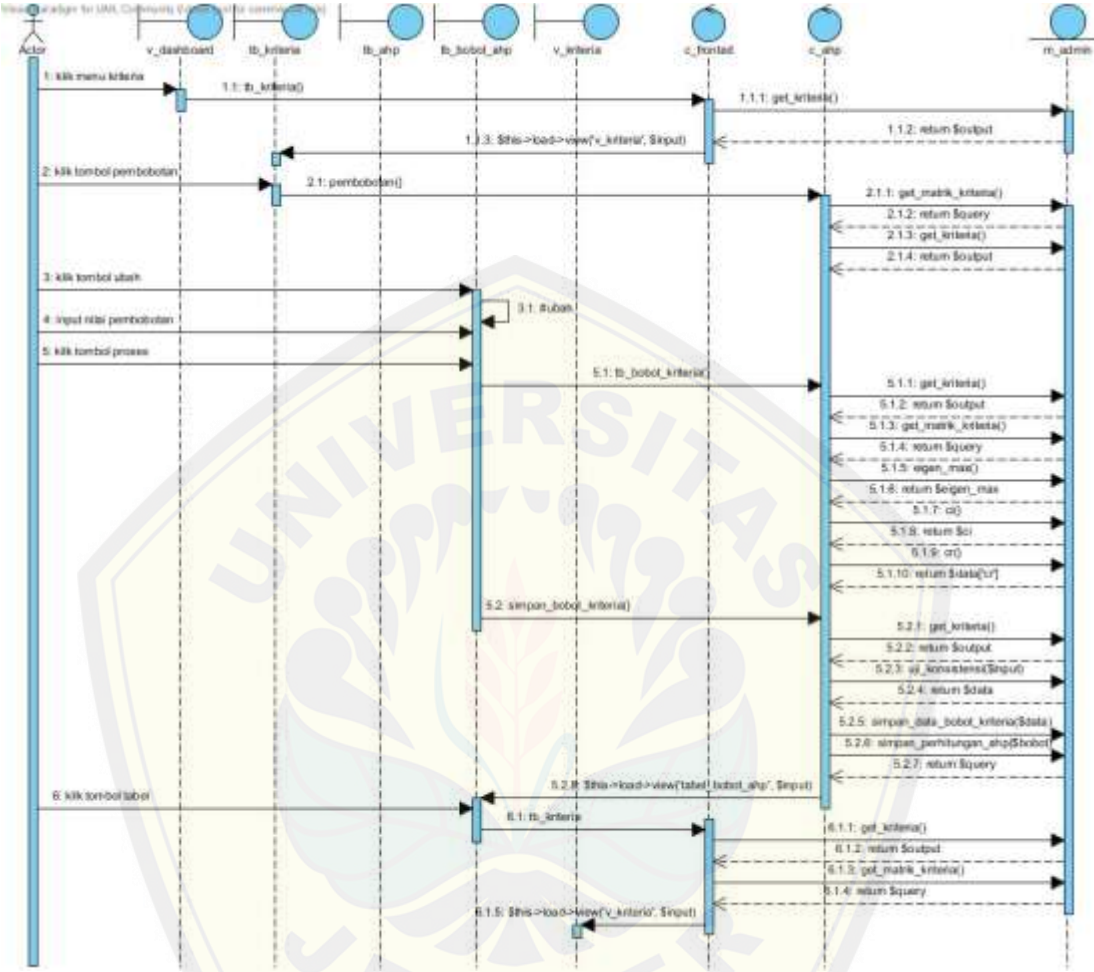
C.1.7. Melihat data desa



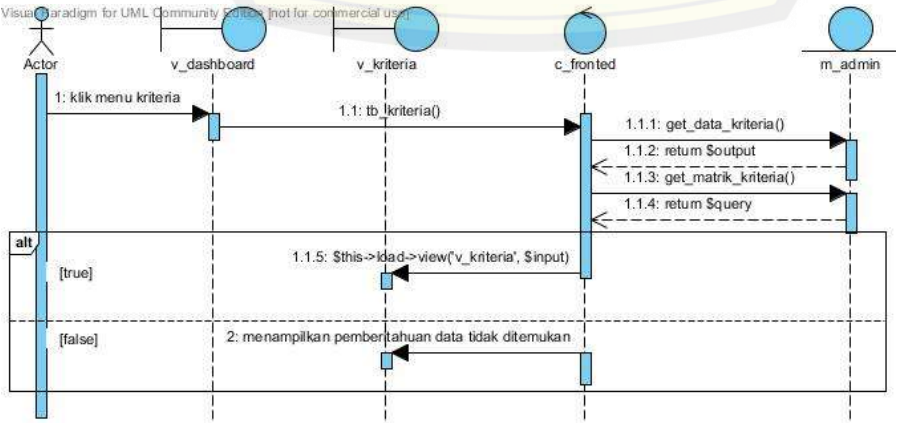
C.1.8. Pembobotan Kriteria



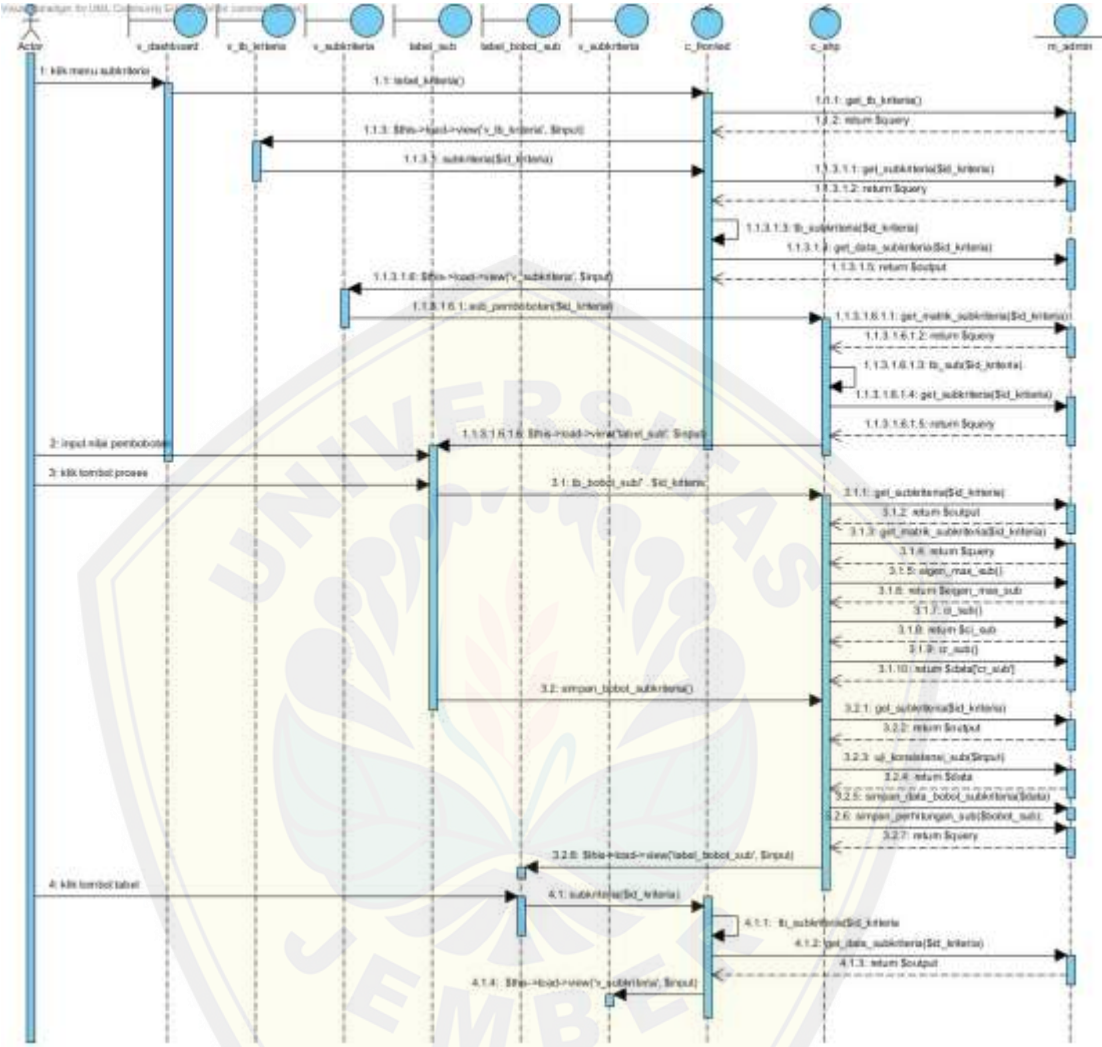
C.1.9. Mengubah bobot kriteria



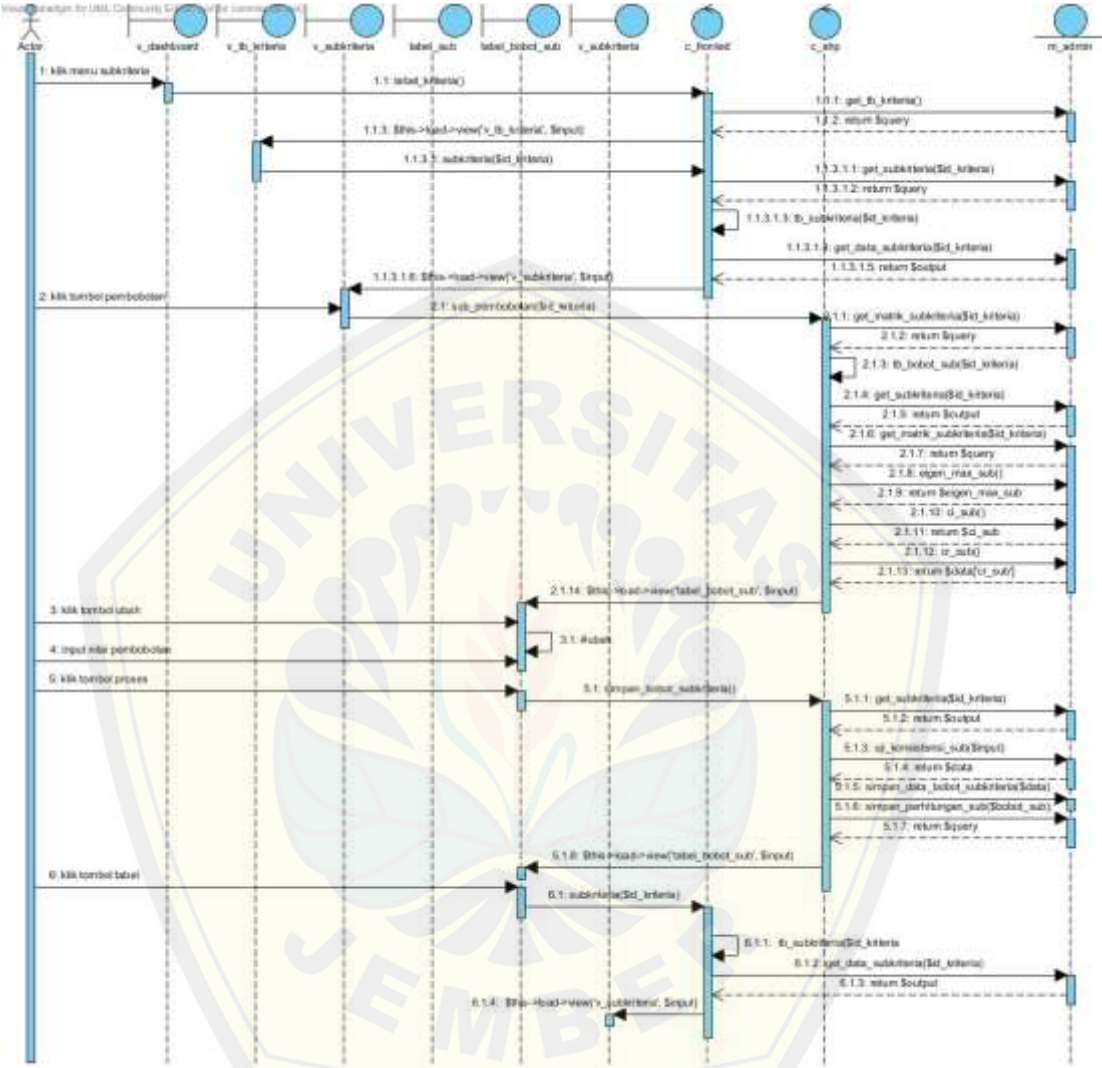
C.1.10. Melihat data kriteria



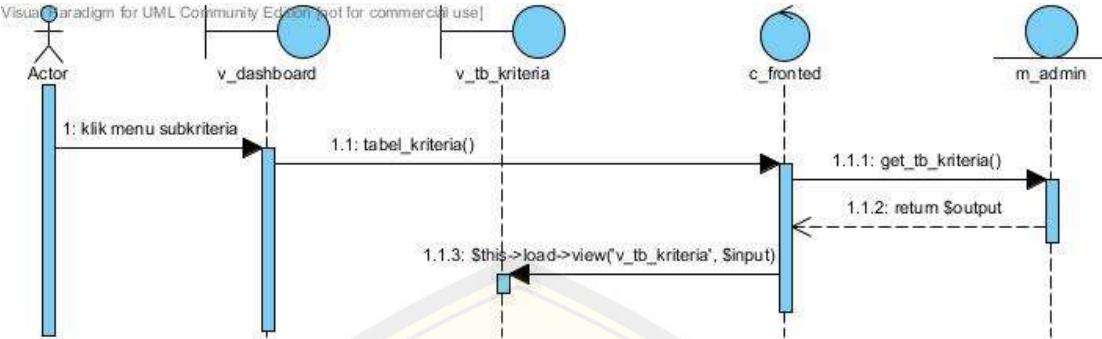
C.1.11. Pembobotan subkriteria



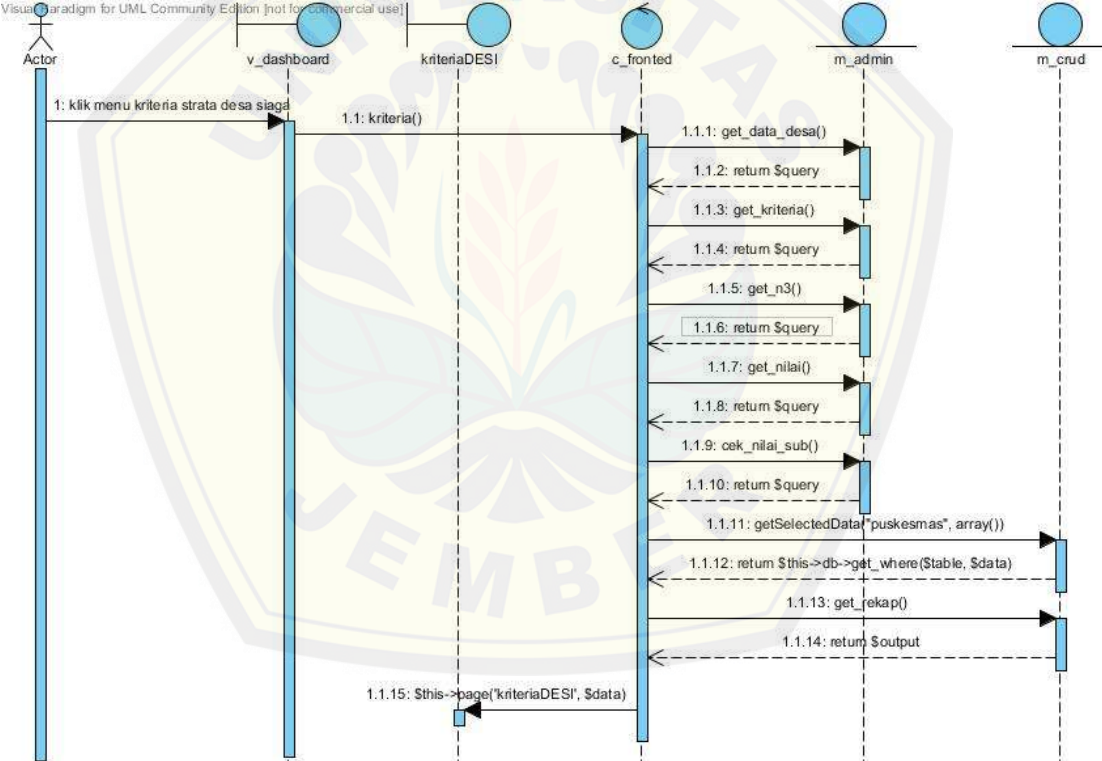
C.1.12. Mengubah bobot subkriteria



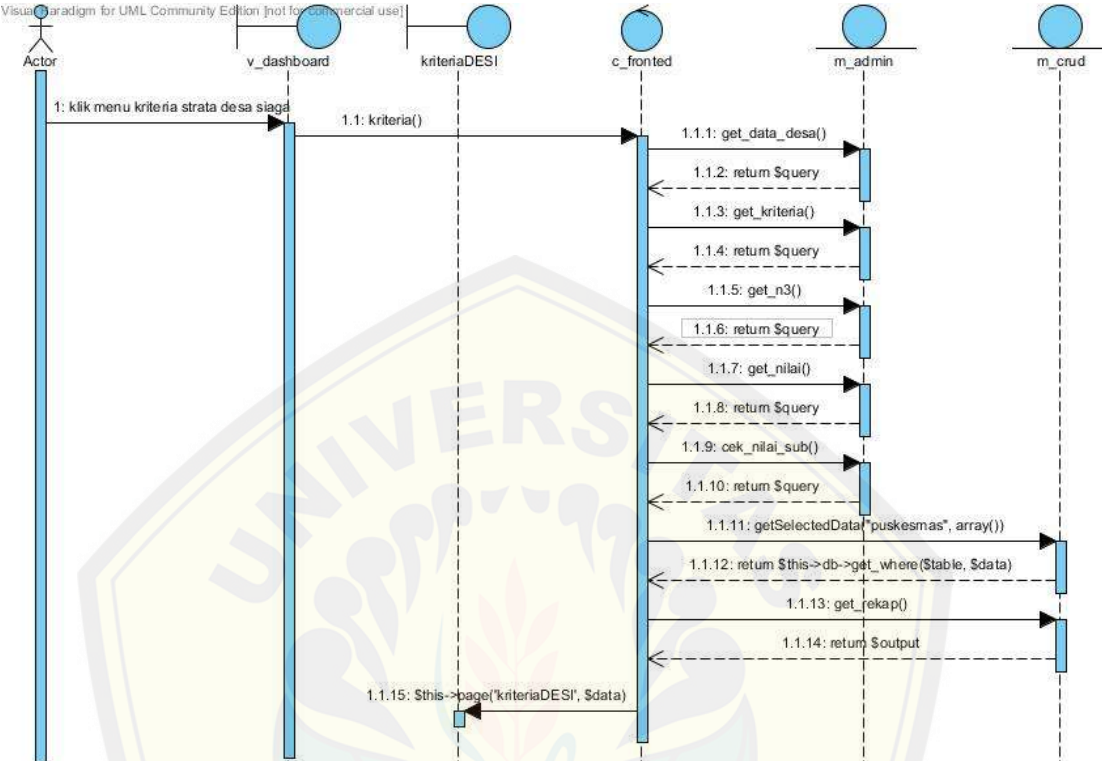
C.1.13. Melihat data subkriteria



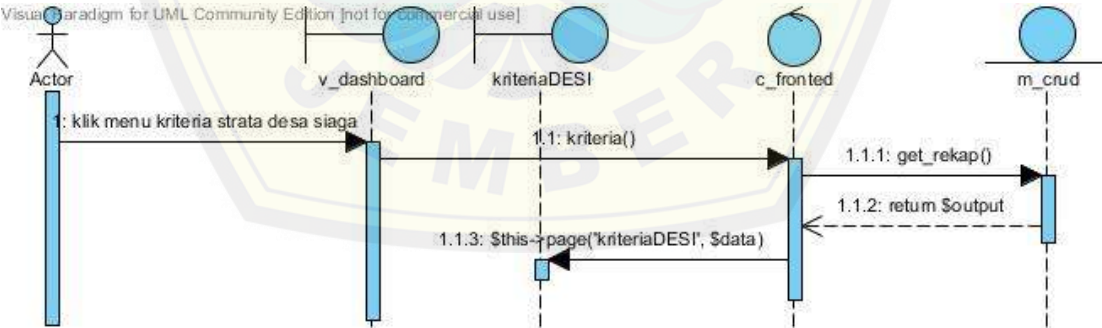
C.1.14. Pembobotan alternatif desa siaga



