



**SISTEM PENDUKUNG KEPUTUSAN PENEMPATAN GURU  
SEKOLAH DASAR NEGERI DI KABUPATEN JEMBER  
MENGUNAKAN METODE *PROFILE MATCHING***

**SKRIPSI**

Oleh

**Rizky Arif Firmansyah  
NIM 102410101025**

**PROGRAM STUDI SISTEM INFORMASI  
UNIVERSITAS JEMBER  
2015**



**SISTEM PENDUKUNG KEPUTUSAN PENEMPATAN GURU  
SEKOLAH DASAR NEGERI DI KABUPATEN JEMBER  
MENGUNAKAN METODE *PROFILE MATCHING***

**SKRIPSI**

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

oleh:

**Rizky Arif Firmansyah  
NIM 102410101025**

**PROGRAM STUDI SISTEM INFORMASI  
UNIVERSITAS JEMBER  
2015**

## PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

1. Ibunda Sundari, Ayahanda Kasim Kuswanto, Adek Rini Agustin Hidayatullah dan keluarga besar tercinta yang telah memberikan dukungan moril, doa dan petuahnya untuk selalu rajin dan bekerja keras dalam hal apapun;
2. Para Pembimbing, Guru dan Pahlawan Tanpa Tanda Jasa sejak TK hingga kuliah yang telah mengajari, memberikan semua ilmunya, dan kesabaran untuk terus membimbing supaya jadi lebih baik lagi;
3. Sahabat terbaik Eka Adji Saharudin (bossadji), Hawwin Mardhiana (Duduk), Ratih Pravitasari, Rizqy Ananta (Oki), Syam Oby M. (Obyto), A. Fauzan Imroni (Ozan), Juniar Priaditama (Junho), Fitriyana Dewi (Pipit), Ardiyanta Prasetywan (Tata), Musawiru Alam S (Wir), Anggi Anugrahdika P , Ach. Ryan Dharmawan (Mawud), Rony S.W (Joe), Yanuarida T.C (Yanster), Abdul Roqib;
4. Partner skripsi seperjuangan Rastra Revolusiane (Revo), Duhita Hastungkara (Dudu), Zaqiyah Qurotul Aini (Ain), Tri Rofiana (Opik), M. Khasib C.U (Umam), Apyu Nila D, Lisna Arifky A , Erwin Dwi Putra, Vivi Rahmawati, Hasan Jindan (Arab), Rahmadinata Syafaat (Rasya), Yusa Dwi (Asuy), Indra Yusuf Kinartha, Arbi Darmaja, Brian Hidayatullah ;
5. Semua teman-teman Sistem Informasi 2010 (ZerOne);
6. Keluarga Besar Galaxy Kost, Djail Kost dan Kontrakan FOK;
7. Adinda Hawwin Mardhiana yang selalu memberikan semangat dan dukungan yang tidak pernah habisnya untuk segera menyelesaikan skripsi dan lulus;
8. Keluarga Besar Program Studi Sistem Informasi dan Almamater Universitas Jember yang selalu kubanggakan sebagai wadah dan bagian yang penting dalam menuntut ilmu dan belajar tentang pengalaman hidup;
9. Laptop HP pavilion G4, Ransel Bodypack dan Motor kesayangan Jupiter Z yang selalu mengantar untuk menuntut ilmu.

## MOTTO

“Maka sesungguhnya bersama kesulitan ada kemudahan. Maka apabila engkau telah selesai (dari sesuatu urusan), tetaplah bekerja keras (untuk urusan yang lain). Dan hanya kepada Tuhanmulah hendaknya engkau berharap.”

(Q.S Al-Insyirah, 6-8)

“Seperti mendaki gunung, hidup punya tujuan yang harus didaki dan dituruni.”

(Retno Lestari Priansari Marsudi)

“The greatest pleasure in life is doing what people say you cannot do.”

(Walter Bagehot)

## PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Rizky Arif Firmansyah

NIM : 102410101025

Menyatakan dengan sesungguhnya bahwa karya tulis ilmiah dengan judul **“Sistem Pendukung Keputusan Penempatan Guru Sekolah Dasar Negeri di Kabupaten Jember menggunakan Metode *Profile Matching*”** adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada institusi manapun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

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

Jember, 11 Maret 2015

Yang menyatakan,

Rizky Arif Firmansyah

NIM 10241010102

**SKRIPSI**

**SISTEM PENDUKUNG KEPUTUSAN PENEMPATAN GURU  
SEKOLAH DASAR NEGERI DI KABUPATEN JEMBER  
MENGUNAKAN METODE *PROFILE MATCHING***

Oleh

**Rizky Arif Firmansyah  
NIM 102410101025**

Pembimbing

Pembimbing Utama : Prof. Drs. Slamir, M.CompSc., Ph.D  
NIP 19670420 199201 1 001

Pembimbing Pendamping : Muhammad Arief Hidayat, S.Kom, M.Kom.  
NIP 19810123 201012 1 003

**PENGESAHAN**

Skripsi berjudul “**Sistem Pendukung Keputusan Penempatan Guru Sekolah Dasar Negeri di Kabupaten Jember Menggunakan Metode *Profile Matching***” telah diuji dan disahkan pada:

Hari : Senin  
Tanggal : 27 April 2015  
Tempat : Program Studi Sistem Informasi Universitas Jember

Tim Penguji

Penguji I

Penguji II

Dr. Saiful Bukhori, ST., M.Kom

Windi Eka Yulia Retnani, S.Kom., MT.

NIP 196811131994121001

NIP 198403052010122002

Mengesahkan  
Ketua Program Studi,

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

**PENGESAHAN PEMBIMBING**

Skripsi berjudul “**Sistem Pendukung Keputusan Penempatan Guru Sekolah Dasar Negeri di Kabupaten Jember Menggunakan Metode *Profile Matching***” telah diuji dan disahkan pada:

Hari, tanggal : Senin, 27 April 2015

Tempat : Program Studi Sistem Informasi Universitas Jember

Disetujui oleh:

Pembimbing Utama,

Pembimbing Pendamping,

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

M. Arief Hidayat S.Kom., M.Kom.  
NIP 198201012010121003

## RINGKASAN

**Sistem Pendukung Keputusan Penempatan Guru Sekolah Dasar Negeri di Kabupaten Jember Menggunakan Metode *Profile Matching***; Rizky Arif Firmansyah, 102410101025; 2015; 210; Program Studi Sistem Informasi.

Badan Kepegawaian Daerah Kabupaten Jember merupakan badan yang memiliki otoritas tertinggi dalam proses pengelolaan penerimaan pegawai, mutasi dan penempatan pegawai baru. Proses penempatan pegawai khususnya guru sekolah dasar selama ini masih menggunakan sistem acak dan terjadi kurangnya pemerataan dalam setiap sekolah dasar negeri yang ada. Untuk itu diperlukan sebuah sistem yang dapat membantu tugas dan fungsi Badan Kepegawaian Daerah (BKD) dalam pengambilan kebijakan untuk menempatkan guru SD sehingga proses penempatan dapat disesuaikan dengan kebutuhan yang ada dan merata disetiap sekolah yang ada di Jember.

Tujuan dari penelitian ini adalah membangun sebuah sistem pendukung keputusan yang mampu memberikan rekomendasi terbaik berdasarkan perhitungan pada metode *profile matching*. Rekomendasi penempatan guru disesuaikan dengan hasil perhitungan metode berdasarkan kriteria yang ditentukan oleh Badan Kepegawaian Daerah (BKD), target kepala sekolah dan pilihan guru tersebut.

Desain dan perancangan yang digunakan dalam penelitian ini adalah model *incremental*. *Incremental* merupakan model perancangan yang dilakukan dengan menganalisa sistem perbagian modul. Model ini sangat fleksibel karena proses analisa dan pengolahan alur sistem dapat dilakukan permodul. Peneliti dapat melanjutkan proses selanjutnya tanpa harus menunggu program lainnya selesai. Analisis kebutuhan sistem berdasarkan *increment*, yaitu manajemen data pengguna, manajemen data sekolah, implementasi metode dan manajemen data guru.

Sistem Pendukung Keputusan Penempatan Guru Sekolah Dasar di Kabupaten Jember menerapkan metode *profile matching* karena metode ini dapat diketahui total nilai *gap*, *core & secondary factor* dan nilai total. Perhitungan pada metode ini memudahkan Badan Kepagawain Daerah dalam mencari ranking guru tertinggi yang akan memberikan rekomendasi terbaik dalam proses penempatan guru. Sistem ini juga dapat menampilkan hasil penempatan pada guru dan memberikan informasi melalui *GoogleMaps*.

Hasil dari penelitian ini adalah sebuah Sistem Pendukung Keputusan Penempatan Guru Sekolah Dasar di Kabupaten Jember (SIPENTARU) dengan harapan dapat memberikan rekomendasi terbaik kepada Badan Kepegawaian Daerah (BKD) dalam proses menentukan penempatan guru. Fitur ini juga dapat memberikan informasi peringkat guru terbaik di setiap sekolah. Hasil perhitungan nilai *gap*, *core* dan *secondary factor* juga ditampilkan oleh sistem.

## PRAKATA

Puji syukur kepada Allah SWT yang telah memberikan rahmat, hidayah dan karunia sehingga penulis dapat menyelesaikan skripsi yang berjudul “Sistem Penunjang Keputusan Penempatan Guru Sekolah Dasar Negeri di Kabupaten Jember menggunakan metode *Profile Matching*”. Skripsi ini disusun guna melengkapi tugas akhir dan sebagai salah satu syarat menyelesaikan pendidikan Strata Satu (S1) pada Program Studi Sistem Informasi Universitas Jember.

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

1. Allah SWT yang telah melimpahkan rahmat dan hidayahNya, serta kesempatan waktu sehingga penulis dapat menyelesaikan sebuah karya tulis ini;
2. Prof Drs. Slamin, M.Comp.Sc., Ph.D selaku Ketua Program Studi Sistem Informasi Universitas Jember;
3. Prof. Drs. Slamin, M.Comp.Sc., Ph.D selaku Dosen Pembimbing Utama, Bapak Muhammad Arief Hidayat, S.Kom.,M.Kom selaku Dosen Pembimbing Pendamping yang telah meluangkan waktu dan bersabar dalam membimbing, arahan dan motivasi dalam proses menyelesaikan skripsi;
4. Seluruh Bapak dan Ibu Dosen beserta staf karyawan Program Studi Sistem Informasi;
5. Badan Kepagawaian Daerah yang telah bersedia memberikan ijin kepada peneliti untuk dijadikan objek dalam penelitian skripsi ini.

Penulis menyadari bahwa laporan ini masih jauh dari sempurna. Demi kesempurnaan skripsi ini, penulis menerima segala masukan baik kritik dan saran dari berbagai pihak yang bersifat membangun. Harapan penulis semoga skripsi ini bermanfaat.

Jember, 20 Maret 2015

Penulis

**DAFTAR ISI**

	Halaman
HALAMAN JUDUL.....	i
PERSEMBAHAN.....	ii
MOTTO .....	iii
PERNYATAAN.....	iv
HALAMAN PEMBIMBINGAN.....	v
HALAMAN PENGESAHAN.....	vi
HALAMAN PENGESAHAN PEMBIMBING.....	vii
RINGKASAN.....	viii
PRAKATA.....	x
DAFTAR ISI.....	xi
DAFTAR GAMBAR.....	xv
DAFTAR TABEL.....	xvii
DAFTAR PERSAMAAN.....	xix
DAFTAR LAMPIRAN.....	xx
BAB 1. PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	3
1.3. Batasan Masalah.....	4
1.4. Tujuan dan Manfaat.....	4
1.4.1. Tujuan.....	4
1.4.2. Manfaat.....	5
1.5. Sistematika Penulisan.....	5
BAB 2. TINJAUAN PUSTAKA.....	6
2.1. Sistem Penunjang Keputusan.....	6
2.2. Guru Sekolah Dasar.....	6
2.3. Penempatan Pegawai.....	7
2.4. Penelitian Terdahulu.....	7

2.5.	Metode Profile Matching.....	8
2.6.	Model <i>Incremental</i> .....	10
2.7.	Kebutuhan Software .....	12
BAB 3.	METODOLOGI PENELITIAN .....	14
3.1.	Pendekatan Penelitian.....	14
3.2.	Tempat Penelitian.....	14
3.3.	Alur Penelitian.....	14
3.3.1.	Studi Literatur .....	15
3.3.2.	Pengumpulan Data .....	16
3.3.3.	Teknik Pengumpulan Data.....	16
3.4.	Metode <i>Profile Matching</i> untuk Menghitung Bobot Guru SD.....	17
3.5.	Metode Pengembangan Sistem .....	17
3.5.1.	Analisis Kebutuhan Sistem .....	18
3.5.2.	Desain.....	19
3.5.3.	<i>Coding</i> .....	19
3.5.4.	Pengujian ( <i>Testing</i> ) .....	19
BAB 4.	ANALISIS DAN PERANCANGAN SISTEM .....	20
4.1.	Pengumpulan Data .....	20
4.2.	Perancangan Sistem.....	26
4.2.1.	<i>Bussiness Process</i> .....	27
4.2.2.	<i>Usecase Diagram</i> .....	28
4.2.3.	Perancangan Modul III Implementasi Metode.....	31
4.2.4.	Perancangan Modul I Manajemen Data Pengguna .....	52
4.2.5.	Perancangan Modul II Manajemen Data Sekolah.....	52
4.2.6.	Perancangan Modul IV Manajemen Data Guru.....	52
4.3.	Pengkodean Sistem.....	52
4.4.	Pengujian Sistem .....	52
4.4.1.	Pengujian <i>White Box</i> .....	52
4.4.2.	Pengujian <i>Black Box</i> .....	53
BAB 5.	HASIL DAN PEMBAHASAN .....	55

5.1.	Hasil Penelitian.....	55
5.2.	Hasil Analisa .....	55
5.2.1.	Atribut Bobot Presentase.....	55
5.2.2.	Faktor dan Kriteria yang Digunakan.....	56
5.2.3.	Skala Faktor .....	56
5.2.4.	Perhitungan Manual .....	61
5.3.	Implementasi Sistem .....	67
5.3.1	<i>Login</i> .....	68
5.3.2	Halaman User Pengguna.....	68
5.3.3	Fitur <i>Input</i> Atribut.....	69
5.3.4	Halaman Utama BKD .....	70
5.3.5	Halaman Utama Kepala Sekolah .....	71
5.4.	Pengujian Sistem .....	71
BAB 6.	KESIMPULAN DAN SARAN .....	71
6.1.	Kesimpulan.....	71
6.2.	Saran .....	71
	DAFTAR PUSTAKA .....	72
	LAMPIRAN A. PERANCANGAN SISTEM.....	74
A.1	Perancangan Modul I Manajemen Data Pengguna .....	74
A.2	Perancangan Modul II Manajemen Data Sekolah.....	83
A.3	Perancangan Modul IV Manajemen Data Guru.....	94
	LAMPIRAN B. IMPLEMENTASI SISTEM .....	119
B.1	Implementasi Modul 1 Manajemen Data Pengguna .....	119
B.3	Implementasi Modul III Implementasi Metode.....	121
B.4	Implementasi Modul IV Manajemen Data Guru.....	123
	LAMPIRAN C. PENGUJIAN WHITE BOX SISTEM.....	125
C.1	Pengujian <i>White Box</i> Modul 1 Manajemen Data Pengguna.....	125
C.2	Pengujian <i>White Box</i> Modul II Manajemen Data Sekolah.....	130
C.3	Pengujian <i>White Box</i> Modul III Implementasi Metode.....	134
C.4	Pengujian <i>White Box</i> Modul IV Manajemen Data Guru .....	146

LAMPIRAN D. PENGUJIAN BLACK BOX SISTEM.....	153
C.1 Pengujian <i>Black Box</i> Modul 1 Manajemen Data Pengguna.....	153
C.2 Pengujian <i>Black Box</i> Modul II Manajemen Data Sekolah .....	156
C.3 Pengujian <i>Black Box</i> Modul III Implementasi Metode .....	159
C.4 Pengujian <i>Black Box</i> Modul IV Manajemen Data Guru .....	165
LAMPIRAN E PENULISAN KODE PROGRAM .....	171



DAFTAR GAMBAR

	Halaman
Gambar 2.1 <i>Incremental Model</i> .....	11
Gambar 3.1 Alur Penelitian Sumber .....	15
Gambar 3.2 <i>Flowchart</i> Penerapan Metode <i>Profile Matching</i> .....	17
Gambar 4.1 <i>Bussines Process</i> Sipentaru .....	27
Gambar 4.2 <i>Use Case Diagram</i> .....	29
Gambar 4.3 <i>Edit</i> Nilai Presentase Atribut .....	38
Gambar 4.4 <i>Input</i> Kriteria .....	39
Gambar 4.5 Hapus Kriteria .....	40
Gambar 4.6 <i>Input</i> Faktor .....	41
Gambar 4.7 Hapus Faktor .....	42
Gambar 4.8 Lihat Hasil Perhitungan .....	43
Gambar 4.9 Penempatan Guru .....	44
Gambar 4.10 Lihat Peringkat Guru per Sekolah .....	45
Gambar 4.11 <i>Sequence Diagram</i> <i>Edit</i> Nilai Presentase Atribut .....	46
Gambar 4.12 <i>Sequence Diagram</i> <i>Input</i> Kriteria .....	46
Gambar 4.13 <i>Sequence Diagram</i> Hapus Kriteria .....	47
Gambar 4.14 <i>Sequence Diagram</i> <i>Input</i> Faktor .....	47
Gambar 4.15 <i>Sequence Diagram</i> Hapus Faktor .....	48
Gambar 4.16 <i>Sequence Diagram</i> Lihat Hasil Perhitungan .....	48
Gambar 4.17 <i>Sequencce Diagram</i> Penempatan Guru .....	49
Gambar 4.18 <i>Sequence Diagram</i> Lihat Peringkat Guru per Sekolah .....	49
Gambar 4.19 <i>Class Diagram</i> Modul Implementasi Metode .....	50
Gambar 4.20 <i>Entity Relationship Diagram</i> Implementasi Metode .....	51
Gambar 5.1 Halaman Presentase Bobot Nilai .....	62

Gambar 5.2 Halaman Hasil Perhitungan GAP.....	64
Gambar 5.3 Halaman Hasil Perhitungan <i>Core</i> dan <i>Secondary Factor</i> .....	66
Gambar 5.4 Halaman Hasil Perhitungan Total dan Perhitungan Akhir.....	67
Gambar 5.5 Tampilan <i>Login</i> .....	68
Gambar 5.6 Tampilan <i>User</i> Pengguna.....	69
Gambar 5.7 Tampilan <i>Input</i> Atribut .....	69
Gambar 5.8 Halaman Utama Badan Kepegawaian Daerah (BKD) .....	70
Gambar 5.9 Halaman Kepala Sekolah .....	71

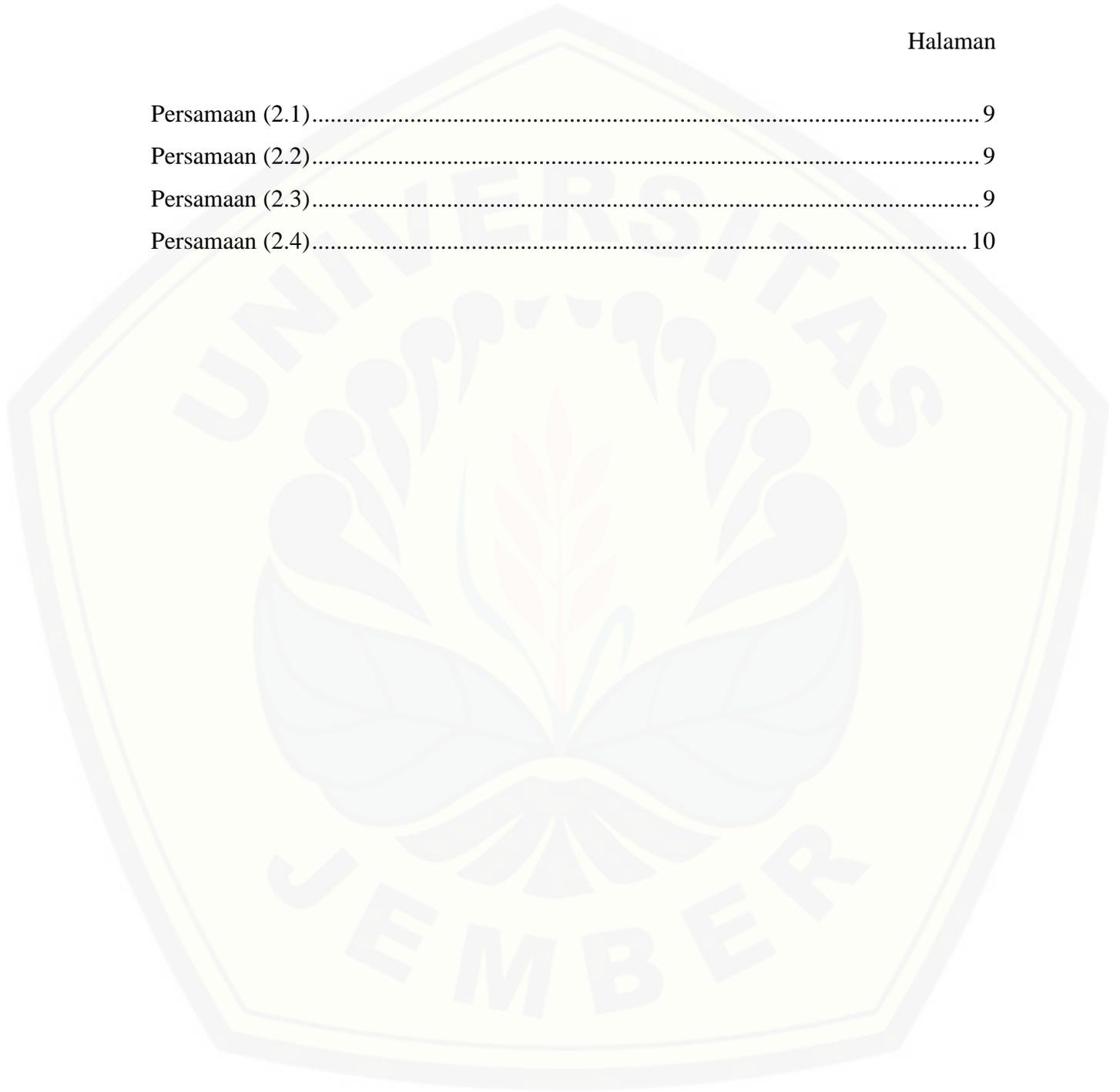
**DAFTAR TABEL**

	Halaman
Tabel 4.1 Tabel Jenis Guru Sekolah Dasar .....	20
Tabel 4.2 Tabel Data Atribut .....	21
Tabel 4.3 Tabel Data Kriteria.....	21
Tabel 4.4 Tabel Data Faktor.....	22
Tabel 4.5 Tabel Jarak .....	22
Tabel 4.6 Tabel Status Wilayah .....	22
Tabel 4.7 Tabel Usia .....	23
Tabel 4.8 Tabel Pendidikan.....	23
Tabel 4.9 Tabel Indeks Prestasi Kumulatif (IPK).....	23
Tabel 4.10 Tabel Status Pernikahan.....	23
Tabel 4.11 Tabel tes Pengetahuan Umum (TPU) .....	24
Tabel 4.12 Tabel Tes Potensi Akademik (TPA) .....	24
Tabel 4.13 Tabel Tes Wawasan Kebangsaan.....	24
Tabel 4.14 Tabel Kemampuan Kosentrasi Bidang .....	24
Tabel 4.15 Tabel Kemampuan Mengajar.....	25
Tabel 4.16 Tabel Kemampuan Berkomunikasi Terhadap Siswa .....	25
Tabel 4.17 Tabel Kebutuhan Fungsional .....	25
Tabel 4.18 Tabel Kebutuhan Non Fungsional .....	26
Tabel 4.19 Tabel Deskripsi Aktor.....	29
Tabel 4.20 Tabel Penjelasan <i>Use Case</i> .....	30
Tabel 4.21 Mengubah Nilai Presentase Atribut .....	32
Tabel 4.22 <i>Input</i> Kriteria.....	33
Tabel 4.23 Hapus Kriteria.....	33
Tabel 4.24 <i>Input</i> Faktor.....	34

Tabel 4.25 Hapus Faktor .....	35
Tabel 4.26 Lihat Hasil Perhitungan .....	36
Tabel 4.27 Penempatan Guru .....	36
Tabel 4.28 Lihat Peringkat Guru Per Sekolah .....	37
Tabel 5.1 Tabel Atribut Bobot Presentase .....	56
Tabel 5.2 Tabel Faktor dan Kriteria yang digunakan .....	56
Tabel 5.3 Tabel Jarak .....	57
Tabel 5.4 Tabel Status Wilayah .....	57
Tabel 5.5 Tabel Usia .....	57
Tabel 5.6 Tabel Pendidikan.....	58
Tabel 5.7 Tabel Indeks Prestasi Kumulatif (IPK).....	58
Tabel 5.8 Tabel Status Pernikahan.....	58
Tabel 5.9 Tabel Tes Pengetahuan Umum (TPU).....	59
Tabel 5.10 Tabel Tes Potensi Akademik (TPA).....	59
Tabel 5.11 Tabel Tes Wawasan Kebangsaan.....	60
Tabel 5.12 Tabel Kemampuan Kosentrasi Bidang .....	60
Tabel 5.13 Tabel Kemampuan Mengajar.....	60
Tabel 5.14 Tabel Kemampuan Berkomunikasi Terhadap Siswa .....	61
Tabel 5.15 Tabel Presentase Bobot Nilai Atribut <i>Core Factor</i> dan <i>Secondary Factor</i> .....	61
Tabel 5.16 Tabel Presentase Kriteria .....	62
Tabel 5.17 Tabel Atribut dan Faktor.....	62
Tabel 5.18 Tabel Bobot Nilai GAP.....	63
Tabel 5.19 Tabel Perhitungan GAP .....	63

**DAFTAR PERSAMAAN**

	Halaman
Persamaan (2.1).....	9
Persamaan (2.2).....	9
Persamaan (2.3).....	9
Persamaan (2.4).....	10



**DAFTAR LAMPIRAN**

	Halaman
LAMPIRAN A. PERANCANGAN SISTEM.....	74
A.1 Perancangan Modul I Manajemen Data Pengguna .....	74
A.2 Perancangan Modul II Manajemen Data Sekolah.....	83
A.3 Perancangan Modul IV Manajemen Data Guru .....	94
LAMPIRAN B. IMPLEMENTASI SISTEM .....	119
B.1 Implementasi Modul 1 Manajemen Data Pengguna .....	119
B.3 Implementasi Modul III Implementasi Metode.....	121
B.4 Implementasi Modul IV Manajemen Data Guru.....	123
LAMPIRAN C. PENGUJIAN WHITE BOX SISTEM.....	125
C.1 Pengujian <i>White Box</i> Modul 1 Manajemen Data Pengguna.....	125
C.2 Pengujian <i>White Box</i> Modul II Manajemen Data Sekolah.....	130
C.3 Pengujian <i>White Box</i> Modul III Implementasi Metode.....	134
C.4 Pengujian <i>White Box</i> Modul IV Manajemen Data Guru .....	146
LAMPIRAN D. PENGUJIAN BLACK BOX SISTEM.....	153
C.1 Pengujian <i>Black Box</i> Modul 1 Manajemen Data Pengguna.....	153
C.2 Pengujian <i>Black Box</i> Modul II Manajemen Data Sekolah .....	156
C.3 Pengujian <i>Black Box</i> Modul III Implementasi Metode .....	159
C.4 Pengujian <i>Black Box</i> Modul IV Manajemen Data Guru .....	165
LAMPIRAN E PENULISAN KODE PROGRAM .....	171

## **BAB 1. PENDAHULUAN**

Bab ini menjelaskan tentang latar belakang, rumusan masalah, batasan masalah, tujuan, dan sistematika penulisan.

### **1.1. Latar Belakang**

Pendidikan adalah hak yang harus dimiliki oleh setiap orang. Pendidikan yang baik akan menghasilkan sumber daya manusia yang baik dan berkompeten. Oleh karena itu pendidikan sangat penting untuk mencapai kesejahteraan rakyat. Dalam pembukaan Undang-Undang Dasar 1945 alenia 4 salah satu tujuan pembentukan Negara Indonesia adalah mencerdaskan kehidupan bangsa. Agar pendidikan di Indonesia lebih maju maka dibutuhkan tenaga pendidik yang berkompeten dan merata.

Kementerian Pendidikan Nasional (Kemendiknas) memprediksi dalam kurun waktu 2010 sampai dengan 2014 ada 171.109 guru pegawai negeri sipil di SD dan SMP pensiun. Jika ditarik lagi hingga 2020, tercatat ada 400 ribu lebih guru yang pensiun. Dunia pendidikan dihantui ancaman kekurangan guru PNS. Prediksi tingginya angka pensiun tersebut disebabkan karena pada 1974 silam muncul pengangkatan dalam jumlah besar guru yang dilakukan oleh pemerintah yang jumlahnya mencapai ratusan ribu orang. Kejadian itu muncul karena tahun itu keluar Instruksi Presiden (Inpres) Nomor 1 yang memudahkan pengangkatan guru, terutama guru sekolah dasar.

Menurut UU Nomor 43 Tahun 1999 yang merupakan perubahan atas UU Nomor 8 Tahun 1974 tentang pokok kepegawaian, dijelaskan bahwa kedudukan dan peran pegawai negeri sipil adalah sangat penting dan menentukan, karena mereka adalah sebagai aparatur negara dan aparatur pemerintah dalam penyelenggaraan pemerintah dan pembangunan dalam usaha mewujudkan suatu tujuan nasional. Untuk itu, perlu adanya suatu proses yang sistematis dan berkelanjutan untuk memperoleh kuantitas, kualitas, komposisi, dan distribusi pegawai yang tepat sesuai dengan

kebutuhan organisasi. Sehingga dapat mewujudkan visi dan misi organisasi menjadi kinerja yang nyata.

Rekrutmen calon pegawai negeri sipil (CPNS) yang menjadi harapan sebagian besar masyarakat untuk dapat menjadi abdi negara kurang disesuaikan dengan kuota daya tampung yang ada. Terlalu banyaknya pendaftar yang berkeinginan kuat untuk dapat lulus tes dan menyanggah status pegawai negeri sipil tidak sebanding dengan besarnya daya tampung yang ada.

Menurut Peraturan Pemerintah Republik Indonesia Nomor 74 Tahun 2008 Bab IV Tentang Pengangkatan, Penempatan, dan Pemindahan Pasal 61 ayat I, menyatakan bahwa guru yang diangkat oleh pemerintah atau pemerintah daerah dapat ditempatkan pada jabatan struktural sesuai dengan ketentuan peraturan perundang-undangan. Kurang tepatnya kebijakan dalam proses pengangkatan dan penempatan pegawai yang tidak disesuaikan dengan kebutuhan yang ada akan berefek pada kurangnya efisiensi dan efektivitas dalam pengelolaan pegawai. Oleh sebab itu, diperlukan tanggung jawab dan ketegasan pengambil kebijakan untuk dapat mempertimbangkan dengan baik dalam hal pengangkatan dan penempatan pegawai agar proses pengelolaan pegawai tepat dan disesuaikan dengan kebutuhan yang ada.

Kesesuaian antara adanya jabatan yang dibutuhkan dengan latar belakang sumber daya manusia pegawai merupakan hal yang sangat penting, karena akan berpengaruh pada kinerja pegawai pada saat bekerja. Kesesuaian bidang jabatan antara kemampuan yang dimiliki oleh pegawai dan kedudukan yang dipegang akan menimbulkan motivasi tersendiri, karena akan memudahkan pegawai dalam proses penyesuaian kerja sehingga akan memicu kreatifitas pegawai dalam menyelesaikan tanggung jawabnya dalam bekerja.

Menurut Hasibuan (2005), penempatan harus didasarkan pada *job description* dan *job specification* yang telah ditentukan serta berpedoman kepada prinsip “*The right man on the right place and the right man behind the job*”. Hal ini akan membawa suatu instansi kepada hasil kerja yang optimal karena terdapat adanya korelasi positif antara penempatan pegawai dengan peningkatan produktifitas kerja.

Perlu adanya keseimbangan antara kesesuaian jabatan dengan rancangan program yang telah dibuat dalam daftar rincian kerja. Masalah lain yang dihadapi adalah kurang efektifnya penempatan lokasi mengajar guru yang seringkali sangat jauh dengan lokasi tempat tinggal. Hal tersebut tentunya akan menambah persoalan yang ada karena selain tidak efektif juga akan berpengaruh pada tingginya integritas dan loyalitas guru

.Permasalahan yang terjadi di Kabupaten Jember adalah masih banyaknya permasalahan kekurangan tenaga kerja guru yang kurang bisa mengakomodasi kebutuhan yang ada. Hal tersebut berimbas kepada kurangnya pemerataan tenaga kerja guru. Mayoritas guru masih berpusat pada Jember wilayah kota saja belum menjangkau ke dalam wilayah pedesaan dan kawasan terpencil. Untuk dapat menyelesaikannya maka dibutuhkan penempatan guru sekolah dasar yang tepat agar masalah ini dapat teratasi dengan baik.

Berdasarkan latar belakang yang telah dipaparkan, maka diperlukan sebuah sistem yang dapat membantu Pemerintah Dinas Kabupaten Jember dalam menyelesaikan masalah penempatan PNS. Sistem yang akan dibangun dalam penelitian ini menggunakan metode *Profile Matching* yang termasuk ke dalam salah satu metode pengambilan keputusan.

## **1.2. Rumusan Masalah**

Berdasarkan uraian latar belakang yang telah dipaparkan, ditemukan beberapa masalah sebagai berikut:

1. Bagaimana menerapkan metode *profile matching* dalam menentukan penempatan guru sekolah dasar di Kabupaten Jember?
2. Bagaimana merancang dan membangun sebuah sistem informasi pendukung keputusan penempatan guru sekolah dasar di Kabupaten Jember berbasis *web* menggunakan metode *profile matching*?

### 1.3. Batasan Masalah

Batasan masalah dalam penelitian ini adalah:

1. Sistem ini akan digunakan untuk menentukan penempatan lokasi guru sekolah dasar negeri di Kabupaten Jember.
2. Data sekolah dasar negeri yang digunakan dalam penelitian ini hanya berasal dari tiga kecamatan, yaitu Sumpalsari, Patrang, dan Kaliwates.
3. Data sekolah yang digunakan dalam penelitian ini bukan merupakan data sekolah swasta atau sekolah dengan basis agama.
4. Metode yang digunakan dalam sistem ini adalah metode *Profile Matching*.
5. Aspek-aspek atau kriteria yang digunakan dalam penelitian ini diperoleh dari Badan Kepegawain Daerah Kabupaten Jember dan studi literatur mengenai penempatan pegawai negeri sipil khususnya guru sekolah dasar negeri.

### 1.4. Tujuan dan Manfaat

Tujuan dan manfaat berisi tentang tujuan dari penelitian dan pembuatan sistem informasi penempatan sekolah dasar dan merupakan jawaban dari rumusan masalah yang sudah disebutkan.

#### 1.4.1. Tujuan

Tujuan dari pembuatan Sistem Informasi Pendukung Keputusan Penempatan Guru Sekolah Dasar di Kabupaten Jember menggunakan metode *profile matching* adalah:

1. Menerapkan metode *profile matching* dalam menentukan penempatan guru sekolah dasar di Kabupaten Jember.
2. Merancang dan membangun sebuah sistem pendukung keputusan penempatan guru sekolah dasar di Kabupaten Jember berbasis *web* menggunakan metode *profile matching*.

#### 1.4.2. Manfaat

Manfaat yang diperoleh dari penelitian dan pembangunan sistem penempatan Guru SD ini adalah:

##### 1. Bagi Peneliti

Memberikan pengalaman dalam mengimplementasikan teori yang diperoleh di bangku kuliah dan meningkatkan keilmuan tentang sistem informasi untuk menyelesaikan tugas akhir jenjang S1 pada Program Studi Sistem Informasi.

##### 2. Bagi Objek Penelitian

Diharapkan dapat membantu Badan Kepegawaian Daerah Kabupaten Jember dalam menempatkan guru sekolah dasar sesuai dengan kriteria.

##### 3. Bagi Peneliti lain

Hasil penelitian ini dapat menambah referensi bagi peneliti selanjutnya.

#### 1.5. Sistematika Penulisan

Sistematika dan keruntutan tugas akhir ini disusun sebagai berikut:

##### 1. Pendahuluan

Bab ini menguraikan latar belakang, rumusan masalah, batasan masalah, tujuan, ruang lingkup, dan sistematika penulisan.

##### 2. Tinjauan Pustaka

Bab ini menguraikan materi dan informasi mengenai istilah penting dan penelitian terdahulu yang digunakan dalam penulisan.

##### 3. Metode Penelitian

Bab ini menguraikan jenis penelitian, tempat penelitian, objek penelitian, jenis dan sumber data, teknik pengumpulan data, teknik analisis, dan perancangan.

##### 4. Analisis dan Perancangan Sistem

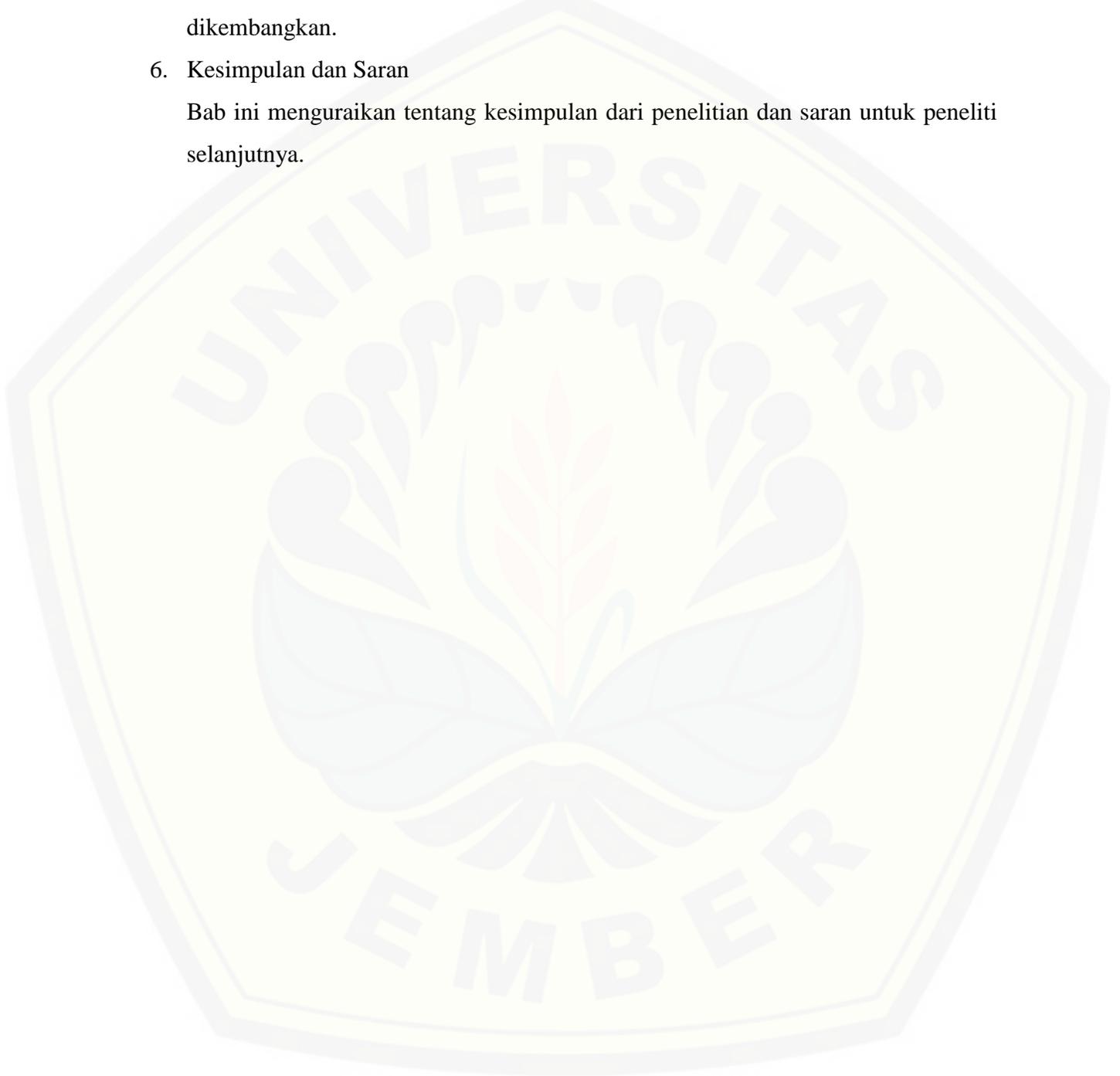
Bab ini menguraikan mengenai analisis dan perancangan sistem yang akan dikembangkan.

5. Hasil dan Pembahasan

Bab ini menjelaskan tentang hasil dan pembahasan dari sistem yang sudah dikembangkan.

6. Kesimpulan dan Saran

Bab ini menguraikan tentang kesimpulan dari penelitian dan saran untuk peneliti selanjutnya.



## BAB 2. TINJAUAN PUSTAKA

Bab ini menjelaskan tinjauan pustaka yang berkaitan dengan dasar teori yang mendukung permasalahan, baik tinjauan pustaka yang terdahulu maupun yang akan diteliti. Berdasarkan pengamatan yang telah dilakukan, penelitian ini membutuhkan dasar teori mengenai sistem, penempatan guru sekolah dasar, sistem penunjang keputusan, metode *profile matching* dan pengembangan sistem

### 2.1. Sistem Penunjang Keputusan

Sistem Penunjang Keputusan (SPK) adalah sistem berbasis komputer yang interaktif, yang membantu pengambil keputusan memanfaatkan data dan model untuk menyelesaikan masalah masalah yang tidak terstruktur (Subakti, 2002).

Menurut Turban (2001), SPK dapat memberikan berbagai manfaat dan keuntungan. Manfaat yang dapat diambil dari SPK adalah:

1. SPK memperluas kemampuan pengambil keputusan dalam memproses data atau informasi bagi pemakainya.
2. SPK membantu pengambil keputusan untuk memecahkan masalah terutama berbagai masalah yang sangat kompleks dan tidak terstruktur.
3. SPK dapat menghasilkan solusi dengan lebih cepat serta hasilnya dapat diandalkan.
4. Walaupun suatu SPK mungkin saja tidak mampu memecahkan masalah yang dihadapi oleh pengambil keputusan, namun dapat menjadi stimulan bagi pengambil keputusan dalam memahami persoalannya, karena mampu menyajikan berbagai alternatif pemecahan.

### 2.2. Guru Sekolah Dasar

Menurut UU nomor 14 Tahun 2005 tentang Guru dan Dosen, Guru adalah pendidik professional dengan tugas utama mendidik, mengajar, membimbing,

mengarahkan, melatih, menilai dan mengevaluasi peserta didik pada pendidikan anak usia dini melalui jalur formal pendidikan dasar dan pendidikan menengah.

Dalam proses pengajaran di tingkat sekolah dasar setiap guru memiliki amanah dan tanggung jawab dalam memberikan pengetahuan khusus dalam cara bersikap sebagai siswa sekolah dasar yang santun dan terpelajar sejak usia dini. Guru yang bertindak sebagai pendidik memiliki tanggung jawab juga dalam menanamkan nilai-nilai moral yaitu berupa budi pekerti dan dasar ilmu pengetahuan yang bermanfaat bagi muridnya.

### **2.3. Penempatan Pegawai**

Penempatan adalah menempatkan posisi seseorang ke posisi pekerjaan yang tepat, seberapa baik seorang karyawan cocok dengan pekerjaannya akan mempengaruhi jumlah dan kualitas pekerjaan (Mathis & Jackson, 2006). Menurut Sastrohadiwiryo (2002), faktor-faktor yang perlu dipertimbangkan dalam penempatan pegawai adalah:

1. Faktor latar belakang pendidikan,
2. Faktor kesehatan jasmani dan rohani,
3. Faktor pengalaman kerja,
4. Faktor umum sumber daya manusia,
5. Faktor jenis kelamin,
6. Faktor status perkawinan,
7. Faktor minat dan hobi.

### **2.4. Penelitian Terdahulu**

Penelitian sebelumnya yaitu Sistem Pendukung Keputusan Penempatan Bidan PTT (Pegawai Tidak Tetap) menggunakan metode *profile matching* yang dilakukan oleh (Iqbal, 2011). Pada penelitian tersebut menggunakan 3 kriteria yang masing-masing kriteria terdapat beberapa subkriteria:

### 1. Jarak Wilayah

Kriteria jarak digunakan untuk mengetahui jangkauan jarak wilayah yang ditempuh dari tempat tinggal desa bidan ke lokasi penempatan bidan, kriteria jarak terdiri atas dua subkriteria, yaitu jarak tempat tinggal bidan dengan desa penempatan dan status wilayah tempat tinggal bidan terhadap desa penempatan.

### 2. Uji Kompetensi dengan sub kriteria:

- a. Tes Kompetensi Standar
- b. Tes Bakat Skolastik
- c. Tes Skala Kematangan

### 3. Evaluasi Diri dengan subkriteria.

- a. Usia
- b. Pendidikan
- c. IPK

## 2.5. Metode Profile Matching

*Profile matching* merupakan suatu proses dalam manajemen sumber daya manusia dimana terlebih dahulu ditentukan kompetensi (kemampuan) yang diperlukan oleh suatu jabatan. Kompetensi atau kemampuan tersebut haruslah dapat dipenuhi oleh pemegang jabatan. Dalam proses *profile matching* secara garis besar merupakan proses membandingkan antara kompetensi individu ke dalam kompetensi jabatan sehingga dapat diketahui perbedaan kompetensinya (*gap*), semakin kecil *gap* yang dihasilkan maka bobot nilainya semakin besar yang berarti memiliki peluang lebih besar untuk karyawan menempati posisi tersebut (Rachma, 2003).

Metode *profile matching* atau pencocokan profil adalah metode yang sering digunakan sebagai mekanisme dalam pengambilan keputusan dengan mengasumsikan bahwa terdapat tingkat *variable predictor* yang ideal yang harus dipenuhi oleh subyek yang diteliti, bukannya tingkat minimal yang harus dipenuhi atau dilewati (Kusrini, 2007). Proses dalam metode *profile matching* adalah sebagai berikut:

1. Menentukan aspek dan sub aspek yang akan diukur atau dinilai,
2. Menentukan standard minimal masing masing aspek (*value target*),
3. Menggolongkan aspek ke dalam faktor utama dan faktor pendukung,
4. Mencari nilai *gap* dan menentukan bobot berdasarkan *gap*.

*Gap* merupakan selisih value masing masing aspek atau atribut yang akan digunakan. Adapun rumus perhitungan gap sebagai berikut:

$$\mathbf{GAP = Value Atribut - Value Target \dots\dots\dots (2.1)}$$

Keterangan:

Value Atribut = Hasil bobot yang dimasukkan oleh pakar

Value Target = Hasil atau bobot yang harus dicapai

5. Menghitung NSF dan NSC

$$\mathbf{NCF = \frac{\sum NC}{\sum IC} \dots\dots\dots (2.2)}$$

Keterangan:

NCF = Rata rata *Core Factor*

NC = Jumlah total *Core Factor*

IC = Jumlah item *Core Factor*

$$\mathbf{NSF = \frac{\sum NS}{\sum IS} \dots\dots\dots (2.3)}$$

Keterangan:

NCF = Rata rata *Secondary Factor*

NC = Jumlah total *Secondary factor*

IC = Jumlah item *Secondary Factor*

6. Menjumlahkan nilai total kriteria dengan rumus:

$$(x)\%NCF(j,e,p,u) + (x)\%NSF(j,e,p,u) = N(j,e,p,u) \dots\dots\dots (2.4)$$

Keterangan:

(x)% = Nilai bobot kriteria yang telah ditentukan

NCF(j, e, p, u) = Nilai rata-rata *core factor*

NSF(j, e, p, u) = Nilai rata-rata *secondary factor*

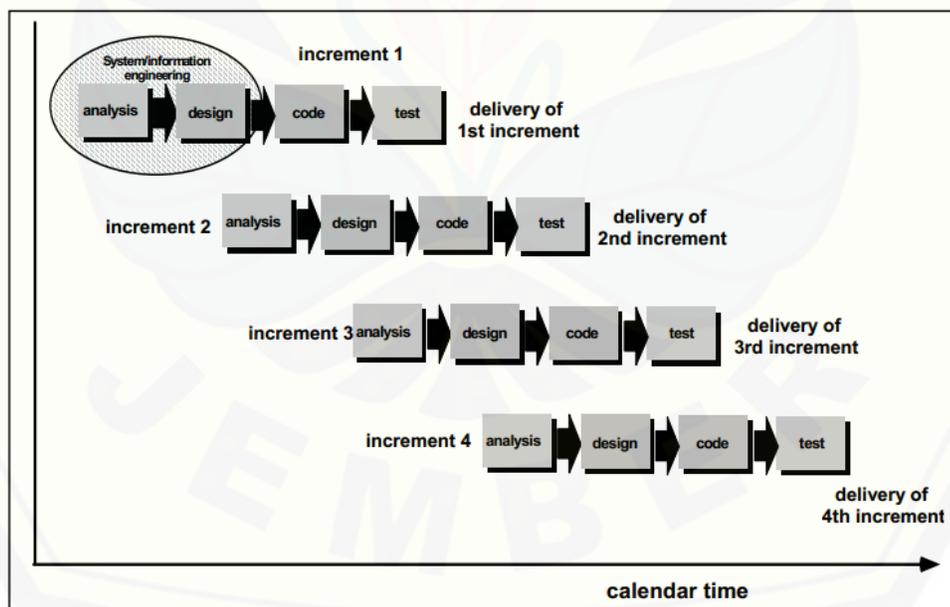
## 2.6. Model Incremental

Model *incremental* merupakan penyempurnaan dari model *waterfall*. Keuntungan dari model ini adalah ketika terjadi penambahan kebutuhan fungsional akan lebih mudah diuji, diverifikasi, dan divalidasi. Dalam *incremental* pembangunan sistem dilakukan bertahap sehingga *user* dapat menggunakan terlebih dahulu bagian yang telah selesai dibangun tanpa harus menunggu sistem selesai dibangun sepenuhnya. Selain itu model *incremental* akan mengurangi beban pekerjajanya dikarenakan setiap modul akan dilakukan pengujian, verifikasi, serta validasi sehingga perubahan yang akan terjadi pada sistem tidak terlalu besar (Proboyekti, 2008). Model *incremental* memiliki beberapa kelebihan dan kekurangan (Mega, 2011) yaitu:

1. Merupakan model dengan manajemen yang sederhana
2. Pelanggan tidak perlu menunggu sampai seluruh sistem dikirim untuk mengambil keuntungan dari sistem tersebut. Inkremen yang pertama sudah memenuhi persyaratan mereka yang paling kritis, sehingga penerkat lunak dapat segera digunakan.
3. Pelanggan dapat memakai inkremen yang pertama sebagai bentuk prototype dan mendapatkan pengalaman yang dapat menginformasikan persyaratan untuk inkremen sistem berikutnya.

