



**IMPLIMENTASI *SIMPLE MULTI-ATTRIBUTE RATING TECHNIQUE*  
(*SMART*) PADA SISTEM INFORMASI PENGAMBILAN KEPUTUSAN  
PEMILIHAN PESERTA PENDIDIKAN PEMBENTUKAN BINTARA  
REGULER (DIKTUBA REG) PERSONIL TENTARA  
NASIONAL INDONESIA  
(STUDI KASUS: DIVISI INFANTERI 2 KOSTRAD MALANG)**

**SKRIPSI**

Oleh

**Derry Rismawan**

**NIM 112410101074**

**PROGRAM STUDI SISTEM INFORMASI**

**UNIVERSITAS JEMBER**

**2017**



**IMPLIMENTASI *SIMPLE MULTI-ATTRIBUTE RATING TECHNIQUE*  
(*SMART*) PADA SISTEM INFORMASI PENGAMBILAN KEPUTUSAN  
PEMILIHAN PESERTA PENDIDIKAN PEMBENTUKAN BINTARA  
REGULER (DIKTUBA REG) PERSONIL TENTARA  
NASIONAL INDONESIA  
(STUDI KASUS: DIVISI INFANTERI 2 KOSTRAD MALANG)**

**SKRIPSI**

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

Oleh

**Derry Rismawan**

**NIM 112410101074**

**PROGRAM STUDI SISTEM INFORMASI  
UNIVERSITAS JEMBER  
2017**

## PERSEMBAHAN

Skripsi ini saya persembahkan untuk :

1. Allah SWT, yang telah memberikan kelancaran dan kemudahan dalam menyelesaikan skripsi ini;
2. Kedua orang tua penulis, Ayahanda Aris Yuana Eryanto, S.H Ibunda Darmawatin yang selalu mendoakan, menasehati, dan memberikan semangat dalam mengerjakan tugas akhir ini;
3. Kakak Dewi Artin Yuanita, Amd, Kakak Dwi Rismawanto, S.Pd yang selalu memberikan dukungan;
4. Keluarga baru yang ada di Banyuwangi dan Jember yang selalu memberikan semangat, dan memotivasi penulis untuk terus menyelesaikan tugas akhir ini;
5. Guru-guru SDN I Karya Maju, SMPN 2 Sekayu, SMAN I Sekayu hingga Perguruan Tinggi Universitas Jember yang mendidik dan memberikan ilmu yang bermanfaat;
6. Teman-teman dari SDN I Karya Maju, SMPN 2 Sekayu, SMAN I Sekayu hingga Perguruan Tinggi Universitas Jember Program Studi Sistem Informasi yang tidak bisa disebutkan satu persatu;
7. Almamater Program Studi Sistem Informasi Universitas Jember;
8. Sahabat-sahabat yang saling mendukung dari awal perjuangan;
9. Keluarga besar NEFOTION.

**MOTTO**

Semua yang ada didunia ini pasti akan ada batasnya, maka dari itu jalani apa adanya tanpa harus di rekayasa (Derry Rismawan).



**PERNYATAAN**

Saya yang bertanda tangan di bawah ini:

Nama : Derry Rismawan

NIM : 112410101074

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul Implimentasi *Simple Multi-Attribute Rating Technique (SMART)* Pada Sistem Informasi Pengambilan Keputusan Pemilihan peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) Personil Tentara Nasional Indonesia (Studi Kasus: Divisi Infanteri 2 Kostrad Malang) adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya, belum pernah diajukan pada institusi mana pun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

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

Jember, 2017

Yang menyatakan,

Derry Rismawan

NIM 112410101074

**SKRIPSI**

**IMPLIMENTASI *SIMPLE MULTI-ATTRIBUTE RATING TECHNIQUE*  
(*SMART*) PADA SISTEM INFORMASI PENGAMBILAN KEPUTUSAN  
PEMILIHAN PESERTA PENDIDIKAN PEMBENTUKAN BINTARA  
REGULER (DIKTUBA REG) PERSONIL TENTARA  
NASIONAL INDONESIA  
(STUDI KASUS: DIVISI INFANTERI 2 KOSTRAD MALANG)**

oleh :

**Derry Rismawan**

**NIM 112410101074**

Pembimbing :

Dosen Pembimbing Utama : Prof. Drs. Slamin, M.Comp.Sc., Ph.D  
NIP.196704201992011001  
Dosen Pembimbing Pendamping : Yanuar Nurdiansyah, ST , M.Cs  
NIP. 198201012010121004

**PENGESAHAN PEMBIMBING**

Skripsi berjudul Implimentasi *Simple Multi-Attribute Rating Technique (SMART)* Pada Sistem Informasi Pengambilan Keputusan Pemilihan Peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) Personil Tentara Nasional Indonesia (Studi Kasus: Divisi Infanteri 2 Kostrad Malang), telah diuji dan disahkan pada :

hari, tanggal : Jum'at, 17 Maret 2017

tempat : Program Studi Sistem Informasi Universitas Jember.

Disetujui oleh :

Pembimbing I,

Pembimbing II,

Prof. Drs. Slamin, M.Comp.Sc., Ph.D

NIP. 196704201992011001

Yanuar Nurdiansyah, ST , M.Cs

NIP. 198201012010121004



**PENGESAHAN PENGUJI**

Skripsi berjudul Implimentasi *Simple Multi-Attribute Rating Technique (SMART)* Pada Sistem Informasi Pengambilan Keputusan Pemilihan Peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) Personil Tentara Nasional Indonesia (Studi Kasus: Divisi Infanteri 2 Kostrad Malang), telah diuji dan disahkan pada :

hari, tanggal : 17 Maret 2017

tempat : Program Studi Sistem Informasi Universitas Jember.

Tim Penguji :

Penguji I,

Penguji II,

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

NIP. 1969092819930121001

M. Arif Hidayat, S.Kom., M.Kom

NIP. 198101232010121003

Mengesahkan  
Ketua Program Studi,

Prof. Drs. Slamin, M.Comp.Sc., Ph.D

NIP 196704201992011001



## RINGKASAN

**Implimentasi *Simple Multi-Attribute Rating Technique (SMART)* Pada Sistem Informasi Pengambilan Keputusan Pemilihan Peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) Personil Tentara Nasional Indonesia (Studi Kasus: Divisi Infanteri 2 Kostrad Malang);** Derry Rismawan; 112410101074; 243 halaman; Program Studi Sistem Informasi Universitas Jember.

Seleksi Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) Tentara Nasional Indonesia Angkatan Darat (TNI AD) merupakan pendidikan alih golongan dari Tamtama menjadi Bintara. Pola seleksi yang membutuhkan data yang banyak dan waktu yang panjang. Permasalahan yang sering terjadi pada divisi kostrad yaitu lamanya proses dalam pelaksanaan seleksi Pemilihan Peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG). Berdasarkan permasalahan tersebut, diperlukan adanya penyelesaian menggunakan sistem informasi pendukung keputusan menggunakan metode *Simple Multi-Attribute Rating Technique (SMART)*. Metode *Simple Multi-Attribute Rating Technique (SMART)* bersifat transparan dalam membuat kriteria, mudah dan bersifat fleksibel dalam memberikan nilai bobot atribut. Kriteria yang digunakan dalam penelitian ini sebanyak 13 antara lain kesehatan umum, akademik pengetahuan umum, akademik pengetahuan militer, akademik pengetahuan militer kecabangan, psikotes, Prestasi olahraga, prestasi tertinggi operasi tempur, umur, jasmani lari, jasmani *push up*, jasmani *sitt up*, jasmani *satle run* dan jasmani renang. Model pengembangan sistem informasi pendukung keputusan pemilihan peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) menggunakan model *waterfall*. Sistem informasi pendukung keputusan pemilihan peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) menghasilkan nilai tertinggi hingga nilai minimum.

## PRAKATA

Puji syukur kehadirat Allah SWT yang telah melimpahkan rahmat dan karunia-Nya, sehingga skripsi yang berjudul “Sistem Informasi Pengambilan Keputusan Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) Menggunakan Metode *Simple Multi Attribute Rating Technique (SMART)* (Studi Kasus: Divisi Infanteri 2 Kostrad Malang)” dapat terselesaikan. Shalawat dan salam senantiasa tercurah kepada Nabi Muhammad Saw. Skripsi ini disusun untuk memenuhi salah satu syarat untuk 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 ucapan terima kasih kepada:

1. Prof. Drs. Slamir, M.Comp.Sc., Ph.D. selaku Ketua Program Studi Sistem Informasi Universitas Jember yang telah memberikan arahan, ilmu, nasehat, serta saran dengan penuh kesabaran;
2. Yanuar Nurdiansyah, ST , M.Cs. selaku Dosen Pembimbing Pendamping yang telah memberikan arahan, ilmu dan petunjuk, nasehat, koreksi serta saran dengan teliti dan penuh kesabaran;
3. Semua dosen Program Studi Sistem Informasi yang telah memberikan bekal ilmu dan pengetahuan selama penulis menjadi mahasiswa.

Penulis menyadari bahwa penulisan skripsi ini masih jauh dari kata sempurna, oleh sebab itu penulis mengharapkan adanya masukan yang bersifat membangun dari semua pihak. Penulis berharap skripsi ini dapat bermanfaat bagi semua pihak. Atas perhatian dan dukungannya, penulis sampaikan terima kasih.

Jember, 2016

Penulis

**DAFTAR ISI**

PERSEMBAHAN .....	ii
MOTTO .....	iii
PERNYATAAN.....	iv
SKRIPSI.....	v
PENGESAHAN PEMBIMBING.....	vi
PENGESAHAN PENGUJI.....	vii
RINGKASAN .....	viii
PRAKATA.....	ix
DAFTAR ISI.....	x
DAFTAR GAMBAR .....	xvi
DAFTAR TABEL.....	xix
DAFTAR LAMPIRAN.....	xxi
BAB 1. PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan dan Manfaat .....	2
1.3.1 Tujuan .....	2
1.3.2 Manfaat .....	3
1.4 Batasan Masalah .....	3
1.5 Sistematika Penulisan .....	4
BAB 2. TINJAUAN PUSTAKA .....	6
2.1 Penelitian Terdahulu .....	6
2.2 Sistem Informasi .....	8
2.3 Sistem Pendukung Keputusan.....	9

2.4 Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) .....	10
2.5 Simple Multi-Attribute Rating Technique .....	10
<b>BAB 3. METODOLOGI PENELITIAN.....</b>	<b>13</b>
3.1 Tempat dan Waktu Penelitian.....	13
3.2 Jenis Penelitian.....	13
3.3 Alur Penelitian .....	13
3.4 Teknik Pengumpulan Data.....	14
3.1.1 Wawancara.....	14
3.1.2 Studi Pustaka.....	15
3.5 Model Pengembangan Sistem.....	15
3.5.1 Analisis Kebutuhan .....	15
3.5.2 Desain Sistem.....	17
3.5.3 Implementasi (Pengkodean).....	18
3.5.4 Pengujian.....	19
3.5.5 Pemeliharaan .....	22
3.6 Gambaran Umum Sistem .....	22
<b>BAB 4. PENGEMBANGAN SISTEM.....</b>	<b>24</b>
4.1.1 Statement Of Purpose (SOP).....	24
4.1.2 Fungsi Sistem.....	24
3.4 Analisis Kebutuhan Sistem.....	25
3.4.4 Kebutuhan Fungsional .....	25
3.4.5 Kebutuhan Non-Fungsional .....	25
3.5 Desain Sistem.....	26
3.5.4 Business Process .....	26
3.5.5 UseCase Diagram.....	26
4.3.3 Skenario .....	35
4.3.4 Activity Diagram.....	40
4.3.4 Sequence Diagram .....	44

4.3.5 Class Diagram .....	50
4.3.6 Entity Relationship Diagram (ERD) .....	50
4.4 Implementasi koding .....	51
4.5 Pengujian Sistem .....	51
4.5.1 Metode White-box .....	52
4.5.2 Metode Black Box .....	55
<b>BAB 5. HASIL DAN PEMBAHASAN.....</b>	<b>56</b>
5.1 Hasil Implimentasi Kode Program Pada Sistem Informasi pengambilan keputusan pemilihan Peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) pada personil Tentara Nasional Indonesia (Studi Kasus: Divisi Infanteri 2 Kostrad Malang) .....	56
5.1.1 Fitur Login .....	56
5.1.2 Fitur Home .....	57
5.1.3 Fitur Menu Mengelola Data Pendaftar DIKTUBA REG.....	58
5.1.4 Fitur Menu Operator mengelola data pendaftar DIKTUBA REG .....	59
5.1.5 Fitur Menu Mengelola Kriteria .....	60
5.1.6 Fitur Menu Edit Bobot .....	61
5.1.7 Fitur Menu Perhitungan Metode .....	61
5.1.8 Fitur Menu Melihat Penerima DIKTUBA REG .....	62
5.1.9 Fitur Menu Melihat Penerima dan Cetak DIKTUBA REG .....	63
5.1.10 Implementasi Metode Simple Multi-Attribute Rating Technique (SMART) .....	63
5.1.11 Pengujian Analisis Data Personil DIKTUBA REG .....	66
5.2 Pembahasan Implementasi Metode Simple Multi-Attribute Rating Technique (SMART) pada Sistem Informasi Pengambilan Keputusan Pendidikan Pembentukan Bintara Reguler.....	83
5.3 Pembahasan Sistem Informasi Pengambilan Keputusan Pendidikan Pembentukan Bintara Reguler (Studi Kasus: Divisi Infanteri 2 Kostrad Malang)	



BAB 6. PENUTUP .....	86
6.1 Kesimpulan .....	86
6.2 Saran .....	87
LAMPIRAN .....	90
Lampiran A. Skenario .....	90
A.1 Skenario Mengelola Data DIKTUBA REG .....	90
A.2 Skenario Mengelola Kriteria Kesehatan Umum .....	93
A.3 Skenario Mengelola Kriteria Akademik Pengetahuan Umum .....	96
A.4 Skenario Mengelola Kriteria Akademik Pengetahuan Militer .....	100
A.5 Skenario Mengelola Kriteria Akademik Pengetahuan Militer Kecabangan .....	104
A.6 Skenario Mengelola Kriteria Psikotes .....	108
A.7 Skenario Mengelola Kriteria Prestasi Olahraga .....	111
A.8 Skenario Mengelola Kriteria Prestasi Operasi Tempur .....	115
A.9 Skenario Mengelola Kriteria Umur .....	119
A.10 Skenario Mengelola Kriteria Jasmani Lari .....	122
A.11 Skenario Mengelola Kriteria Jasmani Push Up .....	125
A.12 Skenario Mengelola Kriteria Sitt Up .....	128
A.13 Skenario Mengelola Kriteria Jasmani Satle Run .....	131
A.14 Skenario Mengelola Kriteria Jasmani Renang .....	134
A.15 Skenario Edit Bobot .....	137
A.16 Skenario Melihat Penerima DIKTUBA REG .....	139
Lampiran B. Activity Diagram .....	141
B.1 Activity Diagram Mengelola data diktuba reg .....	141
B.2 Activity Diagram Mengelola Kriteria Kesehatan Umum .....	142
B.3 Activity Diagram Mengelola Kriteria Akademik Pengetahuan Umum .....	143
B.4 Activity Diagram 3Mengelola Kriteria Akademik Pengetahuan Militer .....	144
B.5 Activity Diagram Mengelola Kriteria Akademik Pengetahuan Militer Kecabangan .....	145

B.6 Activity Diagram Mengelola kriteria Psikotes .....	146
B.7 Activity Diagram Mengelola Kriteria Prestasi Olahraga .....	147
B.8 Activity Diagram 7Mengelola Kriteria Prestasi Olahraga Tempur.....	148
B.9 Activity Diagram Mengelola kriteria umur .....	149
B.10 Activity Diagram Mengelola kriteria Jasmani lari .....	150
B.11 Activity Diagram Mengelola Kriteria Jasmani Push Up.....	151
B.12 Activity Diagram Mengelola Kriteria Jasmani Sitt Up .....	152
B.13 Activity Diagram Mengelola kriteria Jasmani Satle Run.....	153
B.14 Activity Diagram Mengelola kriteria Jasmani Renang .....	154
B.15 Activity Diagram Edit Bobot .....	155
B.16 Activity Diagram Melihat penerima DIKTUBA REG.....	156
Lampiran C. Sequence Diagram .....	157
C.1 Squence Diagram Admin Mengelola Data DIKTUBA REG.....	157
C.2 Squence Diagram Operator Mengelola Data DIKTUBA REG.....	158
C.3 Squence Diagram Admin Mengelola Kriteria kesehatan Umum .....	159
C.4 Squence Diagram Admin Mengelola Kriteria Akademik Pengetahuan Umum	160
C.5 Squence Diagram Admin Mengelola Kriteria Akademik Pengetahuan Militer	161
C.6 Squence Diagram Admin Mengelola Kriteria Akademik Pengetahuan Militer Kecabangan.....	162
C.7 Squence Diagram Admin Mengelola Kriteria Psikotes .....	163
C.8 Squence Diagram Admin Mengelola Kriteria Prestasi Olahraga.....	164
C.9 Squence Diagram Admin Mengelola Kriteria Prestasi Operasi Tempur	165
C.10 Squence Diagram Admin Mengelola Kriteria Umur .....	166
C.11 Squence Diagram Admin Mengelola Kriteria Jasmani Lari .....	167
C.12 Squence Diagram Admin Mengelola Kriteria Jasmani Push Up .....	168
C.13 Squence Diagram Admin Mengelola Kriteria Jasmani Sitt Up .....	169
C.14 Squence Diagram Admin Mengelola Kriteria Jasmani Satle Run .....	170
C.15 Squence Diagram Admin Mengelola Kriteria Jasmani Renang.....	171
C.16 Squence Diagram Admin Edit Bobot .....	172



C.17 Squence Diagram Operator melihat Penerima DIKTUBA REG .....	172
C.18 Squence Diagram Admin Melihat Penerima dan Mencetak DIKTUBA REG	173
Lampiran D. Implementasi Coding.....	174
D.1 Login .....	174
D.2 Home Admin .....	175
D.3 Home operator.....	176
D.4 Admin Input Penerima .....	178
D.5 Admin Penerima DIKTUBA REG.....	182
D.6 Kriteria Akademik Pengetahuan Militer .....	183
D.7 Kriteria Akademik Pengetahuan Umum .....	184
D.8 Kriteria Akademik Pengetahuan Militer kecabangan .....	185
D.9 Kriteria Psikotes .....	186
D.10 Kriteria Jasmani Lari.....	187
D.11 Kriteria Jasmani Push Up.....	188
D.12 Kriteria Jasmani Sitt Up .....	189
D.13 Kriteria Jasmani Satle Run.....	190
D.14 Kriteria Jasmani Renang .....	191
D.15 Kriteria Umur .....	192
D.16 Kriteria Kesehatan Umum.....	193
D.17 Kriteria Prestasi Olahraga .....	194
D.18 Kriteria Prestasi Operasi Tempur.....	195
D.19 Operator Melihat Penerima DIKTUBA REG .....	196
D.20 Admin Melihat Penerima DIKTUBA REG .....	198
Lampiran E. Black Box Testing.....	200

**DAFTAR GAMBAR**

Gambar 2.1 Flowchart Metode SMART.....	12
Gambar 3.1 Alur Penelitian.....	14
Gambar 3.2 Gambar Diagram tahapan model waterfall .....	15
Gambar 3.3 Contoh Listing Programe .....	19
Gambar 3.4 Contoh Diagram Alir.....	20
Gambar 3.5 Gambar Umum Sistem .....	23
Gambar 4.1 Gambar Bussiness Process DIKTUBA REG .....	26
Gambar 4.2 Usecase Diagram DIKTUBA REG.....	28
Gambar 4.3 Gambar Activity Diagram Perhitungan Metode SMART.....	44
Gambar 4.4 Gambar Sequence Diagram Perhitungan SMART.....	45
Gambar 5.1 Gambar Fitur Login.....	57
Gambar 5.2 Gambar Fitur Home .....	57
Gambar 5.3 Gambar Fitur Home Operator .....	58
Gambar 5.4 Gambar Fitur Menu mengelola data pendaftar DIKTUBA REG .....	59
Gambar 5.5 Gambar Fitur Operator Menu mengelola data pendaftar DIKTUBA REG.....	60
Gambar 5.6 Gambar Fitur Menu Mengelola Kriteria .....	60
Gambar 5.7 Gambar Fitur Menu Edit Bobot .....	61
Gambar 5.8 Fitur Menu Perhitungan Metode .....	62
Gambar 5.9 Gambar Fitur Menu Melihat Penerima DIKTUBA REG .....	62
Gambar 5.10 Fitur Menu Melihat Penerima dan Cetak DIKTUBA REG .....	63
Gambar 5.11 Code Programe memberikan Nilai Utility .....	64
Gambar 5.12 Code Programe Menghitung Hasil.....	65
Gambar B.1 <i>Activity Diagram</i> Mengelola Mengelola data diktuba reg.....	141
Gambar B.2 <i>Activity Diagram</i> Mengelola Kriteria Kesehatan Umum .....	142
Gambar B.3 <i>Activity Diagram</i> Mengelola Kriteria Akademik Pengetahuan Umum .....	143

Gambar B.4 <i>Activity Diagram</i> Mengelola Kriteria Akademik Pengetahuan Militer .....	144
Gambar B.5 <i>Activity Diagram</i> Mengelola Kriteria Akademik Pengetahuan Militer Kecabangan .....	145
Gambar B.6 <i>Activity Diagram</i> Mengelola kriteria Psikotes.....	146
Gambar B.7 <i>Activity Diagram</i> Mengelola Kriteria Prestasi Olahraga.....	147
Gambar B.8 <i>Activity Diagram</i> Mengelola Kriteria Prestasi Olahraga Tempur ..	148
Gambar B.9 <i>Activity Diagram</i> Mengelola kriteria umur .....	149
Gambar B.10 <i>Activity Diagram</i> Mengelola kriteria Jasmani lari.....	150
Gambar B.11 <i>Activity Diagram</i> Mengelola Kriteria Jasmani Push Up .....	151
Gambar B.12 <i>Actiivity Diagram</i> Mengelola Kriteria Jasmani Sitt Up.....	152
Gambar B.13 <i>Activity Diagram</i> Mengelola kriteria Jasmani Satle Run .....	153
Gambar B.14 <i>Activity Diagram</i> Mengelola kriteria Jasmani Renang.....	154
Gambar B.15 <i>Activity Diagram Edit</i> Bobot .....	155
Gambar B.16 <i>Activity</i> Melihat penerima DIKTUBA REG.....	156
Gambar C.1 <i>Squence Diagram</i> Admin Mengelola Data DIKTUBA REG.....	157
Gambar C.2 <i>Squence Diagram</i> Operator Mengelola Data DIKTUBA REG.....	158
Gambar C.3 <i>Squence Diagram Admin</i> Mengelola Kriteria kesehatan Umum....	159
Gambar C.4 <i>Squence Diagram Admin</i> Mengelola Kriteria Akademik Pengetahuan Umum.....	160
Gambar C.5 <i>Squence Diagram Admin</i> Mengelola Kriteria Akademik Pengetahuan Militer.....	161
Gambar C.6 <i>Squence Diagram Admin</i> Mengelola Kriteria Akademik Pengetahuan Militer Kecabangan.....	162
Gambar C.7 <i>Squence Diagram Admin</i> Mengelola Kriteria Psikotes .....	163
Gambar C.8 <i>Squence Diagram Admin</i> Mengelola Kriteria Prestasi Olahraga ...	164
Gambar C.9 <i>Squence Diagram Admin</i> Mengelola Kriteria Prestasi Operasi Tempur .....	165
Gambar C.10 <i>Squence Diagram Admin</i> Mengelola Kriteria Umur .....	166
Gambar C.11 <i>Squence Diagram Admin</i> Mengelola Kriteria Jasmani Lari .....	167
Gambar C.12 <i>Squence Diagram Admin</i> Mengelola Kriteria Jasmani Push Up..	168

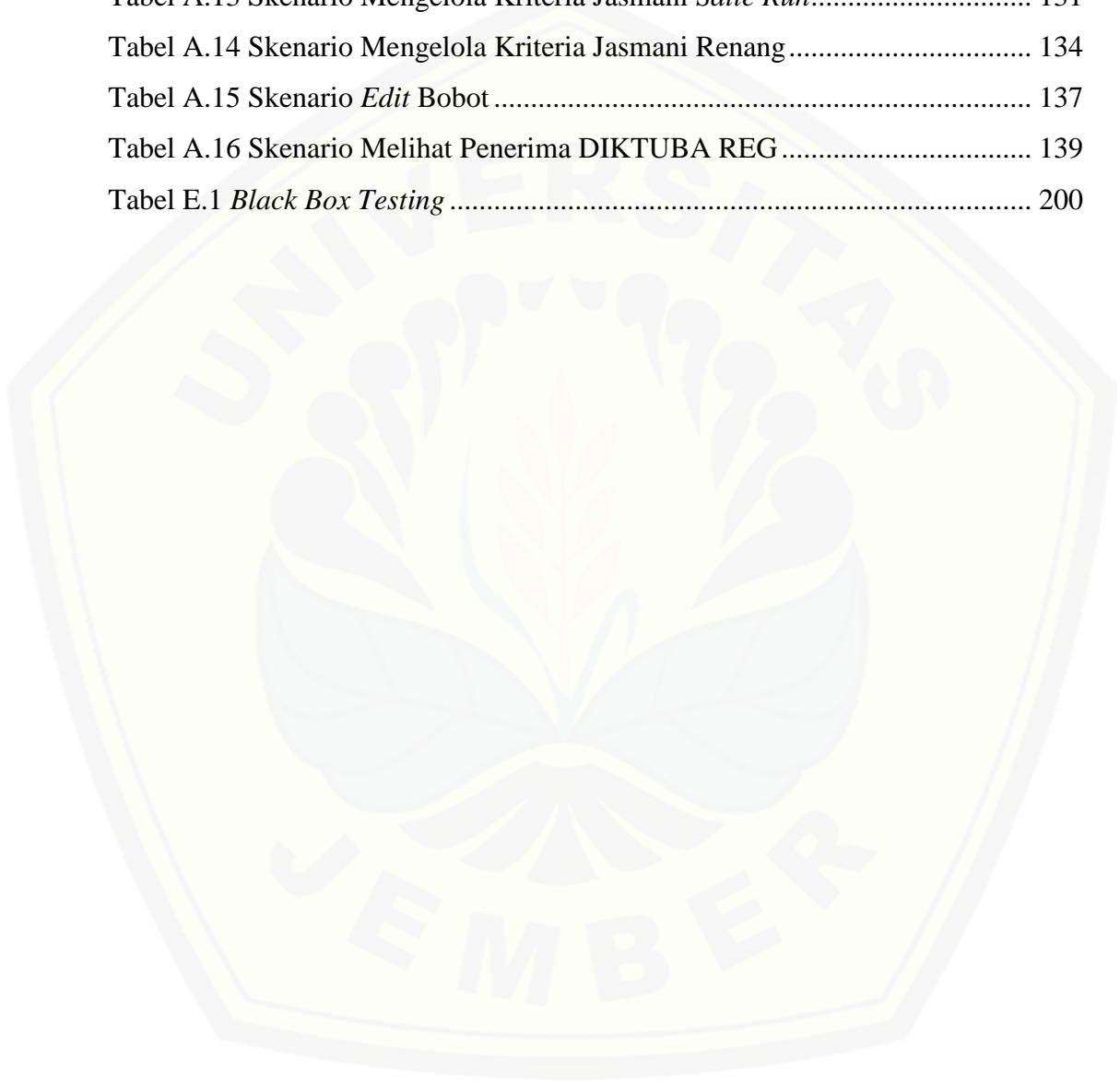
Gambar C.13 <i>Squence Diagram Admin</i> Mengelola Kriteria Jasmani Sitt Up ....	169
Gambar C.14 <i>Squence Diagram Admin</i> Mengelola Kriteria Jasmani Satle Run	170
Gambar C.15 <i>Squence Diagram Admin</i> Mengelola Kriteria Jasmani Renang ...	171
Gambar C.16 <i>Squence Diagram Admin</i> Edit Bobot .....	172
Gambar C.17 <i>Squence Diagram Operator</i> melihat Penerima DIKTUBA REG .	172
Gambar C.18 <i>Squence Diagram Admin</i> Melihat Penerima dan Mencetak DIKTUBA REG.....	173
Gambar D.1 <i>Login</i> .....	174
Gambar D.2 <i>Home Admin</i> .....	176
Gambar D.3 <i>Home Operator</i> .....	177
Gambar D.4 <i>Admin Input Penerima</i> .....	181
Gambar D.5 <i>Admin Penerima DIKTUBA REG</i> .....	183
Gambar D.6 Kriteria Akademik Pengetahuan Militer .....	183
Gambar D.7 Kriteria Akademik Pengetahuan Umum .....	184
Gambar D.8 Kriteria Akademik Pengetahuan Militer kecabangan .....	185
Gambar D.9 Kriteria Psikotes .....	186
Gambar D.10 Kriteria Jasmani Lari .....	187
Gambar D.11 Kriteria Jasmani <i>Push Up</i> .....	188
Gambar D.12 <i>Sitt Up</i> .....	189
Gambar D.13 Kriteria Jasmani <i>Satle Run</i> .....	190
Gambar D.14 Kriteria Jasmani Renang .....	191
Gambar D.15 Kriteria Umur .....	192
Gambar D.16 Kriteria Kesehatan Umum.....	193
Gambar D.17 Kriteria Prestasi Olahraga .....	194
Gambar D.18 Kriteria Prestasi Operasi Tempur .....	195
Gambar D.19 Operator Melihat Penerima DIKTUBA REG .....	197
Gambar D.20 <i>Admin</i> Melihat dan Mencetak Penerima DIKTUBA REG .....	199

**DAFTAR TABEL**

Tabel 3.1 Tabel Kriteria Pendidikan Pembentukan Bintara Reguler.....	16
Tabel 3.2 Tabel Uji <i>Black-box</i> .....	22
Tabel 4.1 Tabel Definisi <i>UseCase</i> Diagram .....	29
Tabel 4.2 Tabel Skenario Mengelola Data Perhitungan DIKTUBA REG .....	36
Tabel 4.3 Tabel <i>Test Case</i> Metode SMART DIKTUBA REG .....	54
Tabel 5.1 Tabel Kriteria, Atribut, dan Nilai Utility .....	66
Tabel 5.2 Tabel Inputan Personil Pertama .....	68
Tabel 5.3 Tabel Inputan Personil Kedua.....	69
Tabel 5.4 Tabel Inputan Personil Ketiga.....	70
Tabel 5.5 Tabel Inputan Personil Keempat.....	71
Tabel 5.6 Tabel Inputan Personil Kelima .....	72
Tabel 5.7 Tabel Inputan Personil Keenam.....	73
Tabel 5.8 Tabel Inputan Personil Ketujuh .....	74
Tabel 5.9 Tabel Inputan Personil Kedelapan .....	76
Tabel 5.10 Tabel Inputan Personil Kesembilan .....	77
Tabel 5.11 Tabel Inputan Personil Kesepuluh .....	78
Tabel 5.12 Tabel Inputan Personil Kesebelas .....	79
Tabel 5.13 Tabel Inputan Personil Keduabelas.....	80
Tabel 5.14 Tabel Inputan Personil Ketigabelas .....	81
Tabel 5.16 Tabel Hasil Perhitungan Metode <i>SMART</i> Nilai Rata-Rata Minimal .	70
Tabel A.1 Skenario Mengelola Data DIKTUBA REG .....	90
Tabel A.2 Tabel Skenario Mengelola Kriteria Kesehatan Umum .....	93
Tabel A.3 Skenario Mengelola Kriteria Akademik Pengetahuan Umum.....	96
Tabel A.4 Skenario Mengelola Kriteria Akademik Pengetahuan Militer.....	100
Tabel A.5 Skenario Mengelola Kriteria Akademik Pengetahuan Militer Kecabangan.....	104
Tabel A.6 Skenario Mengelola Kriteria Psikotes.....	108
Tabel A.7 Skenario Mengelola Kriteria Prestasi Olahraga.....	111
Tabel A.8 Skenario Mengelola Kriteria Prestasi Operasi Tempur .....	115

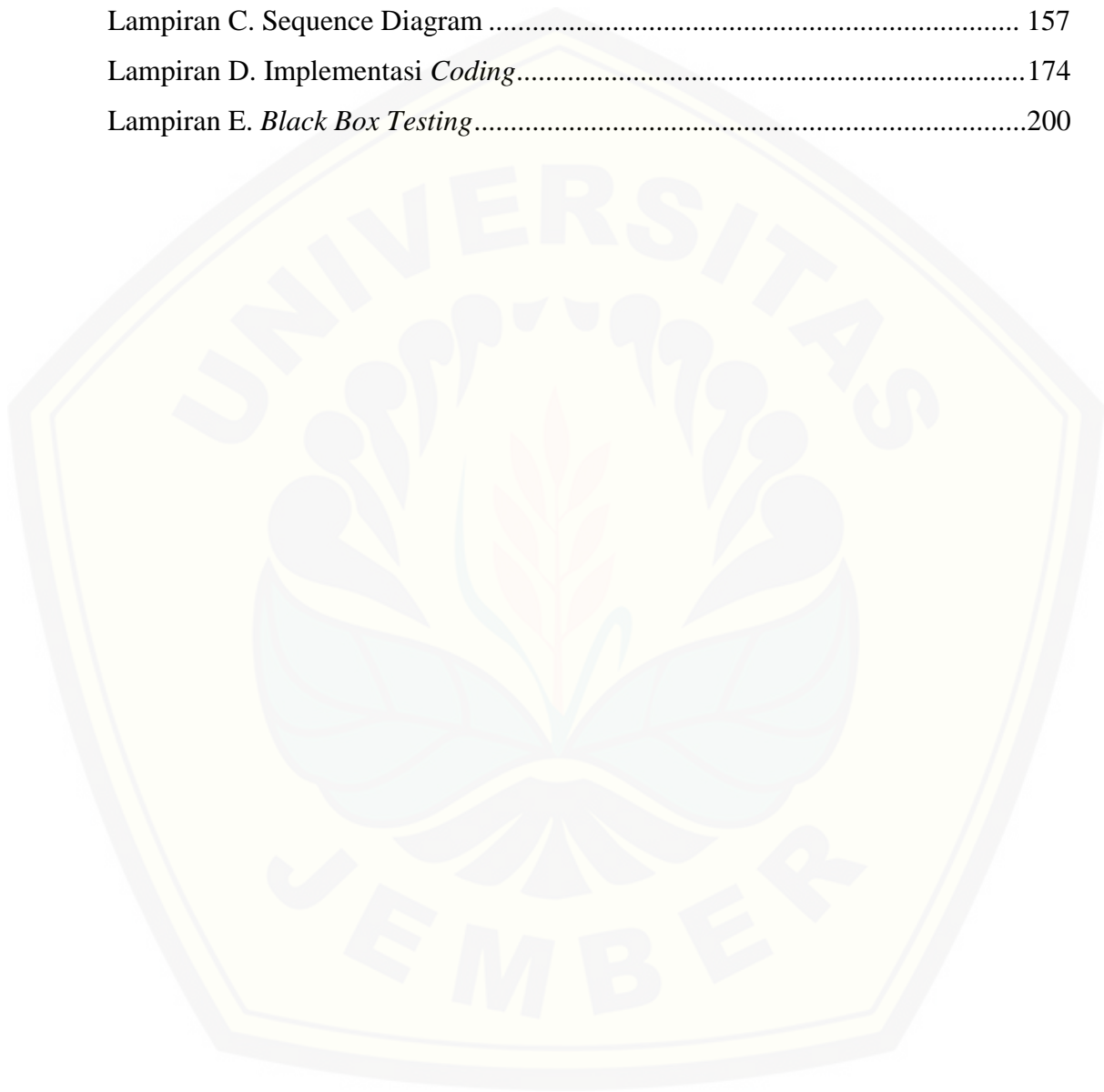


Tabel A.9 Skenario Mengelola Kriteria Umur.....	119
Tabel A.10 Skenario Mengelola Kriteria Jasmani Lari .....	122
Tabel A.11 Skenario Mengelola Kriteria Jasmani <i>Push Up</i> .....	125
Tabel A.12 Skenario Mengelola Kriteria <i>Sitt Up</i> .....	128
Tabel A.13 Skenario Mengelola Kriteria Jasmani <i>Satle Run</i> .....	131
Tabel A.14 Skenario Mengelola Kriteria Jasmani Renang .....	134
Tabel A.15 Skenario <i>Edit Bobot</i> .....	137
Tabel A.16 Skenario Melihat Penerima DIKTUBA REG .....	139
Tabel E.1 <i>Black Box Testing</i> .....	200



**DAFTAR LAMPIRAN**

Lampiran A. Skenario .....	90
Lampiran B. Activity Diagram .....	141
Lampiran C. Sequence Diagram .....	157
Lampiran D. Implementasi <i>Coding</i> .....	174
Lampiran E. <i>Black Box Testing</i> .....	200





## BAB 1. PENDAHULUAN

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

### 1.1 Latar Belakang

Divisi Infanteri 2 KOSTRAD merupakan salah satu realisasi dari reorganisasi di jajaran TNI AD pada umumnya dan KOSTRAD pada khususnya. Jika dilihat dari komposisi dan kekuatannya terdapat perbedaan dua divisi tersebut. Di saat itu Pimpinan TNI AD menyadari perubahan itu jika ditinjau dari segi pembinaan satuan akan lebih tepat apabila perubahan itu menjadi DIVIF LINUD dan Divisi Infanteri. Akan tetapi mengingat terbatasnya dana yang tersedia pada saat itu, maka satuan yang ada disusun sedemikian rupa sehingga tugas pokoknya tetap dapat terlaksana dengan baik (PENGKOSTRAD, 2009).

Divisi Infanteri 2 KOSTRAD sendiri memiliki beberapa program pendidikan yaitu salah satunya Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG). Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) adalah alih golongan dari TAMTAMA ke BINTARA. Pendidikan Pembentukan Bintara TNI AD bertujuan untuk membentuk Bintara Siswa menjadi Bintara TNI AD yang memiliki sikap dan perilaku sebagai prajurit sapta marga dan sumpah prajurit. Salah satu masalah pada Pendidikan Pembentukan Bintara Reguler adalah banyaknya calon pendaftar dan membutuhkan waktu yang lama dalam pelaksanaannya. Hal ini dilakukan mengingat Bintara adalah tulang punggung kesatuan di militer yang berperan sebagai penghubung antara Perwira dengan Tamtama atau sebaliknya dalam segi operasional.

Pada pendaftaran calon diktuba masih dilakukan secara manual. Agar pemrosesan semakin praktis, dibutuhkanlah sebuah sistem yang mampu mempermudah proses pendaftaran diktuba tersebut. Pada sistem yang akan dibangun dibutuhkan juga suatu metode pengambil keputusan agar penyeleksian calon diktuba semakin cepat dan tepat. Karena dalam pemilihan keputusan untuk

peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG), banyak permasalahan yang sifatnya *multicriteria* dan *multiobject*. Salah satu metode yang tepat digunakan dari permasalahan tersebut adalah metode *Simple Multi-Attribute Rating Technique* (SMART). Metode SMART dipilih karena metode ini dapat memaksimalkan banyaknya kriteria yang ada dalam pengambilan keputusan pemilihan peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG). Dengan semakin banyak kriteria yang dipakai, maka hasil pengambilan keputusan akan semakin akurat.

## 1.2 Rumusan Masalah

Berdasarkan uraian diatas dapat dirumuskan permasalahan sebagai berikut:

1. Bagaimana merancang dan membangun sistem informasi penunjang keputusan pemilihan peserta Pendidikan Pembentukan Bintara Regular (DIKTUBA REG) personil Tentara Nasional Indonesia (TNI) menggunakan metode *Simple Multi-Attribute Rating Technique* (SMART) ?
2. Bagaimana cara mengimplementasikan metode *Simple Multi-Attribute Rating Technique* (SMART) untuk mendukung keputusan pemilihan peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) personil Tentara Nasional Indonesia (TNI) ?

## 1.3 Tujuan dan Manfaat

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

### 1.3.1 Tujuan

Tujuan dari penelitian adalah untuk menganalisa:

1. Memberikan pendukung keputusan untuk keputusan pemilihan peserta pendidikan Pembentukan Bintara Regular (DIKTUBA REG) personil Tentara Nasional Indonesia (TNI);
2. Membuat sistem informasi pendukung keputusan pemilihan peserta Pendidikan Pembentukan Bintara Regular (DIKTUBA REG) personil Tentara

Nasional Indonesia (TNI) menggunakan metode *Simple Multi-Attribute Rating Technique (SMART)*.

### 1.3.2 Manfaat

Penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

#### 1.3.2.1 Bagi Akademis

Penelitian yang dilakukan diharapkan memberikan hasil yang mampu memberikan masukan informasi yang terkait dengan judul penelitian kepada pembaca pada umumnya dan kepada Program Studi Sistem Informasi Universitas Jember pada khususnya.

#### 1.3.2.2 Bagi Peneliti

1. Mengetahui data yang digunakan dalam menentukan personil yang memenuhi persyaratan Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG);
2. Mengetahui proses penerapan sistem informasi pengambilan keputusan dalam menentukan penerimaan Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) pada Divisi Infanteri 2 Kostrad Malang;
3. Mengetahui cara kerja metode *Simple Multy Attribute Rating Technique (SMART)* dalam menentukan sebuah sistem informasi pengambilan keputusan penerimaan Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG).

#### 1.3.2.3 Bagi Objek Penelitian

1. Diharapkan dapat membantu Divisi Infanteri 2 Kostrad Malang dalam pengambilan keputusan penerimaan Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG).
2. Diharapkan dapat menjadi acuan dalam penerimaan Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG).

## 1.4 Batasan Masalah

Agar tidak terjadi penyimpangan dalam proses penelitian dan pembangunan sistem, maka ditetapkan beberapa batasan permasalahan. Adapun batasan masalah dalam penelitian ini adalah :

1. Sistem hanya dapat digunakan untuk mengambil keputusan pemilihan peserta Pendidikan Pembentukan Bintara Regular (DIKTUBA REG) Divisi Infantri 2 Kostrad Malang;
2. Sistem menggunakan metode *Simple Multi-Attribute Rating Technique (SMART)*;
3. Sistem dibangun berbasis *desktop*;
4. Menggunakan *database MySQL*;
5. Sistem ini dapat menampilkan nilai pesonil tertinggi hingga nilai batas minimum Pendidikan Pembentukan Bintara Regular (DIKTUBA REG);
6. Sistem ini menggunakan 13 kriteria yang terdiri dari kesehatan umum, akademik pengetahuan umum, akademik pengetahuan militer, akademik pengetahuan militer kecabangan, psikotes, umur, prestasi olahraga, prestasi operasi tempur, jasmani lari, jasmani *push up*, jasmani *sitt up*, jasmani *satle run*, dan jasmani renang;

### 1.5 Sistematika Penulisan

Adapun sistematika penulisan skripsi ini adalah sebagai berikut:

#### 1. Pendahuluan

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

#### 2. Tinjauan Pustaka

Bab ini memaparkan tinjauan terhadap hasil-hasil penelitian terdahulu berkaitan dengan rumusan masalah penelitian, sistem informasi, sistem pendukung keputusan, Pendidikan Pembentukan Bintara Regular (DIKTUBA REG), dan kajian teori metode *Simple Multi Attribute Rating Technique (SMART)* yang berkaitan dengan rumusan masalah dalam penelitian.