C.1.15. Mengubah nilai alternatif desa siaga

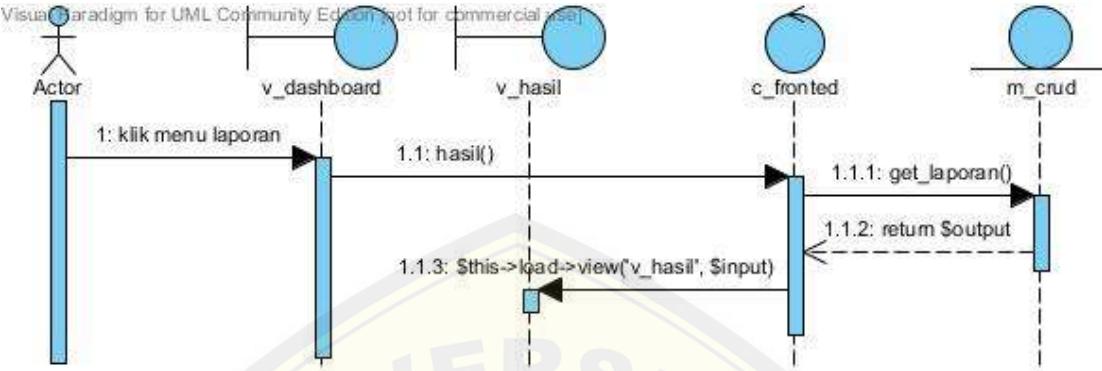


C.1.16. Melihat hasil pembobotan alternative

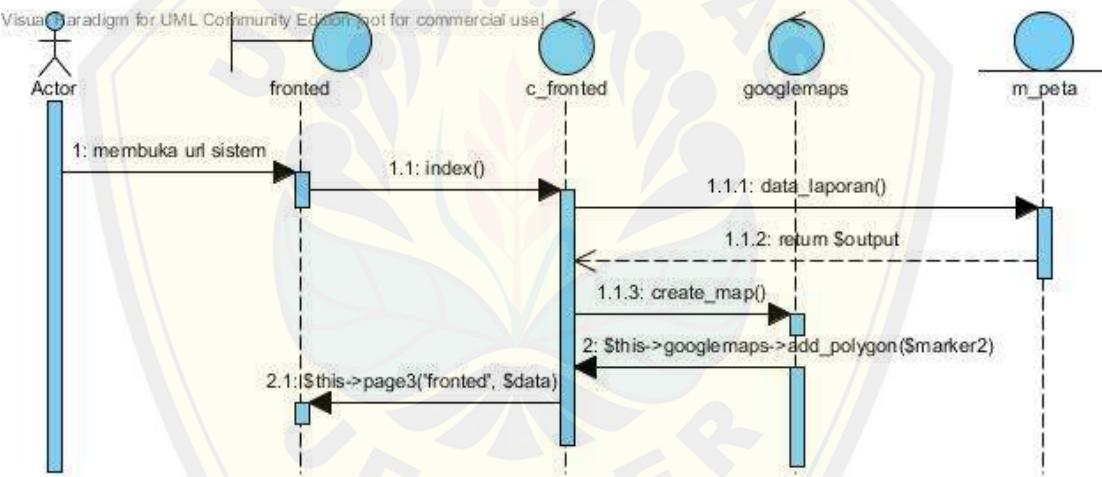


C.2. Increment Laporan

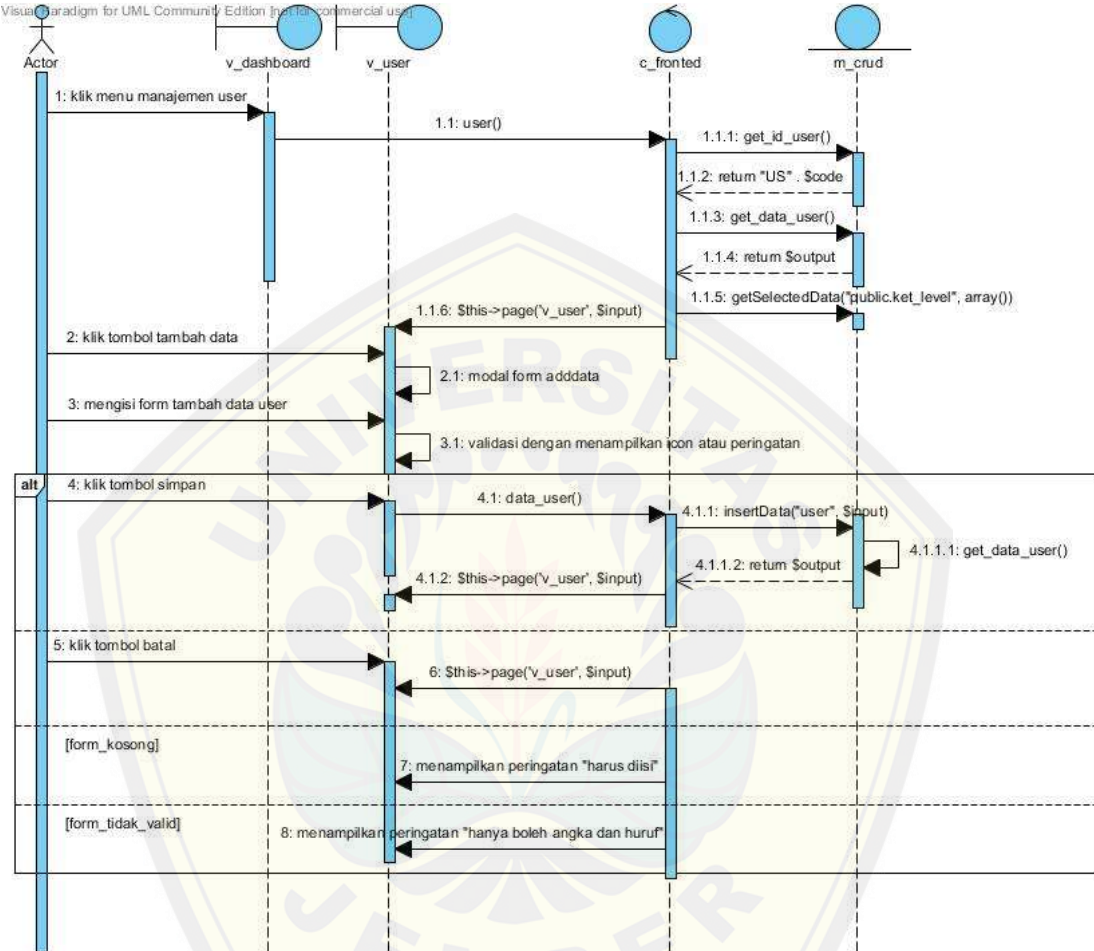
C.2.1. Melihat laporan



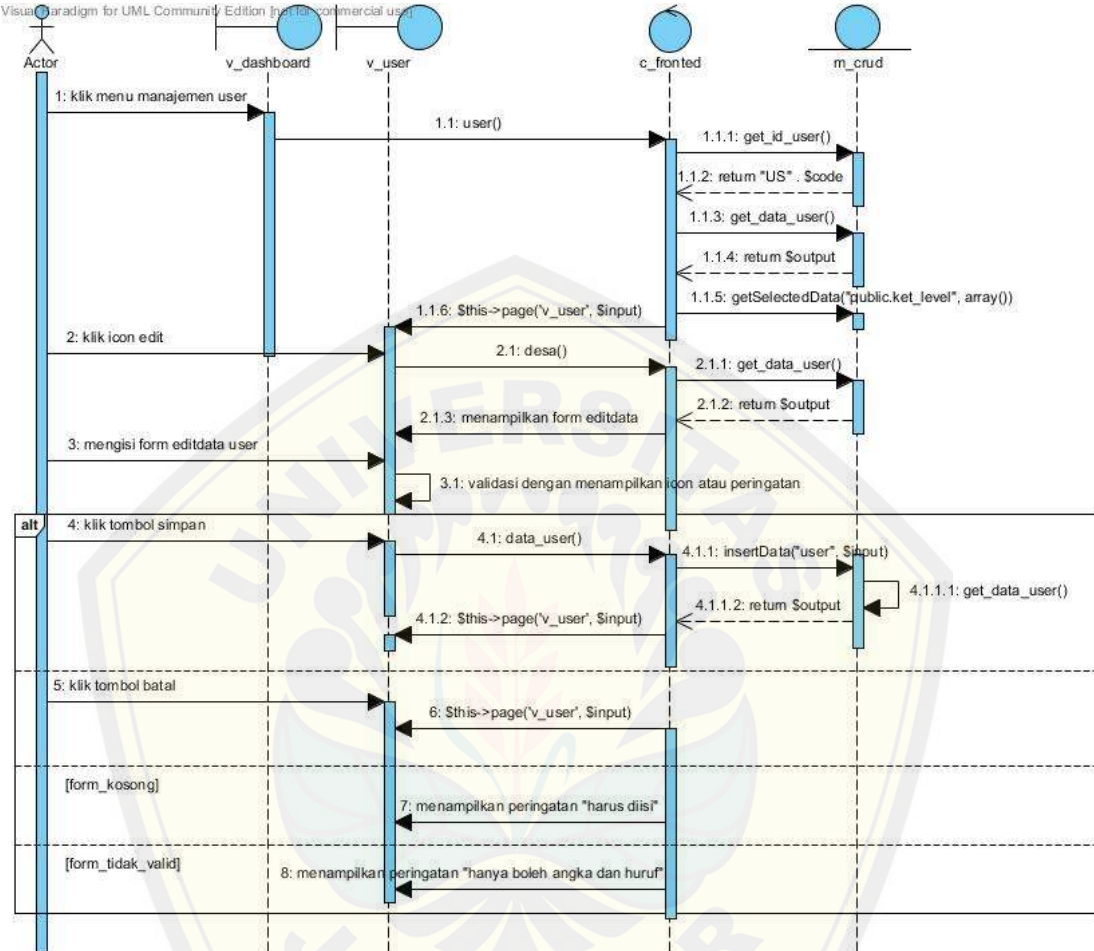
C.3. Increment Pemetaan



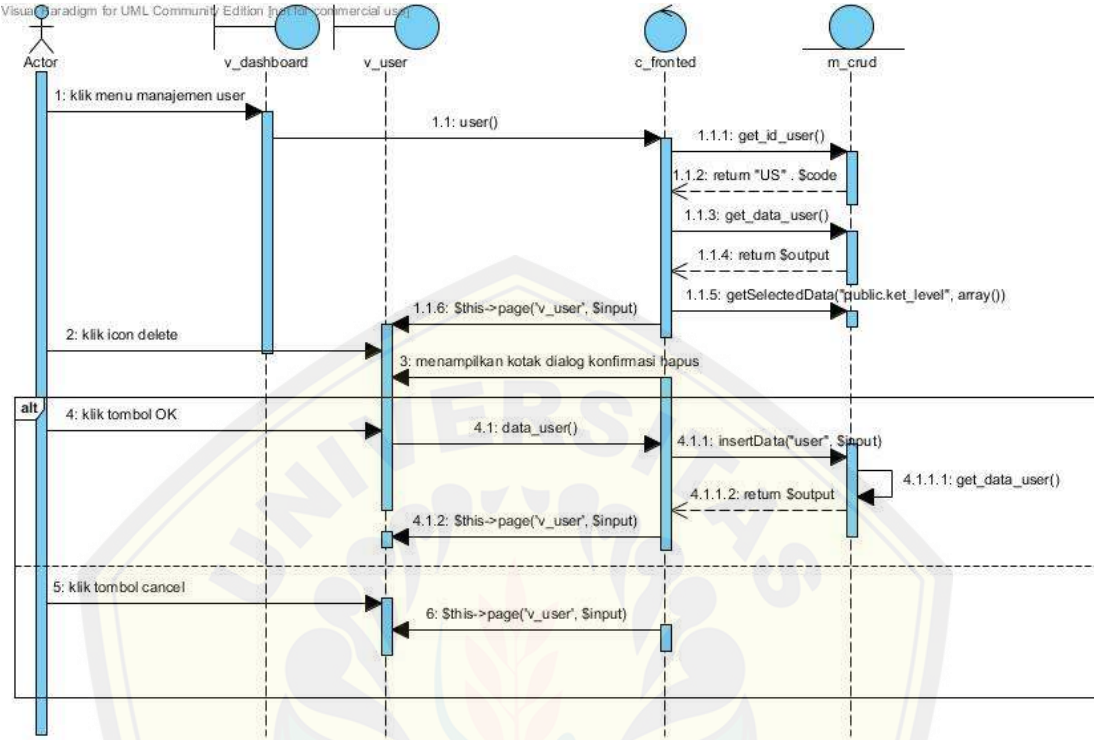
C.4. Increment Manajemen User
C.4.1. Menambah data user



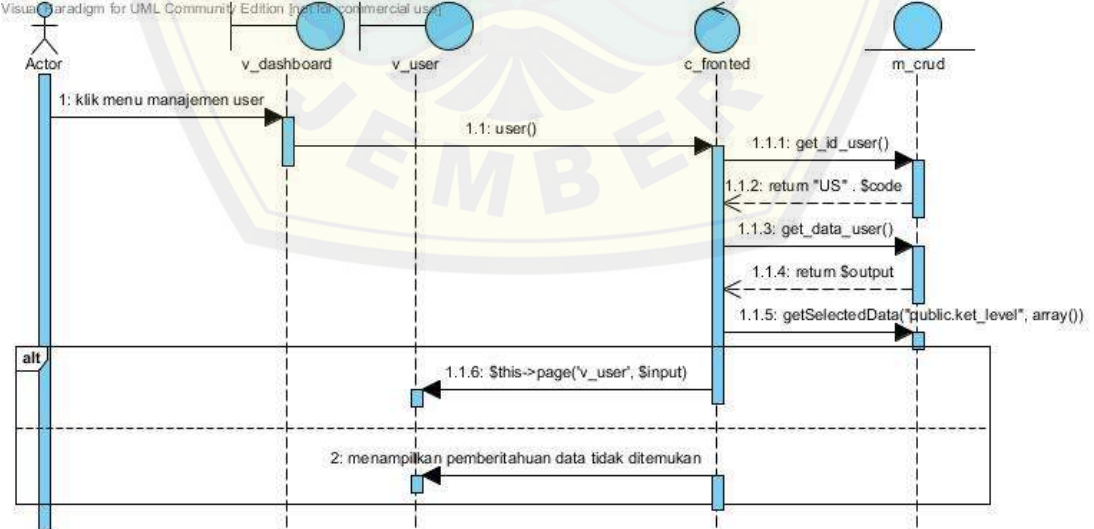
C.4.2. Mengubah data user



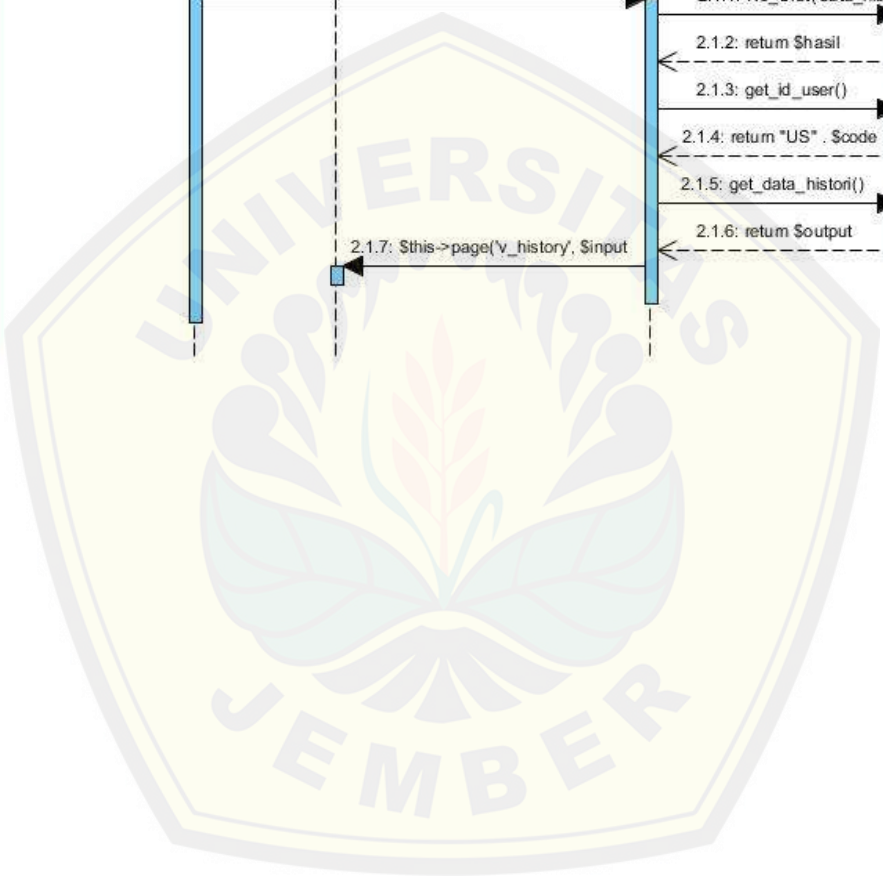
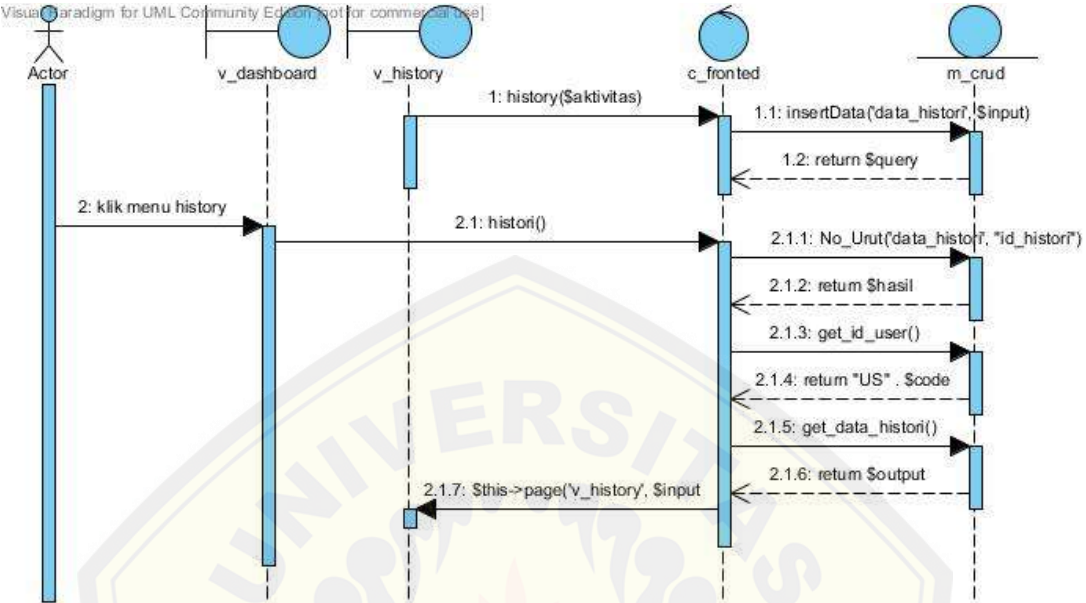
C.4.3. Menghapus data user