4. Resiko kegagalan proyek secara keseluruhan lebih rendah. Walaupun masalah dapat ditemukan pada beberapa inkremen, biasanya beberapa inkremen diserahkan dengan sukses kepada pelanggan.
5. Karena layanan dengan prioritas tertinggi diserahkan pertama dan inkremen berikutnya diintegrasikan dengannya, sngatlah penting bahwa layanan sistem yang paling penting mengalami pengujian yang paling ketat. Ini berarti bahwa pelanggan akan memiliki kemungkinan kecil untuk memenuhi kegagalan perangkat lunak pada inkremen sistem yang paling kecil.

Kekurangan penggunaan *incremental model* adalah *increment* harus relatif lebih kecil (tidak lebih dari 20.000 baris kode) dan setiap inkremen harus menyediakan sebagian dari fungsional sistem. Adanya kesulitan untuk memetakan persyaratan pelanggan pada inkremen dengan ukuran yang benar dan penjelasan model *incremental* terdapat pada Gambar 2.1 Pembangunan sistem dilakukan secara bertahap dan bergantian sesuai dengan Gambar 2.1



Gambar 2.1 *Incremental Model*  
Sumber: (Hasil Analisis, 2015)

## 2.7. Kebutuhan Software

Kebutuhan *software* dalam membuat sistem pendukung keputusan ini terdiri dari:

### 1. Bahasa pemrograman PHP

*Page Hypertext Preprocessor* adalah salah satu bahasa pemrograman yang berjalan dalam sebuah web *server*. *Script-script* PHP yang dibuat harus tersimpan dalam sebuah *server* dan dieksekusi atau diproses dalam *server* tersebut. Pengguna program PHP memungkinkan sebuah *website* atau *computer client* akan diolah dan disimpan dalam *database web server* dan bisa ditampilkan kembali apabila diakses (Andi, 2008 ).

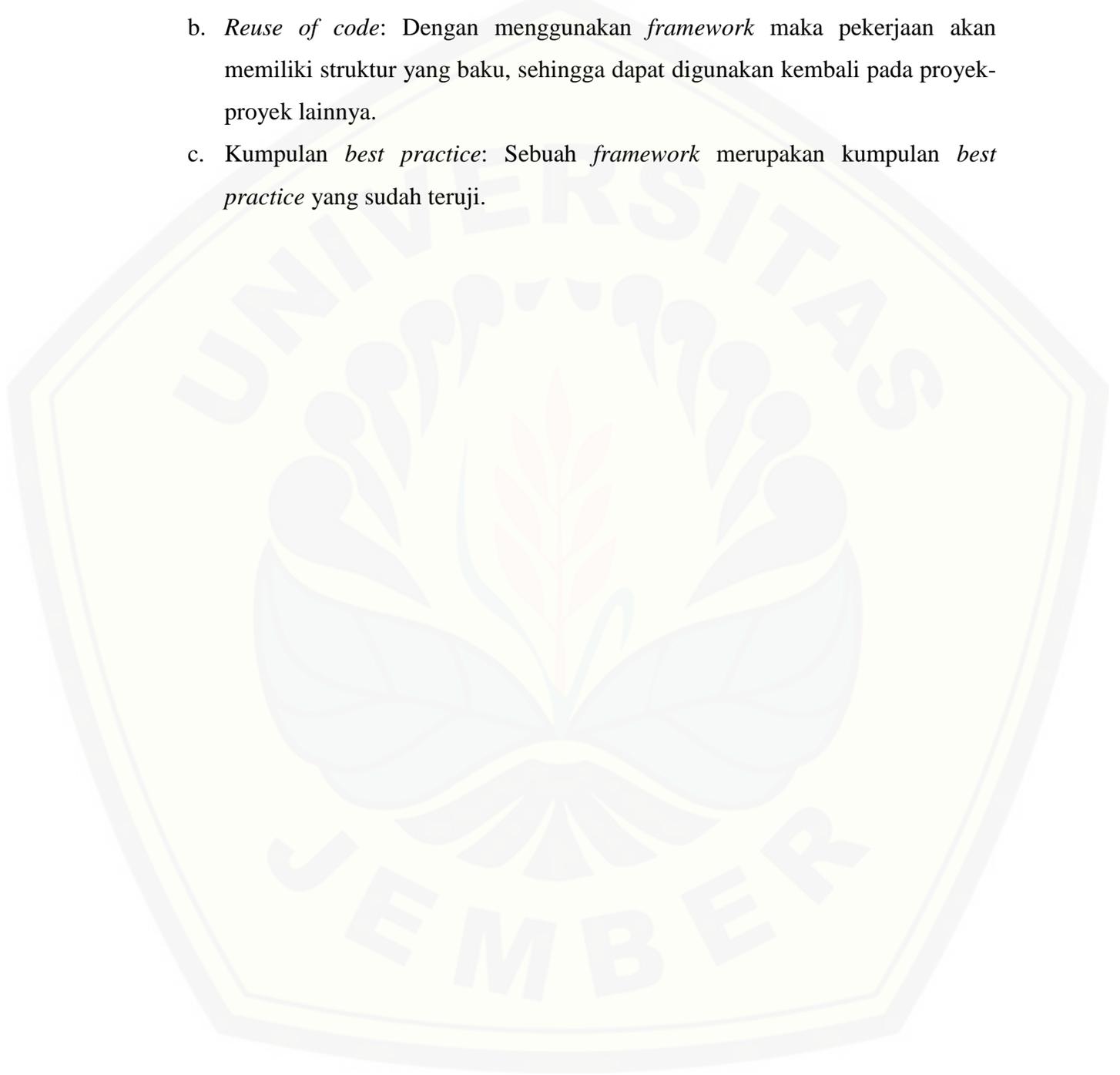
### 2. *Server database* MySQL

MySQL merupakan perangkat yang berperan sebagai *server database*, yang selanjutnya akan digunakan untuk mempelajari kode-kode PHP yang berkaitan atau membutuhkan akses ke *server database*. MySQL ini banyak digunakan untuk pengembangan aplikasi *web* dikarenakan gratis, pengelolaan datanya sederhana, memiliki tingkat keamanan yang bagus, mudah diperoleh, dan lain-lain. (Enjang, 2012).

### 3. *Framework CodeIgniter*

*Framework* adalah sebuah struktur konseptual dasar yang digunakan untuk memecahkan sebuah permasalahan, bahkan isu-isu kompleks yang ada. *Framework* berisi sekumpulan arsitektur atau konsep-konsep yang dapat mempermudah dalam pemecahan sebuah masalah terutama dalam membangun sebuah aplikasi (Ibnu, 2011). Membangun aplikasi menggunakan *framework* tidak perlu membuat program dari awal, tetapi telah disediakan *library* fungsi-fungsi yang sudah diorganisasikan untuk menyelesaikan suatu program. Pembangun dapat memfokuskan kerja pada penyelesaian masalah, tidak banyak lagi bekerja untuk membuat *library* fungsi dan prosedur. Keuntungan menggunakan *framework* menurut (Ibnu, 2011) adalah:

- a. Menghemat waktu pengembangan: Dengan struktur dan *library* yang telah disediakan oleh *framework* maka tidak perlu lagi memikirkan hal-hal tersebut.
- b. *Reuse of code*: Dengan menggunakan *framework* maka pekerjaan akan memiliki struktur yang baku, sehingga dapat digunakan kembali pada proyek-proyek lainnya.
- c. Kumpulan *best practice*: Sebuah *framework* merupakan kumpulan *best practice* yang sudah teruji.



## BAB 3. METODOLOGI PENELITIAN

Bab ini menjelaskan mengenai metode-metode yang digunakan selama melakukan penelitian mulai dari pendekatan penelitian, tempat dan waktu penelitian, alur penelitian, studi literatur, sumber data penelitian, dan metode yang digunakan dalam penelitian untuk menyelesaikan tujuan yang telah dijelaskan.

### 3.1. Pendekatan Penelitian

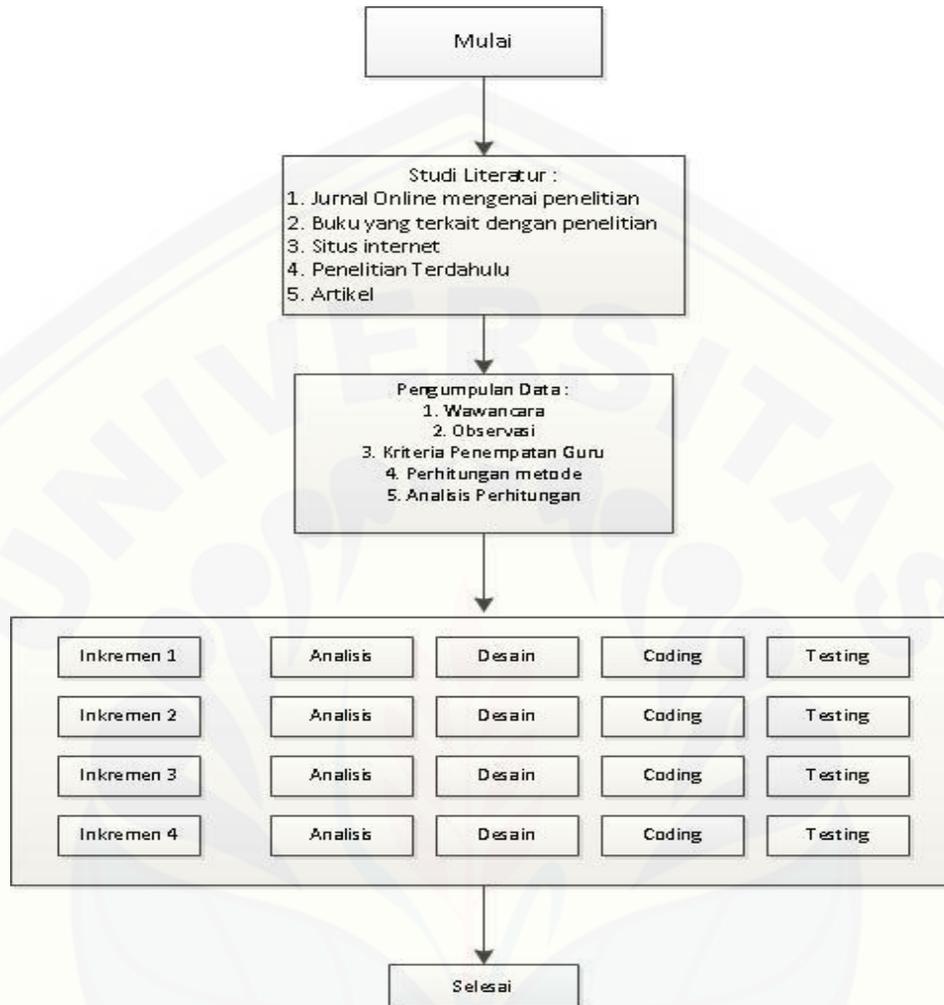
Tujuan dari penelitian ini adalah mengkaji secara mendalam mengenai metode *profile matching* dalam membantu penempatan guru SD di Jember. Pendekatan yang dilakukan peneliti menggunakan jenis penelitian kuantitatif karena jenis data yang digunakan dalam peneliti menggunakan angka. Selain itu, penelitian ini juga menggunakan penelitian kualitatif karena penelitian ini menganalisis studi literatur yang berhubungan dengan indikator untuk penempatan guru SD di Jember.

### 3.2. Tempat Penelitian

Penelitian ini dilakukan di Dinas Pendidikan Kabupaten Jember Jl. Dr. Subandi No. 29 dan di Badan Kepegawaian Daerah Jember Jl. Sudarman No.1 Jember.

### 3.3. Alur Penelitian

Alur penelitian dalam penelitian ini menggunakan metode pengumpulan data, metode yang diimplementasikan dalam sistem yaitu metode *profile matching* dan metode pengembangan sistem. Metode tersebut digunakan untuk menjawab rumusan masalah yang ada pada objek penelitian. Alur Penelitian pada penelitian ini akan dijelaskan pada Gambar 3.1.



Gambar 3.1 Alur Penelitian  
Sumber: (Hasil Analisis, 2015)

### 3.3.1. Studi Literatur

Tahap pertama dalam penelitian ini adalah studi literatur digunakan untuk mendapatkan gambaran tentang apa yang telah dipelajari dalam penelitian ini untuk menunjang pemahaman dan pengetahuan penulis tentang teori, konsep, ilmu dan metode yang digunakan dalam penelitian ini. Dalam studi literatur ini, peneliti menggunakan *e-journal*, penelitian terdahulu, *e-book* tentang metode *profile matching* dan hal hal yang terkait dengan penelitian ini.

### 3.3.2. Pengumpulan Data

Pengumpulan data digunakan untuk memperoleh data yang dibutuhkan untuk mencapai tujuan penelitian. Data yang digunakan berdasarkan sumber data dalam penelitian ini adalah:

#### 1. Data Primer

Data primer merupakan data yang diperoleh melalui observasi dan wawancara secara langsung pada pihak Badan Kepegawaian Daerah Kabupaten Jember tentang penempatan guru sekolah dasar dan dinas pendidikan bagian tenaga pendidik.

#### 2. Data Sekunder

Data sekunder adalah data yang diperoleh dari studi literatur seperti buku, jurnal, internet dan peraturan penempatan guru SD di Kabupaten Jember.

### 3.3.3. Teknik Pengumpulan Data

Pengumpulan data dalam penelitian ini digunakan untuk mendapatkan data yang dibutuhkan peneliti dalam membangun sistem penempatan guru SD. Proses untuk mendapatkan data tersebut adalah:

#### 1. Observasi

Observasi dilakukan oleh peneliti dengan mendatangi langsung Dinas Pendidikan bagian tenaga pendidik dan Badan Kepegawaian Daerah Kabupaten Jember yang bertujuan untuk mengetahui alur penempatan guru khususnya guru SD di Kabupaten Jember.

#### 2. Wawancara

Peneliti melakukan wawancara langsung terhadap pegawai Dinas Pendidikan dan pegawai Badan Kepegawaian Daerah Kabupaten Jember terkait dengan penempatan guru SD. Data yang dikumpulkan melalui wawancara meliputi alur penempatan guru SD, data kriteria penempatan guru SD dan data guru SD.

### 3.4. Metode *Profile Matching* untuk Menghitung Bobot Guru SD

Metode *profile matching* ini akan penulis gunakan untuk membangun sistem penempatan guru SD di Kabupaten Jember berdasarkan kriteria yang ditetapkan oleh Badan Kepegawaian Negara. *Flowchart* penerapan metode *profile matching* terdapat pada Gambar 3.2.



Gambar 3.2. *Flowchart* Penerapan Metode *Profile Matching*  
Sumber: (Hasil Analisis, 2015)

### 3.5. Metode Pengembangan Sistem

Teknik pengembangan sistem ini menggunakan model *incremental*. *Incremental* merupakan penyempurnaan dari model *waterfall*. Penulis menggunakan model *incremental* karena akan mengurangi beban penulis dalam membangun sistem.

Pembangunan sistem ini dilakukan secara bertahap setiap *increment*, setiap *increment* akan dilakukan pengujian, verifikasi dan validasi sehingga perubahan yang dilakukan tidak terlalu besar. Dalam sistem ini, penulis membagi dalam empat *increment* yaitu:

1. Manajemen Data Pengguna
2. Manajemen Data Sekolah
3. Implementasi Metode
4. Manajemen Data Guru

Pembangunan sistem ini dapat dilakukan secara bertahap sesuai dengan *increment*. Fitur yang dibangun disusun berdasarkan prioritas kepentingan sistem, kesamaan fitur yang dimiliki dan kebutuhan sistem yang harus diselesaikan terlebih dahulu. Adapun tahapan setiap *increment* terdiri dari analisis, desain, *coding*, dan *increment* yang sudah jadi boleh dilakukan pengujian oleh *user*.

#### 3.5.1. Analisis Kebutuhan Sistem

Analisis kebutuhan dalam penelitian ini adalah data yang dibutuhkan dalam membangun sistem ini serta fitur-fitur yang nantinya akan dibangun. Analisis kebutuhan dalam membangun aplikasi ini adalah informasi mengenai penempatan Guru Sekolah Dasar di Kabupaten Jember. Data tersebut diperoleh dari Dinas Pendidikan Kabupaten Jember dan Badan Kepegawaian Daerah Kabupaten Jember. Analisis kebutuhan dalam membangun sistem berdasarkan *increment* adalah:

1. *Increment* manajemen Data Pengguna
2. *Increment* manajemen Data Sekolah
3. *Increment* implementasi Metode
4. *Increment* manajemen Data Guru

### 3.5.2. Desain

Sistem penempatan guru ini dirancang menggunakan beberapa diagram untuk menentukan fitur dan kebutuhan fungsional sistem yang akan dibangun. Diagram tersebut antara lain:

1. *Business Process*
2. *Use Case*
3. *Use Case Scenario*
4. *Sequence Diagram*
5. *Class Diagram*
6. *Entity Relationship Diagram (ERD)*

### 3.5.3. Coding

Pada tahapan *coding* akan dilakukan sesuai dengan desain sistem yang telah dibuat dalam tahap perencanaan. Dalam tahap ini penulisan kode program (*coding*) menggunakan bahasa pemrograman PHP (*Page Hyper Text Pre-Processor*), CSS (*Cascading Style Sheet*), dan *Javascript*. Manajemen menggunakan *framework CI* (*CodeIgniter*) data dalam proses implementasi ini menggunakan DBMS MySQL.

### 3.5.4. Pengujian (*Testing*)

Pada tahap ini, pengujian atau *testing* dilakukan untuk memastikan bahwa kebutuhan perangkat lunak dipenuhi. Sistem ini akan diuji menggunakan dua metode yaitu:

1. *White Box*

*White box testing* merupakan cara pengujian dengan melihat ke dalam modul untuk meneliti kode-kode program yang ada, dan menganalisis apakah ada kesalahan atau tidak. Jika ada modul yang menghasilkan *output* yang tidak sesuai dengan proses bisnis yang dilakukan, maka baris-baris program, variabel, dan parameter yang terlibat pada unit tersebut akan dicek satu persatu dan diperbaiki, kemudian di-*compile* ulang. Pengembang akan melakukan pengukuran dengan menghitung *Cyclomatic Complexity (CC)*.

## 2. *Black Box*

*Black box testing* merupakan metode pengujian perangkat lunak yang memeriksa fungsionalitas dari aplikasi yang bertentangan dengan struktur internal atau kerja. Pengetahuan khusus dari kode aplikasi atau struktur internal dan pengetahuan pemrograman pada umumnya tidak diperlukan. Pengujian sistem ini akan dilakukan oleh dosen pembimbing, beberapa mahasiswa Program Studi Sistem Informasi Universitas Jember serta pihak Badan Kepagawain Daerah Kabupaten Jember.



## BAB 4. ANALISIS DAN PERANCANGAN SISTEM

Bab ini menjelaskan mengenai tahapan pembuatan sistem berdasarkan model *incremental*.

### 4.1. Pengumpulan Data

Data yang diperoleh oleh penulis diperoleh dengan beberapa cara, diantaranya adalah melalui wawancara, studi literatur, observasi objek penelitian. Pengumpulan data dilakukan agar sistem dapat disesuaikan dengan kebutuhan pengguna. Data yang dibutuhkan adalah data sekolah dasar negeri di Kabupaten Jember dan data alur penempatan guru sekolah dasar negeri di Kabupaten Jember. Peneliti mendapatkan data sebagai berikut:

#### 1. Data Sekolah

Data sekolah yang diteliti oleh penulis adalah data sekolah dasar negeri di Kabupaten Jember.

#### 2. Data Jenis Guru SD

Guru sekolah dasar secara umum terbagi atas beberapa bagian. Hal ini disesuaikan dengan klasifikasi dan kualitas sumber daya manusia setiap guru. Klasifikasi guru SD dijelaskan pada Tabel 4.1.

Tabel 4.1 Tabel Jenis Guru Sekolah Dasar

No	Jenis Guru	Keterangan
1	Guru Kelas	Bertugas sebagai wali kelas biasanya juga memiliki kemampuan mengajar semua mata pelajaran umum yang diajarkan pada tingkat sekolah dasar
2	Guru Agama	Bertugas mengajar mata pelajaran kerohanian (agama islam)
3	Guru Penjaskes	Bertugas mengajar mata pelajaran keolahragaan yang berhubungan dengan kesehatan jasmani dan kebugaran.

Sumber: (Hasil Analisis, 2015)

### 3. Data Atribut

Data Atribut merupakan data presentase yang didapatkan dari data pembagian dua jenis tipe nilai prioritas. Pada perhitungan metode *profile matching* dibagi menjadi dua kriteria yaitu *core factor* dan *secondary factor*. *Core factor* merupakan faktor utama yang memiliki pengaruh yang besar dalam perhitungan karena memiliki bobot paling besar. *Secondary factor* ialah bobot tambahan yang nantinya hasil perhitungannya akan digabungkan dengan *core factor*. Presentase bobot dijelaskan pada Tabel 4.2

Tabel 4.2 Tabel Data Atribut

Jenis Atribut	Bobot Presentase
<i>Core factor</i>	60%
<i>Secondary factor</i>	40%

Sumber: (Hasil Analisis, 2015)

### 4. Data Kriteria

Kriteria yang digunakan dalam penelitian ini terdiri atas beberapa jenis kriteria umum. Kriteria dijelaskan pada Tabel 4.3

Tabel 4.3 Tabel Data Kriteria

No	Nama Kriteria	Presentase
1	Jarak	20
2	Kepribadian Diri	30
3	Nilai Tes	50

Sumber: (Hasil Analisis, 2015)

### 5. Data Faktor

Data Faktor merupakan data sub bagian dari data kriteria. Data faktor mengklasifikan data ke bagian yang lebih rinci. Hal ini dibutuhkan untuk proses pengolahan data yang memerlukan banyak kriteria dalam proses perhitungan sesuai dengan metode yang digunakan. Data faktor yang digunakan dalam sistem dijelaskan pada Tabel 4.4

Tabel 4.4 Tabel Data Faktor

No	Nama Faktor	Atribut	Kriteria
1	Jarak	CF	Jarak
2	Status Wilayah	SF	Jarak
3	Usia	CF	Kepribadian Diri
4	Pendidikan	SF	Kepribadian Diri
5	Indeks Prestasi Kumulatif (IPK)	CF	Kepribadian Diri
6	Status Pernikahan	SF	Kepribadian Diri
7	Tes Pengetahuan Umum	SF	Nilai Test
8	Tes Potensi Akademik	SF	Nilai Test
9	Tes Wawasan Kebangsaan	SF	Nilai Test
10	Kemampuan Kosentrasi Bidang	CF	Nilai Test
11	Kemampuan Mengajar	CF	Nilai Test
12	Kemampuan Berkomunikasi Terhadap Siswa	CF	Nilai Test

Sumber: (Hasil Analisis, 2015)

#### 6. Data Skala Faktor

Data skala faktor merupakan skala kriteria yang diinputkan ke dalam sistem oleh guru sedangkan *user* kepala sekolah menggunakannya sebagai target untuk menentukan kualifikasi guru terbaik bagi sekolahnya. Tiap skala faktor memiliki bobot nilai yang berbeda. Berikut adalah tabel skala dari faktor yang ada.

##### a. Jarak

Tabel 4.5 Tabel Jarak

Skala Faktor	Bobot
0- 3	5
3,1 – 6	4
6,1 - 9	3
9,1 - 12	2
$\geq 12,1$	1

Sumber: (Hasil Analisis, 2015)

##### b. Status Wilayah

Tabel 4.6 Tabel Status Wilayah

Skala Faktor	Bobot
Kecamatan Sama	3
Kecamatan Beda Dalam Kota	2
Luar Kota	1

Sumber: (Hasil Analisis, 2015)

## c. Usia

Tabel 4.7 Tabel Usia

Skala Faktor	Bobot
18-21	5
22-25	4
26-29	3
30-33	2
34-40	1

Sumber: (Hasil Analisis, 2015)

## d. Pendidikan

Tabel 4.8 Tabel Pendidikan

Skala Faktor	Bobot
S2	2
S1	1

Sumber: (Hasil Analisis, 2015)

## e. IPK (Indeks Prestasi Kumulatif)

Tabel 4.9 Tabel Indeks Prestasi Kumulatif (IPK)

Skala Faktor	Bobot
3,50 – 4,00	5
3,00 – 3,49	4
2,75 – 2,99	3
2,50 – 2,74	2
0 – 2,49	1

Sumber: (Hasil Analisis, 2015)

## f. Status Pernikahan

Tabel 4.10 Tabel Status Pernikahan

Skala Faktor	Bobot
Belum Menikah	2
Sudah Menikah	1

Sumber: (Hasil Analisis, 2015)

## g. Tes Pengetahuan Umum (TPU)

Tabel 4.11 Tabel Tes Pengetahuan Umum (TPU)

Skala Faktor	Bobot
80-100	5
70-79	4
60-69	3
50-59	2
0-49	1

Sumber: (Hasil Analisis, 2015)

## h. Tes Potensi Akademik (TPA)

Tabel 4.12 Tabel Tes Potensi Akademik (TPA)

Skala Faktor	Bobot
80-100	5
70-79	4
60-69	3
50-59	2
0-49	1

Sumber: (Hasil Analisis, 2015)

## i. Tes Wawasan Kebangsaan

Tabel 4.13 Tabel Tes Wawasan Kebangsaan

Skala Faktor	Bobot
80-100	5
70-79	4
60-69	3
50-59	2
0-49	1

Sumber: (Hasil Analisis, 2015)

## j. Kemampuan Kosentrasi Bidang

Tabel 4.14 Tabel Kemampuan Kosentrasi Bidang

Skala Faktor	Bobot
Sangat Baik	5
Baik	4
Cukup	3
Kurang	2
Sangat Kurang	1

Sumber: (Hasil Analisis, 2015)

## k. Kemampuan Mengajar

Tabel 4.15 Tabel Kemampuan Mengajar

Skala Faktor	Bobot
Sangat Baik	5
Baik	4
Cukup	3
Kurang	2
Sangat Kurang	1

Sumber: (Hasil Analisis, 2015)

## 1. Kemampuan Berkomunikasi Terhadap Siswa

Tabel 4.16 Tabel Kemampuan Berkomunikasi Terhadap Siswa

Skala Faktor	Bobot
Sangat Baik	5
Baik	4
Cukup	3
Kurang	2
Sangat Kurang	1

Sumber: (Hasil Analisis, 2015)

## 7. Analisis Kebutuhan Data

Data yang didapatkan penulis dari berbagai sumber dan literatur selanjutnya dianalisis agar kebutuhan sistem dapat didefinisikan dalam bentuk kebutuhan fungsional dan kebutuhan nonfungsional. Kebutuhan fungsional menggambarkan proses apa saja yang harus dapat dilakukan oleh sistem. Kebutuhan yang berkaitan dengan fungsional atau SRSF (*Software Requirement Specification Functional*) akan dijelaskan pada Tabel 4.17.

Tabel 4.17 Tabel Kebutuhan Fungsional

SRSF_ID	Identifikasi
SRSF_ID 01	Sistem dapat melakukan log in pengguna berdasarkan hak akses
SRSF_ID 02	Sistem dapat mengolah data sekolah
SRSF_ID 03	Sistem dapat mengolah data guru
SRSF_ID 04	Sistem dapat memberikan rekomendasi penempatan guru
SRSF_ID 05	Sistem dapat menampilkan hasil tes guru SD

SRSF_ID 06	Sistem dapat menyimpan data mengenai data sekolah, data guru, data kriteria, data faktor, data target pada database
SRSF_ID 07	Sistem dapat mengelola data pengguna

Sumber: (Hasil Analisis, 2015)

Kebutuhan nonfungsional merupakan fungsi yang ditawarkan oleh sistem mengenai layanan sistem atau *Requirement Specification Non Functional* (SRSNF). Kebutuhan non fungsional pada sistem yang akan dibangun akan dijelaskan pada Tabel 4.18.

Tabel 4.18 Tabel Kebutuhan Non Fungsional

SRSNF_ID	Identifikasi
SRSNF_ID 01	Sistem menggunakan autentifikasi pengguna dengan username dan password
SRSNF_ID 02	Sistem dapat dioperasikan di semua OS
SRSNF_ID 03	Tampilan sistem dibuat semudah mungkin agar dapat dimengerti pengguna (user friendly)
SRSNF_ID 04	Bahasa yang digunakan mudah dipahami oleh pengguna

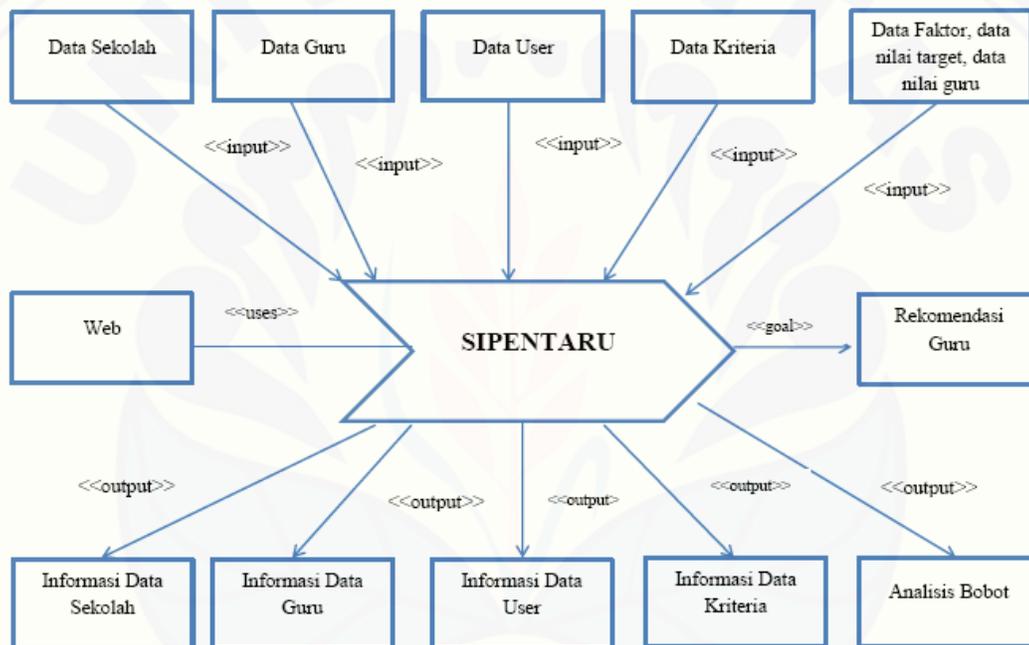
Sumber: (Hasil Analisis, 2015)

## 4.2. Perancangan Sistem

Perancangan sistem merupakan tahapan untuk memodelkan sistem informasi penempatan guru sekolah dasar yang akan dibuat. Dokumen yang akan dibuat meliputi *Bussiness Process*, *Usecase Diagram*, *Usecase Scenario*, *Activity Diagram*, *Sequence Diagram*, *Class diagram* dan *Entity Relationship*. Dengan adanya diagram tersebut diharapkan dapat menggambarkan fungsi dan alur sistem yang dibangun. Berdasarkan model *incremental* yang digunakan pembuatan sistem ini dibagi menjadi empat *increment* yaitu manajemen data pengguna, manajemen data sekolah, analisis data dan manajemen data guru. Fitur utama pada sistem yang dibangun oleh peneliti terdapat pada modul 3 yaitu analisis data, untuk penjelasan perancangan modul *increment* lainnya akan dijelaskan pada lampiran A.

#### 4.2.1. *Bussiness Process*

*Bussiness process* digunakan untuk memodelkan proses yang dilakukan oleh sistem untuk mencapai hasil yang dibutuhkan oleh pengguna. Komponen pada business process ini meliputi *input*, *output*, *uses* dan *goal*. Pada gambar 4.1 dijelaskan mengenai komponen input, output, uses dan goal pada sistem yang akan dibangun oleh peneliti agar jelas data apa saja yang dibutuhkan untuk mencapai tujuan pembuatan sistem.



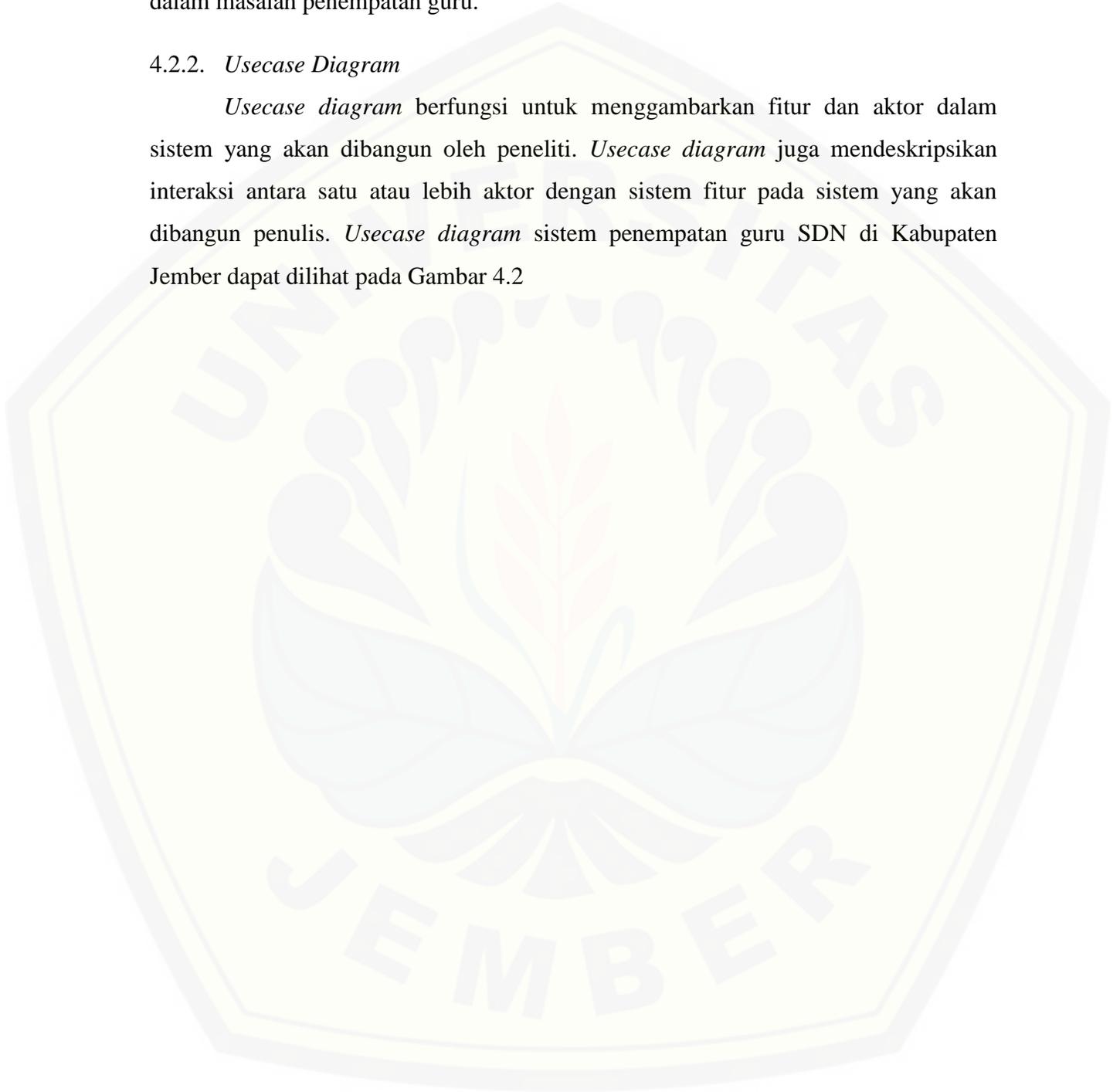
Gambar 4.1 *Bussines Process* Sipentaru  
Sumber: (Hasil Analisis, 2015)

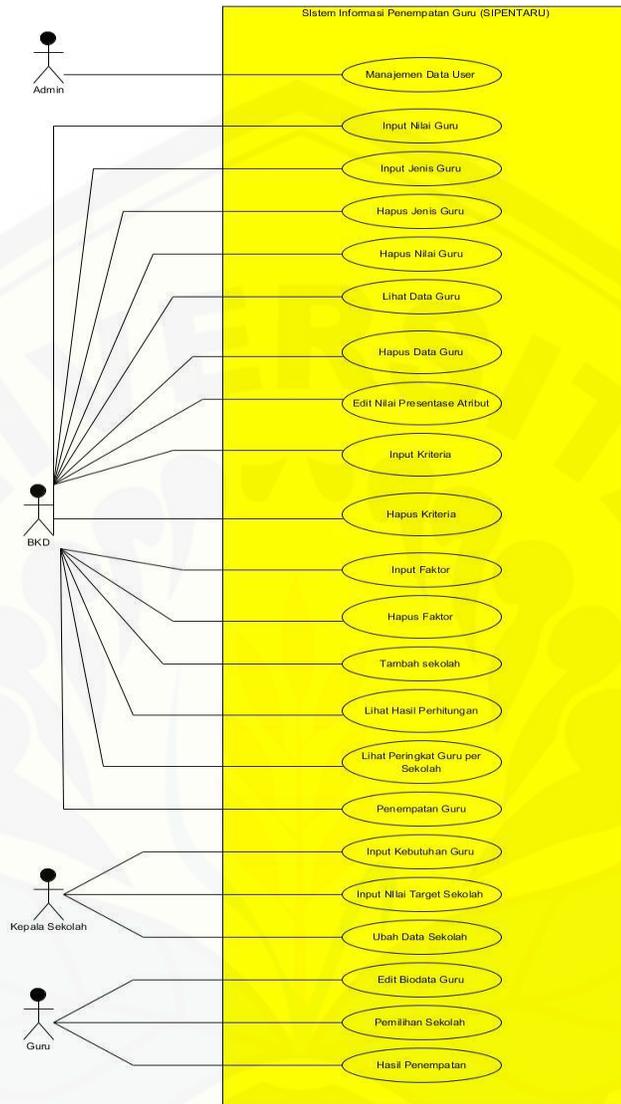
Gambar 4.1 menjelaskan bahwa sistem yang dikembangkan berbasis web. Data yang dimasukkan adalah data sekolah, data guru, data *user*, data kriteria, data faktor, nilai target dan nilai guru. Data tersebut digunakan sebagai data *input* pada SIPENTARU. *Output* dari sistem ini adalah berupa informasi data sekolah, informasi data guru, informasi *user*, informasi data kriteria dan analisis bobot. Tujuan dari

adanya sistem ini adalah sebuah sistem yang membantu rekomendasi terbaik guru dalam masalah penempatan guru.

#### 4.2.2. *Usecase Diagram*

*Usecase diagram* berfungsi untuk menggambarkan fitur dan aktor dalam sistem yang akan dibangun oleh peneliti. *Usecase diagram* juga mendeskripsikan interaksi antara satu atau lebih aktor dengan sistem fitur pada sistem yang akan dibangun penulis. *Usecase diagram* sistem penempatan guru SDN di Kabupaten Jember dapat dilihat pada Gambar 4.2





Gambar 4.2 Use Case Diagram  
 Sumber: (Hasil Analisis, 2015)

Usecase diagram pada Gambar 4.2 terdapat empat jenis klasifikasi aktor yang dapat mengakses sistem yaitu, admin, BKD, kepala sekolah dan guru. Deskripsi aktor dijelaskan pada Tabel 4.19

Tabel 4.19 Tabel Deskripsi Aktor

No	Aktor	Deskripsi
1	Admin	Aktor yang memiliki kewenangan untuk menambah, mengubah, menghapus dan melihat data pengguna

2	Kepala Sekolah	Aktor yang memiliki kewenangan untuk mengubah data sekolah, memasukkan nilai target sekolah dan melihat hasil rekomendasi
3	BKD	Aktor yang memiliki kewenangan untuk menambah, mengubah dan menghapus data sekolah, data kriteria, data factor dan data guru.
4	Guru	Aktor yang memiliki kewenangan untuk memasukkan data pribadi, melihat hasil penempatan

Sumber: (Hasil Analisis, 2015)

Pada *usecase* tersebut dijelaskan bahwa terdapat 25 *usecase* dengan penjelasan setiap *usecase*-nya pada Tabel 4.20 seperti dibawah ini.

Tabel 4.20 Tabel Penjelasan *Use Case*

No	Usecase	Penjelasan
1	<i>Login</i>	Komponen ini digunakan untuk mengakses sistem yang dilakukan dengan
2	Menambah data <i>user</i>	Komponen yang digunakan admin untuk melakukan penambahan user
3	Mengubah data <i>user</i>	Komponen ini digunakan untuk proses ubah data pengguna pada system
4	Menghapus data <i>user</i>	Komponen yang digunakan untuk melakukan hapus data pengguna apabila dianggap tidak perlu
5	Menginput nilai guru	Komponen yang berfungsi untuk melakukan inputan nilai guru yang dilakukan oleh BKD
6	Menginput jenis guru	Komponen yang memiliki fungsi untuk melakukan klasifikasi pembedaan jenis guru pada sistem. Hak akses oleh BKD
7	Menghapus jenis guru	Komponen yang berfungsi untuk melakukan hapus jenis guru. Hak akses oleh BKD
8	Menghapus nilai guru	Komponen yang berfungsi untuk melakukan hapus nilai guru. Hak akses oleh BKD
9	Melihat data guru	Komponen yang memiliki fungsi untuk menampilkan data guru yang akan ditempatkan. Hak akses oleh BKD
10	Menghapus data guru	Komponen yang berfungsi untuk melakukan hapus semua data guru yang telah terdaftar dan akan ditempatkan. Hak akses oleh BKD
11	Mengedit nilai presentase atribut	Komponen yang berfungsi untuk mengubah nilai presentase atribut. Hak akses oleh BKD
12	Menginput kriteria	Komponen yang memiliki fungsi untuk proses input data kriteria. Hak akses oleh BKD
13	Menghapus kriteria	Komponen yang berfungsi untuk melakukan hapus data kriteria. Hak akses oleh BKD
14	Menginput faktor	Komponen yang berfungsi untuk melakukan inputan data faktor yang dibutuhkan oleh sistem. Hak akses oleh BKD
15	Menghapus faktor	Komponen yang bertugas untuk menghapus data faktor

		pada sistem. Hak akses oleh BKD
16	Menambah sekolah	Komponen yang bertugas untuk melakukan penambahan jumlah sekolah yang terdaftar di database BKD. Hak akses oleh BKD
17	Melihat hasil perhitungan	Komponen ini berfungsi untuk menampilkan hasil perhitungan metode yang dilakukan oleh sistem. Hak akses oleh BKD
18	Melihat peringkat guru persekolah	Komponen ini berfungsi untuk melihat data peringkat guru persekolah melalui peringkangan yang telah dihitung oleh sistem itu sendiri.
19	Melakukan penempatan guru	Komponen yang bertugas untuk dapat menampilkan yang digunakan untuk melakukan penempatan guru.
20	Menginput kebutuhan guru	Komponen ini berfungsi untuk melakukan input kebutuhan guru yang disesuaikan oleh kebutuhan kepala sekolah. Hak akses kepala sekolah
21	Menginput nilai target sekolah	Komponen ini melakukan inputan nilai target yang dilakukan pihak sekolah dalam menentukan nilai target yang dibutuhkan sekolah. Hak akses oleh kepala sekolah
22	Mengubah data sekolah	Komponen ini bertugas untuk dapat mengubah data sekolah beisi tentang detail mengenai sekolah termasuk koordinat lokasi. Hak akses oleh kepala sekolah
23	Mengedit biodata guru	Komponen ini bertugas untuk melakukan edit biodata guru yang dilakukan dan diakses oleh guru itu sendiri. Hak akses oleh guru.
24	Melakukan pemilihan sekolah	Komponen ini melakukan pilihan sekolah yang nanti akan berpengaruh pada daftar sekolah yang diinginkan oleh guru itu sendiri. Hak akses oleh guru.
25	Melihat hasil penempatan	Komponen ini bertugas menampilkan data hasil penempatan guru yang telah diproses oleh sistem berdasarkan keinginan dan kesesuaian guru. Hak akses oleh guru.

Sumber: (Hasil Analisis, 2015)

#### 4.2.3. Perancangan Modul III Implementasi Metode

##### 1. Usecase skenario Modul III Implementasi Metode

###### a. Mengubah nilai presentasi atribut

*Usecase* skenario pada fitur mengubah nilai presentase dapat dilihat pada Tabel 4.21. Tabel menjelaskan aksi aktor dan sistem pada skenario normal dan alternatif pada sistem.

Tabel 4.21 Mengubah Nilai Presentase Atribut

Nama		Mengubah nilai presentase atribut	
Aktor		BKD	
Entry Condition		Aktor mengubah nilai presentase atribut	
Exit Condition		Nilai presentasi atribut berhasil diubah	
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Memilih menu Kriteria		
2	Pilih submenu Atribut		
		3	Menampilkan halaman nilai atribut
4	Pilih atribut yang akan diubah presentasinya		
5	Klik Aksi Ubah presentase		
		6	Menampilkan form ubah nilai presentase
7	Memasukkan nilai presentase		
8	Klik OK		
		9	Data berhasil diubah sesuai dengan presentase yang dimasukkan
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
7a.	Memasukkan nilai kosong		
8a.	Klik OK		
		9a.	Menampilkan alert bahwa data belum mencapai 100 persen
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-2</b>			
7b.	Memasukkan nilai kosong		
8b.	Klik Cancel		
		9b.	Kembali pada halaman nilai presentase atribut

Sumber: (Hasil Analisis, 2015)

## b. Input Kriteria

*Usecase* skenario pada fitur *input* kriteria dapat dilihat pada Tabel 4.22. Tabel menjelaskan aksi aktor dan sistem pada skenario normal dan alternatif pada sistem.

Tabel 4.22 *Input* Kriteria

Nama	Menambah data kriteria		
Aktor	BKD		
Entry Condition	Aktor menambah data kriteria		
Exit Condition	Data kriteria berhasil ditambahkan		
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Memilih menu Kriteria		
2	Pilih submenu Kriteria		
		3	Menampilkan halaman tabel kriteia dan form tambah kriteria
4	Masukkan kriteria		
5	Klik Submit		
		6	Kriteria berhasil ditambahkan dan muncul pada tabel kriteria
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
4	Memasukkan data kriteria kosong		
5	Klik Submit		
		6	Muncul alert pada form pengisian dan data tidak berhasil ditambahkan

Sumber: (Hasil Analisis, 2015)

## c. Hapus Kriteria

*Usecase* skenario pada fitur hapus kriteria dapat dilihat pada Tabel 4.23. Tabel menjelaskan aksi aktor dan sistem pada skenario normal dan alternatif pada sistem.

Tabel 4.23 Hapus Kriteria

Nama	Menghapus data kriteria		
Aktor	BKD		
Entry Condition	Aktor menghapus data kriteria		
Exit Condition	Data kriteria berhasil dihapus		
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Memilih menu Kriteria		
2	Pilih submenu Kriteria		
		3	Menampilkan halaman tabel kriteia
4	Memilih kriteria yang akan dihapus		

5	Klik Hapus		
		6	Muncul alert apakah yakin akan menghapus
7	Klik OK		
		8	Data Kriteria berhasil dihapus
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
7	Klik Cancel		
		8	Kembali pada halaman tabel kriteria dan data tidak berhasil dihapus

Sumber: (Hasil Analisis, 2015)

d. *Input* Faktor

*Usecase* skenario pada fitur *input* faktor dapat dilihat pada Tabel 4.24. Tabel menjelaskan aksi aktor dan sistem pada skenario normal dan alternatif pada sistem.

Tabel 4.24 *Input* Faktor

Nama	Menambah data factor		
Aktor	BKD		
Entry Condition	Aktor menambahkan data factor		
Exit Condition	Data factor berhasil ditambahkan		
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Memilih menu kriteria		
2	Memilih submenu factor		
		3	Menampilkan halaman input factor dan tabel data factor
4	Masukkan data faktor, apabila data factor terdapat skala maka memilih tipe isian data “ambil dari skala”		
5	Klik Submit		
		6	Data Faktor berhasil ditambahkan dan muncul pada tabel factor
7	Klik menu Input Skala pada tabel data faktor data yang baru ditambahkan		
		8	Menampilkan halaman input skala dan tabel skala
9	Masukkan Skala sesuai data yang dibutuhkan		
10	Klik Submit		
		11	Data skala berhasil ditambahkan
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			

4a	Masukan data faktor, data faktor berupa data biasa atau tidak berskala		
5a	Klik Submit		
		6a	Data berhasil ditambahkan
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-2</b>			
4b	Data tidak lengkap		
5b	Klik Submit		
		6b	Menampilkan alert pada form yang tidak lengkap
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-3</b>			
9c	Data tidak lengkap atau kosong		
10c	Klik Submit		
		11c	Menampilkan alert pada form yang kurang lengkap

Sumber: (Hasil Analisis, 2015)

e. Hapus Faktor

*Usecase* skenario pada fitur mengubah nilai presentase dapat dilihat pada Tabel 4.25. Tabel menjelaskan aksi aktor dan sistem pada skenario normal dan alternatif pada sistem.

Tabel 4.25 Hapus Faktor

Nama	Menghapus data factor		
Aktor	BKD		
Entry Condition	Aktor menghapus data factor		
Exit Condition	Data faktor berhasil dihapus		
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Memilih menu Kriteria		
2	Pilih submenu factor		
		3	Menampilkan halaman tabel faktor
4	Memilih faktor yang akan dihapus		
5	Klik Hapus		
		6	Muncul alert apakah yakin akan menghapus
7	Klik OK		
		8	Data faktor berhasil dihapus
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
7	Klik Batal		
		8	Kembali pada halaman tabel faktor dan data tidak berhasil dihapus

Sumber: (Hasil Analisis, 2015)

f. Lihat hasil perhitungan

*Usecase* skenario pada fitur lihat hasil perhitungan dapat dilihat pada Tabel 4.25. Tabel menjelaskan aksi aktor dan sistem pada skenario normal dan alternatif pada sistem

Tabel 4.26 Lihat Hasil Perhitungan

Nama	Melihat hasil perhitungan
Aktor	BKD
Entry Condition	Aktor melihat data hasil perhitungan
Exit Condition	Data berhasil dilihat
<b>SKENARIO NORMAL SN-UC-1</b>	
1	Memilih menu metode
2	Pilih submenu Hasil Perhitungan
	3 Menampilkan tabel hasil perhitungan
4	Pilih submenu Nilai Core dan Secondary
	5 Menampilkan tabel hasil perhitungan nilai core dan secondary
6	Pilih submenu nilai total
	7 Menampilkan tabel hasil nilai keseluruhan guru

Sumber: (Hasil Analisis, 2015)

g. Penempatan guru

*Usecase* skenario pada fitur penempatan guru dapat dilihat pada Tabel 4.26. Tabel menjelaskan aksi aktor dan sistem pada skenario normal dan alternatif pada sistem.