#### 3. Metode Penelitian

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

#### 4. Pengembangan Sistem

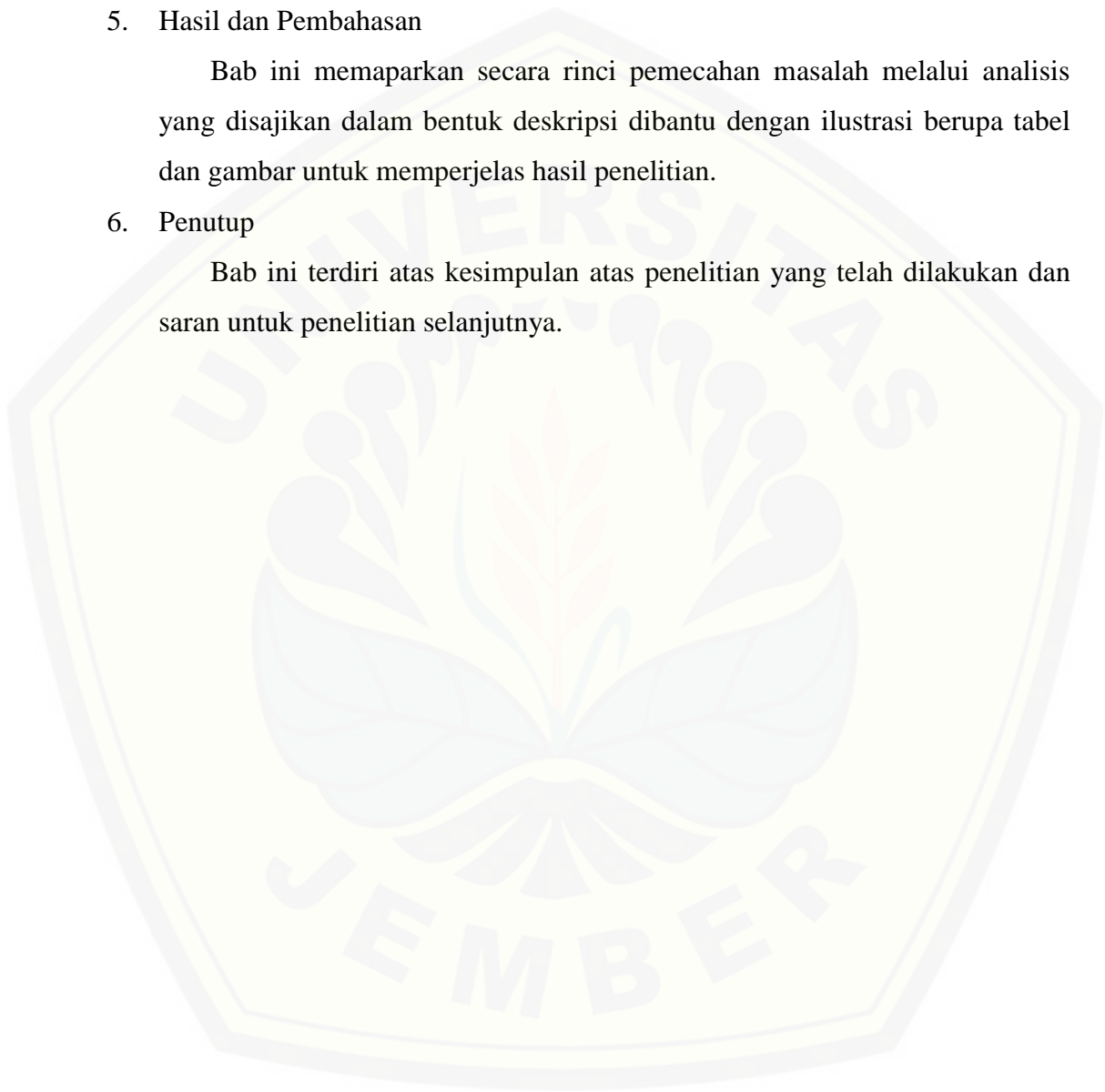
Bab ini berisi uraian tentang langkah-langkah yang ditempuh dalam proses menganalisis dan merancang sistem yang hendak dibangun meliputi desain, pengkodean, dan pengujian sistem.

#### 5. Hasil dan Pembahasan

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

#### 6. Penutup

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





## BAB 2. TINJAUAN PUSTAKA

Dalam bab ini dipaparkan teori-teori dan pustaka yang digunakan dalam penelitian. Teori-teori dan pustaka yang digunakan dan dibahas dalam penelitian ini, yaitu penelitian terdahulu, sistem informasi, sistem pendukung keputusan, Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG), dan metode *Simple Multi Attribute Rating Technique (SMART)*.

### 2.1 Penelitian Terdahulu

Penelitian ini dilakukan berdasarkan teori dan studi - studi yang dilakukan oleh peneliti terdahulu. Adapun penelitian terdahulu tersebut antara lain:

1. Penelitian terdahulu dengan judul “Metode *Simple Multi Attribute Rating Technique (SMART)* Pada Sistem Pendukung Keputusan Pemilihan Pembelian Mobil Keluarga” (Atiqah, 2013) menjelaskan bahwa metode *Simple Multi Attribute Rating Technique (SMART)* cocok untuk menyelesaikan masalah yang sifatnya *multicriteria* dan *multiobject*. Adapun kriteria yang digunakan dalam pemilihan mobil keluarga mulai dari harga, fasilitas, kebutuhan dan kualitasnya. Tahapan untuk penilaiannya pertama ditentukan beberapa mobil yang akan dipilih, kemudian dipilih empat mobil dari beberapa mobil tersebut. Penilaian pada setiap kriteria dan pada setiap atributnya diberikan nilai pembobotan sendiri sehingga hasil yang didapat maksimal. Setiap kriteria memiliki nilai sendiri dan setiap atribut di berikan bobot sendiri, sehingga perbandingan nilai yang diperoleh bukan hanya dari bobot atribunya saja tetapi juga dihitung dengan nilai kreteria yang di berikan.
2. Penelitian terdahulu dengan judul “Penerapan Metode Simple Multi Attribute Rating Technique untuk Penentuan Bintang Nararia Bhayangkara ” (Nasution, 2013) Untuk mengambil keputusan terhadap pemilihan personel kepolisian yang berhak menerima bintang nararia (Studi Kasus Kepolisian Daerah Riau)”. Pada penelitian ini metode *Simple Multi Attribute Rating Technique* diterapkan sebagai teknik yang dalam menentukan personel yang

berhak menerima bintang nararia. Adapun kriterianya yaitu pelanggaran, kehadiran, pemahaman terhadap tugas/keahlian, penghargaan, sistem manajemen kerja (SMK) penilaian, produktivitas dan kecepatan pelayanan. Seleksi dilakukan dalam 2 tahapan, yaitu persyaratan administrasi, dan persyaratan perhitungan. Persyaratan administrasi meliputi masa dinas minimal 24 tahun, memiliki satya lencana 8, 16, dan 24 tahun serta tidak pernah melakukan pelanggaran pidana. Sedangkan persyaratan perhitungan meliputi pelanggaran (kode etik profesi dan disiplin), kehadiran, pemahaman terhadap tugas/keahlian, penghargaan yang diperoleh (sambang nusa, fungsi operasional, dan fungsi staff), sistem manajemen kerja penilaian (generik dan spesifik), produktivitas, dan kecepatan pelayanan. Hasil penelitian ini metode *Simple Multi Attribute Rating Technique* dapat diimplementasikan pada pemilihan personel kepolisian yang berhak menerima bintang nararia Demi efisiensi dan efektifitas kerja.

Dari ke dua penelitian terdahulu dapat disimpulkan bahwa terdapat kekurangan dalam memaksimalkan hasil yang diperoleh. Kreteria yang digunakan dalam mengambil keputusan sangat kurang banyak dan kurang variatif, sehingga perbandingan antar atribut sangat dekat dengan nilai atribut lainnya. Pemberian nilai bobot yang terlalu dekat tersebut menyebabkan kurang maksimalnya hasil akhir penilaian keputusan yang didapat. Setelah mengkaji kedua artikel tersebut metode SMART digunakan untuk menghadapi sebuah permasalahan yang memiliki perbandingan banyak kriteria seperti pemilihan mobil keluarga. Pada penelitian sistem pendukung keputusan untuk pemilihan peserta Pendidikan Pembentukan Bintara (DIKTUBA REG) ini juga memiliki persamaan banyak kreteria yang akan di gunakan. Banyaknya kreteria tersebut penggunaan metode SMART untuk digunakan dalam mendukung keputusan pemilihan peserta Pendidikan Pembentukan Bintara (DIKTUBA REG) sangat cocok, serta dapat diterapkan dengan baik dalam Sistem informasi pengambilan keputusan pemilihan peserta Pendidikan Pembentukan Bintara (DIKTUBA REG) di Divisi Infanteri 2 Kostrad Malang.



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

Pada permasalahan pengambilan keputusan pemilihan peserta Pendidikan Pembentukan Bintara (DIKTUBA REG), metode SMART sangatlah tepat digunakan, mengingat pada permasalahan tersebut banyaknya atribut yang dipakai untuk menunjang keputusan yang diambil. Semakin banyak atribut yang digunakan dalam metode SMART semakin baik juga mengambil nilai keputusan akhir yang dihasilkan. Metode SMART diharapkan dapat memaksimalkan pengambilan keputusan untuk peserta yang berhak mendapatkan pendidikan pembentukan Bintara Reguler (DIKTUBA REG) dengan memperhitungkan banyak atribut yang digunakan secara detail dan menyeluruh dalam pemberian bobot penilaiannya.

## **2.2 Sistem Informasi**

Sistem adalah beberapa komponen yang memiliki fungsi yang berbeda dan saling berkait bekerja sama untuk mencapai suatu tujuan, sedangkan informasi adalah data/fakta yang telah diproses dan telah memiliki arti yang bermanfaat terutama dalam kebutuhan komunikasi. Jadi sistem informasi mencakup sejumlah komponen (manusia, komputer, teknologi informasi,

dan prosedur kerja), ada sesuatu yang diproses (data menjadi informasi), dan dimaksudkan untuk mencapai suatu sasaran atau tujuan (Yunitarini, 2013).

Sistem Informasi (SI) adalah kombinasi dari teknologi informasi dan aktifitas orang yang menggunakan teknologi untuk mendukung operasi dan manajemen (Dudung, 2015). Berdasarkan penjelasan diatas dapat disimpulkan bahwa sistem informasi adalah gabungan dari teknologi informasi yang digunakan untuk membantu serta mendukung operasi dan manajemen sehingga mempermudah dalam komunikasi.

### **2.3 Sistem Pendukung Keputusan**

Sistem pendukung keputusan/*Decision Support Sistem (DSS)* termasuk bagian dari sistem informasi berdasarkan dukungan kepada pemakainya, dapat juga dikatakan sebagai sistem komputer yang mengolah data menjadi informasi untuk mengambil keputusan dari masalah semi terstruktur yang spesifik. Sebelum mengetahui definisi sistem pendukung keputusan alangkah baiknya mengetahui konsep sistem pendukung keputusan terlebih dahulu, mulai dari sistem, informasi, keputusan dan sistem informasi (Monica, 2015).

Definisi Sistem Penunjang Keputusan (SPK) secara sederhana adalah sebuah sistem yang digunakan sebagai alat bantu menyelesaikan masalah untuk membantu pengambil keputusan (manajer) dalam menunjang keputusan tetapi tidak untuk menggantikan kapasitas manajer hanya memberikan pertimbangan. Sistem Penunjang Keputusan (SPK) ditujukan untuk keputusan-keputusan yang memerlukan penilaian atau pada keputusan-keputusan yang sama sekali tidak dapat didukung oleh algoritma (Turban, 2005).

Pada dasarnya sistem penunjang keputusan merupakan pengembangan lebih lanjut dari Sistem Informasi Manajemen terkomputerisasi yang dirancang sedemikian rupa sehingga bersifat interaktif dengan pemakainya. Dengan tujuan untuk memudahkan integrasi antara berbagai komponen dalam proses pengambilan keputusan seperti prosedur, kebijakan, analisis,

pengalaman dan wawasan manajer untuk mengambil keputusan yang lebih baik (Setyawan, 2015).

Dari ketiga penjelasan diatas dapat penulis simpulkan bahwa sistem penunjang keputusan adalah sistem manajemen yang dibuat untuk mempermudah dalam memberikan keputusan. Sistem yang mampu memberikan menyelesaikan masalah secara maksimal dalam memecahkan masalah.

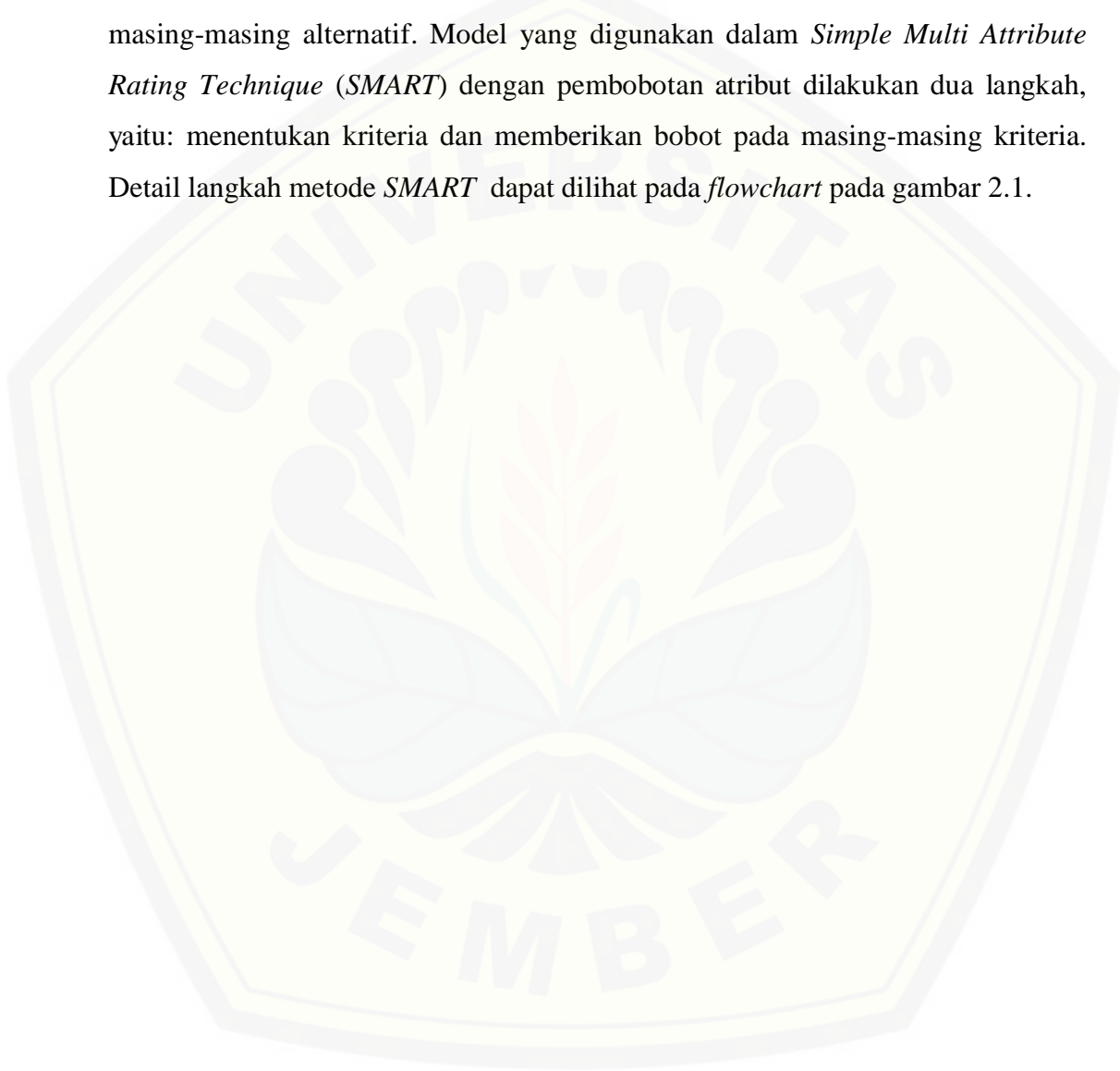
#### **2.4 Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG)**

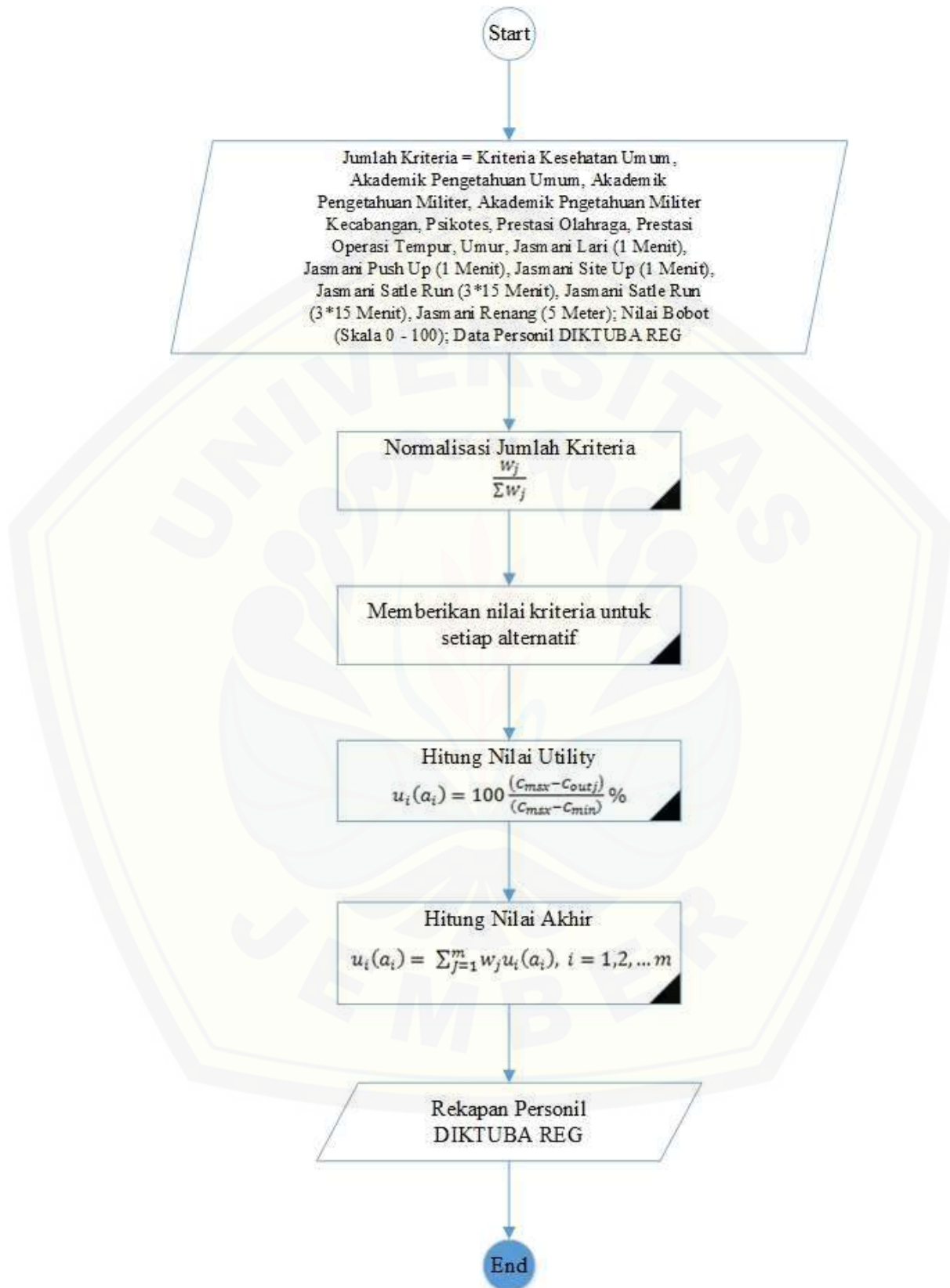
Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) Tentara Nasional Indonesia Angkatan Darat (TNI AD) merupakan pendidikan alih golongan dari Tamtama menjadi Bintara, bertujuan untuk membentuk Bintara Siswa menjadi Bintara Tentara Nasional Indonesia Angkatan Darat (TNI AD) yang memiliki sikap dan prilaku sebagai Prajurit Sapta Marga dan Sumpah Prajurit, berpengetahuan dan berketrampilan Dasar Golongan Bintara. Prajurit Golongan Bintara merupakan tulang punggung dan ujung tombak satuan serta pemimpin lapangan yang senantiasa berinteraksi langsung dengan bawahannya Sebagai ujung tombak pimpinan terdepan, harus memahami dan menyadari bahwa seorang pemimpin adalah sosok panutan bagi anak buah dan lingkungannya, yang senantiasa harus menjadi contoh tauladan dalam berfikir, bersikap dan bertindak didasari moral yang baik (Pussenarhanud Kodiklat TNI AD, 2015).

#### **2.5 Simple Multi-Attribute Rating Technique**

*Simple Multi Attribute Rating Technique (SMART)* merupakan metode pengambilan keputusan yang *multiatribut* yang dikembangkan oleh *Edward* pada tahun 1977. Teknik pembuatan keputusan multiatribut ini digunakan untuk mendukung pembuat keputusan dalam memilih antara beberapa alternatif. Setiap pembuat keputusan harus memilih sebuah alternatif yang sesuai dengan tujuan yang telah dirumuskan. Setiap alternatif terdiri dari sekumpulan atribut dan setiap atribut mempunyai nilai-nilai. Nilai ini dirata-rata dengan skala tertentu. Setiap

atribut mempunyai bobot yang menggambarkan seberapa penting dibandingkan dengan atribut lain. Pembobotan dan pemberian peringkat ini digunakan untuk menilai setiap alternatif agar diperoleh alternatif terbaik. Pembobotan pada *Simple Multi Attribute Rating Technique (SMART)* menggunakan skala antara 0 sampai 100, sehingga mempermudah perhitungan dan perbandingan nilai pada masing-masing alternatif. Model yang digunakan dalam *Simple Multi Attribute Rating Technique (SMART)* dengan pembobotan atribut dilakukan dua langkah, yaitu: menentukan kriteria dan memberikan bobot pada masing-masing kriteria. Detail langkah metode *SMART* dapat dilihat pada *flowchart* pada gambar 2.1.





Gambar 2.1 Flowchart Metode SMART



### **BAB 3. METODOLOGI PENELITIAN**

Pada bab metodologi penelitian dipaparkan beberapa hal yang meliputi jenis penelitian, alur penelitian, teknik pengumpulan data, model pengembangan sistem, dan gambaran umum sistem.

#### **3.1 Tempat dan Waktu Penelitian**

Penelitian dilakukan di Divisi Infanteri 2 Kostrad Singosari Malang. Waktu dilaksanakannya adalah selama 5 bulan yaitu pada bulan Oktober 2016 hingga Maret 2017.

#### **3.2 Jenis Penelitian**

Penelitian ini menggunakan jenis penelitian kualitatif dan jenis penelitian kuantitatif. Menggunakan penelitian jenis kualitatif karena pada penelitian ini menganalisa studi literatur yang berhubungan dengan indikator untuk menentukan personil mana yang lulus dalam seleksi Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG). Menggunakan penelitian jenis kuantitatif karena data yang diolah dalam bentuk angka.

#### **3.3 Alur Penelitian**

Alur penelitian merupakan urutan langkah penelitian yang dilakukan mulai dari studi literatur, pengumpulan data, pengolahan data, perancangan sistem, dan pengimplementasian rancangan sistem. Diagram alur pada penelitian ini dapat dilihat pada gambar 3.1 berikut.





Gambar 3.1 Alur Penelitian

### 3.4 Teknik Pengumpulan Data

Bagian ini menguraikan langkah-langkah yang ditempuh dan teknik yang digunakan untuk mengumpulkan data.

#### 3.1.1 Wawancara

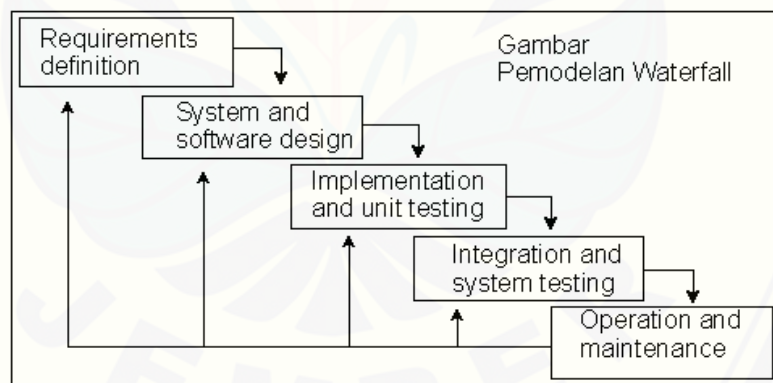
Wawancara adalah percakapan dengan maksud tertentu. Percakapan itu dilakukan oleh dua pihak, yaitu pewawancara (*interviewer*) yang mengajukan pertanyaan dan terwawancara (*interviewee*) yang memberikan jawaban atas pertanyaan itu (Moleong: 2011). Penelitian ini menggunakan teknik wawancara tidak terstruktur. Wawancara tidak berstruktur adalah wawancara yang hanya memuat garis besar pertanyaan dan narasumbernya atau terwawancara merupakan orang-orang tertentu yang memiliki sifat khas. Tujuan dari teknik wawancara dalam penelitian ini, yaitu untuk memperoleh informasi yang ada relevansinya dengan pokok persoalan penelitian mengenai latar belakang objek penelitian dan data berupa informasi yang diperlukan dalam membangun sistem informasi pengambilan keputusan Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG).

### 3.1.2 Studi Pustaka

Studi pustaka adalah teknik pencarian data dari arsip-arsip instansi dan teori yang berhubungan dengan permasalahan yang diangkat dalam penelitian ini. Dari hasil studi pustaka dapat menunjang dalam menguraikan data kualitatif berupa informasi mengenai seleksi Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) dan teori-teori yang digunakan dalam membangun sistem informasi pengambilan keputusan pemilihan peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA) pada studi kasus Divisi Infanteri 2 Kostrad Malang.

### 3.5 Model Pengembangan Sistem

Untuk membangun sebuah sistem dibutuhkan sebuah model dalam pengembangannya. Dalam penelitian ini model yang digunakan adalah model *waterfall*. Model ini dipakai karena umum digunakan dalam pengembangan *software* berskala kecil. Lima tahap dalam model *waterfall* sebagai berikut, yaitu *Requirement Analysis, System Design, Implimentation & Unit Testing, Integration & Testing, Operations & Maintenance*. Dapat dilihat pada gambar 3.2.



Gambar 3.2 Gambar Diagram tahapan model *waterfall*

#### 3.5.1 Analisis Kebutuhan

Tahap *requirement analysis* adalah tahap menganalisa apa yang dibutuhkan oleh sistem. Data kebutuhan dibagi menjadi kebutuhan fungsional dan non-fungsional. Pada sistem ini dibutuhkan juga data-data kriteria yang akan digunakan sebagai Perhitungan *SMART*. Data yang sudah terkumpul akan menentukan bagaimana fitur yang akan dibangun pada sistem. Berikut pada tabel

3.1 adalah kriteria yang digunakan dalam mengambil keputusan pemilihan peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG).

Tabel 3.1 Tabel Kriteria Pendidikan Pembentukan Bintara Reguler

No.	Penilaian		Bobot (%)
	Kriteria	Atribut	
1.	Kesehatan umum	Baik	3
		Tidak baik	
2.	Akademik pengetahuan umum	Baik (75 - 100)	10
		Sedang (55 - 74)	
		Rendah (<54)	
3.	Akademik pengetahuan militer	Baik (75 - 100)	10
		Sedang (55 - 74)	
		Rendah (<54)	
4.	Akademik pengetahuan militer Kecabangan	Baik (75 - 100)	10
		Sedang (55 - 74)	
		Rendah (<54)	
5.	Psikotes	Tinggi	2
		Sedang	
		Rendah	
6.	Prestasi olahraga	Tingkat internasional	10
		Tingkat nasional	
		Tingkat TNI	
		Tidak ada	
7.	Prestasi operasi tempur	Tingkat internasional	10
		Tingkat nasional	
		Tingkat TNI	
		Tidak ada	
8.	Umur	37 – 40 tahun	2

No.	Penilaian		Bobot (%)
	Kriteria	Atribut	
9.	Jasmani lari (1 menit)	34 – 36 tahun	10
		30 – 33 tahun	
		>3910 m	
		3710 – 3900 m	
		3510 – 3700 m	
10.	Jasmani <i>push up</i> (1 menit)	3200 – 3500 m	10
		<319 m	
		>40 kali	
		29 – 39 kali	
11.	Jasmani <i>site up</i> (1 menit)	17 – 28 kali	10
		<17 kali	
		>45 kali	
		36 – 45 kali	
12.	Jasmani <i>satle run</i> (3 * 15 menit)	24 – 35 kali	10
		<23 kali	
12.	Jasmani <i>satle run</i> (3 * 15 menit)	<14,59 menit	10
		15 – 17 menit	
12.	Jasmani <i>satle run</i> (3 * 15 menit)	18 – 20 menit	10
		>21 menit	
13.	Jasmani renang (5 meter)	<0,60 menit	3
		1,01 – 1,29 menit	
		1,30 – 1,59 menit	
		>2 menit	

### 3.5.2 Desain Sistem

Tahap *system design* adalah tahap yang dilakukan setelah data analisis telah terkumpul. Pada tahap desain ini menggunakan bahasa *Unified Modeling Language (UML)*. Bahasa ini digunakan karena mendukung konsep *Object*

*Oriented Programming (OOP)* sesuai dengan kode program yang akan digunakan. Beberapa diagram pada *UML* adalah sebagai berikut:

1. *Business Process*

*Business process* adalah diagram yang memperlihatkan inputan yang digunakan sistem, *output* yang dihasilkan sistem, dan tujuan pembuatan sistem.

2. *UseCase Diagram*

*Usecase diagram* adalah gambaran fitur dari sistem yang dijalankan oleh aktor. Pada diagram ini dapat dilihat juga hak akses dari aktor.

3. *Squence Diagram*

*Squence diagram* adalah diagram yang menggambarkan interaksi antara objek satu dengan yang lain di dalam sistem yang dibangun pada urutan waktu. Diagram juga menggambarkan interaksi antara aktor, fitur, dan data yang berjalan.

4. *Activity Diagram*

*Activity diagram* adalah penggambaran alir sistem yang akan dibangun, bagaimana sistem dari awal hingga sistem ditutup, serta bagaimana alir sistem ketika diimplementasikan dengan metode yang digunakan.

5. *Class Diagram*

*Class diagram* adalah diagram yang menggambarkan kelas-kelas dalam sebuah sistem dan hubungannya antara satu kelas dengan yang lain. Dalam kelas ini juga ditampilkan atribut dan operasi yang ada pada sistem.

6. *Entity Relation Diagram (ERD)*

*Entity Relation Diagram (ERD)* adalah diagram yang menggambarkan relasi objek-objek dasar data dalam sebuah basis data.

### 3.5.3 Implementasi (Pengkodean)

Desain yang sudah dibuat pada tahap sebelumnya digunakan sebagai acuan dalam pembangunan sistem. Sistem dibangun menggunakan bahasa pemrograman *java* dengan *tool* yang digunakan *netbeans* dan *database* yang digunakan adalah *DBMS XAMPP*. Proses pengkodean menggunakan konsep *Object Oriented Programming (OOP)*.



### 3.5.4 Pengujian

Tahap testing harus dilakukan sebelum sistem diserahkan kepada *user*. Tahap ini dilakukan agar dapat mengetahui apakah sistem yang dibangun sesuai dengan kebutuhan yang telah dianalisis diawal dan mengetahui apakah terdapat kesalahan pada sistem yang dibangun. Tahap *testing* dilakukan guna menyempurnakan sistem sebelum diserahkan kepada *user*. Pada tahap *testing* ini dilakukan pengujian dengan dengan metode *white-box* dan metode *black-box*.

#### 3.4.4.1 White-box Testing

*White-box testing* merupakan cara pengujian dengan melihat modul yang telah dibuat dengan program-program yang ada dan menganalisa apakah terjadi kesalahan atau tidak pada penulisan kode program. Pengujian *white-box* merupakan teknik pengujian jalur dasar yang digunakan untuk menentukan kompleksitas logis dengan menentukan rangkaian dasar jalur eksekusinya. Tahapan teknik pengujian jalur dasar meliputi dari *listing progame*, grafik alir, kompleksitas siklomatik, jalur program independen, dan pengujian basis set.

##### 1. Listing Progame

Merupakan baris-baris kode yang nantinya akan diuji. Setiap langkah dari kode-kode yang ada diberi nomor baik menjalankan *statement* biasa atau penggunaan kondisi dalam program. Contoh penerapan tahapan ini dapat dilihat pada gambar 3.2 di bawah ini.

```
Spanjang = $ POST['p']; (1)
Slebar   = $ POST['l'];
if(Spanjang == Slebar) (2)
{
    $jenisBangun = 'Persegi'; (3)
}
else
{
    $jenisBangun = 'Persegi Panjang'; (4)
}
$luas = Spanjang * Slebar;
echo 'Luas bangun '$jenisBangun.' adalah '$luas; (5)
```

Gambar 3.3 Contoh *Listing Progame*

Sumber: (Pressman: 2002)



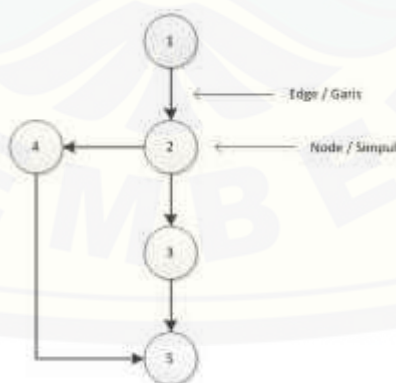
## 2. Grafik Alir

Notasi yang digunakan untuk menggambarkan jalur eksekusi adalah grafik alir (atau grafik program) yang menggunakan notasi lingkaran (simpul atau *node*) dan anak panah (*link* atau *edge*). Notasi ini menggambarkan aliran kontrol logika yang digunakan dalam suatu bahasa pemrograman.

Grafik alir merupakan sebuah notasi sederhana yang digunakan untuk mempresentasikan aliran kontrol (Pressman: 2002). Aliran kontrol yang digambarkan merupakan hasil penomoran dari *listing programe*. Grafik alir digambarkan dengan *node-node* (simpul) yang dihubungkan dengan *edge-edge* (garis) yang menggambarkan alur jalannya program. Contoh penggambaran diagram alir dapat dilihat pada gambar 3.3 di bawah ini.

## 3. Cyclomatic Complexity

*Cyclomatic complexity* adalah alat pengukuran untuk mengidentifikasi kompleksitas dari suatu program dengan cara menelusuri nomor dari jalur independen melalui *source code*-nya. Kompleksitas siklomatik merupakan metrik perangkat lunak yang menyediakan ukuran kuantitatif dari kompleksitas logis suatu program (Pressman: 2002). Rumus yang digunakan untuk menghitung kompleksitas siklomatik, yaitu:



Gambar 3.4 Contoh Diagram Alir

Sumber: (Pressman: 2002)

$$V(G) = E - N + 2 \dots\dots\dots(\text{persamaan 4})$$

Keterangan:

$V(G)$  : Kompleksitas siklomatik

$E$  : Jumlah *edge*  $N$  : Jumlah *node* Berdasarkan grafik alir yang ada pada tahapan kedua diketahui jumlah *edge* adalah 5 dan jumlah *node* adalah 5, sehingga dapat dihitung kompleksitas siklomatik  $V(G) = E - N + 2 = 5 - 5 + 2 = 2$ . Jadi jumlah jalur independen adalah 2 jalur.

#### 4. Jalur Program Independen

Jalur program independen atau *independent path* adalah alur dari manapun dalam program yang memperkenalkan sedikitnya satu kumpulan perintah pemrosesan atau kondisi baru (Pressman: 2002). Bila dinyatakan dalam grafik alir, jalur independen harus bergerak setidaknya sepanjang satu *edge* yang belum dilintasi sebelum jalur tersebut didefinisi (Pressman: 2002). Dari perhitungan kompleksitas siklomatik *Basis Set* pada gambar 3.3 yang dihasilkan dari jalur independen secara linier adalah 2 jalur, yaitu:

Jalur / *Path* 1 : 1-2-3-5

Jalur / *Path* 2 : 1-2-4-5

#### 5. Pengujian Basis Set

Pada bagian ini diberikan contoh data yang akan memaksa pelaksanaan jalur di *basis set*. Data yang dieksekusi dimasukkan ke dalam grafik alir apakah sudah melewati *basis set* yang tersedia. Sistem telah memenuhi syarat kelayakan *software* jika salah satu jalur yang dieksekusi setidaknya satu kali. Dari tahap sebelumnya telah diketahui 2 *basis set*. Jika kemudian diuji dengan memasukkan data panjang = 5 dan lebar = 3, maka *basis set* jalur yang digunakan adalah 1-2-4-5. Dapat dilihat bahwa jalur telah dieksekusi satu kali. Berdasarkan ketentuan tersebut dari segi kelayakan *software*, aplikasi ini telah memenuhi syarat.

#### 3.4.4.2 Black-box Testing

*Black-box testing* adalah menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan

untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi apakah sesuai dengan spesifikasi yang dibutuhkan (Shalahuddin & Rosa: 2011). Pengujian *black-box* seperti pada tabel 3.2 berikut:

Tabel 3. 2 Tabel Uji *Black-box*

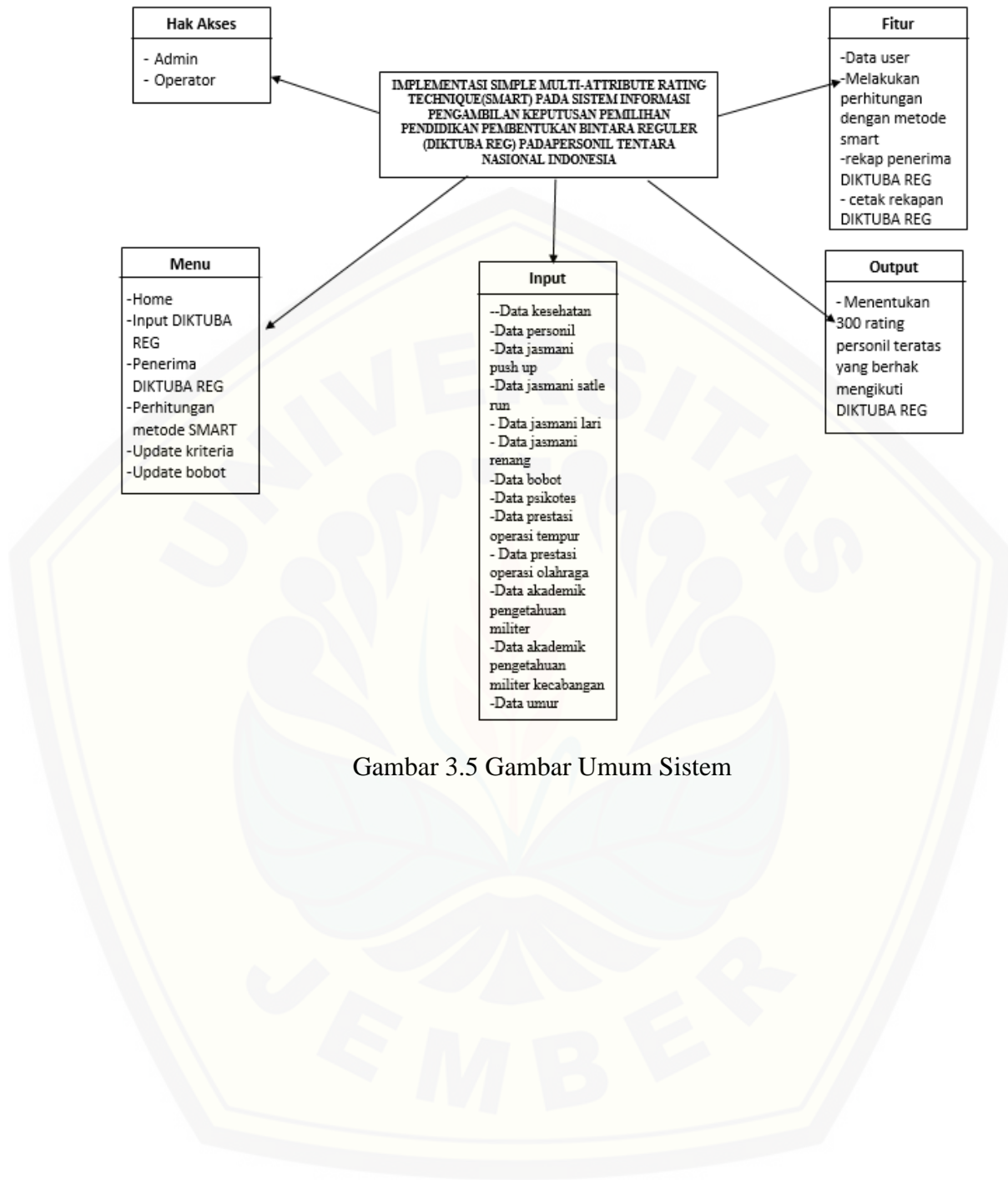
Kelas Uji	Skenario Uji	Hal yang Diharapkan	Kesimpulan

### 3.5.5 Pemeliharaan

Pemeliharaan aplikasi diperlukan ketika aplikasi telah digunakan oleh *user*. Ketika aplikasi dijalankan mungkin saja masih terjadi kesalahan atau *error* yang tidak ditemukan sebelumnya. Sehingga diperlukan perbaikan pada aplikasi tersebut.

### 3.6 Gambaran Umum Sistem

Sistem Pendukung Keputusan Pemilihan peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) Personil Tentara Nasional Indonesia adalah sistem yang akan dibangun guna membantu dalam mengambil keputusan agar lebih akurat dan terarah. Sistem ini menggunakan metode *Simple Multi-Attribute Rating Technique (SMART)*. Sistem ini memiliki fitur, antara lain fitur *login* untuk hak akses pengguna sistem, *input* data personil, input data kriteria, input nilai bobot, informasi rekapan hasil seleksi dengan nilai tertinggi hingga nilai minimal dan cetak rekapan hasil perhitungan. Dapat dilihat pada gambar 3.4.



Gambar 3.5 Gambar Umum Sistem

## BAB 4. PENGEMBANGAN SISTEM

Pada bab pengembangan sistem dipaparkan beberapa hal yang meliputi SOP, fungsi sistem, analisis kebutuhan sistem, desain sistem, implementasi coding, dan pengujian sistem.

### 4.1.1 *Statement Of Purpose (SOP)*

Sistem informasi diperlukan untuk pengambilan keputusan pemilihan peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) menggunakan metode *Simple Multi-Attribute Rating Technique (SMART)*. Pengambilan keputusan yang dilakukan dalam mengambil keputusan berdasarkan kriteria dan bobot yang telah diinputkan. Dari kriteria dan bobot yang telah diinputkan, sistem akan melakukan perhitungan menggunakan metode *Simple Multi-Attribute Rating Technique (SMART)* dengan memberikan *output* nilai tertinggi hingga nilai minimum menggunakan perhitungan data kriteria (kesehatan umum, akademik pengetahuan umum, akademik pengetahuan militer, akademik pengetahuan militer kecabangan, psikotes, prestasi olahraga, prestasi tertinggi operasi tempur, umur, jasmani lari, jasmani *push up*, jasmani *sitt up*, jasmani *satle run*, jasmani renang dan bobot) dan perhitungan numerik yang meliputi, normalisasi, nilai *utility* dan menghitung rata-rata. Setelah melakukan tahapan satu persatu maka akan dihasilkan rekapan dari perhitungan *Simple Multi-Attribute Rating Technique (SMART)*.