C.4.4. Melihat data user

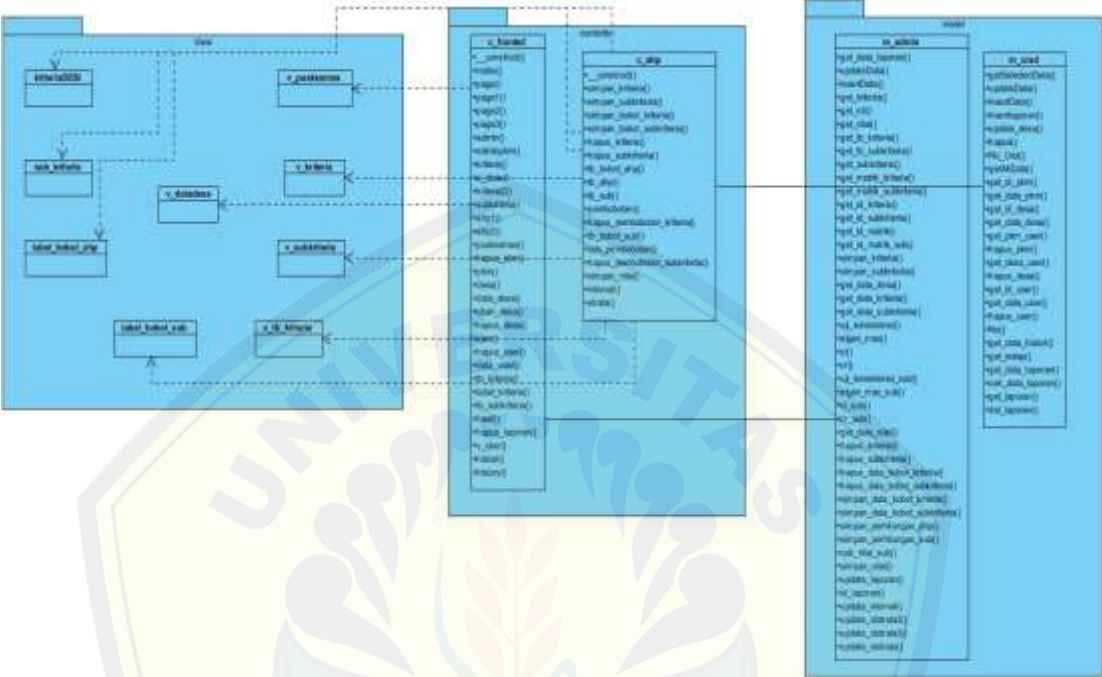


C.4.5. Melihat *history*

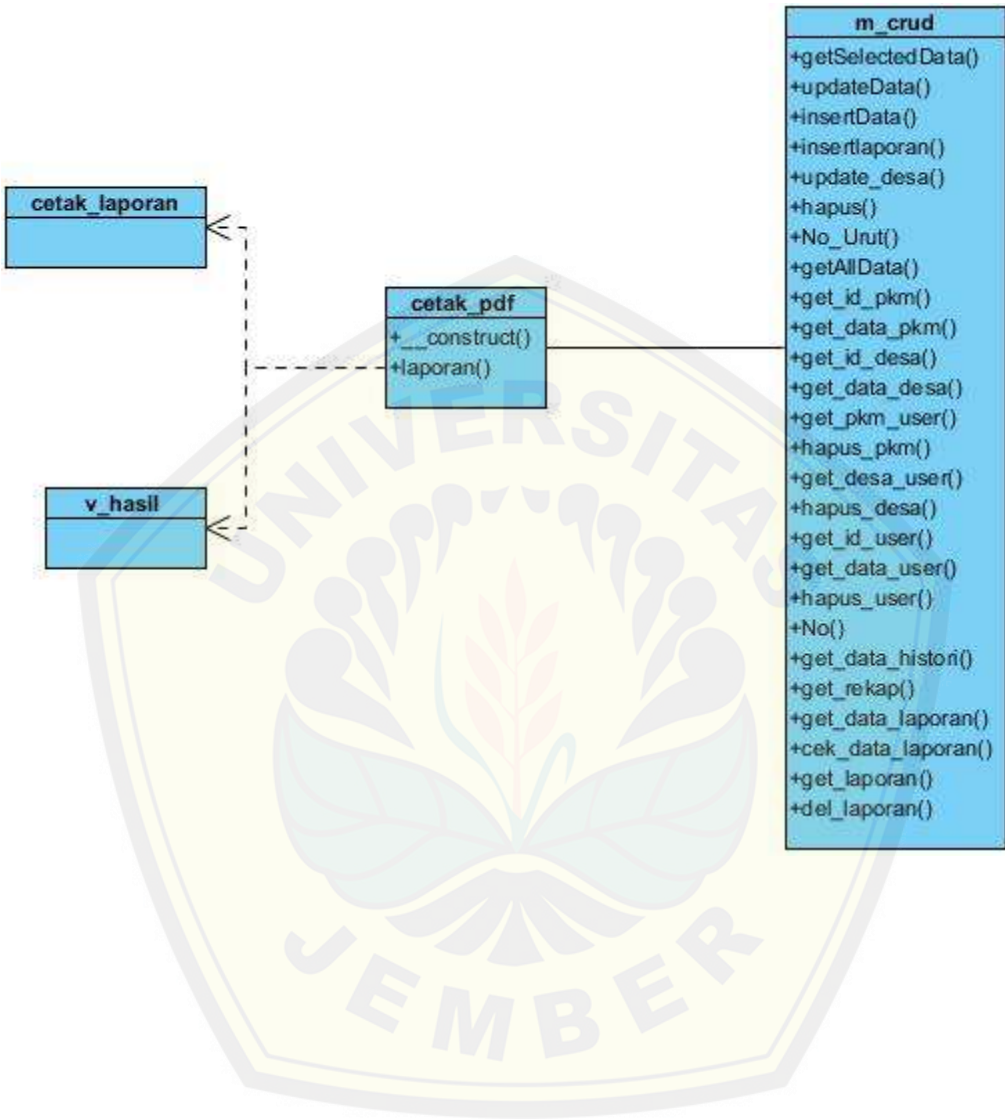


LAMPIRAN D Class Diagram

D.1. Increment Proses Penentuan Strata Desa Siaga

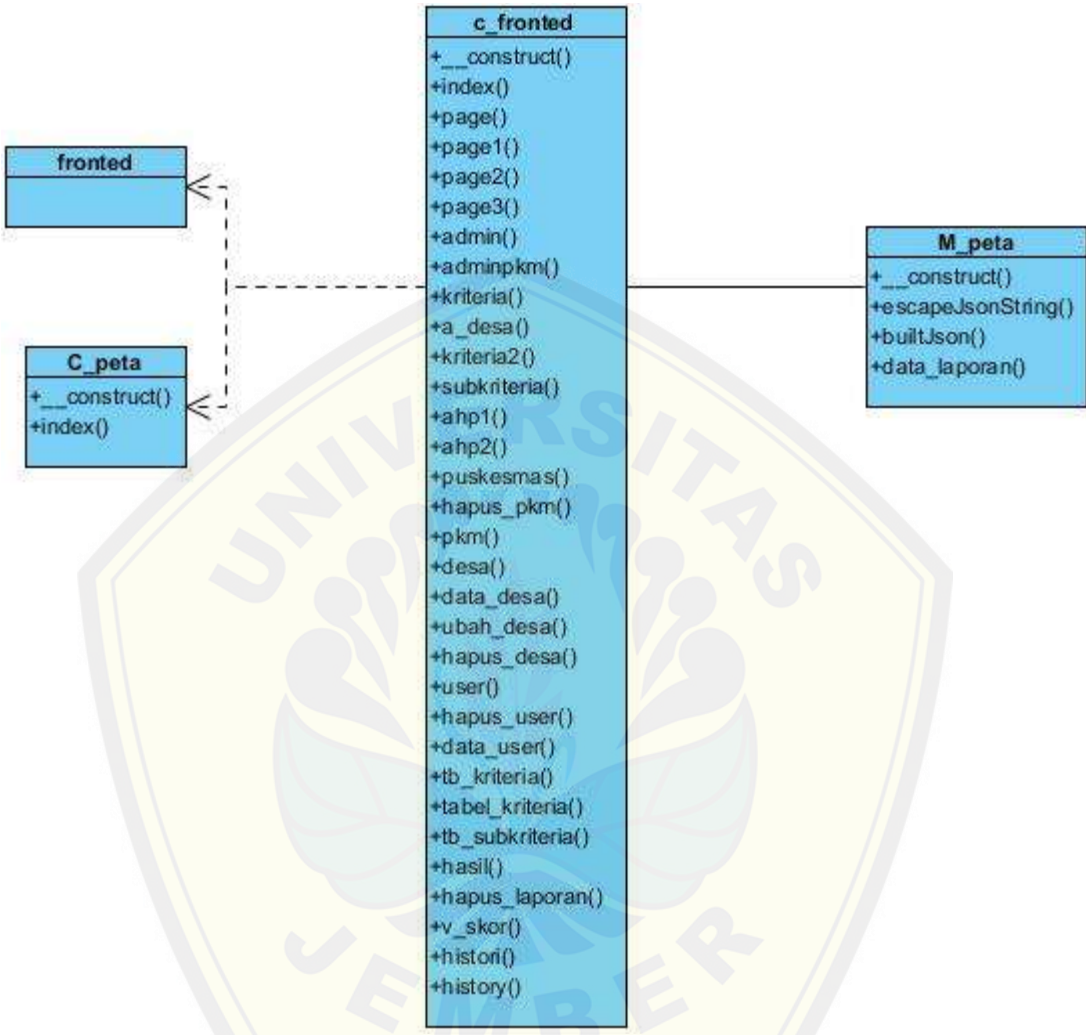


D.2. Increment Laporan

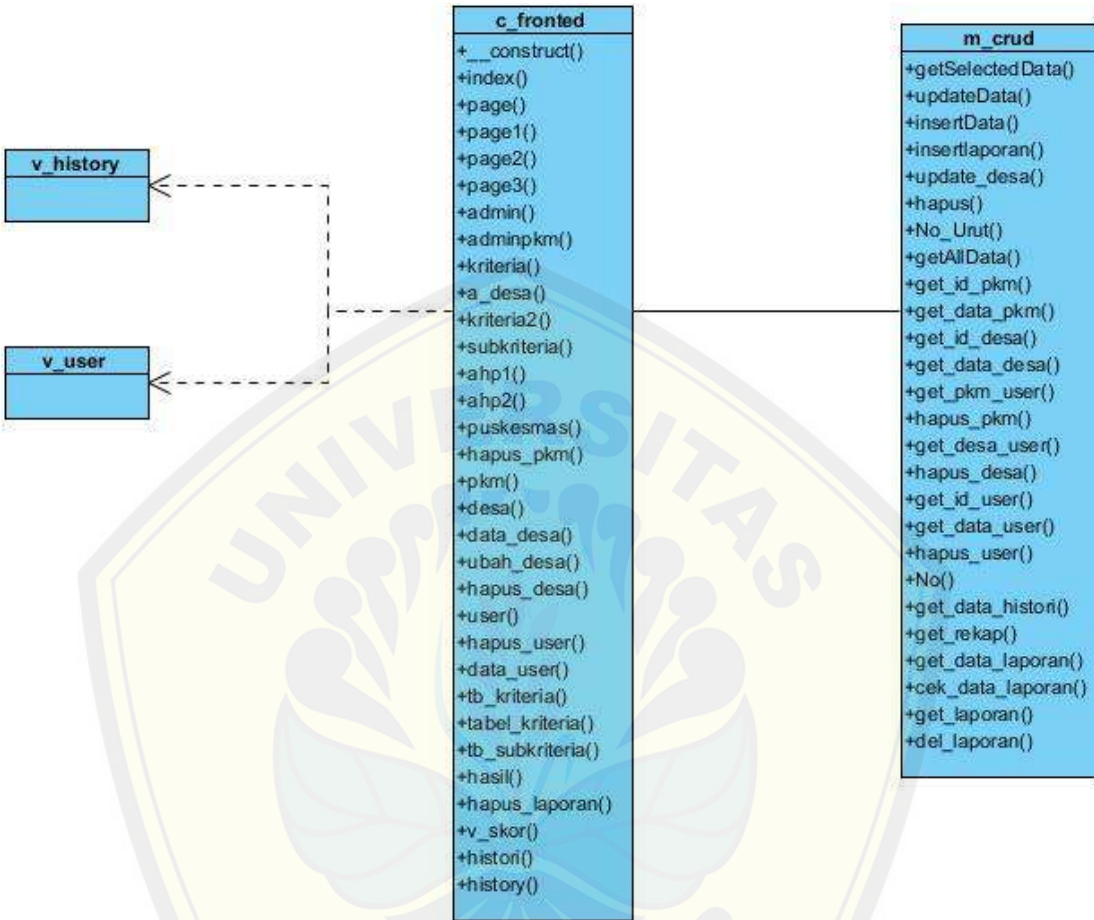


m_crud
+getSelectedData()
+updateData()
+insertData()
+insertlaporan()
+update_desa()
+hapus()
+No_Urut()
+getAllData()
+get_id_pkm()
+get_data_pkm()
+get_id_desa()
+get_data_desa()
+get_pkm_user()
+hapus_pkm()
+get_desa_user()
+hapus_desa()
+get_id_user()
+get_data_user()
+hapus_user()
+No()
+get_data_histori()
+get_rekap()
+get_data_laporan()
+cek_data_laporan()
+get_laporan()
+del_laporan()