Tabel 4.27 Penempatan Guru

Nama	Penempatan guru
Aktor	BKD
Entry Condition	Aktor melakukan penempatan guru
Exit Condition	Guru berhasil ditempatkan
<b>SKENARIO NORMAL SN-UC-1</b>	
1	Memilih menu metode
2	Pilih submenu penempatan
	3 Menampilkan tabel kebutuhan guru pada sekolah
4	Memilih sekolah yang akan dilakukan penempatan guru
5	Klik lakukan penempatan
	6 Menampilkan daftar guru yang memilih sekolah tersebut

7	Memilih guru yang akan ditempatkan		
8	Klik Pilih guru		
		9	Menampilkan alert “pilihan tidak dapat diubah, Apakah anda yakin”?
10	Klik OK		
		11	Guru berhasil ditempatkan
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
10	Klik Batal		
		11	Kembali pada halaman penempatan

Sumber: (Hasil Analisis, 2015)

h. Lihat peringkat guru per sekolah

*Usecase* skenario pada fitur lihat peringkat guru per sekolah dapat dilihat pada Tabel 4.27. Tabel menjelaskan aksi aktor dan sistem pada skenario normal dan alternatif pada sistem.

Tabel 4.28 Lihat Peringkat Guru Per Sekolah

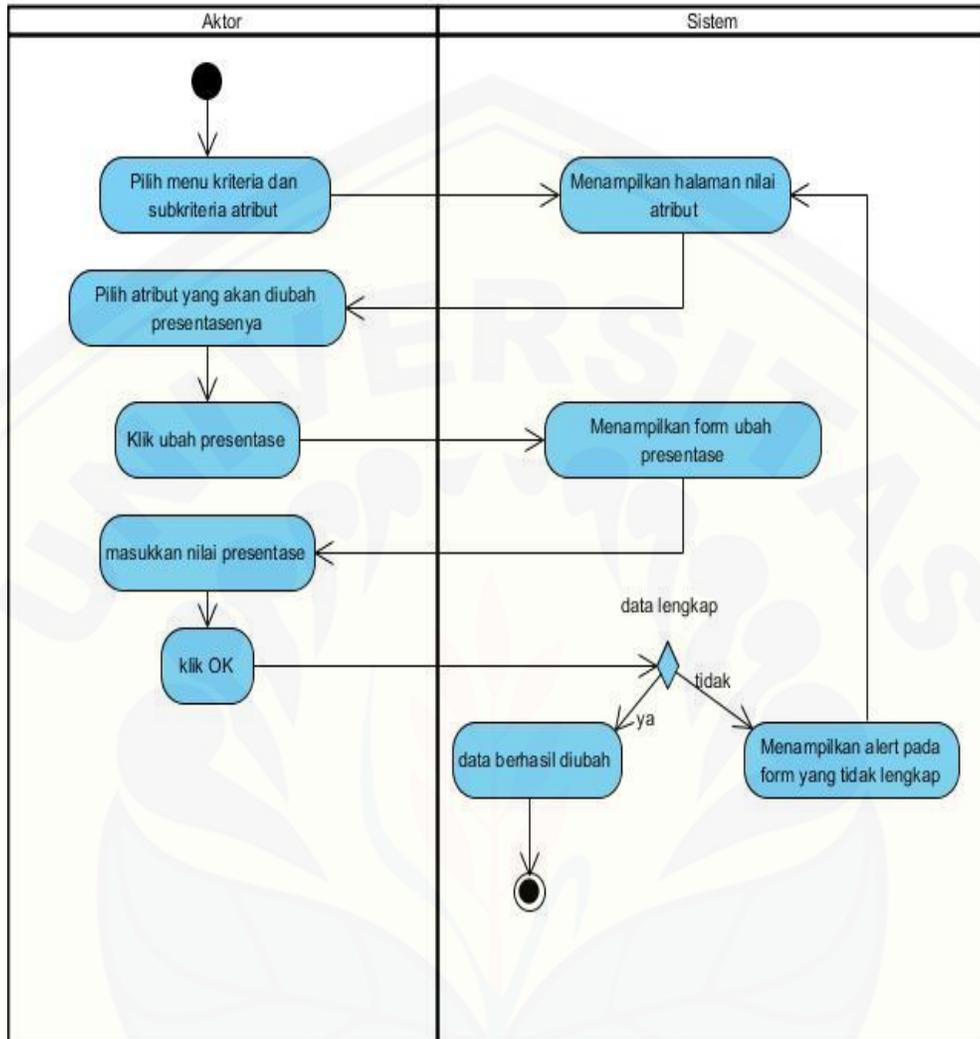
Nama	Melihat hasil peringkat guru per sekolah	
Aktor	BKD	
Entry Condition	Aktor melihat peringkat guru per sekolah	
Exit Condition	Data berhasil dilihat	
<b>SKENARIO NORMAL SN-UC-1</b>		
1	Memilih menu metode	
2	Pilih submenu peringkat per sekolah	
	3	Menampilkan tabel pilihan sekolah
4	Pilih sekolah yang akan dilihat	
5	Klik Submit	
	6	Menampilkan data peringkat guru pada sekolah tersebut

Sumber: (Hasil Analisis, 2015)

2. Activity Diagram

a. Edit nilai presentase atribut

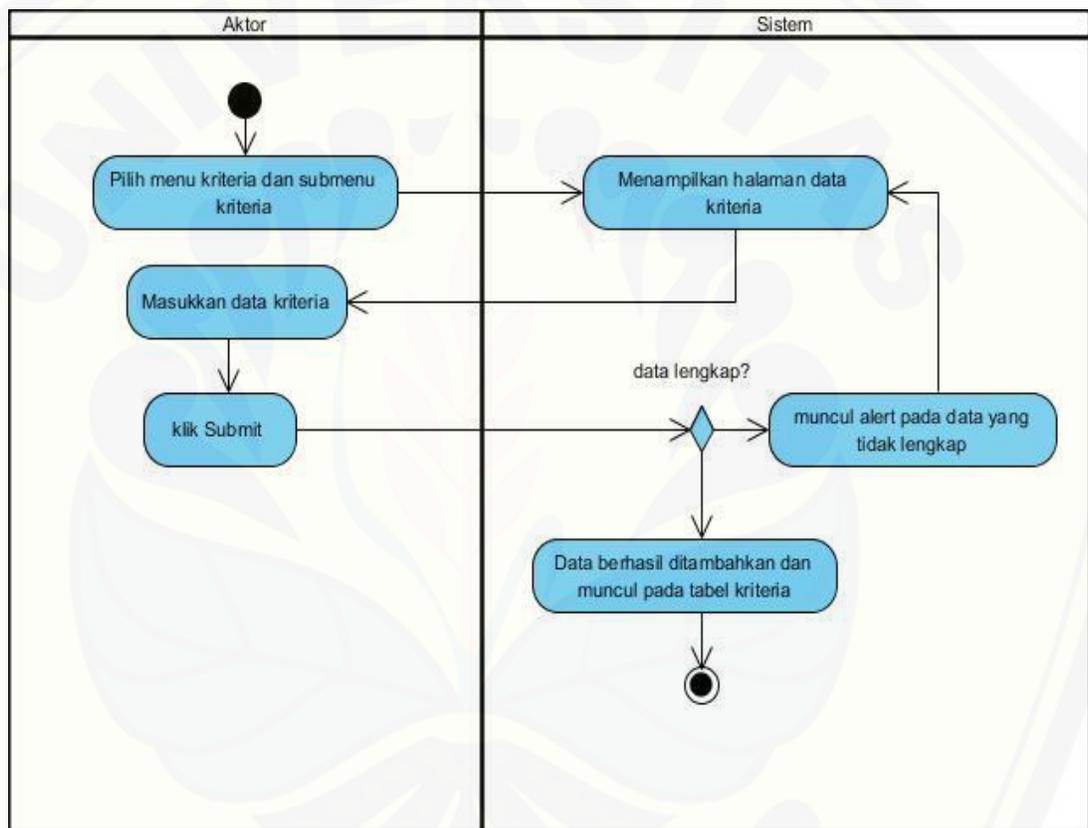
*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur *edit* nilai presentase atribut. Adapun penjelasan mengenai *Activity diagram* ini digambarkan pada Gambar 4.3.



Gambar 4.3 *Edit Nilai Presentase Atribut*  
 Sumber: (Hasil Analisis, 2015)

**b. Input Kriteria**

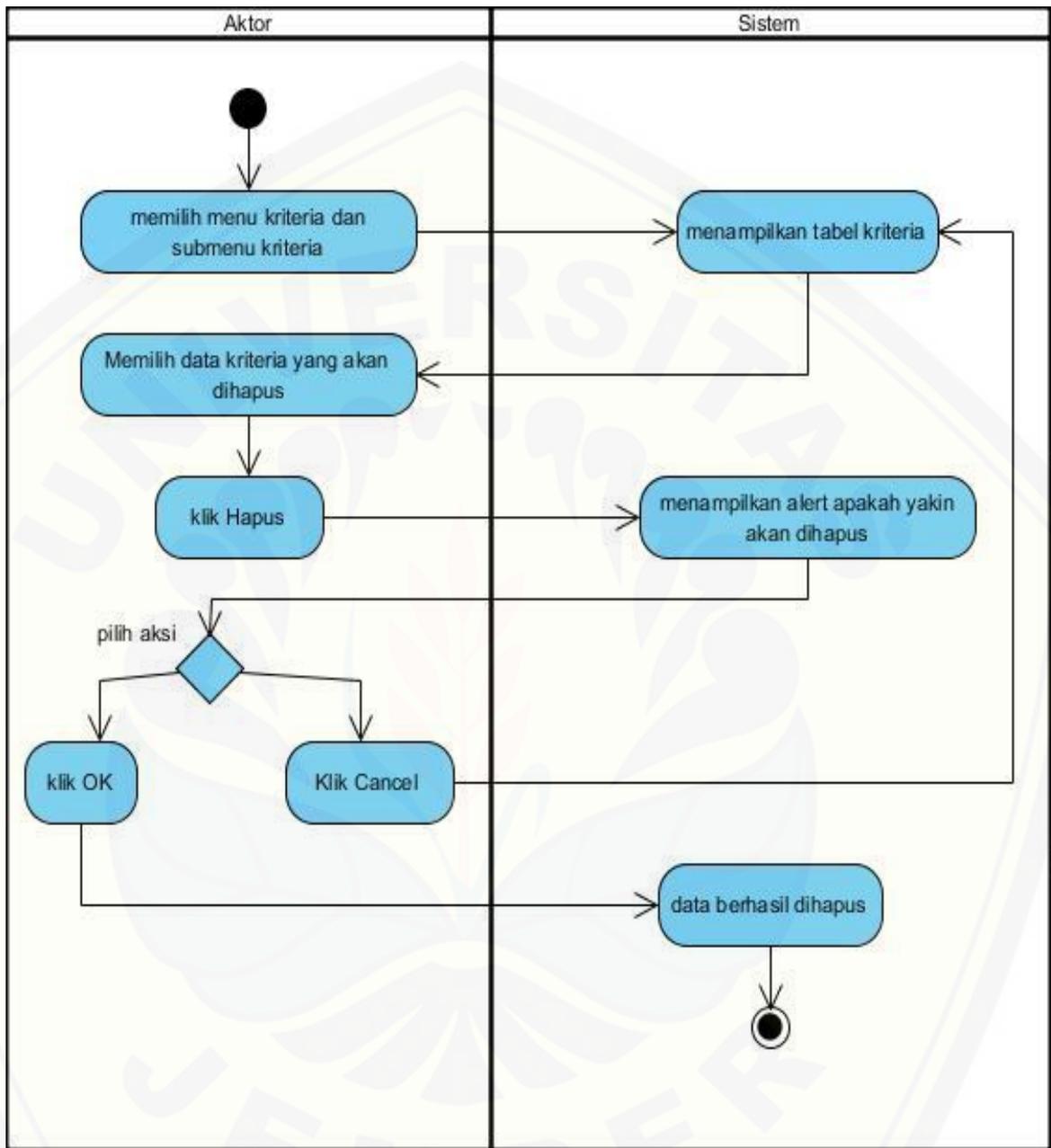
*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur *input* kriteria. Fitur ini berfungsi untuk menginputkan nilai kriteria pada sistem sesuai dengan kebutuhan yang ada. Adapun penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 4.4.



Gambar 4.4 *Input* Kriteria  
Sumber: (Hasil Analisis, 2015)

**c. Hapus Kriteria**

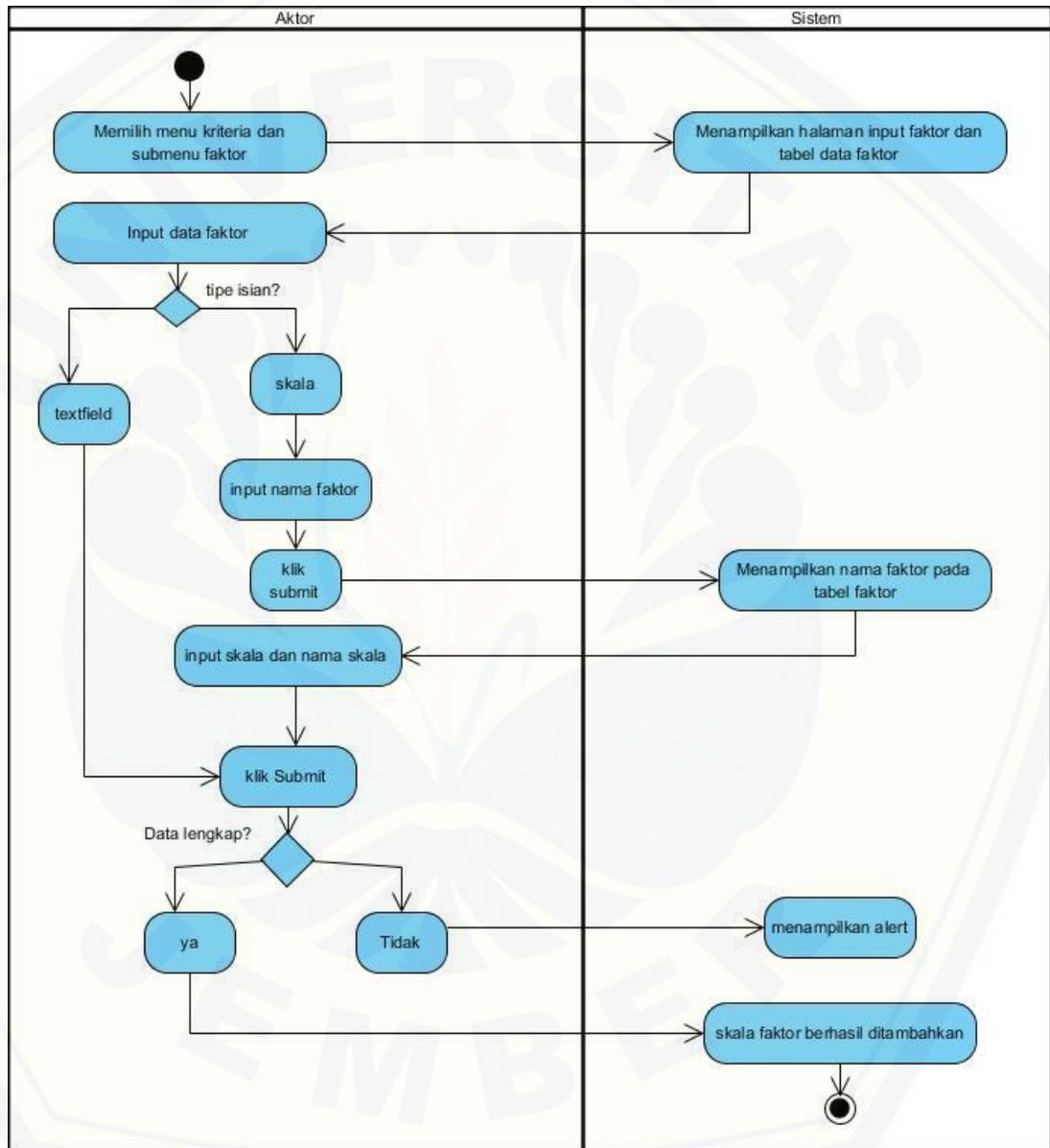
*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur hapus kriteria. Fitur ini berfungsi untuk menghilangkan salah satu kriteria yang tidak digunakan atau akan dihapus. Adapun penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 4.5.



Gambar 4.5 Hapus Kriteria  
 Sumber: (Hasil Analisis, 2015)

d. *Input Faktor*

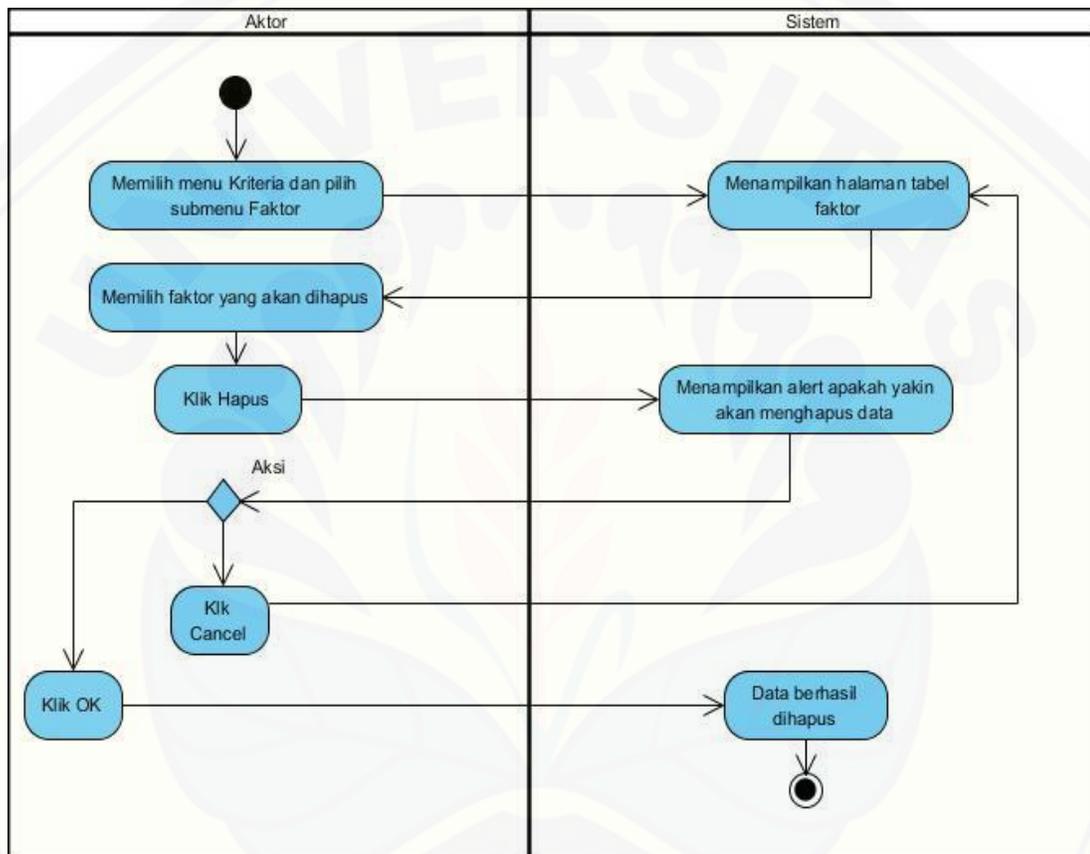
*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur *input faktor*. Adapun penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 4.6.



Gambar 4.6 *Input Faktor*  
Sumber: (Hasil Analisis, 2015)

## e. Hapus Faktor

*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur hapus faktor. Adapun penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 4.7.

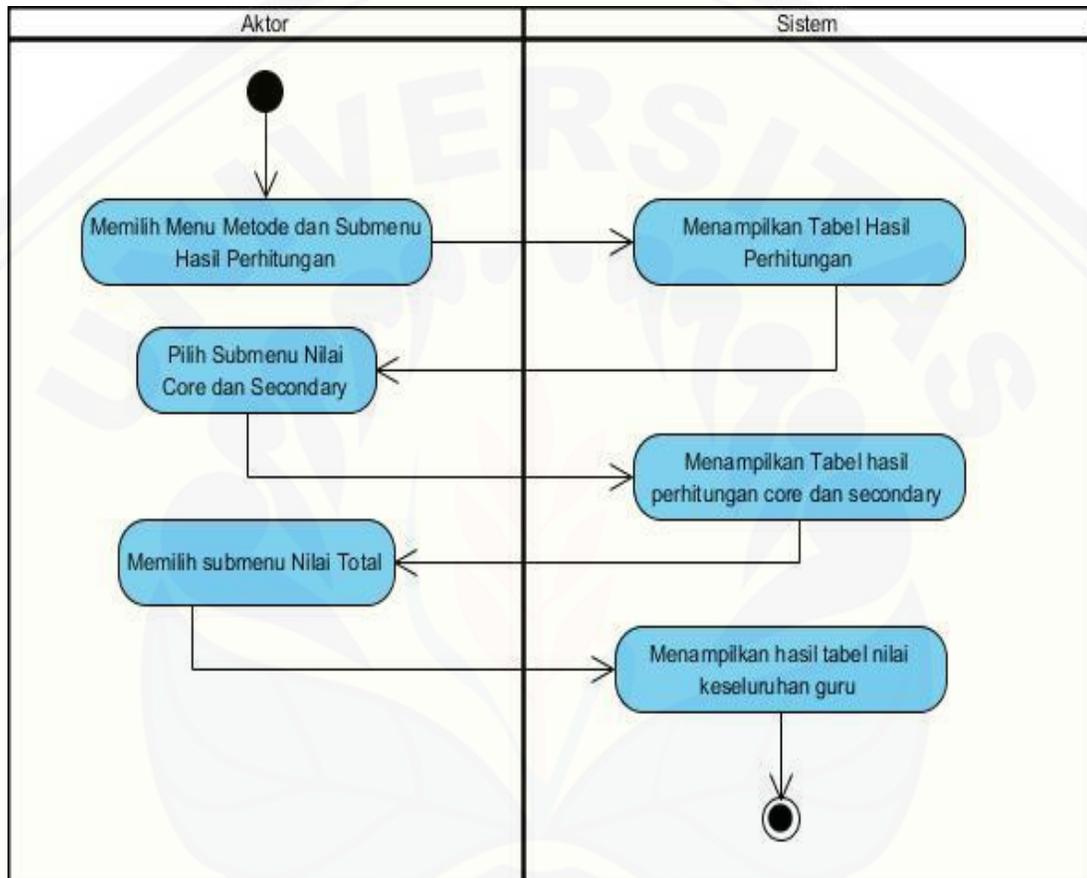


Gambar 4.7 Hapus Faktor  
Sumber: (Hasil Analisis, 2015)

## f. Lihat Hasil Perhitungan

*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur lihat hasil perhitungan. Fitur ini berfungsi untuk membantu *user* dalam melihat hasil perhitungan yang dilakukan oleh sistem. Sistem melakukan

perhitungan sesuai dengan metode *profile matching* yang telah dimasukkan kedalam baris *coding* yang ada pada sistem. Adapun penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 4.8.



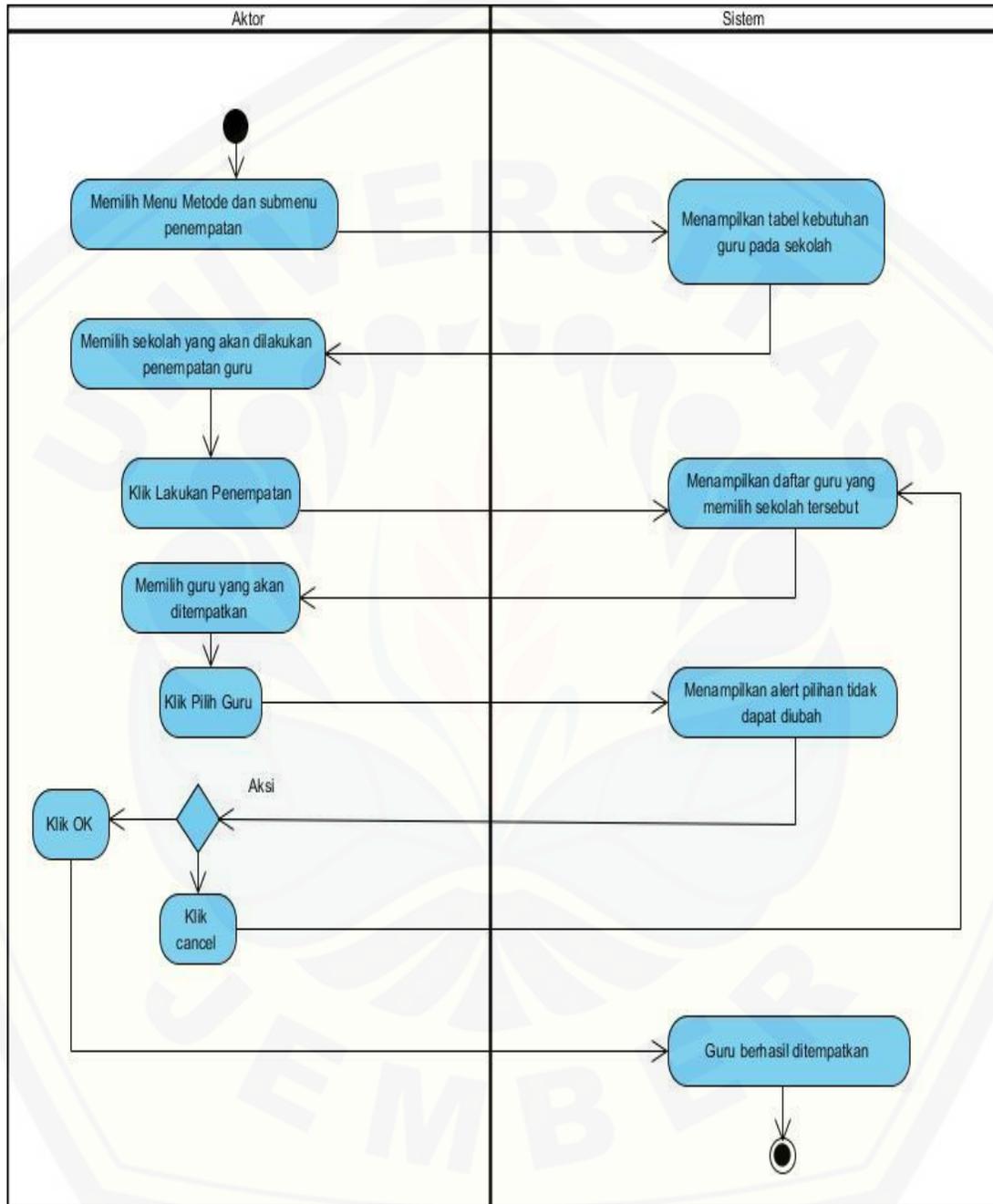
Gambar 4.8 Lihat Hasil Perhitungan

Sumber: (Hasil Analisis, 2015)

g. Penempatan guru

*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur penempatan guru. Fitur ini digunakan oleh *user* untuk melihat dimana guru ditempatkan. Penempatan guru dilakukan oleh BKD. *User* guru hanya

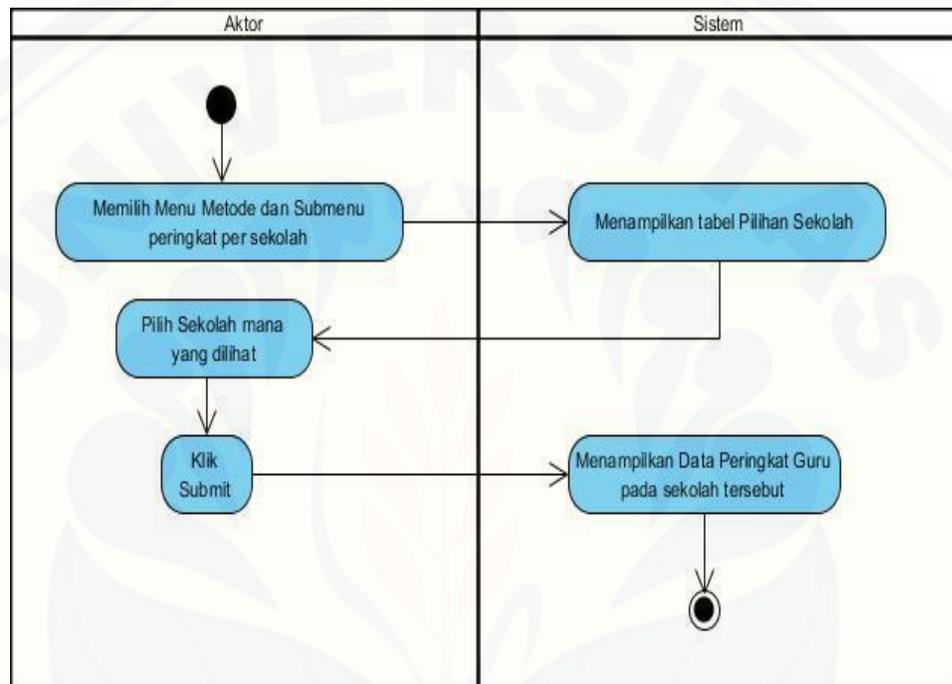
bisa melihat hasil penempatan. Penjelasan mengenai *activity diagram* pada fitur penempatan guru digambarkan secara detail pada Gambar 4.9.



Gambar 4.9 Penempatan Guru  
 Sumber: (Hasil Analisis, 2015)

#### h. Lihat Peringkat Guru per Sekolah

*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur lihat guru per sekolah. Fitur lihat peringkat guru per sekolah berfungsi untuk menampilkan peringkat guru berdasarkan sekolah tertentu. Adapun penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 4.10.

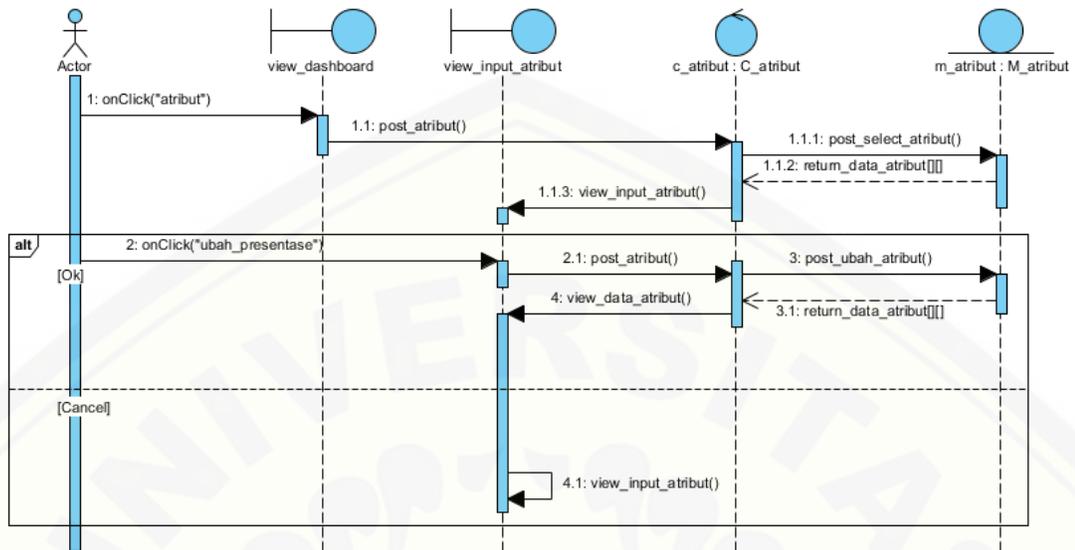


Gambar 4.10 Lihat Peringkat Guru per Sekolah  
Sumber: (Hasil Analisis, 2015)

### 3. Sequence Diagram

#### a. Edit Nilai Presentase Atribut

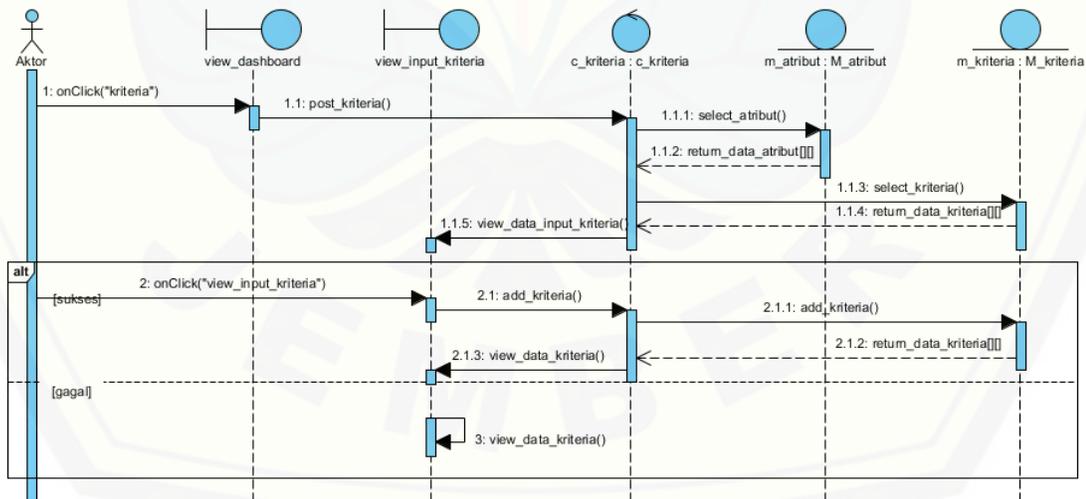
Gambar 4.11 adalah diagram *sequence* untuk menggambarkan skenario dan memodelkan aliran logika yang menghubungkan *view*, *controller* dan model pada fitur *edit* nilai presentase atribut.



Gambar 4.11 *Sequence Diagram Edit Nilai Presentase Atribut*  
 Sumber: (Hasil Analisis, 2015)

b. Input Kriteria

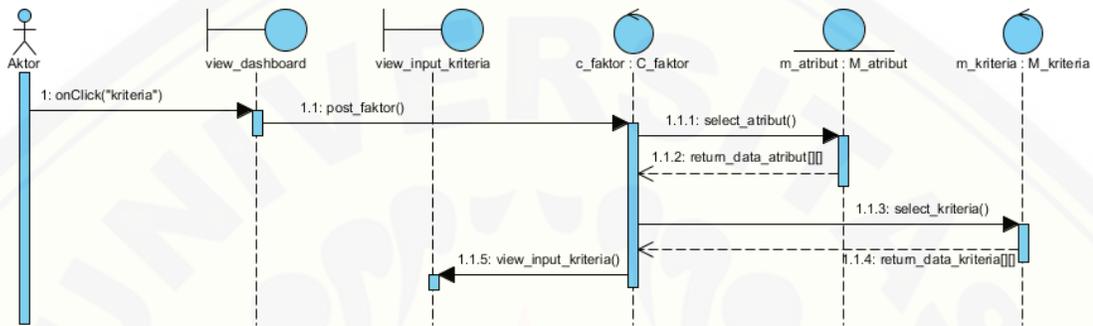
Gambar 4.12 adalah diagram *sequence* untuk menggambarkan skenario dan memodelkan aliran logika yang menghubungkan *view*, *controller* dan model pada fitur *input kriteria*.



Gambar 4.12 *Sequence Diagram Input Kriteria*  
 Sumber: (Hasil Analisis, 2015)

c. Hapus Kriteria

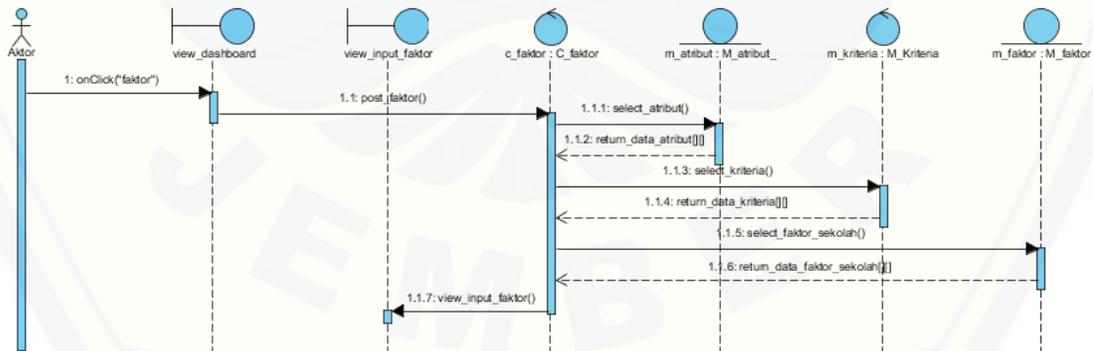
Gambar 4.13 adalah diagram *sequence* untuk menggambarkan skenario dan memodelkan aliran logika yang menghubungkan *view*, *controller* dan model pada fitur hapus kriteria.



Gambar 4.13 *Sequence Diagram* Hapus Kriteria  
 Sumber: (Hasil Analisis, 2015)

d. *Input* Faktor

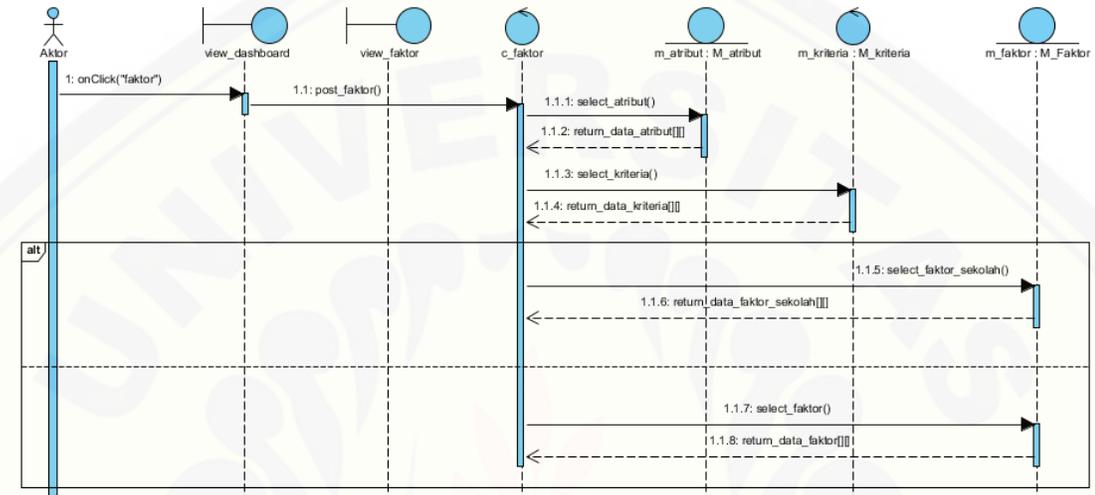
Gambar 4.14 adalah diagram *sequence* untuk menggambarkan skenario dan memodelkan aliran logika yang menghubungkan *view*, *controller* dan model pada fitur *input* faktor.



Gambar 4.14 *Sequence Diagram* *Input* Faktor  
 Sumber: (Hasil Analisis, 2015)

e. Hapus Faktor

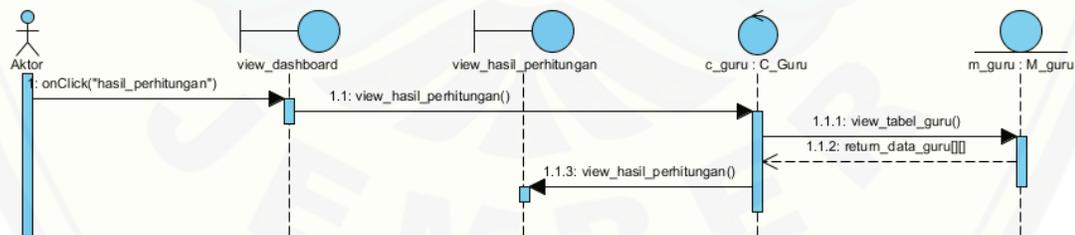
Gambar 4.15 adalah diagram *sequence* untuk menggambarkan skenario dan memodelkan aliran logika yang menghubungkan *view*, *controller* dan model pada fitur hapus kriteria.



Gambar 4.15 *Sequence Diagram* Hapus Faktor  
 Sumber: (Hasil Analisis, 2015)

f. Lihat Hasil Perhitungan

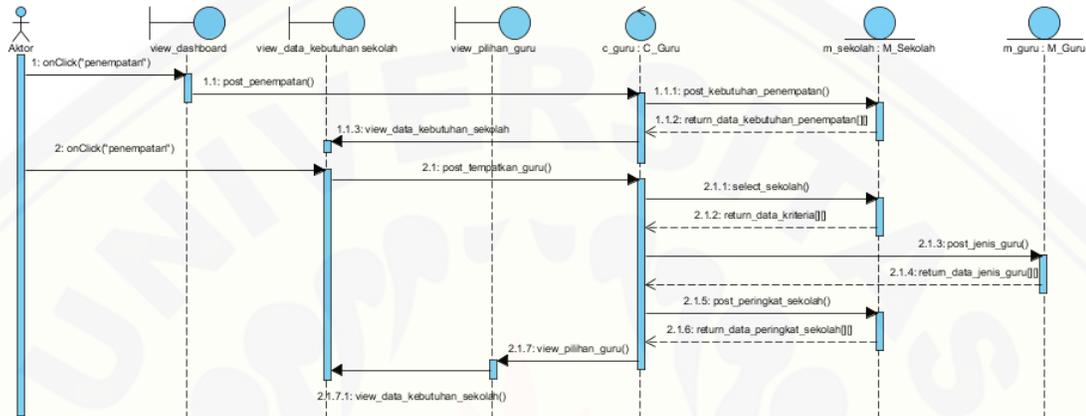
Gambar 4.16 adalah diagram *sequence* untuk menggambarkan skenario dan memodelkan aliran logika yang menghubungkan *view*, *controller* dan model pada fitur lihat hasil perhitungan.



Gambar 4.16 *Sequence Diagram* Lihat Hasil Perhitungan  
 Sumber: (Hasil Analisis, 2015)

g. Penempatan Guru

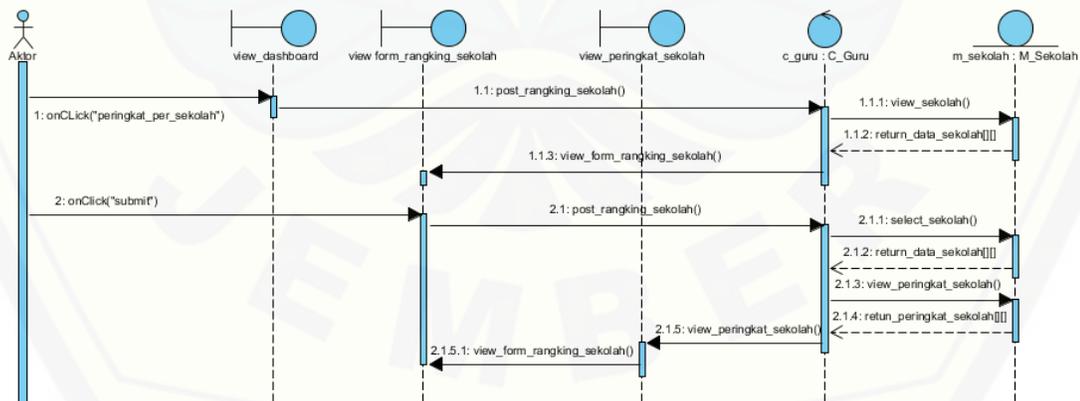
Gambar 4.17 adalah diagram *sequence* untuk menggambarkan skenario dan memodelkan aliran logika yang menghubungkan *view*, *controller* dan model pada fitur penempatan guru.



Gambar 4.17 *Sequencce Diagram* Penempatan Guru  
 Sumber: (Hasil Analisis, 2015)

h. Lihat Peringkat Guru Per Sekolah

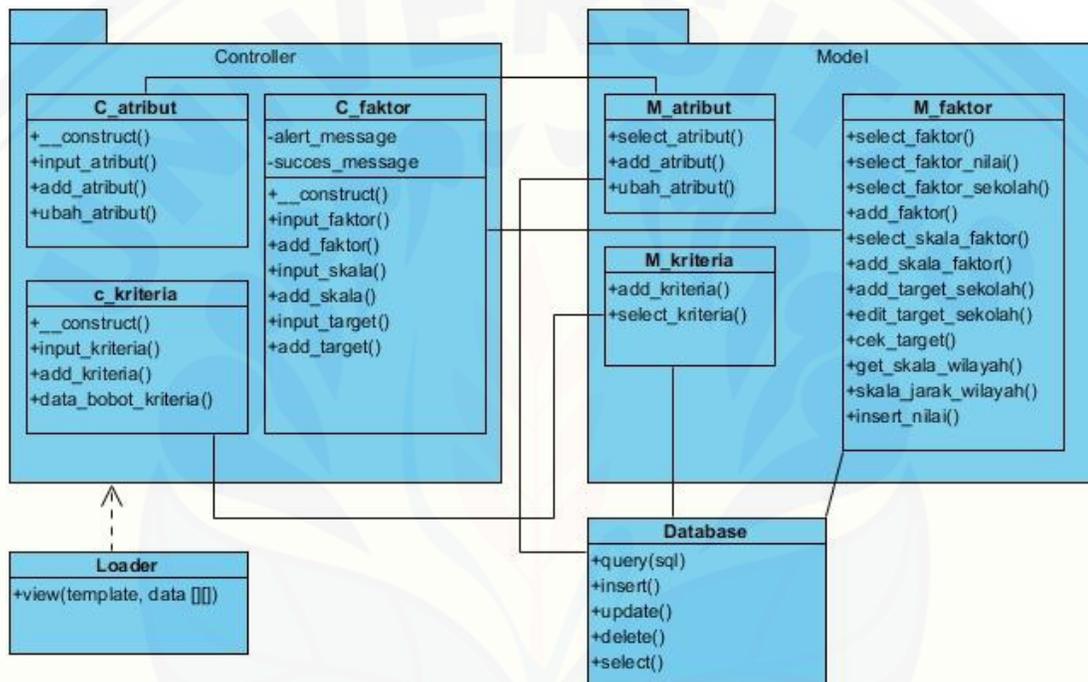
Gambar 4.18 diagram *sequence* skenario dan memodelkan aliran logika menghubungkan *view*, *controller* dan model pada fitur lihat peringkat guru per sekolah.



Gambar 4.18 *Sequence Diagram* Lihat Peringkat Guru per Sekolah  
 Sumber: (Hasil Analisis, 2015)

#### 4. Class Diagram

Gambar 4.19 merupakan gambaran dari *class diagram* implementasi metode yang menghasilkan beberapa jenis operasi yang digunakan dalam alur kerja sistem. Pada class diagram ini menghubungkan method pada *controller*, *model*, *database* dan *loader*. Gambaran dari *class diagram* dijelaskan pada Gambar 4.19.

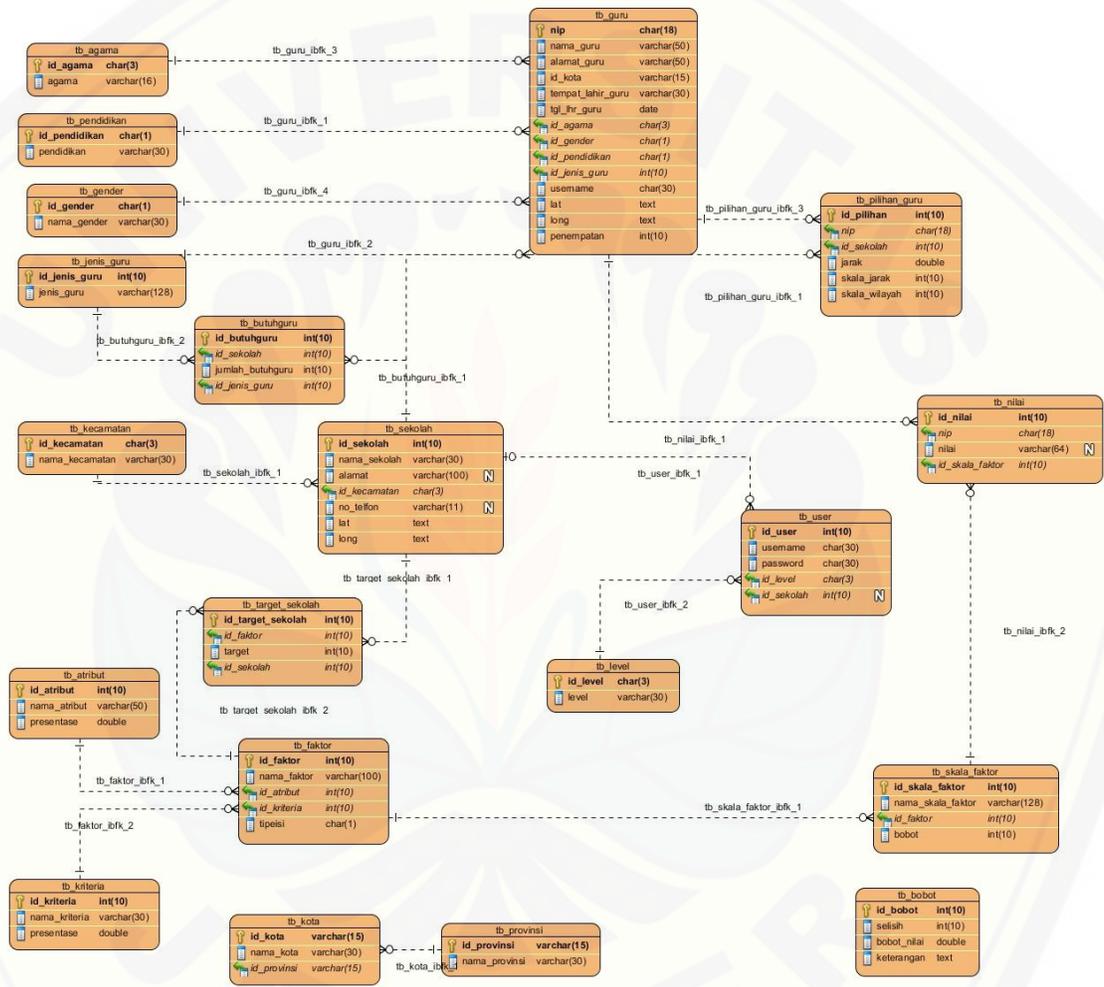


Gambar 4.19 *Class Diagram* Modul Implementasi Metode

Sumber: (Hasil Analisis, 2015)

5. ERD

Entity Relationship Diagram modul implementasi metode berisi gambaran relasi data yang terhubung dalam sistem dan database. Gambaran Entity Relationship Diagram dijelaskan pada Gambar 4.20.



Gambar 4.20 Entity Relationship Diagram Implementasi Metode  
Sumber: (Hasil Analisis, 2015)

#### 4.2.4. Perancangan Modul I Manajemen Data Pengguna

Perancangan Modul I Manajemen Data pengguna meliputi menambah data pengguna, mengubah data pengguna, menghapus data pengguna. Perancangan modul ini akan dijelaskan pada lampiran A.1.

#### 4.2.5. Perancangan Modul II Manajemen Data Sekolah

Perancangan modul II Manajemen Data Sekolah meliputi tambah data sekolah, *input* nilai target sekolah, mengubah data sekolah. Perancangan modul ini akan dijelaskan pada lampiran A.2.

#### 4.2.6. Perancangan Modul IV Manajemen Data Guru

Perancangan modul 4 Manajemen data guru meliputi *input* jenis guru, hapus jenis guru, *input* nilai guru, hapus nilai guru, lihat data guru, hapus data guru, *input* kebutuhan guru, *edit* biodata guru dan pemilihan sekolah. Perancangan modul ini akan dijelaskan pada lampiran A.3.