### 4.1.2 Fungsi Sistem

Sistem yang akan dibuat berupa sistem informasi pemilihan peserta Pembentukan Bintara Reguler (DIKTUBA REG) dengan memanfaatkan metode *Simple Multi-Attribute Rating Technique (SMART)* dengan studi kasus Divisi Infanteri 2 Kostrad Malang. Sistem ini merupakan sistem yang dapat memberikan pemecahan masalah yang terjadi untuk mengetahui personil mana yang lulus dengan mengurutkan nilai tertinggi hingga nilai minimum. Awalnya, Pembentukan Bintara Reguler (DIKTUBA REG) ini menggunakan sistem yang



bersifat manual. Tetapi pada sistem ini akan memberikan solusi perhitungan secara sistematis menggunakan metode *Simple Multi-Attribute Rating Technique (SMART)* agar lebih mudah dan akurat.

### 3.4 Analisis Kebutuhan Sistem

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

#### 3.4.4 Kebutuhan Fungsional

Kebutuhan fungsional aplikasi 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 pengambilan keputusan pembentukan bintanga reguler, yaitu:

- a. Sistem mampu mengelola data personil meliputi (tambah, *edit* dan hapus).
- b. Sistem mampu mengelola data kriteria meliputi (tambah, *edit* dan hapus).
- c. Sistem mampu mengelola data bobot meliputi (*edit* dan hapus).
- d. Sistem dapat menampilkan hasil dengan nilai tertinggi hingga nilai minimum menggunakan metode *Simple Multi-Attribute Rating Technique (SMART)*.
- e. Sistem ini dapat mencetak hasil rekapan nilai.
- f. Sistem *login* dengan *user password*.

#### 3.4.5 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 ini, yaitu :

1. Sistem dijalankan pada aplikasi dekstop PC/Laptop



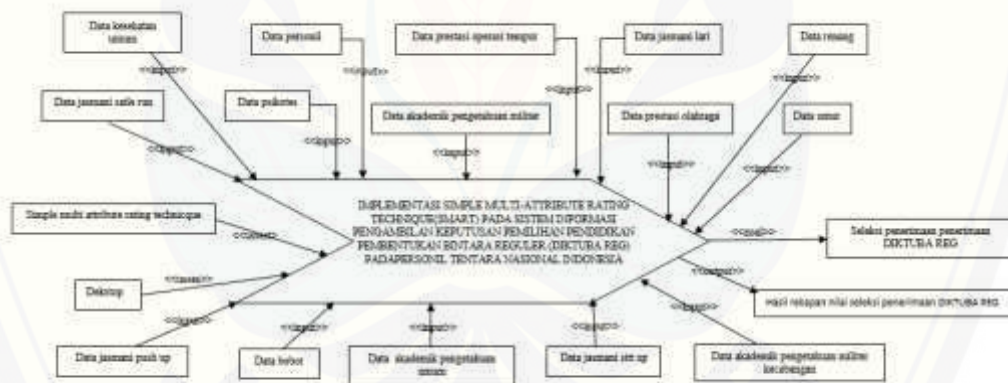
## 2. Tampilan sistem *user friendly*.

### 3.5 Desain Sistem

Tahapan yang dilakukan setelah melakukan analisis kebutuhan sistem yaitu tahap perencanaan pembangunan sistem yang dapat digambarkan dengan desain sistem. Desain sistem informasi pengambilan keputusan Pembentukan Bintara Reguler (DIKTUBA REG), meliputi *business process*, *usecase diagram*, *usecase skenario*, *activity diagram*, *sequence diagram*, dan *class diagram*.

#### 3.5.4 Business Process

*Business process* digunakan untuk menggambarkan sebuah proses suatu aplikasi secara keseluruhan, lengkap dengan *resources* dan informasi yang dibutuhkan seperti *input* dan *output* aplikasi, sehingga mendorong terjadinya proses dan tujuan yang telah ditentukan, seperti pada gambar 4.1.



Gambar 4.1 Gambar *Business Process* DIKTUBA REG

#### 3.5.5 UseCase Diagram

*Usecase diagram* merupakan model atau diagram yang menggambarkan fungsi atau tugas yang dilakukan *user*, baik manusia maupun sistem/komputer. *Usecase* digambarkan dari beberapa aktor, *usecase*, dan interaksi diantara komponen tersebut yang dapat memberikan informasi dari sistem yang akan

dibangun. Fitur-fitur pada sistem ini terdapat 18 fitur yang digambarkan dengan *elips* dan terdapat 2 tipe *user*.

*User* yang pertama, yaitu *admin*. *User* tipe ini memiliki beberapa fitur, yaitu mengelola data DIKTUBA REG, mengelola kriteria kesehatan umum, mengelola kriteria Akademik Pengetahuan umum, mengelola kriteria Akademik Pengetahuan militer, mengelola kriteria Akademik Pengetahuan militer kecabangan, mengelola kriteria psikotes, mengelola kriteria prestasi olahraga, mengelola kriteria prestasi tertinggi operasi tempur, mengelola kriteria umur, mengelola kriteria jasmani lari, mengelola kriteria jasmani *push up*, mengelola kriteria jasmani *sitt up*, mengelola kriteria jasmani *satle run*, mengelola kriteria jasmani renang, mengedit bobot melihat peneriam DIKTUBA REG dan mencetak hasil penerimaan. *User* kedua, yaitu operator. *User* tipe ini memiliki beberapa fitur, yaitu mengelola data DIKTUBA REG, perhitungan metode *Simple Multi Attribute Rating Technique (SMART)* DIKTUBA REG dan melihat penerimaan DIKTUBA REG. Pada gambar 4.2 digambarkan *usecase diagram* sistem informasi pengambilan keputusan pemilihan peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) yang akan dibangun.



Tabel 4.1 Tabel Definisi *Usecase Diagram*

No	UseCase	Deskripsi
1	<b>Mengelola Data DIKTUBA REG</b>	
	Tambah Personil	<i>Usecase yang menggambarkan proses admin dan operator menambahkan personil ke sistem.</i>
	Edit Personil	<i>Usecase yang menggambarkan proses admin dan operator mengubah personil yang disimpan dalam sistem.</i>
	Hapus Personil	<i>Usecase yang menggambarkan proses admin dan operator menghapus personil dari sistem.</i>
	Kosongkan	<i>Usecase yang menggambarkan proses admin dan operator mengosongkan form</i>
2	<b>Mengelola Data Kriteria Kesehatan Umum</b>	
	Tambah data	<i>Usecase yang menggambarkan proses admin menambahkan data ke sistem.</i>
	Edit data	<i>Usecase yang menggambarkan proses admin mengubah data yang disimpan dalam sistem.</i>
	Hapus data	<i>Usecase yang menggambarkan proses admin menghapus data dari sistem.</i>
	Kosongkan	<i>Usecase yang menggambarkan proses admin mengosongkan form</i>
3	<b>Mengelola Data Kriteria Akademik Pengetahuan Umum</b>	
	Tambah data	<i>Usecase yang menggambarkan proses admin menambahkan data ke sistem.</i>

No	UseCase	Deskripsi
	Ubah data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengubah data yang disimpan dalam sistem.
	Hapus data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menghapus data dari sistem.
	Kosongkan	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengosongkan form
<b>4</b>	<b>Mengelola Data Kriteria Akademik Pengetahuan Militer</b>	
	Lihat Simpanan	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menambahkan data ke sistem.
	Edit data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengubah data yang disimpan dalam sistem.
	Hapus Data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menghapus data dari sistem.
	Kosongkan	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengosongkan form
<b>5</b>	<b>Mengelola Data Kriteria Akademik Pengetahuan Militer</b>	
	Tambah Data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menambahkan data ke sistem.
	Edit data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengubah data yang disimpan dalam sistem.
	Hapus	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menghapus data dari sistem.
	Kosongkan	<i>Usecase</i> yang menggambarkan proses



No	UseCase	Deskripsi
		<i>admin</i> mengosongkan form
<b>6</b>	<b>Mengelola Data Kriteria Psikotes</b>	
	Tambah data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menambahkan data ke sistem.
	<i>Edit</i> data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengubah data yang disimpan dalam sistem.
	Hapus data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menghapus data dari sistem.
	Kosongkan	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengosongkan form
<b>7</b>	<b>Mengelola Data Kriteria Prestasi Olahraga</b>	
	Tambah data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menambahkan data ke sistem.
	<i>Edit</i> data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengubah data yang disimpan dalam sistem.
	Hapus data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menghapus data dari sistem.
	Kosongkan	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengosongkan form
<b>8</b>	<b>Mengelola Data Kriteria Prestasi Tertinggi Operasi Tempur</b>	
	Tambah data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menambahkan data ke sistem.
	<i>Edit</i> data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengubah data yang disimpan



No	UseCase	Deskripsi
	Hapus data	dalam sistem. <i>Usecase</i> yang menggambarkan proses <i>admin</i> menghapus data dari sistem.
	Kosongkan	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengosongkan form
9	<b>Mengelola Data Kriteria Umur</b>	
	Tambah data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menambahkan data ke sistem.
	Edit data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengubah data yang disimpan dalam sistem.
	Hapus data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menghapus data dari sistem.
	Kosongkan	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengosongkan form
10	<b>Mengelola Data Kriteria Jasmani Lari</b>	
	Tambah data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menambahkan data ke sistem.
	Edit data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengubah data yang disimpan dalam sistem.
	Hapus data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menghapus data dari sistem.
	Kosongkan	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengosongkan form

No	UseCase	Deskripsi
11	<b>Mengelola Data Kriteria Jasmani Push Up</b>	
	Tambah data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menambahkan data ke sistem.
	<i>Edit</i> data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengubah data yang disimpan dalam sistem.
	Hapus data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menghapus data dari sistem.
	Kosongkan	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengosongkan form
12	<b>Mengelola Data Kriteria Jasmani Sitt Up</b>	
	Tambah data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menambahkan data ke sistem.
	<i>Edit</i> data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengubah data yang disimpan dalam sistem.
	Hapus data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menghapus data dari sistem.
	Kosongkan	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengosongkan form
13	<b>Mengelola Data Kriteria Jasmani Satle Run</b>	
	Tambah data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> menambahkan data ke sistem.
	<i>Edit</i> data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mengubah data yang disimpan dalam sistem.
	Hapus data	<i>Usecase</i> yang menggambarkan proses

No	UseCase	Deskripsi
	Kosongkan	<p><i>admin</i> menghapus data dari sistem.</p> <p><i>Usecase</i> yang menggambarkan proses <i>admin</i> mengosongkan form</p>
14	<b>Mengelola Data Kriteria Jasmani Renang</b>	
	Tambah data	<p><i>Usecase</i> yang menggambarkan proses <i>admin</i> menambahkan data ke sistem.</p>
	Edit data	<p><i>Usecase</i> yang menggambarkan proses <i>admin</i> mengubah data yang disimpan dalam sistem.</p>
	Hapus data	<p><i>Usecase</i> yang menggambarkan proses <i>admin</i> menghapus data dari sistem.</p>
	Kosongkan	<p><i>Usecase</i> yang menggambarkan proses <i>admin</i> mengosongkan form</p>
15	<b>Mengedit Bobot</b>	
	Edit data	<p><i>Usecase</i> yang menggambarkan proses <i>admin</i> mengubah data yang disimpan dalam sistem.</p>
	Hapus data	<p><i>Usecase</i> yang menggambarkan proses <i>admin</i> menghapus data dari sistem.</p>
	Cek total nilai	<p><i>Usecase</i> yang menggambarkan proses <i>admin</i> mengecek total nilai.</p>
16	<b>Perhitungan Metode SMART DIKTUBA REG</b>	
	Tambah data	<p><i>Usecase</i> yang menggambarkan proses operator menambahkan data ke sistem.</p>
	Edit data	<p><i>Usecase</i> yang menggambarkan proses operator mengubah data yang disimpan dalam sistem.</p>
	Hapus data	<p><i>Usecase</i> yang menggambarkan proses</p>

No	UseCase	Deskripsi
	Kosongkan	operator menghapus data dari sistem. <i>Usecase</i> yang menggambarkan proses operator mengosongkan form
<b>17</b>	<b>Melihat Hasil Penerimaan DIKTUBA REG dan Mencetak</b>	
	Cari tahun angkatan	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mencari tahun angkatan.
	Filter jumlah nilai minimum	<i>Usecase</i> yang menggambarkan proses <i>admin</i> memasukkan nilai minimum.
	Cari personil	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mencari personil.
	Cetak data	<i>Usecase</i> yang menggambarkan proses <i>admin</i> mencetak data
<b>18</b>	<b>Melihat Penerimaan DIKTUBA REG</b>	
	Cari tahun angkatan	<i>Usecase</i> yang menggambarkan proses operator mencari tahun angkatan.
	Filter jumlah nilai minimum	<i>Usecase</i> yang menggambarkan proses operator memasukkan nilai minimum.
	Cari personil	<i>Usecase</i> yang menggambarkan proses operator mencari personil.

#### 4.3.3 Skenario

Skenario diagram digunakan untuk menjelaskan atau menceritakan fitur atau isi yang ada di *usecase* diagram. Skenario menjelaskan alur sistem dan keadaan yang akan terjadi ketika terjadi suatu *event* tertentu.

Skenario pada sistem informasi pengambilan keputusan pemilihan peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) dapat disajikan pada tabel 4.2 dibawah ini.

Tabel 4.2 Tabel Skenario Mengelola Data Perhitungan DIKTUBA REG

Nomor <i>Usecase</i>	UC-01
Nama	Mengelola data perhitungan DIKTUBA REG
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i>
<i>Post Condition</i>	Operator berhasil mengelola data DIKTUBA REG

**SKENARIO NORMAL MENGELOLA DATA PERHITUGAN METODE SMART  
DIKTUBA REG**

**TAMBAH DATA PERHITUGAN METODE SMART DIKTUBA REG**

Aktor	Sistem
1. Memilih menu Mengelola Perhitungan Metode <i>SMART</i> .	2. Menampilkan halaman Perhitungan Metode <i>SMART</i> DIKTUBA REG dengan tombol TAMBAH, <i>EDIT</i> , HAPUS, dan kosongkan.
3. Mengisi form Perhitungan Metode <i>SMART</i> DIKTUBA REG.	5. Menampilkan peringatan "PERSONIL CALON DIKTUBA REG BERHASIL DITAMBAH DAN MASUK SELEKSI PENERIMAAN".
4. Klik tombol TAMBAH.	
6. Klik tombol <i>OK</i>	7. Menampilkan halaman Perhitungan Metode <i>SMART</i> DIKTUBA REG dengan tombol TAMBAH, <i>EDIT</i> , HAPUS, dan kosongkan.

**Aktor****Sistem**

**SKENARIO ALTERNATIF TAMBAH DATA PERHITUNGAN METODE SMART  
DIKTUBA REG  
(DATA TIDAK LENGKAP)**

- 4a. Klik tombol TAMBAH
- 5a. Menampilkan peringatan “Nilai akhir belum tercapai, pilih semua kriteria yang ada!!”.

**SKENARIO ALTERNATIF TAMBAH DATA PERHITUNGAN METODE SMART  
DIKTUBA REG  
(KLIK TOMBOL KOSONGKAN)**

- 3b. Klik tombol kosongkan.
- 4b. Menampilkan halaman Perhitungan Metode *SMART* DIKTUBA REG dengan tombol TAMBAH, *EDIT*, HAPUS, dan kosongkan.

**SKENARIO NORMAL MENGELOLA DATA PERHITUNGAN METODE  
SMART DIKTUBA REG  
EDIT DATA PERHITUNGAN METODE SMART**

1. Memilih menu Mengelola Perhitungan Metode *SMART*.
2. Menampilkan halaman Perhitungan Metode *SMART* DIKTUBA REG dengan tombol TAMBAH, *EDIT*, HAPUS, dan kosongkan.
3. Pilih data yang akan di *edit*.
4. Menampilkan halaman Perhitungan Metode *SMART* DIKTUBA REG dengan tombol TAMBAH, *EDIT*, HAPUS, dan kosongkan.
5. Mengisi Form Calon Pendaftar DIKTUBA REG.



Aktor	Sistem
6. Klik tombol <i>EDIT</i> .	7. Menampilkan peringatan “PERSONIL CALON DIKTUBA REG BERHASIL DIEDIT DAN MASUK SELEKSI PENERIMAAN”.
8. Klik tombol <i>OK</i> .	9. Menampilkan halaman Perhitungan Metode <i>SMART</i> DIKTUBA REG dengan tombol <i>TAMBAH</i> , <i>EDIT</i> , <i>HAPUS</i> , dan kosongkan.

**SKENARIO ALTERNATIF EDIT DATA PERHITUNGAN METODE *SMART*  
DIKTUBA REG  
(DATA TIDAK LENGKAP!)**

6a. klik tombol <i>EDIT</i> .	7a. Menampilkan peringatan “Nilai akhir belum tercapai, pilih semua kriteria yang ada!!”.
-------------------------------	---

**SKENARIO ALTERNATIF EDIT DATA PERHITUNGAN METODE *SMART*  
DIKTUBA REG  
(KLIK TOMBOL KOSONGKAN)**

5b. Klik tombol kosongkan.	6b. Menampilkan halaman Perhitungan Metode <i>SMART</i> DIKTUBA REG dengan tombol <i>TAMBAH</i> , <i>EDIT</i> , <i>HAPUS</i> , dan kosongkan.
----------------------------	---

**SKENARIO NORMAL HAPUS DATA PERHITUNGAN METODE *SMART*  
DIKTUBA REG**

**HAPUS DATA PERHITUNGAN METODE *SMART* DIKTUBA REG**

1. Memilih menu Mengelola Perhitungan Metode <i>SMART</i> .	2. Menampilkan halaman Perhitungan Metode <i>SMART</i> DIKTUBA REG
---	--

Aktor	Sistem
3. Pilih data yang akan dihapus 4. Klik tombol HAPUS.	dengan tombol TAMBAH, <i>EDIT</i> , HAPUS, dan kosongkan.
6. Klik Tombol yes	5. Menampilkan peringatan “Apakah anda yakin?”
	7. Menampilkan halaman Perhitungan Metode <i>SMART</i> DIKTUBA REG dengan tombol TAMBAH, <i>EDIT</i> , HAPUS, dan kosongkan.

**SKENARIO ALTERNATIF EDIT DATA PERHITUNGAN METODE *SMART* DIKTUBA REG  
(KLIK TOMBOL *NO*)**

6a. Klik tombol No	7b. Menampilkan halaman Perhitungan Metode <i>SMART</i> DIKTUBA REG dengan tombol TAMBAH, <i>EDIT</i> , HAPUS, dan kosongkan.
--------------------	---

Tabel A.1 merupakan skenario dari *usecase* mengelola data perhitungan metode *SMART* DIKTUBA REG. Skenario mengelola data perhitungan metode *SMART* DIKTUBA REG. menjelaskan alur tambah, *edit*, hapus dan kosongkan. Skenario mengelola data perhitungan metode *SMART* DIKTUBA REG dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari tambah, *edit*, dan hapus. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap, Klik tombol *No* dan kosongkan. Kondisi setelah skenario ini dijalankan adalah Operator berhasil mengelola data DIKTUBA REG.

#### 4.3.4 Activity Diagram

*Activity diagram* menggambarkan aktivitas aktor dan sistem yang saling berhubungan dalam suatu aktivitas atau *event*. *Activity Diagram* menggambarkan berbagai alur aktivitas dalam sistem yang dirancang berawal *decision*. *Activity diagram* harus sesuai dengan skenario sistem yang telah dirancang. Sistem memberikan respon pada aktivitas yang dilakukan aktor. Berikut adalah penjelasan *activity diagram* dari sistem yang dirancang.

##### 4.3.4.1 Activity Diagram Mengelola Kesehatan Umum

*Activity diagram* menggambarkan alur aktivitas pada fitur mengelola kesehatan umum. Fitur mengelola kesehatan umum memiliki aktivitas utama, yaitu menambah, mnegubah, mengedit dan mengosongkan. Detail urutan aktifitas dalam mengelola kesehatan umum ditunjukkan pada lampiran gambar B.2.

##### 4.3.4.2 Activity Diagram Mengelola Data Diktuba REG

*Activity diagram* mengelola *data* DIKTUBA REG menggambarkan alur aktivitas pada fitur mengelola data DIKTUBA REG. Fitur mengelola *Data* DIKTUBA REG memiliki aktivitas utama, yaitu menambah, mengubah, dan menghapus. Detail urutan aktifitas dalam mengelola *user* ditunjukkan pada lampiran B gambar B.1.

##### 4.3.4.3 Activity Diagram Mengelola Kriteria Akademik Pengetahuan Umum

*Activity diagram* menggambarkan alur aktivitas pada fitur mengelola kriteria akademik pengetahuan umum. Fitur mengelola kriteria akademik pengetahuan umum memiliki aktivitas utama, yaitu menambah, mnegubah, mengedit dan mengosongkan. Detail urutan aktifitas dalam mengelola kriteria akademik pengetahuan umum ditunjukkan pada lampiran gambar B.3.

##### 4.3.4.4 Activity Diagram Mengelola Kriteria Akademik Pengetahuan Militer

*Activity diagram* menggambarkan alur aktivitas pada fitur mengelola kriteria akademik pengetahuan militer. Fitur mengelola kriteria akademik pengetahuan

militer memiliki aktivitas utama, yaitu menambah, mengubah, mengedit dan mengosongkan. Detail urutan aktifitas dalam mengelola kriteria akademik pengetahuan militer ditunjukkan pada lampiran gambar B.4.

#### 4.3.4.5 *Activity Diagram* Mengelola Kriteria Akademik Pengetahuan Militer

*Activity diagram* menggambarkan alur aktivitas pada fitur mengelola kriteria akademik pengetahuan militer kecabangan. Fitur mengelola kriteria akademik pengetahuan militer kecabangan memiliki aktivitas utama, yaitu menambah, mengubah, mengedit dan mengosongkan. Detail urutan aktifitas dalam mengelola kriteria akademik pengetahuan militer kecabangan ditunjukkan pada lampiran gambar B.5.

#### 4.3.4.6 *Activity Diagram* Mengelola Kriteria Psikotes

*Activity diagram* menggambarkan alur aktivitas pada fitur mengelola kriteria psikotes. Fitur mengelola kriteria psikotes memiliki aktivitas utama, yaitu menambah, mengubah, mengedit dan mengosongkan. Detail urutan aktifitas dalam mengelola kriteria psikotes ditunjukkan pada lampiran gambar B.6.

#### 4.3.4.7 *Activity Diagram* Mengelola Kriteria Prestasi Olahraga

*Activity diagram* menggambarkan alur aktivitas pada fitur mengelola kriteria prestasi olahraga. Fitur mengelola kriteria prestasi olahraga memiliki aktivitas utama, yaitu menambah, mengubah, mengedit dan mengosongkan. Detail urutan aktifitas dalam mengelola kriteria prestasi olahraga ditunjukkan pada lampiran gambar B.7.

#### 4.3.4.8 *Activity Diagram* Mengelola Kriteria Prestasi Operasi Tempur

*Activity diagram* menggambarkan alur aktivitas pada fitur mengelola kriteria Prestasi Operasi Tempur. Fitur mengelola kriteria Prestasi Operasi Tempur memiliki aktivitas utama, yaitu menambah, mengubah, mengedit dan mengosongkan. Detail urutan aktifitas dalam mengelola kriteria Prestasi Operasi Tempur ditunjukkan pada lampiran gambar B.8.

#### 4.3.4.9 *Activity Diagram* Mengelola Kriteria Umur

*Activity diagram* menggambarkan alur aktivitas pada fitur mengelola kriteria umur. Fitur mengelola kriteria umur memiliki aktivitas utama, yaitu menambah, mengubah, mengedit dan mengosongkan. Detail urutan aktifitas dalam mengelola kriteria umur ditunjukkan pada lampiran gambar B.9.

#### 4.3.4.10 *Activity Diagram* Mengelola Kriteria Jasmani Lari

*Activity diagram* menggambarkan alur aktivitas pada fitur mengelola kriteria jasmani lari. Fitur mengelola kriteria jasmani lari memiliki aktivitas utama, yaitu menambah, mengubah, mengedit dan mengosongkan. Detail urutan aktifitas dalam mengelola kriteria jasmani lari ditunjukkan pada lampiran gambar B.10.

#### 4.3.4.11 *Activity Diagram* Mengelola Kriteria Jasmani *Push Up*

*Activity diagram* menggambarkan alur aktivitas pada fitur mengelola kriteria jasmani *push up*. Fitur mengelola kriteria jasmani *push up* memiliki aktivitas utama, yaitu menambah, mengubah, mengedit dan mengosongkan. Detail urutan aktifitas dalam mengelola kriteria jasmani *push up* ditunjukkan pada lampiran gambar B.11

#### 4.3.4.12 *Activity Diagram* Mengelola Kriteria Jasmani *Sitt Up*

*Activity diagram* menggambarkan alur aktivitas pada fitur mengelola kriteria jasmani *sitt up*. Fitur mengelola kriteria jasmani *sitt up* memiliki aktivitas utama, yaitu menambah, mengubah, mengedit dan mengosongkan. Detail urutan aktifitas dalam mengelola kriteria jasmani *sitt up* ditunjukkan pada lampiran gambar B.12.

#### 4.3.4.13 *Activity Diagram* Mengelola Kriteria Jasmani *Satle Run*

*Activity diagram* menggambarkan alur aktivitas pada fitur mengelola kriteria jasmani *satle run*. Fitur mengelola kriteria jasmani *satle run* memiliki aktivitas utama, yaitu menambah, mengubah, mengedit dan mengosongkan. Detail urutan aktifitas dalam mengelola kriteria jasmani *satle run* ditunjukkan pada lampiran gambar B.13.



#### 4.3.4.14 *Activity Diagram* Mengelola Kriteria Jasmani Renang

*Activity diagram* menggambarkan alur aktivitas pada fitur mengelola kriteria jasmani renang. Fitur mengelola kriteria jasmani renang memiliki aktivitas utama, yaitu menambah, mengubah, mengedit dan mengosongkan. Detail urutan aktifitas dalam mengelola kriteria jasmani renang ditunjukkan pada lampiran gambar B.14.

#### 4.3.4.15 *Activity Diagram* Mengedit Bobot

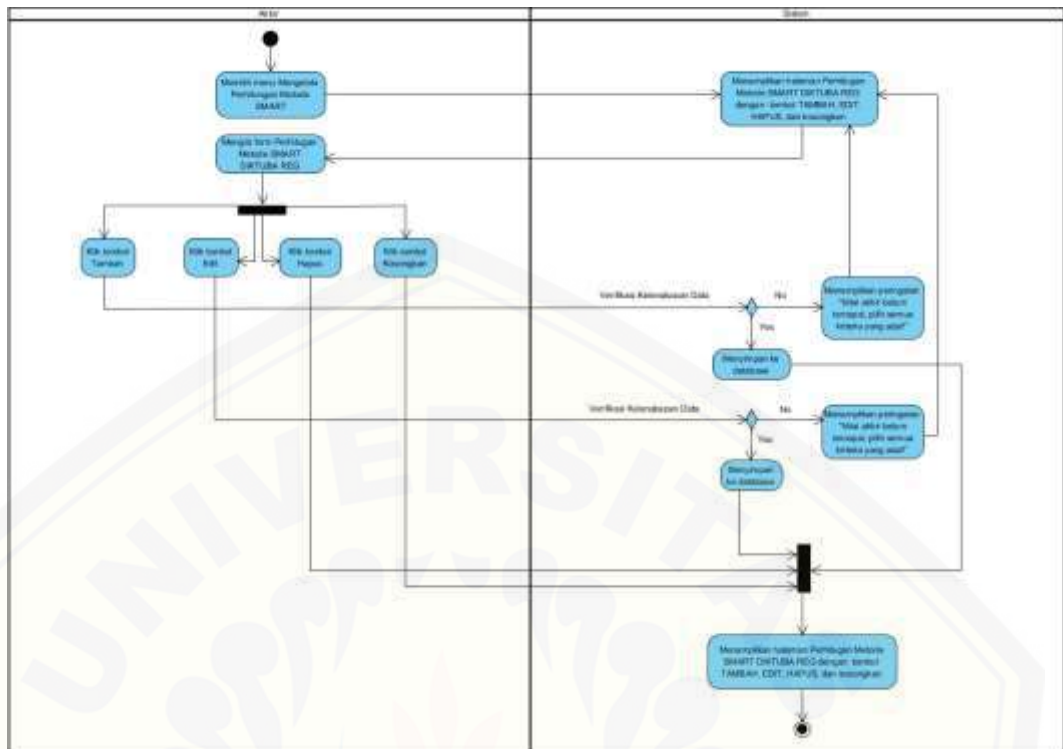
*Activity diagram* menggambarkan alur aktivitas pada fitur mengedit bobot. Fitur mengedit bobot memiliki aktivitas utama, yaitu mengedit, cek total nilai dan mengosongkan. Detail urutan aktifitas dalam mengedit bobot ditunjukkan pada lampiran gambar B.15.

#### 4.3.4.16 *Activity Diagram* Melihat Penerima DIKTUBA REG

*Activity diagram* menggambarkan alur aktivitas pada fitur melihat penerima DIKTUBA REG. Fitur melihat penerima DIKTUBA REG memiliki aktivitas utama, yaitu mencari. Detail urutan aktifitas dalam melihat penerima DIKTUBA REG ditunjukkan pada lampiran gambar B.16.

#### 4.3.4.17 *Activity Diagram* Perhitungan Metode *SMART*

*Activity diagram* menggambarkan alur aktivitas pada fitur perhitungan metode *Simple Multi Attribute Rating Technique (SMART)*. Fitur perhitungan metode *Simple Multi Attribute Rating Technique (SMART)* memiliki aktivitas utama, yaitu menambah, mengubah, mengedit dan mengosongkan.. Detail urutan aktifitas dalam perhitungan metode *Simple Multi Attribute Rating Technique (SMART)* ditunjukkan pada gambar 4.3.



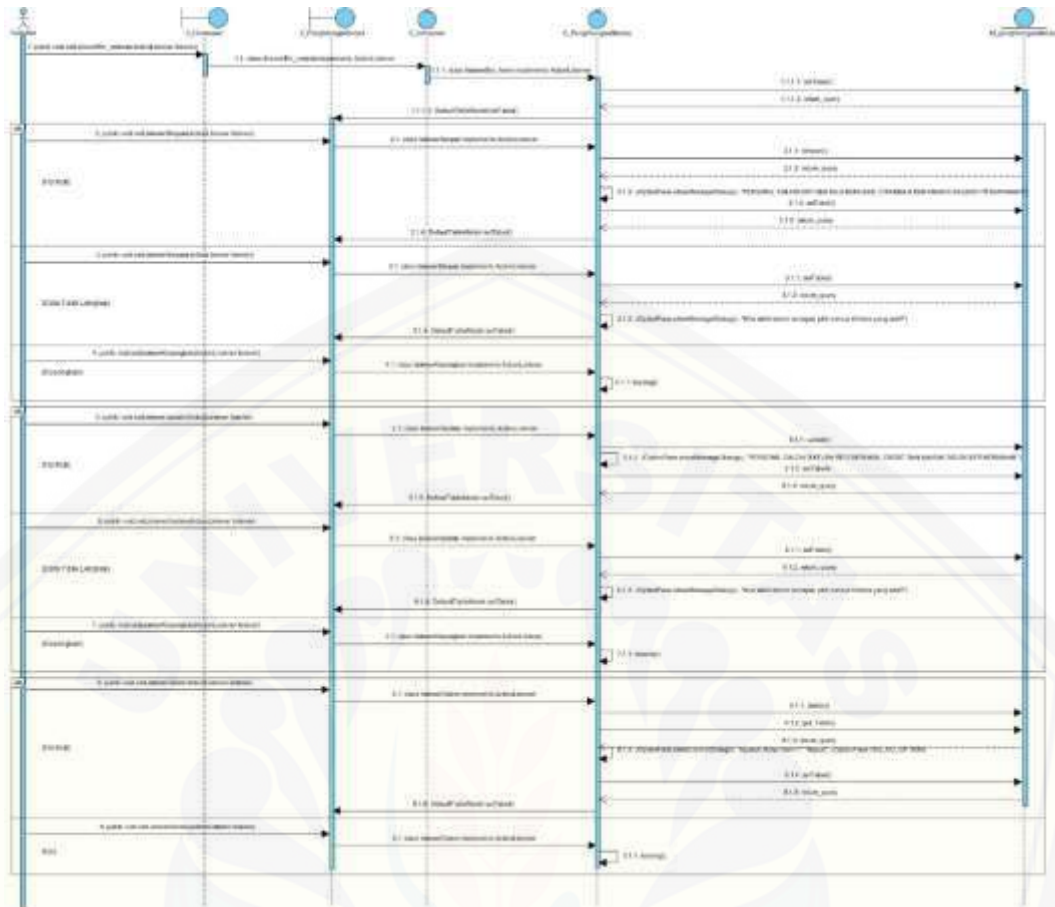
Gambar 4.3 Gambar *Activity Diagram* Perhitungan Metode SMART

#### 4.3.4 *Sequence Diagram*

*Sequence diagram* digunakan untuk menggambarkan perilaku pada sebuah skenario. Berikut adalah penjelasan *sequence diagram* dari sistem yang dirancang.

##### 4.3.4.1 *Sequence Diagram* Perhitungan Metode

*Sequence diagram* perhitungan metode digunakan untuk menunjukkan interaksi antar objek pada fitur perhitungan *Simple Multi Attribute Rating Technique (SMART)*. Objek yang terlibat pada fitur perhitungan *Simple Multi Attribute Rating Technique (SMART)* dengan aktor operator. Detail *sequence diagram* perhitungan *Simple Multi Attribute Rating Technique (SMART)* ditunjukkan pada gambar 4.4.



Gambar 4.4 Gambar *Sequence Diagram* Perhitungan SMART

#### 4.3.4.2 *Sequence Diagram* Operator Mengelola Data DIKTUBA REG

*Sequence diagram* mengelola data DIKTUBA REG digunakan untuk menunjukkan interaksi antar objek pada fitur mengelola data DIKTUBA REG. Objek yang terlibat pada fitur mengelola data DIKTUBA REG dengan aktor *Operator*, antara lain *view* (v\_adminHome;v\_userInputPenerima), *controller* (c\_userHome;c\_userInputPenerima), *model* (m\_userInputPenerima). Detail *sequence diagram* mengelola data DIKTUBA REG ditunjukkan pada lampiran C gambar C.2.

#### 4.3.4.3 *Sequence Diagram* Admin Mengelola Data DIKTUBA REG

*Sequence diagram* mengelola data DIKTUBA REG digunakan untuk menunjukkan interaksi antar objek pada fitur mengelola data DIKTUBA REG. Objek yang terlibat pada fitur mengelola data DIKTUBA REG dengan aktor

*Admin*, antara lain *view* (*v\_adminHome*; *v\_adminInputPenerima*), *controller* (*c\_adminHome*; *c\_adminInputPenerima*), *model* (*m\_adminInputPenerima*). Detail *sequence diagram* mengelola data DIKTUBA REG ditunjukkan pada lampiran C gambar C.1.

#### 4.3.4.4 *Sequence Diagram* Mengelola Kriteria Kesehatan Umum

*Sequence diagram* mengelola kriteria kesehatan umum digunakan untuk menunjukkan interaksi antar objek pada fitur mengelola kriteria kesehatan umum. Objek yang terlibat pada fitur mengelola data DIKTUBA REG dengan aktor *Admin*, antara lain *view* (*v\_adminHome*; *v\_atributkesehatanumum*), *controller* (*c\_adminHome*; *c\_atributkesehatanumum*), *model* (*m\_atributkesehatanumum*). Detail *sequence diagram* mengelola kriteria kesehatan umum ditunjukkan pada lampiran C gambar C.3.

#### 4.3.4.5 *Sequence Diagram* Mengelola Kriteria Akademik Pengetahuan Umum

*Sequence diagram* mengelola kriteria akademik pengetahuan umum digunakan untuk menunjukkan interaksi antar objek pada fitur mengelola kriteria akademik pengetahuan umum. Objek yang terlibat pada fitur mengelola data DIKTUBA REG dengan aktor *Admin*, antara lain *view* (*v\_adminHome*; *v\_atributakademikpengetahuanumum*), *controller* (*c\_adminHome*; *c\_atributakademikpengetahuanumum*), *model* (*m\_atributakademikpengetahuanumum*). Detail *sequence diagram* mengelola kriteria akademik pengetahuan umum ditunjukkan pada lampiran C gambar C.4.

#### 4.3.4.6 *Sequence Diagram* Mengelola Kriteria Akademik Pengetahuan Militer

*Sequence diagram* mengelola kriteria akademik pengetahuan militer digunakan untuk menunjukkan interaksi antar objek pada fitur mengelola kriteria akademik pengetahuan militer. Objek yang terlibat pada fitur mengelola data DIKTUBA REG dengan aktor *Admin*, antara lain *view* (*v\_adminHome*; *v\_atributakademikpengetahuanmiliter*), *controller* (*c\_adminHome*; *c\_atributakademikpengetahuanmiliter*), *model* (*m\_*

atributakademikpengetahuanmiliter). Detail *sequence diagram* mengelola kriteria akademik pengetahuan militer ditunjukkan pada lampiran C gambar C.5.

#### 4.3.4.7 *Sequence Diagram* Mengelola Kriteria Akademik Pengetahuan Militer Kecabangan

*Sequence diagram* mengelola kriteria akademik pengetahuan militer kecabangan digunakan untuk menunjukkan interaksi antar objek pada fitur mengelola kriteria akademik pengetahuan militer kecabangan. Objek yang terlibat pada fitur mengelola data DIKTUBA REG dengan aktor *Admin*, antara lain *view* (v\_adminHome;v\_atributakademikpengetahuanmiliterkecabangan), *controller* (c\_adminHome;c\_atributakademikpengetahuanmiliterkecabangan), *model* (m\_atributakademikpengetahuanmiliterkecabangan). Detail *sequence diagram* mengelola kriteria akademik pengetahuan militer kecabangan ditunjukkan pada lampiran C gambar C.6.

#### 4.3.4.8 *Sequence Diagram* Mengelola Kriteria Psikotes

*Sequence diagram* mengelola kriteria psikotes digunakan untuk menunjukkan interaksi antar objek pada fitur mengelola kriteria psikotes. Objek yang terlibat pada fitur mengelola data DIKTUBA REG dengan aktor *Admin*, antara lain *view* (v\_adminHome;v\_atributpsikotes), *controller* (c\_adminHome;c\_atributpsikotes), *model* (m\_atributpsikotes). Detail *sequence diagram* mengelola kriteria psikotes ditunjukkan pada lampiran C gambar C.7.

#### 4.3.4.9 *Sequence Diagram* Mengelola Kriteria Prestasi Olahraga

*Sequence diagram* mengelola kriteria prestasi olahraga digunakan untuk menunjukkan interaksi antar objek pada fitur mengelola kriteria prestasi olahraga. Objek yang terlibat pada fitur mengelola data DIKTUBA REG dengan aktor *Admin*, antara lain *view* (v\_adminHome;v\_atributprestasiolahraga), *controller* (c\_adminHome;c\_atributprestasiolahraga), *model* (m\_atributprestasiolahraga). Detail *sequence diagram* mengelola kriteria prestasi olahraga ditunjukkan pada lampiran C gambar C.8.



#### 4.3.4.10 *Sequence Diagram* Mengelola Kriteria Prestasi Operasi Tempur

*Sequence diagram* mengelola kriteria prestasi operasi tempur digunakan untuk menunjukkan interaksi antar objek pada fitur mengelola kriteria prestasi operasi tempur. Objek yang terlibat pada fitur mengelola data DIKTUBA REG dengan aktor *Admin*, antara lain *view* (*v\_adminHome*; *v\_atributprestasioperasitempur*), *controller* (*c\_adminHome*; *c\_atributprestasioperasitempur*), *model* (*m\_atributprestasioperasitempur*). Detail *sequence diagram* mengelola kriteria prestasi operasi tempur ditunjukkan pada lampiran C gambar C.9.

#### 4.3.4.11 *Sequence Diagram* Mengelola Kriteria Umur

*Sequence diagram* mengelola kriteria umur digunakan untuk menunjukkan interaksi antar objek pada fitur mengelola kriteria umur. Objek yang terlibat pada fitur mengelola data DIKTUBA REG dengan aktor *Admin*, antara lain *view* (*v\_adminHome*; *v\_atributumur*), *controller* (*c\_adminHome*; *c\_atributumur*), *model* (*m\_atributumur*). Detail *sequence diagram* mengelola umur ditunjukkan pada lampiran C gambar C.10.

#### 4.3.4.12 *Sequence Diagram* Mengelola Kriteria Jasmani Lari

*Sequence diagram* mengelola kriteria jasmani lari digunakan untuk menunjukkan interaksi antar objek pada fitur mengelola kriteria jasmani lari. Objek yang terlibat pada fitur mengelola data DIKTUBA REG dengan aktor *Admin*, antara lain *view* (*v\_adminHome*; *v\_atribjasmanilari*), *controller* (*c\_adminHome*; *c\_atributjasmanilari*), *model* (*m\_atributjasmanilari*). Detail *sequence diagram* mengelola jasmani lari ditunjukkan pada lampiran C gambar C.11.

#### 4.3.4.13 *Sequence Diagram* Mengelola Kriteria Jasmani *Push Up*

*Sequence diagram* mengelola kriteria jasmani *push up* digunakan untuk menunjukkan interaksi antar objek pada fitur mengelola kriteria jasmani *push up*. Objek yang terlibat pada fitur mengelola data DIKTUBA REG dengan aktor

*Admin*, antara lain *view* (v\_adminHome;v\_atribjasmanipushup), *controller* (c\_adminHome;c\_atributjasmanipushup), *model* (m\_ atributjasmanipushup). Detail *sequence diagram* mengelola jasmani *push up* ditunjukkan pada lampiran C gambar C.12.

#### 4.3.4.14 *Sequence Diagram* Mengelola Kriteria Jasmani *Sitt Up*

*Sequence diagram* mengelola kriteria jasmani *sitt up* digunakan untuk menunjukkan interaksi antar objek pada fitur mengelola kriteria jasmani *sitt up*. Objek yang terlibat pada fitur mengelola data DIKTUBA REG dengan aktor *Admin*, antara lain *view* (v\_adminHome;v\_atribjasmanisittup), *controller* (c\_adminHome;c\_atributjasmanisittup), *model* (m\_ atributjasmanisittup). Detail *sequence diagram* mengelola jasmani *sitt up* ditunjukkan pada lampiran C gambar C.13.