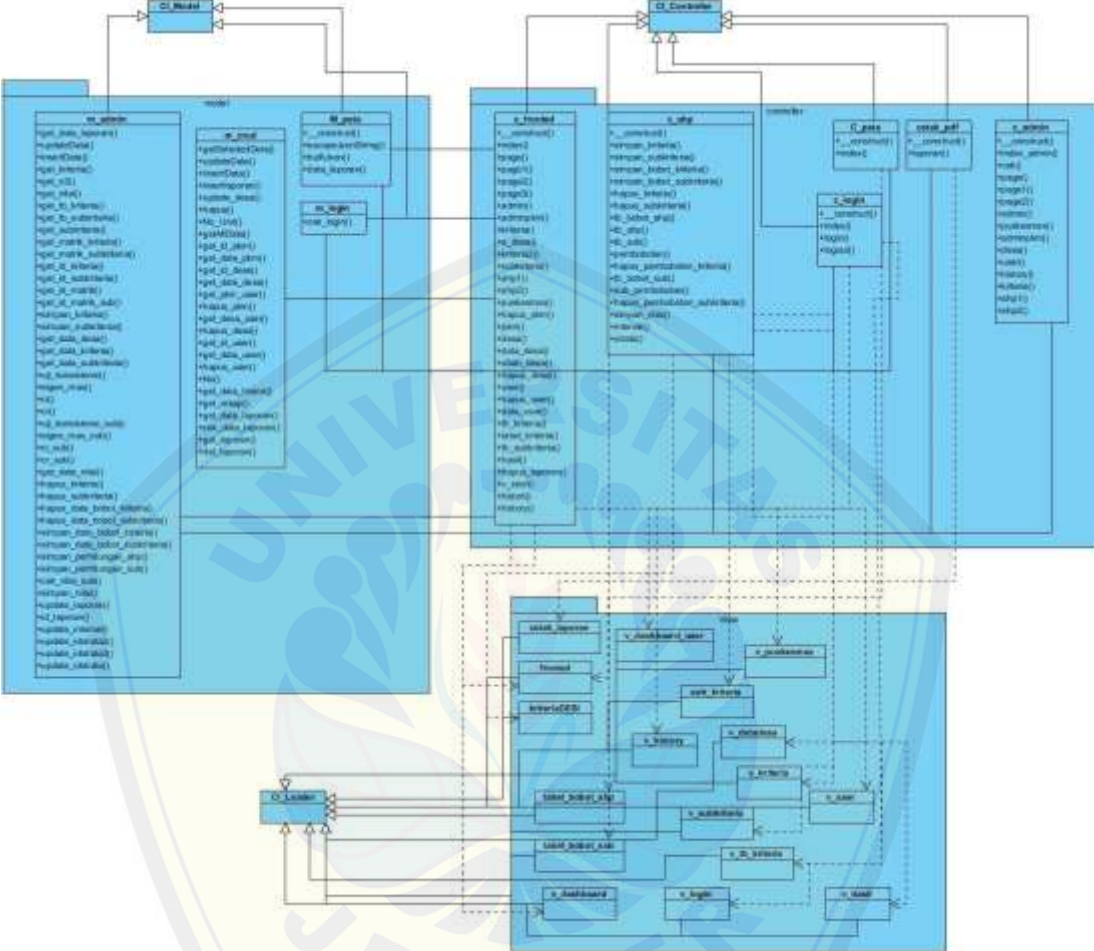
D.3. Increment Pemetaan



D.4. Increment Manajemen User

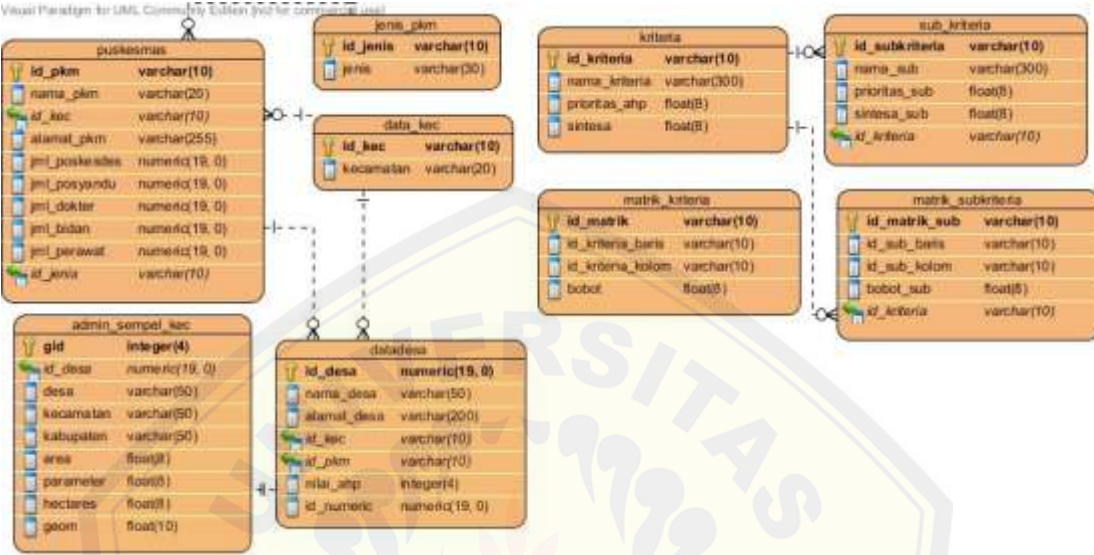


D.5. Integrasi Sistem

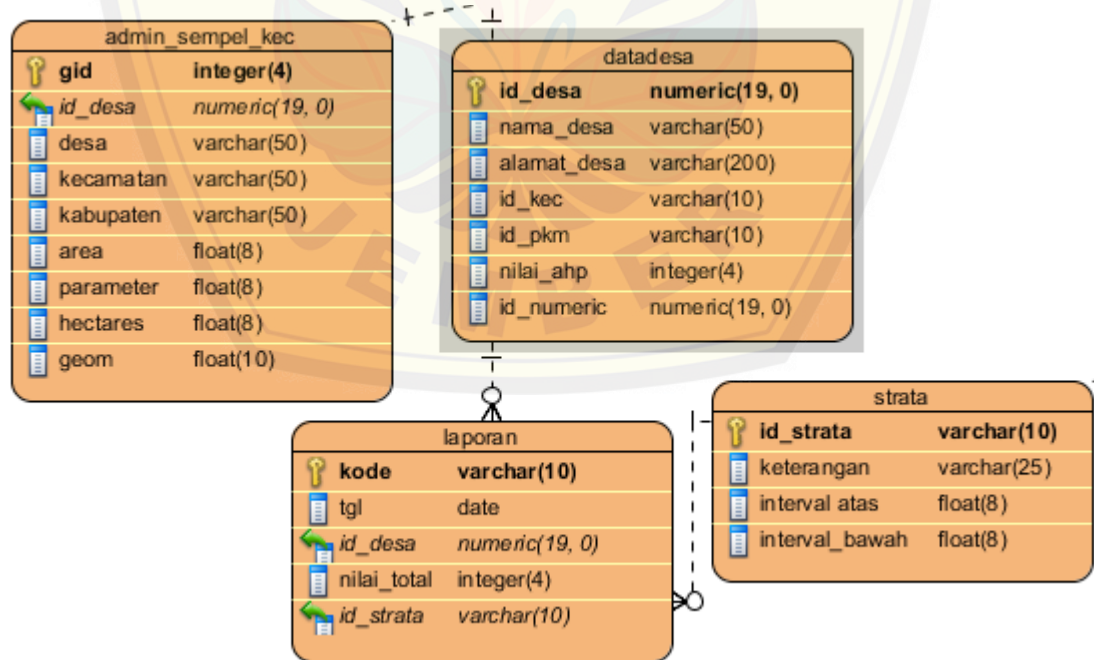


LAMPIRAN E Entity Relationship Diagram

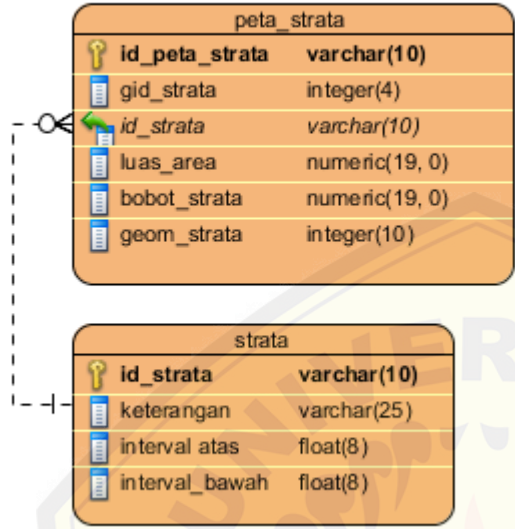
E.1. Increment Proses Penentuan Strata Desa Siaga



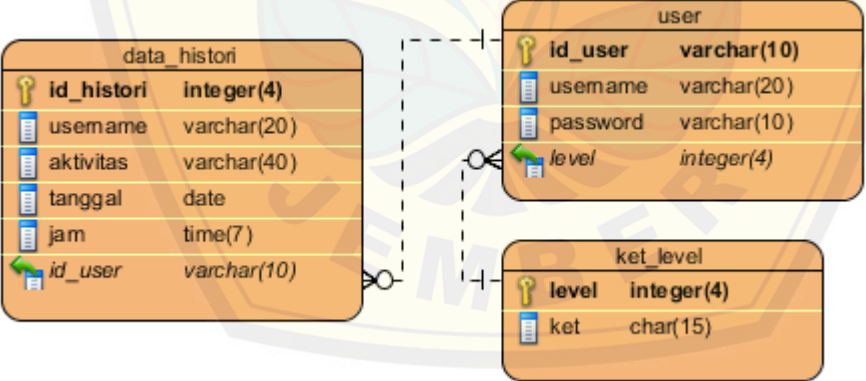
E.2. Increment Laporan



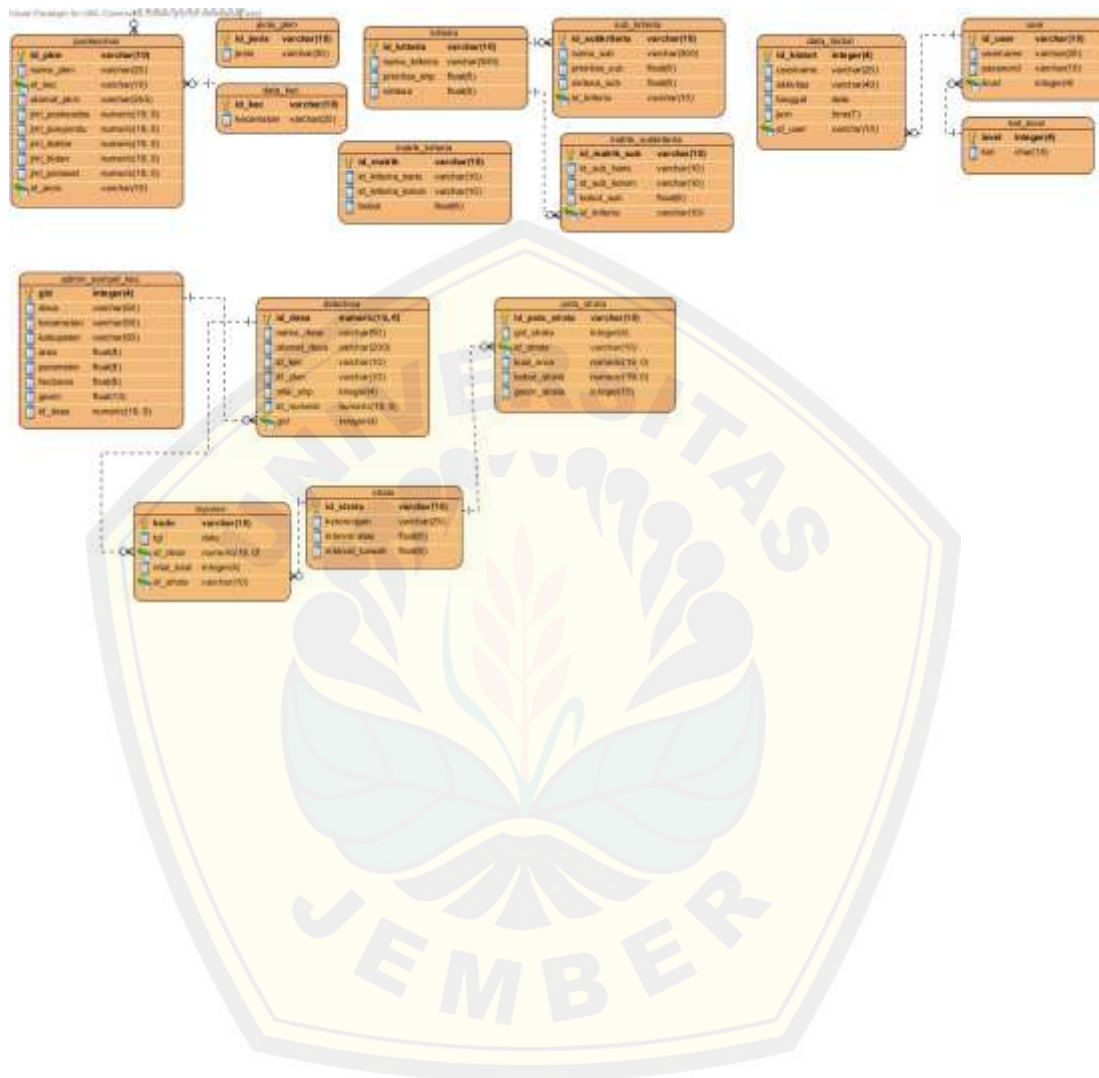
E.3. Increment Pemetaan



E.4. Increment Manajemen User



E.5. Integrasi Sistem



LAMPIRAN F Implementasi Metode AHP

F.1. *Class controller c_fronted*

```

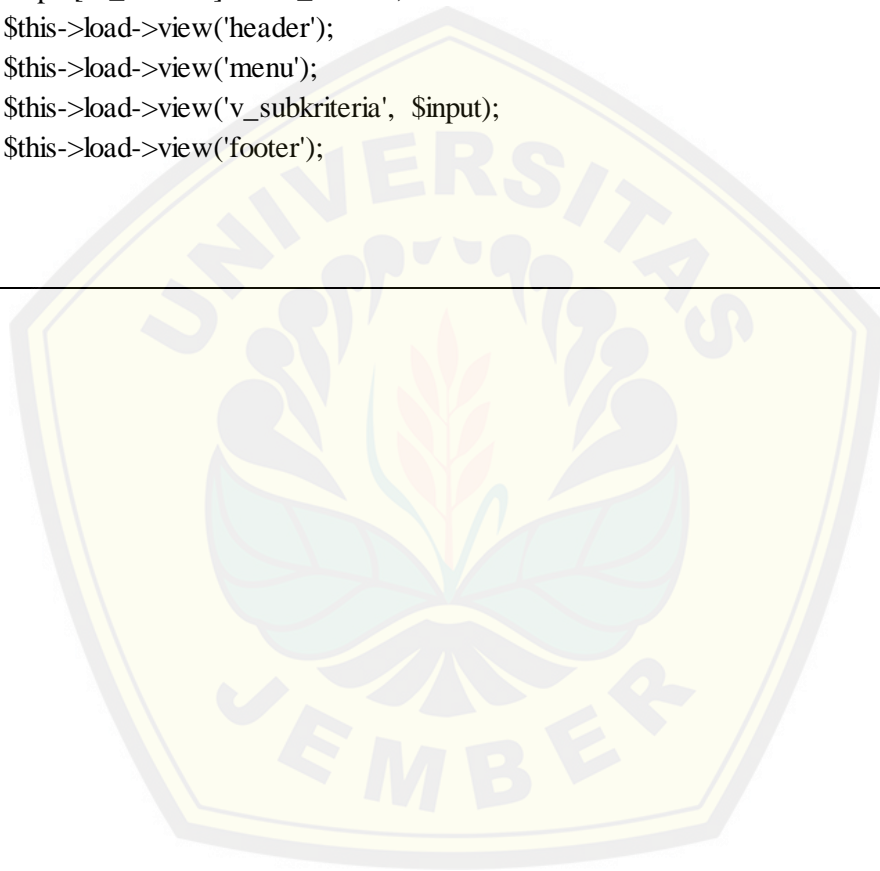
<?php

class c_fronted extends CI_Controller {
    public function __construct() {
        parent::__construct();
        $this->load->model('m_login');
        $this->load->model('m_crud');
        $this->load->model('m_admin');
    }
    public function kriteria() {
        $data['datads'] = $this->m_admin->get_data_desa();
        $data['data_kriteria'] = $this->m_admin->get_kriteria();
        $data['data_n3'] = $this->m_admin->get_n3();
        $data['data_nilai'] = $this->m_admin->get_nilai();
        $data['ceknilai'] = $this->m_admin->cek_nilai_sub();
        $data['datakec'] = $this->m_crud->getSelectedData("puskesmas", array());
        $data['getrekap'] = $this->m_crud->get_rekap();
        $this->page('kriteriaDESI', $data);
    }
    public function subkriteria($id_kriteria) {
        $data['data_subkriteria'] = $this->m_admin->get_subkriteria($id_kriteria);
        $data['id_kriteria'] = $id_kriteria;
        $this->tb_subkriteria($id_kriteria);
    }
    function tb_kriteria() {
        $input['data_kriteria'] = $this->m_admin->get_data_kriteria();
        $input['data_bobot_kriteria'] = $this->m_admin->get_matrik_kriteria();
        $this->load->view('header');
        $this->load->view('menu');
        $this->load->view('v_kriteria', $input);
        $this->load->view('footer');
    }

    function tabel_kriteria() {
        $input['tbkriteria'] = $this->m_admin->get_tb_kriteria();
        $this->load->view('header');
    }
}

```

```
$this->load->view('menu');  
$this->load->view('v_tb_kriteria', $input);  
$this->load->view('footer');  
}  
  
function tb_subkriteria($id_kriteria) {  
    $input['data_subkriteria'] = $this->m_admin->get_data_subkriteria($id_kriteria);  
    $input['id_kriteria'] = $id_kriteria;  
    $this->load->view('header');  
    $this->load->view('menu');  
    $this->load->view('v_subkriteria', $input);  
    $this->load->view('footer');  
}  
}  
?>
```



F.2. Class controller *c_ahp*

```

<?php

class c_ahp extends CI_Controller {

    public function __construct() {
        date_default_timezone_set('ASIA/Jakarta');
        parent::__construct();
        $this->load->model('m_admin');
        $this->load->model('m_crud');
    }
    function simpan_bobot_kriteria() {
        $data_kriteria = $this->m_admin->get_kriteria();
        $jml_k = $data_kriteria->num_rows();

        for ($i = 1; $i <= $jml_k; $i++) {
            for ($j = 1; $j <= $jml_k; $j++) {
                $input['bb' . $i . $j] = $this->input->post('k' . $i . $j);
            }
            $input['id' . $i] = $this->input->post('id' . $i);
        }
        $input['jml_k'] = $jml_k;
        $data3 = $this->m_admin->uji_konsistensi($input);

        if ($data3['cr'] < 0.1) {
            for ($i = 1; $i <= $jml_k; $i++) {
                for ($j = 1; $j <= $jml_k; $j++) {
                    $data['id_k1'] = $this->input->post('id' . $i);
                    $data['id_k2'] = $this->input->post('id' . $j);
                    $data2['id_kr1'] = $this->input->post('id' . $i);
                    $data2['id_kr2'] = $this->input->post('id' . $j);
                    $data['bobot'] = $this->input->post('k' . $i . $j);
                    $data = array(
                        'id_matrik' => $this->m_admin->get_id_matrik(),
                        'id_kriteria_baris' => $data['id_k1'],
                        'id_kriteria_kolom' => $data['id_k2'],
                        'bobot' => $data['bobot'],
                    );
                }
            }
        }
    }
}

```

```

        $logic2 = $this->m_admin->simpan_data_bobot_kriteria($data);
    }
}
$data_kriteria = $this->m_admin->get_kriteria();

$i = 1;

foreach ($data_kriteria->result_array() as $row) {

    $bobot['id_kriteria'] = $row['id_kriteria'];
    $bobot['bobot_prioritas'] = $data3['bobot_prioritas' . $i];
    $bobot['bobot_sintesa'] = $data3['bobot_sintesa' . $i];

    $this->m_admin->simpan_perhitungan_ahp($bobot);

    $i++;
}
if ($logic2==true) {
    $this->m_admin->update_interval();
    redirect('index.php/c_ahp/tb_bobot_ahp/');
}
else{
    $this->session->set_flashdata('message', 'Pembobotan GAGAL! Karena Matriks
Tidak Konsisten, Lakukan Pembobotan Ulang dengan Benar!!!');
//redirect("index.php/c_ahp/tb_bobot_ahp/");
}
}

function simpan_bobot_subkriteria() {
    $id_kriteria = $this->input->post('id_kriteria');
    $data_subkriteria = $this->m_admin->get_subkriteria($id_kriteria);
    $jml_k = $data_subkriteria->num_rows();
    for ($i = 1; $i <= $jml_k; $i++) {
        for ($j = 1; $j <= $jml_k; $j++) {
            $input['bb' . $i . $j] = $this->input->post('k' . $i . $j);
        }
        $input['id' . $i] = $this->input->post('id' . $i);
    }
    echo "aa";
    $input['jml_k'] = $jml_k;
}

```

```

$data3 = $this->m_admin->uji_konsistensi_sub($input);

if ($data3['cr_sub'] < 0.1) {

    for ($i = 1; $i <= $jml_k; $i++) {
        for ($j = 1; $j <= $jml_k; $j++) {
            $data['id_k1'] = $this->input->post('id' . $i);
            $data['id_k2'] = $this->input->post('id' . $j);
            $data2['id_kr1'] = $this->input->post('id' . $i);
            $data2['id_kr2'] = $this->input->post('id' . $j);
            $data['bobot'] = $this->input->post('k' . $i . $j);

            $data = array(
                'id_matrik_sub' => $this->m_admin->get_id_matrik_sub(),
                'id_sub_baris' => $data['id_k1'],
                'id_sub_kolom' => $data['id_k2'],
                'bobot_sub' => $data['bobot'],
                'id_kriteria' => $id_kriteria
            );

            $log2 = $this->m_admin->simpan_data_bobot_subkriteria($data);
        }
    }
    $data_subkriteria = $this->m_admin->get_subkriteria($id_kriteria);
    $i = 1;
    foreach ($data_subkriteria->result_array() as $row) {
        $bobot_sub['id_subkriteria'] = $row['id_subkriteria'];
        $bobot_sub['bobot_prioritas_sub'] = $data3['bobot_prioritas_sub' . $i];
        $bobot_sub['bobot_sintesa_sub'] = $data3['bobot_sintesa_sub' . $i];
        $this->m_admin->simpan_perhitungan_sub($bobot_sub);
        $i++;
    }
}
$this->m_admin->update_interval();
redirect('index.php/c_ahp/tb_bobot_sub/' . $id_kriteria);
}

function tb_bobot_ahp() {
    // error_reporting(0);