### 4.3. Pengkodean Sistem

Pengkodean Sistem pada sistem informasi penempatan guru sekolah dasar negeri di Kabupaten Jember (SIPENTARU) menggunakan bahasa pemrograman *Page Hyper Text Pre-Process (PHP)*. Manajemen *database* menggunakan PHP MySQL dengan dukungan *software XAMPP*.

### 4.4. Pengujian Sistem

Pengujian sistem dilakukan untuk mengetahui kualitas dan standar dari sistem itu sendiri apakah sudah layak dan sesuai dengan kebutuhan *user* atau belum. Pengujian sistem dilakukan melalui dua metode yaitu *White Box* dan *Black Box*.

#### 4.4.1. Pengujian *White Box*

Teknik pengujian *white box* dilakukan dengan menggunakan pengujian jalur dasar (*basic path testing*) yang di dalamnya terdapat beberapa tahapan pengujian,

yaitu pembuatan diagram alir, penentuan jalur independen, penghitungan kompleksitas siklomatik jalur independen dan *test case*. Pengujian dilakukan dalam beberapa tahapan, antara lain:

1. Penomoran *listing* program
2. Pembuatan diagram alir
3. Pembuatan grafik air
4. Penghitungan kompleksitas siklomatik

#### 4.4.2. Pengujian *Black Box*

Pengujian sistem menggunakan metode *black box* adalah metode pengujian yang dilakukan oleh *tester* secara langsung dengan menjalankan fungsi sistem yang ada yang kemudian mengecek apa semua fitur dapat berjalan sebagaimana mestinya. Pengujian *black box* berfokus pada *input* yang dimasukkan atau dilakukan pengguna dan *output* yang dihasilkan oleh sistem. Hasil pengujian *black box* dapat dilihat pada lampiran D.

## BAB 5. HASIL DAN PEMBAHASAN

### 5.1. Hasil Penelitian

Sistem Informasi Pendukung Keputusan Penempatan Guru Sekolah Dasar (SIPENTARU) merupakan sebuah sistem informasi berbasis *web* yang digunakan untuk menentukan penempatan guru sekolah dasar di Kabupaten Jember. Sistem ini memiliki fitur utama yaitu perhitungan metode *profile matching* yang digunakan dalam menentukan penempatan guru sesuai dengan kriteria yang digunakan.

### 5.2. Hasil Analisa

Analisa dalam penelitian skripsi ini dilakukan dengan memasukan data kriteria, faktor, atribut yang telah ditentukan oleh Badan Kepagawaian Daerah (BKD) sebagai indikator dalam melakukan perhitungan metode *profile matching* yang disesuaikan dengan target yang telah ditentukan oleh kepala sekolah. Hasil analisis dalam penelitian ini dijelaskan sebagai berikut:

#### 5.2.1. Atribut Bobot Presentase

Atribut yang digunakan dalam perhitungan memiliki bobot presentase yang berbeda. Bobot presentase yang digunakan dalam skripsi ini disesuaikan dengan prioritas dalam metode *profile matching* dan analisa kebutuhan yang ada. Dalam penelitian ini jenis atribut yang digunakan adalah:

1. *Core factor*

Merupakan jenis atribut pokok yang memiliki bobot presentase yang lebih besar daripada *secondary factor* karena merupakan prioritas utama dalam perhitungan.

2. *Secondary factor*

Merupakan jenis atribut pendamping atau pelengkap dari *core factor* sebagai atribut tambahan dalam perhitungan metode *profile matching*. Bobot presentase yang dimiliki lebih kecil daripada *core factor*. Data jenis atribut dijelaskan pada Tabel 5.1.

Tabel 5.1 Tabel Atribut Bobot Presentase

Jenis Atribut	Bobot Presentase
<i>Core factor</i>	60%
<i>Secondary factor</i>	40%

Sumber: (Hasil Analisis, 2015)

### 5.2.2. Faktor dan Kriteria yang Digunakan

Faktor dan kriteria yang digunakan dalam penelitian ini digunakan untuk menentukan kriteria apa saja yang dibutuhkan untuk menentukan calon guru yang akan ditempatkan dan digunakan untuk perhitungan metode *profile matching*. Faktor yang telah ada kemudian ditentukan termasuk dalam kategori atribut *core factor* atau *secondary factor*. Tahapan terakhir adalah menentukan termasuk ke dalam kriteria jarak, kepribadian diri atau kedalaman nilai test. Faktor dan kriteria dijelaskan pada Tabel 5.2.

Tabel 5.2 Tabel Faktor dan Kriteria yang digunakan

No	Nama Faktor	Atribut	Kriteria
1	Jarak	CF	Jarak
2	Status Wilayah	SF	Jarak
3	Usia	CF	Kepribadian Diri
4	Pendidikan	SF	Kepribadian Diri
5	Indeks Prestasi Kumulatif (IPK)	CF	Kepribadian Diri
6	Status Pernikahan	SF	Kepribadian Diri
7	Tes Pengetahuan Umum	SF	Nilai Test
8	Tes Potensi Akademik	SF	Nilai Test
9	Tes Wawasan Kebangsaan	SF	Nilai Test
10	Kemampuan Kosentrasi Bidang	CF	Nilai Test
11	Kemampuan Mengajar	CF	Nilai Test
12	Kemampuan Berkomunikasi Terhadap Siswa	CF	Nilai Test

Sumber: (Hasil Analisis, 2015)

### 5.2.3. Skala Faktor

Data skala faktor merupakan skala kriteria yang dijadikan input ke dalam sistem oleh guru sedangkan *user* Kepala Sekolah menggunakannya sebagai target untuk menentukan kualifikasi guru terbaik bagi sekolahnya. Tiap skala faktor memiliki bobot nilai yang berbeda. Berikut adalah tabel skala dari faktor yang ada.

### 1. Jarak

Jarak merupakan salah satu kriteria yang digunakan dalam perhitungan metode *profile matching* berfungsi untuk menentukan bobot jauh dekat lokasi tempat guru dengan sekolah tujuan. Penjelasan detail diuraikan dalam Tabel 5.3.

Tabel 5.3 Tabel Jarak

Skala Faktor	Bobot
0- 3	5
3,1 - 6	4
6,1 - 9	3
9,1 - 12	2
$\geq 12,1$	1

Sumber: (Hasil Analisis, 2015)

### 2. Status Wilayah

Status wilayah terdiri atas tiga bagian area, yaitu kecamatan sama dengan bobot tertinggi, kecamatan beda dalam kota dan luar kota. Diuraikan dalam Tabel 5.4.

Tabel 5.4 Tabel Status Wilayah

Skala Faktor	Bobot
Kecamatan Sama	3
Kecamatan Beda Dalam Kota	2
Luar Kota	1

Sumber: (Hasil Analisis, 2015)

### 3. Usia

Usia guru yang akan ditempatkan juga termasuk kedalam kriteria faktor dalam metode perhitungan *profile matching*. Semakin muda usia guru maka semakin besar bobot nilai yang dimiliki. Detail faktor dijelaskan pada Tabel 5.5.

Tabel 5.5 Tabel Usia

Skala Faktor	Bobot
18-21	5
22-25	4
26-29	3
30-33	2
34-40	1

Sumber: (Hasil Analisis, 2015)

#### 4. Pendidikan

Pendidikan dibedakan menjadi dua bagian, yaitu strata 1 (S1) dan strata 2 (S2).

Detail dapat dilihat pada Tabel 5.6.

Tabel 5.6 Tabel Pendidikan

Skala Faktor	Bobot
S2	2
S1	1

Sumber: (Hasil Analisis, 2015)

#### 5. IPK (Indeks Prestasi Kumulatif)

Indeks Prestasi Kumulatif (IPK) merupakan bagian dari faktor yang termasuk kedalam *core factor*. IPK digunakan untuk melihat nilai indeks prestasi kumulatif guru saat lulus dari perguruan tinggi yang juga nanti memiliki peranan penting dalam pembobotan pada metode ini. Detail diuraikan dalam Tabel 5.7.

Tabel 5.7 Tabel Indeks Prestasi Kumulatif (IPK)

Skala Faktor	Bobot
3,50 – 4,00	5
3,00 – 3,49	4
2,75 – 2,99	3
2,50 – 2,74	2
0 – 2,49	1

Sumber: (Hasil Analisis, 2015)

#### 6. Status Pernikahan

Status pernikahan dibedakan menjadi dua, yaitu belum menikah dan sudah menikah. Pembagian bobot dijelaskan pada Tabel 5.8.

Tabel 5.8 Tabel Status Pernikahan

Skala Faktor	Bobot
Belum Menikah	2
Sudah Menikah	1

Sumber: (Hasil Analisis, 2015)

### 7. Tes Pengetahuan Umum (TPU)

Tes Pengetahuan Umum berisi indikator nilai yang didapat guru ketika menjalani tes seleksi masuk pegawai negeri sipil (PNS). Dari skala nilai maka dibedakan menjadi beberapa bagian skala faktor. Secara detail dijelaskan pada Tabel 5.9.

Tabel 5.9 Tabel Tes Pengetahuan Umum (TPU)

Skala Faktor	Bobot
80-100	5
70-79	4
60-69	3
50-59	2
0-49	1

Sumber: (Hasil Analisis, 2015)

### 8. Tes Potensi Akademik (TPA)

Tes Potensi Akademik merupakan tes pengetahuan dasar untuk mengetahui potensi dasar kecerdasan seseorang. Hasil tes ini berpengaruh pada hasil perhitungan metode *profile matching*. Tes Potensi Akademik termasuk ke dalam kriteria *secondary factor*. Bobot nilai dijelaskan pada Tabel 5.10.

Tabel 5.10 Tabel Tes Potensi Akademik (TPA)

Skala Faktor	Bobot
80-100	5
70-79	4
60-69	3
50-59	2
0-49	1

Sumber: (Hasil Analisis, 2015)

### 9. Tes Wawasan Kebangsaan

Tes wawasan kebangsaan merupakan tes yang dilakukan untuk mengetahui tingkat pengetahuan seseorang untuk mengenali sejarah, budaya dan ilmu sosial mengenai daerahnya dan secara luas bangsa Indonesia. Tes wawasan kebangsaan termasuk kedalam bagian *secondary factor*.

Tabel 5.11 Tabel Tes Wawasan Kebangsaan

Skala Faktor	Bobot
80-100	5
70-79	4
60-69	3
50-59	2
0-49	1

Sumber: (Hasil Analisis, 2015)

#### 10. Kemampuan Kosentrasi Bidang

Tes kemampuan kosentrasi bidang merupakan tes yang dilakukan untuk mengetahui tingkat kemampuan atau kemahiran seseorang dalam bidangnya.

Pembagian bobot skala faktornya dijelaskan pada Tabel 5.12.

Tabel 5.12 Tabel Kemampuan Kosentrasi Bidang

Skala Faktor	Bobot
Sangat Baik	5
Baik	4
Cukup	3
Kurang	2
Sangat Kurang	1

Sumber: (Hasil Analisis, 2015)

#### 11. Kemampuan Mengajar

Kemampuan mengajar merupakan bagian dari kriteria yang dinilai dalam perhitungan metode *profile matching*. Nilai diambil berdasarkan kemampuan mengajar guru. Bobot nilai dijelaskan pada Tabel 5.13.

Tabel 5.13 Tabel Kemampuan Mengajar

Skala Faktor	Bobot
Sangat Baik	5
Baik	4
Cukup	3
Kurang	2
Sangat Kurang	1

Sumber: (Hasil Analisis, 2015)

## 12. Kemampuan Berkomunikasi Terhadap Siswa

Kemampuan guru dalam berkomunikasi terhadap siswa merupakan salah satu aspek yang sangat penting. Aspek ini termasuk kedalam atribut *core factor* dan bagian dari kriteria nilai tes. Bobot nilai ditampilkan pada Tabel 5.14.

Tabel 5.14 Tabel Kemampuan Berkomunikasi Terhadap Siswa

Skala Faktor	Bobot
Sangat Baik	5
Baik	4
Cukup	3
Kurang	2
Sangat Kurang	1

Sumber: (Hasil Analisis, 2015)

### 5.2.4. Perhitungan Manual

#### 1. Presentase Bobot *Core Factor* dan *Secondary Factor*

Presentase bobot presentase atribut *core factor* dan *secondary factor* dibagi berdasarkan faktor prioritas yang digunakan. Gambar presentase dijelaskan pada Tabel 5.15.

Tabel 5.15 Tabel Presentase Bobot Nilai Atribut *Core Factor* dan *Secondary Factor*

Jenis Atribut	Bobot Presentase
<i>Core factor</i>	60%
<i>Secondary factor</i>	40%

Sumber: (Hasil Analisis, 2015)

Setelah ditentukan presentase bobot nilai atribut *core factor* dan *secondary factor*. User Badan Kepegawain Daerah (BKD) dapat merubah presentase bobot nilai atribut sesuai dengan kebutuhan yang ada. Atribut yang dipakai presentase adalah berjumlah total 100%. Apabila melebihi dari jumlah itu maka sistem akan memberikan alert bahwa jumlah presentase melebihi batasan sistem. Secara detail dijelaskan pada Gambar 5.1

No.	Nama Atribut	Presentase (%)
1	Core Factor	60
2	Secondary Factor	40

Gambar 5.1 Halaman Presentase Bobot Nilai  
Sumbet: (Hasil Analisis, 2015)

a. Presentase Kriteria

Presentase kriteria dibagi menjadi tiga bagian utama, yaitu jarak (30%), Kepribadian diri (30%) dan Nilai Tes (50%). Presentase disesuaikan dengan kebutuhan yang ada. Detail presentase dijelaskan di Tabel 5.16

Tabel 5.16 Tabel Presentase Kriteria

No	Nama Kriteria	Presentase
1	Jarak	20%
2	Kepribadian Diri	30%
3	Nilai Tes	50%

Sumber: (Hasil Analisis, 2015)

b. Atribut dan Faktor

Atribut dan faktor digunakan untuk mengetahui mana yang termasuk kedalam jenis *core factor* dan *secondary factor*. Detail dijelaskan pada Tabel 5.17

Tabel 5.17 Tabel Atribut dan Faktor

Faktor	Atribut	Kriteria
Jarak	CF	Jarak
Status wilayah	SF	Jarak
Usia	CF	Kepribadian diri
Pendidikan	SF	Kepribadian diri
IPK	CF	Kepribadian diri
Status pernikahan	SF	Kepribadian diri
Tes pengetahuan umum	SF	Nilai tes
Tes potensi akademik	SF	Nilai tes

Tes wawasan kebangsaan	SF	Nilai tes
Kemampuan konsentrasi bidang	CF	Nilai tes
Kemampuan mengajar	CF	Nilai tes
Kemampuan komunikasi dengan siswa	CF	Nilai tes

Sumber: (Hasil Analisis, 2015)

c. Tabel Bobot Nilai GAP

Bobot nilai GAP ditentukan berdasarkan selisih yang disesuaikan dengan skala bobot nilai angka. Setiap nilai selisih dilihat berpengaruh pada kompetensi individu. Secara detail dijelaskan pada Tabel 5.18

Tabel 5.18 Tabel Bobot Nilai GAP

Selisih	Bobot Nilai	Keterangan
0	5	Tidak ada selisih (kompetensi sesuai dg yg dibutuhkan)
1	4,5	Kompetensi individu kelebihan 1 tingkat
-1	4	Kompetensi individu kekurangan 1 tingkat
2	3,5	Kompetensi individu kelebihan 2 tingkat
-1	3	Kompetensi individu kekurangan 2 tingkat
3	2,5	Kompetensi individu kelebihan 3 tingkat
-3	2	Kompetensi individu kekurangan 3 tingkat
4	1,5	Kompetensi individu kelebihan 4 tingkat
-4	1	Kompetensi individu kekurangan 4 tingkat

Sumber: (Hasil Analisis, 2015)

d. Perhitungan GAP pada SDN Kebonsari 02

Berikut adalah contoh kasus pada perhitungan nilai GAP pada salah satu guru. Secara detail dijelaskan pada Tabel 5.19.

Tabel 5.19 Tabel Perhitungan GAP

Kriteria	Faktor	Atribut	Nilai	Nilai target	GAP	Bobot GAP
Jarak	Jarak	CF	4	3	1	4,5
	Status wilayah	SF	3	2	1	4,5
Kepribadian Diri	Usia	CF	4	2	2	3,5
	Pendidikan	SF	1	1	0	5
	IPK	CF	4	3	1	4,5

	Status pernikahan	SF	2	1	1	4,5
Nilai Tes	Tes pengetahuan umum	SF	4	5	-1	4
	Tes potensi akademik	SF	4	2	2	3,5
	Tes wawasan kebangsaan	SF	4	3	1	4,5
	Kemampuan konsentrasi bidang	CF	4	3	1	4,5
	Kemampuan mengajar	CF	4	4	0	5
	Kemampuan komunikasi dengan siswa	CF	3	3	0	5

Sumber: (Hasil Analisis, 2015)

e. Tampilan Sistem Hasil Perhitungan

Perhitungan GAP yang dilakukan oleh sistem kemudian ditampilkan pada Gambar 5.2.

<b>8</b>	<b>199203212014121205 Fauzan Imroni S.Pd</b>			
1.SDN KEBONSARI 02				
- Jarak	4	3	1	4.5
- Status Wilayah	3	2	1	4.5
- Usia	4	2	2	3.5
- Pendidikan	1	1	0	5
- IPK	4	3	1	4.5
- Status Pernikahan	2	1	1	4.5
- Test Pengetahuan Umum	4	5	-1	4
- Test Potensi Akademik	4	2	2	3.5
- Test Wawasan Kebangsaan	4	3	1	4.5
- Kemampuan Konsentrasi Bidang	4	3	1	4.5
- Kemampuan Mengajar	4	4	0	5
- Kemampuan Berkomunikasi Terhadap Siswa	3	3	0	5

Gambar 5.2 Halaman Hasil Perhitungan GAP

Sumber: (Hasil Analisis, 2015)

f. Perhitungan *Core Factor* (CF) dan *Secondary Factor* (SF)

## 1. Jarak

$$CF = \frac{4,5}{1} = 4,5$$

$$SF = \frac{4,5}{1} = 4,5$$

## 2. Kepribadian diri

$$CF = \frac{3,5+4,5}{2} = 4$$

$$SF = \frac{5+4,5}{2} = 4,75$$

## 3. Nilai Test

$$CF = \frac{4,5+5+5}{3} = 4,83$$

$$SF = \frac{4+3,5+4,5}{3} = 4$$

g. Tampilan Sistem Hasil Perhitungan *Core Factor* dan *Secondary Factor*.

Hasil perhitungan *core factor* dan *secondary factor* oleh sistem ditampilkan secara detail pada Gambar 5.3

8	199203212014121205	Fauzan Imroni S.Pd
1.SDN KEBONSARI 02		
Jarak	4,5	4,5
Kepribadian Diri	4	4,75
Nilai Test	4,833333333333333	4

Gambar 5.3 Halaman Hasil Perhitungan *Core* dan *Secondary Factor*

Sumber: (Hasil Analisis, 2015)

h. Perhitungan Total

1. Jarak

$$= (4,5 \times 60\%) + (4,5 \times 40\%)$$

$$= 4,5$$

2. Kepribadian diri

$$= (4 \times 60\%) + (4,75 \times 40\%)$$

$$= 4,3$$

3. Nilai Test

$$= (4,83 \times 60\%) + (4,3 \times 40\%)$$

$$= 4,5$$

i. Perhitungan akhir

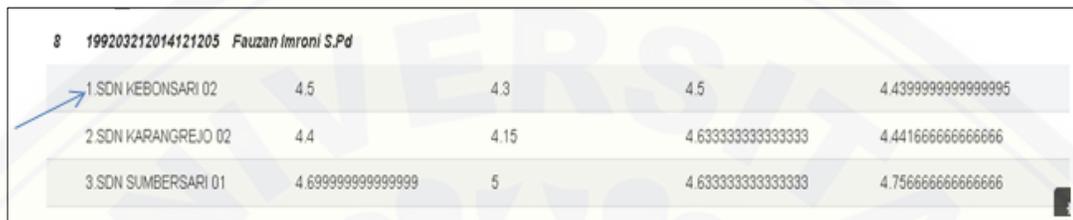
$$= (4,5 \times 20\%) + (4,3 \times 30\%) + (4,5 \times 50\%)$$

$$= 4,439$$

j. Tampilan Sistem Hasil Perhitungan Total dan Perhitungan Akhir

Hasil perhitungan total dan hasil perhitungan akhir merupakan proses perhitungan terakhir dalam metode *profile matching*. Perhitungan total diambil dari penjumlahan total yang didapat dari nilai total jarak, kepribadian diri dan nilai test

yang dikalikan dengan presentase atribut *core* dan *secondary factor*. Perhitungan akhir diperoleh dari hasil akhir dari hasil perhitungan yang diperoleh dari perhitungan total yang kemudian dikalikan presentase dari kriteria yang ada. Gambaran perhitungan total dan perhitungan akhir kemudian ditampilkan pada Gambar 5.4



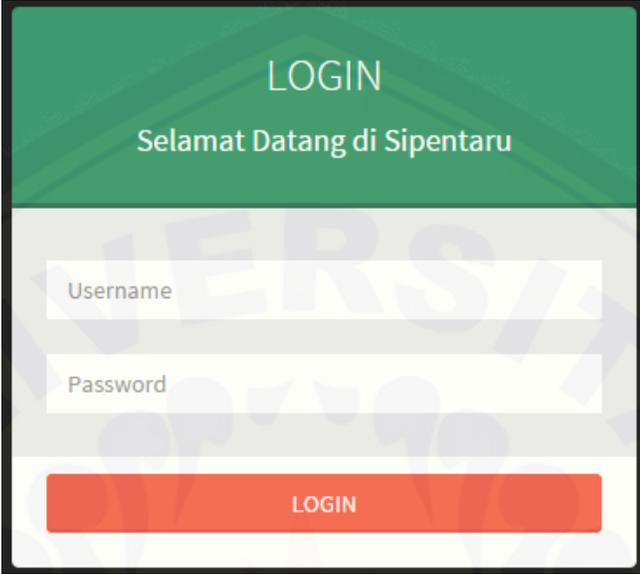
8 199203212014121205 Fauzan Imroni S.Pd				
1.SDN KEBONSARI 02	4.5	4.3	4.5	4.4399999999999995
2.SDN KARANGREJO 02	4.4	4.15	4.633333333333333	4.441666666666666
3.SDN SUMBERSARI 01	4.699999999999999	5	4.633333333333333	4.756666666666666

Gambar 5.4 Halaman Hasil Perhitungan Total dan Perhitungan Akhir  
Sumber: (Hasil Analisis, 2015)

### 5.3. Implementasi Sistem

Tahapan Implementasi Sistem merupakan penerapan dari pengkodean dan perancangan desain sistem yang telah dibuat. Pada proses pengkodean peneliti menggunakan bahasa pemrograman PHP, dalam proses ini menghasilkan tampilan sistem yang telah dibuat. Akses sistem dimulai dengan melakukan *login*. *Login* merupakan akses utama *user* untuk dapat mengakses sistem. *User* melakukan *input username* dan *password* untuk dapat menggunakan hak aksesnya kedalam sistem yang telah dibuat. *User* pada sistem ini dibedakan menjadi empat jenis yaitu, *superadmin*, guru, badan kepegawaian daerah dan kepala sekolah. Hak akses setiap *user* berbeda tergantung pada kebutuhan yang ada. Tampilan *Login* digambarkan pada Gambar 5.5.

### 5.3.1 Login



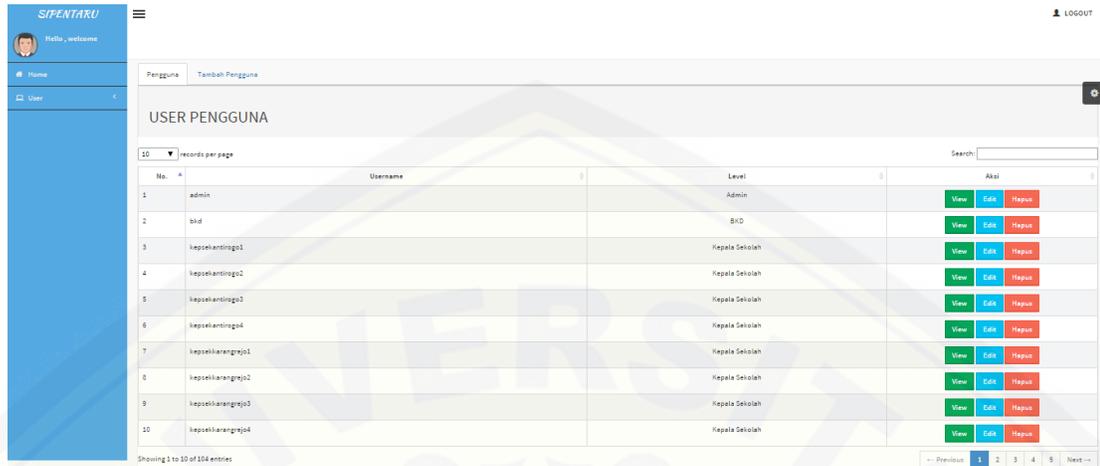
The image shows a login interface with a green header containing the text 'LOGIN' and 'Selamat Datang di Sipentaru'. Below the header are two input fields: 'Username' and 'Password'. At the bottom of the form is a red button labeled 'LOGIN'.

Gambar 5.5 Tampilan *Login*  
Sumber: (Hasil Analisis, 2015)

Untuk dapat mengakses sistem maka *user* harus menginputkan *username* dan *password* yang dimiliki. Apabila *user* salah dalam menginputkan nama *username* dan *password* maka akan muncul peringatan bahwa *username* dan *password* salah. Otoritas dalam hak akses sistem dibedakan menurut kebutuhan yang ada.

### 5.3.2 Halaman User Pengguna

Tampilan menu *user* admin terdiri dari menu dashboard dan menu *user* dengan submenu *user* pengguna. Menu *user* pengguna digunakan untuk melihat *user* pengguna, menambah pengguna, mengubah *user* pengguna dan menghapus *user* pengguna. Tampilan menu data *user* pengguna seperti pada Gambar 5.6.



The screenshot shows the 'USER PENGGUNA' management page. It features a table with columns for 'No.', 'Username', 'Level', and 'Aksi'. The table lists 10 users, including an 'admin' user and several 'Kepala Sekolah' and 'Kepala Sekolah Rangkap' users. Each user row has 'View', 'Edit', and 'Hapus' buttons. The interface includes a sidebar with navigation options like 'Home', 'User', 'Guru', 'Kriteria', 'Sekolah', and 'Metode'. A search bar and pagination controls are also visible.

No.	Username	Level	Aksi
1	admin	Admin	View Edit Hapus
2	bid	BKD	View Edit Hapus
3	kepseleanting01	Kepala Sekolah	View Edit Hapus
4	kepseleanting02	Kepala Sekolah	View Edit Hapus
5	kepseleanting03	Kepala Sekolah	View Edit Hapus
6	kepseleanting04	Kepala Sekolah	View Edit Hapus
7	kepseleanting01	Kepala Sekolah	View Edit Hapus
8	kepseleanting02	Kepala Sekolah	View Edit Hapus
9	kepseleanting03	Kepala Sekolah	View Edit Hapus
10	kepseleanting04	Kepala Sekolah	View Edit Hapus

Gambar 5.6 Tampilan *User* Pengguna  
Sumber: (Hasil Analisis, 2015)

Pada Gambar 5.6 juga ditampilkan username untuk yang dapat mengakses sistem ini. Setiap *user* memiliki *username*, *level*, dan *password* yang berbeda. Setiap user disesuaikan dengan hak akses yang dimilikinya.

### 5.3.3 Fitur *Input* Atribut

Halaman *input* atribut merupakan bagian dari hak akses *user* Badan Kepagawaian Daerah (BKD). Fitur ini berfungsi untuk mengedit nilai presentase atribut yang digunakan dalam perhitungan metode pada sistem. Untuk lebih jelasnya dijelaskan pada Gambar 5.7.



The screenshot shows the 'TABEL ATRIBUT' management page. It features a table with columns for 'No.', 'Nama Atribut', 'Presentase (%)', and 'Aksi'. The table lists two attributes: 'Core Factor' with a 60% presentase and 'Secondary Factor' with a 40% presentase. Each attribute row has an 'Ubah Presentase' button. The interface includes a sidebar with navigation options like 'Home', 'User', 'Guru', 'Kriteria', 'Sekolah', and 'Metode'. A search bar and pagination controls are also visible.

No.	Nama Atribut	Presentase (%)	Aksi
1	Core Factor	60	Ubah Presentase
2	Secondary Factor	40	Ubah Presentase

Gambar 5.7 Tampilan *Input* Atribut  
Sumber: (Hasil Analisis, 2015)

*Input* atribut berfungsi untuk mengubah presentase yang dimiliki oleh *core factor* dan *secondary factor*. Setiap bobot berpengaruh pada besarnya prioritas pada sistem. *Core factor* pada sistem ini memiliki prioritas perhitungan lebih besar karena merupakan bobot utama yang berpengaruh pada perhitungan metode *profile matching*. Sedangkan *secondary factor* memiliki bobot lebih rendah nilainya karena merupakan bobot tambahan perhitungan pada sistem.

#### 5.3.4 Halaman Utama BKD

Halaman Utama Badan Kepegawaian Daerah memiliki beberapa menu yang disesuaikan hak akses yang dimiliki. Menu yang ada diantaranya adalah menu guru kriteria sekolah dan metode. Menu guru memiliki submenu input nilai guru, data guru, data jenis guru dan hasil tes. Menu kriteria memiliki submenu atribut, kriteria dan faktor. Menu sekolah memiliki menu data sekolah. Menu metode memiliki beberapa submenu yaitu hasil perhitungan, peringkat per sekolah dan metode. Detailnya akan dijelaskan pada Gambar 5.8.



Gambar 5.8 Halaman Utama Badan Kepegawaian Daerah (BKD)

Sumber: (Hasil Analisis, 2015)

### 5.3.5 Halaman Utama Kepala Sekolah

Halaman utama kepala sekolah menu guru, menu kriteria dan menu sekolah. Menu guru memiliki sub bab *input* kebutuhan guru. Menu kriteria memiliki sub bab *input* target sekolah. Menu sekolah terdapat sub bab ubah data sekolah. Kepala sekolah memiliki hak akses untuk dapat menginputkan kebutuhan guru, menginputkan target sekolah dan mengubah data sekolah. Halaman utama kepala sekolah ditampilkan pada Gambar 5.9.



Gambar 5.9 Halaman Kepala Sekolah  
Sumber: (Hasil Analisis, 2015)

## 5.4. Pengujian Sistem

Pengujian sistem digunakan untuk mengetahui apakah sistem sudah sesuai dengan kebutuhan yang ada. Pengujian sistem dilakukan dengan dua cara, yaitu menggunakan metode *black box* dan *white box*. Metode *white box* dilakukan dengan membuat *Listing* program pada file *controller* dan model yang diuji, menggambarkan diagram alir dan grafik alir, melakukan perhitungan siklomatik kompleksitas, menentukan *basis set* dan melakukan *test case*. Metode *black box* dilakukan untuk mengetahui kesesuaian antara input dan output yang dijalankan oleh sistem yang dibangun. Dokumentasi *testing white box* lebih lengkapnya terdapat pada lampiran C dan dokumen *testing black box* terdapat pada lampiran D.

## BAB 6. KESIMPULAN DAN SARAN

### 6.1. Kesimpulan

Kesimpulan yang didapatkan oleh peneliti mengenai sistem yang telah dibuat adalah:

1. Metode *profile matching* dapat digunakan untuk membantu pengguna (BKD) dalam menentukan penempatan guru sesuai dengan kebutuhan sekolah.
2. Pada perhitungan metode *profile matching* penempatan guru sekolah dasar negeri di Kabupaten Jember berdasarkan perhitungan dari nilai total guru dan sesuai dengan target yang dibutuhkan sekolah.
3. Sistem Informasi Penentuan Penempatan Guru Sekolah Dasar (SIPENTARU) yang sudah dikembangkan dapat menjadi solusi rekomendasi kepada pengambil keputusan sesuai dengan kriteria yang ada dan memenuhi target yang dibutuhkan sekolah.

### 6.2. Saran

1. Untuk lebih meningkatkan hasil yang lebih akurat dan maksimal, Sistem Pendukung Keputusan Penempatan Guru Sekolah Dasar Negeri di Kabupaten Jember dapat dikombinasikan dengan metode perhitungan lainnya.
2. Pengembangan sistem selanjutnya diharapkan dapat menambahkan kriteria lain yang dibutuhkan.
3. Karena keterbatasan pada hasil tampilan peta oleh *GoogleMap*, maka untuk pengembangan sistem selanjutnya dapat menggunakan jenis peta yang lain misalnya *BingMap* atau *Leaflet*.

**DAFTAR PUSTAKA**

- Enjang, R. B. (2012). *Modul Pemrograman Web HTML, PHP & MySQL*. Bandung: Modula.
- Hasibuan, Malayu S.P. (2005). *Manajemen Sumber Daya Manusia*. Jakarta: Bumi Aksara.
- Ibnu, D. (02, Februari, 2011). *Framework Codeigniter*. dikutip Juni 28, 2014, dari Framework Codeigniter: Pekanbaru: <http://ellislab.com/codeigniter>.
- Iqbal & S. Hartati. (2011). *Aplikasi Sistem Pendukung Keputusan Penempatan Bidan PTT (Pegawai Tidak Tetap) pada Kabupaten Bireun*. Seminar Nasional Ilmu Komputer. Gama.
- Kusrini. 2007. *Konsep dan Aplikasi Sistem Pendukung Keputusan*. Yogyakarta: Andi.
- Mathis, L.R dan J.H Jackson. (2006). *Manajemen Sumber Daya Manusia (Human Resource Management)*. Jakarta: Salemba Empat.
- Mega, C.(12, Maret, 2012). *megaboy16.blogspot.com*. Dikutip 18, Agustus, 2014 dari [megaboy16.blogspot.com:http://megaboy16.blogspot.com/2011/03/model-incremental.html](http://megaboy16.blogspot.com/2011/03/model-incremental.html)
- Nur Fadhila, A, Hakim, Abdul, Siswidiyanto. *Pengaruh Penempatan Pegawai Terhadap Kinerja Pegawai*. *Jurnal Administrasi Publik (JAP)*, Vol 1. No.5. Malang: Universitas Brawijaya.
- Proboyekti, S.M (1 Agustus 2008). *Software Process Model I*. Dikutip 25 Februari 2014 dari *Software Process Model I*: <http://www.scribd.com/doc/192850316/Software-Process>.

- Republik Indonesia. 1999. Undang-Undang Nomor 43 Tahun 1999 tentang pokok kepegawaian. Lembaran Negara RI Tahun 1999. Sekretariat Negara. Jakarta.
- Republik Indonesia. 2005. Undang-Undang Nomor 14 Tahun 2005 tentang Guru dan Dosen. Lembaran Negara RI Tahun 2005. Sekretariat Negara. Jakarta.
- Republik Indonesia. 2008. Peraturan Pemerintah Republik Indonesia Nomor 74 Tahun 2008 Bab IV Tentang Pengangkatan, Penempatan, dan Pemindahan Pasal 61 ayat 1. Lembaran Negara Tahun 2008. Sekretariat Negara. Jakarta.
- Sastrohadiwiryono, Siswanto, B. (2002). Manajemen Tenaga Kerja Indonesia. Jakarta: Bumi Aksara.
- Setyaningsih, W. (28 April 2013). Decision Support System menggunakan metode *profile matching*. Dikutip dari [http:// www.slideserve.com/decision-support-system-menggunakan-metode-profile-matching](http://www.slideserve.com/decision-support-system-menggunakan-metode-profile-matching).
- Subakti, Irfan. 2002. *Sistem Penunjang Keputusan*. Surabaya: Institut Teknologi 10 Nopember.

**LAMPIRAN A. PERANCANGAN SISTEM****A.1 Perancangan Modul I Manajemen Data Pengguna**

## a. Usecase Scenario

## 1. Input Data Pengguna

Nama	Menambah Data Pengguna		
Aktor	Admin		
Entry Condition	Aktor menambahkan data pengguna		
Exit Condition	Data pengguna berhasil ditambahkan		
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Memilih menu data pengguna		
		2	Menampilkan halaman data pengguna
3	Memilih menu tambah data pengguna		
		4	Menampilkan form tambah data pengguna
5	Mengisi data pada form tambah data pengguna		
6	Klik Tambah		
		7	Data berhasil ditambahkan
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
5	Mengisi data tidak lengkap atau username sudah digunakan		
6	Klik Tambah		
		7	Menampilkan alert pada form yang salah

## 2. Hapus Data Pengguna

Nama	Menghapus Data Pengguna		
Aktor	Admin		
Entry Condition	Aktor menghapus data pengguna		
Exit Condition	Data pengguna berhasil dihapus		
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Memilih menu data pengguna		
		2	Menampilkan halaman data pengguna
3	Memilih data yang akan diedit		
4	Klik Hapus		
		5	Menampilkan alert apakah data akan dihapus
6	Klik OK		
		7	Data berhasil dihapus
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
6	Klik Cancel		
		7	Kembali ke halaman data pengguna dan data tidak dihapus

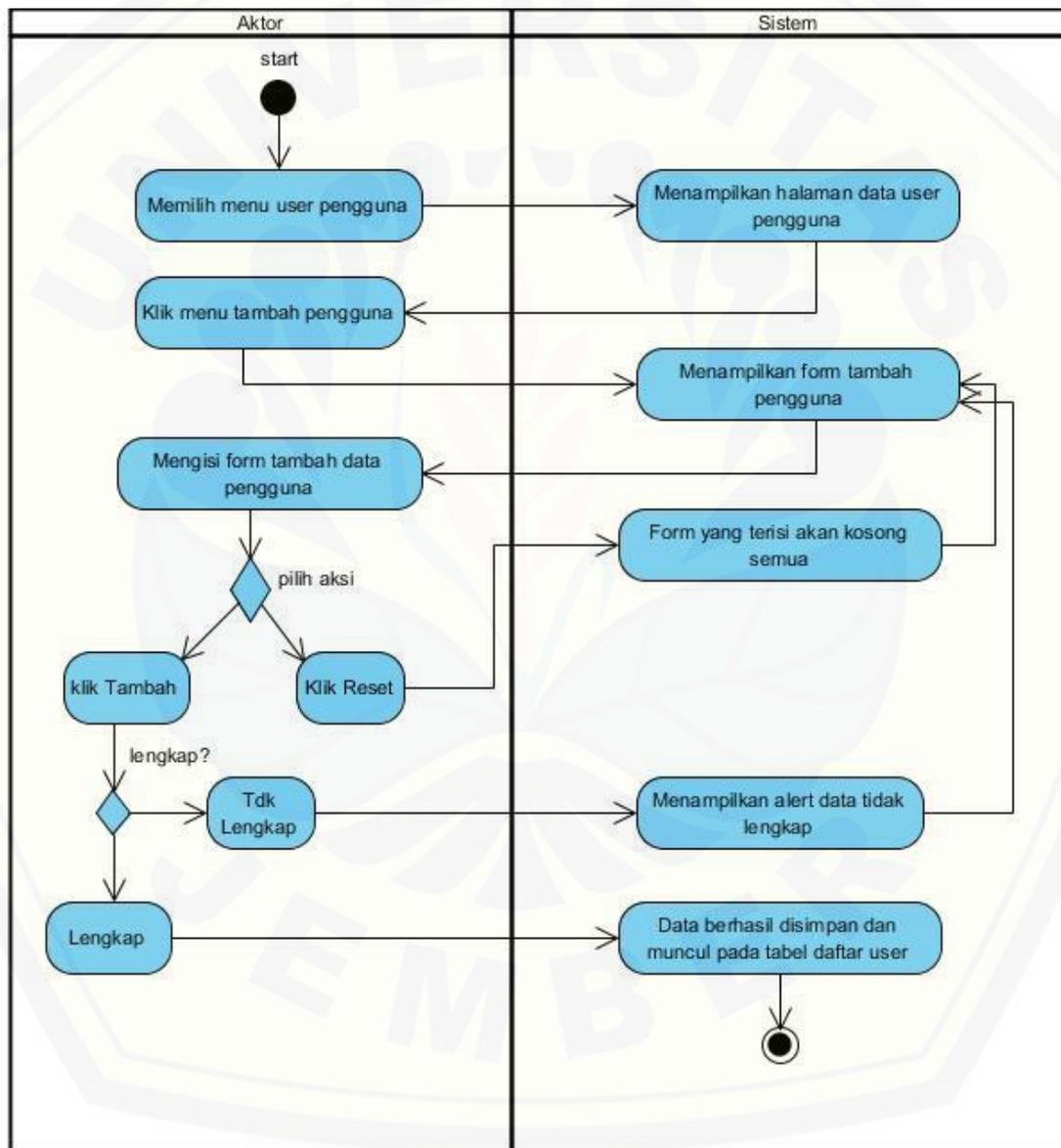
## 3. Mengubah data pengguna

Nama	Mengubah Data Pengguna		
Aktor	Admin		
Entry Condition	Aktor mengubah data pengguna		
Exit Condition	Data pengguna berhasil diubah		
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Memilih menu data pengguna		
		2	Menampilkan halaman data pengguna
3	Memilih menu edit		
		4	Menampilkan edit data pengguna
5	Mengubah data pengguna		
6	Klik OK		
		7	Data berhasil diubah
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
6	Klik Kembali		
		7	Kembali pada halaman data user pengguna
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-2</b>			
5	Data yang diubah tidak lengkap		
		7	Menampilkan alert pada form yang salah

b. *Activity Diagram* Manajemen Data Pengguna

1. *Input Data* Pengguna

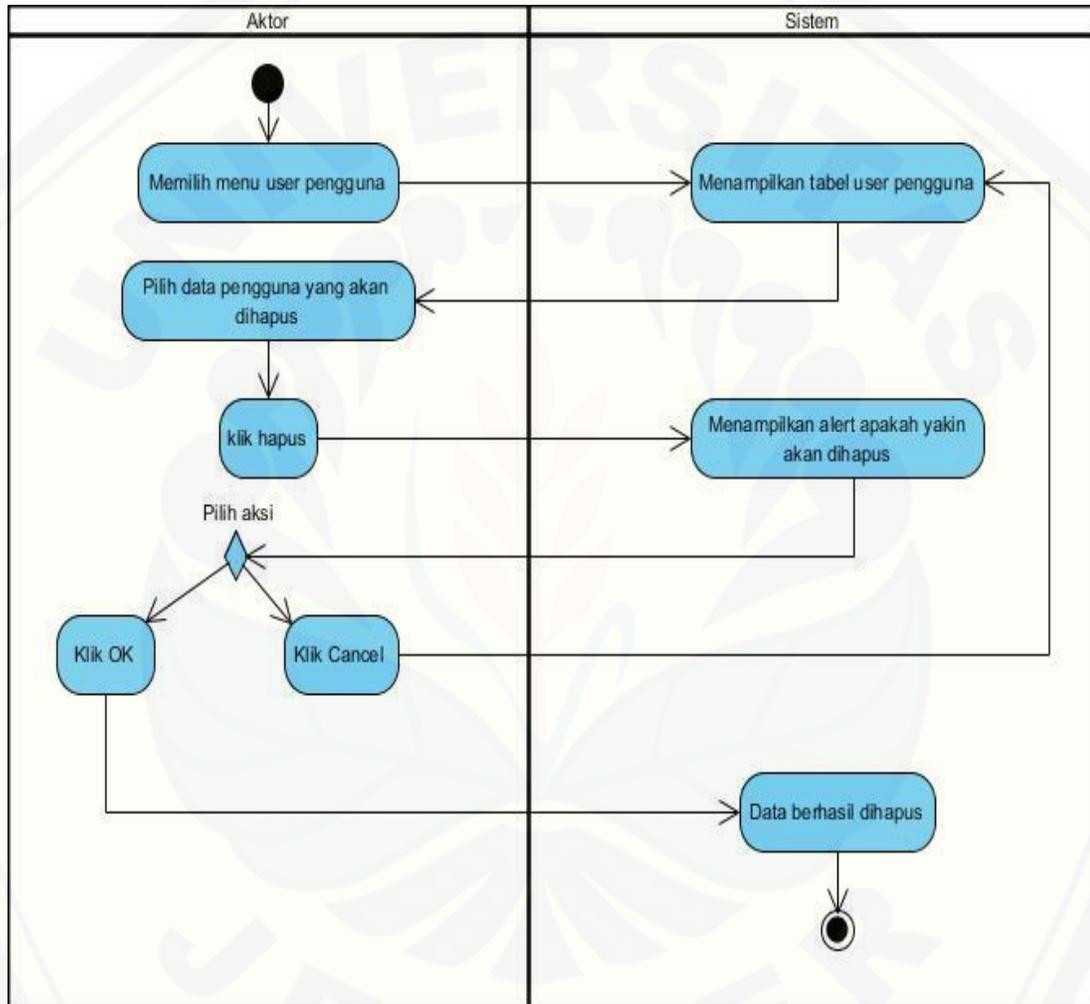
*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur *input data* pengguna. Adapun penjelasan mengenai *Activity diagram* ini digambarkan pada Gambar 1



Gambar 1. *Activity Diagram* Input Data Pengguna

## 2. Hapus Data pengguna

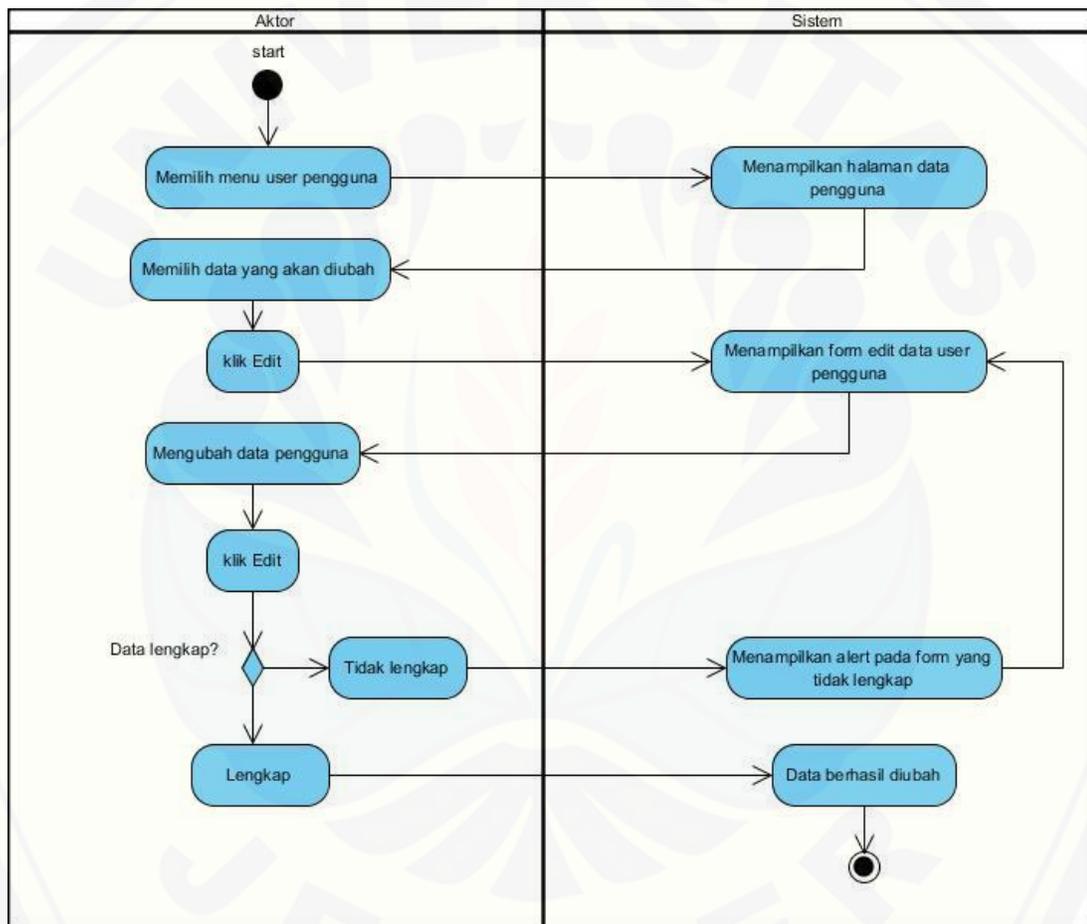
*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur hapus data pengguna. Adapun penjelasan mengenai *Activity diagram* ini digambarkan pada Gambar 2



Gambar 2. *Activity Diagram* Hapus Data Pengguna

3. Mengubah Data Pengguna

*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur mengubah data pengguna. Adapun penjelasan mengenai *Activity diagram* ini digambarkan pada Gambar 3

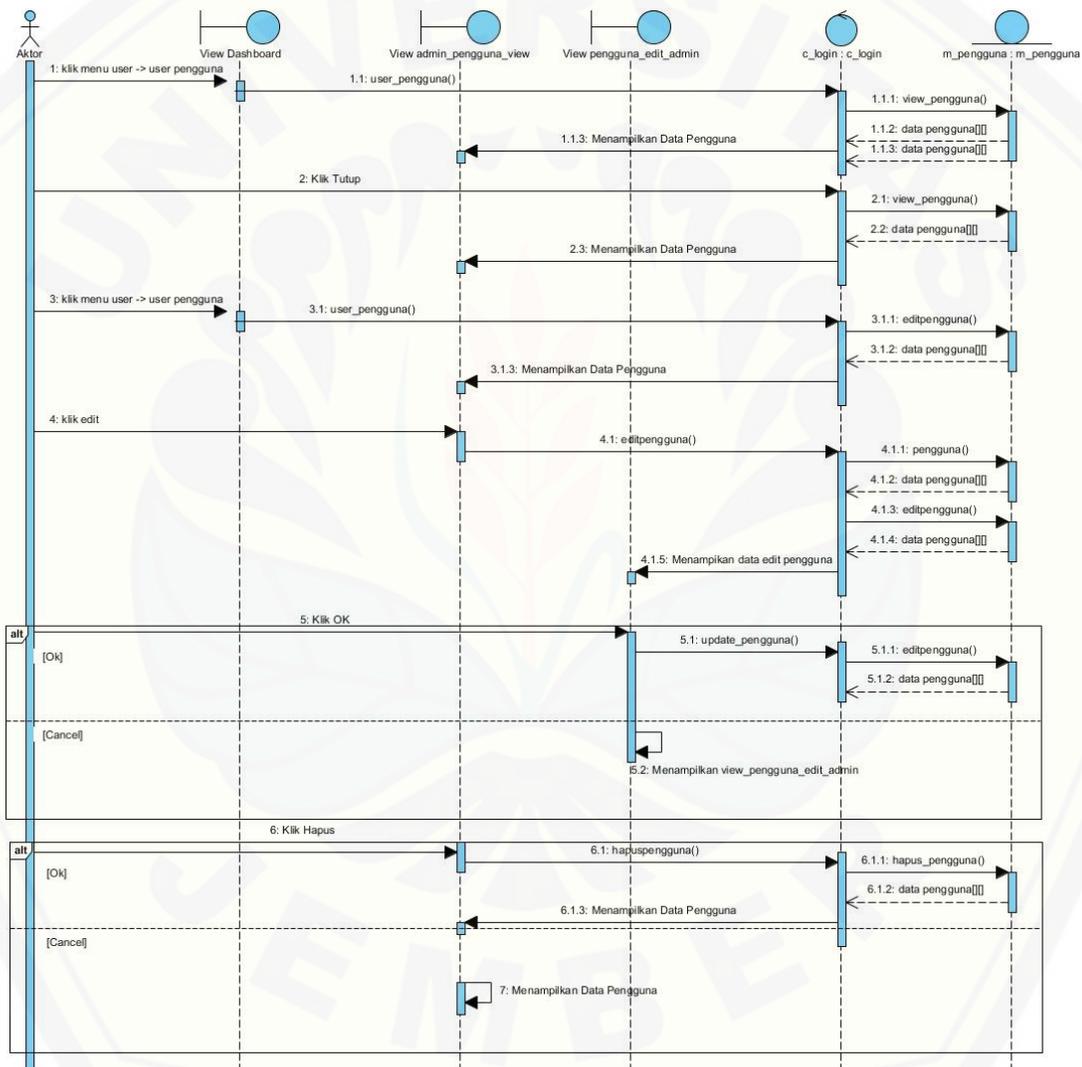


Gambar 3. *Activity Diagram* Mengubah Data Pengguna

c. Sequence Diagram Manajemen Data Pengguna

1. Sequence Diagram Data Pengguna

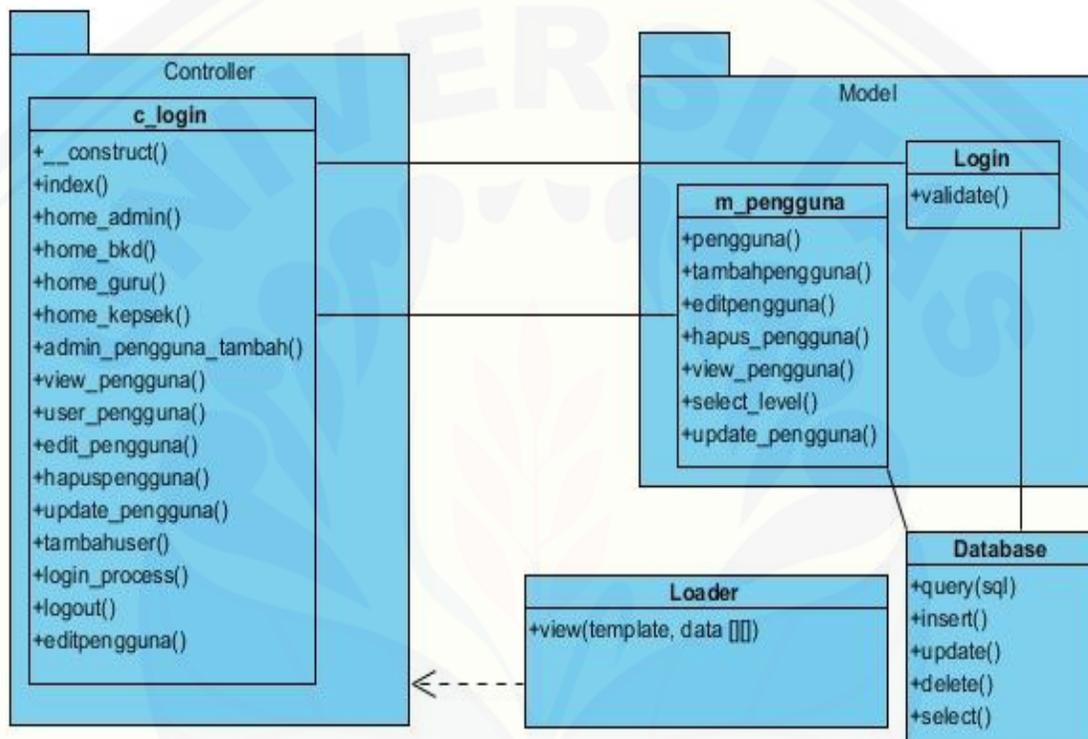
Sequence diagram manajemen data pengguna digunakan untuk memodelkan aliran logika pada fitur manajemen data pengguna dan akan dijelaskan pada Gambar 4.