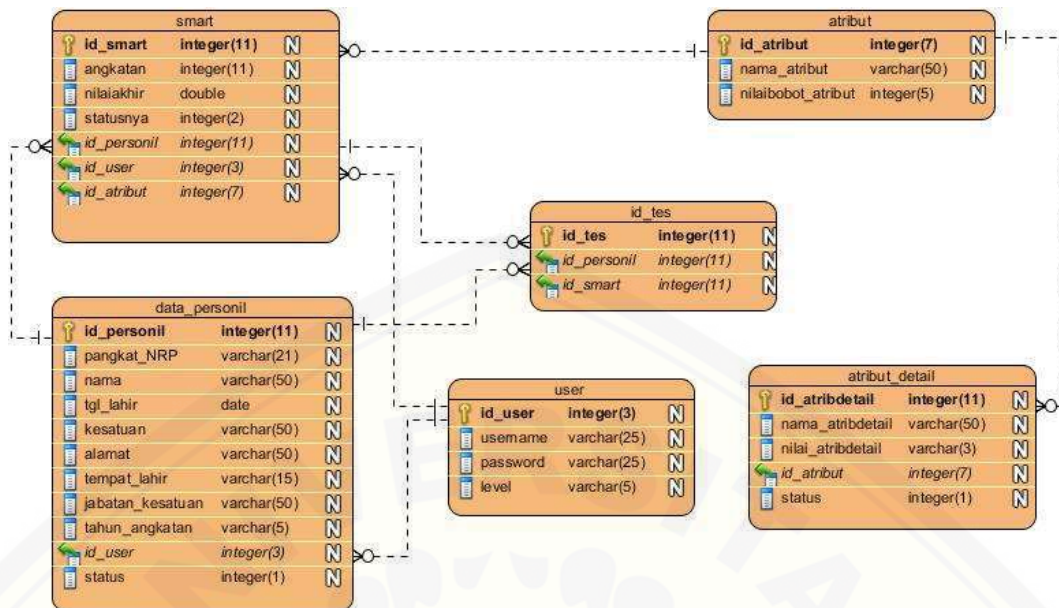
#### 4.3.4.15 *Sequence Diagram* Mengelola Kriteria Jasmani *Satle Run*

*Sequence diagram* mengelola kriteria jasmani *satle run* digunakan untuk menunjukkan interaksi antar objek pada fitur mengelola kriteria jasmani *satle run*. Objek yang terlibat pada fitur mengelola data DIKTUBA REG dengan aktor *Admin*, antara lain *view* (v\_adminHome;v\_atribjasmanisatlerun), *controller* (c\_adminHome;c\_atributjasmanisatlerun), *model* (m\_ atributjasmanisatlerun). Detail *sequence diagram* mengelola jasmani *satle run* ditunjukkan pada lampiran C gambar C.14.

#### 4.3.4.16 *Sequence Diagram* Mengelola Kriteria Jasmani Renang

*Sequence diagram* mengelola kriteria jasmani renang digunakan untuk menunjukkan interaksi antar objek pada fitur mengelola kriteria jasmani renang. Objek yang terlibat pada fitur mengelola data DIKTUBA REG dengan aktor *Admin*, antara lain *view* (v\_adminHome;v\_atribjasmanirenang), *controller* (c\_adminHome;c\_atributjasmanirenang), *model* (m\_ atributjasmanirenang). Detail *sequence diagram* mengelola jasmani renang ditunjukkan pada lampiran C gambar C.15.





Gambar 4.6 Gambar *Entity Relationship Diagram* DIKTUBA REG

#### 4.4 Implementasi koding

Setelah tahap desain perancangan selesai, tahap selanjutnya dalam penelitian ini yaitu tahap pengimplementasian desain perancangan ke dalam bahasa pemrograman. Bahasa pemrograman yang dipakai adalah bahasa pemrograman *Java* dan menggunakan *database MySQL*. Tahap implementasi perancangan ini menjelaskan tentang fitur – fitur yang terdapat pada sistem informasi pengambilan keputusan pemilihan peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG). Fitur-fitur tersebut yaitu mengelola data calon Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG), mengelola data Kriteria, mengelola dan data bobot. Pada tahap Perhitungan *SMART* menggunakan metode *Simple Multy Atribut Rating Technique* di dalam barisan kode program.

#### 4.5 Pengujian Sistem

Tahapan pengujian aplikasi 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 ini dilakukan dengan menggunakan dua metode, yaitu *white box* dan *black box*.

#### 4.5.1 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 fitur yang dinilai dapat mewakili sistem ini. Tahapan pengujian jalur dasar meliputi:

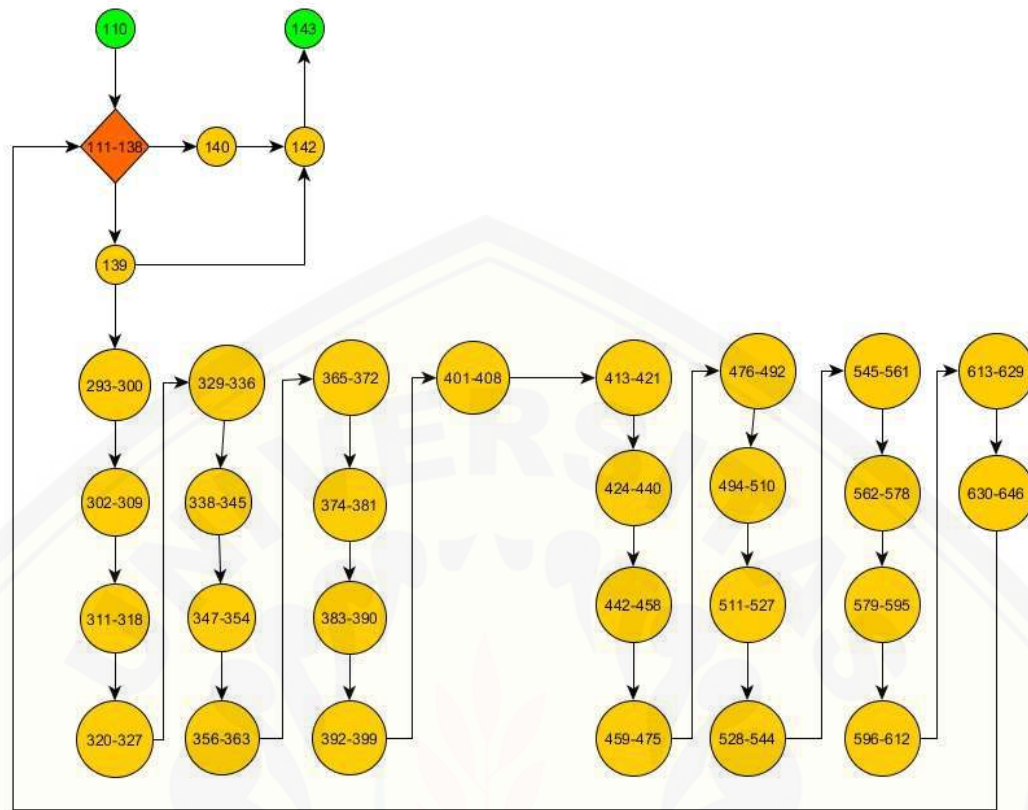
a. *Listing Programe*

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

b. Diagram Alir

Diagram alir merupakan notasi sederhana yang digunakan untuk merepresentasikan aliran kontrol. Aliran kontrol yang digambarkan merupakan hasil penomoran dari *listing* program. Diagram alir digambarkan dengan *node-node* (simpul) yang dihubungkan dengan *edge-edge* (garis) yang menggambarkan alur jalannya program. Diagram alir perhitungan DIKTUBA REG dapat dilihat pada gambar 4.7.





Gambar 4.7 Gambar Diagram Alir Metode *SMART* DIKTUBA REG

c. Kompleksitas Siklomatik

Kompleksitas siklomatik merupakan metrik perangkat lunak yang menyediakan ukuran kuantitatif dari kompleksitas logis suatu program. Bila digunakan dalam konteks teknik pengujian jalur dasar, nilai yang dihitung untuk kompleksitas siklomatik mendefinisikan jumlah jumlah jalur independen dalam basis set suatu program. Kompleksitas siklomatik DIKTUBA REG berdasarkan diagram alir sebagai berikut:

$$CC = \text{EDGE} - \text{NODE} + 2$$

$$CC = 33 - 33 + 2$$

$$CC = 2$$

d. Jalur Program Independen

Jalur independen adalah setiap jalur yang melalui program yang memperkenalkan setidaknya satu kumpulan pernyataan-pernyataan

pemrosesan atau kondisi baru. Bila dinyatakan dalam grafik alir, jalur independen harus bergerak setidaknya sepanjang satu *edge* yang belum dilintasi sebelum jalur tersebut didefinisi. Jalur program independen perhitungan *SMART* sebagai berikut:

Jalur 1 = 110-111-138-139-293-300-302-309-311-318-320-327-329-336-338-345-347-354-356-363-365-372-374-381-383-390-392-399-401-408-413-421-424-440-442-458-459-475-476-492-494-510-511-527-528-544-545-561-562-578-59-595-596-612-613-629-630-646-142-143

Jalur 2 = 110-111-138-140-142-143

#### e. Pengujian Basis Set

Pada bagian ini diberikan contoh data yang akan memaksa pelaksanaan jalur di basis set. Data yang dieksekusi dimasukkan ke dalam grafik alir apakah sudah melewati basis set yang tersedia. sistem telah memenuhi syarat kelayakan perangkat lunak jika salah satu jalur yang dieksekusi setidaknya satu kali. Pengujian basis set perhitungan *SMART* dapat dilihat pada tabel 4.3.

Tabel 4.3 Tabel *Test Case* Metode *SMART* DIKTUBA REG

Jalur 1	
<i>Test Case</i>	Jika nilai diinputkan maka nilai akhir akan ditampilkan
Target yang diharapkan	Menampilkan nilai akhir
HasilPengujian	Benar
Path/Jalur	110-111-138-139-293-300-302-309-311-318-320-327-329-336-338-345-347-354-356-363-365-372-374-381-383-390-392-399-401-408-413-421-424-440-442-458-459-475-476-492-494-510-511-527-528-544-545-561-562-578-59-595-596-612-613-629-630-646-142-143
Jalur 2	
<i>Test Case</i>	Jika nilai tidak diinputkan maka nilai akhir tidak ditampilkan
Target yang diharapkan	Menampilkan nilai akhir
HasilPengujian	Benar

---

Path/Jalur

110-111-138-140-142-143

---

#### 4.5.2 Metode *Black Box*

Pengujian *black-box* merupakan pengujian yang berfokus pada persyaratan fungsional perangkat lunak dengan berusaha menemukan kesalahan seperti fungsi-fungsi yang tidak benar atau hilang, kesalahan *interface* (Pressman, 2012). Pengujian *black box* berfungsi untuk menguji aplikasi 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 kepala panitia Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) Kostrad Malang. Hasil pengujian dengan metode *black box* dapat dilihat pada lampiran

## BAB 6. 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. Sistem Informasi Pengambilan Keputusan Pemilihan Peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) memiliki 2 hak akses yaitu *admin* dan *operator*. Peran *admin* adalah mengelola data DIKTUBA REG, mengelola kriteria, mengedit bobot, melihat penerimaan DIKTUBA REG serta cetak rekapan hasil. Peran *operator* adalah mengelola data DIKTUBA REG, mengelola perhitungan metode, melihat penerimaan DIKTUBA REG. Tahap pembangunan dan pengembangannya sistem ini menggunakan model pengembangan waterfall yang dapat digunakan dalam pembangunan sistem skala kecil, sehingga dokumentasi pengembangan dapat terorganisir dengan baik.
2. Proses Pemilihan Peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) dapat lebih diperkuat dengan mengkombinasikan metode Simple Multy Rating Technique (SMART) sebagai pembantu. SMART diimplementasikan pada Pemilihan Peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) dengan membuat sebuah kriteria yang akan digunakan sebagai pembanding dalam penilaian, kriteria tersebut juga harus memiliki sebuah atribut untuk menunjang penilaian pada setiap kriteria. dalam penelitian ini kriteria yang dipakai berjumlah tiga belas kriteria dengan masing-masing atribut. Pada tahapan selanjutnya membuat bobot untuk setiap kriteria yang digunakan yang bertujuan untuk pembanding tingkat kepentingan antar bobot. Dengan adanya data tersebut metode SMART dapat mengolah data dengan menggunakan data yang ada dan diproses dalam

metode sehingga dapat menghasilkan pendukung keputusan Pemilihan Peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG).

3. Sistem Informasi Pengambilan Keputusan Pemilihan Peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) yang dibangun dengan mengimplementasikan metode SMART didalamnya dapat lebih mudah dan lebih dapat menghasilkan hasil yang akurat, karena pada perhitungan metode SMART yang dilakukan secara manual dapat diimplementasikan secara benar pada sistem, jadi proses Pemilihan Peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) yang untuk direkomendasi dapat dengan cepat serta akurat sesuai penilaian yang ada. Disamping itu Sistem Informasi Pengambilan Keputusan Pemilihan Peserta Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) dapat menyesuaikan penilaian yang akan dipakai, jika sewaktu waktu atau setiap tahun berubah makan bobot bisa disesuaikan dengan penilaian yang diberikan.
4. Sistem Informasi Pengambilan Keputusan Pendidikan Pembentukan Bintara Reguler (DIKTUBA REG) menghasilkan sejumlah personel Tentara Nasional Indonesia (TNI) yang lulus dengan nilai minimum yang telah ditetapkan.

## 6.2 Saran

Pengembangan lebih lanjut Pengembangan lebih lanjut untuk penelitian ini dapat dilakukan dengan membangun sistem prediksi dalam berbagai platform seperti web, android, iOS, dan windows phone agar dapat menginputkan data secara real-time. Selain itu disarankan pengembangan sistem peneliti selanjutnya dapat menambahkan fitur tambah kriteria dan sistem terintegrasi keseluruhan batalyon yang ada di Indonesia.



**DAFTAR PUSTAKA**

- A. S. Rosa & Shalahuddin, M. (2011). *“Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)”*. Bandung: Modula.
- Agissa, W. (2013). *“White Box and Black Box Testing”*. Jakarta: PT Elex Media Komputindo.
- Atiqah. (2013). *“Implementasi Metode SMART Pada Sistem Pendukung Keputusan Pemilihan Pembelian Mobil Keluarga”*. Program Studi Teknik Informatika STMIK Budi Darma Medan: Jl. Sisingamangaraja No. 338 Sp. Limun Medan.
- Dudung. (2015). *“12 Pengertian Dan Fungsi Sistem Informasi Menurut Para Ahli”*. <http://dosenpendidikan.com/12pengertiandanfungsisisteminformasi-menurutparaahli.html>. (1 Oktober 2016).
- Moleong, L. J. (2011). *“Metodologi Penelitian Kualitatif”*. Bandung: PT Remaja Rosdakarya.
- Monica, Elena, dkk. (2015). *“Sistem Pendukung Keputusan Penentuan Status Kesehatan Bayi Baru Lahir Berdasarkan Pemeriksaan Antropometri Dengan Metode Weighted Product (Studi Kasus: RS. Bersalin Muhammadiyah Cirebon)”*. Jurnal Online ICT STMIK IKMI. Vol. 14 No. 2. Desember 2015. <http://stmik-ikmi-cirebon.net/e-journal/>. (5 Oktober 2016).
- Nasution, A. N. (2013). *“Penerapan Metode Simple Multi Attribute Rating Technique Untuk Penentuan Bintang Nararia Bhayangkara (Studi Kasus: Kepolisian Daerah Riau)”*. Pekanbaru Riau: Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim.
- PENGKOSTRAD. (2009). *“Sejarah Difisi Infanteri 2 Kostrad”*.
- Pressman, R. S. (2002). *“Rekayasa Perangkat Lunak Pendekatan Praktisi”*. Yogyakarta: Andi.
- Pussenarhanud Kodiklat TNI AD. (2015). *“Upacara Pembukaan DIKTUKBA ARHANUD TNI AD Tahap I TA 2015”*. 5 January 2016.

- Rouf, A. (2012). *“Pengujian Perangkat Lunak Dengan Menggunakan Metode White Box Dan Black Box”*. Semarang: Sistem Informasi – STMIK HIMSYA.
- Setiawan, A. (2010). *“Pemodelan Software Dengan Metode Waterfall Dan Programming: Studi Perbandingan”*. Program Sistem Informasi, Universitas Bina Dharma.
- Setyawan, A. (2015). *“Sistem Pendukung Keputusan Pemilihan Mahasiswa, page 7-8”*. Kediri: Universitas Nusantara PGRI.
- Staf Penerangan Divisi Infanteri 2 Kostrad. (2016). *“Sejarah Difisi Infanteri 2 Kostrad”*. <http://divif2kostrad.com/sejarah-divisi-infanteri-2-kostrad/>. (4 September 2016).
- Turban. (2005). *“Dessicion Support System and Intellegent System”*. Yogyakarta: Andi.
- Yunitarini, R. (2013). *“Sistem Pendukung Keputusan Pemilihan Penyiar RadioTerbaik, page 45”*. Madura: Fakultas Teknik Universitas Trunojoyo Program Studi Teknik Informatika.

## LAMPIRAN

## Lampiran A. Skenario

## A.1 Skenario Mengelola Data DIKTUBA REG

Tabel A.1 Skenario Mengelola Data DIKTUBA REG

Nomor <i>UseCase</i>	UC-01
Nama	Mengelola data DIKTUBA REG
Aktor	<i>Admin</i> dan Operator
<i>Pre Condition</i>	<i>Admin</i> dan Operator harus sudah <i>login</i>
<i>Post Condition</i>	<i>Admin</i> dan Operator berhasil mengelola data DIKTUBA REG

## SKENARIO NORMAL MENGELOLA DATA DIKTUBA REG

## TAMBAH DATA DIKTUBA REG

Aktor	Sistem
8. Memilih menu Mengelola DIKTUBA REG.	9. Menampilkan halaman Calon Pendaftar DIKTUBA REG dengan tombol TAMBAH, <i>EDIT</i> , HAPUS, dan kosongkan.
10. Mengisi form Calon Pendaftar DIKTUBA REG.	
11. Klik tombol TAMBAH.	12. Menampilkan peringatan “INPUT DATA Personil BERHASIL !!!”.
13. Klik tombol <i>OK</i>	14. Menampilkan halaman Calon

**Aktor****Sistem**

Pendaftar DIKTUBA REG dengan tombol TAMBAH, *EDIT*, HAPUS, dan Kosongkan.

**SKENARIO ALTERNATIF TAMBAH DATA DIKTUBA REG  
(DATA TIDAK LENGKAP)**

3a. Klik tombol TAMBAH

4a. Menampilkan peringatan “data tidak lengkap”.

**SKENARIO ALTERNATIF TAMBAH DATA DIKTUBA REG  
(KLIK TOMBOL KOSONGKAN)**

3b. Klik tombol kosongkan.

4b. Menampilkan halaman Calon Pendaftar DIKTUBA REG dengan tombol TAMBAH, *EDIT*, HAPUS, dan kosongkan.

**SKENARIO NORMAL MENGELOLA DATA DIKTUBA REG  
*EDIT* DATA DIKTUBA REG**

10. Memilih menu Mengelola DIKTUBA REG.

11. Menampilkan halaman Calon Pendaftar DIKTUBA REG dengan tombol TAMBAH, *EDIT*, HAPUS, dan kosongkan.

12. Pilih data yang akan di *edit*.

13. Menampilkan halaman Calon Pendaftar DIKTUBA REG dengan tombol TAMBAH, *EDIT*, HAPUS, dan kosongkan.

14. Mengisi Form Calon Pendaftar DIKTUBA REG.

**Aktor**

**Sistem**

15. Klik tombol *EDIT*.

16. Menampilkan peringatan “UPDATE DATA BERHASIL !!!”.

17. Klik tombol *OK*.

18. Menampilkan halaman Calon Pendaftar DIKTUBA REG dengan tombol TAMBAH, *EDIT*, HAPUS, dan kosongkan.

**SKENARIO ALTERNATIF *EDIT* DATA DIKTUBA REG  
(DATA TIDAK LENGKAP!)**

6a. klik tombol *EDIT*.

7a. Menampilkan peringatan “DATA TIDAK LENGKAP!”

**SKENARIO ALTERNATIF *EDIT* DATA DIKTUBA REG  
(KLIK TOMBOL KOSONGKAN)**

5b. Klik tombol kosongkan.

6b. Menampilkan halaman Calon Pendaftar DIKTUBA REG dengan tombol TAMBAH, *EDIT*, HAPUS, dan kosongkan.

**SKENARIO NORMAL HAPUS DATA DIKTUBA REG  
HAPUS DATA DIKTUBA REG**

8. Memilih menu Mengelola DIKTUBA REG.

9. Menampilkan halaman Calon Pendaftar DIKTUBA REG dengan tombol TAMBAH, *EDIT*, HAPUS, dan kosongkan.

10. Pilih data yang akan dihapus



Aktor	Sistem
11. Klik tombol HAPUS.	12. Menampilkan halaman Calon Pendaftar DIKTUBA REG dengan tombol TAMBAH, <i>EDIT</i> , HAPUS, dan kosongkan.

Tabel A.1 merupakan skenario dari *usecase* mengelola data DIKTUBA REG. Skenario mengelola data DIKTUBA REG. menjelaskan alur tambah, *edit*, hapus dan kosongkan. Skenario mengelola data DIKTUBA REG dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari tambah, *edit*, dan hapus. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap dan kosongkan. Kondisi setelah skenario ini dijalankan adalah *Admin* dan Operator berhasil mengelola data DIKTUBA REG.

#### A.2 Skenario Mengelola Kriteria Kesehatan Umum

Tabel A.2 Tabel Skenario Mengelola Kriteria Kesehatan Umum

Nomor <i>UseCase</i>	UC-02
Nama	Mengelola Kriteria Kesehatan Umum
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i> sistem
<i>Post Condition</i>	Operator berhasil Mengelola Kriteria Kesehatan Umum

#### SKENARIO NORMAL MENGELOLA KRITERIA KESEHATAN UMUM TAMBAH KRITERIA KESEHATAN UMUM

Aktor	Sistem
1. Memilih menu Mengelola Kriteria.	2. Menampilkan halaman Kesehatan

**Aktor****Sistem**

- Umum dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
3. Mengisi Form Kesehatan Umum.
  4. Klik tombol Tambah.
  5. Menampilkan peringatan “TAMBAH DATA BERHASIL !!!”.
  6. Klik *OK*.
  7. Menampilkan halaman Kesehatan Umum dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO ALTERNATIF TAMBAH KRITERIA KESEHATAN UMUM  
(DATA BELUM LENGKAP)**

- 4a. Klik tombol Tambah.
- 5a. Menampilkan peringatan ”Data Belum Lengkap”

**SKENARIO ALTERNATIF TAMBAH KRITERIA KESEHATAN UMUM  
(KLIK TOMBOL KOSONGKAN)**

- 3b. Klik tombol Kosongkan
- 4b. Menampilkan halaman Kesehatan Umum dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO NORMAL MENGELOLA KRITERIA KESEHATAN UMUM  
*EDIT* KRITERIA KESEHATAN UMUM**

1. Memilih menu Mengelola Kriteria.
2. Menampilkan halaman Kesehatan Umum dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
3. Pilih data yang akan di *edit*.
4. Menampilkan halaman Kesehatan Umum dengan tombol Tambah,

**Aktor****Sistem**

- |  |   |
|--|---|
| <p>5. Mengisi form kesehatan umum.</p> <p>6. Klik tombol <i>Edit</i>.</p> <p>8. Klik tombol OK</p> | <p><i>Edit</i>, Hapus dan Kosongkan.</p> <p>7. Menampilkan peringatan “UPDATE DATA BERHASIL !!!”</p> <p>9. Menampilkan halaman Kesehatan Umum dengan tombol Tambah, <i>Edit</i>, Hapus dan Kosongkan.</p> |
|--|---|

**SKENARIO ALTERNATIF *EDIT* KRITERIA KESEHATAN UMUM  
(DATA BELUM LENGKAP)**

- |                                     |   |
|-------------------------------------|---|
| <p>6a. Klik tombol <i>Edit</i>.</p> | <p>7a. Menampilkan peringatan “Data Belum Lengkap!!!”</p> |
|-------------------------------------|---|

**SKENARIO ALTERNATIF *EDIT* KRITERIA KESEHATAN UMUM  
(KLIK TOMBOL KOSONGKAN)**

- |  |   |
|--|---|
| <p>5b. Klik tombol kesehatan umum.</p> | <p>6b. Menampilkan halaman Kesehatan Umum dengan tombol Tambah, <i>Edit</i>, Hapus dan Kosongkan.</p> |
|--|---|

**SKENARIO NORMAL HAPUS KRITERIA KESEHATAN UMUM  
(KLIK TOMBOL DETAIL JAMINAN)**

- |   |  |
|---|--|
| <p>1. Memilih menu Mengelola Kriteria.</p> <p>3. Pilih data yang akan dihapus</p> <p>4. Klik tombol Hapus</p> | <p>2. Menampilkan halaman Kesehatan Umum dengan tombol Tambah, <i>Edit</i>, Hapus dan Kosongkan.</p> <p>5. Menampilkan halaman Kesehatan Umum dengan tombol Tambah, <i>Edit</i>,</p> |
|---|--|
-

**Aktor**

**Sistem**

Hapus dan Kosongkan.

Tabel A.2 merupakan skenario dari *usecase* mengelola Kriteria Kesehatan Umum. Skenario mengelola Kriteria Kesehatan Umum menjelaskan alur tambah, *edit*, hapus, dan kosongkan. Skenario mengelola Kriteria Kesehatan Umum dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari tambah, *edit*, hapus dan kosongkan. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap. Kondisi setelah skenario ini dijalankan adalah *admin* berhasil mengelola Kriteria Kesehatan Umum.

A.3 Skenario Mengelola Kriteria Akademik Pengetahuan Umum

Tabel A.3 Skenario Mengelola Kriteria Akademik Pengetahuan Umum

Nomor <i>UseCase</i>	UC-03
Nama	Mengelola Kriteria Akademik Pengetahuan Umum
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i> sistem
<i>Post Condition</i>	Operator berhasil Mengelola Kriteria Akademik Pengetahuan Umum

**SKENARIO NORMAL MENGELOLA KRITERIA AKADEMIK  
PENGETAHUAN UMUM**

**TAMBAH KRITERIA AKADEMIK PENGETAHUAN UMUM**

**Aktor**

**Sistem**

- Memilih menu Kriteria

      Akademik Pengetahuan umum

- Menampilkan halaman Akademik Pengetahuan Umum.

Dengan tombol Tambah, *Edit* ,  
Hapus dan Kosongkan

3. Mengisi Form Akademik  
Pengetahuan Umum.
4. Klik tombol Tambah.

5. Menampilkan peringatan  
“TAMBAH DATA BERHASIL  
!!!”.

6. Klik *OK*.

- i. 7. Menampilkan halaman  
Akademik Pengetahuan Umum  
dengan tombol Tambah, *Edit*,  
Hapus dan Kosongkan.

**SKENARIO ALTERNATIF TAMBAH KRITERIA AKADEMIK  
PENGETAHUAN UMUM  
(DATA BELUM LENGKAP)**

- 4a. Klik tombol Tambah.

- 5a. Menampilkan peringatan ”Data  
Belum Lengkap”

**SKENARIO ALTERNATIF TAMBAH KRITERIA AKADEMIK  
PENGETAHUAN UMUM  
(KLIK TOMBOL KOSONGKAN)**

- 3b. Klik tombol Kosongkan

- 4b. Menampilkan halaman Kesehatan  
Umum dengan tombol Tambah,  
*Edit*, Hapus dan Kosongkan.



**SKENARIO NORMAL MENGELOLA KRITERIA AKADEMIK  
PENGETAHUAN UMUM**

***EDIT* KRITERIA AKADEMIK PENGETAHUAN UMUM**

1. Klik tombol Akademik Pengetahuan Umum.
2. Menampilkan halaman Akademik Pengetahuan Umum dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
3. Pilih data yang akan di *edit*.
4. Menampilkan halaman Akademik Pengetahuan Umum dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
5. Mengisi form Akademik Pengetahuan Umum.
6. Klik tombol *Edit*.
7. Menampilkan peringatan “UPDATE DATA BERHASIL !!!”
8. Klik tombol OK
9. Menampilkan halaman Akademik Pengetahuan Umum dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO ALTERNATIF *EDIT* KRITERIA AKADEMIK  
PENGETAHUAN UMUM  
(DATA BELUM LENGKAP)**

6a. Klik tombol *Edit*.

7a. Menampilkan peringatan “Data  
Belum Lengkap!!!”

**SKENARIO ALTERNATIF KRITERIA AKADEMIK PENGETAHUAN  
UMUM  
(KLIK TOMBOL KOSONGKAN)**

5b. Klik tombol *Edit*.

6b. Menampilkan halaman Akademik  
Pengetahuan Umum dengan  
tombol Tambah, *Edit*, Hapus dan  
Kosongkan.

**SKENARIO NORMAL KRITERIA AKADEMIK PENGETAHUAN  
UMUM  
KLIK HAPUS**

1. Klik tombol Akademik  
Pengetahuan Umum.

2. Menampilkan halaman  
Akademik Pengetahuan Umum  
dengan tombol Tambah, *Edit*,  
Hapus dan Kosongkan.

3. Pilih data yang akan dihapus

4. Klik tombol Hapus

5. Menampilkan halaman  
Akademik Pengetahuan Umum  
dengan tombol Tambah, *Edit*,

---

Hapus dan Kosongkan.

Tabel A.3 merupakan skenario dari *usecase* mengelola Kriteria Akademik Pengetahuan Umum. Skenario mengelola Kriteria akademik pengetahuan umum menjelaskan alur tambah, *edit*, hapus, dan kosongkan. Skenario mengelola Kriteria Akademik Pengetahuan Umum dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari tambah, *edit* dan hapus. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap dan kosongkan. Kondisi setelah skenario ini dijalankan adalah *admin* berhasil mengelola Kriteria Akademik Pengetahuan Umum.

#### A.4 Skenario Mengelola Kriteria Akademik Pengetahuan Militer

Tabel A.4 Skenario Mengelola Kriteria Akademik Pengetahuan Militer

Nomor <i>UseCase</i>	UC-04
Nama	Mengelola Kriteria Akademik Pengetahuan Militer
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i> sistem
<i>Post Condition</i>	Operator berhasil Mengelola Kriteria Akademik Pengetahuan Militer

**SKENARIO NORMAL MENGELOLA KRITERIA AKADEMIK  
PENGETAHUAN MILITER  
TAMBAH KRITERIA AKADEMIK PENGETAHUAN MILITER**

**Aktor**

**Sistem**

1. Memilih menu Kriteria Akademik Pengetahuan Militer.
2. Menampilkan halaman Akademik Pengetahuan Militer.

Dengan tombol Tambah, *Edit*,  
Hapus dan Kosongkan.

3. Mengisi Form Akademik  
Pengetahuan Militer.
4. Klik tombol Tambah.

5. Menampilkan peringatan  
“TAMBAH DATA BERHASIL  
!!!”.

6. Klik *OK*.

- i. 7. Menampilkan halaman  
Akademik Pengetahuan Militer  
dengan tombol Tambah, *Edit*,  
Hapus dan Kosongkan.

**SKENARIO ALTERNATIF TAMBAH AKADEMIK PENGETAHUAN  
MILITER  
(DATA BELUM LENGKAP)**

- 4a. Klik tombol Tambah.

- 5a. Menampilkan peringatan ”Data  
Belum Lengkap”

**SKENARIO ALTERNATIF TAMBAH KRITERIA AKADEMIK  
PENGETAHUAN MILITER  
(KLIK TOMBOL KOSONGKAN)**

- 3b. Klik tombol Kosongkan

- 4b. Menampilkan halaman Akademik  
Pengetahuan Militer dengan tombol  
Tambah, *Edit*, Hapus dan  
Kosongkan.

**SKENARIO NORMAL MENGELOLA KRITERIA AKADEMIK  
PENGETAHUAN MILITER**

***EDIT* KRITERIA AKADEMIK PENGETAHUAN MILITER**

1. Klik tombol Akademik Pengetahuan Militer.
2. Menampilkan halaman Akademik Pengetahuan Militer dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
3. Pilih data yang akan di *edit*.
4. Menampilkan halaman Akademik Pengetahuan Militer dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
5. Mengisi form akademik pengetahuan umum.
6. Klik tombol *Edit*.
7. Menampilkan peringatan “UPDATE DATA BERHASIL !!!”
8. Klik tombol OK
9. Menampilkan halaman Akademik Pengetahuan Militer dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO ALTERNATIF *EDIT* KRITERIA AKADEMIK  
PENGETAHUAN MILITER  
(DATA BELUM LENGKAP)**



6a. Klik tombol *Edit*.

7a. Menampilkan peringatan “Data  
Belum Lengkap!!!”

**SKENARIO ALTERNATIF KRITERIA AKADEMIK PENGETAHUAN  
MILITER  
(KLIK TOMBOL KOSONGKAN)**

5b. Klik tombol *Edit*.

6b. Menampilkan halaman Akademik  
Pengetahuan Militer dengan tombol  
Tambah, *Edit*, Hapus dan  
Kosongkan.

**SKENARIO NORMAL KRITERIA AKADEMIK PENGETAHUAN  
MILITER  
KLIK HAPUS**

1. Klik tombol Akademik  
Pengetahuan Militer.

2. Menampilkan halaman Akademik  
Pengetahuan Militer dengan tombol  
Tambah, *Edit*, Hapus dan  
Kosongkan.

3. Pilih data yang akan dihapus

4. Klik tombol Hapus

5. Menampilkan halaman Akademik  
Pengetahuan Militer dengan tombol  
Tambah, *Edit*, Hapus dan  
Kosongkan.

---

Tabel A.4 merupakan skenario dari *usecase* mengelola Kriteria Akademik Pengetahuan Militer. Skenario mengelola Kriteria Akademik Pengetahuan Militer menjelaskan alur tambah, *edit*, hapus, dan kosongkan. Skenario mengelola Kriteria Akademik Pengetahuan Militer dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari tambah, *edit* dan hapus. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap dan kosongkan. Kondisi setelah skenario ini dijalankan adalah *admin* berhasil mengelola Kriteria Akademik Pengetahuan Militer.

#### A.5 Skenario Mengelola Kriteria Akademik Pengetahuan Militer Kecabangan

Tabel A.5 Skenario Mengelola Kriteria Akademik Pengetahuan Militer Kecabangan

Nomor <i>UseCase</i>	UC-05
Nama	Mengelola Kriteria Akademik Pengetahuan Militer Kecabangan
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i> sistem
<i>Post Condition</i>	Operator berhasil Mengelola Kriteria Akademik Pengetahuan Militer Kecabangan

**SKENARIO NORMAL MENGELOLA KRITERIA AKADEMIK  
PENGETAHUAN MILITER KECABANGAN  
TAMBAH KRITERIA AKADEMIK PENGETAHUAN MILITER  
KECABANGAN**

**Aktor**

**Sistem**

1. Memilih menu Kriteria Akademik Pengetahuan Militer Kecabangan.
2. Menampilkan halaman Akademik

Pengetahuan Militer Kecabangan.  
Dengan tombol Tambah, *Edit* ,  
Hapus dan Kosongkan

3. Mengisi Form Akademik

Pengetahuan Militer Kecabangan.

4. Klik tombol Tambah.

5. Menampilkan peringatan

“TAMBAH DATA BERHASIL  
!!!”.

6. Klik *OK*.

7. Menampilkan halaman Akademik  
Pengetahuan Militer Kecabangan  
dengan tombol Tambah, *Edit*,  
Hapus dan Kosongkan.

**SKENARIO ALTERNATIF TAMBAH AKADEMIK PENGETAHUAN  
MILITER KECABANGAN  
(DATA BELUM LENGKAP)**

4a. Klik tombol Tambah.

5a. Menampilkan peringatan ”Data  
Belum Lengkap”

**SKENARIO ALTERNATIF TAMBAH KRITERIA AKADEMIK  
PENGETAHUAN MILITER KECABANGAN  
(KLIK TOMBOL KOSONGKAN)**

3b. Klik tombol Kosongkan

4b. Menampilkan halaman Akademik  
Pengetahuan Militer Kecabangan  
dengan tombol Tambah, *Edit*,

Hapus dan Kosongkan.

**SKENARIO NORMAL MENGELOLA KRITERIA AKADEMIK  
PENGETAHUAN MILITER KECABANGAN  
EDIT KRITERIA AKADEMIK PENGETAHUAN MILITER  
KECABANGAN**

1. Klik tombol Akademik  
Pengetahuan Militer  
Kecabangan.
2. Menampilkan halaman  
Akademik Pengetahuan Militer  
Kecabangan dengan tombol  
Tambah, *Edit*, Hapus dan  
Kosongkan.
3. Pilih data yang akan di *edit*.
4. Menampilkan halaman  
Akademik Pengetahuan Militer  
Kecabangan dengan tombol  
Tambah, *Edit*, Hapus dan  
Kosongkan.
5. Mengisi form akademik  
pengetahuan umum  
Kecabangan.
6. Klik tombol *Edit*.
7. Menampilkan peringatan  
“UPDATE DATA BERHASIL  
!!!”
8. Klik tombol OK
9. Menampilkan halaman

Akademik Pengetahuan Militer  
Kecabangan dengan tombol  
Tambah, *Edit*, Hapus dan  
Kosongkan.

**SKENARIO ALTERNATIF *EDIT* KRITERIA AKADEMIK PENGETAHUAN  
MILITER KECABANGAN  
(DATA BELUM LENGKAP)**

6a. Klik tombol *Edit*.

7a. Menampilkan peringatan “Data  
Belum Lengkap!!!”

**SKENARIO ALTERNATIF KRITERIA AKADEMIK PENGETAHUAN  
MILITER KECABANGAN  
(KLIK TOMBOL KOSONGKAN)**

5b. Klik tombol *Edit*.

6b. Menampilkan halaman Akademik  
Pengetahuan Militer Kecabangan  
dengan tombol Tambah, *Edit*,  
Hapus dan Kosongkan.

**SKENARIO NORMAL KRITERIA AKADEMIK PENGETAHUAN MILITER  
KECABANGAN  
KLIK HAPUS**

1. Klik tombol Akademik  
Pengetahuan Militer  
Kecabangan.

2. Menampilkan halaman  
Akademik Pengetahuan Militer  
Kecabangan dengan tombol  
Tambah, *Edit*, Hapus dan  
Kosongkan.



3. Pilih data yang akan dihapus
4. Klik tombol Hapus
5. Menampilkan halaman Akademik Pengetahuan Militer Kecabangan dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

Tabel A.5 merupakan skenario dari *usecase* mengelola Kriteria Akademik Pengetahuan Militer Kecabangan. Skenario mengelola Kriteria Akademik Pengetahuan Militer menjelaskan alur tambah, *edit*, hapus, dan kosongkan. Skenario mengelola Kriteria Akademik Pengetahuan Militer Kecabangan dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari tambah, *edit* dan hapus. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap dan kosongkan. Kondisi setelah skenario ini dijalankan adalah *admin* berhasil mengelola Kriteria Akademik Pengetahuan Militer Kecabangan.

#### A.6 Skenario Mengelola Kriteria Psikotes

Tabel A.6 Skenario Mengelola Kriteria Psikotes

Nomor <i>UseCase</i>	UC-06
Nama	Mengelola Kriteria Psikotes
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i> sistem
<i>Post Condition</i>	Operator berhasil Mengelola Kriteria Psikotes

#### SKENARIO NORMAL MENGELOLA KRITERIA PSIKOTES

##### TAMBAH KRITERIA PSIKOTES

**Aktor**

**Sistem**

1. Memilih menu Kriteria Psikotes.

2. Menampilkan halaman Psikotes.  
Dengan tombol Tambah, *Edit* ,  
Hapus dan Kosongkan
3. Mengisi Form Psikotes.
4. Klik tombol Tambah.
  5. Menampilkan peringatan  
“TAMBAH DATA BERHASIL  
!!!”.
6. Klik *OK*.
  7. Menampilkan halaman Psikotes  
dengan tombol Tambah, *Edit*,  
Hapus dan Kosongkan.

**SKENARIO ALTERNATIF TAMBAH PSIKOTES  
(DATA BELUM LENGKAP)**

- 4a. Klik tombol Tambah.
  - 5a. Menampilkan peringatan ”Data  
Belum Lengkap”

**SKENARIO ALTERNATIF TAMBAH KRITERIA PSIKOTES  
(KLIK TOMBOL KOSONGKAN)**

- 3b. Klik tombol Kosongkan
  - 4b. Menampilkan halaman Psikotes  
dengan tombol Tambah, *Edit*,  
Hapus dan Kosongkan.

**SKENARIO NORMAL MENGELOLA KRITERIA PSIKOTES  
*EDIT* KRITERIA PSIKOTES**

1. Klik tombol Psikotes.
  2. Menampilkan halaman Psikotes  
dengan tombol Tambah, *Edit*,

- Hapus dan Kosongkan.
3. Pilih data yang akan di *edit*.
  4. Menampilkan halaman Psikotes dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
  5. Mengisi form Psikotes.
  6. Klik tombol *Edit*.
  7. Menampilkan peringatan “UPDATE DATA BERHASIL !!!”
  8. Klik tombol OK
  9. Menampilkan halaman Psikotes dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO ALTERNATIF *EDIT* KRITERIA PSIKOTES  
(DATA BELUM LENGKAP)**

- 6a. Klik tombol *Edit*.
- 7a. Menampilkan peringatan “Data Belum Lengkap!!!”

**SKENARIO ALTERNATIF KRITERIA PSIKOTES  
(KLIK TOMBOL KOSONGKAN)**

- 5b. Klik tombol *Edit*.
- 6b. Menampilkan halaman Psikotes dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO NORMAL KRITERIA PSIKOTES  
KLIK HAPUS**

1. Klik tombol Psikotes.

2. Menampilkan halaman Psikotes dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
3. Pilih data yang akan dihapus
4. Klik tombol Hapus
5. Menampilkan halaman Psikotes dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

---

Tabel A.6 merupakan skenario dari *usecase* mengelola Kriteria Psikotes. Skenario mengelola Kriteria Psikotes menjelaskan alur tambah, *edit*, hapus, dan kosongkan. Skenario mengelola Kriteria Psikotes dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari tambah, *edit* dan hapus. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap dan kosongkan. Kondisi setelah skenario ini dijalankan adalah *admin* berhasil mengelola Kriteria Psikotes.

#### A.7 Skenario Mengelola Kriteria Prestasi Olahraga

Tabel A.7 Skenario Mengelola Kriteria Prestasi Olahraga

---

Nomor <i>UseCase</i>	UC-07
Nama	Mengelola Kriteria Prestasi Olahraga
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i> sistem
<i>Post Condition</i>	Operator berhasil Mengelola Prestasi Olahraga

**SKENARIO NORMAL MENGELOLA KRITERIA PRESTASI  
OLAHRAGA  
TAMBAH KRITERIA PSIKOTES**

Aktor	Sistem
1. Memilih menu Kriteria Prestasi Olahraga.	2. Menampilkan halaman Prestasi Olahraga. Dengan tombol Tambah, <i>Edit</i> , Hapus dan Kosongkan
3. Mengisi Form Prestasi Olahraga.	
4. Klik tombol Tambah.	5. Menampilkan peringatan “TAMBAH DATA BERHASIL !!!”.
6. Klik <i>OK</i> .	7. Menampilkan halaman Prestasi Olahraga dengan tombol Tambah, <i>Edit</i> , Hapus dan Kosongkan.