```



```
$input['data_kriteria'] = $this->m_admin->get_kriteria();
$input['data_bobot_kriteria'] = $this->m_admin->get_matrik_kriteria();
$input['eigen_max'] = $this->m_admin->eigen_max();
$input['ci'] = $this->m_admin->ci();
$input['cr'] = $this->m_admin->cr();
$this->load->view('header');
$this->load->view('tabel_bobot_ahp', $input);
}

function pembobotan() {
//    $this->cek();
    $cek = $this->m_admin->get_matrik_kriteria();
    if ($cek->num_rows() > 0) {
        $this->tb_bobot_ahp();
    } else {
        $this->tb_ahp();
    }
}

function tb_bobot_sub($id_kriteria) {
//    $this->cek();
    error_reporting(0);
    $input['data_subkriteria'] = $this->m_admin->get_subkriteria($id_kriteria);
    $input['data_bobot_subkriteria'] = $this->m_admin->get_matrik_subkriteria($id_kriteria);
    $input['eigen_max_sub'] = $this->m_admin->eigen_max_sub();
    $input['ci_sub'] = $this->m_admin->ci_sub();
    $input['cr_sub'] = $this->m_admin->cr_sub();
    $input['id_kriteria'] = $id_kriteria;
    $this->load->view('header');
    $this->load->view('menu2');
    $this->load->view('tabel_bobot_sub', $input);
//    $this->load->view('footer');
}

function sub_pembobotan($id_kriteria) {
//    $this->cek();
    $cek = $this->m_admin->get_matrik_subkriteria($id_kriteria);
    if ($cek->num_rows() > 0) {
```

```
        $this->tb_bobot_sub($id_kriteria);
    } else {
        $this->tb_sub($id_kriteria);
    }
}

function simpan_nilai() {
    $jml_k = $this->m_admin->get_tb_kriteria();
    $data['nilai_total'] = 0;
    $data['id_desa'] = $this->input->post('id_desa');

    $data2['kode'] = $this->m_admin->id_laporan();
    $data2['tgl'] = date('Y-m-d');
    $data2['id_desa'] = $this->input->post('id_desa');
    $data2['nilai_total'] = 0;

    $nilai = 0;

    for ($i = 1; $i <= $jml_k->num_rows(); $i++) {
        echo $nilai = $this->input->post('n' . $i);

        $data2['nilai_total'] += $nilai;
    }
    $a = $this->m_admin->simpan_nilai($data);
    echo $a;
    if ($a) {
        $this->m_admin->update_idstrata2($data['id_desa']);
    }

    $date = date('Y-m-d');
    $cek = $this->m_crud->cek_data_laporan($date, $data2['id_desa']);

    if (count($cek) > 0) {
        foreach ($cek as $value) {
            $this->m_admin->update_laporan($data2['nilai_total'], $value['kode']);
            $this->m_admin->update_idstrata($value['kode']);
        }
    } else {
        $b = $this->m_crud->insertlaporan($data2);
    }
}
```

```
        if ($b) {  
            $this->m_admin->update_idstrata($data2['kode']);  
        }  
    }  
    redirect(base_url("index.php/c_froned/kriteria"));  
}  
  
function interval() {  
    $this->m_admin->update_interval();  
}  
  
function strata() {  
    $this->m_admin->update_idstrata2();  
}  
}  
?>
```



F.3. Class model *m_admin*

```

<?php

class m_admin extends CI_model {
    function updateData($table, $data, $field_key) {
        $query = $this->db->update($table, $data, $field_key);
    }
    function insertData($table, $data) {
        $query = $this->db->insert($table, $data);
    }

    public function get_kriteria() {
        $query = $this->db->query("select * from kriteria");
        return $query;
    }
    public function get_n3() {
        $query = $this->db->query("select nama_kriteria, (prioritas_ahp*100)::integer from
kriteria");
        return $query;
    }

    public function get_nilai() {
        $query = $this->db->query("SELECT k.id_kriteria, k.nama_kriteria,
(k.prioritas_ahp*100)::integer as kriteria, s.id_subkriteria, s.nama_sub,
(s.prioritas_sub*100)::integer as subkri from kriteria k, sub_kriteria s WHERE
k.id_kriteria = s.id_kriteria");
        return $query;
    }

    public function get_tb_kriteria() {
        $query = $this->db->query("select
id_kriteria,nama_kriteria,(prioritas_ahp*100)::integer as prioritas_ahp from public.kriteria
");
        return $query;
    }

    public function get_tb_subkriteria() {
        $query = $this->db->query("select id_subkriteria,nama_sub,prioritas_sub from
public.sub_kriteria ");
    }
}

```

```
return $query;
}

public function get_subkriteria($id_kriteria) {
    $query = $this->db->query("select * from sub_kriteria WHERE id_kriteria = '" .
    $id_kriteria . "'");
    return $query;
}

public function get_matrik_kriteria() {
    $query = $this->db->query("select * from matrik_kriteria");
    return $query;
}

public function get_matrik_subkriteria($id_kriteria) {
    $query = $this->db->query("select * from matrik_subkriteria WHERE id_kriteria = '" .
    $id_kriteria . "'");
    return $query;
}

public function get_data_kriteria() {
    $output = "";
    $query = $this->db->query("select * from kriteria ");
    $cek = $this->get_matrik_kriteria();
    $no = 0;
    foreach ($query->result_array() as $row) {
        if ($cek->num_rows() <= 0) {
            $output.='<tr>
                <td align=center>
                    ' . (++$no) . '
                </td>
                <td align=center>
                    ' . $row["id_kriteria"] . '
                </td>
                <td align=center>
                    ' . $row["nama_kriteria"] . '
                </td>
                <td align=center>
```

```

        '. $row["prioritas_ahp"] .'
    </td>
</tr>;
} else {
    $output.='<tr>
        <td align=center>
            '. (++$no) .'
        </td>
        <td align=center>
            '. $row["id_kriteria"] .'
        </td>
        <td align=center>
            '. $row["nama_kriteria"] .'
        </td>
        <td align=center>
            '. $row["prioritas_ahp"] .'
        </td>
    </tr>';
}
}
return $output;
}

public function get_data_subkriteria($id_kriteria) {
    $output = "";
    $query = $this->db->query("select * from sub_kriteria WHERE id_kriteria = '" .
$id_kriteria . "'");
    $cek = $this->get_matrik_kriteria();
    $no = 0;
    foreach ($query->result_array() as $row) {
        if ($cek->num_rows() <= 0) {
            $output.='<tr>
                <td align=center>
                    '. (++$no) .'
                </td>
                <td align=center>
                    '. $row["id_subkriteria"] .'
                </td>
                <td align=center>

```

```

        '. $row["nama_sub"] .'
    </td>
    <td align=center>
        '. $row["prioritas_sub"] .'
    </td>
</tr>;
} else {
    $output.='<tr>
    <td align=center>
        '. (++$no) .'
    </td>
    <td align=center>
        '. $row["id_subkriteria"] .'
    </td>
    <td align=center>
        '. $row["nama_sub"] .'
    </td>
    <td align=center>
        '. $row["prioritas_sub"] .'
    </td>
    </tr>;
}
}
return $output;
}

function uji_konsistensi($data) {
    $temp_jml = 0;
    for ($i = 1; $i <= $data['jml_k']; $i++) {
        for ($j = 1; $j <= $data['jml_k']; $j++) {
            $input = $data['bb' . $j . $i];
            $temp_jml = $temp_jml + $input;
        }
        $data['jml_kolom' . $i] = $temp_jml;
        $temp_jml = 0;
    }

    $temp_jml = 1;
    for ($i = 1; $i <= $data['jml_k']; $i++) {

```

```
for ($j = 1; $j <= $data['jml_k']; $j++) {
    $input = $data['bb' . $i . $j];
    $temp_jml = $temp_jml * $input;
}
$data['jml_kali_baris' . $i] = pow($temp_jml, (1 / $data['jml_k']));
$temp_jml = 1;
}
$temp_jml = 0;
for ($i = 1; $i <= $data['jml_k']; $i++) {
    $input = $data['jml_kali_baris' . $i];
    $temp_jml = $temp_jml + $input;
}

$data['total_kali_baris'] = $temp_jml;
//eigen vector
for ($i = 1; $i <= $data['jml_k']; $i++) {
    $data['bobot_prioritas' . $i] = $data['jml_kali_baris' . $i] / $data['total_kali_baris'];
}

for ($i = 1; $i <= $data['jml_k']; $i++) {
    for ($j = 1; $j <= $data['jml_k']; $j++) {
//menentukan ranking kriteria
        $data['cell' . $i . $j] = $data['bb' . $i . $j] / $data['jml_kolom' . $j];
    }
}

$temp_jml = 0;

for ($i = 1; $i <= $data['jml_k']; $i++) {
    for ($j = 1; $j <= $data['jml_k']; $j++) {
//jumlah per baris
        $input = $data['cell' . $i . $j];
        $temp_jml = $temp_jml + $input;
    }

    $data['bobot_sintesa' . $i] = $temp_jml;
    $temp_jml = 0;
}
}
```



```

    for ($i = 1; $i <= $data['jml_k']; $i++) {
//eigen normalisasi
        $data['sin_div_pri' . $i] = $data['bobot_sintesa' . $i] / $data['bobot_prioritas' . $i];
        $temp_jml = $temp_jml + $data['sin_div_pri' . $i];
    }

    $data['total_div'] = $temp_jml;
//    $eigen_max=0;
    $data['eigen_max'] = $data['total_div'] / $data['jml_k'];

//menghitung CI
    $data['ci'] = ($data['eigen_max'] - $data['jml_k']) / ($data['jml_k'] - 1);
    $data['cr'] = $data['ci'] / 1.41;

    return $data;
}

function eigen_max() {

    $query = $this->db->query("SELECT * FROM kriteria");

    $total = 0;
    $jml_k = $query->num_rows();
    foreach ($query->result_array() as $row) {
        $total += $row['sintesa_ahp'] / $row['prioritas_ahp'];
    }
    $eigen_max = $total / $jml_k;
    return $eigen_max;
}

function ci() {
    $query = $this->db->query("SELECT * FROM kriteria");
    $jml_k = $query->num_rows();
    $eigen_max = $this->eigen_max();

    $ci = ($eigen_max - $jml_k) / ($jml_k - 1);

    return $ci;
}

```

```
function cr() {  
  
    $query = $this->db->query("SELECT * FROM kriteria");  
    $data['jml_k'] = $query->num_rows();  
  
    $data['ci'] = $this->ci();  
  
    if ($data['jml_k'] == 1) {  
        $data['cr'] = 0;  
    } else if ($data['jml_k'] == 2) {  
        $data['cr'] = 0;  
    } else if ($data['jml_k'] == 3) {  
        $data['cr'] = $data['ci'] / 0.58;  
    } else if ($data['jml_k'] == 4) {  
        $data['cr'] = $data['ci'] / 0.90;  
    } else if ($data['jml_k'] == 5) {  
        $data['cr'] = $data['ci'] / 1.12;  
    } else if ($data['jml_k'] == 6) {  
        $data['cr'] = $data['ci'] / 1.24;  
    } else if ($data['jml_k'] == 7) {  
        $data['cr'] = $data['ci'] / 1.32;  
    } else if ($data['jml_k'] == 8) {  
        $data['cr'] = $data['ci'] / 1.41;  
    } else if ($data['jml_k'] == 9) {  
        $data['cr'] = $data['ci'] / 1.45;  
    } else {  
        $data['cr'] = $data['ci'] / 1.49;  
    }  
  
    return $data['cr'];  
}  
  
function uji_konsistensi_sub($data) {  
  
    $temp_jml = 0;  
  
    for ($i = 1; $i <= $data['jml_k']; $i++) {  
        for ($j = 1; $j <= $data['jml_k']; $j++) {
```

```
$input = $data['bb' . $j . $i];
$temp_jml = $temp_jml + $input;
}

$data['jml_kolom' . $i] = $temp_jml;
$temp_jml = 0;
}

$temp_jml = 1;

for ($i = 1; $i <= $data['jml_k']; $i++) {
    for ($j = 1; $j <= $data['jml_k']; $j++) {

        $input = $data['bb' . $i . $j];
        $temp_jml = $temp_jml * $input;
    }

    $data['jml_kali_baris' . $i] = pow($temp_jml, (1 / $data['jml_k']));
    $temp_jml = 1;
}

$temp_jml = 0;

for ($i = 1; $i <= $data['jml_k']; $i++) {
    $input = $data['jml_kali_baris' . $i];
    $temp_jml = $temp_jml + $input;
}

$data['total_kali_baris'] = $temp_jml;

for ($i = 1; $i <= $data['jml_k']; $i++) {
    $data['bobot_prioritas_sub' . $i] = $data['jml_kali_baris' . $i] /
    $data['total_kali_baris'];
}

for ($i = 1; $i <= $data['jml_k']; $i++) {
    for ($j = 1; $j <= $data['jml_k']; $j++) {
```

```
        $data['cell' . $i . $j] = $data['bb' . $i . $j] / $data['jml_kolom' . $j];
    }
}

$stemp_jml = 0;

for ($i = 1; $i <= $data['jml_k']; $i++) {
    for ($j = 1; $j <= $data['jml_k']; $j++) {

        $input = $data['cell' . $i . $j];
        $stemp_jml = $stemp_jml + $input;
    }

    $data['bobot_sintesa_sub' . $i] = $stemp_jml;
    $stemp_jml = 0;
}

for ($i = 1; $i <= $data['jml_k']; $i++) {

    $data['sin_div_pri' . $i] = $data['bobot_sintesa_sub' . $i] / $data['bobot_prioritas_sub'
    . $i];
    $stemp_jml = $stemp_jml + $data['sin_div_pri' . $i];
}

$data['total_div'] = $stemp_jml;

$data['eigen_max_sub'] = $data['total_div'] / $data['jml_k'];

$data['ci_sub'] = ($data['eigen_max_sub'] - $data['jml_k']) / ($data['jml_k'] - 1);

/* $data['ir1'] = 0.00;
   $data['ir2'] = 0.00;
   $data['ir3'] = 0.58;
   $data['ir4'] = 0.90;
   $data['ir5'] = 1.12;
   $data['ir6'] = 1.24;
   $data['ir7'] = 1.32;
   $data['ir8'] = 1.41;
   $data['ir9'] = 1.45;
```

```
$data['ir10'] = 1.49; */

if ($data['jml_k'] == 1) {
    $data['cr_sub'] = 0;
} else if ($data['jml_k'] == 2) {
    $data['cr_sub'] = 0;
} else if ($data['jml_k'] == 3) {
    $data['cr_sub'] = $data['ci_sub'] / 0.58;
} else if ($data['jml_k'] == 4) {
    $data['cr_sub'] = $data['ci_sub'] / 0.90;
} else if ($data['jml_k'] == 5) {
    $data['cr_sub'] = $data['ci_sub'] / 1.12;
} else if ($data['jml_k'] == 6) {
    $data['cr_sub'] = $data['ci_sub'] / 1.24;
} else if ($data['jml_k'] == 7) {
    $data['cr_sub'] = $data['ci_sub'] / 1.32;
} else if ($data['jml_k'] == 8) {
    $data['cr_sub'] = $data['ci_sub'] / 1.41;
} else if ($data['jml_k'] == 9) {
    $data['cr_sub'] = $data['ci_sub'] / 1.45;
} else {
    $data['cr_sub'] = $data['ci_sub'] / 1.49;
}

return $data;
}

function eigen_max_sub() {

    $query = $this->db->query("SELECT * FROM sub_kriteria");

    $total = 0;
    $jml_k = $query->num_rows();
    //echo $jml_k;
    foreach ($query->result_array() as $row) {
        $total += $row['sintesa_sub'] / $row['prioritas_sub'];
    }
    //echo $total;
    $eigen_max_sub = $total / $jml_k;
```

```
//echo $eigen_max_sub;

return $eigen_max_sub;
}

function ci_sub() {

    $query = $this->db->query("SELECT * FROM sub_kriteria");
    $jml_k = $query->num_rows();
    $eigen_max_sub = $this->eigen_max_sub();

    $ci_sub = ($eigen_max_sub - $jml_k) / ($jml_k - 1);
    //$ci = number_format($ci, 2, '.', '');

    return $ci_sub;
}

function cr_sub() {

    $query = $this->db->query("SELECT * FROM sub_kriteria");
    $data['jml_k'] = $query->num_rows();

    $data['ci_sub'] = $this->ci_sub();

    if ($data['jml_k'] == 1) {
        $data['cr_sub'] = 0;
    } else if ($data['jml_k'] == 2) {
        $data['cr_sub'] = 0;
    } else if ($data['jml_k'] == 3) {
        $data['cr_sub'] = $data['ci_sub'] / 0.58;
    } else if ($data['jml_k'] == 4) {
        $data['cr_sub'] = $data['ci_sub'] / 0.90;
    } else if ($data['jml_k'] == 5) {
        $data['cr_sub'] = $data['ci_sub'] / 1.12;
    } else if ($data['jml_k'] == 6) {
        $data['cr_sub'] = $data['ci_sub'] / 1.24;
    } else if ($data['jml_k'] == 7) {
        $data['cr_sub'] = $data['ci_sub'] / 1.32;
    } else if ($data['jml_k'] == 8) {
```

```

        $data['cr_sub'] = $data['ci_sub'] / 1.41;
    } else if ($data['jml_k'] == 9) {
        $data['cr_sub'] = $data['ci_sub'] / 1.45;
    } else {
        $data['cr_sub'] = $data['ci_sub'] / 1.49;
    }
    return $data['cr_sub'];
}

function get_data_nilai() {

    $query = $this->db->query("SELECT k.kecamatan, d.desa, d.nilai_ahp FROM
        data_kec k, datadesa d where k.id_kec=d.id_kec");
    return $query;
}

function simpan_data_bobot_kriteria($data) {
    return $this->db->insert('matrik_kriteria', $data);
}

function simpan_data_bobot_subkriteria($data) {
    return $this->db->insert('matrik_subkriteria', $data);
}

function simpan_perhitungan_ahp($bobot) {
    $query = $this->db->query("UPDATE kriteria SET
        prioritas_ahp = '$bobot[bobot_prioritas]',
        sintesa_ahp = '$bobot[bobot_sintesa]'
        WHERE id_kriteria = '$bobot[id_kriteria]'"
    );

    return $query;
}

function simpan_perhitungan_sub($bobot_sub) {
    $query = $this->db->query("UPDATE sub_kriteria SET
        prioritas_sub = '$bobot_sub[bobot_prioritas_sub]',
        sintesa_sub = '$bobot_sub[bobot_sintesa_sub]'
    ");
}

```

```

        WHERE id_subkriteria = '$bobot_sub[id_subkriteria]'
    );

    return $query;
}

function cek_nilai_sub() {
    $query = $this->db->query("SELECT distinct k.id_kriteria, k.nama_kriteria,
(k.prioritas_ahp*100)::integer as prioritas
    from kriteria k, sub_kriteria s
    WHERE k.id_kriteria NOT IN (SELECT distinct id_kriteria from sub_kriteria)");
    return $query;
}

function simpan_nilai($data) {
    $query = $this->db->query("UPDATE datadesa SET
        nilai_ahp = ($data[nilai_total]::integer)
        WHERE id_desa = '$data[id_desa]'");
    return $query;
}

function update_laporan($nilai_total,$kode) {
    $query = $this->db->query("UPDATE laporan
        SET nilai_total= ($nilai_total::integer)
        WHERE kode = '$kode'");

    return $query;
}

function update_interval() {
    $select_strata = $this->db->query('select * from strata');
    $i = 1;
    $locika = FALSE;
    foreach ($select_strata->result() as $row) {
        //echo $i;
        if ($this->db->query("update strata set
            interval_atas = (select interval" . $i . " from interval),
            interval_bawah = (select interval" . ($i + 1) . " from interval)
            where id_strata = " . $row->id_strata . "")) {