Gambar 4. Sequence Diagram Manajemen Data Pengguna

## d. Class Diagram

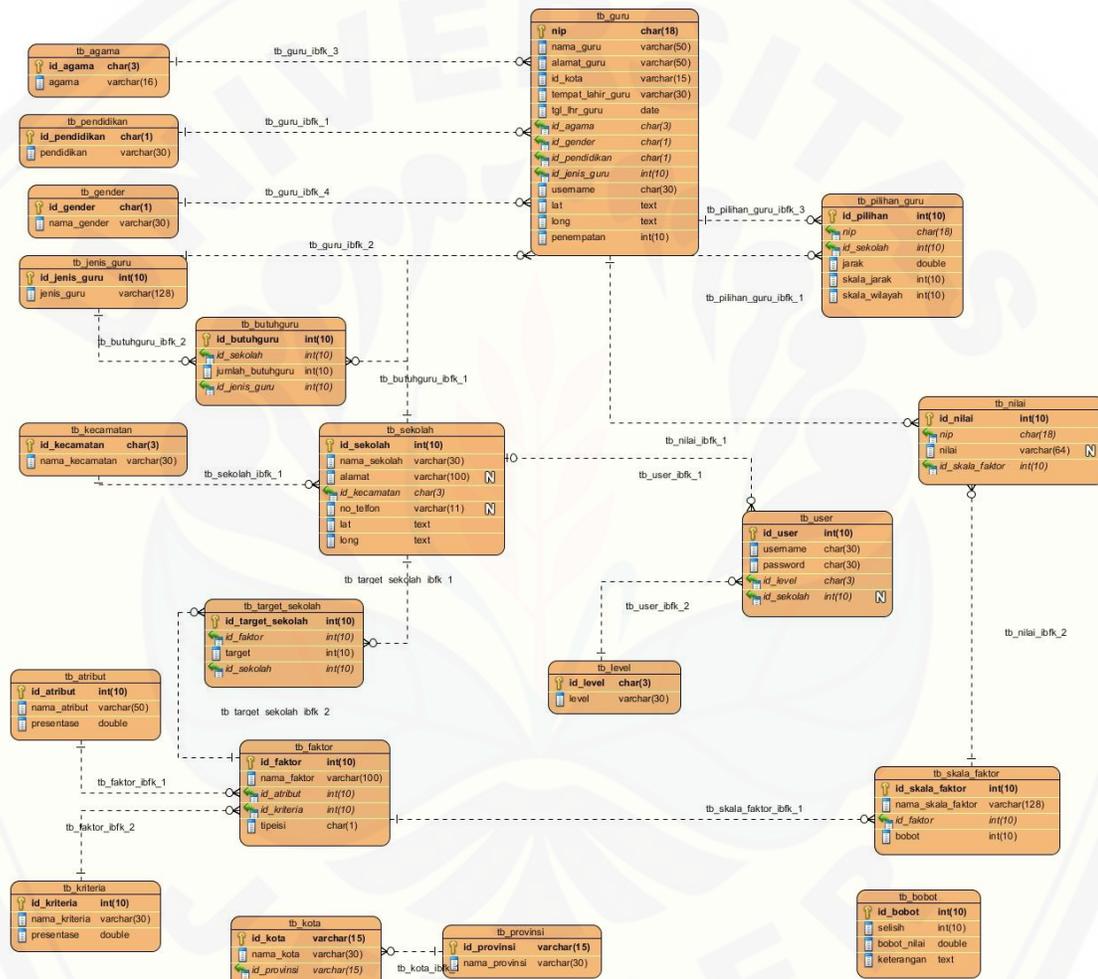
Gambar 5 merupakan gambaran dari *class diagram* implementasi metode yang menghasilkan beberapa jenis operasi yang digunakan dalam alur kerja sistem. Gambaran dari *class diagram* manajemen data pengguna dijelaskan pada Gambar 5.



Gambar 5. Class Diagram Manajemen Data Pengguna

e. ERD

Entity Relationship Diagram modul implementasi metode berisi gambaran relasi data yang terhubung dalam sistem dan database. Gambaran Entity Relationship Diagram dijelaskan pada Gambar 6



Gambar 6. Entity Relationship Diagram Manajemen Data Pengguna.

## A.2 Perancangan Modul II Manajemen Data Sekolah

### a. Usecase Scenario

#### 1. Tambah sekolah

Nama	Menambah data sekolah		
Aktor	BKD		
Entry Condition	Aktor menambah data sekolah		
Exit Condition	Data sekolah berhasil ditambahkan		
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Memilih menu sekolah		
2	Pilih submenu Data sekolah		
		3	Menampilkan halaman tabel sekolah dan form tambah sekolah
4	Masukkan data sekolah		
5	Klik Submit		
		6	Data sekolah berhasil ditambahkan dan muncul pada tabel kriteria
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
4	Memasukkan data sekolah kosong		
5	Klik Submit		
		6	Muncul alert pada form pengisian dan data tidak berhasil ditambahkan

## 2. Input nilai target sekolah

Nama		Input nilai target sekolah	
Aktor		Kepala Sekolah	
Entry Condition		Aktor menambahkan nilai target	
Exit Condition		Nilai target berhasil ditambahkan	
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Pilih menu kriteria		
2	Pilih submenu beri target sekolah		
		3	Menampilkan tabel factor yang akan dimasukkan nilai target
4	Pilih factor yang akan ditambahkan nilai target		
5	Klik Input target		
		6	Menampilkan halaman input skala factor
7	Mengisi nilai target factor		
8	Klik Submit		
		9	Data berhasil ditambahkan
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
7	Mengisi nilai kosong atau tidak lengkap		
8	Klik Submit		
		9	Menampilkan alert pada form yang belum lengkap

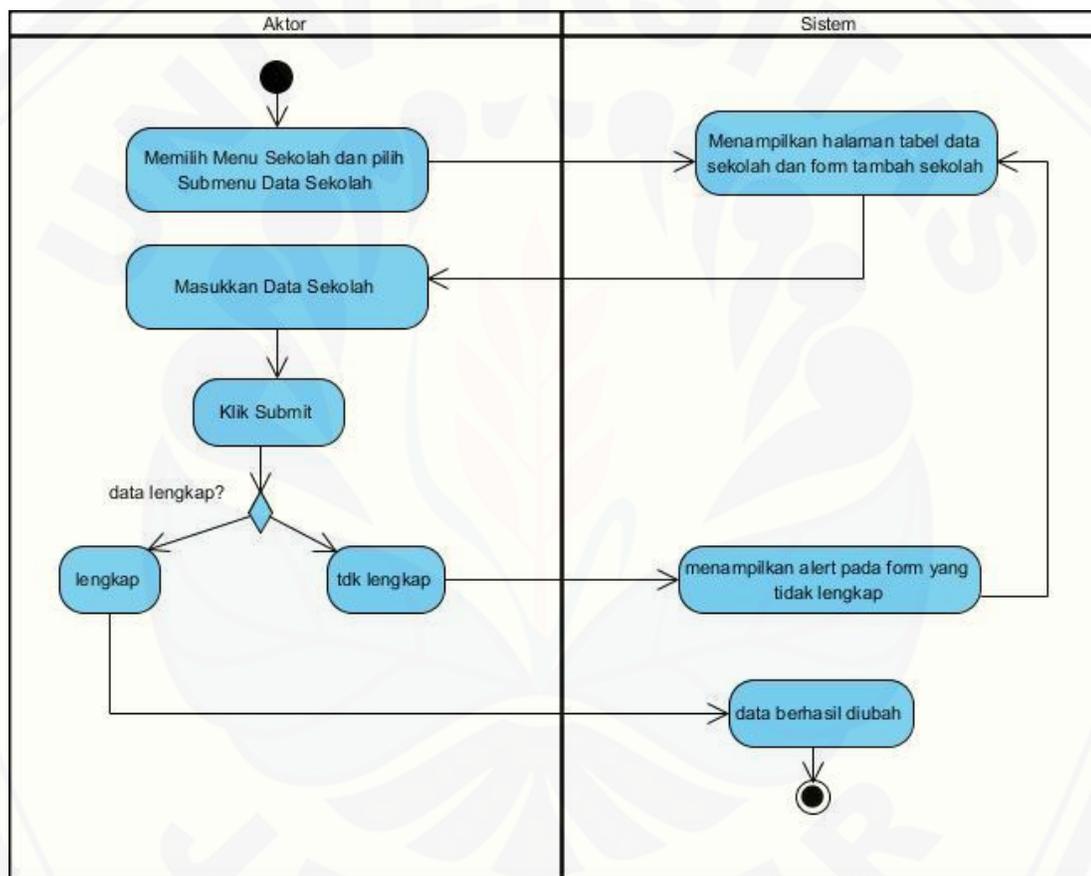
## 3. Ubah data sekolah

Nama		Ubah data sekolah	
Aktor		Kepala Sekolah	
Entry Condition		Aktor merubah data sekolah	
Exit Condition		Data sekolah berhasil diubah	
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Pilih menu sekolah		
2	Pilih submenu ubah data sekolah		
		3	Menampilkan form ubah data sekolah
4	Merubah data sekolah sesuai dengan kebutuhan		
5	Klik Submit		
		6	Data sekolah berhasil diubah
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
4	Data yang diubah tidak lengkap		
5	Klik Submit		
		6	Menampilkan alert pada form yang tidak lengkap

b. Activity Diagram Manajemen Data Sekolah

1. Tambah sekolah

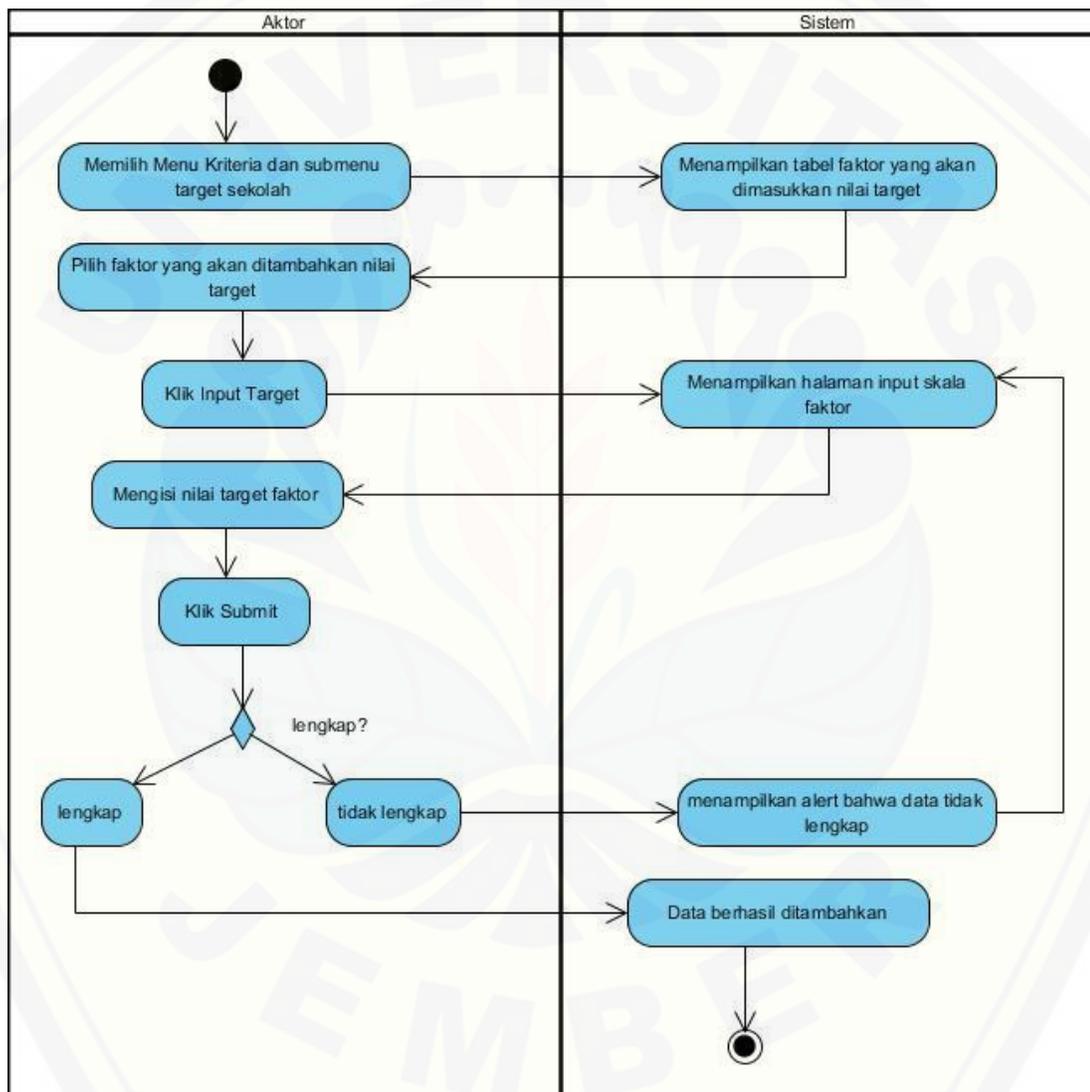
*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur tambah sekolah. Adapun penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 7



Gambar 7. Activity Diagram Tambah Sekolah

## 2. Input nilai target sekolah

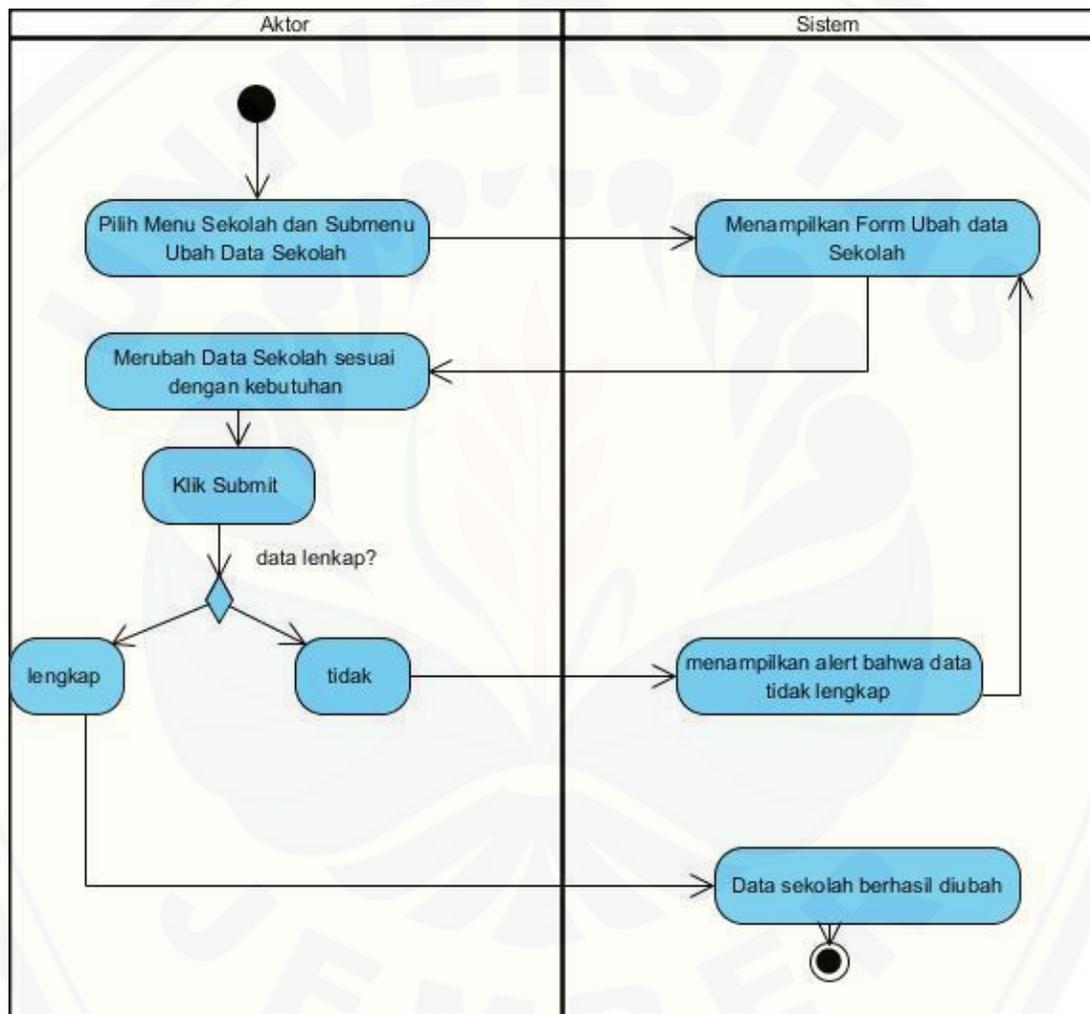
*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur *input* nilai target sekolah. Adapun penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 8



Gambar 8. *Activity Diagram Input Nilai Target Sekolah*

### 3. Ubah data sekolah

*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur ubah data sekolah. Adapun penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 9

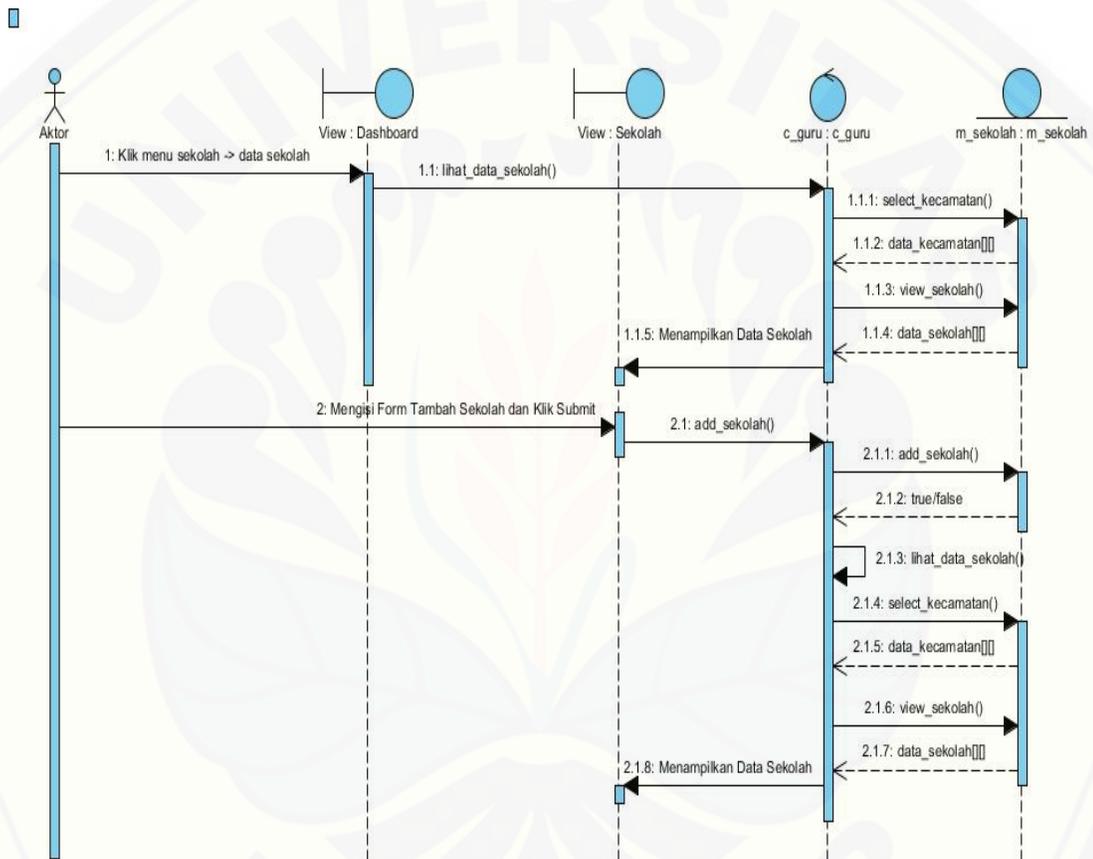


Gambar 9. *Activity Diagram* Ubah Data Sekolah

c. Sequence Diagram

1. Tambah sekolah

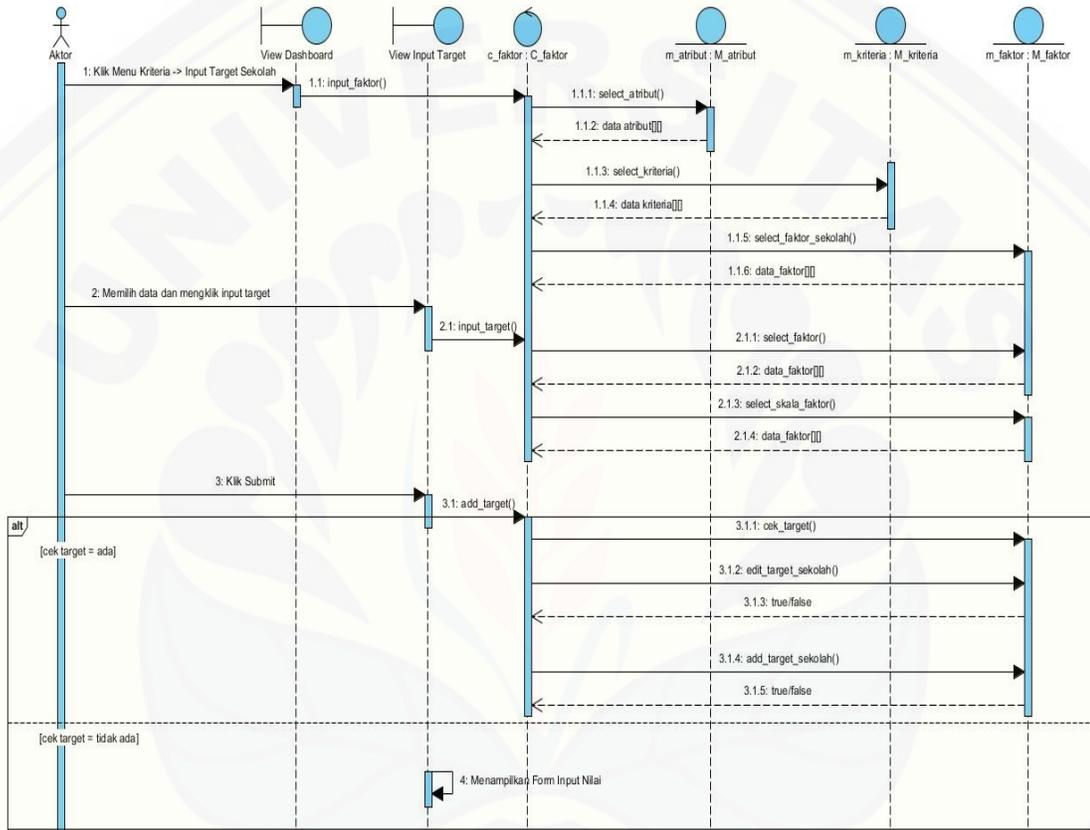
*Sequence diagram* pada fitur tambah sekolah digunakan untuk memodelkan aliran logika pada sistem yang dibuat. Penggambaran *sequence diagram* dijelaskan pada Gambar 10.



Gambar 10. *Sequence Diagram* Tambah Sekolah

2. Input nilai target sekolah

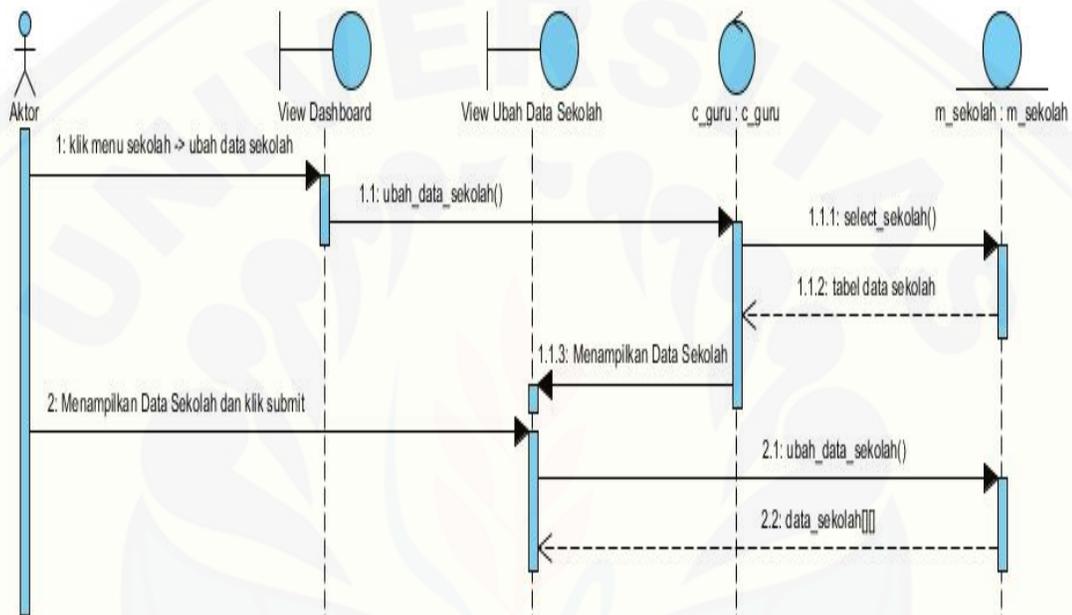
*Sequence diagram* pada fitur *input* nilai target sekolah digunakan untuk memodelkan aliran logika pada sistem yang dibuat. Penggambaran *sequence diagram* dijelaskan pada Gambar 11.



Gambar 11. *Sequence Diagram* Input Nilai Target Sekolah

### 3. Ubah data sekolah

*Sequence diagram* pada fitur ubah data sekolah digunakan untuk memodelkan aliran logika pada sistem yang dibuat. Penggambaran *sequence diagram* dijelaskan pada Gambar 12.



Gambar 12. *Sequence Diagram* Ubah Data Sekolah

### d. Class Diagram

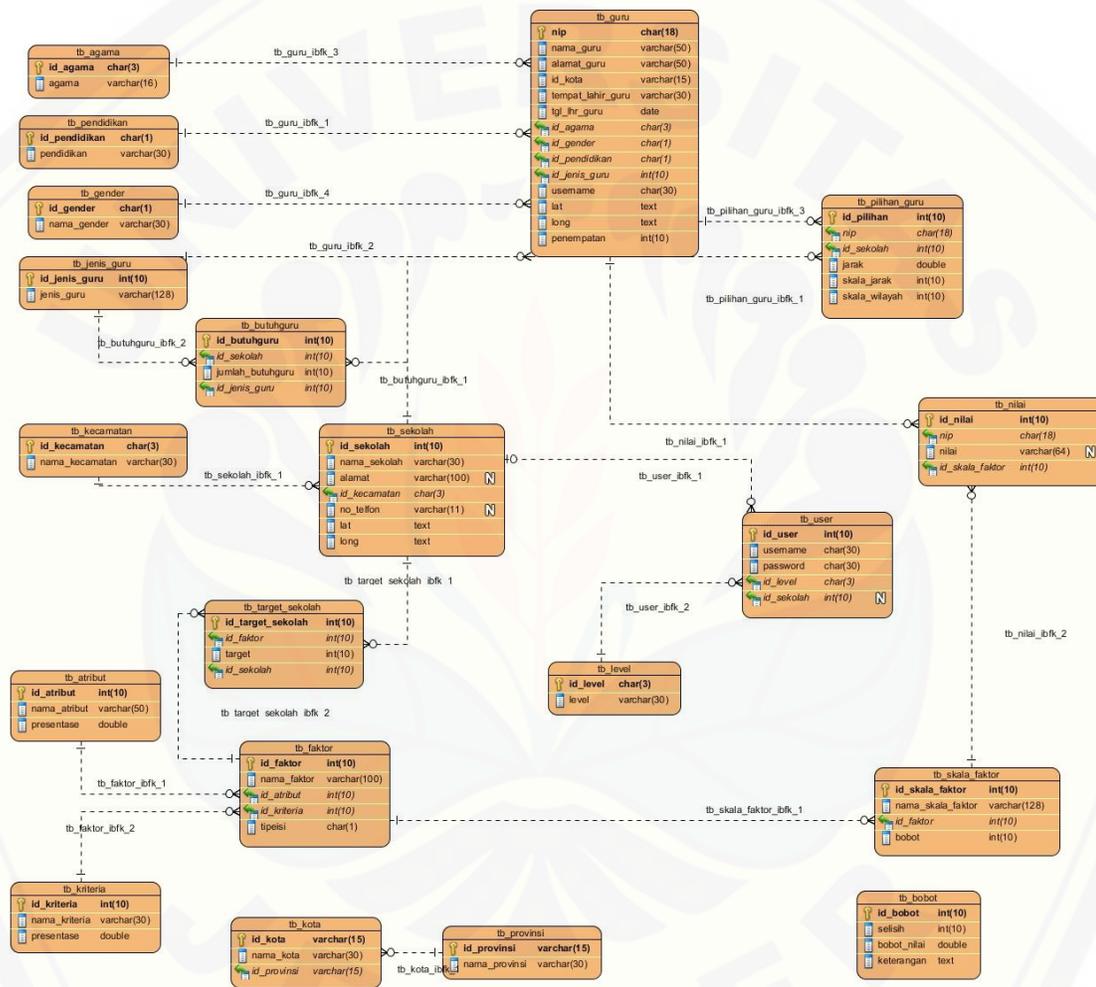
Gambar 13 merupakan gambaran dari *class diagram* manajemen data sekolah yang menghasilkan beberapa jenis operasi yang digunakan dalam alur kerja sistem. Gambaran dari *class diagram* manajemen data pengguna dijelaskan pada Gambar 13.



Gambar 13. Class Diagram Manajemen Data Sekolah

e. Entity Relationship Diagram

Entity Relationship Diagram modul manajemen data sekolah berisi gambaran relasi data yang terhubung dalam sistem dan database. Gambaran Entity Relationship Diagram dijelaskan pada Gambar 14.



Gambar 14. Entity Relationship Diagram Manajemen Data Sekolah

### A.3 Perancangan Modul IV Manajemen Data Guru

#### a. Usecase Scenario

##### 1. Input jenis guru

Nama	Input Jenis Guru		
Aktor	BKD		
Entry Condition	Aktor memasukkan jenis guru		
Exit Condition	Data jenis guru berhasil ditambahkan		
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Memilih menu Guru		
2	Memilih submenu data jenis guru		
		3	Menampilkan form tambah jenis guru dan tabel jenis guru
4	Mengisi data jenis guru		
5	Klik Submit		
		6	Data jenis guru berhasil ditambahkan
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
4	Data yang diisi kosong atau tidak lengkap		
5	Klik Submit		
		6	Menampilkan alert pda form yang kosong atau tidak lengkap

## 2. Hapus jenis guru

Nama	Menghapus jenis guru
Aktor	BKD
Entry Condition	Aktor menghapus data jenis guru
Exit Condition	Data jenis guru berhasil dihapus
<b>SKENARIO NORMAL SN-UC-1</b>	
1	Memilih menu guru
2	Memilih submenu data jenis guru
	3 Menampilkan form tambah jenis guru dan tabel jenis guru
4	Memilih data yang akan dihapus
5	Klik Hapus
	6 Menampilkan alert apakah data akan dihapus
7	Klik OK
	8 Data berhasil dihapus
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>	
6	Klik Cancel
	7 Kembali ke halaman data jenis guru dan data tidak dihapus

## 3. Input nilai guru

Nama	Input Nilai Guru		
Aktor	BKD		
Entry Condition	Aktor memasukkan nilai guru		
Exit Condition	Data nilai berhasil ditambahkan		
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Memilih menu Input Nilai Guru		
		2	Menampilkan halaman data guru
3	Klik menu Input		
		4	Menampilkan data detail guru dan form input nilai
5	Memasukkan nilai guru pada form data guru		
6	Klik Simpan		
		7	Data berhasil ditambahkan
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
5	Data yang dimasukkan tidak lengkap		
6	Klik Simpan		
		7	Menampilkan alert pda form yang tidak lengkap

## 4. Hapus nilai guru

Nama	Hapus Nilai Guru		
Aktor	BKD		
Entry Condition	Aktor menghapus nilai guru		
Exit Condition	Data nilai berhasil dihapus		
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Memilih menu Input Nilai Guru		
		2	Menampilkan halaman data guru
3	Klik menu Hapus		
		4	Menampilkan alert apakah yakin akan menghapus
5	Klik OK		
		6	Data berhasil dihapus
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
5	Klik Cancel		
		6	Kembali ke halaman sebelumnya dan tidak terhapus

## 5. Lihat data guru

Nama	Lihat Data Guru		
Aktor	BKD		
Entry Condition	Aktor melihat Data Guru		
Exit Condition	Data nilai berhasil dilihat		
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Memilih menu Data Guru		
		2	Menampilkan halaman data guru
3	Klik menu view pada data yang ingin		

dilihat		
	4	Menampilkan data guru secara detail

## 6. Hapus data guru

Nama	Hapus Data Guru	
Aktor	BKD	
Entry Condition	Aktor menghapus Data Guru	
Exit Condition	Data nilai berhasil dihapus	
<b>SKENARIO NORMAL SN-UC-1</b>		
1	Memilih menu Data Guru	
		2 Menampilkan halaman data guru
3	Pilih data yang akan dihapus dan klik hapus	
		4 Menampilkan alert apakah yakin akan menghapus data yang dipilih
5	Klik OK	
		6 Data berhasil dihapus
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>		
5	Klik Cancel	
		6 Kembali ke halaman semula dan data tidak dihapus

## 7. Input kebutuhan guru

Nama		Penempatan guru	
Aktor		Kepala Sekolah	
Entry Condition		Aktor melakukan input kebutuhan guru	
Exit Condition		Data kebutuhan guru berhasil ditambahkan	
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Memilih menu guru		
2	Pilih submenu input kebutuhan guru		
		3	Menampilkan form tambah kebutuhan guru dan tabel kebutuhan guru
4	Mengisi data jenis guru dan tambah guru		
5	Klik Submit		
		6	Data kebutuhan guru berhasil ditambahkan
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
4	Mengisi data tidak lengkap		
5	Klik Sumbit		
		6	Menampilkan alert pada form yang tidak lengkap

## 8. Edit biodata guru

Nama		Edit Data Guru	
Aktor		Guru	
Entry Condition		Aktor mengubah data guru	
Exit Condition		Data guru berhasil diubah	
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Pilih menu guru		
2	Pilih submenu input data guru		
		3	Menampilkan form ubah data guru
4	Merubah data guru sesuai dengan identitas guru		
5	Klik Submit		
		6	Data sekolah berhasil diubah
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
4	Data yang diubah tidak lengkap		
5	Klik Submit		
		6	Menampilkan alert pada form yang tidak lengkap

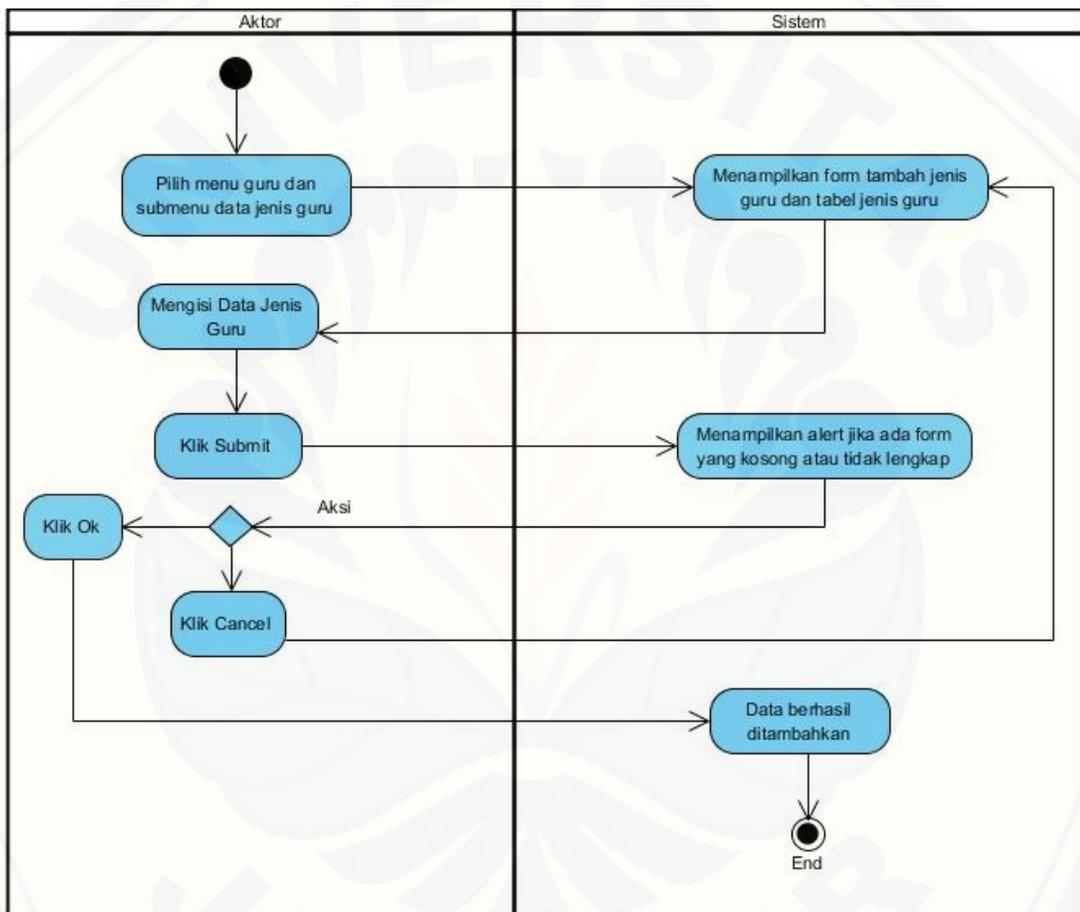
## 9. Pemilihan sekolah

Nama		Pemilihan sekolah	
Aktor		Guru	
Entry Condition		Aktor memilih sekolah	
Exit Condition		Data sekolah berhasil dimasukkan	
<b>SKENARIO NORMAL SN-UC-1</b>			
1	Pilih menu sekolah		
2	Pilih submenu pilih sekolah tujuan		
		3	Menampilkan form pilihan data sekolah
4	Memilih sekolah yang akan dituju		
5	Klik Tambah		
		6	Data sekolah berhasil ditambahkan dan muncul pada tabel sekolah pilihan
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-1</b>			
4	Memilih sekolah yang akan dituju		
5	Klik Reset		
		6	Data yang diisikan akan terhapus
<b>SKENARIO ALTERNATIVE SA-Usecase Scenario-2</b>			
4	Memilih lebih dari 3 sekolah		
5	Klik Tambah		
		6	Menampilkan alert bahwa tidak boleh lebih dari 3 sekolah

b. Activity Diagram

1. Input jenis guru

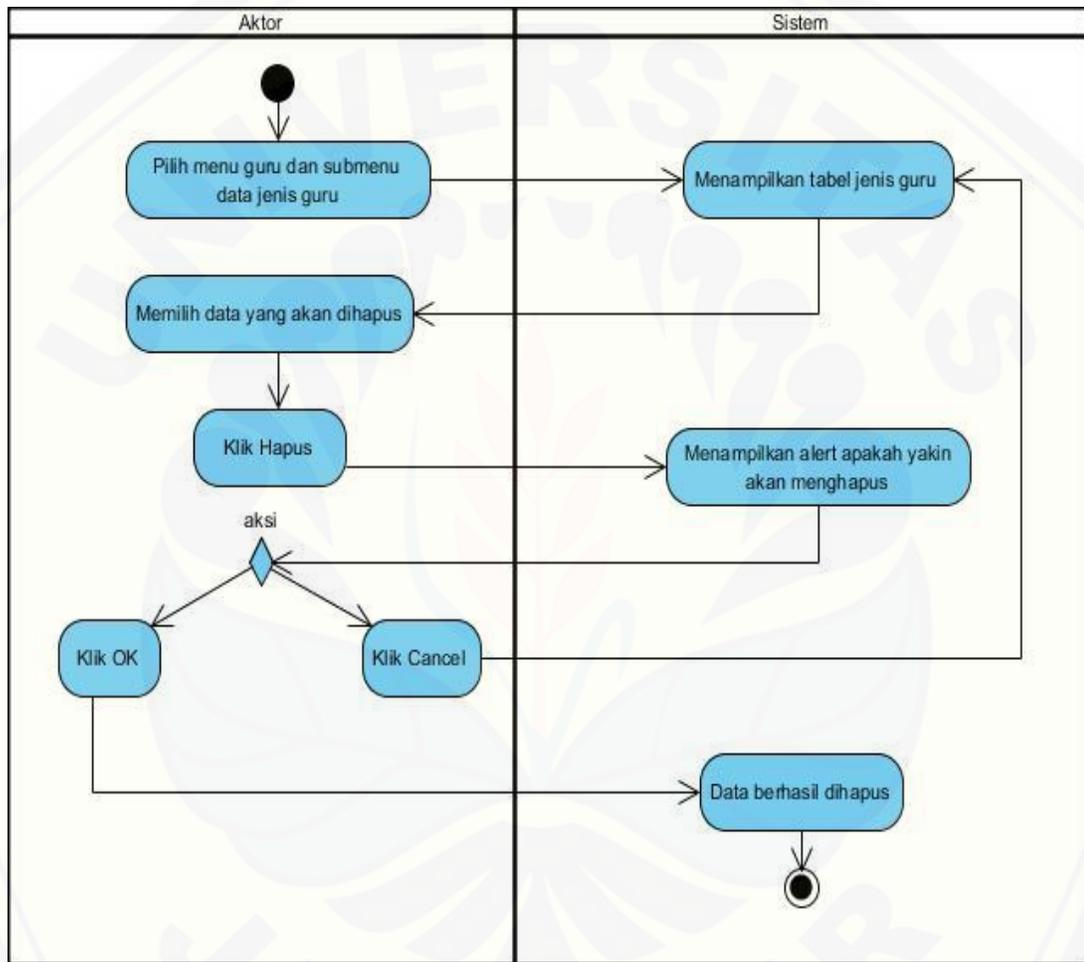
Activity diagram menjelaskan aliran aktivitas dalam sistem informasi pada fitur *input* jenis guru. Penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 15



Gambar 15. Activity Diagram Input Jenis Guru

## 2. Hapus jenis guru

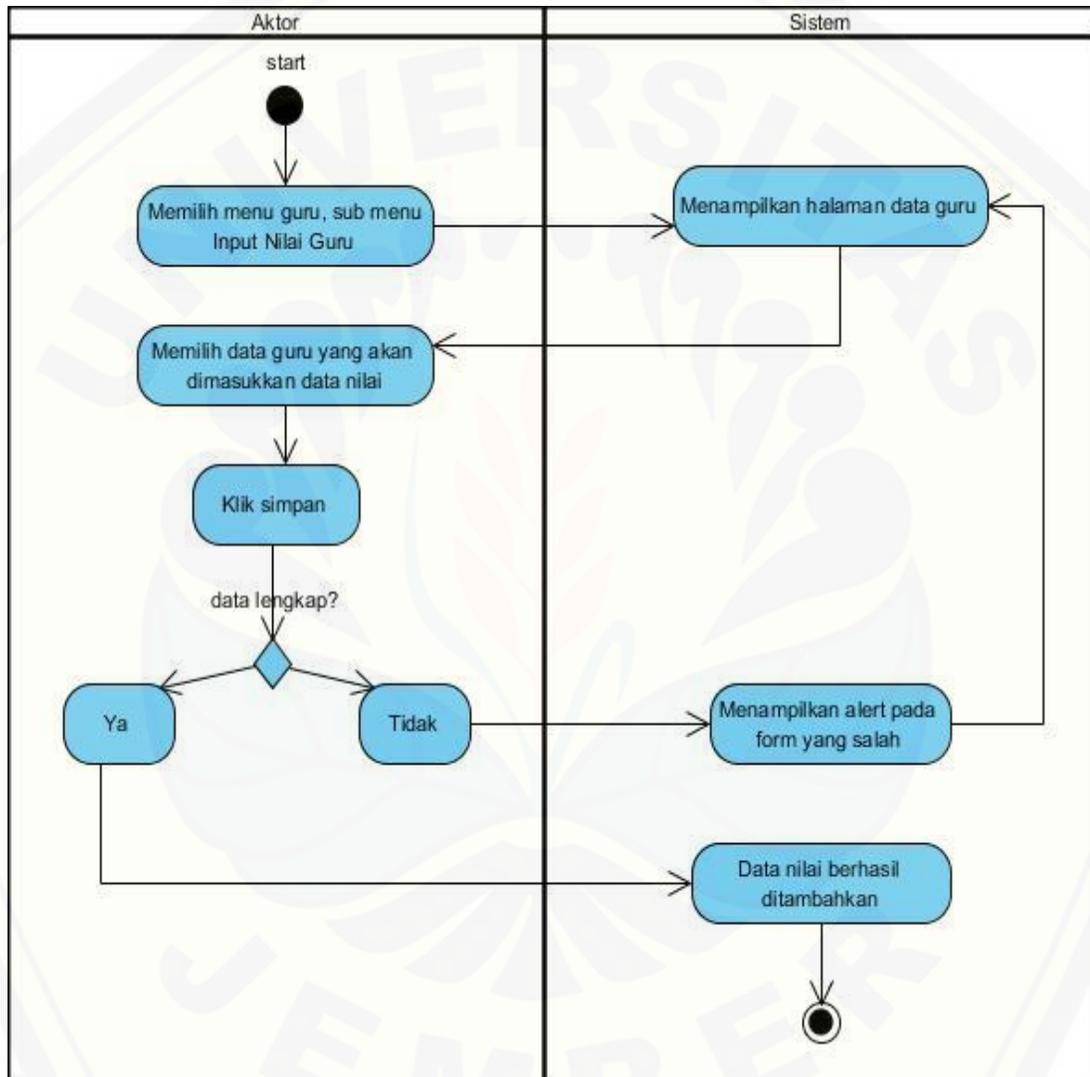
*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur hapus jenis guru. Penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 16