**SKENARIO ALTERNATIF TAMBAH PRESTASI OLAHRAGA  
(DATA BELUM LENGKAP)**

4a. Klik tombol Tambah.	5a. Menampilkan peringatan ”Data Belum Lengkap”
-------------------------	---

**SKENARIO ALTERNATIF TAMBAH KRITERIA PRESTASI  
OLAHRAGA  
(KLIK TOMBOL KOSONGKAN)**

3b. Klik tombol Kosongkan	4b. Menampilkan halaman Prestasi
---------------------------	----------------------------------



Olahraga dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO NORMAL MENGELOLA KRITERIA PRESTASI  
OLAHRAGA  
*EDIT* KRITERIA PRESTASI OLAHRAGA**

1. Klik tombol Prestasi Olahraga.
2. Menampilkan halaman Prestasi Olahraga dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
3. Pilih data yang akan di *edit*.
4. Menampilkan halaman Prestasi Olahraga dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
5. Mengisi form Prestasi Olahraga.
6. Klik tombol *Edit*.
7. Menampilkan peringatan “UPDATE DATA BERHASIL !!!”
8. Klik tombol OK
9. Menampilkan halaman Prestasi Olahraga dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO ALTERNATIF *EDIT* KRITERIA PRESTASI OLAHRAGA  
(DATA BELUM LENGKAP)**

- 6a. Klik tombol *Edit*.
- 7a. Menampilkan peringatan “Data

Belum Lengkap!!!”

**SKENARIO ALTERNATIF KRITERIA PRESTASI OLAHRAGA  
(KLIK TOMBOL KOSONGKAN)**

5b. Klik tombol *Edit*.

6b. Menampilkan halaman Prestasi Olahraga dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO NORMAL KRITERIA PRESTASI OLAHRAGA  
KLIK HAPUS**

1. Klik tombol Prestasi Olahraga.

2. Menampilkan halaman Prestasi Olahraga dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

3. Pilih data yang akan dihapus

4. Klik tombol Hapus

5. Menampilkan halaman Prestasi Olahraga dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

---

Tabel A.7 merupakan skenario dari *usecase* mengelola Kriteria Prestasi Olahraga. Skenario mengelola Kriteria Prestasi Olahraga menjelaskan alur tambah, *edit*, hapus, dan kosongkan. Skenario mengelola Kriteria Prestasi Olahraga dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari tambah, *edit* dan hapus. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap dan kosongkan. Kondisi setelah skenario ini dijalankan adalah *admin* berhasil mengelola Kriteria Prestasi Olahraga.

A.8 Skenario Mengelola Kriteria Prestasi Operasi Tempur

Tabel A.8 Skenario Mengelola Kriteria Prestasi Operasi Tempur

Nomor <i>Use Case</i>	UC-08
Nama	Mengelola Kriteria Prestasi Operasi Tempur
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i> sistem
<i>Post Condition</i>	Operator berhasil Mengelola Prestasi Operasi Tempur

**SKENARIO NORMAL MENGELOLA KRITERIA PRESTASI OPERASI TEMPUR**

**TAMBAH KRITERIA PRESTASI OPERASI TEMPUR**

Aktor	Sistem
1. Memilih menu Kriteria Prestasi Operasi Tempur.	2. Menampilkan halaman Prestasi Operasi Tempur. Dengan tombol Tambah, <i>Edit</i> , Hapus dan Kosongkan
3. Mengisi Form Prestasi Operasi Tempur.	
4. Klik tombol Tambah.	5. Menampilkan peringatan "TAMBAH DATA BERHASIL !!!".
6. Klik <i>OK</i> .	7. Menampilkan halaman Prestasi Operasi Tempur dengan tombol

**Aktor**

**Sistem**

Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO ALTERNATIF TAMBAH PRESTASI OPERASI TEMPUR  
(DATA BELUM LENGKAP)**

4a. Klik tombol Tambah.

5a. Menampilkan peringatan "Data Belum Lengkap"

**SKENARIO ALTERNATIF TAMBAH KRITERIA PRESTASI OPERASI  
TEMPUR  
(KLIK TOMBOL KOSONGKAN)**

3b. Klik tombol Kosongkan

4b. Menampilkan halaman Prestasi Operasi Tempur dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO NORMAL MENGELOLA KRITERIA PRESTASI OPERASI  
TEMPUR**

***EDIT* KRITERIA PRESTASI OPERASI TEMPUR**

1. Klik tombol Prestasi Operasi Tempur.

2. Menampilkan halaman Prestasi Operasi Tempur dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

3. Pilih data yang akan di *edit*.

**Aktor**

**Sistem**

- |   |   |
|---|---|
| <p>5. Mengisi form Prestasi Operasi Tempur.</p> <p>6. Klik tombol <i>Edit</i>.</p> <p>8. Klik tombol OK</p> | <p>4. Menampilkan halaman Prestasi Operasi Tempur dengan tombol Tambah, <i>Edit</i>, Hapus dan Kosongkan.</p> <p>7. Menampilkan peringatan “UPDATE DATA BERHASIL !!!”</p> <p>9. Menampilkan halaman Prestasi Operasi Tempur dengan tombol Tambah, <i>Edit</i>, Hapus dan Kosongkan.</p> |
|---|---|

**SKENARIO ALTERNATIF *EDIT* KRITERIA PRESTASI OPERASI TEMPUR (DATA BELUM LENGKAP)**

- |                                     |   |
|-------------------------------------|---|
| <p>6a. Klik tombol <i>Edit</i>.</p> | <p>7a. Menampilkan peringatan “Data Belum Lengkap!!!”</p> |
|-------------------------------------|---|

**SKENARIO ALTERNATIF KRITERIA PRESTASI OPERASI TEMPUR (KLIK TOMBOL KOSONGKAN)**

- |                                     |   |
|-------------------------------------|---|
| <p>5b. Klik tombol <i>Edit</i>.</p> | <p>6b. Menampilkan halaman Prestasi Operasi Tempur dengan tombol Tambah, <i>Edit</i>, Hapus dan</p> |
|-------------------------------------|---|



Aktor

Sistem

Kosongkan.

**SKENARIO NORMAL KRITERIA PRESTASI OPERASI TEMPUR****KLIK HAPUS**

1. Klik tombol Prestasi Operasi Tempur.
2. Menampilkan halaman Prestasi Operasi Tempur dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
3. Pilih data yang akan dihapus
4. Klik tombol Hapus
5. Menampilkan halaman Prestasi Operasi Tempur dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

---

Tabel A.8 merupakan skenario dari *usecase* mengelola Kriteria Prestasi Operasi Tempur. Skenario mengelola Kriteria Prestasi Operasi Tempur menjelaskan alur tambah, *edit*, hapus, dan kosongkan. Skenario mengelola Kriteria Prestasi Operasi Tempur dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari tambah, *edit* dan hapus. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap dan kosongkan. Kondisi setelah skenario ini dijalankan adalah *admin* berhasil mengelola Kriteria Prestasi Operasi Tempur.

## A.9 Skenario Mengelola Kriteria Umur

Tabel A.9 Skenario Mengelola Kriteria Umur

Nomor <i>Use Case</i>	UC-09
Nama	Mengelola Kriteria Umur
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i> sistem
<i>Post Condition</i>	Operator berhasil Mengelola Umur

**SKENARIO NORMAL MENGELOLA KRITERIA UMUR  
 TAMBAH KRITERIA PRESTASI UMUR**

**Aktor****Sistem**

- |   |   |
|---|---|
| <ol style="list-style-type: none"> <li>1. Memilih menu Kriteria Umur.</li> <li>3. Mengisi Form Umur.</li> <li>4. Klik tombol Tambah.</li> <li>6. Klik <i>OK</i>.</li> </ol> | <ol style="list-style-type: none"> <li>2. Menampilkan halaman Umur. Dengan tombol Tambah, <i>Edit</i>, Hapus dan Kosongkan</li> <li>5. Menampilkan peringatan “TAMBAH DATA BERHASIL !!!”.</li> <li>7. Menampilkan halaman Umur dengan tombol Tambah, <i>Edit</i>, Hapus dan Kosongkan.</li> </ol> |
|---|---|

**SKENARIO ALTERNATIF TAMBAH UMUR  
 (DATA BELUM LENGKAP)**

- |   |   |
|---|---|
| <ol style="list-style-type: none"> <li>4a. Klik tombol Tambah.</li> </ol> | <ol style="list-style-type: none"> <li>5a. Menampilkan peringatan “Data Belum Lengkap”</li> </ol> |
|---|---|

**SKENARIO ALTERNATIF TAMBAH KRITERIA UMUR  
(KLIK TOMBOL KOSONGKAN)**

3b. Klik tombol Kosongkan

4b. Menampilkan halaman Umur dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO NORMAL MENGELOLA KRITERIA UMUR  
EDIT KRITERIA UMUR**

1. Klik tombol Umur.

2. Menampilkan halaman Umur dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

3. Pilih data yang akan di *edit*.

4. Menampilkan halaman Umur dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

5. Mengisi form Umur.

6. Klik tombol *Edit*.

7. Menampilkan peringatan "UPDATE DATA BERHASIL !!!"

8. Klik tombol OK

9. Menampilkan halaman Umur dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO ALTERNATIF *EDIT* KRITERIA UMUR  
(DATA BELUM LENGKAP)**

6a. Klik tombol *Edit*.

- 7a. Menampilkan peringatan “Data Belum Lengkap!!!”

### SKENARIO ALTERNATIF KRITERIA UMUR

#### (KLIK TOMBOL KOSONGKAN)

- 5b. Klik tombol *Edit*.

- 6b. Menampilkan halaman Umur dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

### SKENARIO NORMAL KRITERIA UMUR

#### KLIK HAPUS

1. Klik tombol Umur.
2. Menampilkan halaman Umur dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
3. Pilih data yang akan dihapus
4. Klik tombol Hapus
5. Menampilkan halaman Umur dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

---

Tabel A.9 merupakan skenario dari *usecase* mengelola Kriteria Umur. Skenario mengelola Kriteria Umur menjelaskan alur tambah, *edit*, hapus, dan kosongkan. Skenario mengelola Kriteria Umur dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari tambah, *edit* dan hapus. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap dan kosongkan. Kondisi setelah skenario ini dijalankan adalah *admin* berhasil mengelola Kriteria Umur.

A.10 Skenario Mengelola Kriteria Jasmani Lari

Tabel A.10 Skenario Mengelola Kriteria Jasmani Lari

Nomor <i>Use Case</i>	UC-10
Nama	Mengelola Kriteria Jasmani Lari
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i> sistem
<i>Post Condition</i>	Operator berhasil Mengelola Jasmani Lari

**SKENARIO NORMAL MENGELOLA KRITERIA JASMANI LARI  
 TAMBAH KRITERIA PRESTASI JASMANI LARI**

Aktor	Sistem
8. Memilih menu Kriteria Jasmani Lari.	9. Menampilkan halaman Jasmani Lari. Dengan tombol Tambah, <i>Edit</i> , Hapus dan Kosongkan
10. Mengisi Form Jasmani Lari.	
11. Klik tombol Tambah.	12. Menampilkan peringatan "TAMBAH DATA BERHASIL !!!".
13. Klik <i>OK</i> .	14. Menampilkan halaman Jasmani Lari dengan tombol Tambah, <i>Edit</i> , Hapus dan Kosongkan.

**SKENARIO ALTERNATIF TAMBAH JASMANI LARI  
 (DATA BELUM LENGKAP)**

4a. Klik tombol Tambah.	5a. Menampilkan peringatan "Data
-------------------------	----------------------------------



Belum Lengkap”

**SKENARIO ALTERNATIF TAMBAH KRITERIA JASMANI LARI  
(KLIK TOMBOL KOSONGKAN)**

3b. Klik tombol Kosongkan

4b. Menampilkan halaman Jasmani Lari dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO NORMAL MENGELOLA KRITERIA JASMANI LARI  
*EDIT* KRITERIA JASMANI LARI**

1. Klik tombol Jasmani Lari.

2. Menampilkan halaman Jasmani Lari dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

3. Pilih data yang akan di *edit*.

4. Menampilkan halaman Jasmani Lari dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

5. Mengisi form Jasmani Lari.

6. Klik tombol *Edit*.

7. Menampilkan peringatan “UPDATE DATA BERHASIL !!!”

8. Klik tombol OK

9. Menampilkan halaman Jasmani Lari dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO ALTERNATIF *EDIT* KRITERIA JASMANI LARI  
(DATA BELUM LENGKAP)**

6a. Klik tombol *Edit*.

7a. Menampilkan peringatan “Data  
Belum Lengkap!!!”

**SKENARIO ALTERNATIF KRITERIA JASMANI LARI  
(KLIK TOMBOL KOSONGKAN)**

5b. Klik tombol *Edit*.

6b. Menampilkan halaman Jasmani Lari  
dengan tombol Tambah, *Edit*,  
Hapus dan Kosongkan.

**SKENARIO NORMAL KRITERIA JASMANI LARI  
KLIK HAPUS**

1. Klik tombol Jasmani Lari.

2. Menampilkan halaman Jasmani Lari  
dengan tombol Tambah, *Edit*,  
Hapus dan Kosongkan.

3. Pilih data yang akan dihapus

4. Klik tombol Hapus

5. Menampilkan halaman Jasmani  
Lari dengan tombol Tambah, *Edit*,  
Hapus dan Kosongkan.

---

Tabel A.10 merupakan skenario dari *use case* mengelola Kriteria Jasmani Lari. Skenario mengelola Kriteria Jasmani Lari menjelaskan alur tambah, *edit*, hapus, dan kosongkan. Skenario mengelola Kriteria Jasmani Lari dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari tambah, *edit* dan hapus. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap dan

kosongkan. Kondisi setelah skenario ini dijalankan adalah *admin* berhasil mengelola Kriteria Jasmani Lari.

#### A.11 Skenario Mengelola Kriteria Jasmani *Push Up*

Tabel A.11 Skenario Mengelola Kriteria Jasmani *Push Up*

Nomor <i>Use Case</i>	UC-12
Nama	Mengelola Kriteria Jasmani <i>Push Up</i>
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i> sistem
<i>Post Condition</i>	Operator berhasil Mengelola Jasmani <i>Push Up</i>

**SKENARIO NORMAL MENGELOLA KRITERIA JASMANI *PUSH UP*  
TAMBAH KRITERIA PRESTASI JASMANI *PUSH UP***

Aktor	Sistem
1. Memilih menu Kriteria Jasmani <i>Push Up</i> .	2. Menampilkan halaman Jasmani <i>Push Up</i> . Dengan tombol Tambah, <i>Edit</i> , Hapus dan Kosongkan
3. Mengisi Form Jasmani <i>Push Up</i> .	5. Menampilkan peringatan "TAMBAH DATA BERHASIL !!!".
4. Klik tombol Tambah.	7. Menampilkan halaman Jasmani <i>Push Up</i> dengan tombol Tambah, <i>Edit</i> , Hapus dan Kosongkan.
6. Klik <i>OK</i> .	

**SKENARIO ALTERNATIF TAMBAH JASMANI *PUSH UP*  
(DATA BELUM LENGKAP)**

4a. Klik tombol Tambah.

5a. Menampilkan peringatan "Data  
Belum Lengkap"

**SKENARIO ALTERNATIF TAMBAH KRITERIA JASMANI *PUSH UP*  
(KLIK TOMBOL KOSONGKAN)**

3b. Klik tombol Kosongkan

4b. Menampilkan halaman Jasmani  
*Push Up* dengan tombol Tambah,  
Edit, Hapus dan Kosongkan.

**SKENARIO NORMAL MENGELOLA KRITERIA JASMANI *PUSH UP*  
*EDIT KRITERIA JASMANI *PUSH UP****

1. Klik tombol Jasmani *Push Up*.

2. Menampilkan halaman Jasmani  
*Push Up* dengan tombol Tambah,  
*Edit*, Hapus dan Kosongkan.

3. Pilih data yang akan di *edit*.

4. Menampilkan halaman Jasmani  
*Push Up* dengan tombol Tambah,  
*Edit*, Hapus dan Kosongkan.

5. Mengisi form Jasmani *Push Up*.

6. Klik tombol *Edit*.

7. Menampilkan peringatan  
"UPDATE DATA BERHASIL  
!!!"

8. Klik tombol OK

9. Menampilkan halaman Jasmani

*Push Up* dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO ALTERNATIF *EDIT* KRITERIA JASMANI *PUSH UP*  
(DATA BELUM LENGKAP)**

6a. Klik tombol *Edit*.

7a. Menampilkan peringatan “Data Belum Lengkap!!!”

**SKENARIO ALTERNATIF KRITERIA JASMANI *PUSH UP*  
(KLIK TOMBOL KOSONGKAN)**

5b. Klik tombol *Edit*.

6b. Menampilkan halaman Jasmani *Push Up* dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO NORMAL KRITERIA UMU JASMANI *PUSH UP*  
KLIK HAPUS**

1. Klik tombol *Push Up*.

2. Menampilkan halaman Jasmani *Push Up* dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

3. Pilih data yang akan dihapus

4. Klik tombol Hapus

5. Menampilkan halaman Jasmani *Push Up* dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

---

Tabel A.11 merupakan skenario dari *usecase* mengelola Kriteria Jasmani *Push Up*. Skenario mengelola Kriteria Jasmani *Push Up* menjelaskan alur tambah, edit, hapus, dan kosongkan. Skenario mengelola Kriteria Jasmani *Push Up* dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari tambah, *edit* dan hapus. Sedangkan skenario alternatif



merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap dan kosongkan. Kondisi setelah skenario ini dijalankan adalah *admin* berhasil mengelola Kriteria Jasmani *Push Up*.

#### A.12 Skenario Mengelola Kriteria *Sitt Up*

Tabel A.12 Skenario Mengelola Kriteria *Sitt Up*

Nomor <i>UseCase</i>	UC-10
Nama	Mengelola Kriteria <i>Sitt Up</i>
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i> sistem
<i>Post Condition</i>	Operator berhasil Mengelola Jasmani <i>Sitt Up</i>

#### SKENARIO NORMAL MENGELOLA KRITERIA JASMANI *SITT UP* TAMBAH KRITERIA PRESTASI JASMANI *SITT UP*

Aktor	Sistem
1. Memilih menu Kriteria Jasmani <i>Sitt Up</i> .	2. Menampilkan halaman Jasmani <i>Sitt Up</i> Dengan tombol Tambah, <i>Edit</i> , Hapus dan Kosongkan
3. Mengisi Form <i>Sitt Up</i> .	5. Menampilkan peringatan "TAMBAH DATA BERHASIL !!!".
4. Klik tombol Tambah.	7. Menampilkan halaman Jasmani <i>Sitt Up</i> dengan tombol Tambah, <i>Edit</i> , Hapus dan Kosongkan.
6. Klik <i>OK</i> .	

**SKENARIO ALTERNATIF TAMBAH JASMANI *SITT UP*  
(DATA BELUM LENGKAP)**

4a. Klik tombol Tambah.

5a. Menampilkan peringatan "Data  
Belum Lengkap"

**SKENARIO ALTERNATIF TAMBAH KRITERIA JASMANI *SITT UP*  
(KLIK TOMBOL KOSONGKAN)**

3b. Klik tombol Kosongkan

4b. Menampilkan halaman Jasmani *Sitt Up* dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO NORMAL MENGELOLA KRITERIA JASMANI *SITT UP*  
*EDIT KRITERIA JASMANI *SITT UP****

1. Klik tombol Jasmani *Sitt Up*.

2. Menampilkan halaman Jasmani *Sitt Up* dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

3. Pilih data yang akan di *edit*.

4. Menampilkan halaman Jasmani *Sitt Up* dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

5. Mengisi form Jasmani *Sitt Up*.

6. Klik tombol *Edit*.

7. Menampilkan peringatan  
"UPDATE DATA BERHASIL"

!!!”

8. Klik tombol *OK*

9. Menampilkan halaman *Jasmani Sitt Up* dengan tombol *Tambah*, *Edit*, *Hapus* dan *Kosongkan*.

**SKENARIO ALTERNATIF *EDIT* KRITERIA *JASMANI SITT UP*  
(DATA BELUM LENGKAP)**

6a. Klik tombol *Edit*.

7a. Menampilkan peringatan “Data Belum Lengkap!!!”

**SKENARIO ALTERNATIF KRITERIA *JASMANI SITT UP*  
(KLIK TOMBOL KOSONGKAN)**

5b. Klik tombol *Edit*.

6b. Menampilkan halaman *Jasmani Sitt Up* dengan tombol *Tambah*, *Edit*, *Hapus* dan *Kosongkan*.

**SKENARIO NORMAL KRITERIA *JASMANI SITT UP*  
KLIK HAPUS**

1. Klik tombol *Jasmani Sitt Up*.

2. Menampilkan halaman *Jasmani Sitt Up* dengan tombol *Tambah*, *Edit*, *Hapus* dan *Kosongkan*.

3. Pilih data yang akan dihapus

4. Klik tombol *Hapus*

5. Menampilkan halaman *Jasmani Sitt Up* dengan tombol *Tambah*, *Edit*, *Hapus* dan *Kosongkan*.

---

Tabel A.12 merupakan skenario dari *usecase* mengelola *Kriteria Jasmani Sitt Up*. Skenario mengelola *Kriteria Jasmani Sitt Up* menjelaskan alur *tambah*, *edit*, *hapus*,

dan kosongkan. Skenario mengelola Kriteria Jasmani *Sitt Up* dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari tambah, *edit* dan hapus. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap dan kosongkan. Kondisi setelah skenario ini dijalankan adalah *admin* berhasil mengelola Kriteria Jasmani *Sitt Up*.

#### A.13 Skenario Mengelola Kriteria Jasmani *Satle Run*

Tabel A.13 Skenario Mengelola Kriteria Jasmani *Satle Run*

Nomor <i>UseCase</i>	UC-13
Nama	Mengelola Kriteria Jasmani <i>Satle Run</i>
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i> sistem
<i>Post Condition</i>	Operator berhasil Mengelola Jasmani <i>Satle Run</i>

#### SKENARIO NORMAL MENGELOLA KRITERIA JASMANI *SATLE RUN*

##### TAMBAH KRITERIA PRESTASI JASMANI *SATLE RUN*

Aktor	Sistem
1. Memilih menu Kriteria Jasmani <i>Satle Run</i> .	
	2. Menampilkan halaman Jasmani <i>Satle Run</i> . Dengan tombol Tambah, <i>Edit</i> , Hapus dan Kosongkan
3. Mengisi Form Jasmani <i>Satle Run</i> .	
4. Klik tombol Tambah.	
	5. Menampilkan peringatan "TAMBAH DATA BERHASIL !!!".
6. Klik <i>OK</i> .	

7. Menampilkan halaman Jasmani *Satle Run* dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO ALTERNATIF TAMBAH JASMANI *SATLE RUN***  
**(DATA BELUM LENGKAP)**

4a. Klik tombol Tambah.

- 5a. Menampilkan peringatan "Data Belum Lengkap"

**SKENARIO ALTERNATIF TAMBAH KRITERIA JASMANI *SATLE RUN***  
**(KLIK TOMBOL KOSONGKAN)**

3b. Klik tombol Kosongkan

- 4b. Menampilkan halaman Jasmani *Satle Run* dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO NORMAL MENGELOLA KRITERIA JASMANI *SATLE RUN***  
***EDIT KRITERIA JASMANI *SATLE RUN****

1. Klik tombol Jasmani *Satle Run*.
2. Menampilkan halaman Jasmani *Satle Run* dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
3. Pilih data yang akan di *edit*.
4. Menampilkan halaman Jasmani *Satle Run* dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
5. Mengisi form Jasmani *Satle Run*.



6. Klik tombol *Edit*.
7. Menampilkan peringatan “UPDATE DATA BERHASIL !!!”
8. Klik tombol OK
9. Menampilkan halaman Jasmani *Satle Run* dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO ALTERNATIF *EDIT* KRITERIA JASMANI *SATLE RUN*  
(DATA BELUM LENGKAP)**

- 6a. Klik tombol *Edit*.
- 7a. Menampilkan peringatan “Data Belum Lengkap!!!”

**SKENARIO ALTERNATIF KRITERIA JASMANI *SATLE RUN*  
(KLIK TOMBOL KOSONGKAN)**

- 5b. Klik tombol *Edit*.
- 6b. Menampilkan halaman Jasmani *Satle Run* dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO NORMAL KRITERIA UMU JASMANI *SATLE RUN*  
KLIK HAPUS**

1. Klik tombol *Satle Run*.
2. Menampilkan halaman Jasmani *Satle Run* dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
3. Pilih data yang akan dihapus
4. Klik tombol Hapus

5. Menampilkan halaman Jasmani *Satle Run* dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

Tabel A.13 merupakan skenario dari *usecase* mengelola Kriteria Jasmani *Satle Run*. Skenario mengelola Kriteria Jasmani *Satle Run* menjelaskan alur tambah, *edit*, hapus, dan kosongkan. Skenario mengelola Kriteria Jasmani *Satle Run* dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari tambah, *edit* dan hapus. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap dan kosongkan. Kondisi setelah skenario ini dijalankan adalah *admin* berhasil mengelola Kriteria Jasmani *Satle Run*.

#### A.14 Skenario Mengelola Kriteria Jasmani Renang

Tabel A.14 Skenario Mengelola Kriteria Jasmani Renang

Nomor <i>UseCase</i>	UC-14
Nama	Mengelola Kriteria Jasmani Renang
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i> sistem
<i>Post Condition</i>	Operator berhasil Mengelola Jasmani Renang

#### SKENARIO NORMAL MENGELOLA KRITERIA JASMANI RENANG

#### TAMBAH KRITERIA PRESTASI JASMANI RENANG

- | Aktor  | Sistem   |
|--|--|
| <ol style="list-style-type: none"> <li>1. Memilih menu Kriteria Jasmani Renang.</li> </ol> | <ol style="list-style-type: none"> <li>2. Menampilkan halaman Jasmani Renang. Dengan tombol Tambah, <i>Edit</i> , Hapus dan Kosongkan</li> </ol> |
| <ol style="list-style-type: none"> <li>3. Mengisi Form Jasmani Renang.</li> </ol>          |  |

4. Klik tombol Tambah.
5. Menampilkan peringatan  
“TAMBAH DATA BERHASIL  
!!!”.
6. Klik *OK*.
7. Menampilkan halaman Jasmani Renang dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO ALTERNATIF TAMBAH JASMANI RENANG  
(DATA BELUM LENGKAP)**

- 4a. Klik tombol Tambah.
- 5a. Menampilkan peringatan ”Data Belum Lengkap”

**SKENARIO ALTERNATIF TAMBAH KRITERIA JASMANI RENANG  
(KLIK TOMBOL KOSONGKAN)**

- 3b. Klik tombol Kosongkan
- 4b. Menampilkan halaman Jasmani Renang dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

**SKENARIO NORMAL MENGELOLA KRITERIA JASMANI RENANG  
*EDIT* KRITERIA JASMANI RENANG**

1. Klik tombol Jasmani Renang.
2. Menampilkan halaman Jasmani Renang dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.
3. Pilih data yang akan di *edit*.
4. Menampilkan halaman Jasmani Renang dengan tombol Tambah,

*Edit, Hapus dan Kosongkan.*

5. Mengisi form Jasmani Renang.
6. Klik tombol *Edit*.
7. Menampilkan peringatan  
“UPDATE DATA BERHASIL  
!!!”
8. Klik tombol OK
9. Menampilkan halaman Jasmani  
Renang dengan tombol Tambah,  
*Edit, Hapus dan Kosongkan.*

#### **SKENARIO ALTERNATIF *EDIT* KRITERIA JASMANI RENANG (DATA BELUM LENGKAP)**

- 6a. Klik tombol *Edit*.
- 7a. Menampilkan peringatan “Data  
Belum Lengkap!!!”

#### **SKENARIO ALTERNATIF KRITERIA JASMANI RENANG (KLIK TOMBOL KOSONGKAN)**

- 5b. Klik tombol *Edit*.
- 6b. Menampilkan halaman Jasmani  
Renang dengan tombol Tambah,  
*Edit, Hapus dan Kosongkan.*

#### **SKENARIO NORMAL KRITERIA JASMANI RENANG KLIK HAPUS**

1. Klik tombol Renang.
2. Menampilkan halaman Jasmani  
Renang dengan tombol Tambah,  
*Edit, Hapus dan Kosongkan.*
3. Pilih data yang akan dihapus

4. Klik tombol Hapus

5. Menampilkan halaman Jasmani Renang dengan tombol Tambah, *Edit*, Hapus dan Kosongkan.

Tabel A.13 merupakan skenario dari *usecase* mengelola Kriteria Jasmani Renang. Skenario mengelola Kriteria Jasmani Renang menjelaskan alur tambah, *edit*, hapus, dan kosongkan. Skenario mengelola Kriteria Jasmani Renang dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari tambah, *edit* dan hapus. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap dan kosongkan. Kondisi setelah skenario ini dijalankan adalah *admin* berhasil mengelola Kriteria Jasmani Renang.

#### A.15 Skenario *Edit* Bobot

Tabel A.15 Skenario *Edit* Bobot

Nomor <i>UseCase</i>	UC-015
Nama	<i>Mengedit</i> Bobot
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i> sistem
<i>Post Condition</i>	Operator berhasil <i>Mengedit</i> Bobot

### SKENARIO NORMAL MENGEDIT BOBOT

#### *EDIT* BOBOT

- Memilih menu *Edit* Bobot.
- Menampilkan halaman Nilai Bobot dengan tombol *Edit*, Kosongkan dan Cek Total Nilai.
- Pilih data yang akan di *edit*.
- Menampilkan halaman Nilai Bobot dengan tombol *Edit*,



Kosongkan dan Cek Total Nilai.

5. Mengisi form Nilai Bobot.

6. Klik tombol *Edit*.

7. Menampilkan peringatan “*EDIT DATA BERHASIL !!!*”

8. Klik tombol OK

9. Menampilkan halaman Nilai Bobot dengan tombol *Edit*, Kosongkan dan Cek Total Nilai.

**SKENARIO ALTERNATIF *EDIT BOBOT*  
(DATA BELUM LENGKAP)**

6a. Klik tombol *Edit*.

7a. Menampilkan peringatan “Data Belum Lengkap”

**SKENARIO ALTERNATIF *EDIT BOBOT*  
(KLIK TOMBOL KOSONGKAN)**

5b. Klik tombol *Edit*.

6b. Menampilkan halaman Nilai Bobot dengan tombol *Edit*, Kosongkan dan Cek Total Nilai.

**SKENARIO ALTERNATIF NORMAL *EDIT BOBOT*  
(KLIK CEK TOTAL NILAI)**

8c. Klik OK

9c. Klik Cek Total Bobot

10.c Menampilkan peringatan “Maaf Nilai Kurang Dari 100”

11c. Klik OK

12c. Menampilkan halaman Nilai

---

Bobot dengan tombol *Edit*,  
Kosongkan dan Cek Total Nilai.

Tabel A.15 merupakan skenario dari *usecase* Mengedit Bobot. Skenario Mengedit Bobot Renang menjelaskan alur *edit*, kosongkan, dan cek total nilai. Skenario mengedit bobot dibagi menjadi dua bagian, yaitu skenario normal dan skenario alternatif. Skenario normal merupakan alur utama dari *edit*. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses data tidak lengkap kosongkan, Cek Total Nilai dan kurang dari 100. Kondisi setelah skenario ini dijalankan adalah *admin* berhasil mengedit total bobot.

#### A.16 Skenario Melihat Penerima DIKTUBA REG

Tabel A.16 Skenario Melihat Penerima DIKTUBA REG

Nomor <i>UseCase</i>	UC-017
Nama	Melihat Calon Penerima DIKTUBA REG
Aktor	Operator
<i>Pre Condition</i>	Operator harus sudah <i>login</i> sistem
<i>Post Condition</i>	Operator berhasil melihat Calon Penerima DIKTUBA REG

#### SKENARIO NORMAL MELIHAT CALON PENERIMA DIKTUBA REG

Aktor	Sistem
1. Memilih menu Penerima DIKTUBA REG.	2. Menampilkan halaman Daftar Penerima DIKTUBA REG Per-Periode yang berisi form cari.
3. Mengisi form Cari Tahun Angkatan	
4. Klik tombol Cari	5. Menampilkan data yang diinginkan.

**SKENARIO ALTERNATIF MELIHAT CALON PENERIMA DIKTUBA  
REG  
(FORM JUMLAH TERBAIK)**

- 3a. Mengisi form Jumlah yang diinginkan.
- 4a. Klik tombol Cari
- 5a. Menampilkan data yang diinginkan.

**SKENARIO ALTERNATIF MELIHAT CALON PENERIMA DIKTUBA  
REG  
(FORM CARI PERSONIL)**

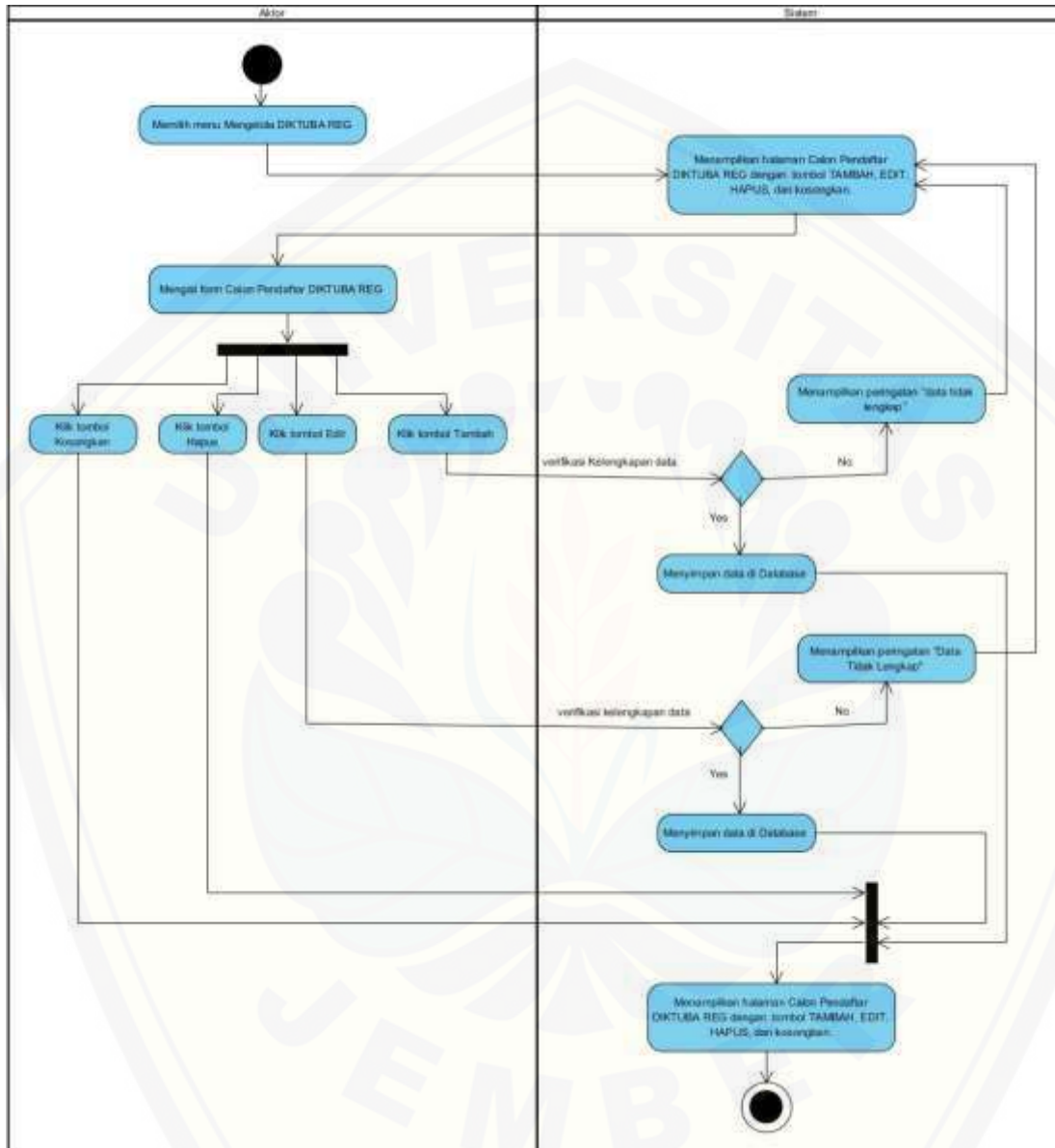
- 3b. Mengisi form Pangkat/NRP dan Tahun Angkatan.
- 4b. Klik tombol Cari Personil
- 4b. Menampilkan data yang diinginkan.

---

Tabel A.16 merupakan skenario dari *usecase* melihat Calon Penerima DIKTUBA REG. Skenario melihat Calon Penerima DIKTUBA REG menjelaskan alur melihat Calon Penerima DIKTUBA REG. Kondisi setelah skenario ini dijalankan adalah Operator berhasil melihat Calon Penerima DIKTUBA REG.

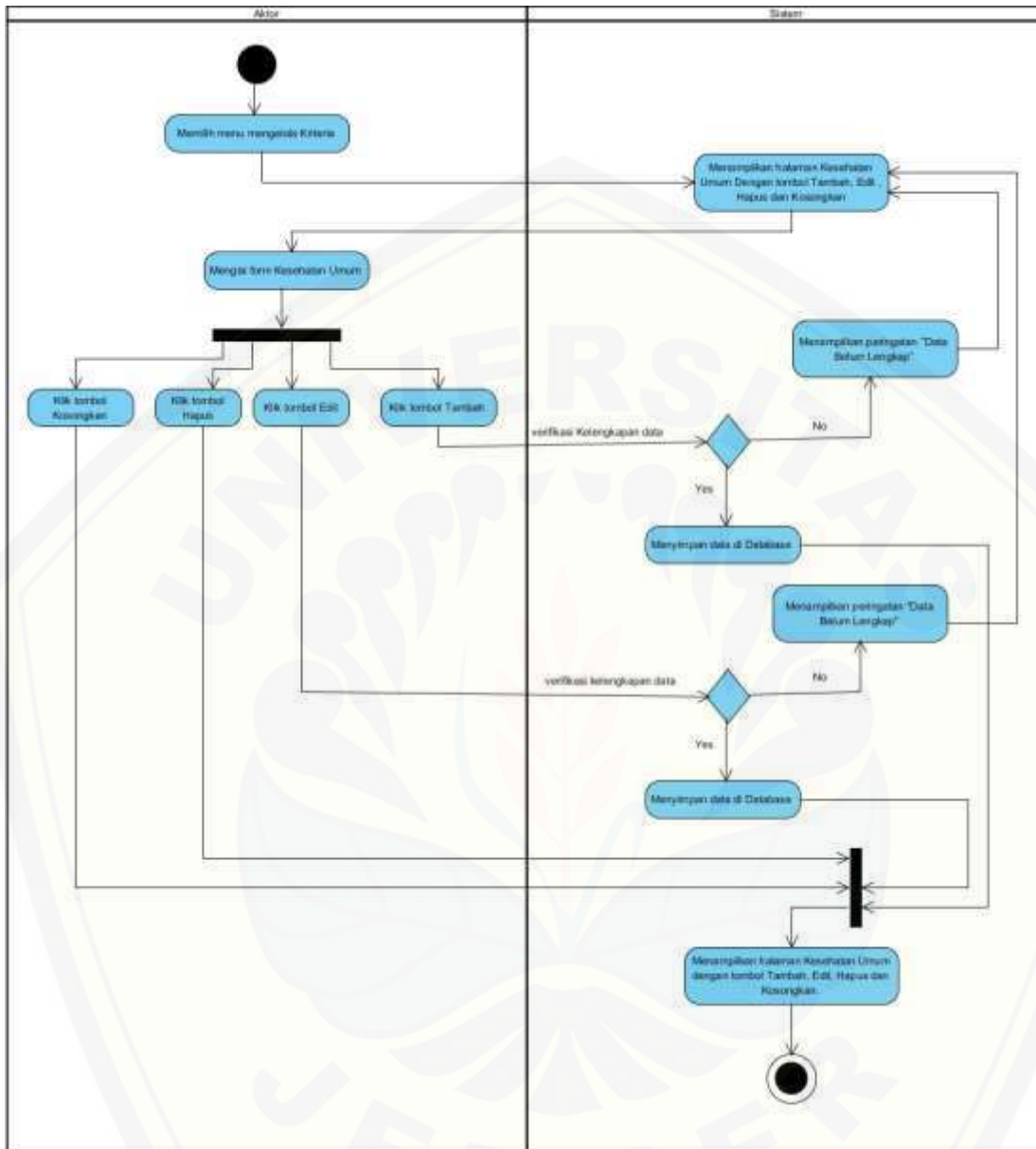
Lampiran B. Activity Diagram

B.1 Activity Diagram Mengelola data diktuba reg



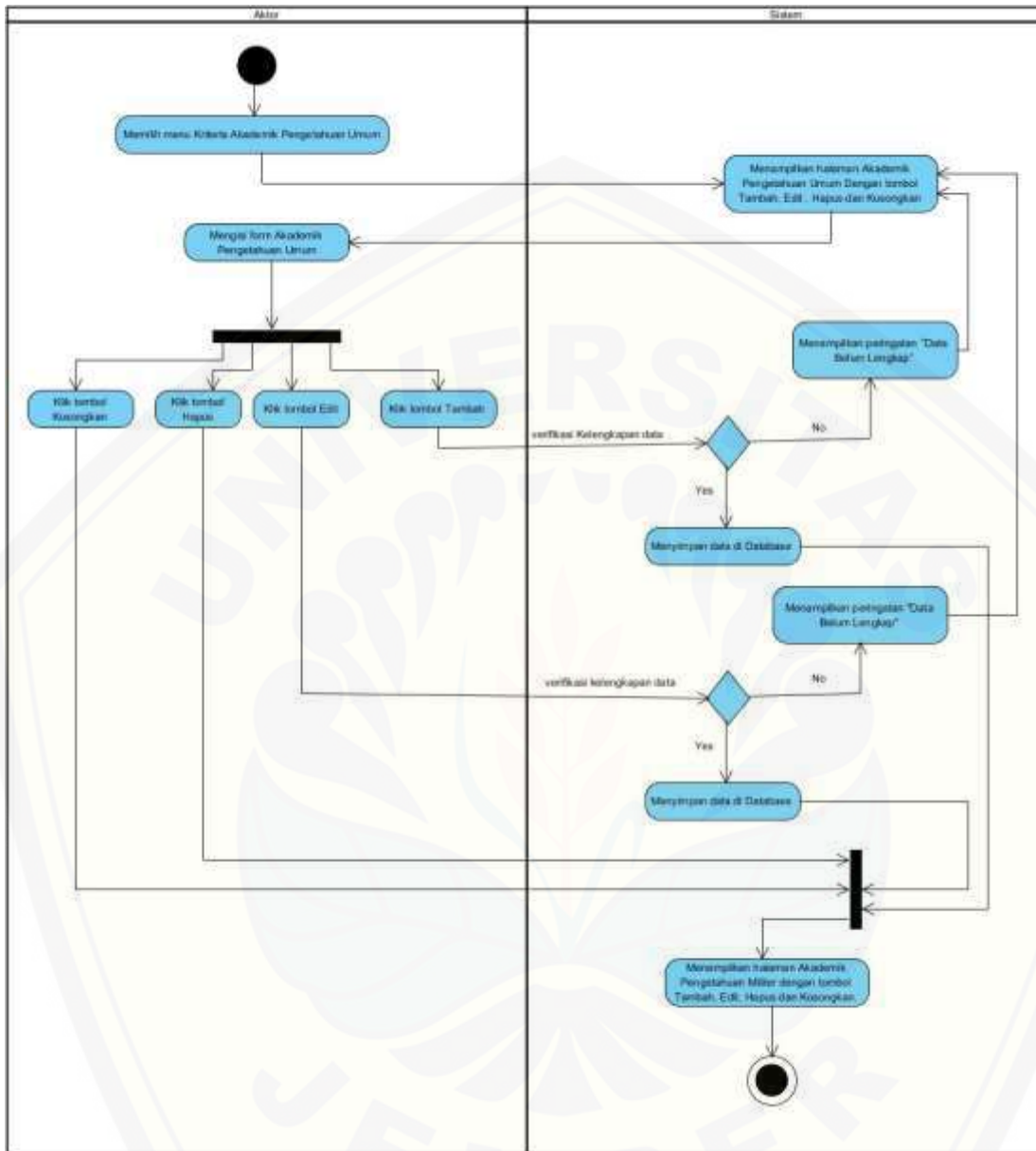
Gambar B.1 Activity Diagram Mengelola Mengelola data diktuba reg