```



```

        $locika = TRUE;
    } else {
        $locika = FALSE;
        break;
    }
    $i++;
}
return $locika;
}

function update_idstrata2($id_desa) {

$a=$this->db->query("select nilai_ahp from datadesa where id_desa='$id_desa'");
$i = 1;
foreach ($a->result() as $row) {
    $bobot = $row->nilai_ahp;

    $b = $this->db->query('select * from interval');
    $hem = $b->result_array();

    if ($hem[0]['interval1'] >= $bobot && $bobot >= $hem[0]['interval2']) {
        $query = $this->db->query('update datadesa set "id_strata" = 1 where id_desa=' .
$id_desa);
        // echo 'interval 1';
    } else if ($hem[0]['interval2'] > $bobot && $bobot >= $hem[0]['interval3']) {
        $query = $this->db->query('update datadesa set "id_strata" = 2 where id_desa=' .
$id_desa);
        // echo 'interval 2';
    } else if ($hem[0]['interval3'] > $bobot && $bobot >= $hem[0]['interval4']) {
        $query = $this->db->query('update datadesa set "id_strata" = 3 where id_desa=' .
$id_desa);
        // echo 'interval 3';
    } else if ($hem[0]['interval4'] > $bobot && $bobot >= $hem[0]['interval5']) {
        $query = $this->db->query('update datadesa set "id_strata" = 4 where id_desa=' .
$id_desa);
    } else {
    }
}
}
}

```

```

function update_idstrata3() {

$a=$this->db->query('select nilai_total from laporan');
$i = 1;
  foreach ($a->result() as $row) {
    $bobot = $row->nilai_total;

    $b = $this->db->query('select * from interval');
    $hem = $b->result_array();

    if ($hem[0]['interval1'] >= $bobot && $bobot >= $hem[0]['interval2']) {
      $query = $this->db->query('update laporan set "id_strata" = 1 where nilai_total=' .
    $bobot);
      // echo 'interval 1';
    } else if ($hem[0]['interval2'] > $bobot && $bobot >= $hem[0]['interval3']) {
      $query = $this->db->query('update laporan set "id_strata" = 2 where nilai_total=' .
    $bobot);
      // echo 'interval 2';
    } else if ($hem[0]['interval3'] > $bobot && $bobot >= $hem[0]['interval4']) {
      $query = $this->db->query('update laporan set "id_strata" = 3 where nilai_total=' .
    $bobot);
      // echo 'interval 3';
    } else if ($hem[0]['interval4'] > $bobot && $bobot >= $hem[0]['interval5']) {
      $query = $this->db->query('update laporan set "id_strata" = 4 where nilai_total=' .
    $bobot);
      // echo 'interval 4';
    } else {
      // $query = $this->db->query('update datadesa set "id_strata" = 5 where nilai_ahp='
    . $bobot);
      //echo 'bingung';
    }
  }
}

function update_idstrata($kode) {

$a=$this->db->query("select nilai_total from laporan where kode='$kode'");
$i = 1;
  foreach ($a->result() as $row) {
    $bobot = $row->nilai_total;

```

```
$b = $this->db->query('select * from interval');
$sehem = $b->result_array();

if ($sehem[0]['interval1'] >= $bobot && $bobot >= $sehem[0]['interval2']) {
    $query = $this->db->query("update laporan set id_strata = 1 where kode='$kode'");
    // echo 'interval 1';
} else if ($sehem[0]['interval2'] > $bobot && $bobot >= $sehem[0]['interval3']) {
    $query = $this->db->query("update laporan set id_strata = 2 where
kode='$kode'");
    // echo 'interval 2';
} else if ($sehem[0]['interval3'] > $bobot && $bobot >= $sehem[0]['interval4']) {
    $query = $this->db->query("update laporan set id_strata = 3 where
kode='$kode'");
    // echo 'interval 3';
} else if ($sehem[0]['interval4'] > $bobot && $bobot >= $sehem[0]['interval5']) {
    $query = $this->db->query("update laporan set id_strata = 4 where
kode='$kode'");
    // echo 'interval 4';
} else {
}
}
}
?>
```

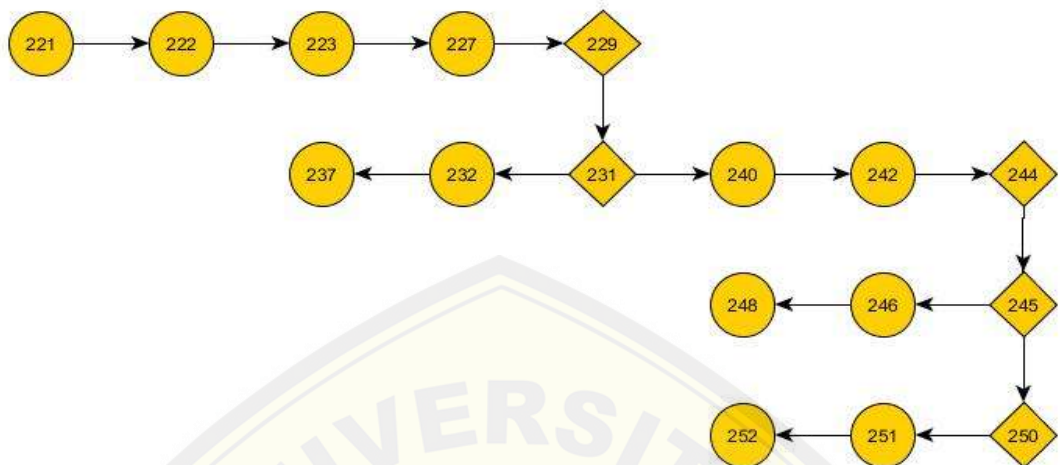
LAMPIRAN G Pengujian *White Box*G.1 *Increment* Proses Penentuan Strata Desa Siaga

1. Insert Desa

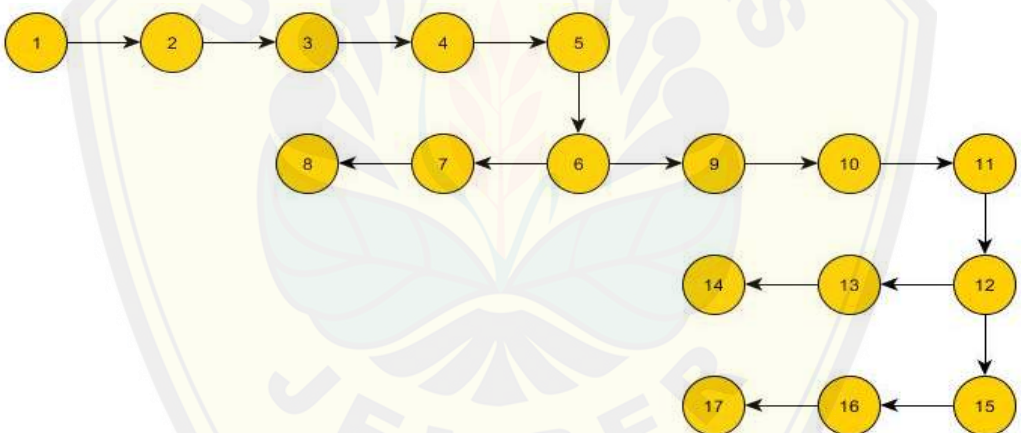
A. Listing Program

```
221 public function desa() {
222     //     $this->cek();
223     // $input['id_desa'] = $this->m_crud->get_id_desa();
224     $input['dt_desa'] = $this->m_crud->getSelectedData("admin_sempel_kec", array());
225     $input['kec'] = $this->m_crud->getSelectedData("data_kec", array());
226     $input['wilayah'] = $this->m_crud->getSelectedData("puskesmas", array());
227     $input['desa'] = $this->m_crud->get_data_desa();
228     $this->page('v_datadesa', $input);
229 }
230
231 public function data_desa() {
232     //     $this->cek();
233     if ($this->input->get_post('id_desa')) {
234         $input = array('id_desa' => $this->input->post('id_desa'),
235             'nama_desa' => $this->input->post('nama_desa'),
236             'alamat_desa' => $this->input->post('alamat_desa'),
237             'id_pkm' => $this->input->post('id_pkm'),
238             'id_kec' => $this->input->post('id_kec'),
239         );
240
241
242         $i = $this->input->post("id_desa");
243
244         $id['id_desa'] = $this->input->post('id_desa');
245         $hasil = $this->m_crud->getSelectedData("datadesa", $id);
246         $row = $hasil->num_rows();
247         if ($row > 0) {
248             $this->m_crud->updateData("datadesa", $input, $id);
249             $this->history("edit data desa $i");
250             redirect(base_url("index.php/c_fronted/desa"));
```

B. Diagram Alir



C. Grafik Alir



D. Cyclomatic Complexity

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

$$V(G) = 17 - 16 + 2$$

$$V(G) = 3$$

E. Jalur Independen

- Jalur 1 = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8
- Jalur 2 = 1 - 2 - 3 - 4 - 5 - 6 - 9 - 10 - 11 - 12 - 13 - 14
- Jalur 3 = 1 - 2 - 3 - 4 - 5 - 6 - 9 - 10 - 11 - 12 - 15 - 16 - 17

F. Test Case

No	Pengujian	Jalur yang Diharapkan	Jalur Hasil Pengamatan	Ket.
1.	Menampilkan form untuk menambah data desa baru	1 – 2 – 3 – 4 –	1 – 2 – 3 – 4 – 5 –	[√] berhasil
		5 – 6 – 7 – 8	6 – 7 – 8	[] gagal
2.	Menekan tombol batal pada form data desa baru	1 – 2 – 3 – 4 –	1 – 2 – 3 – 4 – 5 –	[√] berhasil
		5 – 6 – 9 – 10	6 – 9 – 10 – 11 –	[] gagal
		– 11 – 12 – 13 – 14	12 – 13 – 14	
3.	Menekan tombol simpan pada form data desa baru dan menyimpan data pada database	1 – 2 – 3 – 4 –	1 – 2 – 3 – 4 – 5 –	[√] berhasil
		5 – 6 – 9 – 10	6 – 9 – 10 – 11 –	[] gagal
		– 11 – 12 – 15 – 16 – 17	12 – 15 – 16 – 17	

2. Insert Puskesmas

A. Listing Program

```

public function puskesmas() {
//    $this->cek();
    $input['data_pkm'] = $this->m_crud->get_data_pkm();
    $input['idpkm'] = $this->m_crud->get_id_pkm();
    $input['kec'] = $this->m_crud->getSelectedData("data_kec", array());
    $input['jenispkm'] = $this->m_crud->getSelectedData("jenis_pkm", array());
    $this->page('v_puskesmas', $input);
}
public function pkm() {
//    $this->cek();
    if ($this->input->get_post('id_pkm')) {
        $input = array('id_pkm' => $this->input->post('id_pkm'),
            'nama_pkm' => $this->input->post('nama_pkm'),
            'id_kec' => $this->input->post('id_kec'),
            'alamat_pkm' => $this->input->post('alamat_pkm'),
            'id_jenis' => $this->input->post('id_jenis'),
            'jml_desa' => $this->input->post('jml_desa'),
            'jml_dokter' => $this->input->post('jml_dokter'),
            'jml_bidan' => $this->input->post('jml_bidan'),
    
```

```

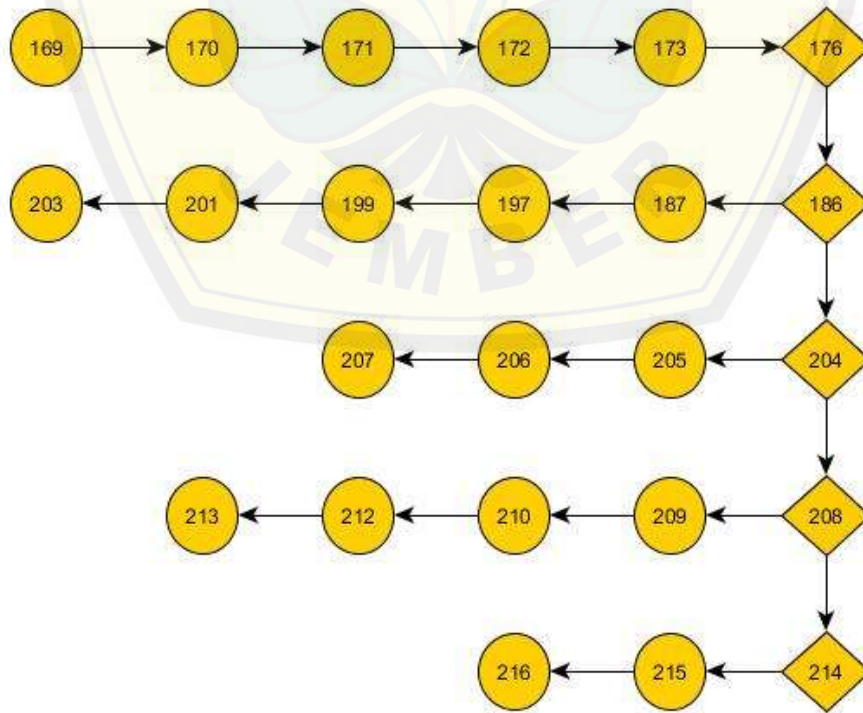
'jml_perawat' => $this->input->post('jml_perawat'),
'jml_poskesdes' => $this->input->post('jml_poskesdes'),
'jml_posyandu' => $this->input->post('jml_posyandu'));

$si = $this->input->post("id_pkm");
$did['id_pkm'] = $this->input->post('id_pkm');
$hasil = $this->m_crud->getSelectedData("puskesmas", $did);
$rowse = $hasil->num_rows();
if ($rowse > 0) {
    $this->m_crud->updateData("puskesmas", $input, $did);
    $this->history("edit data puskesmas $si");
    redirect(base_url("index.php/c_froned/puskesmas"));
} else {
    $this->m_crud->insertData("puskesmas", $input);
    $this->history("insert data puskesmas $si");

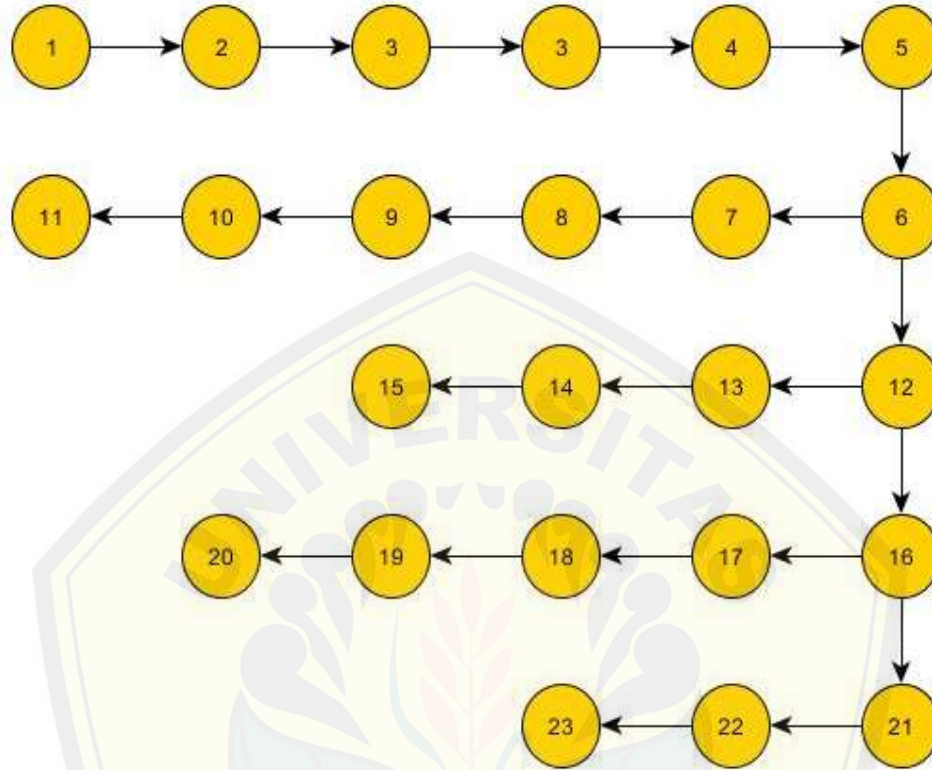
    redirect(base_url("index.php/c_froned/puskesmas"));
}
} else {
    redirect(base_url("index.php/c_froned/puskesmas"));
}
}

```

B. Diagram Alir



C. Grafik Alir



D. Cyclomatic Complexity

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

$$V(G) = 23 - 23 + 2$$

$$V(G) = 2$$

E. Jalur Independen

Jalur 1 = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11

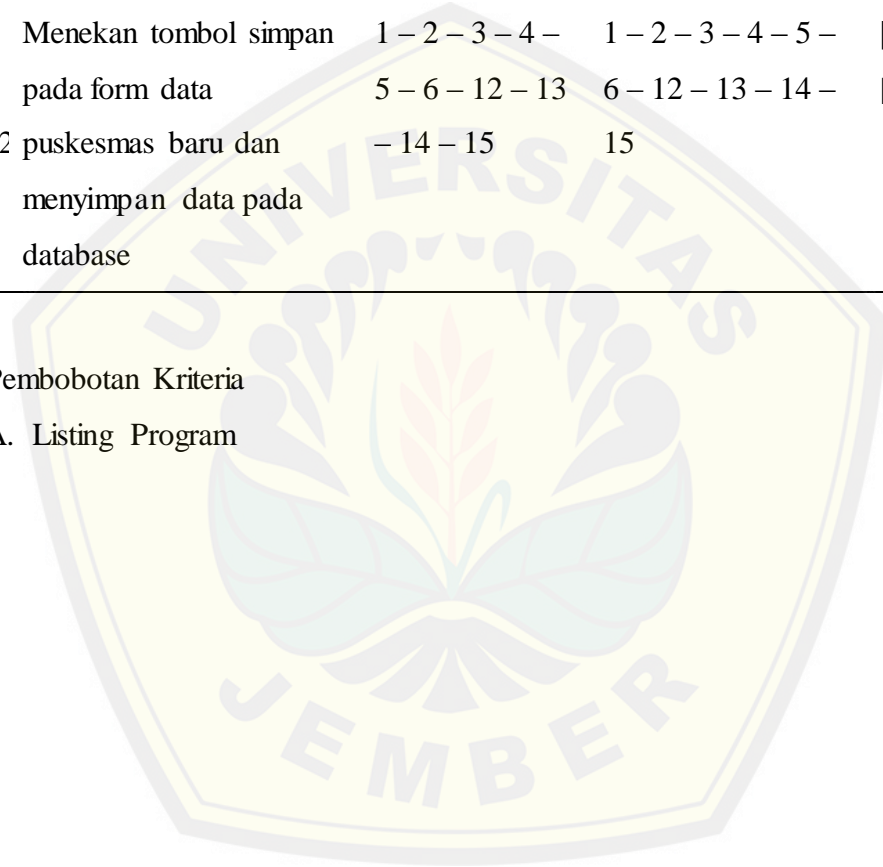
Jalur 2 = 1 - 2 - 3 - 4 - 5 - 6 - 12 - 13 - 14 - 15

F. Test Case

No	Pengujian	Jalur yang Diharapkan	Jalur Hasil Pengamatan	Ket.
1.	Menampilkan form untuk menambah data puskesmas baru	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11	<input checked="" type="checkbox"/> berhasil <input type="checkbox"/> gagal
	Menekan tombol simpan pada form data puskesmas baru dan menyimpan data pada database	1 – 2 – 3 – 4 – 5 – 6 – 12 – 13 – 14 – 15	1 – 2 – 3 – 4 – 5 – 6 – 12 – 13 – 14 – 15	<input checked="" type="checkbox"/> berhasil <input type="checkbox"/> gagal