Gambar 16. *Activity Diagram* Hapus Jenis Guru

## 3. Input nilai guru

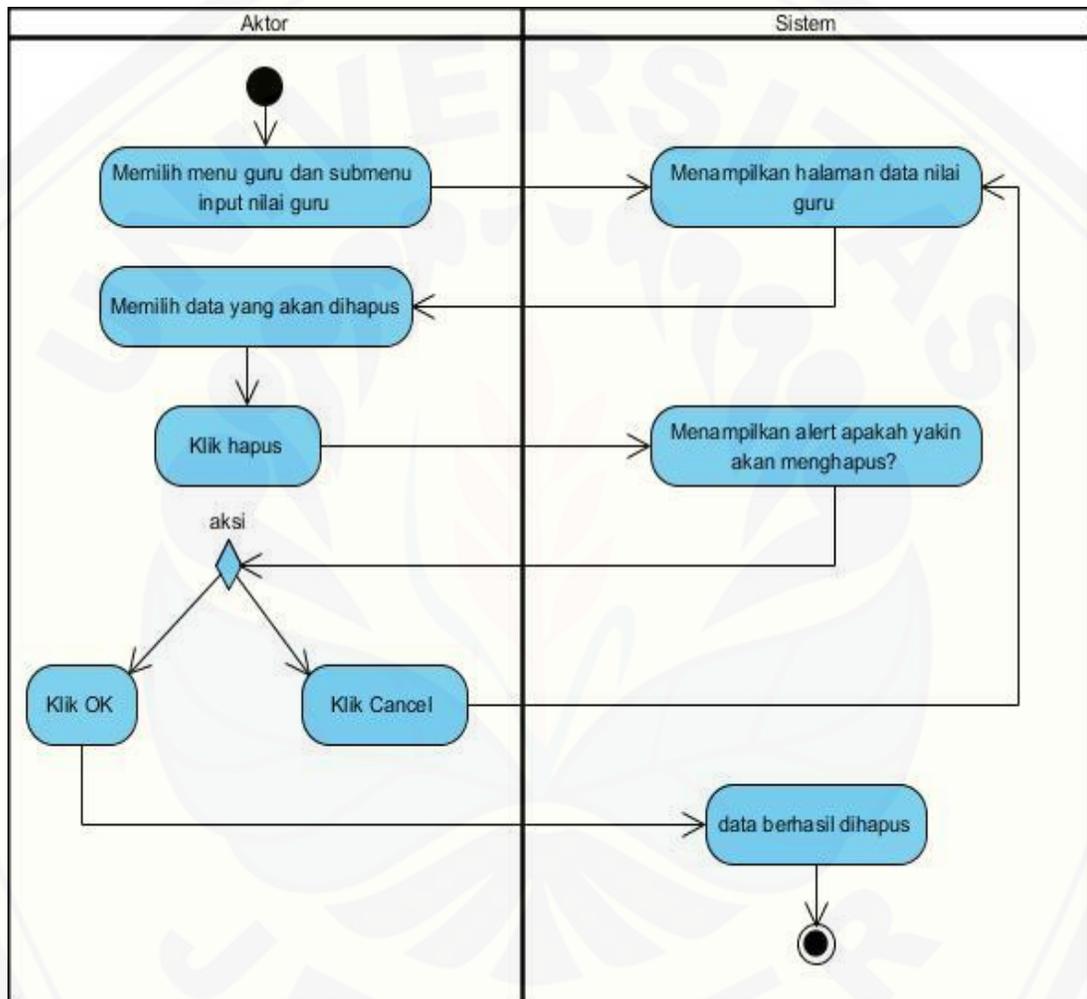
*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur *input* nilai guru. Penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 17



Gambar 17. *Activity Diagram Input Nilai Guru*

## 4. Hapus nilai guru

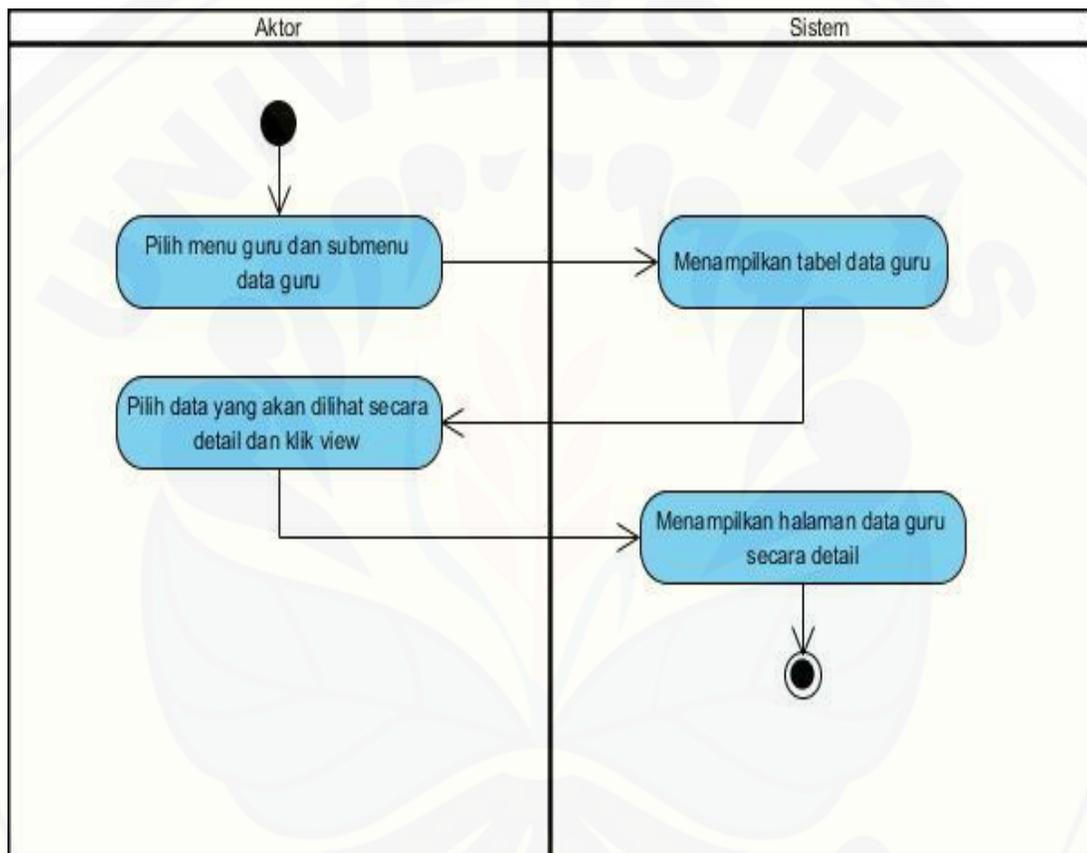
*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur hapus nilai guru. Penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 18



Gambar 18. *Activity Diagram* Hapus Nilai Guru

## 5. Lihat data guru

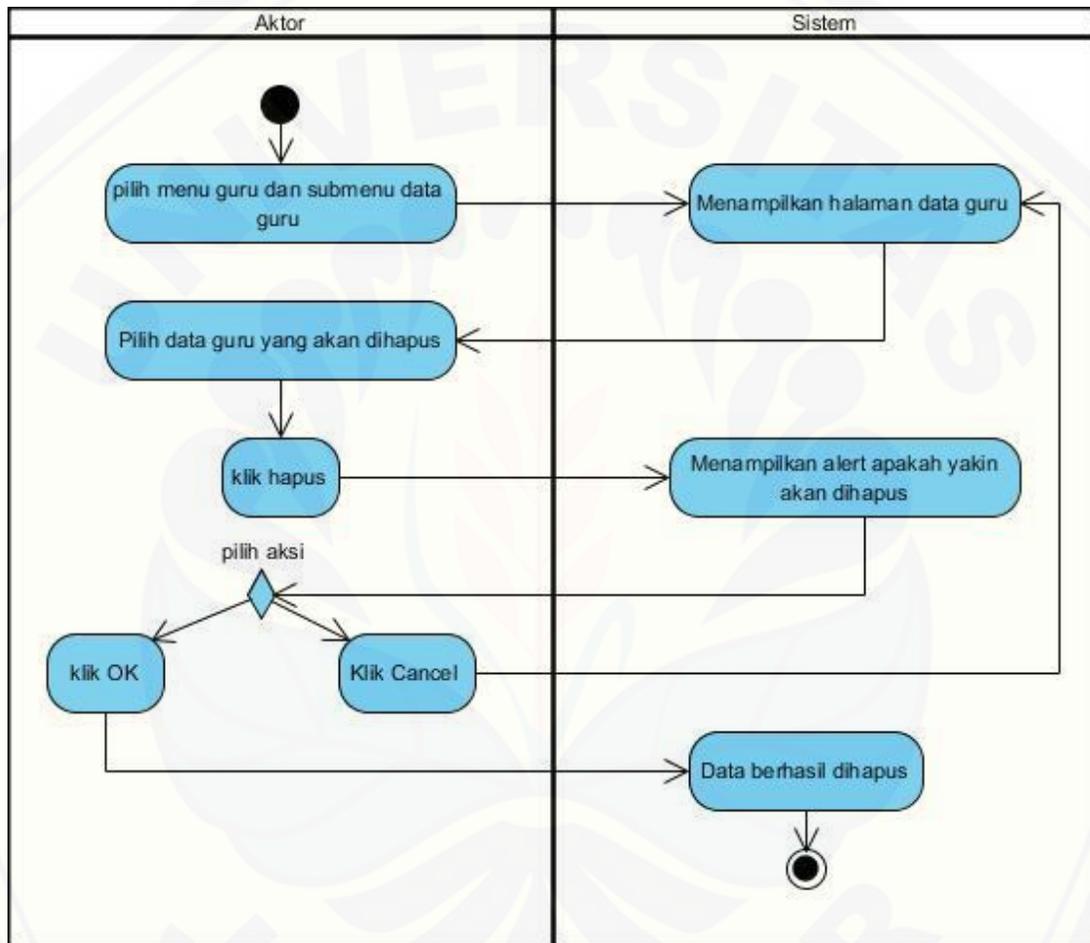
*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur lihat data guru. Penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 19



Gambar 19. *Activity Diagram* Lihat Data Guru

## 6. Hapus data guru

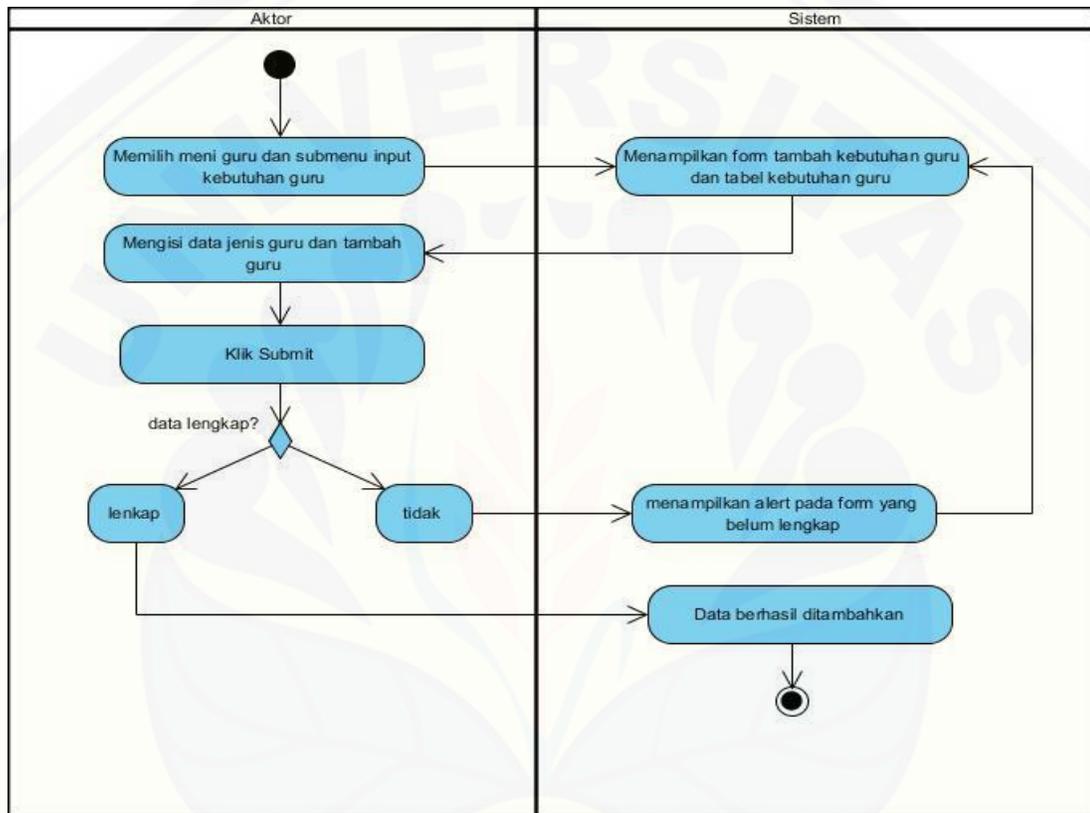
*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur hapus data guru. Penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 20.



Gambar 20. *Activity Diagram* Hapus Data Guru

## 7. Input Kebutuhan Guru

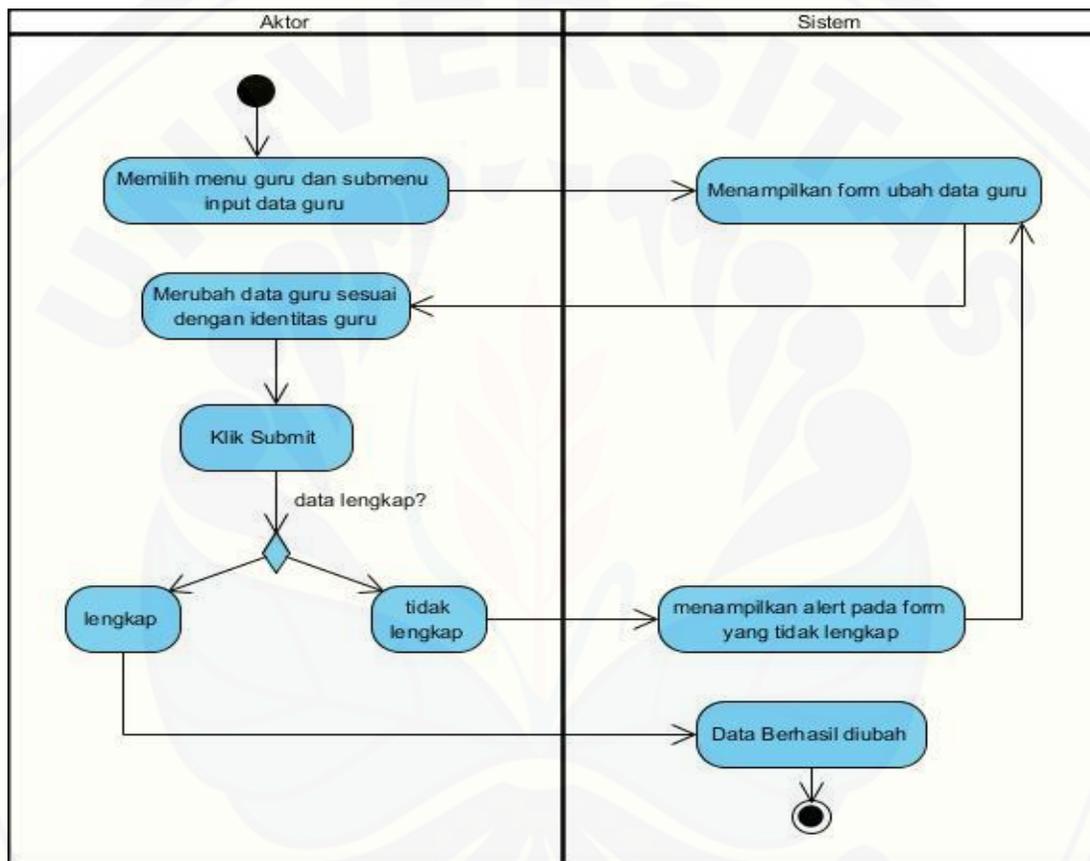
*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur *input* kebutuhan guru. Penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 21



Gambar 21. *Activity Diagram Input Kebutuhan Guru*

## 8. Edit Biodata Guru

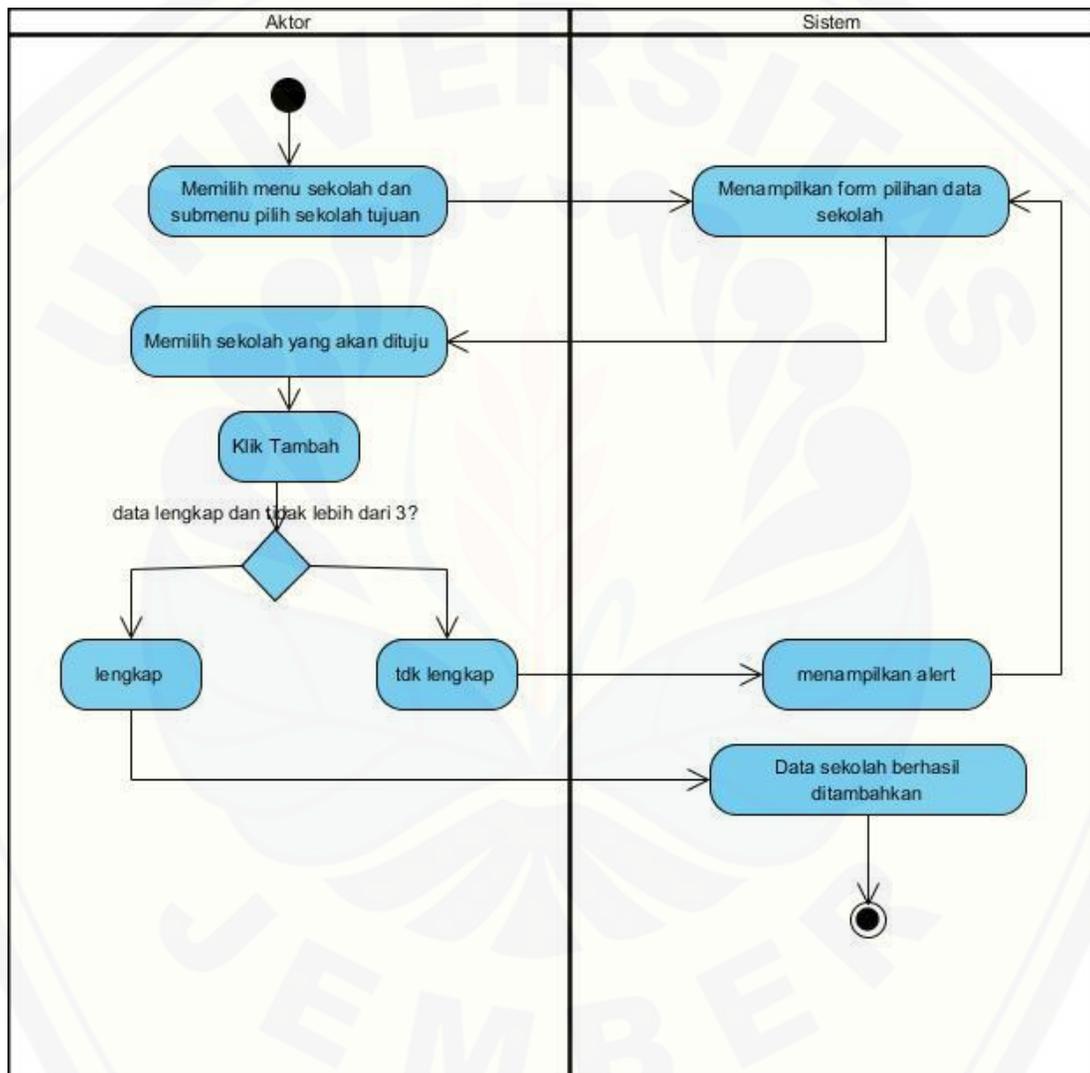
*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur *edit* biodata guru. Penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 22.



Gambar 22. *Activity Diagram Edit Biodata Guru*

## 9. Pemilihan sekolah

*Activity diagram* menjelaskan aliran aktivitas dalam sistem informasi pada fitur pemilihan sekolah. Penjelasan mengenai *activity diagram* ini digambarkan pada Gambar 23.

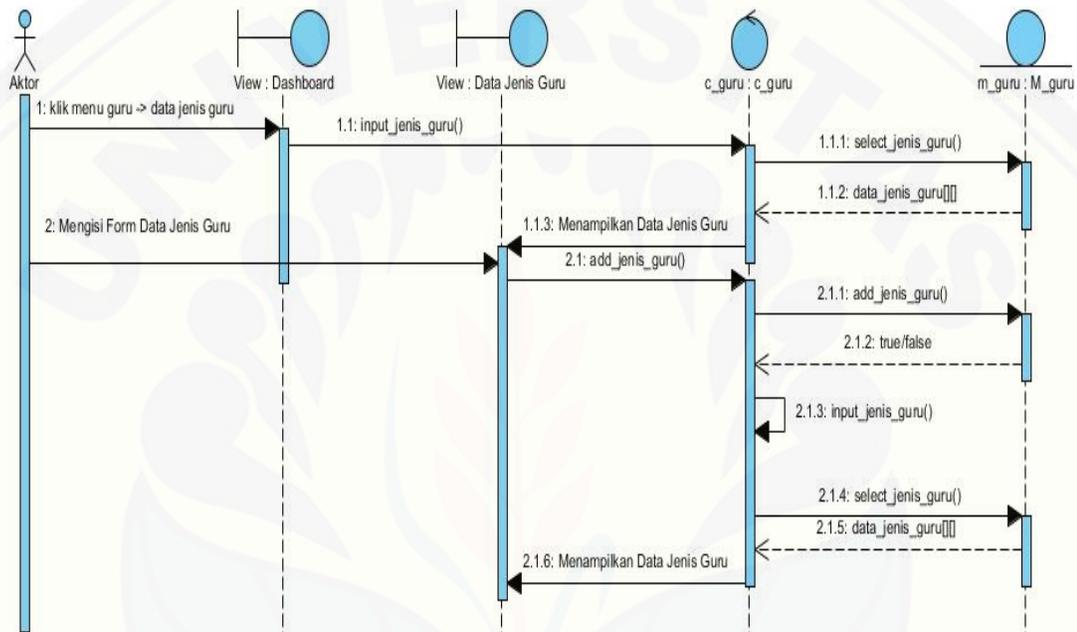


Gambar 23. *Activity Diagram* Pemilihan Sekolah

c. Sequence Diagram

1. Input jenis guru

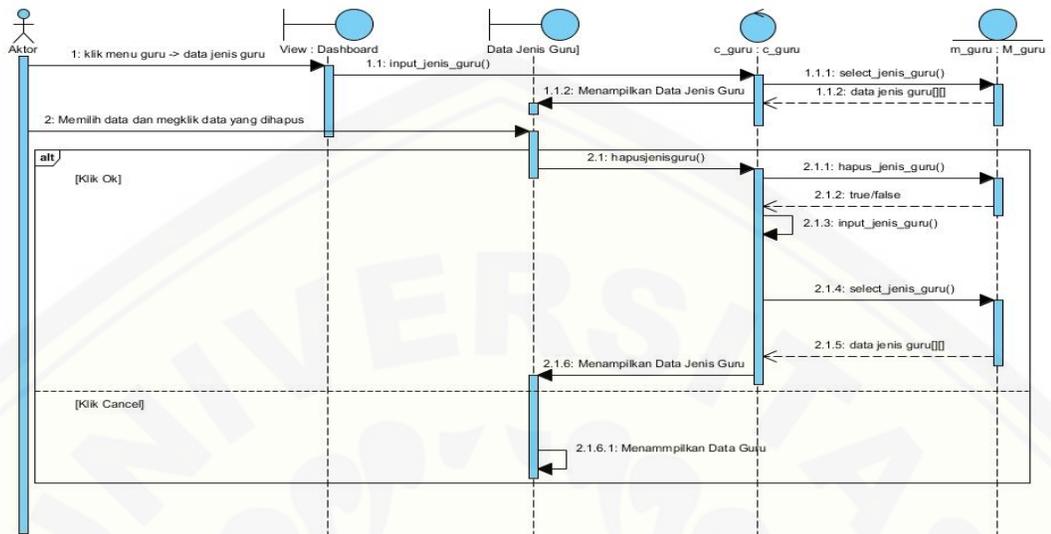
*Sequence diagram* pada fitur *input* jenis guru digunakan untuk memodelkan aliran logika pada sistem yang dibuat. Penggambaran *sequence diagram* dijelaskan pada Gambar 24.



Gambar 24. *Sequence Diagram* Input Jenis Guru

2. Hapus jenis guru

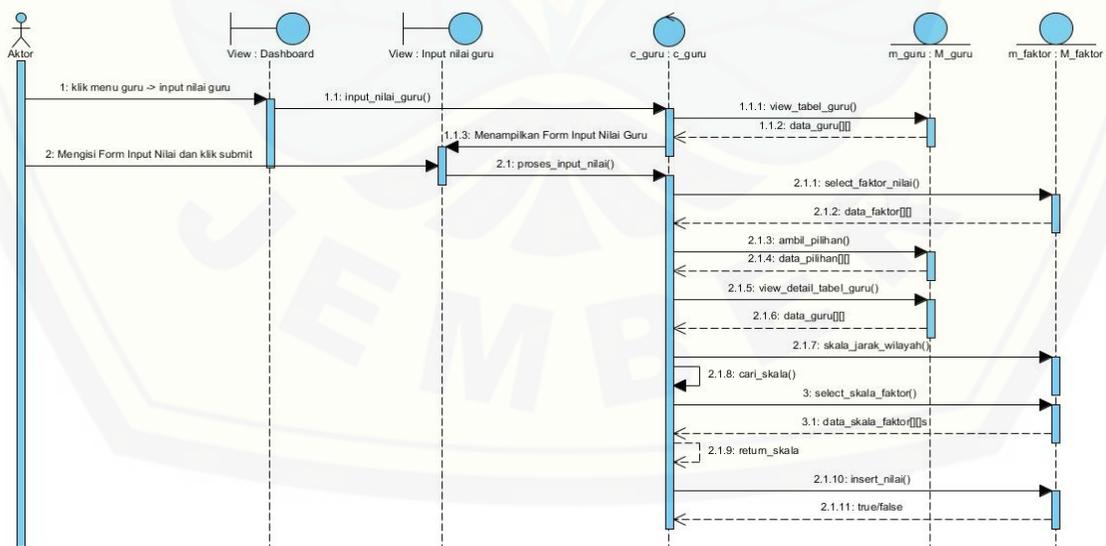
*Sequence diagram* pada fitur hapus jenis guru digunakan untuk memodelkan aliran logika pada sistem yang dibuat. Penggambaran *sequence diagram* dijelaskan pada Gambar 25.



Gambar 25. *Sequence Diagram* Hapus Jenis Guru

3. Input nilai guru

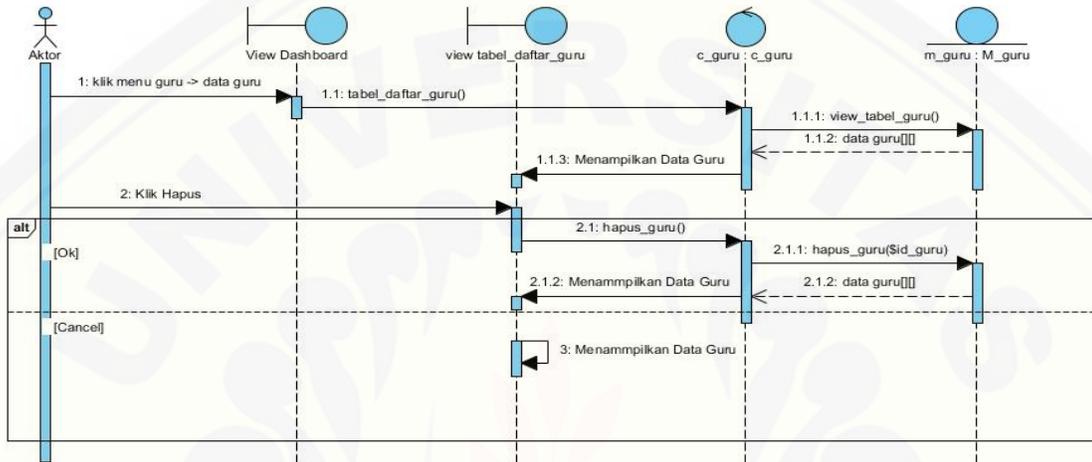
*Sequence diagram* pada fitur *input* nilai guru digunakan untuk memodelkan aliran logika pada sistem yang dibuat. Penggambaran *sequence diagram* dijelaskan pada Gambar 26.



Gambar 26. *Sequence Diagram* Input Nilai Guru.

4. Hapus nilai guru

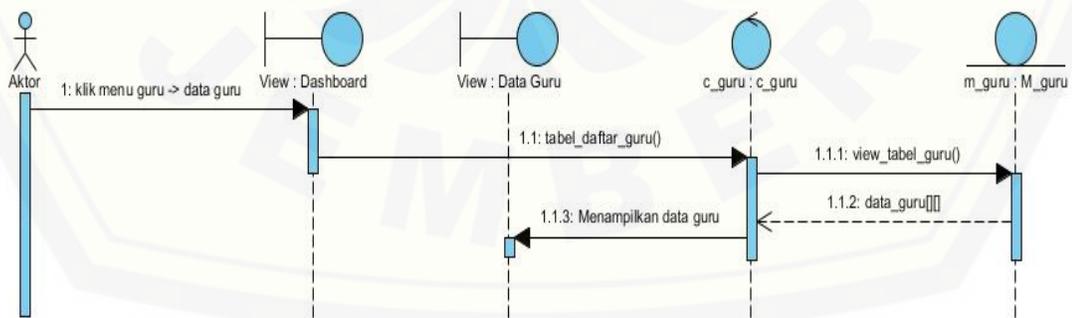
*Sequence diagram* pada fitur hapus nilai guru digunakan untuk memodelkan aliran logika pada sistem yang dibuat. Penggambaran *sequence diagram* dijelaskan pada Gambar 27



Gambar 27. *Sequence Diagram* Hapus Nilai Guru

5. Lihat data guru

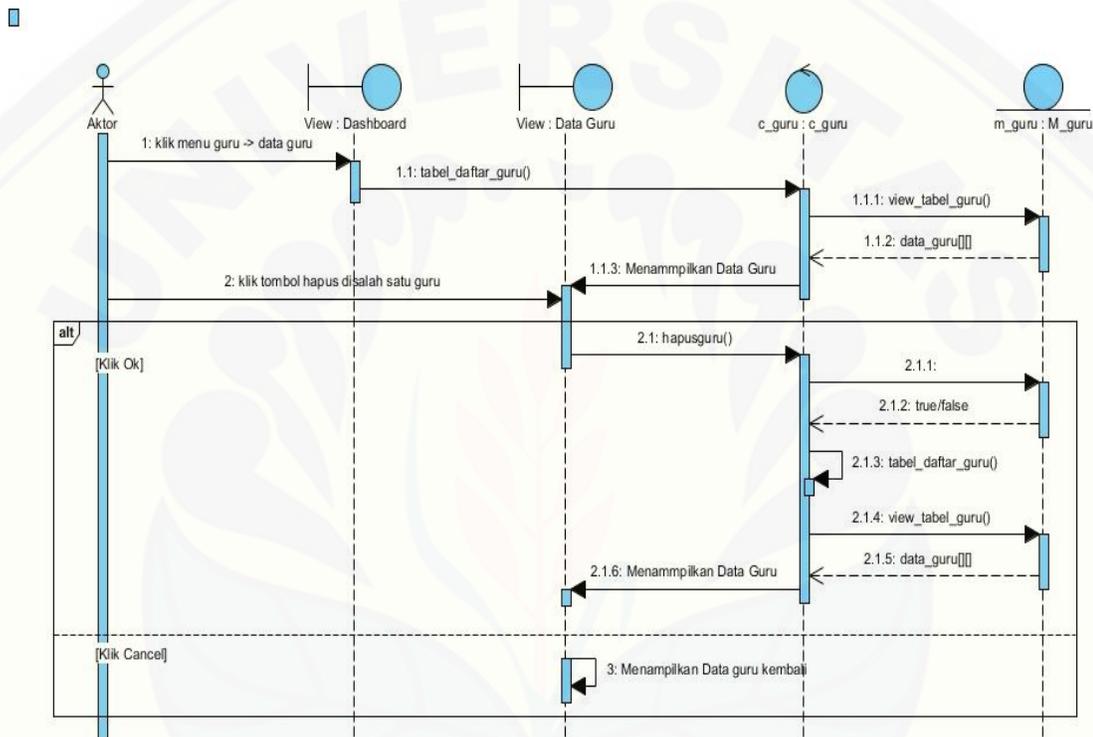
*Sequence diagram* pada fitur lihat data guru digunakan untuk memodelkan aliran logika pada sistem yang dibuat. Penggambaran *sequence diagram* dijelaskan pada Gambar 28.



Gambar 28. *Sequence Diagram* Lihat Data Guru

## 6. Hapus data guru

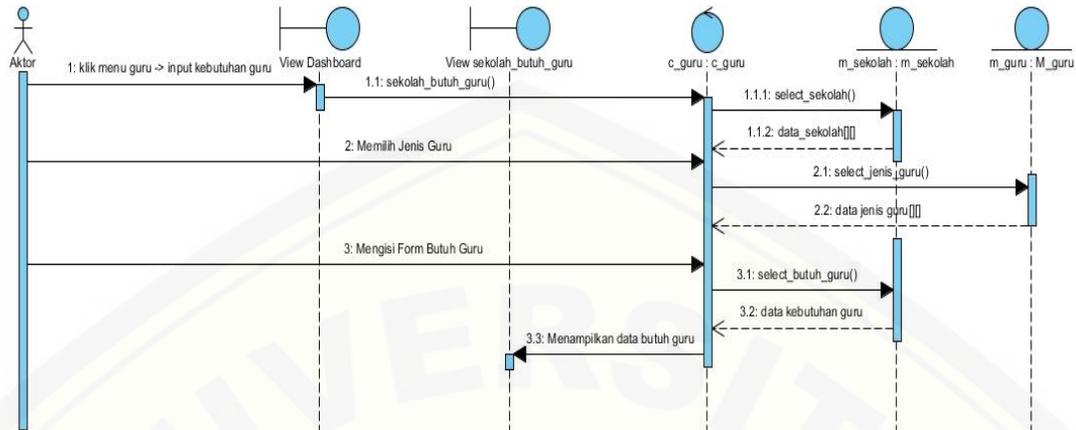
*Sequence diagram* pada fitur hapus data guru digunakan untuk memodelkan aliran logika pada sistem yang dibuat. Penggambaran *sequence diagram* dijelaskan pada Gambar 29.



Gambar 29. *Sequence Diagram* Hapus Data Guru

## 7. Input Kebutuhan Guru

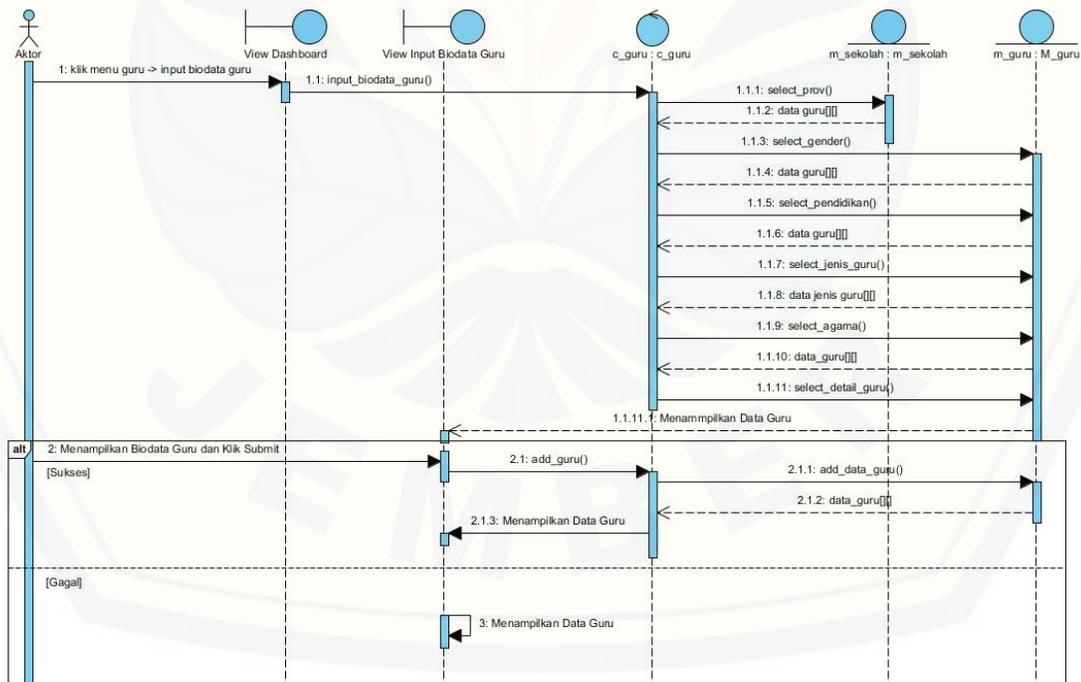
*Sequence diagram* pada fitur *input* kebutuhan guru digunakan untuk memodelkan aliran logika pada sistem yang dibuat. Penggambaran *sequence diagram* dijelaskan pada Gambar 30.



Gambar 30. Sequence Diagram Input Kebutuhan Guru

8. Edit biodata guru

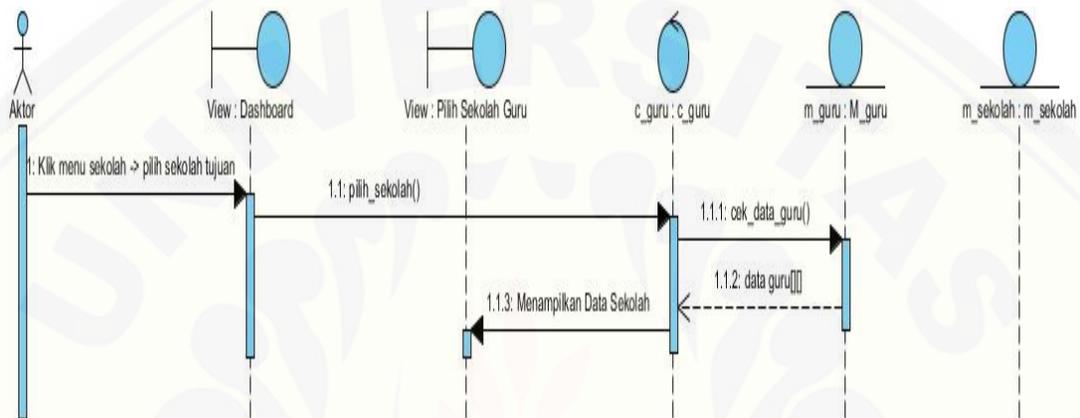
Sequence diagram pada fitur edit biodata guru digunakan untuk memodelkan aliran logika pada sistem yang dibuat. Penggambaran sequence diagram dijelaskan pada Gambar 31.



Gambar 31. Sequence Diagram Edit Biodata Guru

## 9. Pemilihan sekolah

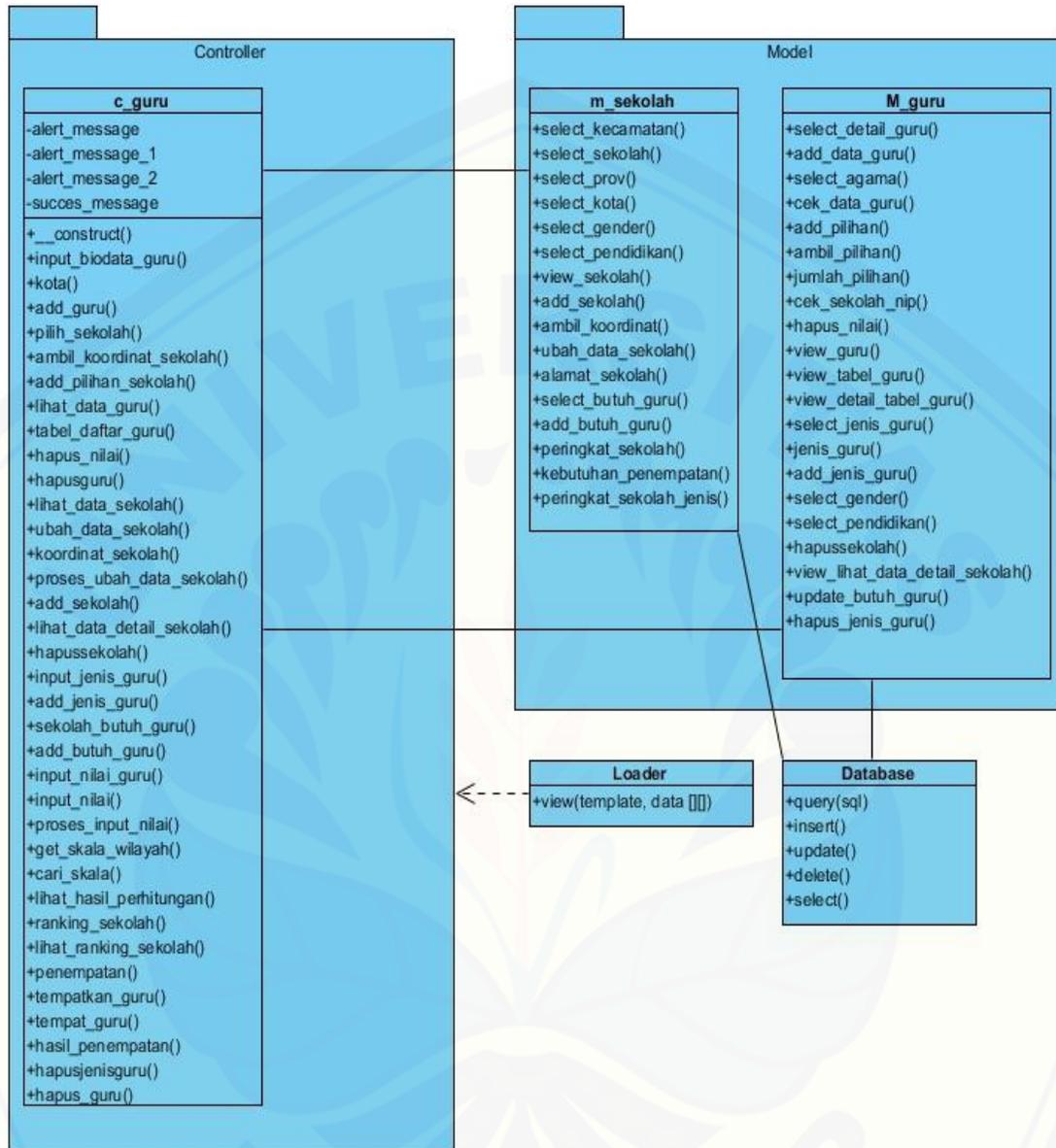
*Sequence diagram* pada fitur pemilihan sekolah digunakan untuk memodelkan aliran logika pada sistem yang dibuat. Penggambaran *sequence diagram* dijelaskan pada Gambar 32.



Gambar 32. *Sequence Diagram* Pemilihan Sekolah  
Sumber: (Hasil Analisis, 2015)

## d. Class Diagram

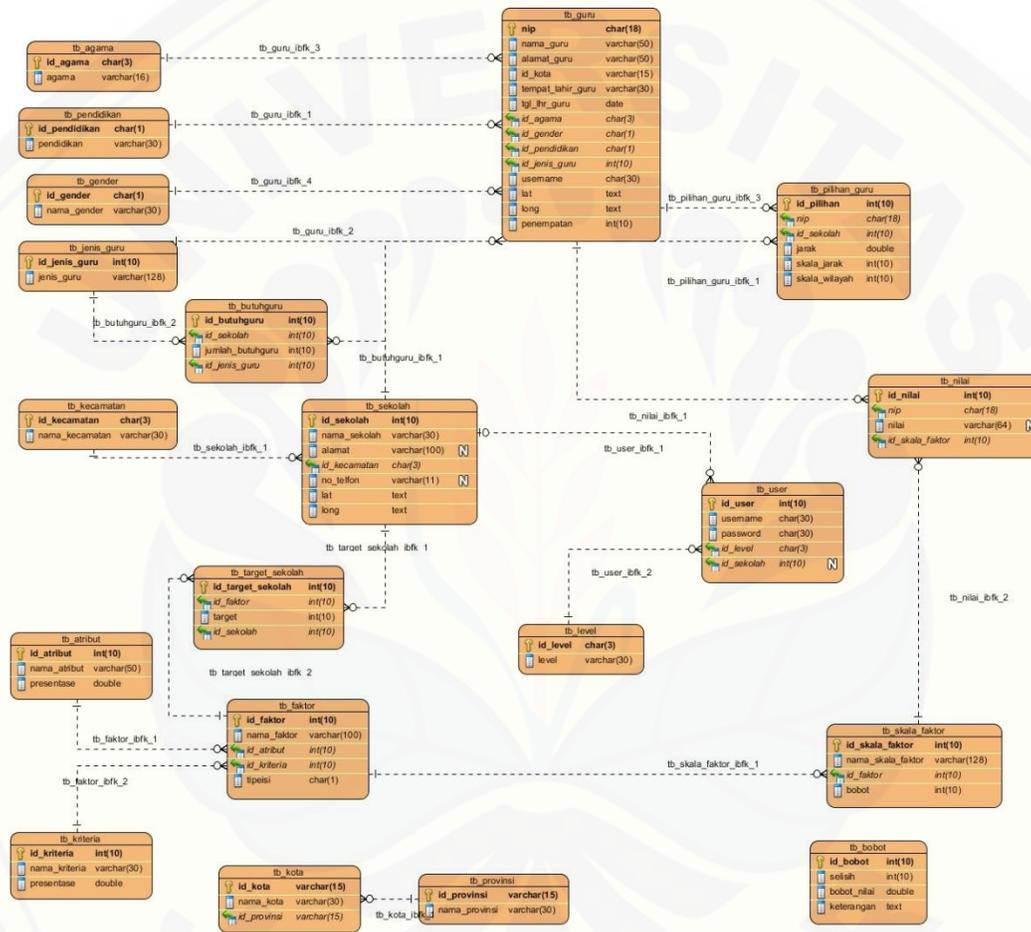
Gambar 33 merupakan gambaran dari *class diagram* manajemen data guru yang menghasilkan beberapa jenis operasi yang digunakan dalam alur kerja sistem. Gambaran dari *class diagram* manajemen data pengguna dijelaskan pada Gambar 33.



Gambar 33. Class Diagram Manajemen Data Guru  
 Sumber: (Hasil Analisis, 2015)

e. *Entity Relationship Diagram*

*Entity Relationship Diagram* modul manajemen data guru berisi gambaran relasi data yang terhubung dalam sistem dan database. Gambaran *Entity Relationship Diagram* dijelaskan pada Gambar 34.



Gambar 34. *Entity Relationship Diagram* Manajemen Data Guru  
Sumber: (Hasil Analisis, 2015)

## LAMPIRAN B. IMPLEMENTASI SISTEM

### B.1 Implementasi Modul 1 Manajemen Data Pengguna

1. Halaman menambah pengguna merupakan halaman yang digunakan oleh *user* untuk menambah pengguna baru yang dapat mengakses sistem yang telah dibuat. Hak akses ini dimiliki oleh *user admin*. Halaman menambah *user* pengguna pada Gambar 35.

The screenshot shows the 'Tambahkan User Baru' (Add New User) page. The form includes fields for Username, Password, and dropdown menus for selecting Level and Sekolah. Below the form is a table titled 'TABEL DAFTAR USER' (User List Table) with columns for No., Username, and Level.

No.	Username	Level
1	admin	Admin
2	bid	BKD
3	kepsekantirogo1	Kepala Sekolah
4	kepsekantirogo2	Kepala Sekolah
5	kepsekantirogo3	Kepala Sekolah

Gambar 35. Halaman tambah *user* pengguna.

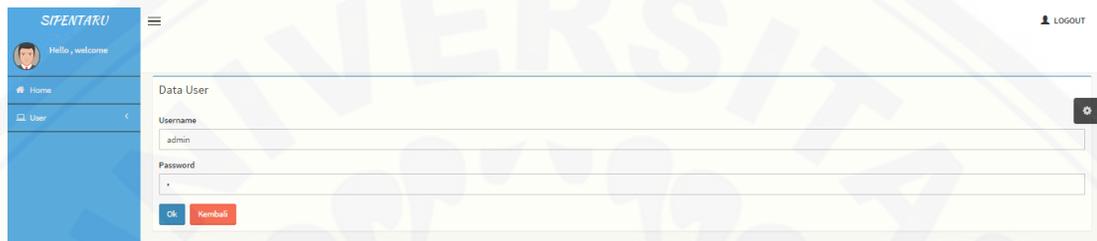
2. Halaman *view* pengguna merupakan halaman yang digunakan oleh *user admin* untuk melihat pengguna tertentu yang dipilihnya. Halaman ini berisi data *username* dan *password* pengguna yang dimaksud. Hak akses ini dimiliki oleh *user admin*. Halaman *view* pengguna pada Gambar 36.

The screenshot shows the 'DETAIL PENGGUNA' (User Detail) page. It displays the details of a selected user, including Username, Password, and Level.

Username	: admin
Password	: 1
Level	: Admin

Gambar 36. Halaman *view* pengguna

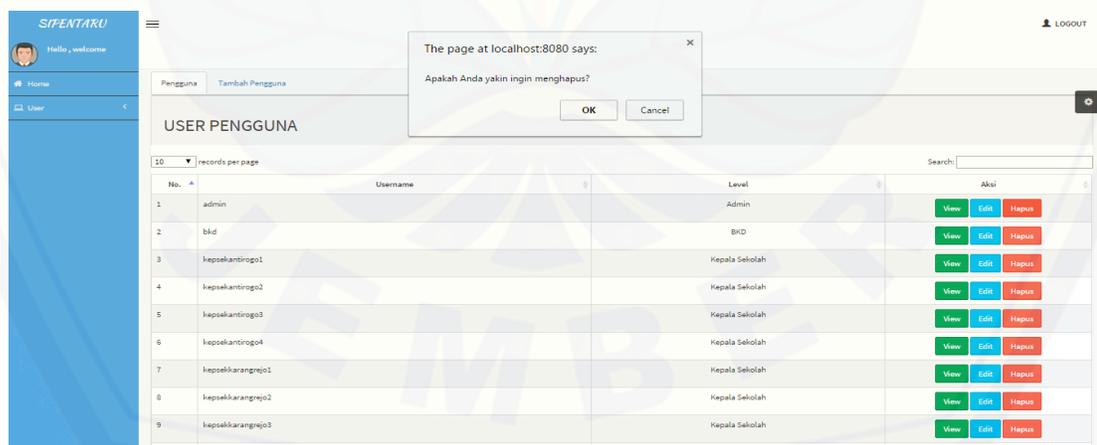
3. Halaman *edit* pengguna merupakan halaman yang digunakan oleh *user admin* untuk mengedit pengguna tertentu yang dipilihnya. Halaman ini berisi data *username* dan *password* pengguna yang dimaksud. Hak akses ini dimiliki oleh *user admin*. Halaman *edit* pengguna pada Gambar 37.



The screenshot shows the 'Data User' form in the SIPENTARU application. The form has two main input fields: 'Username' with the value 'admin' and 'Password' with a masked input. Below the fields are two buttons: 'Ok' (green) and 'Kembali' (red). The application header includes the logo 'SIPENTARU', a user profile, and a 'LOGOUT' link. A sidebar on the left contains navigation options: 'Home' and 'User'.

Gambar 37. Halaman *edit* pengguna

4. Halaman hapus pengguna merupakan halaman yang digunakan oleh *user admin* untuk menghapus pengguna tertentu yang dipilihnya. Halaman ini terletak di halaman utama *user* pengguna. Hak akses ini dimiliki oleh *user admin*. Halaman hapus pengguna pada Gambar 38.



The screenshot shows the 'USER PENGGUNA' management page in the SIPENTARU application. A modal dialog is displayed in the center, asking 'Apakah Anda yakin ingin menghapus?' (Are you sure you want to delete?) with 'OK' and 'Cancel' buttons. The background shows a table of users with columns for 'No.', 'Username', 'Level', and 'Aksi'. The 'Aksi' column contains 'View', 'Edit', and 'Hapus' buttons for each user.