B.2 Activity Diagram Mengelola Kriteria Kesehatan Umum



Gambar B.2 Activity Diagram Mengelola Kriteria Kesehatan Umum

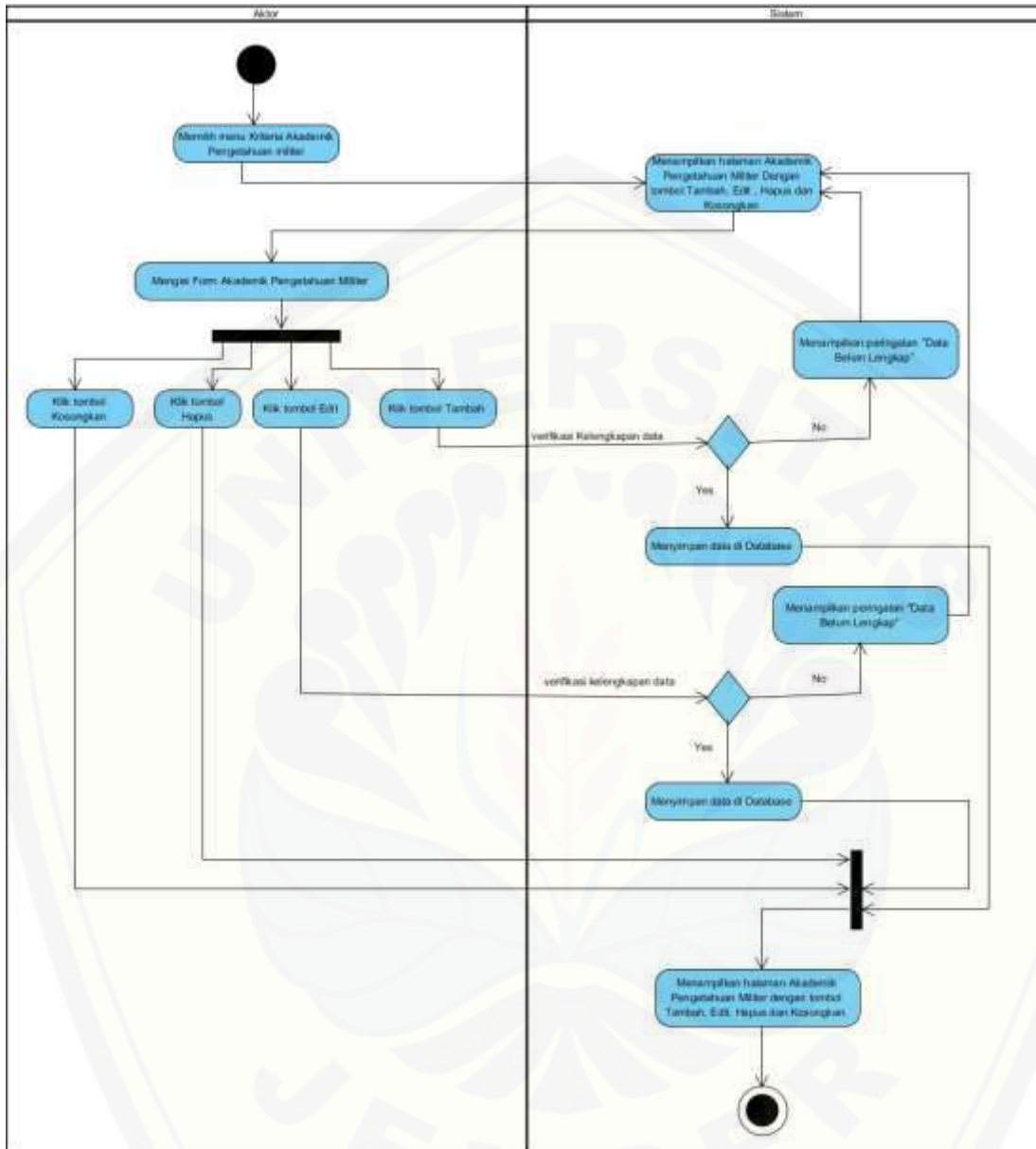
B.3 Activity Diagram Mengelola Kriteria Akademik Pengetahuan Umum



Gambar B.3 Activity Diagram Mengelola Kriteria Akademik Pengetahuan Umum

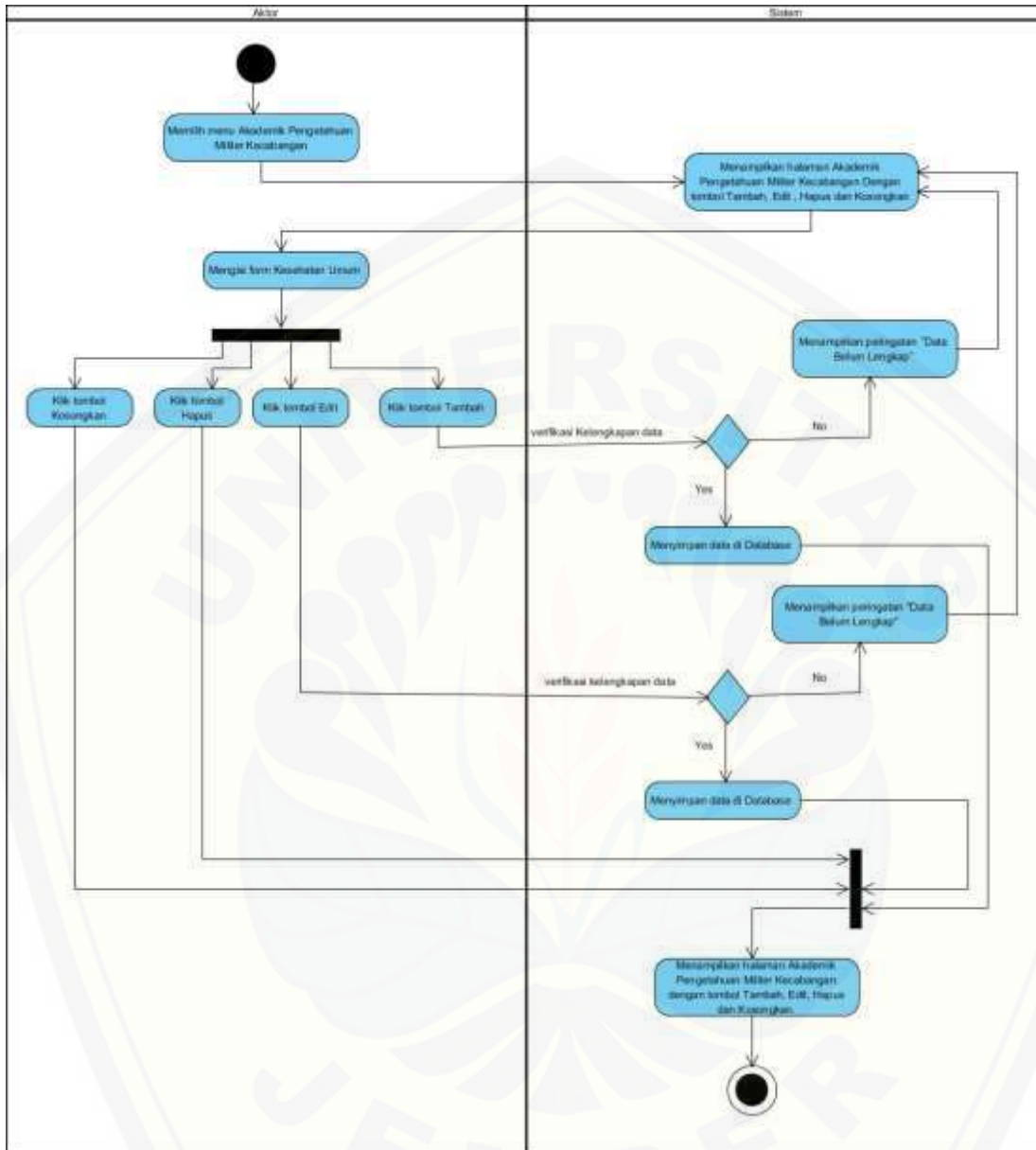


B.4 Activity Diagram 3 Mengelola Kriteria Akademik Pengetahuan Militer



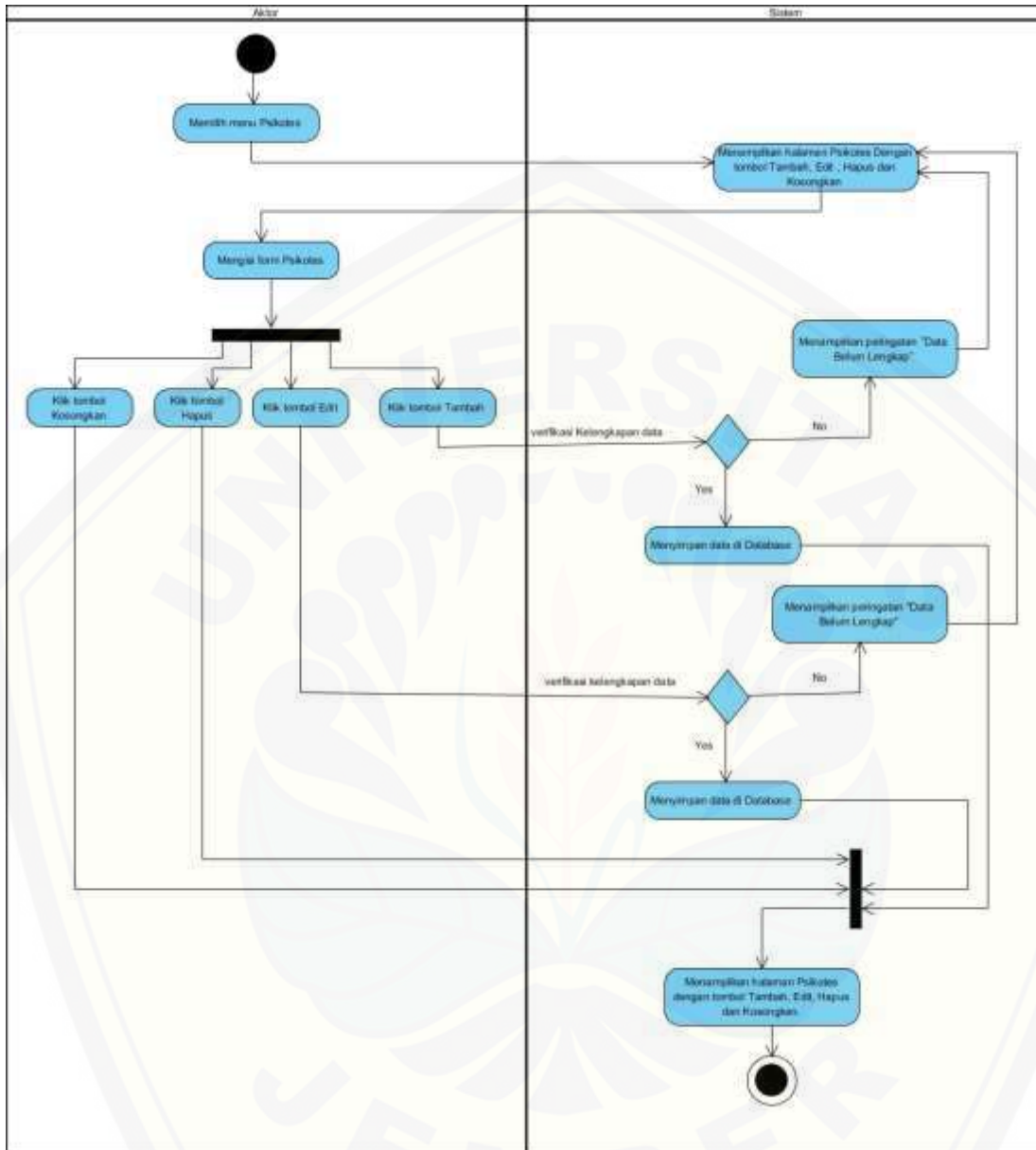
Gambar B.4 Activity Diagram Mengelola Kriteria Akademik Pengetahuan Militer

B.5 Activity Diagram Mengelola Kriteria Akademik Pengetahuan Militer Kecabangan



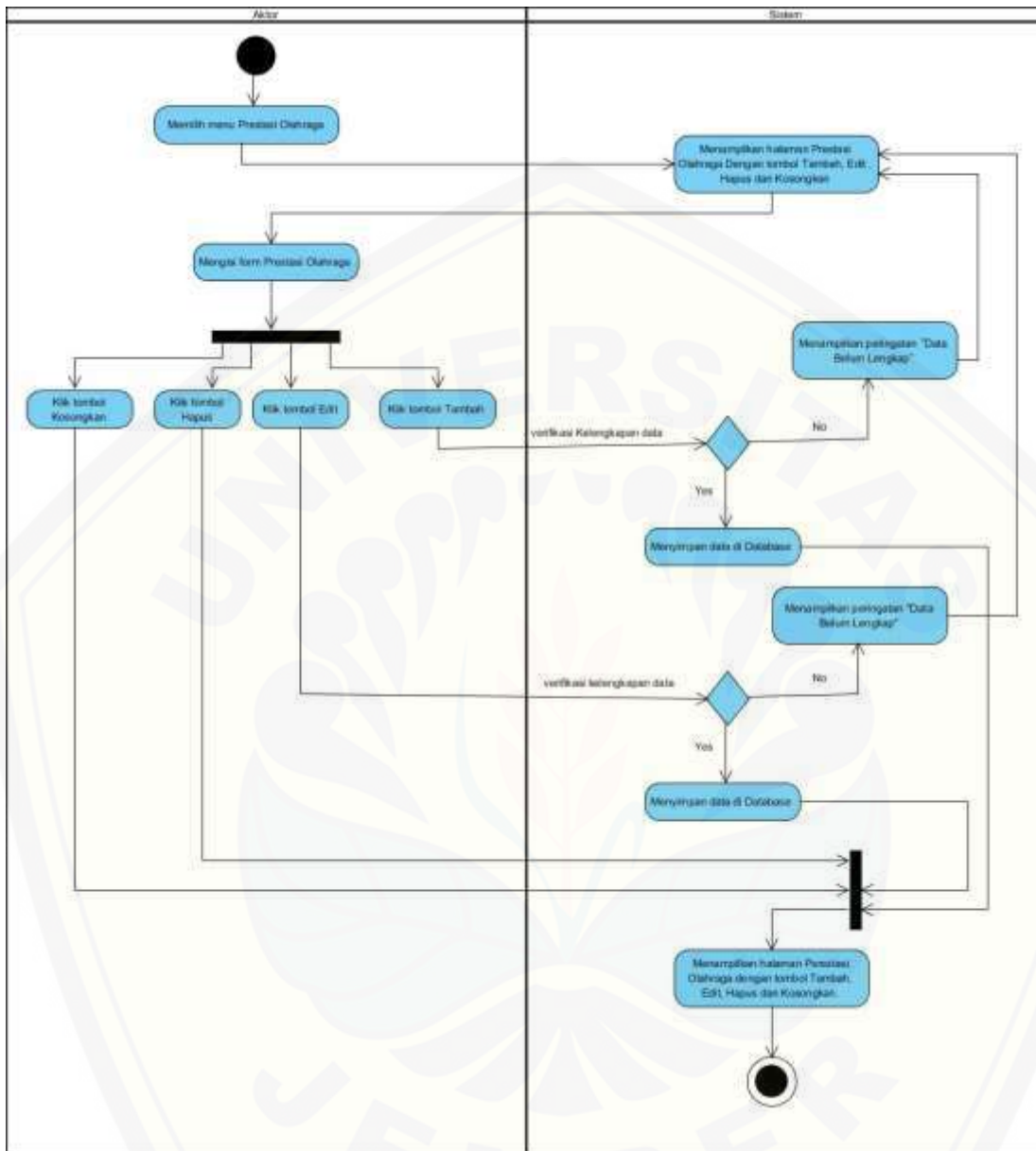
Gambar B.5 Activity Diagram Mengelola Kriteria Akademik Pengetahuan Militer Kecabangan

B.6 Activity Diagram Mengelola kriteria Psikotes



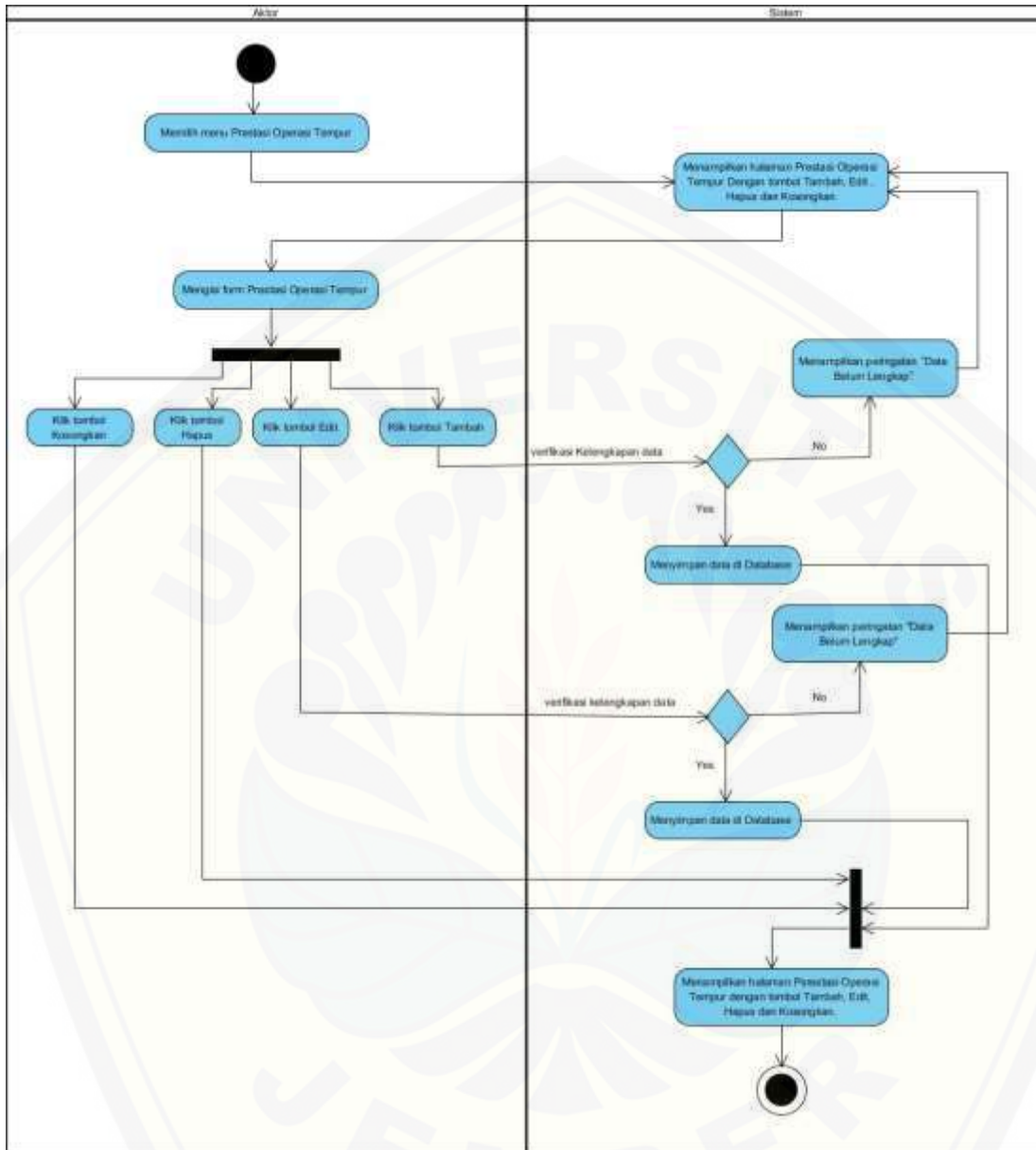
Gambar B.6 Activity Diagram Mengelola kriteria Psikotes

B.7 Activity Diagram Mengelola Kriteria Prestasi Olahraga



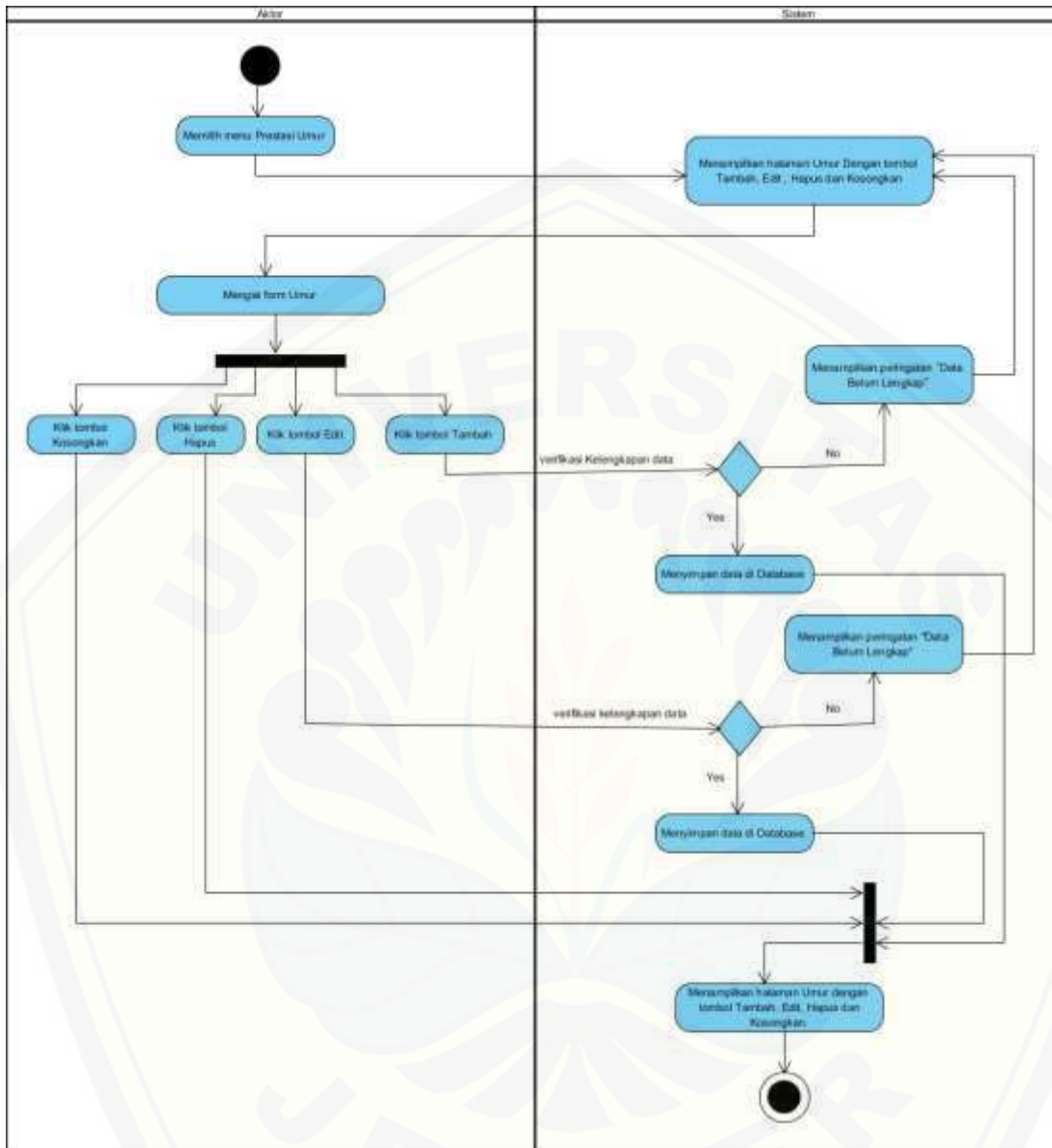
Gambar B.7 Activity Diagram Mengelola Kriteria Prestasi Olahraga

B.8 Activity Diagram 7Mengelola Kriteria Prestasi Olahraga Tempur



Gambar B.8 Activity Diagram Mengelola Kriteria Prestasi Olahraga Tempur

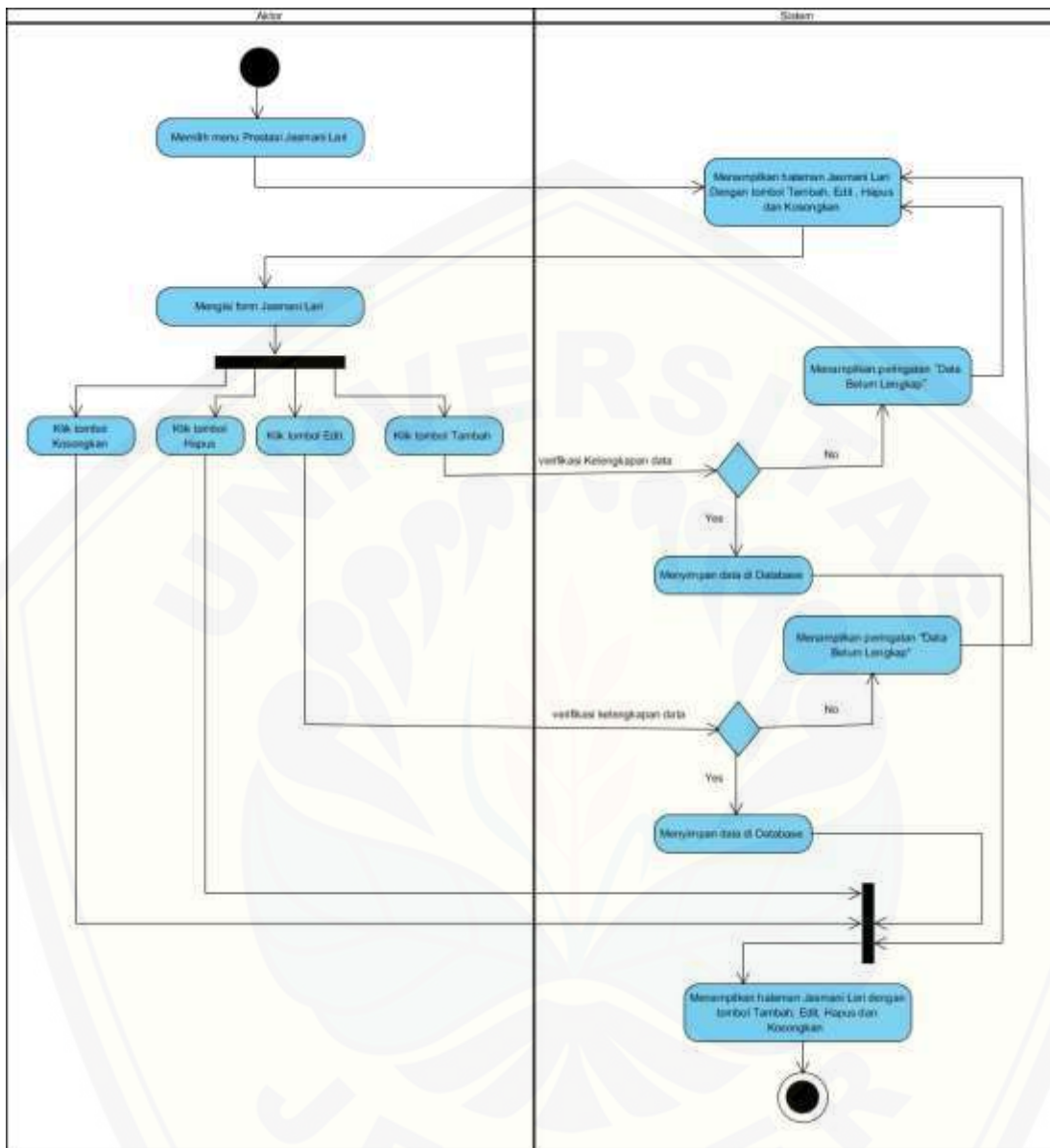
B.9 Activity Diagram Mengelola kriteria umur



Gambar B.9 Activity Diagram Mengelola kriteria umur

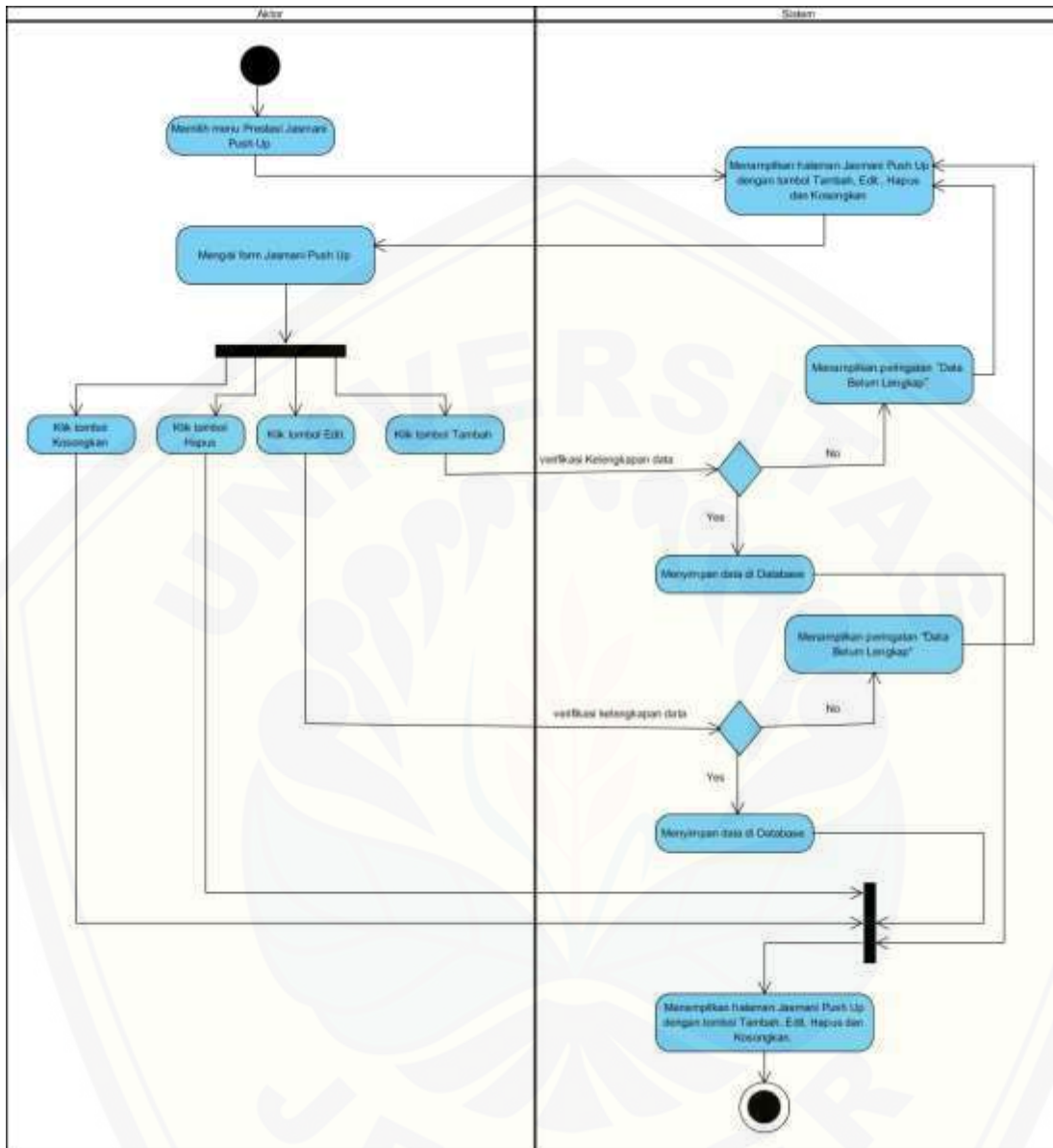


B.10 Activity Diagram Mengelola kriteria Jasmani lari



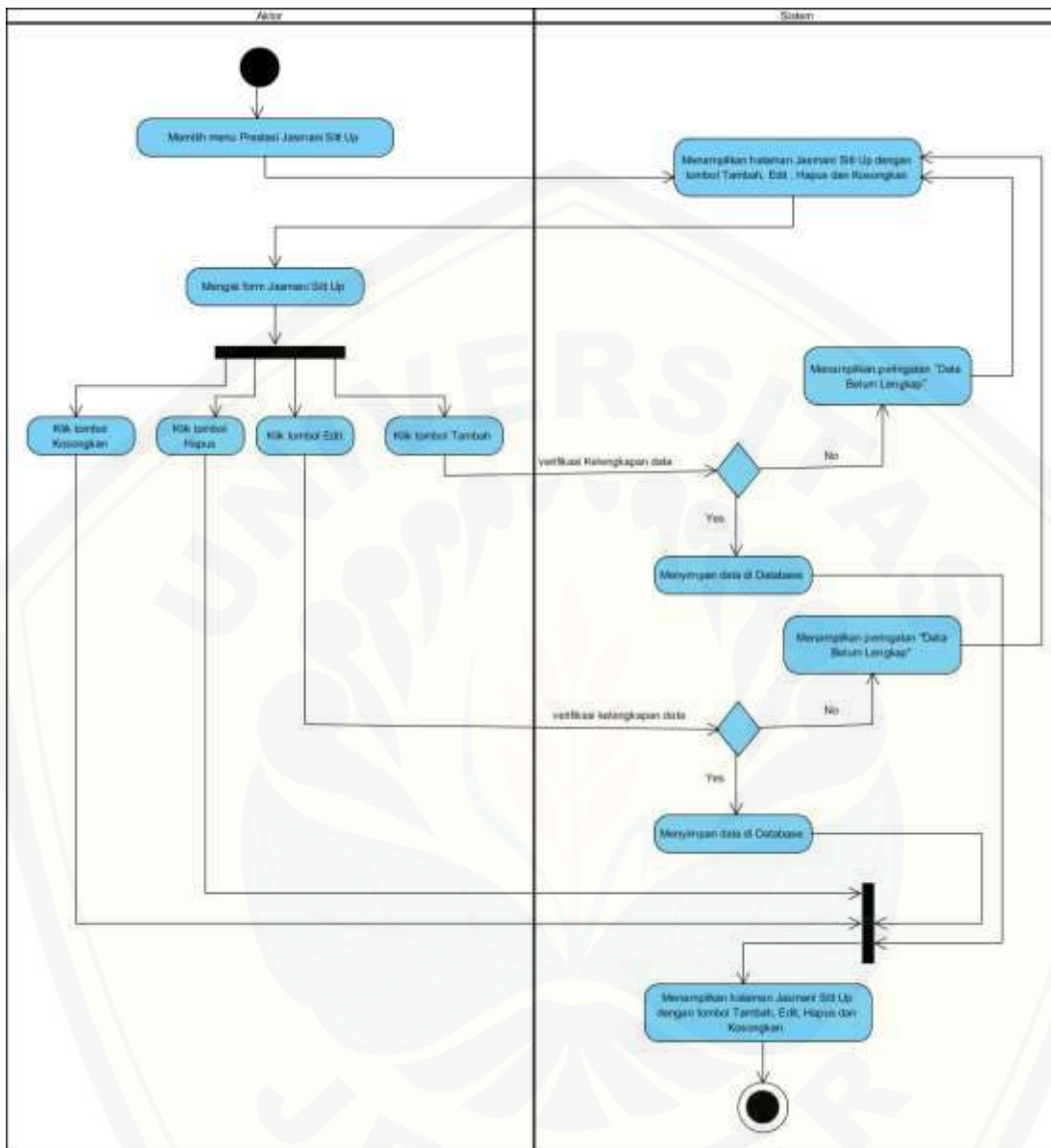
Gambar B.10 Activity Diagram Mengelola kriteria Jasmani lari

B.11 Activity Diagram Mengelola Kriteria Jasmani Push Up



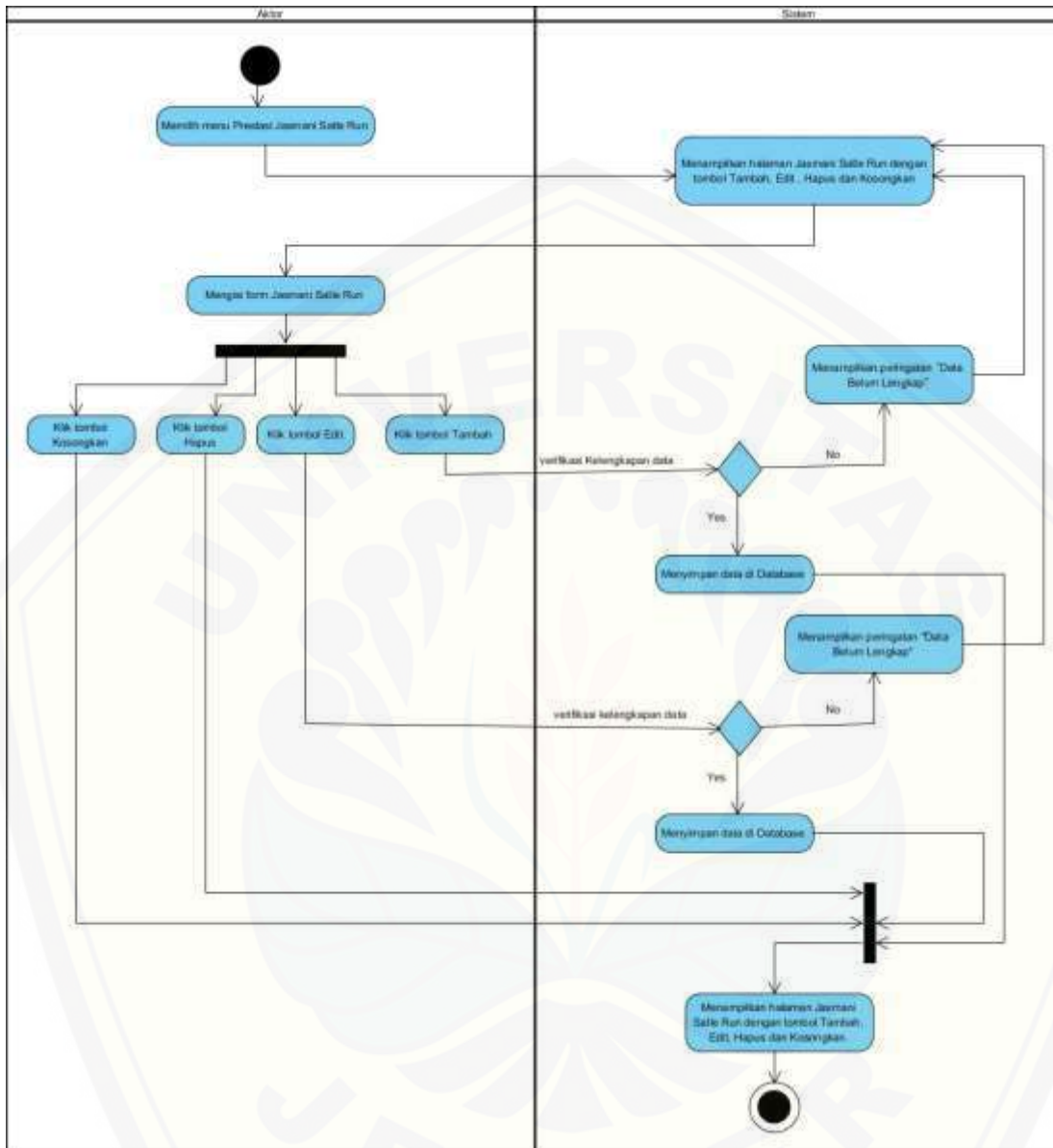
Gambar B.11 Activity Diagram Mengelola Kriteria Jasmani Push Up

B.12 Activity Diagram Mengelola Kriteria Jasmani Sitt Up



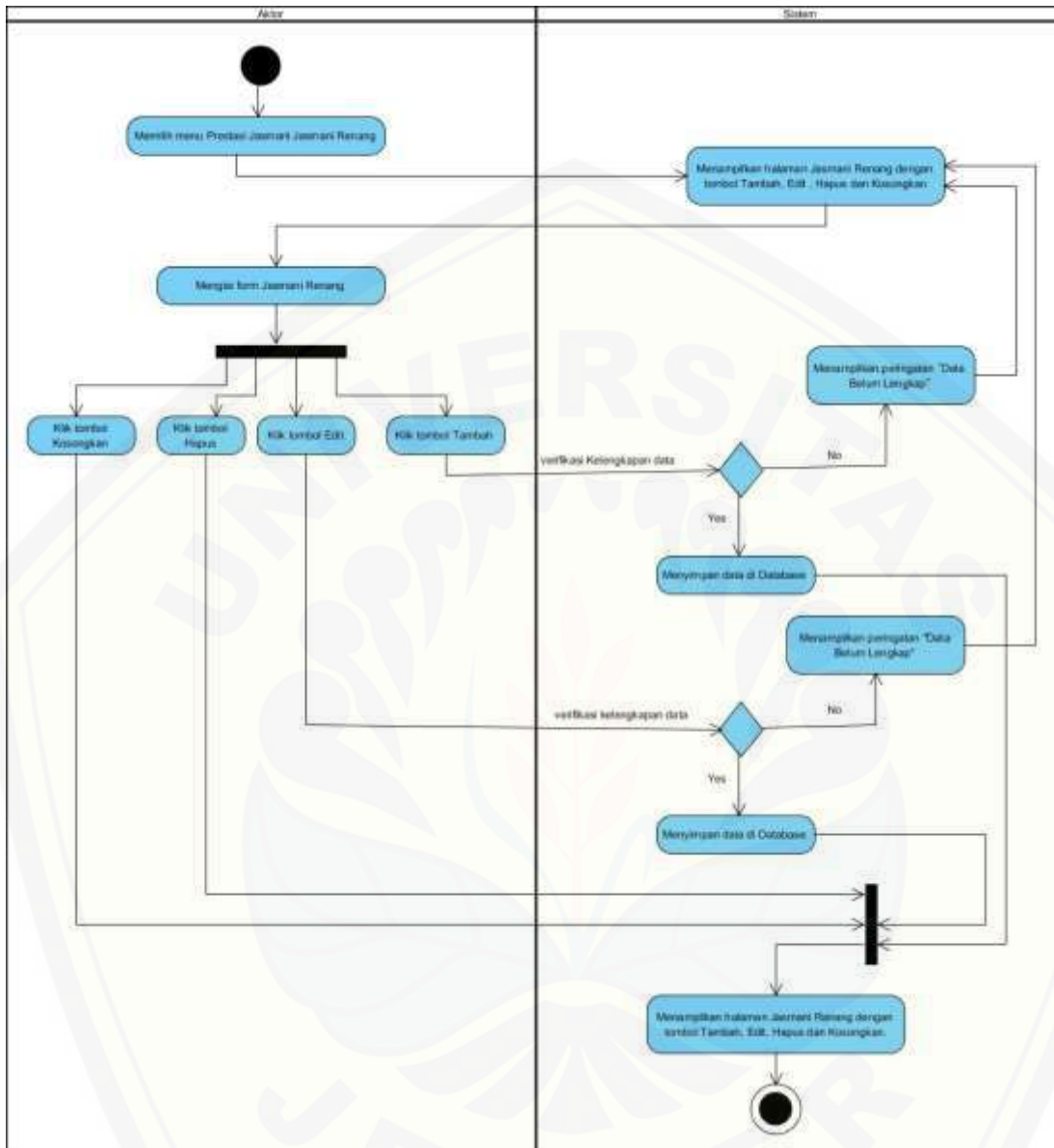
Gambar B.12 Activity Diagram Mengelola Kriteria Jasmani Sitt Up