3. Pembobotan Kriteria

A. Listing Program



```

function tb_kriteria() {
//     $this->cek();
    $input['data_kriteria'] = $this->m_admin->get_data_kriteria();
    $input['data_bobot_kriteria'] = $this->m_admin->get_matrik_kriteria();
    $this->load->view('header');
    $this->load->view('menu');
    $this->load->view('v_kriteria', $input);
    $this->load->view('footer');
}

function pembobotan() {
//     $this->cek();
    $cek = $this->m_admin->get_matrik_kriteria();
    if ($cek->num_rows() > 0) {
        $this->tb_bobot_ahp();
    } else {
        $this->tb_ahp();
    }
}

function simpan_bobot_kriteria() {
    $data_kriteria = $this->m_admin->get_kriteria();
    $jml_k = $data_kriteria->num_rows();

    for ($i = 1; $i <= $jml_k; $i++) {
        for ($j = 1; $j <= $jml_k; $j++) {
            $input['bb' . $i . $j] = $this->input->post('k' . $i . $j);
        }
        $input['id' . $i] = $this->input->post('id' . $i);
    }
    $input['jml_k'] = $jml_k;
    $data3 = $this->m_admin->uji_konsistensi($input);

    if ($data3['cr'] < 0.1) {
//         $logic1 = $this->m_admin->hapus_data_bobot_kriteria()

        for ($i = 1; $i <= $jml_k; $i++) {
            for ($j = 1; $j <= $jml_k; $j++) {
                $data['id_k1'] = $this->input->post('id' . $i);
                $data['id_k2'] = $this->input->post('id' . $j);
                $data2['id_kr1'] = $this->input->post('id' . $i);
                $data2['id_kr2'] = $this->input->post('id' . $j);
                $data['bobot'] = $this->input->post('k' . $i . $j);

                $data = array(
                    'id_matrik' => $this->m_admin->get_id_matrik(),
                    'id_kriteria_baris' => $data['id_k1'],
                    'id_kriteria_kolom' => $data['id_k2'],
                    'bobot' => $data['bobot'],
                );
                $logic2 = $this->m_admin->simpan_data_bobot_kriteria($data);
            }
        }
    }
}

```

```
    }
  }
  $data_kriteria = $this->m_admin->get_kriteria();

  $i = 1;

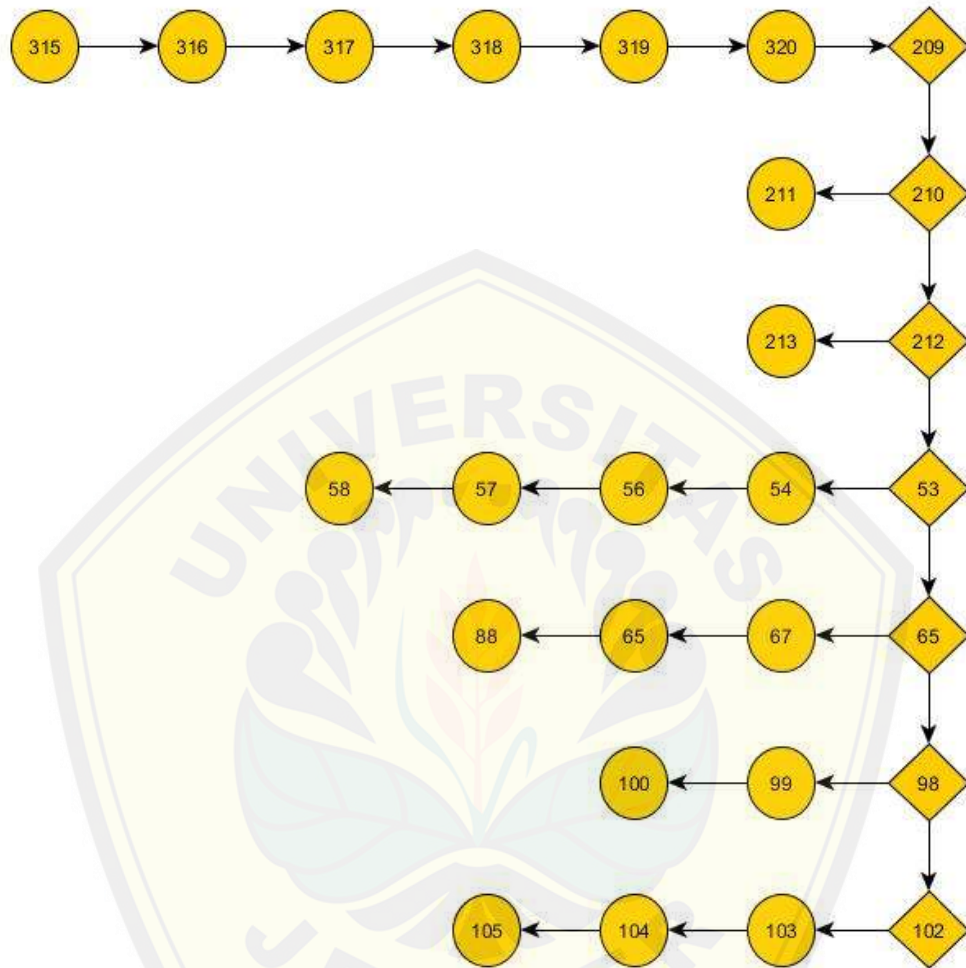
  foreach ($data_kriteria->result_array() as $row) {

    $bobot['id_kriteria'] = $row['id_kriteria'];
    $bobot['bobot_prioritas'] = $data3['bobot_prioritas' . $i];
    $bobot['bobot_sintesa'] = $data3['bobot_sintesa' . $i];

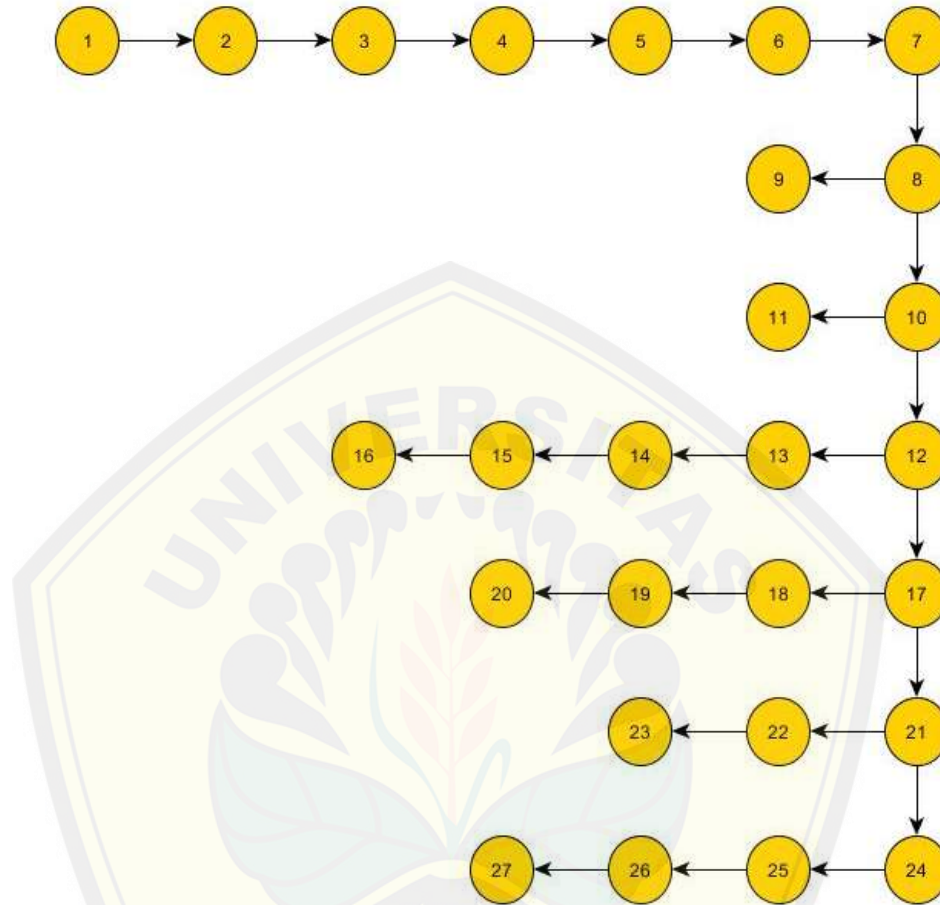
    $this->m_admin->simpan_perhitungan_ahp($bobot);

    $i++;
  }
  if ($logic2==true) {
    $this->m_admin->update_interval();
    redirect('index.php/c_ahp/tb_bobot_ahp/');
  }
  else{
    $this->session->set_flashdata('message','Pembobotan GAGAL! Karena Matriks
    Tidak Konsisten,Lakukan Pembobotan Ulang dengan Benar!!!');
    echo 'gagal coooyyy';
    //redirect('index.php/c_ahp/tb_bobot_ahp/');
  }
}
```

B. Diagram Alir



C. Grafik Alir



D. Cyclomatic Complexity

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

$$V(G) = 27 - 26 + 2$$

$$V(G) = 3$$

E. Jalur Independen

Jalur 1 = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 12 - 13 - 14 - 15 - 16

Jalur 2 = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 12 - 17 - 18 - 19 - 20

Jalur 3 = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 12 - 17 - 21 - 24 - 25 - 26 - 27

F. Test Case

No	Pengujian	Jalur yang Diharapkan	Jalur Hasil Pengamatan	Ket.
1.	Menampilkan form untuk menambahkan bobot	1 – 2 – 3 – 4 –	1 – 2 – 3 – 4 – 5 –	[√] berhasil
		5 – 6 – 7 – 8 –	6 – 7 – 8 – 9 – 10	
		9 – 10 – 12 –	– 12 – 13 – 14 –	[] gagal
		13 – 14 – 15 –	15 – 16	
2.	Menekan tombol proses pada tabel pembobotan	16	1 – 2 – 3 – 4 –	[√] berhasil
		1 – 2 – 3 – 4 –	1 – 2 – 3 – 4 – 5 –	
		5 – 6 – 7 – 8 –	6 – 7 – 8 – 9 – 10	[] gagal
		9 – 10 – 12 –	– 12 – 17 – 18 –	
3	Memproses perhitungan bobot baru	17 – 18 – 19 –	19 – 20	[√] berhasil
		20	1 – 2 – 3 – 4 –	
		1 – 2 – 3 – 4 –	1 – 2 – 3 – 4 – 5 –	[] gagal
		5 – 6 – 7 – 8 –	6 – 7 – 8 – 9 – 10	
9 – 10 – 12 –	– 12 – 17 – 21 –			
17 – 21 – 24 –	24 – 25 – 26 – 27			
25 – 26 – 27				

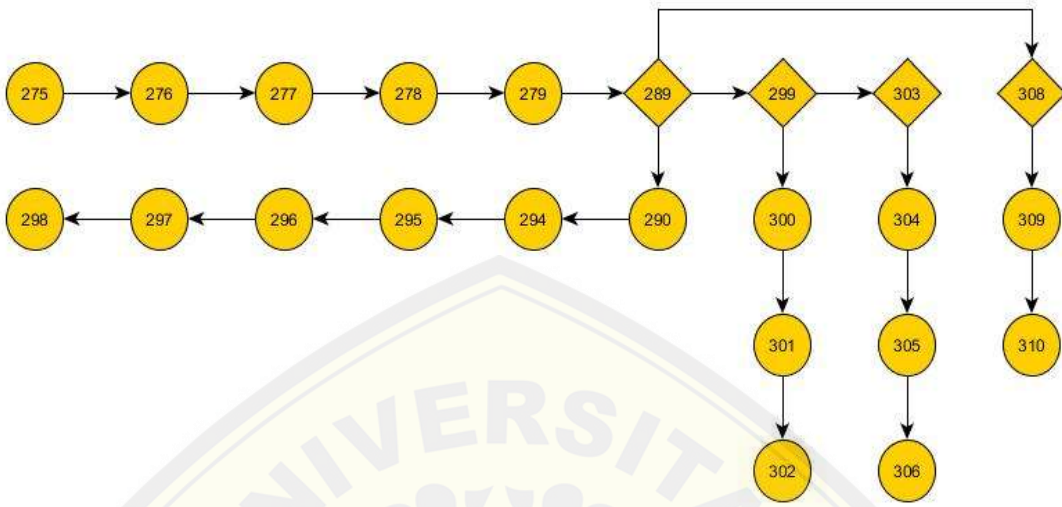
G.2 Increment Manajemen User

1. Menambah Data User

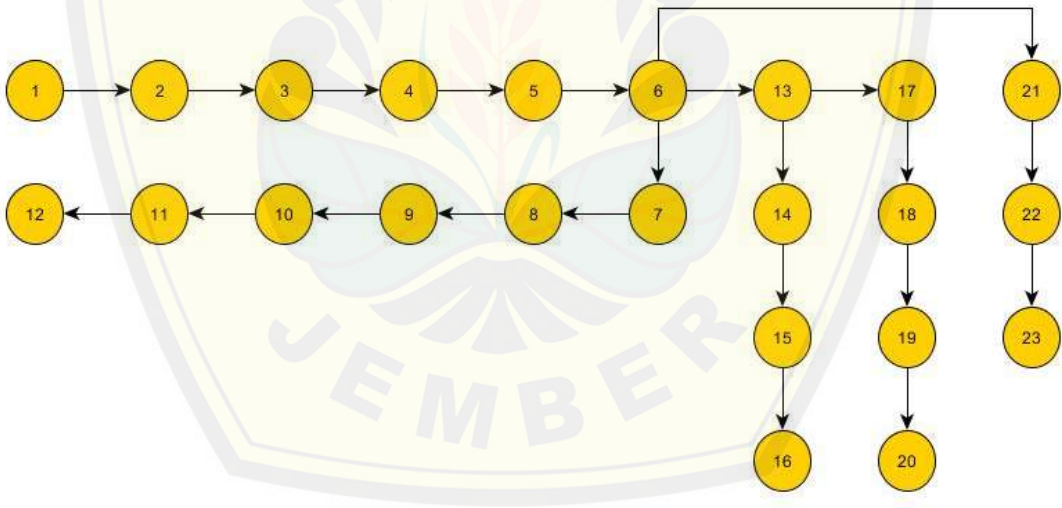
A. Listing Program

```
public function user() {
    $input['id_user'] = $this->m_crud->get_id_user();
    $input['user'] = $this->m_crud->get_data_user();
    $input['ket'] = $this->m_crud->getSelectedData("public.ket_level", array());
    $this->page('v_user', $input);
}
public function data_user() {
    if ($this->input->get_post('id_user')) {
        $input = array('id_user' => $this->input->post('id_user'),
            'username' => $this->input->post('username'),
            'password' => $this->input->post('password'),
            'level' => $this->input->post('level')
        );
        $i = $this->input->post("id_user");
        $id['id_user'] = $this->input->post('id_user');
        $hasil = $this->m_crud->getSelectedData("user", $id);
        $row = $hasil->num_rows();
        if ($row > 0) {
            $this->m_crud->updateData("user", $input, $id);
            $this->history("edit data user $i");
            redirect(base_url("index.php/c_fronted/user"));
        } else {
            $this->m_crud->insertData("user", $input);
            $this->history("insert data user $i");
            redirect(base_url("index.php/c_fronted/user"));
        }
    } else {
        redirect(base_url("index.php/c_fronted/user"));
    }
}
```

B. Diagram Alir



C. Grafik Alir



D. Cyclomatic Complexity

$$V(G) = E - N + 2$$
$$V(G) = 23 - 22 + 2$$
$$V(G) = 3$$

E. Jalur Independen

Jalur 1 = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11 - 12

Jalur 2 = 1 - 2 - 3 - 4 - 5 - 6 - 13 - 14 - 15 - 16

Jalur 3 = 1 - 2 - 3 - 4 - 5 - 6 - 21 - 22 - 23

F. Test Case

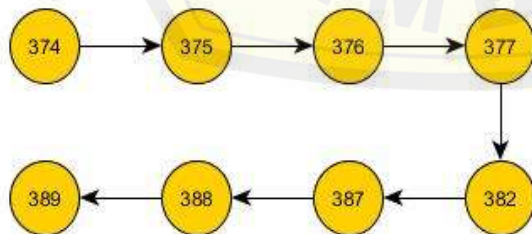
No	Pengujian	Jalur yang Diharapkan	Jalur Hasil Pengamatan	Ket.
1.	Menampilkan form untuk menambahkan data user baru	1 - 2 - 3 - 4 -	1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11 - 12	[√] berhasil [] gagal
		5 - 6 - 7 - 8 -		
		9 - 10 - 11 - 12		
2.	Menekan tombol simpan	1 - 2 - 3 - 4 -	1 - 2 - 3 - 4 - 5 - 6 - 13 - 14 - 15 - - 15 - 16 16	[√] berhasil [] gagal
		5 - 6 - 13 - 14		
		- 15 - 16		
3.	Menekan tombol batal	1 - 2 - 3 - 4 -	1 - 2 - 3 - 4 - 5 - 6 - 21 - 22 - 23 - 23	[√] berhasil [] gagal
		5 - 6 - 21 - 22		
		- 23		

3. History

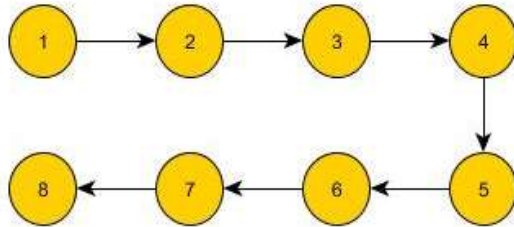
A. Listing Program

```
public function histori() {  
    //    $this->cek();  
    $input['no'] = $this->m_crud->No_Urut('data_histori', "id_histori");  
    $input['id_user'] = $this->m_crud->get_id_user();  
    $input['data_histori'] = $this->m_crud->get_data_histori();  
    $this->page('v_history', $input);  
}  
  
public function history($aktivitas) {  
  
    $input = array('id_histori' => $this->m_crud->No(),  
        'username' => $this->session->userdata('username'),  
        'aktivitas' => $aktivitas,  
        'tanggal' => date('Ymd'),  
        'jam' => date('His')  
    );  
    $this->m_crud->insertData('data_histori', $input);  
}
```

B. Diagram Alir



C. Grafik Alir



D. Cyclomatic Complexity

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

$$V(G) = 7 - 8 + 2$$

$$V(G) = 3$$

E. Jalur Independen

Jalur 1 = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8

F. Test Case

No	Pengujian	Jalur yang Diharapkan	Jalur Hasil Pengamatan	Ket.
1.	Menampilkan form tabel history aktifitas penggunaan sistem	1 - 2 - 3 - 4 - 5 - 6 - 7 - 8	1 - 2 - 3 - 4 - 5 - 6 - 7 - 8	[√] berhasil [] gagal

LAMPIRAN H Pengujian *Black Box*

NO	MENU	SUBMENU	FUNGSI	KASUS	HASIL	KET
1	Administrasi	Data Puskesmas	Menu ini digunakan untuk insert, update dan delete data puskesmas	<p>Ketika user mencari data puskesmas dan tidak ada data yang tersimpan</p> <p>Ketika user mencari data puskesmas dan ada data puskesmas yang tersimpan</p> <p>Ketika user memilih button tambah data</p> <p>Ketika user menyimpan data puskesmas yang telah diinputkan dan datanya kurang lengkap atau tidak sesuai</p> <p>Ketika user klik button batal pada form tambah data</p> <p>Ketika user memilih button update</p>	<p>Menampilkan tabel yang kosong dengan pemberitahuan "Maaf, tidak ada data yang ditemukan"</p> <p>Menampilkan list data puskesmas dalam bentuk tabel</p> <p>Menampilkan form tambah data puskesmas</p> <p>Menampilkan peringatan di bawah kolom yang belum diisi "harus diisi" dan atau "hanya boleh angka dan huruf" dan atau "harus diisi angka"</p> <p>Menampilkan halaman data puskesmas</p> <p>Menampilkan form update data puskesmas sesuai data yang dipilih</p>	<p>Ok</p> <p>Ok</p> <p>Ok</p> <p>Ok</p> <p>Ok</p> <p>Ok</p>

Dilanjutkan...