No.	Username	Level	Aksi
1	admin	Admin	View Edit Hapus
2	blid	BKD	View Edit Hapus
3	kepoekantirog01	Kepala Sekolah	View Edit Hapus
4	kepoekantirog02	Kepala Sekolah	View Edit Hapus
5	kepoekantirog03	Kepala Sekolah	View Edit Hapus
6	kepoekantirog04	Kepala Sekolah	View Edit Hapus
7	kepoekarangrejo1	Kepala Sekolah	View Edit Hapus
8	kepoekarangrejo2	Kepala Sekolah	View Edit Hapus
9	kepoekarangrejo3	Kepala Sekolah	View Edit Hapus

Gambar 38. Halaman hapus pengguna

### B.3 Implementasi Modul III Implementasi Metode

1. Halaman hasil perhitungan nilai *gap* merupakan halaman yang digunakan oleh *user* Badan Kepegawaian Daerah untuk melihat nilai *gap* tiap guru. Halaman ini berisi nilai total tiap guru yang mengikuti seleksi penempatan guru. Hak akses ini dimiliki oleh *user* BKD. Halaman hasil perhitungan nilai *gap* pada Gambar 39.



No.	NIP	Nama	Nilai	Target	GAP	Bobot Gap
1	198902272014121211	Anggi Anugrahika Pratama S.Pd				
1. SDN JEMBER LOR 03						
-	Jarak		5	5	0	5
-	Status Wilayah		2	2	0	5
2. SDN SUMBERSARI 03						
-	Jarak		4	4	0	5
-	Status Wilayah		3	2	1	4,5
3. SDN GEBANG 02						
-	Jarak		5	2	3	2,5
-	Status Wilayah		2	2	0	5
2	199005192014121219	Ryan Dharmawan S.Pd				
1. SD NEGERI MANGLI 02						
-	Jarak		4	3	1	4,5
-	Status Wilayah		2	2	0	5
-	Usia		4	2	2	3,5
-	Pendidikan		1	1	0	5

Gambar 39. Halaman hasil perhitungan nilai *gap*

2. Halaman hasil perhitungan nilai *core* dan *secondary factor* merupakan halaman yang digunakan oleh *user* Badan Kepegawaian Daerah untuk melihat nilai *core* dan *secondary factor* tiap guru. Halaman ini berisi nilai nilai *core* dan *secondary factor* tiap guru yang mengikuti seleksi penempatan guru. Hak akses ini dimiliki oleh *user* BKD. Halaman hasil perhitungan nilai *core* dan *secondary factor* pada Gambar 40.

No.	NIP	Nama	Nilai Core	Nilai Secondary
1	198902272014121211	Anggi Anugrahadika Pratama S.Pd		
		1.SDN JEMBER LOR 03		
		Jarak	5	5
		2.SDN SUMBERSARI 03		
		Jarak	5	4.5
		3.SDN GEBANG 02		
		Jarak	2.5	5
2	199005192014121219	Ryan Dharmawan S.Pd		
		1.SD NEGERI MANGLI 02		
		Jarak	4.5	5
		Kepribadian Diri	4.25	5
		Nilai Test	4.5	4.333333333333333
		2.SDN KRANJIJANGAN 01		
		Jarak	4	5
		Kepribadian Diri	4.75	5
		Nilai Test	4.666666666666667	4.5

Gambar 40. Halaman perhitungan nilai *core* dan *secondary factor*

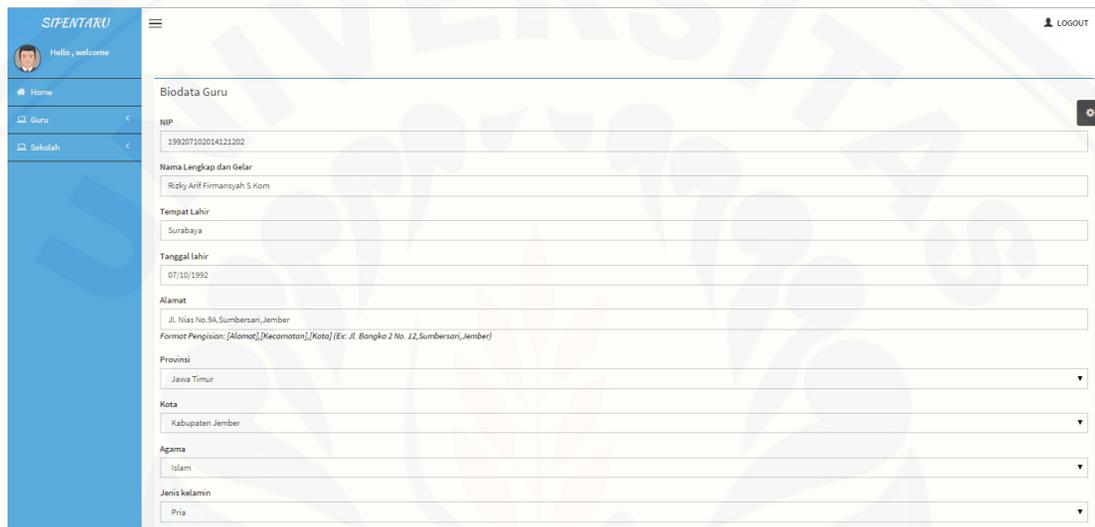
3. Halaman hasil perhitungan nilai total merupakan halaman yang digunakan oleh *user* Badan Kepegawaian Daerah untuk melihat nilai total keseluruhan dari tiap guru yang merupakan hasil akhir dari perhitungan metode *profile matching*. Halaman ini berisi nilai nilai total tiap guru yang mengikuti seleksi penempatan guru. Hak akses ini dimiliki oleh *user* BKD. Halaman hasil perhitungan nilai total pada Gambar 41.

No.	NIP	Nama	Nj	Nk	Nt	Total
1	198902272014121211	Anggi Anugrahadika Pratama S.Pd				
		1.SDN JEMBER LOR 03	5	0	0	1
		2.SDN SUMBERSARI 03	4.6	0	0	0.96
		3.SDN GEBANG 02	3.5	0	0	0.7000000000000001
2	199005192014121219	Ryan Dharmawan S.Pd				
		1.SD NEGERI MANGLI 02	4.699999999999999	4.55	4.433333333333334	4.521666666666665
		2.SDN KRANJIJANGAN 01	4.4	4.85	4.6000000000000005	4.635
		3.SD NEGERI MANGLI 03	3.9000000000000004	4.5	4.533333333333333	4.396666666666665
3	199012012014121213	Musawiru Alam S.Pd				
		1.SDN TEGAL BESAR 04	4.3	4.45	4.766666666666667	4.578333333333333
		2.SDN KARANGREJO 02	4.4	4.05	4.533333333333333	4.361666666666666
		3.SDN KRANJIJANGAN 01	4.8	4.65	4.866666666666667	4.788333333333334
4	199102212014121205	Oby Syam S.Pd				
		1.SDN KEPATIHAN 02	5	4.05	4.066666666666666	4.248333333333333
		2.SDN PATRANG 01	3	4.05	3.933333333333333	3.781666666666663
		3.SDN SUMBERSARI 03	2.8	4.699999999999999	4.1	4.02

Gambar 40. Halaman perhitungan nilai total

## B.4 Implementasi Modul IV Manajemen Data Guru

1. Halaman *input* biodata guru merupakan halaman yang digunakan oleh *user* guru untuk memasukkan inputan data biodata lengkap dirinya. Halaman ini berisi informasi detail tentang guru itu sendiri yang mengikuti seleksi penempatan guru. Hak akses ini dimiliki oleh *user* guru. Halaman *input* biodata guru pada Gambar 41.

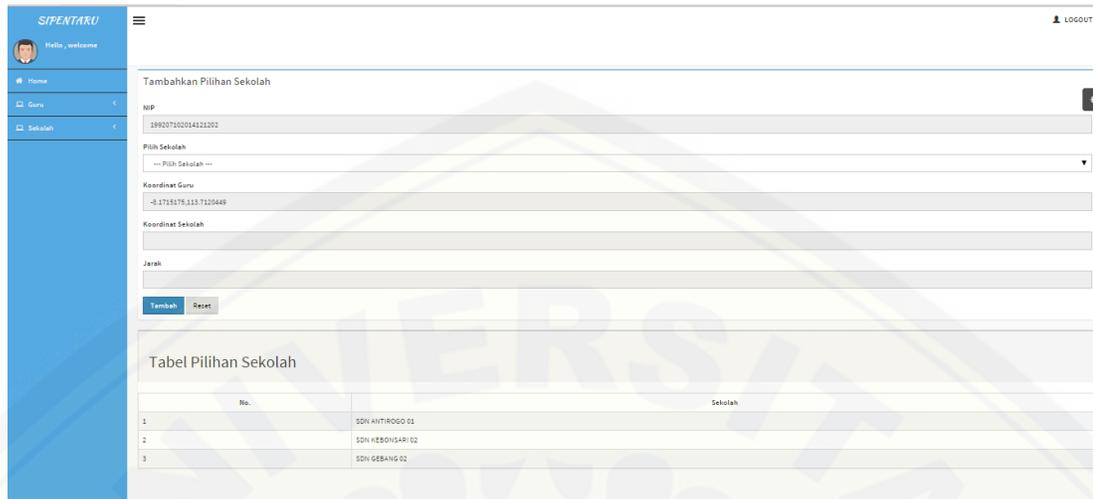


The screenshot shows a web application interface for 'SIPENTARU'. The left sidebar contains navigation options: Home, Guru, and Sekolah. The main content area is titled 'Biodata Guru' and contains a form with the following fields:

- NIP: 199207102014121202
- Nama Lengkap dan Gelar: Rizky Arif Firmansyah S.Kom
- Tempat Lahir: Surabaya
- Tanggal lahir: 07/10/1992
- Alamat: Jl. Nias No.9A,Sumbersari,Jember  
Format Pengisian: [Alamat],[Kecamatan],[Kota] (Ex: Jl. Bangka 2 No. 12,Sumbersari,Jember)
- Provinsi: Jawa Timur
- Kota: Kabupaten Jember
- Agama: Islam
- Jenis Kelamin: Pria

Gambar 41. Halaman *input* biodata guru

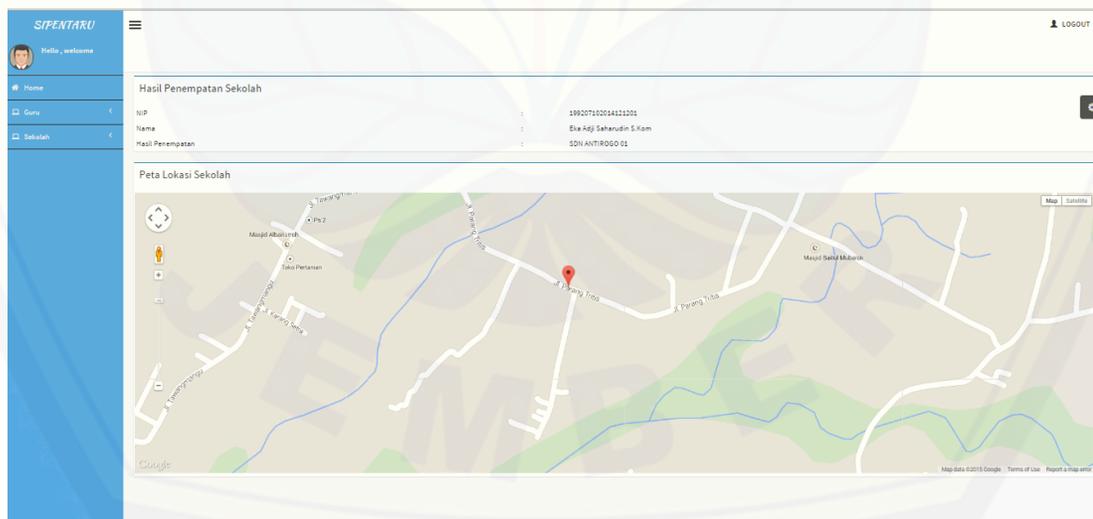
2. Halaman pilih sekolah tujuan merupakan halaman yang digunakan oleh *user* guru untuk memasukkan daftar pilihan sekolah yang diinginkan oleh guru tersebut. Guru memasukkan data pilihan sekolah yang diinginkan setelah itu sistem menampilkan koordinat lokasi sekolah tersebut yang digunakan untuk memberi tahu lokasi sekolah yang digunakan untuk membantu guru mengetahui lokasi sekolah melalui bantuan GoogleMap. Batas pilihan sekolah adalah 3 sekolah apabila lebih akan muncul alert. Hak akses ini dimiliki oleh *user* guru. Halaman pilih sekolah tujuan pada Gambar 42.



No.	Sekolah
1	SDN ANTIROGO 01
2	SDN KEBONHARI 02
3	SDN GEBANG 02

Gambar 42. Halaman pilih sekolah tujuan

3. Halaman hasil penempatan guru merupakan halaman yang digunakan oleh *user* guru untuk mengetahui hasil penempatan sekolah untuknya. Halaman ini berisi informasi tentang nip dan nama guru tersebut dan tampilan lokasi sekolah yang ditampilkan oleh GoogleMap. Hak akses ini dimiliki oleh *user* guru. Halaman hasil penempatan guru pada Gambar 43.



Gambar 43. Halaman Hasil Penempatan Guru

**LAMPIRAN C. PENGUJIAN WHITE BOX SISTEM**

**C.1 Pengujian *White Box* Modul 1 Manajemen Data Pengguna**

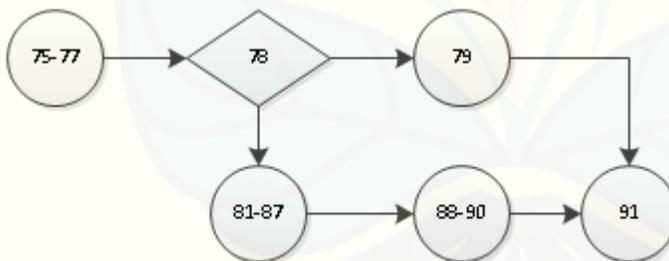
Nama Modul	Manajemen Data Pengguna (Menambah data pengguna, mengubah data pengguna, menghapus data pengguna)
File	c_login

1. Listing Program Tambah Data User

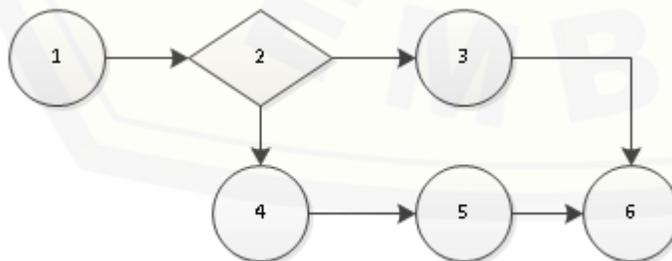
```

1 75 public function tambahuser() {
76     //function yang dipanggil ketika ingin memasukan produk ke dalam database
77     $id_sekolah = $this->input->post('sekolah');
78     if($id_sekolah == 0){
79         $id_sekolah = NULL; 2
80
81     data = array(
82         'username' => $this->input->post('username'),
83         'password' => $this->input->post('password'),
84         'id_level' => $this->input->post('level'),
85         'id_sekolah' => $id_sekolah
86     );
87
88     5 $this->m_pengguna->tambahpengguna($data); //passing variable $data ke m_pengguna
89
90     redirect('c_login/admin_pengguna_tambah'); //redirect page ke halaman utama controller admin
91 }
6
    
```

a. Diagram Alir Tambah Data *User*



b. Grafik Alir Tambah Data *User*



c. Penghitungan Kompleksitas Siklomatik

Kompleksitas siklomatik dari tambah data *user* dapat diperoleh dengan perhitungan berikut :

$$V(G) = E - N + 2 = 6 - 6 + 2 = 2$$

d. Basis set

*Basis set* tambah data *user* menghasilkan 2 jalur independent dari perhitungan skompleksitas siklomatik, yaitu :

Jalur 1 : 1-2-3-6

Jalur 2 : 1-2-4-5-6

e. Test Case

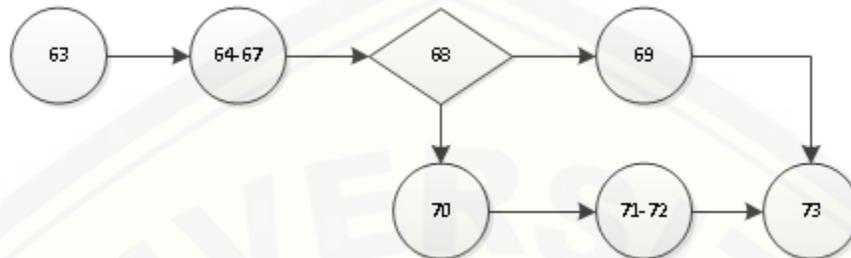
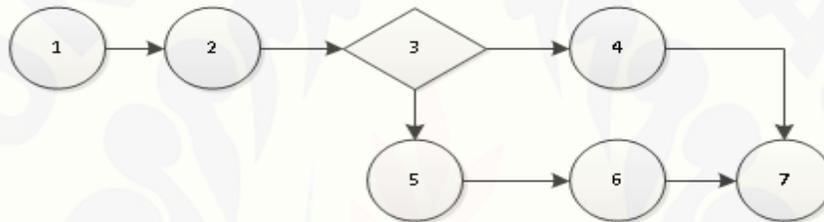
*Test Case*: Tambah Data User

No	Pengujian	Jalur	Kesimpulan	Target yang diharapkan
1	Ketika user menamahkan data pengguna	1-2-3-6	[√] berhasil [ ] gagal	Data berhasil disimpan ke database
2	Ketika user menamahkan data pengguna	1-2-4-5-6	[√] berhasil [ ] gagal	Data tidak disimpan dalam database

2. Listing Program Mengubah Data User

```

63 | function update_pengguna() { 1
64 |     $id_user = $this->input->post('id_user');
65 |     $username = $this->input->post('username'); 2
66 |     $password = $this->input->post('password');
67 |     $name = $this->input->post('name');
68 |     if ($this->m_pengguna->update_pengguna($id_user, $username, $password)) { 3
69 |         redirect('c_login/user_pengguna'); 4
70 |     } else { 5
71 |         redirect('c_login/user_pengguna');
72 |     } 6
73 | } 7
    
```

a. Diagram Alir UbahData *User*b. Grafik Alir Ubah Data *User*

## c. Penghitungan Kompleksitas Siklomatik

Kompleksitas siklomatik dari tambah data *user* dapat diperoleh dengan perhitungan berikut :

$$V(G) = E - N + 2 = 7 - 7 + 2 = 2$$

d. Basis *set*

*Basis set* tambah data *user* menghasilkan 2 jalur independent dari perhitungan skompleksitas siklomatik, yaitu :

Jalur 1 : 1-2-3-4-7

Jalur 2 : 1-2-3-5-6-7

e. Test Case

*Test Case: Tambah Data User*

No	Pengujian	Jalur	Kesimpulan	Target yang diharapkan
1	Ubah data pengguna	1-2-3-4-7	[√] berhasil [ ] gagal	Data pengguna berhasil diubah
2	ubah data pengguna	1-2-3-5-6-7	[ ] berhasil [√]gagal	Data pengguna tidak berhasil diubah

3. Listing Program Menghapus Data User

```

58 public function hapuspengguna($id) { 1
59     $this->m_pengguna->hapus_pengguna($id); 2
60     redirect('c_login/user_pengguna');
61 } 3
    
```

a. Diagram Alir Hapus Data User



b. Grafik Alir Hapus Data User



**c. Penghitungan Kompleksitas Siklomatik**

Kompleksitas siklomatik dari tambah data *user* dapat diperoleh dengan perhitungan berikut :

$$V(G) = E - N + 2 = 2 - 3 + 2 = 1$$

**d. Basis set**

*Basis set* tambah data *user* menghasilkan 1 jalur independent dari perhitungan skompleksitas siklomatik, yaitu :

Jalur 1 : 1-2-3

**e. Test Case**

*Test Case: Tambah Data User*

No	Pengujian	Jalur	Kesimpulan	Target yang diharapkan
1	Menghapus data pengguna	1-2-3	[✓] berhasil [ ] gagal	Data berhasil dihapus

### C.2 Pengujian *White Box* Modul II Manajemen Data Sekolah

Nama Modul	Manajemen Data Sekolah (menambah data sekolah, mengubah data sekolah, menghapus data sekolah)
File	c_guru, m_sekolah

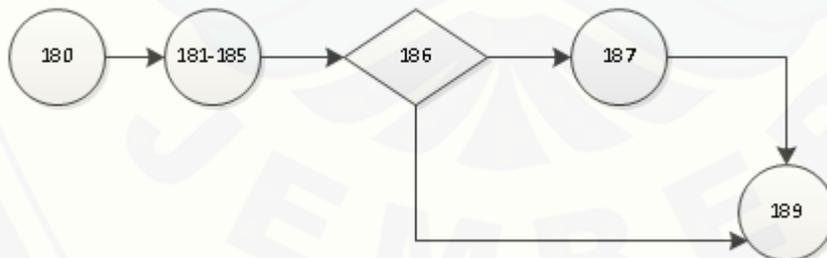
#### 1. Listing Program Tambah Data Sekolah

```

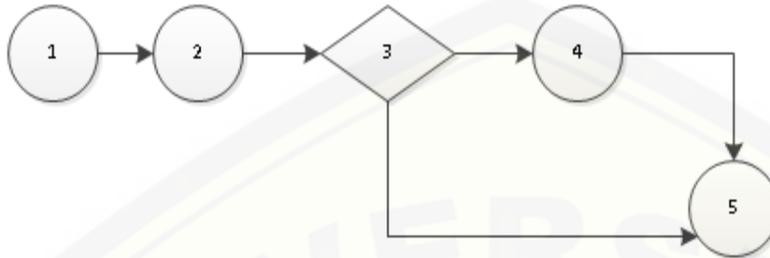
180     function add_sekolah(){ 1
181         $data = array(
182             'id_kecamatan' => $this->input->post('nama_kecamatan'),
183             'nama_sekolah' => $this->input->post('sekolah')
184         );
185         $logic = $this->m_sekolah->add_sekolah($data);
186         if($logic){
187             redirect('c_guru/lihat_data_sekolah'); 4
188         }
189     } 5

```

#### a. Diagram Alir Tambah Data Sekolah



b. Grafik Alir Tambah Data Sekolah



c. Penghitungan Kompleksitas Siklomatik

Kompleksitas siklomatik dari tambah data sekolah dapat diperoleh dengan perhitungan berikut :

$$V(G) = E - N + 2 = 5 - 5 + 2 = 2$$

d. Basis set

Basis set tambah data sekolah menghasilkan 2 jalur independent dari perhitungan skompleksitas siklomatik, yaitu :

Jalur 1 : 1-2-3-5

Jalur 2 : 1-2-3-4-5

e. Test Case

*Test Case:* tambah data sekolah

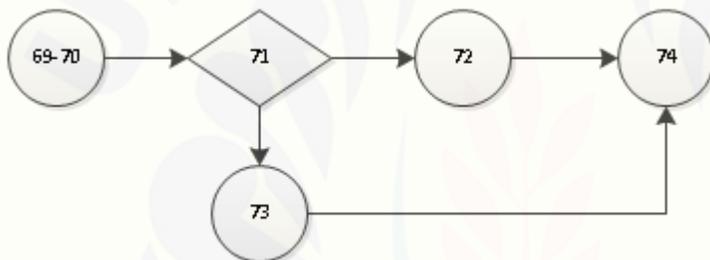
No	Pengujian	Jalur	Kesimpulan	Target yang diharapkan
1	Menambah data sekolah	1-2-3-5	[✓] berhasil [ ] gagal	Data berhasil ditambahkan ke dalam database
2	Menambah data sekolah	1-2-3-4-5	[ ] berhasil [✓] gagal	data sekola tidak ditambahkan ke dalam database

2. Listing Program Mengubah Data Sekolah

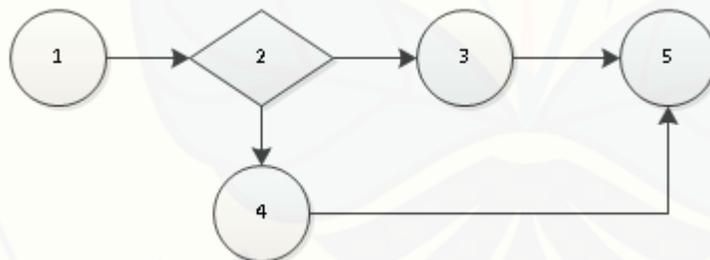
```

69 function ubah_data_sekolah($data) { 1
70     $this->db->where('id_sekolah', $this->session->userdata('id_sekolah'));
71     2 if($this->db->update('tbl_sekolah', $data)) {
72         return true; 3
73     }else{ 4
74         return false;
75     }
76 } 5
    
```

a. Diagram Alir Mengubah Data Sekolah



b. Grafik Alir Mengubah Data Sekolah



c. Penghitungan Kompleksitas Siklomatik

Kompleksitas siklomatik dari Mengubah Data Sekolah dapat diperoleh dengan perhitungan berikut :

$$V(G) = E - N + 2 = 5 - 5 + 2 = 2$$

**d.** Basis set

*Basis set* tambah data *user* menghasilkan 2 jalur independent dari perhitungan skompleksitas siklomatik, yaitu :

Jalur 1 : 1-2-3-5

Jalur 2 : 1-2-3-4-5

**e.** Test Case

*Test Case: Tambah Data User*

No	Pengujian	Jalur	Kesimpulan	Target yang diharapkan
1	Mengubah data sekolah	1-2-3-5	<input checked="" type="checkbox"/> berhasil <input type="checkbox"/> gagal	Data sekolah berhasil diubah
2	Mengubah data sekolah	1-2-3-4-5	<input type="checkbox"/> berhasil <input checked="" type="checkbox"/> gagal	Data sekolah tidak berhasil diubah

**C.3 Pengujian *White Box* Modul III Implementasi Metode**

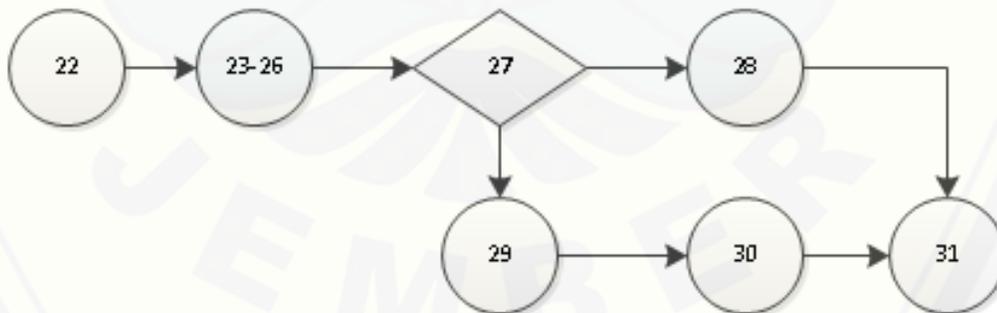
Nama Modul	Analisis Data (,mengubah nilai atribut input kriteria, hapus kriteria,Input factor, hapus factor, input nilai target, , melihat hasil perhitungan)
File	c_atribut, c_faktor, c_kriteria, m_atribut

1. Listing Program mengubah nilai atribut

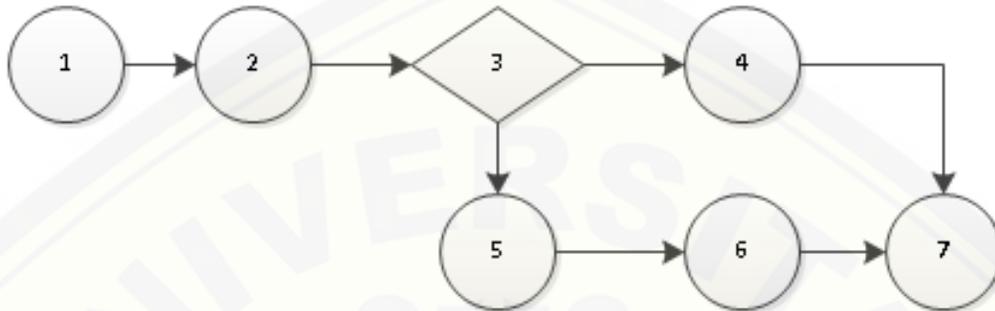
```

22 function ubah_atribut($persen, $id){
23     $persen_2 = 100 - $persen;
24     $this->db->trans_begin();
25     $this->db->query("UPDATE tb_atribut SET presentase = ".$persen." WHERE id_atribut = ".$id."");
26     $this->db->query("UPDATE tb_atribut SET presentase = ".$persen_2." WHERE id_atribut <> ".$id."");
27     if($this->db->trans_status() === FALSE){
28         $this->db->trans_rollback();
29     }else{
30         $this->db->trans_commit();
31     }
32 }
    
```

a. Diagram Alir mengubah nilai atribut



b. Grafik Alir mengubah nilai atribut



c. Penghitungan Kompleksitas Siklomatik

Kompleksitas siklomatik dari mengubah nilai atribut dapat diperoleh dengan perhitungan berikut :

$$V(G) = E - N + 2 = 7 - 7 + 2 = 2$$

d. Basis set

Basis set mengubah nilai atribut menghasilkan 2 jalur independent dari perhitungan skompleksitas siklomatik, yaitu :

Jalur 1 : 1-2-3-4-7

Jalur 2 : 1-2-3-5-6-7

e. Test Case

*Test Case:* mengubah nilai atribut

No	Pengujian	Jalur	Kesimpulan	Target yang diharapkan
1	Mengubah nilaiatribut	1-2-3-4-7	[√] berhasil [ ]gagal	Data berhasil diubah
2	Mengubah nilai atribut	1-2-3-5-6-7	[ ] berhasil [√]gagal	Data nilai atribut tidak berhasil diubah

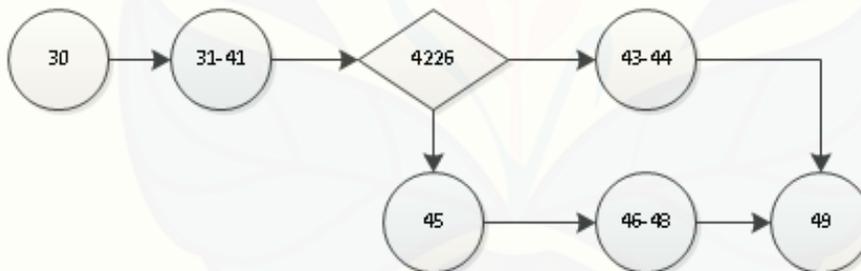
2. Listing Program input faktor

```

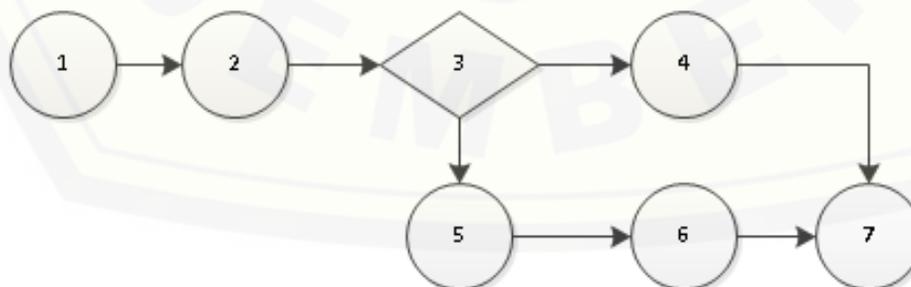
30 public function add_faktor(){ 1
31     $nama_faktor = $this->input->post('nama_faktor');
32     $atribut = $this->input->post('atribut');
33     $kriteria = $this->input->post('kriteria');
34     2 $tipeisi = $this->input->post('tipeisi');
35     $data = array(
36         'nama_faktor' => $nama_faktor,
37         'id_atribut' => $atribut,
38         'id_kriteria' => $kriteria,
39         'tipeisi' => $tipeisi
40     );
41     $sukses = $this->m_faktor->add_faktor($data);
42     3 if ($sukses) {
43         $this->session->set_flashdata('message', $this->succes_message); 4
44         redirect('c_faktor/input_faktor');
45     } else {
46     5 $this->session->set_flashdata('message', $this->alert_message);
47         6 redirect('c_faktor/input_faktor');
48     }
49     7

```

a. Diagram Alir input faktor



b. Grafik Alir input faktor



### Penghitungan Kompleksitas Siklomatik

Kompleksitas siklomatik dari input faktor dapat diperoleh dengan perhitungan berikut :

$$V(G) = E - N + 2 = 7 - 7 + 2 = 2$$

#### c. Basis set

*Basis set* input faktor menghasilkan 2 jalur independent dari perhitungan skompleksitas siklomatik, yaitu :

Jalur 1 : 1-2-3-4-7

Jalur 2 : 1-2-3-5-6-7

#### d. Test Case

*Test Case*: input faktor

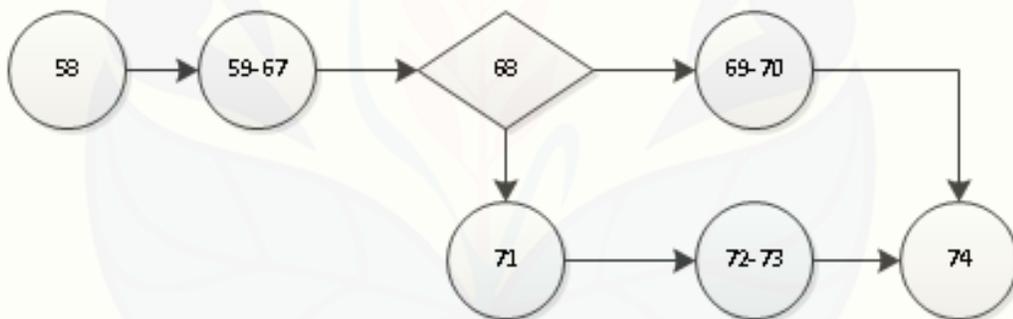
No	Pengujian	Jalur	Kesimpulan	Target yang diharapkan
1	Input data factor	1-2-3-4-7	[√] berhasil [ ] gagal	Data factor berhasil dimasukkan
2	Input data faktor	1-2-3-5-6-7	[√] berhasil [ ] gagal	Data factor tidak berhasil ditambahkan

3. Listing Program tambah skala

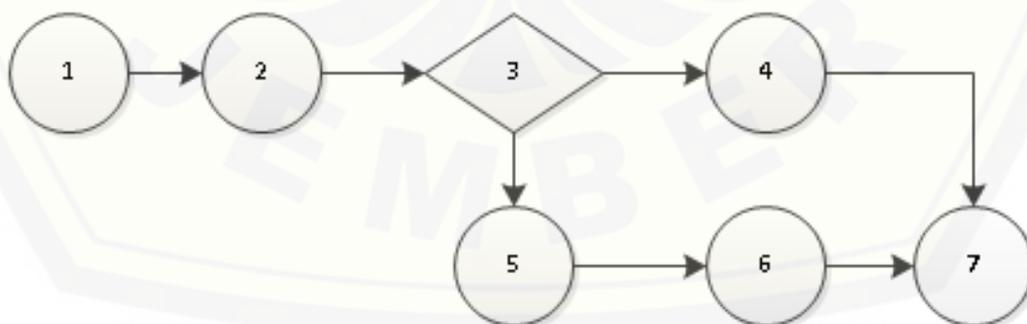
```

58 public function add_skala() { 1
59     $id_faktor = $this->input->post('id_faktor');
60     $skala_faktor = $this->input->post('skala_faktor');
61     $bobot_skala_faktor = $this->input->post('bobot_skala_faktor');
62     2 $data = array(
63         'nama_skala_faktor' => $skala_faktor,
64         'id_faktor' => $id_faktor,
65         'bobot' => $bobot_skala_faktor
66     );
67     $sukses = $this->m_faktor->add_skala_faktor($data);
68     3 if ($sukses) {
69         $this->session->set_flashdata('message', $this->sukses_message); 4
70         redirect('c_faktor/input_skala/'.$id_faktor);
71     } 5 else {
72         $this->session->set_flashdata('message', $this->alert_message); 6
73         redirect('c_faktor/input_skala/'.$id_faktor);
74     }
75     7 }
    
```

a. Diagram Alir tambah skala



b. Grafik Alir tambah skala



c. Penghitungan Kompleksitas Siklomatik

Kompleksitas siklomatik dari tambah skala dapat diperoleh dengan perhitungan berikut :

$$V(G) = E - N + 2 = 7 - 7 + 2 = 2$$

d. *Basis set*

*Basis set* hapus kriteria menghasilkan 2 jalur independent dari perhitungan kompleksitas siklomatik, yaitu :

Jalur 1 : 1-2-3-4-7

Jalur 2 : 1-2-3-5-6-7

e. Test Case

*Test Case*: tambah skala

No	Pengujian	Jalur	Kesimpulan	Target yang diharapkan
1	Menghapus data kriteria	1-2-3-4-7	<input checked="" type="checkbox"/> berhasil <input type="checkbox"/> gagal	Dat kriteria berhasil ditambahkan
2	Menghapus data kriteria	1-2-3-5-6-7	<input type="checkbox"/> berhasil <input checked="" type="checkbox"/> gagal	Data kriteria tidak berhasil ditambahkan

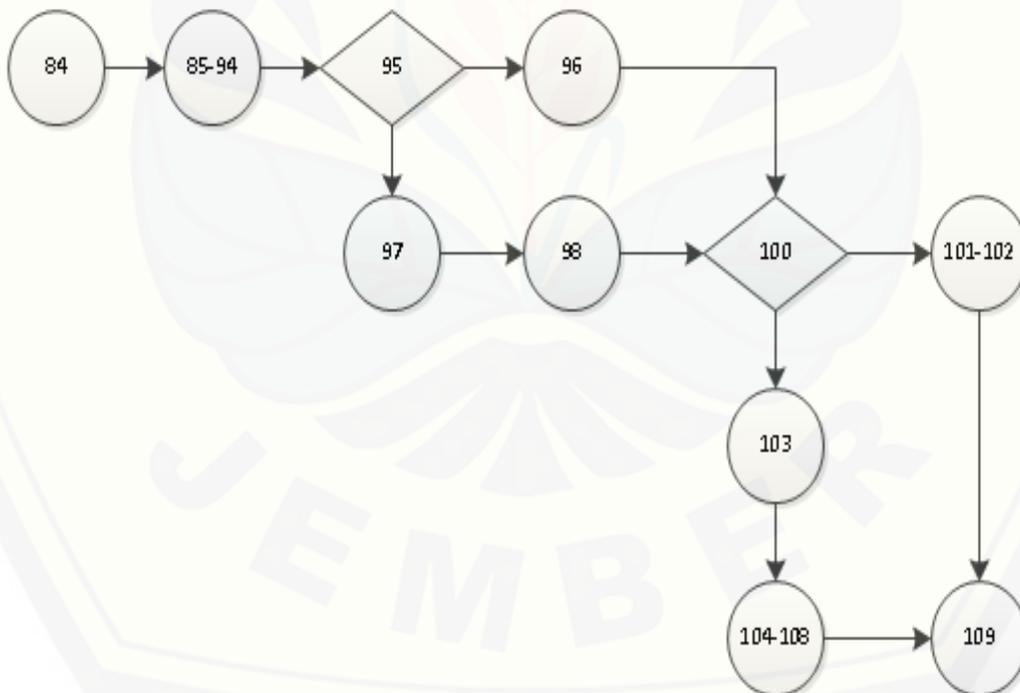
4. Listing Program input nilai target

```

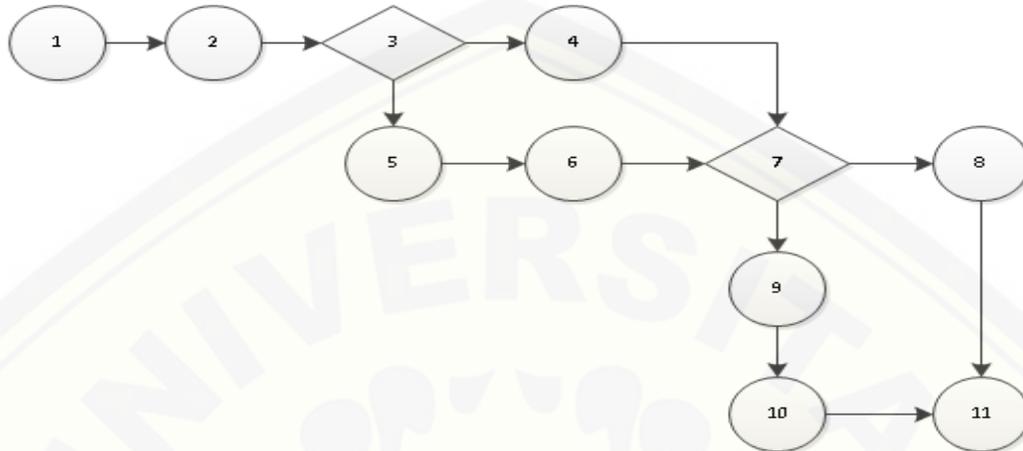
84 public function add_target(){
85     $sukses = false;
86     $id_faktor = $this->input->post('id_faktor');
87     $target_faktor = $this->input->post('target');
88     $id_sekolah = $this->session->userdata('id_sekolah');
89     $data = array(
90         'id_faktor' => $id_faktor,
91         'target' => $target_faktor,
92         'id_sekolah' => $this->session->userdata('id_sekolah')
93     );
94     $cek_data_target = $this->m_faktor->cek_target($id_sekolah, $id_faktor);
95     if($cek_data_target){
96         $sukses = $this->m_faktor->edit_target_sekolah($id_sekolah, $id_faktor, $data);
97     }else {
98         $sukses = $this->m_faktor->add_target_sekolah($data);
99     }
100     if ($sukses) {
101         $this->session->set_flashdata('message', $this->sukses_message);
102         redirect('c_faktor/input_faktor');
103     } else {
104         $this->session->set_flashdata('message', $this->alert_message);
105         redirect('c_faktor/input_faktor');
106     }
107 }
108
109

```

a. Diagram alir input nilai target



b. Grafik Alir input nilai target



c. Penghitungan Kompleksitas Siklomatik

Kompleksitas siklomatik dari input nilai target dapat diperoleh dengan perhitungan berikut :

$$V(G) = E - N + 2 = 12 - 11 + 2 = 3$$

d. Basis set

Basis set input nilai target menghasilkan 2 jalur independent dari perhitungan skompleksitas siklomatik, yaitu :

Jalur 1 : 1-2-3-4-7-8-11

Jalur 2 : 1-2-3-5-6-7-8-11

Jalur 3 : 1-2-3-5-6-7-9-10-11

e. Test Case

Test Case: input nilai target				
No	Pengujian	Jalur	Kesimpulan	Target yang diharapkan
1	User memasukkan nilai target	1-2-3-4-7-8-11	[√] berhasil [ ] gagal	Nilai target berhasil

				ditambahkan
2	User memasukkan nilai target	1-2-3-5-6-7-8-11	[√] berhasil [ ] gagal	Nilai target tidak berhasil ditambahkan
3	User memasukkan nilai target	1-2-3-5-6-7-9-10-11	[√] berhasil [ ] gagal	Sistem akan kembali ke halaman sebelumnya

5. Listing Program lihat hasil perhitungan

```

336 function lihat_hasil_perhitungan() {
337     // $this->data['profile'] = $this->m_guru->profile_matching();
338     $this->data['guru'] = $this->m_guru->view_tabel_guru();
339     $this->load->view('hasil_perhitungan', $this->data);
340 }
    
```

a. Diagram Alir lihat hasil perhitungan



b. Grafik Alir lihat hasil perhitungan



c. Penghitungan Kompleksitas Siklomatik

Kompleksitas siklomatik dari lihat hasil perhitungan dapat diperoleh dengan perhitungan berikut :

$$V(G) = E - N + 2 = 2 - 3 + 2 = 1$$

d. *Basis set*

*Basis set* lihat hasil perhitungan menghasilkan 2 jalur independent dari perhitungan skompleksitas siklomatik, yaitu :

Jalur 1 : 1-2-3

e. *Test Case*

*Test Case*: lihat hasil perhitungan

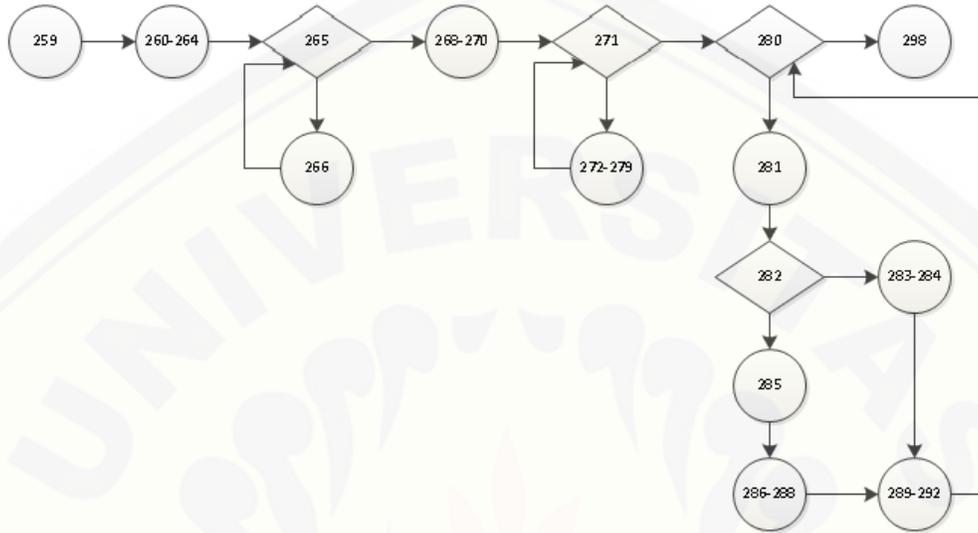
No	Pengujian	Jalur	Kesimpulan	Target yang diharapkan
1	Melihat hasil perhitungan		[√] berhasil [ ] gagal	Data perhitungan nilai berhasil dilihat

6. Listing Program proses input nilai

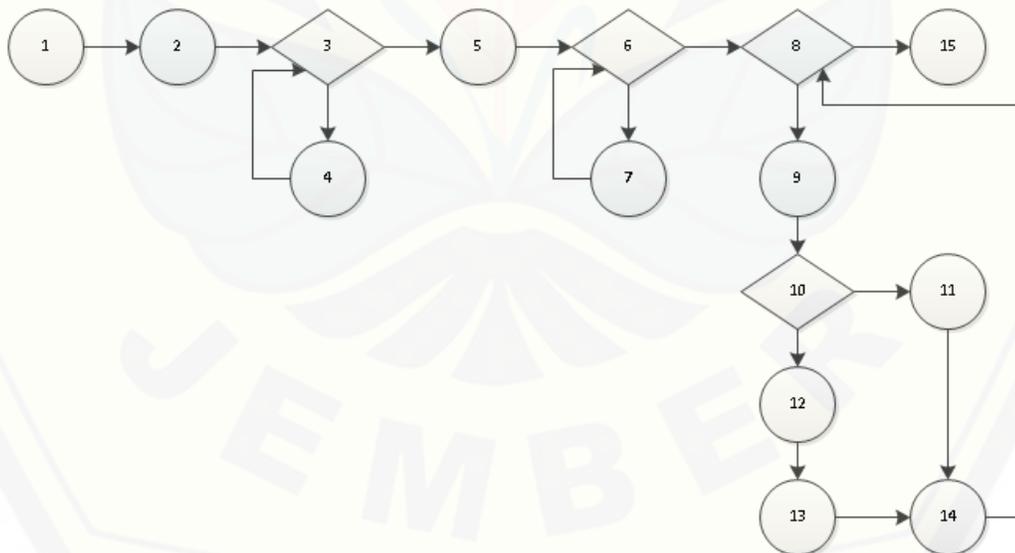
```

259 function proses_input_nilai() {
260     $faktor = $this->m_faktor->select_faktor_nilai();
261     $nip = $this->input->post('nip');
262     $res_jarak = $this->m_guru->ambil_pilihan($nip);
263     $res_guru = $this->m_guru->view_detail_tabel_guru($nip);
264     $alamat;
265     foreach($res_guru as $rows) {
266         $alamat = $rows['alamat_guru'];
267     }
268     $nilai = "";
269     $skala = "";
270     // Jarak
271     foreach($res_jarak as $row) {
272         $param_jarak = $row['jarak'];
273         $skala_jarak = $this->cari_skala(1, $param_jarak);
274         $skala_wilayah = $this->get_skala_wilayah($alamat, $row['alamat']);
275         $this->m_faktor->skala_jarak_wilayah($row['id_pilihan'], $skala_jarak, $skala_wilayah);
276     }
277     // Status Wilayah
278     // Input.
279     foreach($faktor as $row) {
280         $tipe = $this->input->post('tipe_'.$row['id_faktor']);
281         if($tipe == 't') {
282             $nilai = $this->input->post('t_'.$row['id_faktor']);
283             $skala = $this->cari_skala($row['id_faktor'], $nilai);
284         } else {
285             $nilai = NULL;
286             $skala = $this->input->post('s_'.$row['id_faktor']);
287         }
288     }
289     $data_insert = array(
290         'nip' => $nip,
291         'nilai' => $nilai,
292         'id_skala_faktor' => $skala
293     );
294     print_r($data_insert);
295     echo "<br />";
296     $this->m_faktor->insert_nilai($data_insert);
297     endforeach;
298     redirect('c_guru/input_nilai_guru');
    
```

a. Diagram Alir proses input nilai



b. Grafik Alir proses input nilai



c. Penghitungan Kompleksitas Siklomatik

Kompleksitas siklomatik dari proses proses input nilai dapat diperoleh dengan perhitungan berikut :

$$V(G) = E - N + 2 = 18 - 15 + 2 = 5$$

d. *Basis set*

*Basis set* proses input nilai menghasilkan 2 jalur independent dari perhitungan skompleksitas siklomatik, yaitu :

Jalur 1 : 1-2-3-5-6-8-15

Jalur 2 : 1-2-3-4-3-5-6-8-15

Jalur 3 : 1-2-3-5-6-7-6-8-15

Jalur 4 : 1-2-3-4-3-5-6-8-9-10-11-14-15

Jalur 5 : 1-2-3-5-6-7-6-8-9-10-12-13-14-15

e. *Test Case*

<i>Test Case: proses input nilai</i>				
No	Pengujian	Jalur	Kesimpulan	Target yang diharapkan
1	Jalur ketika user melakukan input nilai jarak	1-2-3-5-6-8-15	<input checked="" type="checkbox"/> berhasil <input type="checkbox"/> gagal	Sistem akan melanjutkan ke proses berikutnya
2	Jalur ketika user melakukan input nilai jarak	1-2-3-4-3-5-6-8-15	<input type="checkbox"/> berhasil <input checked="" type="checkbox"/> gagal	Sistem akan kembali ke proses sebelumnya karena data belum berhasil
3	Jalur ketika user melakukan input skala faktor	1-2-3-5-6-7-6-8-15	<input checked="" type="checkbox"/> berhasil <input type="checkbox"/> gagal	Sistem akan melakukan proses selanjutnya
4	Jalur ketika user melakukan input skala faktor	1-2-3-4-3-5-6-8-9-10-11-14-15	<input type="checkbox"/> berhasil <input checked="" type="checkbox"/> gagal	Sistem akan kembali ke proses sebelumnya karena data belum

				berhasil
5	Perhitungan nilai	1-2-3-5-6-7- 6-8-9-10-12- 13-14-15	[√] berhasil [ ] gagal	Data berhasil dihitung

#### C.4 Pengujian *White Box* Modul IV Manajemen Data Guru

Nama Modul	Manajemen Data Guru (menambah data guru, mengubah data guru, input nilai guru, input jenis guru, hapus data guru,)
File	c_guru