B.13 Activity Diagram Mengelola kriteria Jasmani Satele Run



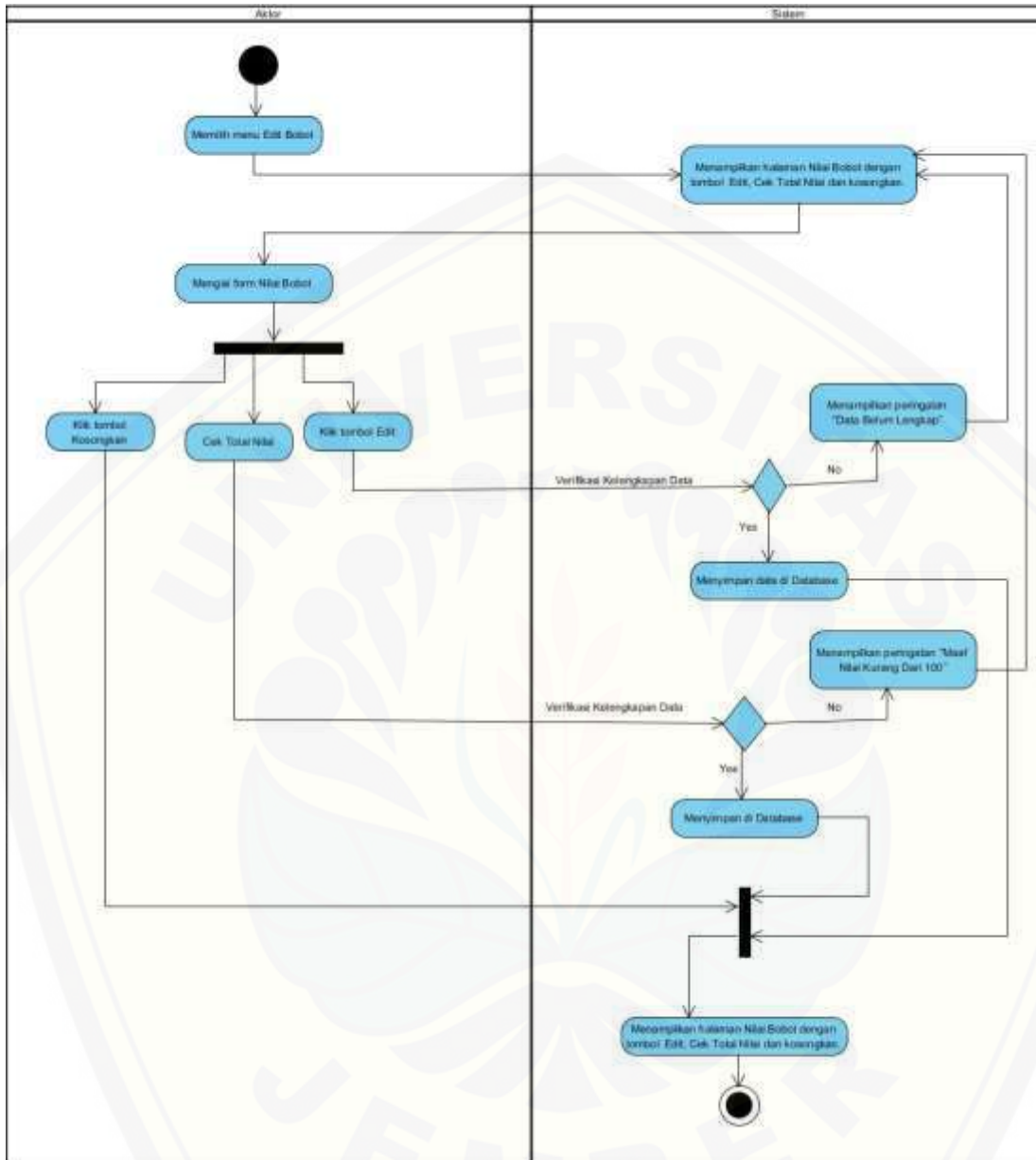
Gambar B.13 Activity Diagram Mengelola kriteria Jasmani Satele Run

B.14 Activity Diagram Mengelola kriteria Jasmani Renang



Gambar B.14 Activity Diagram Mengelola kriteria Jasmani Renang

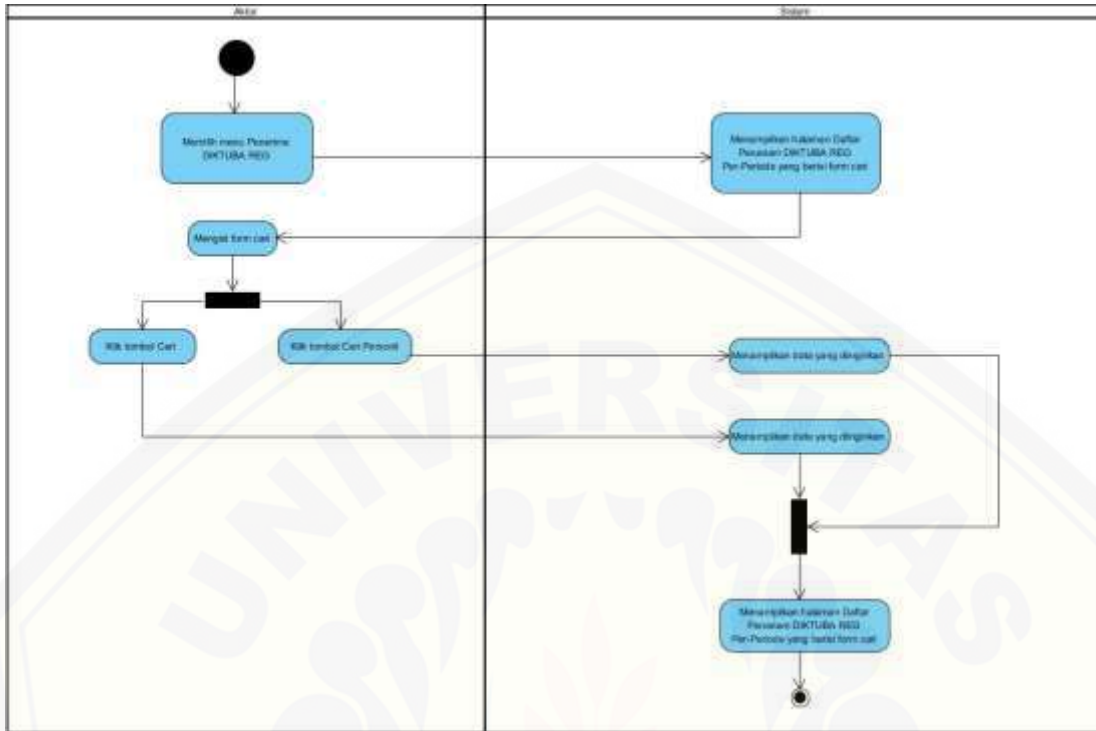
B.15 Activity Diagram Edit Bobot



Gambar B.15 Activity Diagram Edit Bobot



B.16 Activity Diagram Melihat penerima DIKTUBA REG

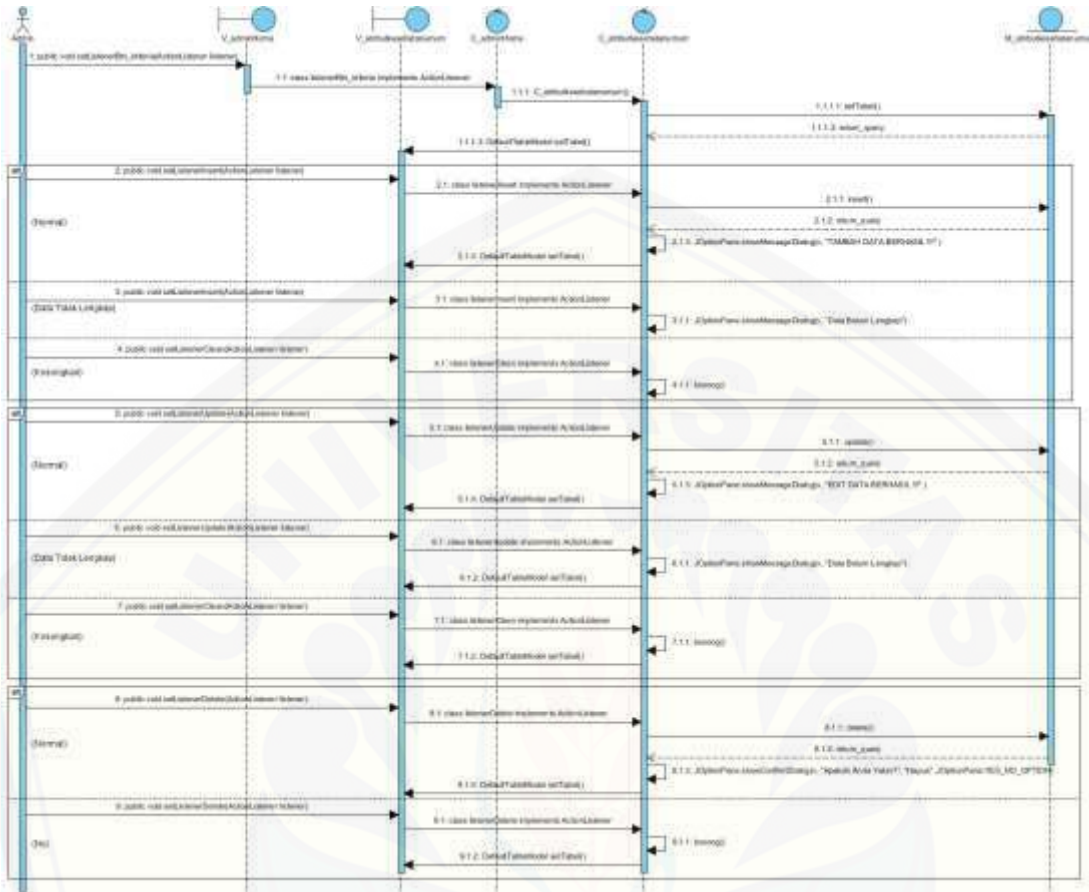


Gambar B.16 Activity Melihat penerima DIKTUBA REG



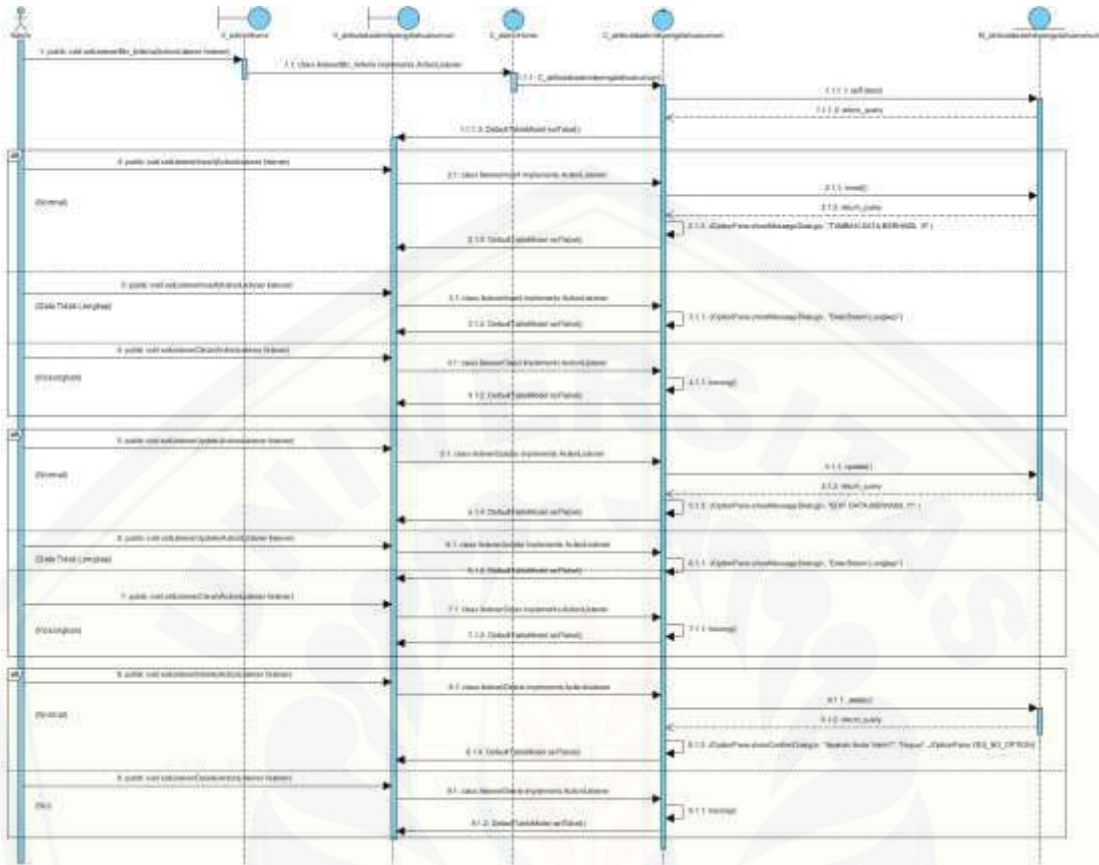


C.3 *Sequence Diagram Admin Mengelola Kriteria kesehatan Umum*



Gambar C.3 *Sequence Diagram Admin Mengelola Kriteria kesehatan Umum*

C.4 Squence Diagram Admin Mengelola Kriteria Akademik Pengetahuan Umum



Gambar C.4 Squence Diagram Admin Mengelola Kriteria Akademik Pengetahuan Umum







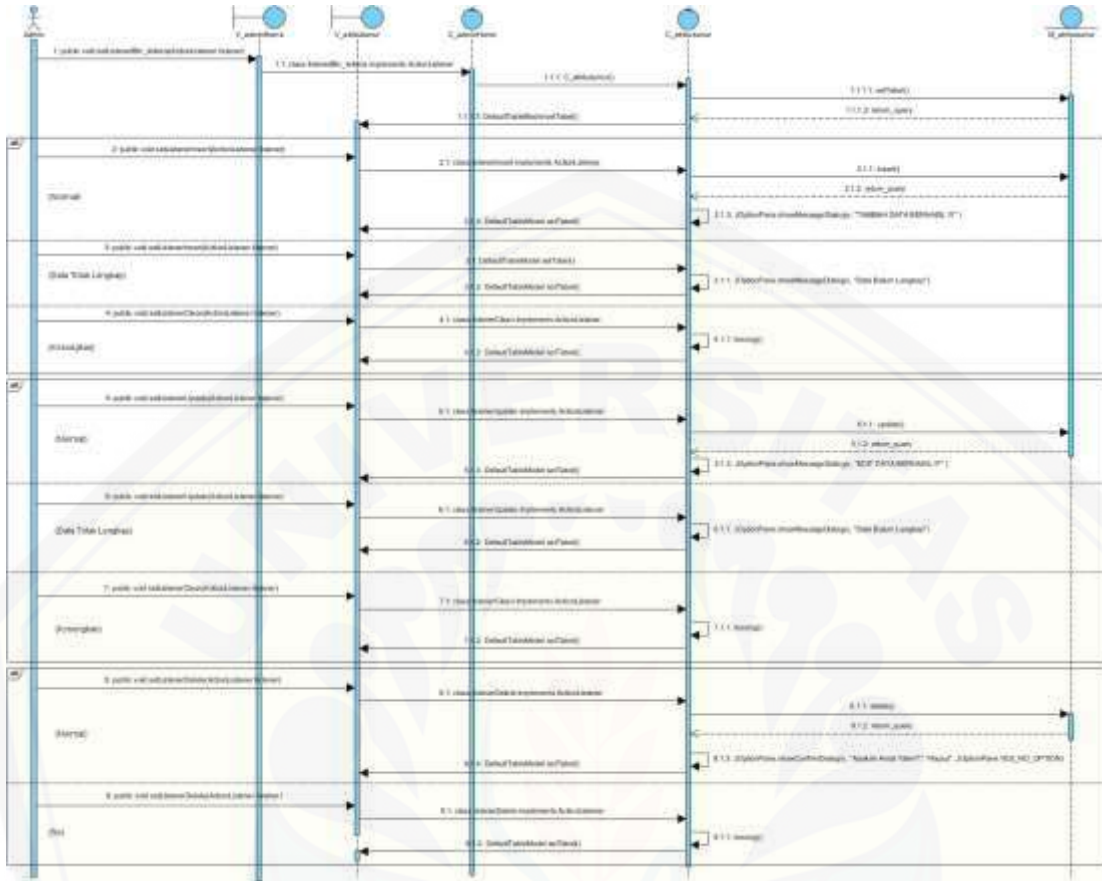








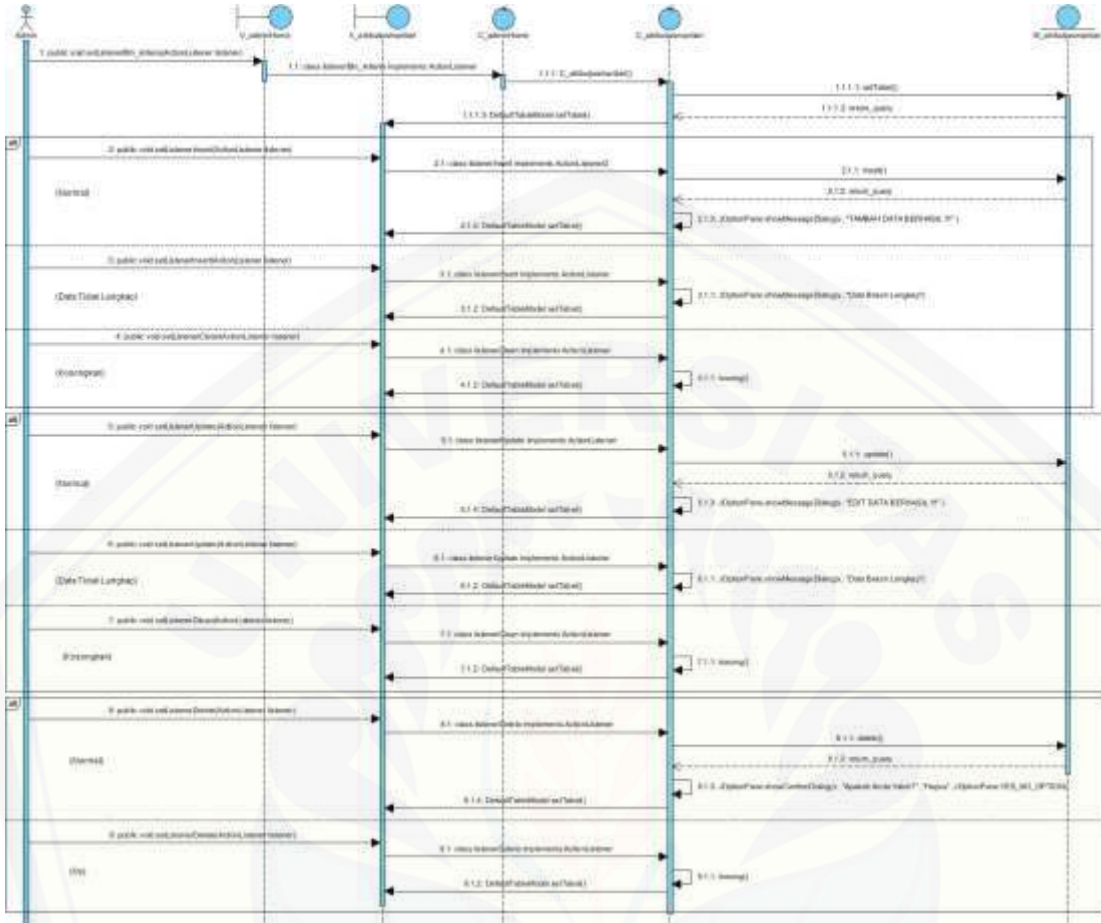
C.10 Sequence Diagram Admin Mengelola Kriteria Umur



Gambar C.10 Sequence Diagram Admin Mengelola Kriteria Umur



C.11 *Sequence Diagram Admin Mengelola Kriteria Jasmani Lari*

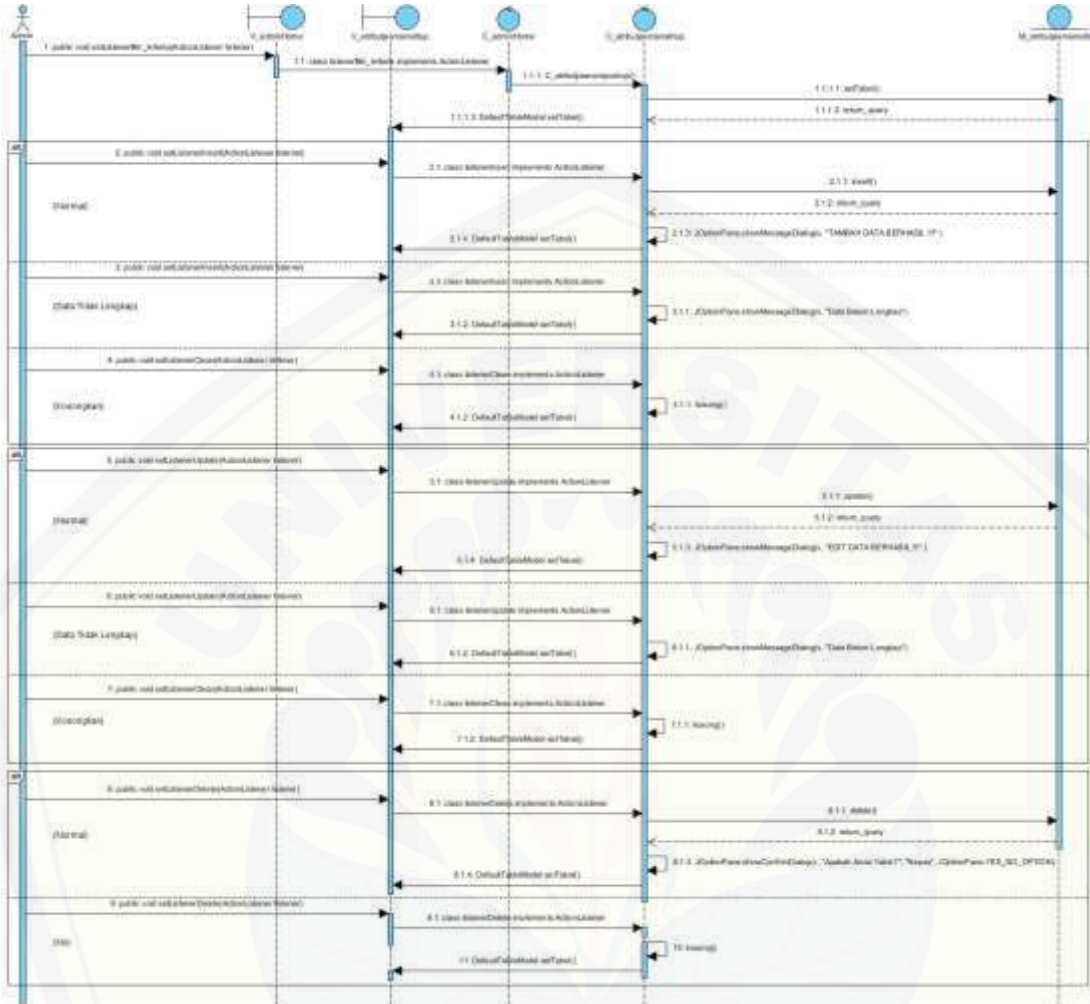


Gambar C.11 *Sequence Diagram Admin Mengelola Kriteria Jasmani Lari*





C.13 *Sequence Diagram Admin Mengelola Kriteria Jasmani Sitt Up*

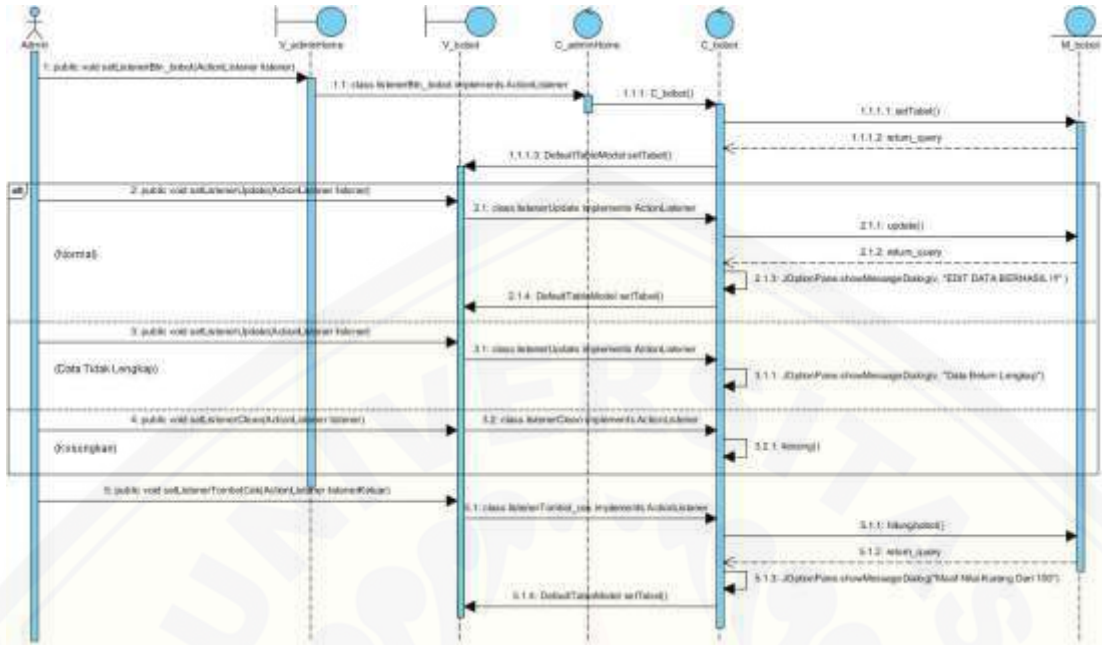


Gambar C.13 *Sequence Diagram Admin Mengelola Kriteria Jasmani Sitt Up*



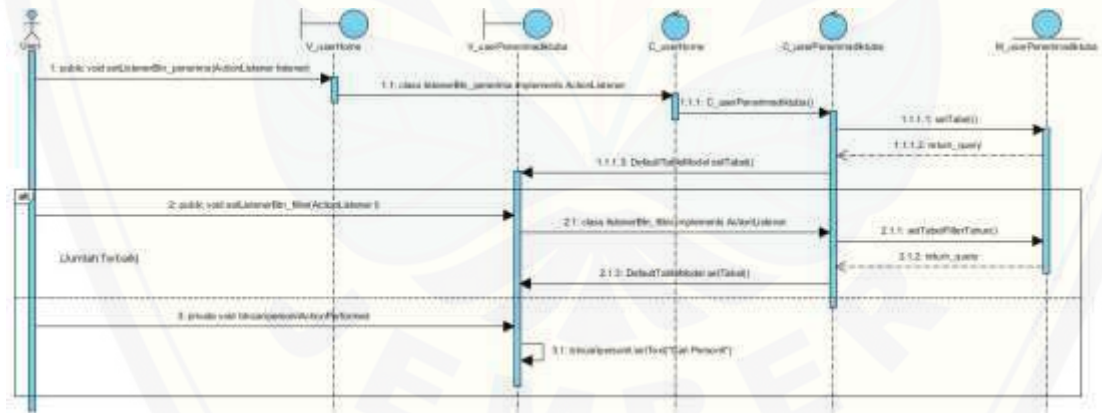


C.16 *Sequence Diagram Admin Edit Bobot*



iGambar C.16 *Sequence Diagram Admin Edit Bobot*

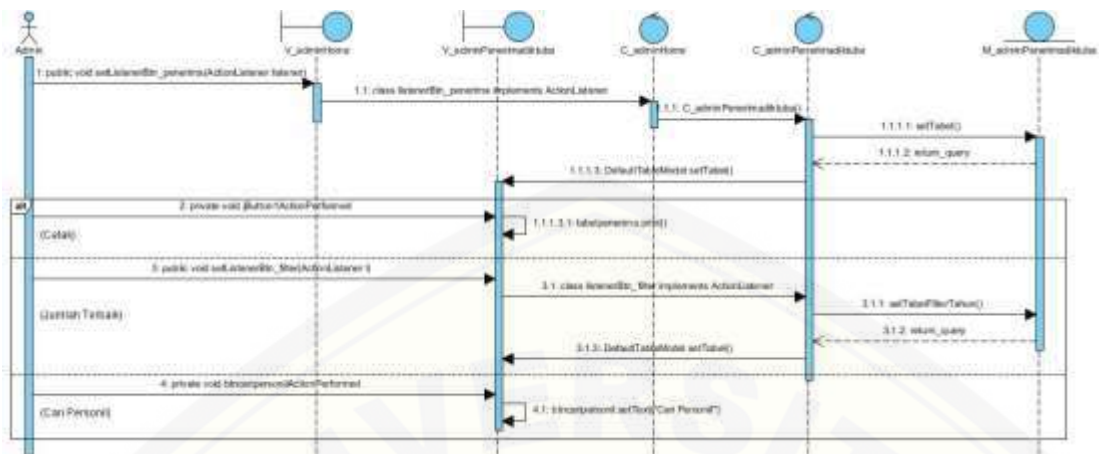
C.17 *Sequence Diagram Operator melihat Penerima DIKTUBA REG*



Gambar C.17 *Sequence Diagram Operator melihat Penerima DIKTUBA REG*



C.18 *Sequence Diagram Admin Melihat Penerima dan Mencetak DIKTUBA REG*



Gambar C.18 *Sequence Diagram Admin Melihat Penerima dan Mencetak DIKTUBA REG*



## Lampiran D. Implementasi Coding

### D.1 Login

```

public class C_login {

    view.V_login log = new view.V_login();
    model.modellogin mlog = new modellogin();

    public static void main(String[] args){

        C_login log = new C_login();

    }

    public C_login() {
        log.setVisible(true);
        log.setListenerLogin(new listenerLogin());
        log.setListenerKeluar(new listenerKeluar());
    }

    class listenerLogin implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent e) {
            Object status = log.getStatus();
            String pass = log.getPassword();
            int ada = mlog.ceklogin(status.toString(), pass);
            if(String.valueOf(status).isEmpty() || pass.isEmpty()){
                JOptionPane.showMessageDialog(log, "Silahkan mengisi Username atau Password");

            }
            else{
                if(ada != 1){
                    JOptionPane.showMessageDialog(log, "Username atau Password salah!");
                }
                else{
                    mlog.login(status.toString(), pass);
                    log.setVisible(false);
                }
            }
        }
    }

    class listenerKeluar implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent e) {
            int ok = JOptionPane.showConfirmDialog(null, "Keluar Dari Sistem Pemilihan DIKTUBA REG?", "E
            if (ok == 0) {
                System.exit(0);
            }
        }
    }
}

```

## D.2 Home Admin

```
public class C_adminHome {

    view.V_adminHome v = new view.V_adminHome();

    public C_adminHome() {
        v.setVisible(true);
        v.setListenerKeluar(new listenerKeluar());
        v.setListenerBtn_home_admin(new listenerBtn_home_admin());
        v.setListenerBtn_penerima(new listenerBtn_penerima());
        v.setListenerBtn_kriteria(new C_adminHome.listenerBtn_kriteria());
        v.setListenerBtn_bobot(new C_adminHome.listenerBtn_bobot());
        v.setListenerBtn_input_admin(new listenerBtn_input_admin());
        v.setListenerLogout(new listenerLogout());
    }

    // ----- BUTTON -----
    class listenerBtn_home_admin implements ActionListener {
        @Override
        public void actionPerformed(ActionEvent e) {
            C_adminHome aa = new C_adminHome();
            v.dispose();
        }
    }

    class listenerBtn_kriteria implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent e) {

            C_atributkesehatanumum aa = new C_atributkesehatanumum();
            v.dispose();
        }
    }

    class listenerBtn_bobot implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent e) {
            C_bobot aa = new C_bobot();
            v.dispose();
        }
    }

    class listenerBtn_input_admin implements ActionListener {
        @Override
        public void actionPerformed(ActionEvent e) {
            C_adminInputPenerima aa = new C_adminInputPenerima();
            v.dispose();
        }
    }

    class listenerLogout implements ActionListener {
        @Override
        public void actionPerformed(ActionEvent e) {
            C_login aa = new C_login();
        }
    }
}
```

```

        v.dispose();
    }
}

class listenerBtn_penerima implements ActionListener {
    @Override
    public void actionPerformed(ActionEvent e) {
        C_adminPenerimaDiktuba aa = new C_adminPenerimaDiktuba();
        v.dispose();
    }
}

class listenerKeluar implements ActionListener {
    @Override
    public void actionPerformed(ActionEvent e) {
        int ok = JOptionPane.showConfirmDialog(null, "Keluar Dari Sistem DIKTUBA REG?", "EXIT", JOptionPane.
        if (ok == 0) {
            System.exit(0);
        }
    }
}
}

```

Gambar D.2 Home Admin

### D.3 Home operator

```

public class C_homeuser {

    view.V_Homeuser v = new view.V_Homeuser();
    // model.M_Homeuser m = new model.M_Homeuser();

    public C_homeuser() {
        v.setVisible(true);
        // v.setTabel(m.setTabel());
        v.setListenerBtn_metode(new listenerBtn_metode());
        v.setListenerBtn_penerima(new listenerBtn_penerima());
        v.setListenerKeluar(new listenerKeluar());
        v.setListenerLogout(new listenerLogout());
        v.setListenerInputPersonil(new listenerInputPersonil());
    }

    //keluar
    class listenerKeluar implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent e) {
            int ok = JOptionPane.showConfirmDialog(null, "Keluar Dari Sistem DIKTUBA REG?", "EXIT", JOp
            if (ok == 0) {
                System.exit(0);
            }
        }
    }

}

class listenerBtn_metode implements ActionListener {

```

```
@Override
public void actionPerformed(ActionEvent e) {
    C_Penghitungandiktuba aa = new C_Penghitungandiktuba();
    v.dispose();
}
}

class listenerBtn_penerima implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        C_userPenerimadiktuba aa = new C_userPenerimadiktuba();
        v.dispose();
    }
}

class listenerLogout implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        C_login aa = new C_login();
        v.dispose();
    }
}

class listenerInputPersonil implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        C_UserInputPenerima cc = new C_UserInputPenerima();
        v.dispose();
    }
}
}
```

Gambar D.3 *Home Operator*

#### D.4 Admin Input Penerima

```
public class C_adminInputPenerima {  
  
    model.M_adminInputPenerima m = new model.M_adminInputPenerima();  
    view.V_admininputPenerima v = new view.V_admininputPenerima();  
  
    public void kosong() {  
        v.setId_personil(m.getID_Auto());  
        v.setpangkat("");  
        v.setNama("");  
        v.setTgl_lahir("");  
        v.setkesatuan("");  
        v.setAlamat("");  
        v.settempat_lahir("");  
        v.setjabatan("");  
        v.setTahun_angkatan("");  
    }  
  
    public C_adminInputPenerima() {  
        v.setVisible(true);  
        v.setTabel(m.setTabel());  
        v.setId_personil(m.getID_Auto());  
        v.setListenerInsert(new listenerInsert());  
        v.setListenerDelete(new listenerDelete());  
        v.setListenerUpdate(new listenerUpdate());  
        v.setListenerKeluar(new listenerKeluar());  
        v.setListenerClean(new listenerClean());  
        v.setListenerBtn_home_admin(new listenerBtn_home_guru());  
    }  
}
```

```
v.setListenerBtn_input_admin(new listenerBtn_input_guru());
}

// -----LISTENER BUTTON-----
class listenerBtn_home_guru implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        C_adminHome aa = new C_adminHome();
        v.dispose();
    }
}

class listenerBtn_input_guru implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        C_adminInputPenerima aa = new C_adminInputPenerima();
        v.dispose();
    }
}

class listenerKeluar implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {

        int ok = JOptionPane.showConfirmDialog(null, "Keluar Dari Sistem?", "EXIT", JOptionPane.YES_NO_
        if (ok == 0) {
            System.exit(0);
        }
    }
}

// ----- LISTENER CHOD -----
class listenerUpdate implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
int tabel = v.getTabel();
        if(tabel == -1){
            JOptionPane.showMessageDialog(v, "Pilih Data Dulu");
            return;
        }
        String pangkat = v.getPangkat();
        String tanggal = v.getTglLahir().toString();
        String nama = v.getNama();
        String kesatuan = v.getKesatuan();
        String alamat = v.getAlamat();
        String tempat_lahir = v.getTempat_lahir();
        String jabatan = v.getJabatan();
        String tahunangkatan = v.getTahun_angkatan();

        if (pangkat.isEmpty()) {
```



```
        JOptionPane.showMessageDialog(v, "Data tidak lengkap!");
    } else if (tanggal.isEmpty()) {
        JOptionPane.showMessageDialog(v, "Data tidak lengkap!");
    } else if (nama.isEmpty()) {
        JOptionPane.showMessageDialog(v, "Data tidak lengkap!");
    } else if (kesatuan.isEmpty()) {
        JOptionPane.showMessageDialog(v, "Data tidak lengkap!");
    } else if (alamat.isEmpty()) {
        JOptionPane.showMessageDialog(v, "Data tidak lengkap!");
    } else if (tempat_lahir.isEmpty()) {
        JOptionPane.showMessageDialog(v, "Data tidak lengkap!");
    } else if (jabatan.isEmpty()) {
        JOptionPane.showMessageDialog(v, "Data tidak lengkap!");
    } else if (tahunangkatan.isEmpty()) {
        JOptionPane.showMessageDialog(v, "Data tidak lengkap!");
    } else {
        m.update(Integer.parseInt(v.getId_personil()), v.getpangkat().toString(), v.getNama());
        JOptionPane.showMessageDialog(v, "UPDATE DATA BERHASIL !!!" );
        v.setTabel(m.setTabel());
        kosong();
        // v.setId_personil(m.getID_Auto());
    }
}
}

class listenerDelete implements ActionListener {

class listenerInsert implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {

        String pangkat = v.getpangkat();
        java.util.Date tanggal = v.getTgllahir();
        String nama = v.getNama();
        String kesatuan = v.getkesatuan();
        String alamat = v.getAlamat();
        String tempat_lahir = v.gettempat_lahir();
        String jabatan = v.getkesatuan();
        String tahunangkatan = v.getTahun_angkatan();

        if (pangkat.isEmpty()) {
            JOptionPane.showMessageDialog(v, "Data tidak lengkap!");
        } else if (String.valueOf(tanggal).isEmpty()) {
            JOptionPane.showMessageDialog(v, "Data tidak lengkap!");
        } else if (nama.isEmpty()) {
            JOptionPane.showMessageDialog(v, "Data tidak lengkap!");
        } else if (kesatuan.isEmpty()) {
            JOptionPane.showMessageDialog(v, "Data tidak lengkap!");
        } else if (alamat.isEmpty()) {
            JOptionPane.showMessageDialog(v, "Data tidak lengkap!");
        } else if (tempat_lahir.isEmpty()) {
            JOptionPane.showMessageDialog(v, "Data tidak lengkap!");
        } else if (jabatan.isEmpty()) {
```

```
class listenerDelete implements ActionListener {  
  
    @Override  
    public void actionPerformed(ActionEvent e) {  
        int tabel = v.getTabel();  
        if(tabel == -1){  
            JOptionPane.showMessageDialog(v, "Pilih Data Dulu");  
            return;  
        }  
        int a = JOptionPane.showConfirmDialog(v, "Apakah Anda Yakin?", "Hapus", JOptionPane.  
        if(a == 0){  
            m.delete(Integer.parseInt(v.getIDHapus()));  
            v.setTabel(m.setTabel());  
            v.setId_personil(m.getID_Auto());  
            kosong();  
        }  
        else{  
            kosong();  
        }  
    }  
}  
}
```

```
class listenerClean implements ActionListener {  
  
    @Override  
    public void actionPerformed(ActionEvent e) {  
        kosong();  
    }  
}
```

Gambar D.4 Admin Input Penerima

## D.5 Admin Penerima DIKTUBA REG

```
public class C_adminPenerimaDiktuba {

    view.V_adminPenerimaDiktuba v = new view.V_adminPenerimaDiktuba();
    model.M_adminPenerimaDiktuba m = new model.M_adminPenerimaDiktuba();

    public C_adminPenerimaDiktuba() {
        v.setVisible(true);
        v.setListenerBtn_keluar(new listenerKeluar());
        v.setListenerBtn_home_admin(new listenerBtn_home_admin());
        v.setListenerBtn_penerima(new listenerBtn_penerima());
        v.setListenerBtn_filter(new listenerBtn_filter());
        v.setTabel(m.setTabel());
        v.setTahun(m.setTahun());
    }

    class listenerBtn_home_admin implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent e) {
            C_adminHome aa = new C_adminHome();
            v.dispose();
        }
    }

    class listenerBtn_penerima implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent e) {
            C_adminPenerimaDiktuba aa = new C_adminPenerimaDiktuba();
            v.dispose();
        }
    }

    class listenerKeluar implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent e) {
            int ok = JOptionPane.showConfirmDialog(null, "Keluar Dari Sistem Pemilihan");
            if (ok == 0) {
                System.exit(0);
            }
        }
    }

    class listenerBtn_filter implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent e) {
            int tahun = v.getTahun2();
            if (tahun == 0) {
                v.setTabel(m.setTabelFilter(v.getFilter()));
            }
            else{

```

```

        v.setTabel(m.setTabelFilterTahun(v.getFilter(), v.getTahun3()));
    }
}
}
}
}

```

Gambar D.5 Admin Penerima DIKTUBA REG

## D.6 Kriteria Akademik Pengetahuan Militer

```

public class C_atributakademikpengetahuanmiliter {

    view.V_atributakademikpengetahuanmiliter v = new view.V_atributakademikpengetahuanmiliter();
    model.M_atributakademikpengetahuanmiliter m = new model.M_atributakademikpengetahuanmiliter();

    public C_atributakademikpengetahuanmiliter() {
        v.setVisible(true);
        v.setTabel(m.setTabel());
        v.setId_atribapm(m.getID_Auto());
        v.setListenerInsert(new C_atributakademikpengetahuanmiliter.listenerInsert());
        v.setListenerDelete(new C_atributakademikpengetahuanmiliter.listenerDelete());
        v.setListenerUpdate(new C_atributakademikpengetahuanmiliter.listenerUpdate());
        v.setListenerClean(new C_atributakademikpengetahuanmiliter.listenerClean());
        v.setListenerKeluar(new listenerKeluar());
        v.setListenerBtn_home(new listenerBtn_home());
        v.setListenerBtn_ku(new listenerBtn_ku());
        v.setListenerBtn_apu(new listenerBtn_apu());
        v.setListenerBtn_apm(new listenerBtn_apm());
        v.setListenerBtn_apmk(new listenerBtn_apmk());
        v.setListenerBtn_psi(new listenerBtn_psi());
        v.setListenerBtn_po(new listenerBtn_po());
        v.setListenerBtn_pot(new listenerBtn_pot());
        v.setListenerBtn_umur(new listenerBtn_umur());
        v.setListenerBtn_jasmanilari(new listenerBtn_jasmanilari());
        v.setListenerBtn_jasmanipushup(new listenerBtn_jasmanipushup());
        v.setListenerBtn_jasmanisitup(new listenerBtn_jasmanisitup());
        v.setListenerBtn_jasmanisatlerun(new listenerBtn_jasmanisatlerun());
        v.setListenerBtn_jasmanirenang(new listenerBtn_jasmanirenang());
    }

    class listenerBtn_apm implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent e) {
            C_atributakademikpengetahuanmiliter aa = new C_atributakademikpengetahuanmiliter();
            v.dispose();
        }
    }
}