NO	MENU	SUBMENU	FUNGSI	KASUS	HASIL	KET
				Ketika user menyimpan data puskesmas yang telah diupdate dan datanya kurang lengkap atau tidak sesuai	Menampilkan peringatan di bawah kolom yang belum diisi "harus diisi" dan atau "hanya boleh angka dan huruf" dan atau "harus diisi angka"	Ok
				Ketika user klik button cancel pada form update data	Menampilkan halaman data puskesmas	Ok
				Ketika user memilih button delete	Menampilkan konfirmasi hapus	Ok
				Ketika user memilih button ok pada konfirmasi hapus	Menghapus data yang dipilih dan kembali ke halaman data puskesmas	Ok
				Ketika user memilih button cancel	Kembali ke halaman data puskesmas	Ok
		Data Desa	Menu ini digunakan untuk insert dan update data desa	Ketika user mencari data desa dan tidak ada data yang tersimpan	Menampilkan tabel yang kosong dengan pemberitahuan "Maaf, tidak ada data yang ditemukan"	Ok
				Ketika user mencari data desa dan ada data desa yang tersimpan	Menampilkan list data desa dalam bentuk tabel	Ok

Dilanjutkan...

NO	MENU	SUBMENU	FUNGSI	KASUS	HASIL	KET
				Ketika user memilih button tambah data	Menampilkan form tambah data desa	Ok
				Ketika user menyimpan data desa yang telah diinputkan dan datanya kurang lengkap atau tidak sesuai	Menampilkan peringatan di bawah kolom yang belum diisi "harus diisi" dan atau "hanya boleh angka dan huruf" dan atau "harus diisi angka"	Ok
				Ketika user klik button batal pada form tambah data	Menampilkan halaman data desa	Ok
				Ketika user memilih button update	Menampilkan form update data desa sesuai data yang dipilih	Ok
				Ketika user menyimpan data desa yang telah diupdate dan datanya kurang lengkap atau tidak sesuai	Menampilkan peringatan di bawah kolom yang belum diisi "harus diisi" dan atau "hanya boleh angka dan huruf" dan atau "harus diisi angka"	Ok
				Ketika user klik button cancel pada form update data	Menampilkan halaman data desa	Ok
2	Kriteria		User melihat data kriteria	Ketika user memilih menu kriteria	Menampilkan tabel data kriteria	Ok

Dilanjutkan...

NO	MENU	SUBMENU	FUNGSI	KASUS	HASIL	KET
			User melakukan pembobotan data kriteria	Ketika user klik botton pembobotan	Menampilkan tabel pembobotan kriteria desa siaga	Ok
				Ketika user klik botton ubah	Mengaktifkan cell yang dapat diubah	Ok
				Ketika user klik botton proses	Mengupdate nilai bobot kriteria	Ok
				Ketika user klik botton batal	Menonaktifkan cell yang telah diaktifkan sebelumnya	Ok
3	Subkriteria		Menu ini digunakan untuk melakukan pembobotan subkriteria	Ketika user klik botton tabel	Menampilkan tabel data kriteria	Ok
				Ketika user klik botton subkriteria	Menampilkan tabel data subkriteria	Ok
				Ketika user klik botton pembobotan	Menampilkan tabel pembobotan subkriteria	Ok
				Ketika user klik botton ubah	Mengaktifkan cell yang dapat diubah	Ok
				Ketika user klik botton proses	Mengupdate nilai bobot subkriteria	Ok
				Ketika user klik botton batal	Menonaktifkan cell yang telah diaktifkan sebelumnya	Ok

Dilanjutkan...

NO	MENU	SUBMENU	FUNGSI	KASUS	HASIL	KET
4	Kriteria Strata Desa Siaga		Menu ini digunakan untuk melakukan pembobotan alternative desa siaga	Ketika user klik botton tabel	Menampilkan tabel data subkriteria	Ok
				Ketika user memilih menu kriteria strata desa siaga	Menampilkan tabel kriteria strata desa siaga	Ok
				Ketika user klik botton tambah data	Menampilkan form pentahapan strata desa siaga	Ok
				Ketika user memilih nama puskesmas	Menampilkan nama-nama desa sesuai wilayah kerja puskesmas secara otomatis	Ok
5	Laporan		Menu ini digunakan untuk melihat dan mencetak hasil pentahapan strata desa siaga	Ketika user mengisi form dengan memilih subkriteria sesuai dengan kondisi desa dan klik botton simpan	Menampilkan tabel strata desa siaga	Ok
				Ketika user mencari data desa yang belum dilakukan pentahapan	Menampilkan tabel yang kosong dengan pemberitahuan "Maaf, tidak ada data yang ditemukan"	Ok

Dilanjutkan...

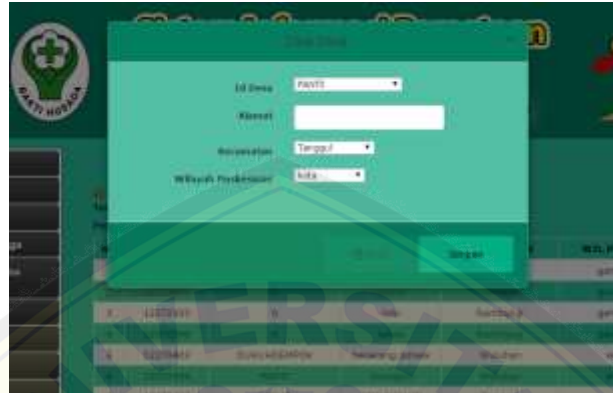
NO	MENU	SUBMENU	FUNGSI	KASUS	HASIL	KET
				Ketika user mencari data desa yang sudah dilakukan pentahapan	Menampilkan data desa dalam tabel	Ok
6	Manajemen User		Menu ini digunakan untuk insert, update dan delete data user	Ketika user klik botton home		
				Ketika admin memilih button tambah data	Menampilkan form tambah data user	Ok
				Ketika admin menyimpan data user yang telah diinputkan dan datanya kurang lengkap	Menampilkan peringatan di bawah kolom yang belum diisi "harus diisi"	Ok
				Ketika admin menginputkan password kurang dari 8 digit	Menampilkan peringatan di bawah kolom "minimal diisi 8 karakter"	Ok
				Ketika admin klik button batal pada form tambah data	Menampilkan halaman data user	Ok
				Ketika admin memilih button update	Menampilkan form update data user sesuai data yang dipilih	Ok
				Ketika admin menyimpan data user yang telah diupdate dan datanya kurang lengkp	Menampilkan peringatan di bawah kolom yang belum diisi "harus diisi"	Ok

NO	MENU	SUBMENU	FUNGSI	KASUS	HASIL	KET
				Ketika admin menginputkan password kurang dari 8 digit	Menampilkan peringatan di bawah kolom "minimal diisi 8 karakter"	Ok
				Ketika admin klik button cancel pada form update data	Menampilkan halaman data user	Ok
				Ketika admin memilih button delete	Menampilkan konfirmasi hapus	Ok
				Ketika admin memilih button ok pada konfirmasi hapus	Menghapus data yang dipilih dan kembali ke halaman data user	Ok
				Ketika admin memilih button cancel	Kembali ke halaman data user	Ok
7	history		Menu ini berfungsi bagi admin untuk dapat mengetahui aktivitas yang dilakukan terhadap sistem baik oleh admin maupun oleh operator	Ketika admin mencari data history yang tidak tersimpan dalam database	Menampilkan tabel kosong dengan pemberitahuan "Maaf, tidak ada data yang ditemukan"	Ok
				Ketika admin mencari data history yang sudah tersimpan dalam database	Menampilkan data history dalam bentuk tabel	Ok

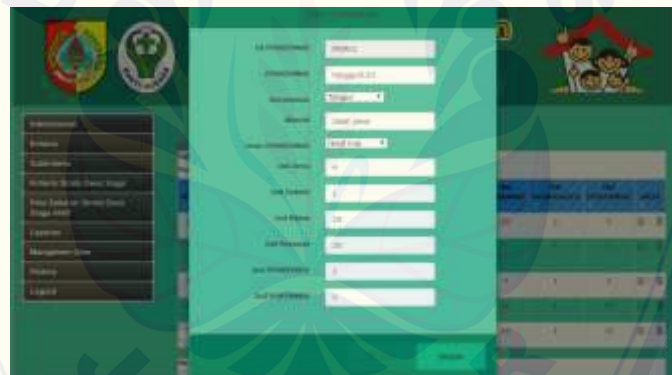
LAMPIRAN I Implementasi Sistem

I.1. *Increment* Proses Penentuan Strata Desa Siaga

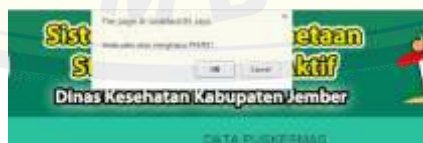
I.1.1. Menambah Data Puskesmas



I.1.2. Mengedit Data Puskesmas



I.1.3. Menghapus Data Puskesmas



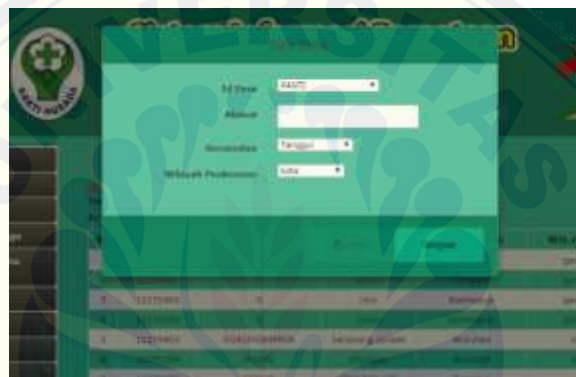
I.1.4. Melihat Data Puskesmas



The screenshot shows the 'DATA PUSKESMAS' section of the system. It features a table with columns for 'No', 'Nama Puskesmas', 'Jumlah Desa', 'Jumlah Desa Siaga Aktif', 'Jumlah Desa Siaga Aktif', 'Jumlah Desa Siaga Aktif', 'Jumlah Desa Siaga Aktif', 'Jumlah Desa Siaga Aktif', 'Jumlah Desa Siaga Aktif', and 'Jumlah Desa Siaga Aktif'. The table contains two rows of data.

No	Nama Puskesmas	Jumlah Desa	Jumlah Desa Siaga Aktif	Jumlah Desa Siaga Aktif	Jumlah Desa Siaga Aktif	Jumlah Desa Siaga Aktif	Jumlah Desa Siaga Aktif	Jumlah Desa Siaga Aktif	Jumlah Desa Siaga Aktif
1	Puskesmas
2	Puskesmas

I.1.5. Menambah Data Desa



The screenshot shows a form for adding a new village. The form fields are: 'No Desa' (dropdown menu), 'Alamat' (text input), 'Kecamatan' (dropdown menu), and 'Wilayah Perekotaan' (dropdown menu). There are 'Simpan' and 'Batal' buttons at the bottom.

I.1.6. Mengedit Data Desa

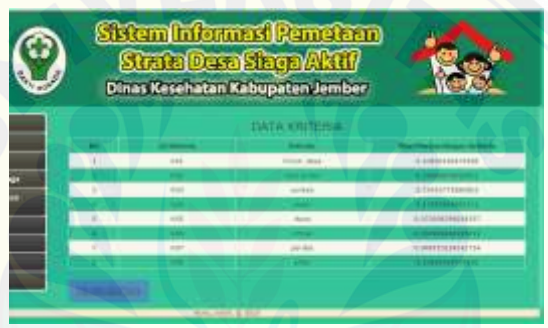


The screenshot shows a form for editing an existing village. The form fields are: 'No Desa' (dropdown menu), 'Alamat' (text input), 'Kecamatan' (dropdown menu), 'Wilayah Perekotaan' (dropdown menu), and 'No Desa' (text input). There are 'Simpan' and 'Batal' buttons at the bottom.

I.1.7. Melihat Data Desa



I.1.8. Melihat Data Kriteria



I.1.9. Pembobotan Kriteria

Pembobotan Kriteria Desa Siaga

ID	Formasi desa	kapas. k. /km	syarat	salin	daur	cahaya	penda	gdbn	VEKTOR PRIORITAS
1	1	1	1	1	1	1	1	1	0.11
2	2	1	3	5	3	5	1	1	0.23
3	2	1	1	5	3	5	1	1	0.20
4	0.13	1	1	5	3	5	1	1	0.17
5	0.2	0.2	0.2	1	1	5	1	1	0.07
6	0.2	0.2	0.2	1	1	1	1	1	0.06
7	0.2	0.2	0.2	0.2	1	1	1	1	0.05
8	1	1	1	1	1	1	1	1	0.11

Prioritas: 1gbr: nilai (0-4) ; k:2003; C:1000; s:100; t:0; D:0403; C:1000; s:100; t:0; D:0403

PROSES UBAH BATAL TABEL

I.1.10. Mengubah Bobot Kriteria

Pembobotan Kriteria Desa Siaga

#	Kriteria Desa	Agus A/Jan	Arhan	Utlan	Desa	Utama	pendes	gila	VEKTOR PRIORITAS
1	Kemampuan Desa	1	1	1	1	1	1	1	0.13
2	Kemampuan Desa	1	1	1	1	1	1	1	0.13
3	Kemampuan Desa	1	1	1	1	1	1	1	0.20
4	Kemampuan Desa	0.20	1	1	1	1	1	1	0.17
5	Kemampuan Desa	0.17	0.20	1	1	1	1	1	0.17
6	Kemampuan Desa	0.17	0.20	0.20	1	1	1	1	0.28
7	Kemampuan Desa	0.17	0.20	0.20	0.20	1	1	1	0.29
8	Kemampuan Desa	0.17	0.20	0.20	0.20	0.20	1	1	0.12

Mengeki Eigen value (Min) : 0.2822 Consistency Index (CI) : 0.0027 Consistency Ratio (CR) : 0.0214

[PROSES](#)
[UBAH](#)
[BATAL](#)
[TAMBAH](#)

I.1.11. Melihat Data Subkriteria

Sistem Informasi Pemetaan Strata Desa Siaga Aktif
Dinas Kesehatan Kabupaten Jember

DATA SUBKRITERIA

#	Subkriteria	Subkriteria	Nilai Pembobotan Subkriteria
1	Salah	Salah	0.404089411026
2	Salah	Salah	0.298070704128
3	Salah	Salah	0.147989295872
4	Salah	Salah	0.150000000000

[PROSES](#)

I.1.12. Pembobotan Subkriteria

Pembobotan Subkriteria

#	Salah	Salah	Salah	Salah	VEKTOR PRIORITAS
Salah	1	1	1	1	0.13
Salah	0.2	1	1	1	0.20
Salah	0.20	0.2	1	1	0.17
Salah	1	0.20	0.20	1	0.28

Mengeki Eigen value (Min) : 0.2425 Consistency Index (CI) : 0.0053 Consistency Ratio (CR) : 0.0248

[PROSES](#)
[UBAH](#)
[BATAL](#)
[TAMBAH](#)

I.1.13. Mengubah Bobot Subkriteria

#	setiap jalan	setiap triwulan	beberapa rutin	ada, belum jalan	VEKTOR PRIORITY
setiap bulan	3	5	3	1	0.43
setiap triwulan	0.2	1	1	2	0.29
beberapa rutin	0.33	0.2	3	3	0.25
ada, belum jalan	1	0.33	1	1	0.13

Prinsip Eigen value (max): 3.8422 Consistency index (CI): 0.0673 Consistency Ratio (CR): 0.2946

Buttons: PROSES, UBAH, BATAL, TABEL

I.1.14. Pembobotan Alternatif Desa Siaga



I.1.15. Mengubah Nilai Pembobotan Alternatif



I.1.16. Melihat Hasil Pembobotan Alternatif



The screenshot displays the 'STRATA DESA SLAGA' interface. At the top, it reads 'Strata Desa Slaga Aktif' and 'Dinas Kesehatan Kabupaten Jember'. Below the title is a table with the following data:

No.	ALTERNATIF	ALTERNATIF	ALTERNATIF	ALTERNATIF	ALTERNATIF
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	0
5	0	0	0	0	0
6	0	0	0	0	0
7	0	0	0	0	0
8	0	0	0	0	0

I.2. Increment Laporan



The screenshot displays the 'Perencanaan Strata Desa Slaga Aktif' interface. It features a table with the following data:

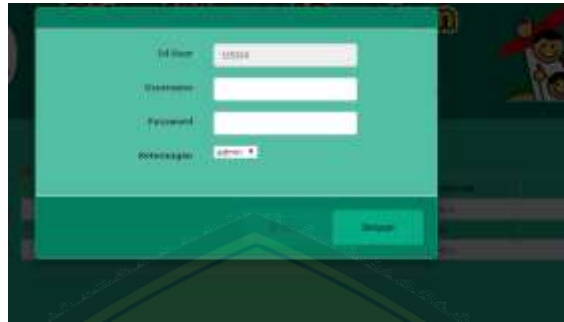
No.	ALTERNATIF	ALTERNATIF	ALTERNATIF	ALTERNATIF	ALTERNATIF	ALTERNATIF
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0

I.3. Increment Pemetaan



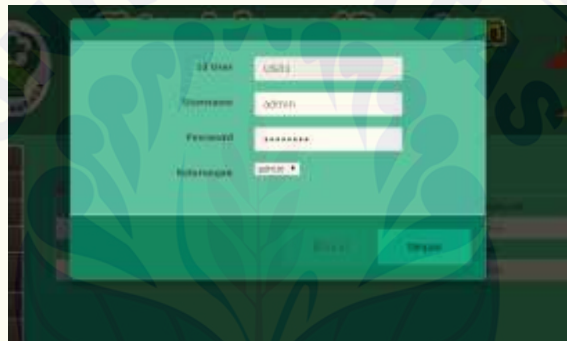
I.4. Increment Manajemen User

1.4.1. Menambah Data User



A screenshot of a user registration form with a teal background. The form contains the following fields: 'Id User' with the value '10001', 'Username' (empty), 'Password' (empty), and 'Kategori' with a dropdown menu showing 'admin'. A 'Simpan' button is located at the bottom right.

1.4.2. Mengubah Data User



A screenshot of a user update form with a teal background. The form contains the following fields: 'Id User' with the value '10001', 'Username' with the value 'admin', 'Password' with masked characters 'XXXXXXXX', and 'Kategori' with a dropdown menu showing 'admin'. A 'Simpan' button is located at the bottom right.

1.4.3. Menghapus Data User



1.4.4. Melihat Data User



A screenshot of the user management interface showing a list of users. The table has the following data:

No	ID USER	USERNAME	KETERANGAN	Aksi
1	10001	admin	admin	[icon]
2	10002	user	user	[icon]
3	10003	super-admin	admin	[icon]

I.5. *Login*