```

Gambar D.6 Kriteria Akademik Pengetahuan Militer

## D.7 Kriteria Akademik Pengetahuan Umum

```
public C_atributakademikpengetahuanumum() {
    v.setVisible(true);

    v.setTabel(m.setTabel());
    v.setId_atribapu(m.getID_Auto());
    v.setListenerInsert(new C_atributakademikpengetahuanumum.listenerInsert());
    v.setListenerDelete(new C_atributakademikpengetahuanumum.listenerDelete());
    v.setListenerUpdate(new C_atributakademikpengetahuanumum.listenerUpdate());
    v.setListenerClean(new C_atributakademikpengetahuanumum.listenerClean());
    v.setListenerKeluar(new listenerKeluar());
    v.setListenerBtn_home(new listenerBtn_home());
    v.setListenerBtn_ku(new listenerBtn_ku());
    v.setListenerBtn_apu(new listenerBtn_apu());
    v.setListenerBtn_apm(new listenerBtn_apm());
    v.setListenerBtn_apmk(new listenerBtn_apmk());
    v.setListenerBtn_psi(new listenerBtn_psi());
    v.setListenerBtn_po(new listenerBtn_po());
    v.setListenerBtn_pot(new listenerBtn_pot());
    v.setListenerBtn_umur(new listenerBtn_umur());
    v.setListenerBtn_jasmanilari(new listenerBtn_jasmanilari());
    v.setListenerBtn_jasmanipushup(new listenerBtn_jasmanipushup());
    v.setListenerBtn_jasmanisitup(new listenerBtn_jasmanisitup());
    v.setListenerBtn_jasmanisatlerun(new listenerBtn_jasmanisatlerun());
    v.setListenerBtn_jasmanirenang(new listenerBtn_jasmanirenang());
}

class listenerBtn_apu implements ActionListener {
    @Override
    public void actionPerformed(ActionEvent e) {
        C_atributakademikpengetahuanumum aa = new C_atributakademikpengetahuanumum();
        v.dispose();
    }
}
```

Gambar D.7 Kriteria Akademik Pengetahuan Umum

## D.8 Kriteria Akademik Pengetahuan Militer kecabangan

```
public C_atributakademikpengetahuanmiliterkecabangan() {
    v.setVisible(true);
    v.setTabel(m.setTabel());
    v.setId_atribapmk(m.getID_Auto());
    v.setListenerInsert(new C_atributakademikpengetahuanmiliterkecabangan.listenerInsert());
    v.setListenerDelete(new C_atributakademikpengetahuanmiliterkecabangan.listenerDelete());
    v.setListenerUpdate(new C_atributakademikpengetahuanmiliterkecabangan.listenerUpdate());
    v.setListenerClean(new C_atributakademikpengetahuanmiliterkecabangan.listenerClean());
    v.setListenerKeluar(new listenerKeluar());
    v.setListenerBtn_home(new listenerBtn_home());
    v.setListenerBtn_ku(new listenerBtn_ku());
    v.setListenerBtn_apu(new listenerBtn_apu());
    v.setListenerBtn_apm(new listenerBtn_apm());
    v.setListenerBtn_apmk(new listenerBtn_apmk());
    v.setListenerBtn_psi(new listenerBtn_psi());
    v.setListenerBtn_po(new listenerBtn_po());
    v.setListenerBtn_pot(new listenerBtn_pot());
}

class listenerBtn_apmk implements ActionListener {
    @Override
    public void actionPerformed(ActionEvent e) {
        C_atributakademikpengetahuanmiliterkecabangan aa = new C_atributakademikpengetahuanmiliter
        v.dispose();
    }
}
```

Gambar D.8 Kriteria Akademik Pengetahuan Militer kecabangan



## D.9 Kriteria Psikotes

```
public C_atributpsikotes() {
    v.setVisible(true);
    v.setTabel(m.setTabel());
    v.setId_atribpsi(m.getID_Auto());
    v.setListenerInsert(new C_atributpsikotes.listenerInsert());
    v.setListenerDelete(new C_atributpsikotes.listenerDelete());
    v.setListenerUpdate(new C_atributpsikotes.listenerUpdate());
    v.setListenerClean(new C_atributpsikotes.listenerClean());
    v.setListenerKeluar(new listenerKeluar());
    v.setListenerBtn_home(new listenerBtn_home());
    v.setListenerBtn_ku(new listenerBtn_ku());
    v.setListenerBtn_apu(new listenerBtn_apu());
    v.setListenerBtn_apm(new listenerBtn_apm());
    v.setListenerBtn_apmk(new listenerBtn_apmk());
    v.setListenerBtn_psi(new listenerBtn_psi());
    v.setListenerBtn_po(new listenerBtn_po());
    v.setListenerBtn_pot(new listenerBtn_pot());
    v.setListenerBtn_umur(new listenerBtn_umur());
    v.setListenerBtn_jasmanilari(new listenerBtn_jasmanilari());
    v.setListenerBtn_jasmanipushup(new listenerBtn_jasmanipushup());
    v.setListenerBtn_jasmanisitup(new listenerBtn_jasmanisitup());
    v.setListenerBtn_jasmanisatlerun(new listenerBtn_jasmanisatlerun());
    v.setListenerBtn_jasmanirenang(new listenerBtn_jasmanirenang());
}

class listenerBtn_psi implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        C_atributpsikotes aa = new C_atributpsikotes();
        v.dispose();
    }
}
```

Gambar D.9 Kriteria Psikotes

## D.10 Kriteria Jasmani Lari

```
public C_atributjasmanilari() {
    v.setVisible(true);

    v.setTabel(m.setTabel());
    v.setId_atribjasmanilari(m.getID_Auto());
    v.setListenerInsert(new C_atributjasmanilari.listenerInsert());
    v.setListenerDelete(new C_atributjasmanilari.listenerDelete());
    v.setListenerUpdate(new C_atributjasmanilari.listenerUpdate());
    v.setListenerClean(new C_atributjasmanilari.listenerClean());
    v.setListenerKeluar(new listenerKeluar());
    v.setListenerBtn_home(new listenerBtn_home());
    v.setListenerBtn_ku(new listenerBtn_ku());
    v.setListenerBtn_apu(new listenerBtn_apu());
    v.setListenerBtn_apm(new listenerBtn_apm());
    v.setListenerBtn_apmk(new listenerBtn_apmk());
    v.setListenerBtn_psi(new listenerBtn_psi());
    v.setListenerBtn_po(new listenerBtn_po());
    v.setListenerBtn_pot(new listenerBtn_pot());
    v.setListenerBtn_umur(new listenerBtn_umur());
    v.setListenerBtn_jasmanilari(new listenerBtn_jasmanilari());
    v.setListenerBtn_jasmanipushup(new listenerBtn_jasmanipushup());
    v.setListenerBtn_jasmanisitup(new listenerBtn_jasmanisitup());
    v.setListenerBtn_jasmanisatlerun(new listenerBtn_jasmanisatlerun());
    v.setListenerBtn_jasmanirenang(new listenerBtn_jasmanirenang());
}

class listenerBtn_jasmanilari implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        C_atributjasmanilari aa = new C_atributjasmanilari();
        v.dispose();
    }
}
```

Gambar D.10 Kriteria Jasmani Lari

D.11 Kriteria Jasmani *Push Up*

```
public C_atributjasmanipushup() {
    v.setVisible(true);

    v.setTabel(m.setTabel());
    v.setId_atribjasmanipushup(m.getID_Auto());
    v.setListenerInsert(new C_atributjasmanipushup.listenerInsert());
    v.setListenerDelete(new C_atributjasmanipushup.listenerDelete());
    v.setListenerUpdate(new C_atributjasmanipushup.listenerUpdate());
    v.setListenerClean(new C_atributjasmanipushup.listenerClean());
    v.setListenerKeluar(new listenerKeluar());
    v.setListenerBtn_home(new listenerBtn_home());
    v.setListenerBtn_ku(new listenerBtn_ku());
    v.setListenerBtn_apu(new listenerBtn_apu());
    v.setListenerBtn_apm(new listenerBtn_apm());
    v.setListenerBtn_apmk(new listenerBtn_apmk());
    v.setListenerBtn_psi(new listenerBtn_psi());
    v.setListenerBtn_po(new listenerBtn_po());
    v.setListenerBtn_pot(new listenerBtn_pot());
    v.setListenerBtn_umur(new listenerBtn_umur());
    v.setListenerBtn_jasmanilari(new listenerBtn_jasmanilari());
    v.setListenerBtn_jasmanipushup(new listenerBtn_jasmanipushup());
    v.setListenerBtn_jasmanisitup(new listenerBtn_jasmanisitup());
    v.setListenerBtn_jasmanisatlerun(new listenerBtn_jasmanisatlerun());
    v.setListenerBtn_jasmanirenang(new listenerBtn_jasmanirenang());
}

class listenerBtn_jasmanipushup implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        C_atributjasmanipushup aa = new C_atributjasmanipushup();
        v.dispose();
    }
}
```

Gambar D.11 Kriteria Jasmani *Push Up*

D.12 Kriteria Jasmani *Sitt Up*

```
public C_atributjasmanisitup() {
    v.setVisible(true);

    v.setTabel(m.setTabel());
    v.setId_atribjasmanisitup(m.getID_Auto());
    v.setListenerInsert(new C_atributjasmanisitup.listenerInsert());
    v.setListenerDelete(new C_atributjasmanisitup.listenerDelete());
    v.setListenerUpdate(new C_atributjasmanisitup.listenerUpdate());
    v.setListenerClean(new C_atributjasmanisitup.listenerClean());
    v.setListenerKeluar(new listenerKeluar());
    v.setListenerBtn_home(new listenerBtn_home());
    v.setListenerBtn_ku(new listenerBtn_ku());
    v.setListenerBtn_apu(new listenerBtn_apu());
    v.setListenerBtn_apm(new listenerBtn_apm());
    v.setListenerBtn_apmk(new listenerBtn_apmk());
    v.setListenerBtn_psi(new listenerBtn_psi());
    v.setListenerBtn_po(new listenerBtn_po());
    v.setListenerBtn_pot(new listenerBtn_pot());
    v.setListenerBtn_umur(new listenerBtn_umur());
    v.setListenerBtn_jasmanilari(new listenerBtn_jasmanilari());
    v.setListenerBtn_jasmanipushup(new listenerBtn_jasmanipushup());
    v.setListenerBtn_jasmanisitup(new listenerBtn_jasmanisitup());
    v.setListenerBtn_jasmanisatlerun(new listenerBtn_jasmanisatlerun());
    v.setListenerBtn_jasmanirenang(new listenerBtn_jasmanirenang());
}

class listenerBtn_jasmanisitup implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        C_atributjasmanisitup aa = new C_atributjasmanisitup();
        v.dispose();
    }
}
```

Gambar D.12 *Sitt Up*

D.13 Kriteria Jasmani *Satle Run*

```
public C_atributjasmanisatlerun() {
    v.setVisible(true);

    v.setTabel(m.setTabel());
    v.setId_atribjasmanisatlerun(m.getID_Auto());
    v.setListenerInsert(new C_atributjasmanisatlerun.listenerInsert());
    v.setListenerDelete(new C_atributjasmanisatlerun.listenerDelete());
    v.setListenerUpdate(new C_atributjasmanisatlerun.listenerUpdate());
    v.setListenerClean(new C_atributjasmanisatlerun.listenerClean());
    v.setListenerKeluar(new listenerKeluar());
    v.setListenerBtn_home(new listenerBtn_home());
    v.setListenerBtn_ku(new listenerBtn_ku());
    v.setListenerBtn_apu(new listenerBtn_apu());
    v.setListenerBtn_apm(new listenerBtn_apm());
    v.setListenerBtn_apmk(new listenerBtn_apmk());
    v.setListenerBtn_psi(new listenerBtn_psi());
    v.setListenerBtn_po(new listenerBtn_po());
    v.setListenerBtn_pot(new listenerBtn_pot());
    v.setListenerBtn_umur(new listenerBtn_umur());
    v.setListenerBtn_jasmanilari(new listenerBtn_jasmanilari());
    v.setListenerBtn_jasmanipushup(new listenerBtn_jasmanipushup());
    v.setListenerBtn_jasmanisitup(new listenerBtn_jasmanisitup());
    v.setListenerBtn_jasmanisatlerun(new listenerBtn_jasmanisatlerun());
    v.setListenerBtn_jasmanirenang(new listenerBtn_jasmanirenang());
}

class listenerBtn_jasmanisatlerun implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        C_atributjasmanisatlerun aa = new C_atributjasmanisatlerun();
        v.dispose();
    }
}
```

Gambar D.13 Kriteria Jasmani *Satle Run*



## D.14 Kriteria Jasmani Renang

```
public C_atributjasmanirenang() {
    v.setVisible(true);

    v.setTabel(m.setTabel());
    v.setId_atribjasmanirenang(m.getID_Auto());
    v.setListenerInsert(new C_atributjasmanirenang.listenerInsert());
    v.setListenerDelete(new C_atributjasmanirenang.listenerDelete());
    v.setListenerUpdate(new C_atributjasmanirenang.listenerUpdate());
    v.setListenerClean(new C_atributjasmanirenang.listenerClean());
    v.setListenerKeluar(new listenerKeluar());
    v.setListenerBtn_home(new listenerBtn_home());
    v.setListenerBtn_ku(new listenerBtn_ku());
    v.setListenerBtn_apu(new listenerBtn_apu());
    v.setListenerBtn_apm(new listenerBtn_apm());
    v.setListenerBtn_apmk(new listenerBtn_apmk());
    v.setListenerBtn_psi(new listenerBtn_psi());
    v.setListenerBtn_po(new listenerBtn_po());
    v.setListenerBtn_pot(new listenerBtn_pot());
    v.setListenerBtn_umur(new listenerBtn_umur());
    v.setListenerBtn_jasmanilari(new listenerBtn_jasmanilari());
    v.setListenerBtn_jasmanipushup(new listenerBtn_jasmanipushup());
    v.setListenerBtn_jasmanisitup(new listenerBtn_jasmanisitup());
    v.setListenerBtn_jasmanisatlerun(new listenerBtn_jasmanisatlerun());
    v.setListenerBtn_jasmanirenang(new listenerBtn_jasmanirenang());
}

class listenerBtn_jasmanirenang implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        C_atributjasmanirenang aa = new C_atributjasmanirenang();
        v.dispose();
    }
}
```

Gambar D.14 Kriteria Jasmani Renang



## D.15 Kriteria Umur

```
public C_tributumur() {
    v.setVisible(true);

    v.setTabel(m.setTabel());
    v.setId_tributumur(m.getID_Auto());
    v.setListenerInsert(new C_tributumur.listenerInsert());
    v.setListenerDelete(new C_tributumur.listenerDelete());
    v.setListenerUpdate(new C_tributumur.listenerUpdate());
    v.setListenerClean(new C_tributumur.listenerClean());
    v.setListenerKeluar(new listenerKeluar());
    v.setListenerBtn_home(new listenerBtn_home());
    v.setListenerBtn_ku(new listenerBtn_ku());
    v.setListenerBtn_apu(new listenerBtn_apu());
    v.setListenerBtn_apm(new listenerBtn_apm());
    v.setListenerBtn_apmk(new listenerBtn_apmk());
    v.setListenerBtn_psi(new listenerBtn_psi());
    v.setListenerBtn_po(new listenerBtn_po());
    v.setListenerBtn_pot(new listenerBtn_pot());
    v.setListenerBtn_umur(new listenerBtn_umur());
    v.setListenerBtn_jasmanilari(new listenerBtn_jasmanilari());
    v.setListenerBtn_jasmanipushup(new listenerBtn_jasmanipushup());
    v.setListenerBtn_jasmanisitup(new listenerBtn_jasmanisitup());
    v.setListenerBtn_jasmanisatlerun(new listenerBtn_jasmanisatlerun());
    v.setListenerBtn_jasmanirenang(new listenerBtn_jasmanirenang());
}

class listenerBtn_umur implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        C_tributumur aa = new C_tributumur();
        v.dispose();
    }
}
```

Gambar D.15 Kriteria Umur

## D.16 Kriteria Kesehatan Umum

```
public C_atributkesehatanumum() {
    v.setVisible(true);
    v.setTabel(m.setTabel());
    v.setId_atribku(m.getID_Auto());
    v.setListenerInsert(new listenerInsert());
    v.setListenerDelete(new listenerDelete());
    v.setListenerUpdate(new listenerUpdate());
    v.setListenerClean(new listenerClean());
    v.setListenerKeluar(new listenerKeluar());
    v.setListenerBtn_home(new listenerBtn_home());
    v.setListenerBtn_ku(new listenerBtn_kesehatanumum());
    v.setListenerBtn_apu(new listenerBtn_apu());
    v.setListenerBtn_apm(new listenerBtn_apm());
    v.setListenerBtn_apmk(new listenerBtn_apmk());
    v.setListenerBtn_psi(new listenerBtn_psi());
    v.setListenerBtn_po(new listenerBtn_po());
    v.setListenerBtn_pot(new listenerBtn_pot());
    v.setListenerBtn_umur(new listenerBtn_umur());
    v.setListenerBtn_jasmanilari(new listenerBtn_jasmanilari());
    v.setListenerBtn_jasmanipushup(new listenerBtn_jasmanipushup());
    v.setListenerBtn_jasmanisitup(new listenerBtn_jasmanisitup());
    v.setListenerBtn_jasmanisatlerun(new listenerBtn_jasmanisatlerun());
    v.setListenerBtn_jasmanirenang(new listenerBtn_jasmanirenang());
}

class listenerBtn_kesehatanumum implements ActionListener {
    @Override
    public void actionPerformed(ActionEvent e) {
        C_atributkesehatanumum aa = new C_atributkesehatanumum();
        v.dispose();
    }
}
```

Gambar D.16 Kriteria Kesehatan Umum

## D.17 Kriteria Prestasi Olahraga

```
public C_atributprestasiolahraga() {
    v.setVisible(true);

    v.setTabel(m.setTabel());
    v.setId_atribpo(m.getID_Auto());
    v.setListenerInsert(new C_atributprestasiolahraga.listenerInsert());
    v.setListenerDelete(new C_atributprestasiolahraga.listenerDelete());
    v.setListenerUpdate(new C_atributprestasiolahraga.listenerUpdate());
    v.setListenerClean(new C_atributprestasiolahraga.listenerClean());
    v.setListenerKeluar(new listenerKeluar());
    v.setListenerBtn_home(new listenerBtn_home());
    v.setListenerBtn_ku(new listenerBtn_ku());
    v.setListenerBtn_apu(new listenerBtn_apu());
    v.setListenerBtn_apm(new listenerBtn_apm());
    v.setListenerBtn_apmk(new listenerBtn_apmk());
    v.setListenerBtn_psi(new listenerBtn_psi());
    v.setListenerBtn_po(new listenerBtn_po());
    v.setListenerBtn_pot(new listenerBtn_pot());
    v.setListenerBtn_umur(new listenerBtn_umur());
    v.setListenerBtn_jasmanilari(new listenerBtn_jasmanilari());
    v.setListenerBtn_jasmanipushup(new listenerBtn_jasmanipushup());
    v.setListenerBtn_jasmanisitup(new listenerBtn_jasmanisitup());
    v.setListenerBtn_jasmanisatlerun(new listenerBtn_jasmanisatlerun());
    v.setListenerBtn_jasmanirenang(new listenerBtn_jasmanirenang());
}

class listenerBtn_po implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        C_atributprestasiolahraga aa = new C_atributprestasiolahraga();
        v.dispose();
    }
}
```

Gambar D.17 Kriteria Prestasi Olahraga

## D.18 Kriteria Prestasi Operasi Tempur

```
public C_atributprestasioperasitempur() {  
    v.setVisible(true);  
    v.setTabel(m.setTabel());  
    v.setId_atribpot(m.getID_Auto());  
    v.setListenerInsert(new listenerInsert());  
    v.setListenerDelete(new C_atributprestasioperasitempur.listenerDelete());  
    v.setListenerUpdate(new C_atributprestasioperasitempur.listenerUpdate());  
    v.setListenerClean(new C_atributprestasioperasitempur.listenerClean());  
    v.setListenerKeluar(new listenerKeluar());  
    v.setListenerBtn_home(new listenerBtn_home());  
    v.setListenerBtn_ku(new listenerBtn_ku());  
    v.setListenerBtn_apu(new listenerBtn_apu());  
    v.setListenerBtn_apm(new listenerBtn_apm());  
    v.setListenerBtn_apmk(new listenerBtn_apmk());  
    v.setListenerBtn_psi(new listenerBtn_psi());  
    v.setListenerBtn_po(new listenerBtn_po());  
    v.setListenerBtn_pot(new listenerBtn_pot());  
    v.setListenerBtn_umur(new listenerBtn_umur());  
    v.setListenerBtn_jasmanilari(new listenerBtn_jasmanilari());  
    v.setListenerBtn_jasmanipushup(new listenerBtn_jasmanipushup());  
    v.setListenerBtn_jasmanisitup(new listenerBtn_jasmanisitup());  
    v.setListenerBtn_jasmanisatlerun(new listenerBtn_jasmanisatlerun());  
    v.setListenerBtn_jasmanirenang(new listenerBtn_jasmanirenang());  
}  
  
class listenerBtn_pot implements ActionListener {  
  
    @Override  
    public void actionPerformed(ActionEvent e) {  
        C_atributprestasioperasitempur aa = new C_atributprestasioperasitempur();  
        v.dispose();  
    }  
}
```

Gambar D.18 Kriteria Prestasi Operasi Tempur

## D.19 Operator Melihat Penerima DIKTUBA REG

```
public class C_userPenerimaDiktuba {

    view.V_userPenerimaDiktuba v = new view.V_userPenerimaDiktuba();
    model.M_userPenerimaDiktuba m = new model.M_userPenerimaDiktuba();

    public C_userPenerimaDiktuba() {
        v.setVisible(true);
        v.setListenerBtn_keluar(new listenerKeluar());
        v.setListenerBtn_home_user(new listenerBtn_home_user());
        v.setListenerBtn_penerima(new listenerBtn_penerima());
        v.setListenerBtn_filter(new listenerBtn_filter());
        v.setListenerBtn_cari(new listenerCariTahun());
        v.setListenerBtn_caripersonil(new listenerCariPersonil());
        v.setListenerBtn_filter(new listenerFilterJumlah());
        v.setTabel(m.setTabel());
        v.setTahun(m.setTahun());
    }

    class listenerBtn_home_user implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent e) {
            C_homeuser aa = new C_homeuser();
            v.dispose();
        }
    }

    class listenerBtn_penerima implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent e) {
            C_userPenerimaDiktuba aa = new C_userPenerimaDiktuba();
            v.dispose();
        }
    }

    class listenerKeluar implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent e) {
            int ok = JOptionPane.showConfirmDialog(null, "Keluar Dari Sistem Pemili");
            if (ok == 0) {
                System.exit(0);
            }
        }
    }

    class listenerBtn_filter implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent e) {
            int tahun = v.getTahun2();
            if (tahun == 0) {
                v.setTabel(m.setTabelFilter(v.getFilter()));
            }
            else{

```



```
        v.setTabel(m.setTabelFilterTahun(v.getFilter(), v.getTahun3()));
    }
}

class listenerCariTahun implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        String tahun = v.getTahun3();
        v.setTabel(m.setTabelTahunAngkatan(tahun));
    }
}

class listenerFilterJumlah implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        String tahun = v.getTahun3();
        String nilai = v.getFilter();
        v.setTabel(m.setTabelFilterTahun(nilai, tahun));
    }
}

class listenerCariPersonil implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        String tahun = v.getTahun3();
        v.setTabel(m.setTabelTahunAngkatan(tahun));
    }
}

class listenerFilterJumlah implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        String tahun = v.getTahun3();
        String nilai = v.getFilter();
        v.setTabel(m.setTabelFilterTahun(nilai, tahun));
    }
}

class listenerCariPersonil implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        String nrp = v.getNrp();
        String angkatan = v.getTahunAngkatan();
        v.setTabel(m.setTabelCariPersonil(angkatan, nrp));
        System.out.println(nrp+" "+angkatan);
    }
}
}
```

Gambar D.19 Operator Melihat Penerima DIKTUBA REG



## D.20 Admin Melihat Penerima DIKTUBA REG

```
public class C_adminPenerimadiktuba {

    view.V_adminPenerimadiktuba v = new view.V_adminPenerimadiktuba();
    model.M_adminPenerimadiktuba m = new model.M_adminPenerimadiktuba();

    public C_adminPenerimadiktuba() {
        v.setVisible(true);
        v.setListenerBtn_keluar(new listenerKeluar());
        v.setListenerBtn_home_admin(new listenerBtn_home_admin());
        v.setListenerBtn_penerima(new listenerBtn_penerima());
        v.setListenerBtn_filter(new listenerBtn_filter());
        v.setTabel(m.setTabel());
        v.setTahun(m.setTahun());
    }

    class listenerBtn_home_admin implements ActionListener {

        @Override
        public void actionPerformed(ActionEvent e) {
            C_adminHome aa = new C_adminHome();
            v.dispose();
        }

        v.dispose();
    }

}

class listenerKeluar implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        int ok = JOptionPane.showConfirmDialog(null, "Keluar Dari Sistem Pemilihan");
        if (ok == 0) {
            System.exit(0);
        }
    }

}

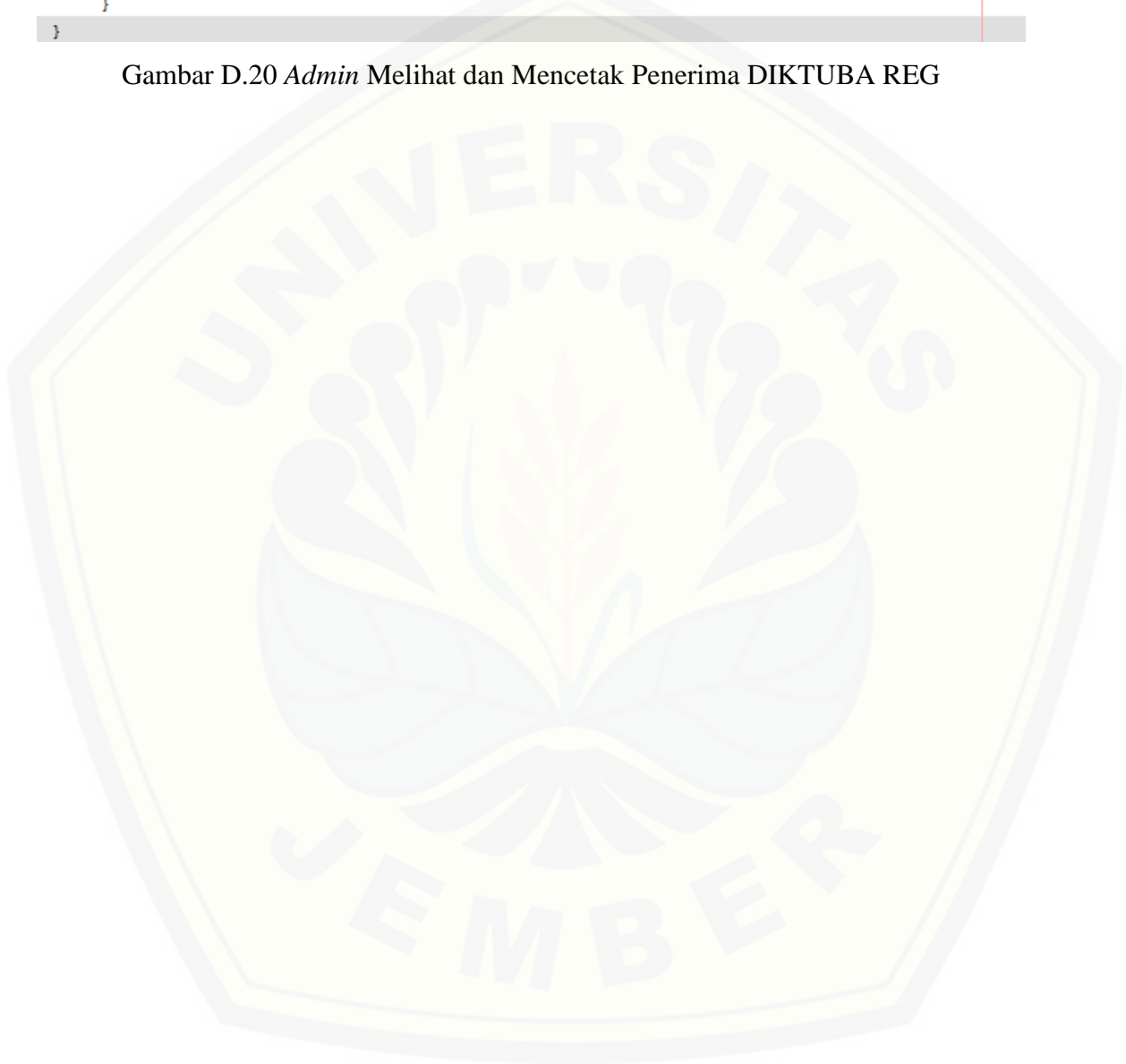
class listenerBtn_filter implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        int tahun = v.getTahun2();
        if(tahun == 0){
            v.setTabel(m.setTabelFilter(v.getFilter()));
        }
        else{
            v.setTabel(m.setTabelFilterTahun(v.getFilter(), v.getTahun3()));
        }
    }

}
```

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
    MessageFormat header = new MessageFormat("Daftar Penerima");  
    MessageFormat footer = new MessageFormat("page{0,number,integer}");  
  
    try {  
        tabelpenerima.print(JTable.PrintMode.NORMAL, header, footer);  
    } catch (java.awt.print.PrinterException e) {  
        System.err.format("Cannot print %s%n", e.getMessage());  
    }  
}
```

Gambar D.20 Admin Melihat dan Mencetak Penerima DIKTUBA REG



## Lampiran E. Black Box Testing

Tabel E.1 Black Box Testing

No.	Menu	Fungsi	Kasus	Hasil	Keterangan
1.	Login	Menu ini berfungsi sebagai keamanan sistem. memilah <i>user</i> yang dapat menggunakan sistem ini.	Ketika <i>user</i> memilih hak akses nama <i>user</i> dan password dengan benar	Menampilkan halaman sesuai level <i>user</i>	Berhasil
2	Home	Menampilkan deskripsi fitur	Ketika <i>user</i> telah melakukan login, sistem akan menampilkan halaman home yang berisi deskripsi	Menampilkan deskripsi fitur	Berhasil
3.	Mengelola DIKTUBA REG	Untuk menambah, mengedit, menghapus dan	Ketika <i>user</i> klik menu mengelola DIKTUBA REG	Menampilkan halaman calon pendaftar DIKTUBA REG	Berhasil

No.	Menu	Fungsi	Kasus	Hasil	Keterangan
		mengosongkan data pendaftar			
			Ketika <i>user</i> klik tombol tambah	Menampilkan form tambah calon pendaftar DIKTUBA REG	Berhasil
			Ketika <i>user</i> klik tombol <i>edit</i>	Menampilkan form <i>edit</i> calon pendaftar DIKTUBA REG	Berhasil
			Ketika <i>user</i> klik tombol hapus	Menampilkan form peringatan	Berhasil
			Ketika <i>user</i> klik tombol kosongkan	Menampilkan form calon pendaftar DIKTUBA REG	Berhasil
4.	Penerima DIKTUBA REG	Untuk melihat daftar penerima DIKTUBA REG per-periode	Ketika <i>user</i> 2 klik menu penerima DIKTUBA REG	Menampilkan halaman daftar penerima DIKTUBA REG per-periode	Berhasil
			Ketika <i>user</i> 2 klik tombol cari	Menampilkan halaman	Berhasil

No.	Menu	Fungsi	Kasus	Hasil	Keterangan
			tahun	daftar penerima DIKTUBA REG berdasarkan tahun	
			Ketika <i>user 2</i> klik tombol filter nilai minimum	Menampilkan halaman daftar penerima DIKTUBA REG berdasarkan nilai minimum	Berhasil
			Ketika <i>user 2</i> klik tombol cari personil	Menampilkan halaman daftar penerima DIKTUBA REG berdasarkan pangkat dan tahun	Berhasil
5.	Mengelola kriteria	Untuk menambah, mengedit, menghapus dan mengosongkan	Ketika <i>user 1</i> klik menu mengelola kriteria	Menampilkan halaman kriteria kesehatan umum	Berhasil



No.	Menu	Fungsi	Kasus	Hasil	Keterangan
		kriteria			
			Ketika <i>user</i> 1 klik tombol tambah	Menampilkan form tambah kesehatan umum	Berhasil
			Ketika <i>user</i> 1 klik tombol <i>edit</i>	Menampilkan form <i>edit</i> kesehatan umum	Berhasil
			Ketika <i>user</i> 1 klik tombol hapus	Menampilkan peringatan	Berhasil
			Ketika <i>user</i> 1 klik tombol kosongkan	Menampilkan form tambah kesehatan umum	Berhasil
			Ketika <i>user</i> 1 klik menu mengelola kriteria akademik pengetahuan umum	Menampilkan halaman kriteria akademik pengetahuan umum	Berhasil
			Ketika <i>user</i> 1 klik tombol tambah	Menampilkan form tambah akademik pengetahuan umum	Berhasil
			Ketika <i>user</i> 1 klik tombol <i>edit</i>	Menampilkan form <i>edit</i> akademik pengetahuan umum	Berhasil

No.	Menu	Fungsi	Kasus	Hasil	Keterangan
			Ketika <i>user</i> 1 klik tombol hapus	Menampilkan peringatan	Berhasil
			Ketika <i>user</i> 1 klik tombol kosongkan	Menampilkan form tambah akademik pengetahuan umum	Berhasil
			Ketika <i>user</i> 1 klik menu mengelola kriteria akademik pengetahuan militer	Menampilkan halaman kriteria akademik pengetahuan militer	Berhasil
			Ketika <i>user</i> 1 klik tombol tambah	Menampilkan form tambah akademik pengetahuan militer	Berhasil
			Ketika <i>user</i> 1 klik tombol <i>edit</i>	Menampilkan form <i>edit</i> akademik pengetahuan militer	Berhasil
			Ketika <i>user</i> klik tombol hapus	Menampilkan peringatan	Berhasil
			Ketika <i>user</i> 1 klik tombol kosongkan	Menampilkan form tambah akademik	Berhasil

No.	Menu	Fungsi	Kasus	Hasil	Keterangan
				pengetahuan militer	
			Ketika <i>user</i> 1 klik menu mengelola kriteria akademik pengetahuan militer kecabangan	Menampilkan halaman kriteria akademik pengetahuan militer kecabangan	Berhasil
			Ketika <i>user</i> 1 klik tombol tambah	Menampilkan form tambah akademik pengetahuan militer kecabangan	Berhasil
			Ketika <i>user</i> 1 klik tombol <i>edit</i>	Menampilkan form <i>edit</i> akademik pengetahuan militer kecabangan	Berhasil
			Ketika <i>user</i> 1 klik tombol hapus	Menampilkan peringatan	Berhasil
			Ketika <i>user</i> 1 klik tombol kosongkan	Menampilkan form tambah akademik pengetahuan militer kecabangan	Berhasil

No.	Menu	Fungsi	Kasus	Hasil	Keterangan
			Ketika <i>user</i> 1 klik menu mengelola kriteria psikotes	Menampilkan halaman kriteria psikotes	Berhasil
			Ketika <i>user</i> 1 klik tombol tambah	Menampilkan form tambah psikotes	Berhasil
			Ketika <i>user</i> 1 klik tombol <i>edit</i>	Menampilkan form <i>edit</i> psikotes	Berhasil
			Ketika <i>user</i> 1 klik tombol hapus	Menampilkan peringatan	Berhasil
			Ketika <i>user</i> 1 klik tombol kosongkan	Menampilkan form tambah psikotes	Berhasil
			Ketika <i>user</i> 1 klik menu mengelola kriteria prestasi olahraga	Menampilkan halaman kriteria prestasi olahraga	Berhasil
			Ketika <i>user</i> 1 klik tombol tambah	Menampilkan form tambah prestasi olahraga	Berhasil
			Ketika <i>user</i> 1 klik tombol <i>edit</i>	Menampilkan form <i>edit</i> prestasi olahraga	Berhasil
			Ketika <i>user</i> 1 klik tombol hapus	Menampilkan peringatan	Berhasil

No.	Menu	Fungsi	Kasus	Hasil	Keterangan
			Ketika <i>user</i> 1 klik tombol kosongkan	Menampilkan form tambah prestasi olahraga	Berhasil
			Ketika <i>user</i> 1 klik menu mengelola kriteria prestasi operasi tempur	Menampilkan halaman kriteria prestasi operasi tempur	Berhasil
			Ketika <i>user</i> 1 klik tombol tambah	Menampilkan form tambah prestasi operasi tempur	Berhasil
			Ketika <i>user</i> 1 klik tombol <i>edit</i>	Menampilkan form <i>edit</i> prestasi operasi tempur	Berhasil
			Ketika <i>user</i> 1 klik tombol hapus	Menampilkan peringatan	Berhasil
			Ketika <i>user</i> 1 klik tombol kosongkan	Menampilkan form tambah prestasi operasi tempur	Berhasil
			Ketika <i>user</i> 1 klik menu mengelola kriteria umur	Menampilkan halaman kriteria umur	Berhasil
			Ketika <i>user</i> 1 klik tombol tambah	Menampilkan form	Berhasil

No.	Menu	Fungsi	Kasus	Hasil	Keterangan
				tambah umur	
			Ketika <i>user</i> 1 klik tombol <i>edit</i>	Menampilkan form <i>edit</i> umur	Berhasil
			Ketika <i>user</i> 1 klik tombol hapus	Menampilkan peringatan	Berhasil
			Ketika <i>user</i> 1 klik tombol kosongkan	Menampilkan form tambah umur	Berhasil
			Ketika <i>user</i> 1 klik menu mengelola kriteria jasmani lari	Menampilkan halaman kriteria jasmani lari	Berhasil
			Ketika <i>user</i> 1 klik tombol tambah	Menampilkan form tambah jasmani lari	Berhasil
			Ketika <i>user</i> 1 klik tombol <i>edit</i>	Menampilkan form <i>edit</i> jasmani lari	Berhasil
			Ketika <i>user</i> 1 klik tombol hapus	Menampilkan peringatan	Berhasil
			Ketika <i>user</i> 1 klik tombol kosongkan	Menampilkan form tambah jasmani lari	Berhasil
			Ketika <i>user</i> 1 klik menu mengelola	Menampilkan halaman	Berhasil



No.	Menu	Fungsi	Kasus	Hasil	Keterangan
			kriteria jasmani <i>push up</i>	kriteria jasmani <i>push up</i>	
			Ketika <i>user</i> 1 klik tombol tambah	Menampilkan form tambah jasmani <i>push up</i>	Berhasil
			Ketika <i>user</i> 1 klik tombol <i>edit</i>	Menampilkan form <i>edit</i> jasmani <i>push up</i>	Berhasil
			Ketika <i>user</i> 1 klik tombol hapus	Menampilkan peringatan	Berhasil
			Ketika <i>user</i> 1 klik tombol kosongkan	Menampilkan form tambah jasmani <i>push up</i>	Berhasil
			Ketika <i>user</i> 1 klik menu mengelola kriteria jasmani <i>sitt up</i>	Menampilkan halaman kriteria jasmani <i>sitt up</i>	Berhasil
			Ketika <i>user</i> 1 klik tombol tambah	Menampilkan form tambah jasmani <i>sitt up</i>	Berhasil
			Ketika <i>user</i> 1 klik tombol <i>edit</i>	Menampilkan form <i>edit</i> jasmani <i>sitt up</i>	Berhasil
			Ketika <i>user</i> 1 klik tombol hapus	Menampilkan peringatan	Berhasil
			Ketika <i>user</i> 1 klik tombol	Menampilkan form	Berhasil

No.	Menu	Fungsi	Kasus	Hasil	Keterangan
			kosongkan	tambah jasmani sitt up	
			Ketika <i>user</i> 1 klik menu mengelola kriteria jasmani <i>satle run</i>	Menampilkan halaman kriteria jasmani <i>satle run</i>	Berhasil
			Ketika <i>user</i> 1 klik tombol tambah	Menampilkan form tambah jasmani <i>satle run</i>	Berhasil
			Ketika <i>user</i> 1 klik tombol <i>edit</i>	Menampilkan form <i>edit</i> jasmani <i>satle run</i>	Berhasil
			Ketika <i>user</i> 1 klik tombol hapus	Menampilkan peringatan	Berhasil
			Ketika <i>user</i> 1 klik tombol kosongkan	Menampilkan form tambah jasmani <i>satle run</i>	Berhasil
			Ketika <i>user</i> 1 klik menu mengelola kriteria jasmani renang	Menampilkan halaman kriteria jasmani renang	Berhasil
			Ketika <i>user</i> 1 klik tombol tambah	Menampilkan form tambah jasmani renang	Berhasil
			Ketika <i>user</i> 1 klik tombol <i>edit</i>	Menampilkan form <i>edit</i> jasmani renang	Berhasil
			Ketika <i>user</i> 1 klik tombol hapus	Menampilkan peringatan	Berhasil

No.	Menu	Fungsi	Kasus	Hasil	Keterangan
			Ketika <i>user</i> 1 klik tombol kosongkan	Menampilkan form tambah jasmani renang	Berhasil
6.	<i>Edit bobot</i>	Untuk mengedit, mengecek total nilai dan mengosongkan bobot	Ketika <i>user</i> 1 klik menu mengedit bobot	Menampilkan halaman nilai bobot	Berhasil
			Ketika <i>user</i> 1 klik tombol <i>edit</i>	Menampilkan form <i>edit</i> nilai bobot	Berhasil
			Ketika <i>user</i> 1 klik tombol cek total nilai	Menampilkan peringatan	Berhasil
			Ketika <i>user</i> 1 klik tombol kosongkan	Menampilkan form nilai bobot	Berhasil
7.	Penerima DIKTUB A REG	Untuk melihat daftar penerima DIKTUBA REG	Ketika <i>user</i> 1 klik menu penerima DIKTUBA REG	Menampilkan halaman daftar penerima DIKTUBA REG per-	Berhasil

No.	Menu	Fungsi	Kasus	Hasil	Keterangan
		per-periode		periode	
			Ketika <i>user</i> 1 klik tombol cari tahun	Menampilkan halaman daftar penerima DIKTUBA REG berdasarkan tahun	Berhasil
			Ketika <i>user</i> 1 klik tombol filter nilai minimum	Menampilkan halaman daftar penerima DIKTUBA REG berdasarkan nilai minimum	Berhasil
			Ketika <i>user</i> 1 klik tombol cari personil	Menampilkan halaman daftar penerima DIKTUBA REG berdasarkan pangkat dan tahun	Berhasil
			Ketika <i>User</i> 1 klik tombol cetak	Mencetak hasil rekapan hasil rekapan	Berhasil