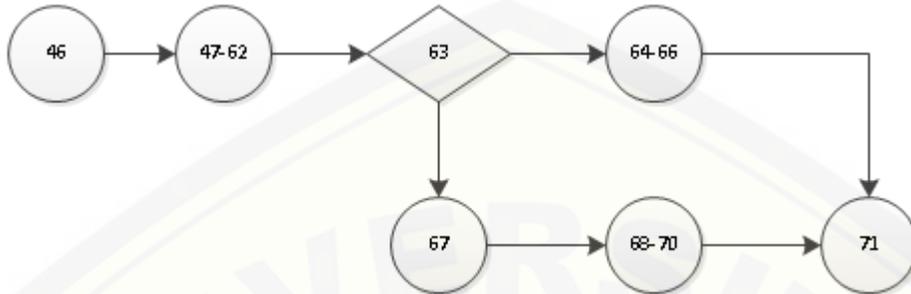
##### 1. Listing Program input biodata guru

```

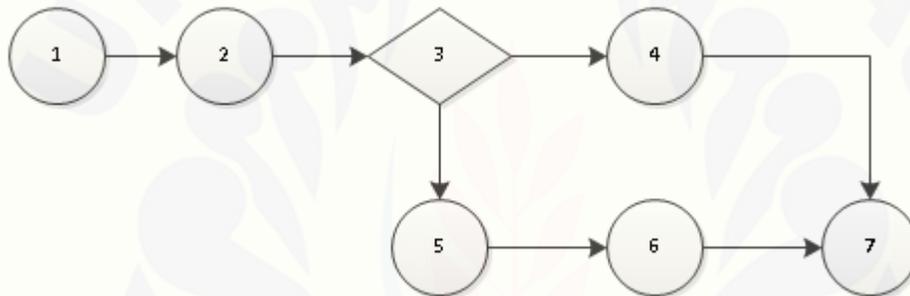
46 1 function add_guru() {
47     $data = array(
48         'nip' => $this->input->post('nip'),
49         'nama_guru' => $this->input->post('nama_guru'),
50         'alamat_guru' => $this->input->post('alamat_guru'),
51         'id_kota' => $this->input->post('kota_guru'),
52         'tempat_lahir_guru' => $this->input->post('tempat_lahir_guru'),
53     2 'tgl_lhr_guru' => $this->input->post('tgl_lhr_guru'),
54         'id_agama' => $this->input->post('agama'),
55         'id_gender' => $this->input->post('id_gender'),
56         'id_pendidikan' => $this->input->post('id_pendidikan'),
57         'id_jenis_guru' => $this->input->post('id_jenis_guru'),
58         'lat' => $this->input->post('lat'),
59         'long' => $this->input->post('long'),
60         'username' => $this->session->userdata('username')
61     );
62     $sukses = $this->m_guru->add_data_guru($data);
63     3 if ($sukses){
64         $this->session->set_flashdata('message', $this->succes_message); 4
65         redirect('c_guru/input_biodata_guru');
66         echo '';
67     5 } else {
68         $this->session->set_flashdata('message', $this->alert_message); 6
69         redirect('c_guru/input_biodata_guru');
70     }
71     7 }

```

a. Diagram Alir input biodata guru



b. Grafik Alir input biodata guru



c. Penghitungan Kompleksitas Siklomatik

Kompleksitas siklomatik dari input biodata guru dapat diperoleh dengan perhitungan berikut :

$$V(G) = E - N + 2 = 7 - 7 + 2 = 2$$

d. Basis set

Basis set input biodata guru menghasilkan 2 jalur independent dari perhitungan skompleksitas siklomatik, yaitu :

Jalur 1 : 1-2-3-4-7

Jalur 2 : 1-2-3-5-6-7

e. Test Case

Test Case: input biodata guru

No	Pengujian	Jalur	Kesimpulan	Target yang diharapkan
----	-----------	-------	------------	------------------------

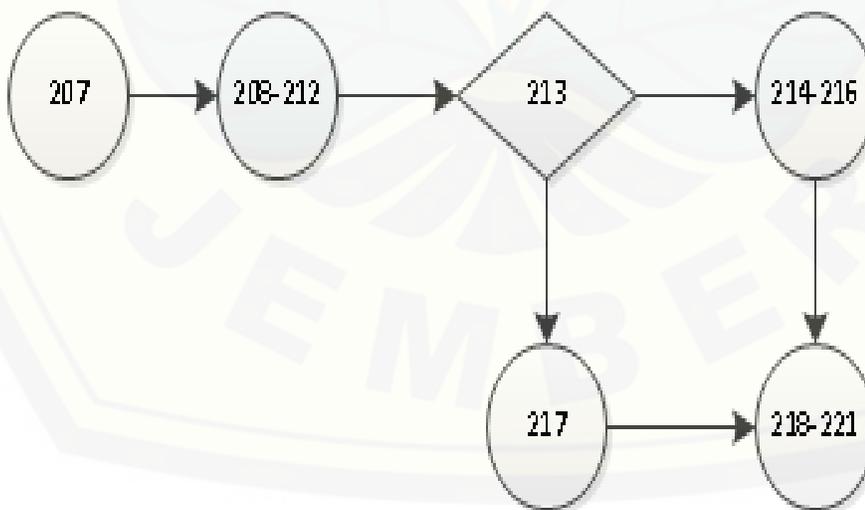
1	User menambahkan data biodata guru	1-2-3-4-7	<input checked="" type="checkbox"/> berhasil <input type="checkbox"/> gagal	Data biodata guru berhasil ditambahkan
2	User menambahkan data biodata guru	1-2-3-5-6-7	<input type="checkbox"/> berhasil <input checked="" type="checkbox"/> gagal	Data biodata guru gagal ditambahkan

2. Listing Program tambah jenis guru

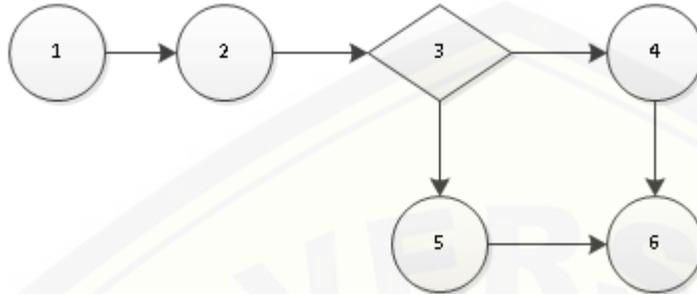
```

207 1 function add_jenis_guru(){
208     $jenis_guru = $this->input->post('jenis_guru');
209     $data = array(
210 2     'jenis_guru' => $jenis_guru
211     );
212     $sukses = $this->m_guru->add_jenis_guru($data);
213 3     if ($sukses === TRUE) {
214         $this->session->set_flashdata('message', $this->succes_message); 4
215         redirect('c_guru/input_jenis_guru');
216         echo '';
217 5     } else {
218         $this->session->set_flashdata('message', $this->alert_message);
219         redirect('c_guru/input_jenis_guru'); 6
220     }
221 }
    
```

a. Diagram Alir tambah jenis guru



b. Grafik Alir tambah jenis guru



c. Penghitungan Kompleksitas Siklomatik

Kompleksitas siklomatik dari tambah jenis guru dapat diperoleh dengan perhitungan berikut :

$$V(G) = E - N + 2 = 6 - 6 + 2 = 2$$

d. Basis set

Basis set tambah jenis guru menghasilkan 2 jalur independent dari perhitungan skompleksitas siklomatik, yaitu :

Jalur 1 : 1-2-3-4-6

Jalur 2 : 1-2-3-5-6

e. Test Case

*Test Case:* tambah jenis guru

No	Pengujian	Jalur	Kesimpulan	Target yang diharapkan
1	Menambahkan data biodata guru	1-2-3-4-6	[√] berhasil [ ] gagal	Data jenis guru berhasil ditambahkan
2	Menambahkan data biodata guru	1-2-3-5-6	[ ] berhasil [√]gagal	Data jenis guru tidak berhasil ditambahkan

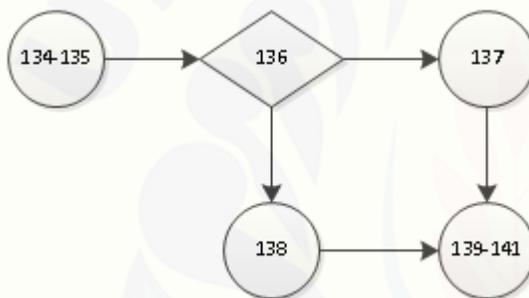
## 3. Listing Program hapus guru

```

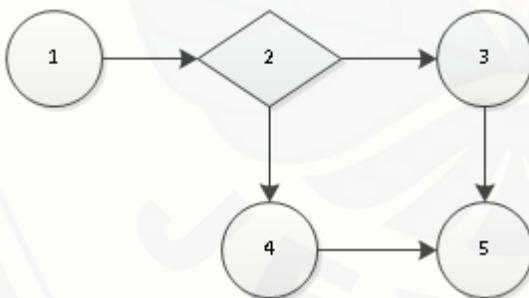
134 1 function hapusguru($id) {
135     $sukses = $this->m_guru->hapus_guru($id);
136     2 if ($sukses == true) {
137         redirect('c_guru/tabel_daftar_guru'); 3
138     4 } else {
139         redirect('c_guru/tabel_daftar_guru');
140     }
141     5 }

```

## a. Diagram Alir hapus guru



## b. Grafik Alir hapus guru



## c. Penghitungan Kompleksitas Siklomatik

Kompleksitas siklomatik dari hapus guru dapat diperoleh dengan perhitungan berikut :

$$V(G) = E - N + 2 = 5 - 5 + 2 = 2$$

d. Basis set

Basis set hapus guru menghasilkan 2 jalur independent dari perhitungan skompleksitas siklomatik, yaitu :

Jalur 1 : 1-2-3-5

Jalur 2 : 1-2-4-5

e. Test Case

Test Case: hapus guru

No	Pengujian	Jalur	Kesimpulan	Target yang diharapkan
1	Menghapus data biodata guru	1-2-3-5	[√] berhasil [ ] gagal	Data guru berhasil dihapus
2	Menghaspus data biodata guru	1-2-4-5	[√] berhasil [ ] gagal	Data guru tidak berhasil dihapus

4. Listing Program penempatan guru

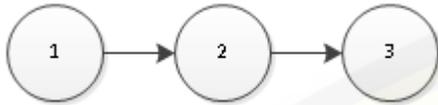
```

355 1 function tempatkan_guru($id_sekolah, $id_jenis_guru) {
356     $this->data['id_sekolah'] = $id_sekolah;
357     $this->data['id_jenis_guru'] = $id_jenis_guru;
358     $this->data['sekolah'] = $this->m_sekolah->select_sekolah($id_sekolah);
359     2 $this->data['jenis_guru'] = $this->m_guru->jenis_guru($id_jenis_guru);
360     $this->data['peringkat'] = $this->m_sekolah->peringkat_sekolah_jenis($id_sekolah, $id_jenis_guru);
361     $this->load->view('pilih_guru',$this->data);
362 3
    
```

a. Diagram Alir penempatan guru



## b. Grafik Alir penempatan guru



## c. Penghitungan Kompleksitas Siklomatik

Kompleksitas siklomatik dari penempatan guru dapat diperoleh dengan perhitungan berikut :

$$V(G) = E - N + 2 = 2 - 3 + 2 = 1$$

## d. Basis set

*Basis set* penempatan guru menghasilkan 2 jalur independent dari perhitungan skompleksitas siklomatik, yaitu :

Jalur 1 : 1-2-3

e. *Test Case*

*Test Case*: penempatan guru

No	Pengujian	Jalur	Kesimpulan	Target yang diharapkan
1	Penempatan guru	1-2-3	[√] berhasil [ ] gagal	Penempatan data guru

**LAMPIRAN D. PENGUJIAN BLACK BOX SISTEM**C.1 Pengujian *Black Box* Modul 1 Manajemen Data Pengguna1. *View* Data Pengguna

Melihat Data <i>User</i>		
Aktor	Sistem	Status Berhasil    Gagal
1 Memilih menu <i>user</i> pilih <i>user</i> pengguna	2 Menampilkan halaman data pengguna	✓
3 Memilih menu view data pengguna	4 Menampilkan halaman detail data pengguna	✓
5 Klik Tutup	6 Menampilkan kembali halaman data pengguna	✓

2. *Edit* Data Pengguna

Mengubah Data <i>User</i>		
Aktor	Sistem	Status Berhasil    Gagal
1 Memilih menu <i>user</i> pilih <i>user</i> pengguna	2 Menampilkan halaman data pengguna	✓
3 Memilih menu edit pada data yang akan diubah	4 Menampilkan form edit data pengguna	✓
5 Mengubah data pada form <i>edit</i> data <i>user</i>	7 Data pengguna Berhasil diubah	✓
6 Klik Ok		
Data yang dimasukkan tidak valid		
5 Data yang diubah tidak lengkap atau <i>username</i> sudah digunakan		

6 Klik Ok

7 Menampilkan alert pada form yang datanya tidak lengkap ✓

### 3. Hapus Data Pengguna

#### Menghapus Data Pengguna

Aktor	Sistem	Status Berhasil	Gagal
1 Memilih menu <i>user</i> pilih <i>user</i> pengguna		✓	
	2 Menampilkan halaman data pengguna	✓	
3 Memilih menu hapus pada data yang akan dihapus		✓	
4 Klik Hapus			
	5 Menampilkan alert “Apakah anda yakin ingin menghapus”?	✓	
6 Klik OK			
	7 Data pengguna berhasil dihapus	✓	
Apabila Klik Cancel			
6 Klik Cancel			
	7 Kembali ke halaman data pengguna dan data tidak dihapus	✓	

## 4. Tambah Data Pengguna

Menambah Data <i>User</i>		
Aktor	Sistem	Status Berhasil    Gagal
1    Memilih menu <i>user</i> pilih <i>user</i> pengguna	2    Menampilkan halaman tambah pengguna	✓
	3    Menampilkan form tambah data pengguna	✓
	5    Mengisi data pada form tambah data	✓
6    Klik Tambah	7    Data pengguna Berhasil ditambahkan	✓
Jika memasukkan data dengan format yang salah		
5    Mengisi form data tidak lengkap atau <i>username</i> sudah digunakan		✓
6    Klik Simpan	7    Menampilkan alert pada form yang datanya tidak lengkap	✓
Jika klik reset		
6    Klik Reset	7    Data yang dimasukkan pada form akan terhapus dan tidak tersimpan	✓

C.2 Pengujian *Black Box* Modul II Manajemen Data Sekolah

## 1. Tambah Sekolah

Menambah data sekolah		
Aktor	Sistem	Status Berhasil   Gagal
1	Memilih menu sekolah pilih ubah data sekolah	✓
2	Menampilkan halaman nilai atribut	✓
3		✓
4	Klik Submit	✓
5	Data sekolah berhasil diubah	✓
Jika memasukkan data dengan format yang salah		
3	Memasukkan data sekolah kosong	✓
4	Klik Submit	✓
6	Muncul alert pada form pengisian dan data tidak berhasil ditambahkan	✓

## 2. Input Nilai Target Sekolah

<i>Input nilai target sekolah</i>			
Aktor	Sistem	Status Berhasil    Gagal	
1	Memilih menu kriteria pilih submenu beri target sekolah	✓	
	2	Menampilkan tabel factor yang akan dimasukkan nilai target	✓
3	Pilih factor yang akan ditambahkan nilai target	✓	
4	Klik Input target		
	5	Menampilkan halaman input skala faktor	✓
6	Mengisi nilai target factor	✓	
7	Klik Submit		
	8	Data berhasil ditambahkan	✓
<u>Jika memasukkan data dengan format yang salah</u>			
6	Mengisi nilai kosong atau tidak lengkap	✓	
7	Klik Submit		
	8	Menampilkan alert pada form yang belum lengkap	✓

## 3. Ubah Data Sekolah

Ubah data sekolah			
Aktor	Sistem	Status Berhasil    Gagal	
1	Memilih menu sekolah pilih submenu ubah data sekolah	✓	
	2	Menampilkan form ubah data sekolah	✓
3	Merubah data sekolah sesuai dengan kebutuhan	✓	
4	Klik Submit		
	5	Data sekolah berhasil diubah	✓
Jika memasukkan data dengan format yang salah			
3	Data yang diubah tidak lengkap	✓	
4	Klik Submit		
	5	Menampilkan alert pada form yang tidak lengkap	✓

C.3 Pengujian *Black Box* Modul III Implementasi Metode

1. Mengubah nilai presentasi atribut			
Mengubah nilai presentasi atribut			
Aktor	Sistem	Status	
		Berhasil    Gagal	
1	Memilih menu kriteria pilih submenu kriteria	✓	
	2	Menampilkan form ubah data sekolah	✓
3	Pilih atribut yang akan diubah presentasinya	✓	
4	Klik ubah presentase		
	5	Menampilkan form ubah nilai presentase	✓
6	Memasukkan nilai presentase	✓	
7	Klik Ok	✓	
	8	Data berhasil diubah sesuai dengan presentase yang dimasukkan	
Jika memasukkan data dengan format yang salah			
3	Memasukkan nilai kosong	✓	
7	Klik Ok		
	8	Menampilkan alert bahwa data belum mencapai 100 persen	✓

2. *Input* Kriteria

<i>Input</i> Kriteria			
Aktor	Sistem	Status	
		Berhasil   Gagal	
1	Memilih menu kriteria pilih submenu atribut	✓	
	2	Menampilkan halaman tabel kriteia dan form tambah kriteria	✓
3	Masukkan kriteria	✓	
4	Klik submit	✓	
	5	Kriteria berhasil ditambahkan dan muncul pada tabel kriteria	✓
Jika memasukkan data dengan format yang salah			
6	Memasukkan data kriteria kosong	✓	
7	Klik Ok	✓	
	8	Muncul alert pada form pengisian dan data tidak berhasil ditambahkan	✓

## 3. Hapus Kriteria

Hapus kriteria			
Aktor	Sistem	Status Berhasil   Gagal	
1	Memilih menu kriteria pilih submenu kriteria	✓	
	2	Menampilkan halaman tabel kriteria	✓
3	Memilih kriteria yang akan dihapus	✓	
4	Klik hapus		
	5	Muncul alert apakah yakin akan menghapus	✓
6	Klik Ok		
	7	Data berhasil dihapus	✓
Jika memasukkan data dengan format yang salah			
6	Klik cancel	✓	
	7	Kembali pada halaman tabel kriteria dan data tidak berhasil dihapus	✓

4. *Input* Faktor

<i>Input</i> faktor			
Aktor	Sistem	Status Berhasil    Gagal	
1	Memilih menu kriteria pilih submenu faktor	✓	
	2	Menampilkan halaman input factor dan tabel data faktor	✓
3	Masukkan data faktor apabila data factor terdapat skala maka memilih tipe isian data "ambil dari skala"	✓	
4	Klik Submit		
	5	Data Faktor berhasil ditambahkan dan muncul pada tabel faktor	✓
6	Klik menu Input Skala pada tabel data faktor data yang baru ditambahkan		
	7	Menampilkan halaman input skala dan tabel skala	✓
8	Masukkan skala sesuai data yang dibutuhkan	✓	
9	Klik submit		
	10	Data skala berhasil ditambahkan	✓

Jika memasukkan data dengan format yang salah

3	Masukan data faktor, data faktor berupa data biasa atau tidak berskala	✓
4	Klik submit	7 Data berhasil ditambahkan
		✓

5. Hapus Faktor

Hapus Faktor

Aktor	Sistem	Status	Berhasil	Gagal
1	Memilih menu kriteria pilih submenu faktor	✓		
	2	Menampilkan halaman tabel faktor	✓	
3	Memilih faktor yang akan dihapus	✓		
4	Klik Hapus	5	Muncul alert apakah yakin akan menghapus	✓
6	Klik Ok	7	Data faktor berhasil dihapus	✓

Jika memasukkan data dengan format yang salah

6	Klik batal	✓
	7 Kembali pada halaman tabel faktor dan data tidak berhasil dihapus	✓

6. Lihat Hasil Perhitungan

Lihat hasil perhitungan

Aktor	Sistem	Status	Berhasil	Gagal
1	Memilih menu metode pilih submenu hasil perhitungan	✓		
	2	Menampilkan halaman tabel hasil perhitungan	✓	
3	Pilih submenu nilai <i>core</i> dan <i>secondary</i>	✓		

- |   |                           |   |  |   |
|---|---------------------------|---|--|---|
| 4 | Klik Hapus                | 5 | Menampilkan halaman tabel hasil perhitungan <i>core</i> dan <i>secondary</i> | ✓ |
| 6 | Pilih submenu nilai total | 7 | Menampilkan tabel hasil perhitungan keseluruhan nilai guru                   | ✓ |

#### 7. Penempatan Guru

##### Penempatan guru

Aktor	Sistem	Status	
		Berhasil	Gagal
1	Memilih menu metode pilih submenu penempatan	✓	
2	Menampilkan halaman total kebutuhan guru persekolah	✓	
3	Memilih sekolah yang akan dilakukan penempatan guru	✓	
4	Klik lakukan penempatan		
5	Menampilkan daftar guru yang memilih sekolah tersebut	✓	
6	Memilih guru yang akan ditempatkan		
7	Klik pilih guru	✓	
8	Menampilkan alert “pilihan tidak dapat dirubah. Apakah anda yakin?”		
9	Klik Ok		
10	Guru berhasil ditempatkan		
Jika memasukkan data dengan format yang salah			
9	Klik batal	✓	
10	Kembali pada halaman penempatan	✓	

## 8. Lihat Peringkat Guru per Sekolah

Lihat peringkat guru per sekolah			
Aktor	Sistem	Status	
		Berhasil	Gagal
1	Memilih menu metode pilih submenu peringkat persekolah	✓	
	2	Menampilkan halaman tabel pilihan sekolah	✓
3	Pilih sekolah yang akan dilihat	✓	
4	Klik submit		
	5	Menampilkan data peringkat guru pada sekolah tersebut	✓

C.4 Pengujian *Black Box* Modul IV Manajemen Data Guru

## 1. Input Jenis Guru

Input Jenis Guru			
Aktor	Sistem	Status	
		Berhasil	Gagal
1	Memilih menu guru pilih submenu data jenis guru	✓	
	2	Menampilkan form tambah jenis guru dan tabel jenis guru	✓
3	Mengisi data jenis guru	✓	
4	Klik submit		
	5	Data jenis guru berhasil ditambahkan	✓
Jika memasukkan data dengan format yang salah			
3	Data yang diisi kosong atau tidak lengkap	✓	
4	Klik submit		
	7	Menampilkan alert pada	✓

form yang kosong atau tidak lengkap

## 2. Hapus Jenis Guru

### Hapus Jenis Guru

Aktor	Sistem	Status	
		Berhasil	Gagal
1 Memilih menu <i>guru</i> pilih submenu data jenis guru	2 Menampilkan form tambah jenis guru dan tabel jenis guru	✓	
3 Memilih data yang akan dihapus	5 Menampilkan alert apakah data akan dihapus	✓	
4 Klik Hapus	7 Data berhasil dihapus	✓	
6 Klik Ok			
Data yang dimasukkan tidak valid			
5 Klik cancel			
6 Klik Ok	7 Kembali ke halaman data jenis guru dan data tidak dihapus	✓	

## 3. Input Nilai Guru

### Input Nilai Guru

Aktor	Sistem	Status	
		Berhasil	Gagal
1 Memilih menu <i>input</i> nilai guru	2 Menampilkan halaman data guru	✓	
3 Klik menu input	4 Menampilkan data detail guru dan form input nilai	✓	
5 Memasukkan nilai guru			

pada form data guru

6	Klik Simpan	7	Data berhasil ditambahkan	√
Data yang dimasukkan tidak valid				
5	Data yang dimasukkan tidak lengkap			√
6	Klik Simpan	7	Menampilkan alert pada form yang tidak lengkap	√

#### 4. Hapus Nilai Guru

Hapus Nilai Guru				
Aktor	Sistem	Status	Berhasil	Gagal
1	Memilih menu <i>input</i> nilai guru		√	
	2	Menampilkan halaman data guru	√	
3	Klik menu hapus	4	Menampilkan alert apakah yakin akan menghapus	√
5	Klik Ok	7	Data berhasil dihapus	√
Data yang dimasukkan tidak valid				
5	Klik cancel	7	Kembali ke halaman sebelumnya dan tidak terhapus	√

#### 5. Lihat Data Guru

Lihat Data Guru				
Aktor	Sistem	Status	Berhasil	Gagal
1	Memilih menu data guru		√	
	2	Menampilkan halaman data guru	√	
3	Klik menu view pada data yang akan dilihat			

4 Menampilkan data guru secara detail ✓

## 6. Hapus Data Guru

### Hapus Data Guru

Aktor	Sistem	Status	
		Berhasil	Gagal
1 Memilih menu data guru		✓	
	2 Menampilkan halaman data guru	✓	
3 Pilih data yang akan dihapus dan klik hapus		✓	
	4 Menampilkan alert apakah yakin akan menghapus data yang dipilih	✓	
5 Klik Ok			
	7 Data berhasil dihapus	✓	
Data yang dimasukkan tidak valid			
5 Klik cancel			
	7 Kembali ke halaman sebelumnya dan tidak terhapus	✓	

## 7. Input Kebutuhan Guru

### Input Kebutuhan Guru

Aktor	Sistem	Status	
		Berhasil	Gagal
1 Memilih menu guru		✓	
2 Pilih submenu input kebutuhan guru		✓	
	3 Menampilkan form tambah kebutuhan guru dan tabel kebutuhan guru	✓	
4 Mengisi data jenis guru dan tambah guru		✓	
5 Klik Submit			
	6 Data kebutuhan guru berhasil	✓	

ditambahkan		
Jika memasukkan data dengan format yang salah		
4	Mengisi data tidak lengkap	✓
5	Klik Submit	
6	Menampilkan alert pada form yang tidak lengkap	✓

### 8. Edit Biodata Guru

Input Kebutuhan Guru		
Aktor	Sistem	Status Berhasil    Gagal
1	Pilih menu guru	✓
2	Pilih submenu input data guru	✓
3	Menampilkan form ubah data guru	✓
4	Mengisi data guru sesuai dengan identitas guru	✓
5	Klik Submit	
6	Data sekolah berhasil diubah	✓
Jika memasukkan data dengan format yang salah		
4	Data yang diubah tidak lengkap	✓
5	Klik Submit	
6	Menampilkan alert pada form yang tidak lengkap	✓

## 9. Pemilihan Sekolah

Input Kebutuhan Guru		
Aktor	Sistem	Status Berhasil    Gagal
1	Pilih menu sekolah	✓
2	Pilih submenu pilih sekolah tujuan	✓
	3 Menampilkan form pilihan data sekolah	✓
4	Memilih sekolah yang akan dituju	✓
5	Klik Tambah	
	6 Data sekolah berhasil ditambahkan dan muncul pada tabel sekolah pilihan	✓
Jika mereset data		
4	Memilih sekolah yang akan dituju	✓
5	Klik Reset	
	6 Data yang diisikan terhapus	✓
Jika memilih lebih dari 3 sekolah		
4	Memilih lebih dari 3 sekolah	✓
5	Klik Tambah	
	6 Menampilkan alert bahwa tidak boleh lebih dari 3 sekolah	✓

## LAMPIRAN E PENULISAN KODE PROGRAM

### 1. c\_login

Fungsi ini merupakan penulisan kode program yang ada pada controller c\_login yang ada pada sistem SIPENTARU.

```
<?php
|
|
| if (!defined('BASEPATH'))
|     exit('No direct script access allowed');
|
| class c_login extends CI_Controller {
|
|     function __construct() {
|         parent::__construct();
|         $this->load->model('Login');
|         $this->load->model('m_pengguna');
|         $this->load->model('m_sekolah');
|     }
|
|     function index() {
|         $this->load->view('Login');
|     }
|
|     function home_admin() {
|         $this->load->view('Home');
|     }
|
|     function home_bkd() {
|         $this->load->view('Home');
|     }
|
|     function home_guru() {
|         $this->load->view('Home');
|     }
|
| }
```

```
function home_kepsek(){
    $this->load->view('Home');
}

function admin_pengguna_tambah() {
    $data['sekolah'] = $this->m_sekolah->view_sekolah();
    $data['data'] = $this->m_pengguna->pengguna();
    $data['level'] = $this->m_pengguna->select_level();
    $this->load->view('admin_pengguna_tambah', $data);
}

function view_pengguna($id) {
    $data['data'] = $this->m_pengguna->view_pengguna($id);
    $this->load->view('admin_pengguna_view', $data);
}

function user_pengguna() {
    $data['data'] = $this->m_pengguna->pengguna();
    $this->load->view('admin_pengguna', $data);
}

function edit_pengguna($id) {
    $data['data'] = $this->m_pengguna->pengguna();
    $data['dataedit'] = $this->m_pengguna->editpengguna($id);
    $this->load->view('pengguna_edit_admin', $data);
}

public function hapuspengguna($id) {
    $this->m_pengguna->hapus_pengguna($id);
    redirect('c_login/user_pengguna');
}

function update_pengguna() {
    $id_user = $this->input->post('id_user');
    $username = $this->input->post('username');
    $password = $this->input->post('password');
    $name = $this->input->post('name');
    if ($this->m_pengguna->update_pengguna($id_user, $username, $password)) {
        redirect('c_login/user_pengguna');
    } else {
        redirect('c_login/user_pengguna');
    }
}
}
```

```
public function tambahuser() {
//function yang dipanggil ketika ingin memasukan produk ke dalam database
    $id_sekolah = $this->input->post('sekolah');
    if($id_sekolah == 0){
        $id_sekolah = NULL;
    }
    $data = array(
        'username' => $this->input->post('username'),
        'password' => $this->input->post('password'),
        'id_level' => $this->input->post('level'),
        'id_sekolah' => $id_sekolah
    );

    $this->m_pengguna->tambahpengguna($data); //passing variable $data ke m_pengguna

    redirect('c_login/admin_pengguna_tambah'); //redirect page ke halaman utama controller admin
}

function login_process() {
    $this->load->model('Login');
    $query = $this->Login->validate();
    if ($query == TRUE) { // if the user's credentials validated...
        //$session_data = $this->session->userdata('login');
        //$this->session->set_userdata('username_login', $this->input->post('username'));
        //$this->session->set_userdata('password_login', $this->input->post('password'));
        //$username = $this->session->userdata('username');
        $level = $this->session->userdata('level');
        if ($level == "L01") {
            redirect('c_login/home_admin');
        } elseif ($level == "L02") {
            redirect('c_login/home_bkd');
        } elseif ($level == "L03") {
            redirect('c_login/home_kepsek');
        } elseif ($level == "L04") {
            redirect('c_login/home_guru');
        }
    } else { // incorrect username or password
        echo '<script>alert("Username atau Password yang Anda Masukkan Salah!")</script>';
        redirect(base_url(), 'refresh');
    }
}

function logout() {
    $this->session->sess_destroy();
    redirect(base_url());
}
}
```

## 2. c\_faktor

Fungsi ini merupakan penulisan kode program yang ada pada controller c\_faktor yang ada pada sistem SIPENTARU.

```
public function add_faktor(){
    $nama_faktor = $this -> input -> post('nama_faktor');
    $atribut = $this -> input -> post('atribut');
    $kriteria = $this -> input -> post('kriteria');
    $tipeisi = $this -> input -> post('tipeisi');
    $data = array(
        'nama_faktor' => $nama_faktor,
        'id_atribut' => $atribut,
        'id_kriteria' => $kriteria,
        'tipeisi' => $tipeisi
    );
    $sukses = $this->m_faktor->add_faktor($data);
    if ($sukses) {
        $this->session->set_flashdata('message', $this->succes_message);
        redirect('c_faktor/input_faktor');
    } else {
        $this->session->set_flashdata('message', $this->alert_message);
        redirect('c_faktor/input_faktor');
    }
}
public function input_skala($id_faktor){
    $data['id_faktor'] = $id_faktor;
    $data['faktor'] = $this -> m_faktor -> select_faktor($id_faktor);
    $data['skala_faktor'] = $this -> m_faktor -> select_skala_faktor($id_faktor);
    $this -> load -> view('input_skala', $data);
}
```

```
public function add_faktor(){
    $nama_faktor = $this -> input -> post('nama_faktor');
    $atribut = $this -> input -> post('atribut');
    $kriteria = $this -> input -> post('kriteria');
    $tipeisi = $this -> input -> post('tipeisi');
    $data = array(
        'nama_faktor' => $nama_faktor,
        'id_atribut' => $atribut,
        'id_kriteria' => $kriteria,
        'tipeisi' => $tipeisi
    );
    $sukses = $this->m_faktor->add_faktor($data);
    if ($sukses) {
        $this->session->set_flashdata('message', $this->succes_message);
        redirect('c_faktor/input_faktor');
    } else {
        $this->session->set_flashdata('message', $this->alert_message);
        redirect('c_faktor/input_faktor');
    }
}

public function input_skala($id_faktor){
    $data['id_faktor'] = $id_faktor;
    $data['faktor'] = $this -> m_faktor -> select_faktor($id_faktor);
    $data['skala_faktor'] = $this -> m_faktor -> select_skala_faktor($id_faktor);
    $this -> load -> view('input_skala', $data);
}

public function add_skala(){
    $id_faktor = $this -> input -> post('id_faktor');
    $skala_faktor = $this -> input -> post('skala_faktor');
    $bobot_skala_faktor = $this -> input -> post('bobot_skala_faktor');
    $data = array(
        'nama_skala_faktor' => $skala_faktor,
        'id_faktor' => $id_faktor,
        'bobot' => $bobot_skala_faktor
    );
    $sukses = $this->m_faktor->add_skala_faktor($data);
    if ($sukses) {
        $this->session->set_flashdata('message', $this->succes_message);
        redirect('c_faktor/input_skala/'.$id_faktor);
    } else {
        $this->session->set_flashdata('message', $this->alert_message);
        redirect('c_faktor/input_skala/'.$id_faktor);
    }
}
}
```

```
public function input_target($id_faktor){
    $data['id_faktor'] = $id_faktor;
    $data['faktor'] = $this -> m_faktor -> select_faktor($id_faktor);
    $data['skala_faktor'] = $this -> m_faktor -> select_skala_faktor($id_faktor);
    $this -> load -> view('input_target', $data);
}
public function add_target(){
    $sukses = false;
    $id_faktor = $this -> input -> post('id_faktor');
    $target_faktor = $this -> input -> post('target');
    $id_sekolah = $this->session->userdata('id_sekolah');
    $data = array(
        'id_faktor' => $id_faktor,
        'target' => $target_faktor,
        'id_sekolah' => $this->session->userdata('id_sekolah')
    );
    $cek_data_target = $this -> m_faktor -> cek_target($id_sekolah, $id_faktor);
    if($cek_data_target){
        $sukses = $this->m_faktor->edit_target_sekolah($id_sekolah, $id_faktor, $data);
    }else {
        $sukses = $this->m_faktor->add_target_sekolah($data);
    }
    if ($sukses) {
        $this->session->set_flashdata('message', $this->sukses_message);
        redirect('c_faktor/input_faktor');
    } else {
        $this->session->set_flashdata('message', $this->alert_message);
        redirect('c_faktor/input_faktor');
    }
}
}
```

### 3. c\_guru

Fungsi ini merupakan penulisan kode program yang ada pada controller c\_guru yang ada pada sistem SIPENTARU.

```

<?php
if (!defined('BASEPATH'))
    exit('No direct script access allowed');

class c_guru extends CI_Controller {
    private $alert_message = '<div class="alert alert-danger alert-dismissible">
    <i class="fa fa-ban"></i><button type="button" class="close"
    data-dismiss="alert" aria-hidden="true"></button><b>Error!</b>Terjadi kesalahan dalam penyimpanan.</div>';
    private $alert_message_1 = '<div class="alert alert-danger alert-dismissible">
    <i class="fa fa-ban"></i><button type="button" class="close"
    data-dismiss="alert" aria-hidden="true"></button><b>Error!</b>Jumlah maksimal pilihan sekolah telah tercapai.
    Jumlah maksimal adalah 3.</div>';
    private $alert_message_2 = '<div class="alert alert-danger alert-dismissible"><i class="fa fa-ban">
    </i><button type="button" class="close" data-dismiss="alert"
    aria-hidden="true"></button><b>Error!</b>Data sekolah pilihan telah diinputkan sebelumnya (tidak boleh sama).</div>';
    private $sukses_message = '<div class="alert alert-success alert-dismissible">
    <i class="fa fa-check"></i><button type="button" class="close" data-dismiss="alert"
    aria-hidden="true"></button><b>Sukses!</b>Penyimpanan berhasil dilakukan.</div>';
    function __construct() {
        parent::__construct();
        $this->load->model('m_pengguna');
        $this->load->model('m_sekolah');
        $this->load->model('m_guru');
        $this->load->model('m_faktor');
        //$this->load->model('model_bkd');
    }

    function input_biadata_guru() {
        $data['provinsi'] = $this->m_sekolah->select_prov();
        $data['gender'] = $this->m_guru->select_gender();
        $data['pendidikan'] = $this->m_guru->select_pendidikan();
        $data['jenis_guru'] = $this->m_guru->select_jenis_guru();
        $data['agama'] = $this->m_guru->select_agama();
        $data['guru'] = $this->m_guru->select_detail_guru($this->session->userdata('username'));
        $this->load->view('input_biadata_guru', $data);
    }

    function kota($id_provinsi) {
        $tmp = '';
        $data = $this->m_sekolah->select_kota($id_provinsi);
        if (!empty($data)) {
            $tmp.="<option''>Pilih Kota</option>";
            foreach ($data as $row) {
                $tmp.="<option value= ' " . $row->id_kota . "'> " . $row->nama_kota . "</option>";
            }
        } else {
            $tmp.="<option value=''>Pilih provinsi dahulu</option>";
        }
        echo $tmp;
    }
}

```

```
function add_guru() {
    $data = array(
        'nip' => $this->input->post('nip'),
        'nama_guru' => $this->input->post('nama_guru'),
        'alamat_guru' => $this->input->post('alamat_guru'),
        'id_kota' => $this->input->post('kota_guru'),
        'tempat_lahir_guru' => $this->input->post('tempat_lahir_guru'),
        'tgl_lhr_guru' => $this->input->post('tgl_lhr_guru'),
        'id_agama' => $this->input->post('agama'),
        'id_gender' => $this->input->post('id_gender'),
        'id_pendidikan' => $this->input->post('id_pendidikan'),
        'id_jenis_guru' => $this->input->post('id_jenis_guru'),
        'lat' => $this->input->post('lat'),
        'long' => $this->input->post('long'),
        'username' => $this->session->userdata('username')
    );
    $sukses = $this->m_guru->add_data_guru($data);
    if ($sukses){
        $this->session->set_flashdata('message', $this->succes_message);
        redirect('c_guru/input_biodata_guru');
        echo '';
    } else {
        $this->session->set_flashdata('message', $this->alert_message);
        redirect('c_guru/input_biodata_guru');
    }
}

function pilih_sekolah(){
    $stampil = false;
    $data_guru = $this->m_guru->cek_data_guru($this->session->userdata('username'));
    if($data_guru == 1){
        $stampil = true;
    }else{
        $stampil = false;
    }
    $guru = $this->m_guru->select_detail_guru($this->session->userdata('username'));
    $data['guru'] = $this->m_guru->select_detail_guru($this->session->userdata('username'));
    $data['sekolah'] = $this->m_sekolah->view_sekolah();
    $data['pilihan'] = $this->m_guru->ambil_pilihan($guru['nip']);
    $data['tampil'] = $stampil;
    $this->load->view('pilih_sekolah_guru', $data);
}

function ambil_koordinat_sekolah(){
    $id_sekolah = $this->input->post('x');
    $koordinat = $this->m_sekolah->ambil_koordinat($id_sekolah);
    $lat = $koordinat['lat'];
    $lat .= ",";
    $lat .= $koordinat['long'];
    echo $lat;
}
```

```
function add_pilihan_sekolah(){
    $jumlah_pilihan = $this->m_guru->jumlah_pilihan($this->input->post('nip'));
    $cek_sekolah_nip = $this->m_guru->cek_sekolah_nip($this->input->post('nip'), $this->input->post('sekolah'));
    if($jumlah_pilihan === 3){
        $this->session->set_flashdata('message', $this->alert_message_1);
        redirect('c_guru/pilih_sekolah');
    }elseif($cek_sekolah_nip === 1){
        $this->session->set_flashdata('message', $this->alert_message_2);
        redirect('c_guru/pilih_sekolah');
    }else{
        $pilihan = array(
            'nip' => $this->input->post('nip'),
            'id_sekolah' => $this->input->post('sekolah'),
            'jarak' => $this->input->post('jarak')
        );
        $sukses = $this->m_guru->add_pilihan($pilihan);
        if ($sukses) {
            $this->session->set_flashdata('message', $this->succes_message);
            redirect('c_guru/pilih_sekolah');
        } else {
            $this->session->set_flashdata('message', $this->alert_message);
            redirect('c_guru/pilih_sekolah');
        }
    }
}

function lihat_data_guru($id) {
    $data['data'] = $this->m_guru->view_detail_tabel_guru($id);
    $this->load->view('lihat_data_guru', $data);
}

function tabel_daftar_guru() {
    $this->data['data'] = $this->m_guru->view_tabel_guru();
    $this->load->view('tabel_daftar_guru', $this->data);
}

function hapus_nilai($id) {
    $this->m_guru->hapus_nilai($id);
    echo
    "<html>
    <head>
    <script>
    function display_alert()
    {
    alert('data berhasil dihapus');
    window.location.href='".base_url()."/index.php/c_guru/input_nilai_guru';
    }
    </script>
    </head>
    <body onload=display_alert()>
    </body>
    </html>
    ";
}
}
```

```

function hapusguru($id) {
    $sukses = $this->m_guru->hapus_guru($id);
    if ($sukses == true) {
        redirect('c_guru/tabel_daftar_guru');
    } else {
        redirect('c_guru/tabel_daftar_guru');
    }
}

function hapusjenisguru($id) {
    $sukses = $this->m_guru->hapus_jenis_guru($id);
    if ($sukses == true) {
        redirect('c_guru/input_jenis_guru');
    } else {
        redirect('c_guru/input_jenis_guru');
    }
}

function lihat_data_sekolah(){
    $data['kecamatan'] = $this->m_sekolah->select_kecamatan();
    $data['sekolah'] = $this->m_sekolah->view_sekolah();
    $this->load->view('lihat_data_sekolah', $data);
}

function ubah_data_sekolah(){
    $data['sekolah'] = $this->m_sekolah->select_sekolah($this->session->userdata('id_sekolah'));
    $this->load->view('ubah_data_sekolah', $data);
}

function koordinat_sekolah(){
    $lat_lng = "";
    $alamat = urlencode($this->input->post('x'));
    $url = "http://maps.googleapis.com/maps/api/geocode/json?address=\$alamat&sensor=false";
    $result = file_get_contents($url);
    $result_json = json_decode($result);
    if ($result_json->status == 'OK'){
        for ($i=0; $i<1; $i++){
            $lat = $result_json->results[$i]->geometry->location->lat;
            $lng = $result_json->results[$i]->geometry->location->lng;
            $lat_lng = $lat.";".$lng;
        }
        echo $lat_lng;
    }else{
        echo 1;
    }
}

function proses_ubah_data_sekolah(){
    $data = array(
        'alamat' => $this->input->post('alamat'),
        'no_telfon' => $this->input->post('telepon'),
        'lat' => $this->input->post('lat'),
        'long' => $this->input->post('long')
    );
    $logic = $this->m_sekolah->ubah_data_sekolah($data);
    if($logic){
        redirect('c_guru/ubah_data_sekolah');
    }
}

function add_sekolah(){
    $data = array(

```

```

function add_sekolah(){
    $data = array(
        'id_kecamatan' => $this->input->post('nama_kecamatan'),
        'nama_sekolah' => $this->input->post('sekolah')
    );
    $logic = $this->m_sekolah->add_sekolah($data);
    if($logic){
        redirect('c_guru/lihat_data_sekolah');
    }
}

function lihat_data_detail_sekolah($id) {
    $data['data'] = $this->m_guru->view_lihat_data_detail_sekolah($id);
    $this->load->view('lihat_data_detail_sekolah', $data);
}

function hapussekolah($id) {
    $sukses = $this->m_guru->hapussekolah($id);
    if ($sukses == true) {
        redirect('c_guru/lihat_data_sekolah');
    } else {
        redirect('c_guru/lihat_data_sekolah');
    }
}

function input_jenis_guru(){
    $data['jenis_guru'] = $this -> m_guru -> select_jenis_guru();
    $this->load->view('data_jenis_guru', $data);
}

function add_jenis_guru(){
    $jenis_guru = $this -> input -> post('jenis_guru');
    $data = array(
        'jenis_guru' => $jenis_guru
    );
    $sukses = $this->m_guru->add_jenis_guru($data);
    if ($sukses === TRUE) {
        $this->session->set_flashdata('message', $this->succes_message);
        redirect('c_guru/input_jenis_guru');
        echo '';
    } else {
        $this->session->set_flashdata('message', $this->alert_message);
        redirect('c_guru/input_jenis_guru');
    }
}

function sekolah_butuh_guru(){
    $data['id_sekolah'] = $this->session->userdata('id_sekolah');
    $data['sekolah'] = $this -> m_sekolah -> select_sekolah($this->session->userdata('id_sekolah'));
    $data['jenis_guru'] = $this -> m_guru -> select_jenis_guru();
    $data['butuh_guru'] = $this -> m_sekolah -> select_butuh_guru($this->session->userdata('id_sekolah'));
    $this -> load -> view('sekolah_butuh_guru', $data);
}

```

```
function add_butuh_guru(){
    $id_sekolah = $this -> input -> post('id_sekolah');
    $jenis_guru = $this -> input -> post('jenis_guru');
    $jumlah_butuh = $this -> input -> post('jumlah_butuh');
    $data = array(
        'id_sekolah' => $id_sekolah,
        'jumlah_butuhguru' => $jumlah_butuh,
        'id_jenis_guru' => $jenis_guru
    );
    $sukses = $this->m_sekolah->add_butuh_guru($data);
    if ($sukses === TRUE) {
        $this->session->set_flashdata('message', $this->succes_message);
        redirect('c_guru/sekolah_butuh_guru/');
        echo '';
    } else {
        $this->session->set_flashdata('message', $this->alert_message);
        redirect('c_guru/sekolah_butuh_guru/');
    }
}
function input_nilai_guru(){
    $this->data['data'] = $this->m_guru->view_tabel_guru();
    $this->load->view('input_nilai_guru', $this->data);
}
function input_nilai($nip){
    $this->data['guru'] = $this -> m_guru -> view_detail_tabel_guru($nip);
    $this->data['faktor'] = $this -> m_faktor -> select_faktor_nilai();
    $this->load->view('input_nilai', $this->data);
}
```

```

function proses_input_nilai(){
    $faktor = $this->m_faktor->select_faktor_nilai();
    $nip = $this->input->post('nip');
    $res_jarak = $this->m_guru->ambil_pilihan($nip);
    $res_guru = $this->m_guru->view_detail_tabel_guru($nip);
    $alamat;
    foreach($res_guru as $rows){
        $alamat = $rows['alamat_guru'];
    }
    $nilai = "";
    $skala = "";
    // Jarak
    foreach($res_jarak as $row){
        $param_jarak = $row['jarak'];
        $skala_jarak = $this->cari_skala(1, $param_jarak);
        $skala_wilayah = $this->get_skala_wilayah($alamat, $row['alamat']);
        $this->m_faktor->skala_jarak_wilayah($row['id_pilihan'], $skala_jarak,$skala_wilayah);
    }
    // Status Wilayah

    // Input.
    foreach($faktor as $row):
        $tipe = $this->input->post('tipe_'.$row['id_faktor']);
        if($tipe === 't'){
            $nilai = $this->input->post('t_'.$row['id_faktor']);
            $skala = $this->cari_skala($row['id_faktor'], $nilai);
        }else{
            $nilai = NULL;
            $skala = $this->input->post('s_'.$row['id_faktor']);
        }
        $data_insert = array(

function get_skala_wilayah($alamat_guru, $alamat_sekolah){
    $exp_alamat_guru = explode(",",$alamat_guru);
    $exp_alamat_sekolah = explode(",",$alamat_sekolah);
    $skala_wilayah;
    if(trim($exp_alamat_guru[1]) == trim($exp_alamat_sekolah[1])){
        $skala_wilayah = $this->m_faktor->get_skala_wilayah("Kecamatan Sama");
    }elseif((trim($exp_alamat_guru[1]) != trim($exp_alamat_sekolah[1])) &&
(trim($exp_alamat_guru[2]) == trim($exp_alamat_sekolah[2]))){
        $skala_wilayah = $this->m_faktor->get_skala_wilayah("Kecamatan Beda Dalam Kota");
    }else{
        $skala_wilayah = $this->m_faktor->get_skala_wilayah("Luar Kota");
    }
    return $skala_wilayah;
}

```

```
function cari_skala($id_faktor, $nilai){
    $skalar = $this->m_faktor->select_skala_faktor($id_faktor);
    $skala = "";
    foreach($skalar as $row):
        $nama = $row['nama_skala_faktor']."<br />";
        if(strpos($nama, "-"){
            $range = explode("-", $nama);
            $range_0 = $range[0] + 0;
            $range_1 = $range[1] + 0;
            if($nilai >= $range_0 && $nilai <= $range_1){
                $skala = $row['id_skala_faktor'];
            }
        }elseif(strpos($nama, "="){
            $range = explode("=", $nama);
            $range_1 = $range[1] + 0;
            if($nilai >= $range_1){
                $skala = $row['id_skala_faktor'];
            }
        }
    }
    endforeach;
    return $skala;
}

function lihat_hasil_perhitungan(){
    //$this->data['profile'] = $this->m_guru->profile_matching();
    $this->data['guru'] = $this->m_guru->view_tabel_guru();
    $this->load->view('hasil_perhitungan', $this->data);
}

function ranking_sekolah(){
    $this->data['sekolah'] = $this->m_sekolah->view_sekolah();
    $this->load->view('ranking_sekolah', $this->data);
}

function lihat_ranking_sekolah(){
    $id_sekolah = $this->input->post('sekolah');
    $this->data['sekolah'] = $this->m_sekolah->select_sekolah($id_sekolah);
    $this->data['peringkat'] = $this->m_sekolah->peringkat_sekolah($id_sekolah);
    $this->load->view('lihat_ranking_sekolah', $this->data);
}

function penempatan(){
    $this->data['penempatan'] = $this->m_sekolah->kebutuhan_penempatan();
    $this->load->view('penempatan', $this->data);
}

function tempatkan_guru($id_sekolah, $id_jenis_guru){
    $this->data['id_sekolah'] = $id_sekolah;
    $this->data['id_jenis_guru'] = $id_jenis_guru;
    $this->data['sekolah'] = $this->m_sekolah->select_sekolah($id_sekolah);
    $this->data['jenis_guru'] = $this->m_guru->jenis_guru($id_jenis_guru);
    $this->data['peringkat'] = $this->m_sekolah->peringkat_sekolah_jenis($id_sekolah, $id_jenis_guru);
    $this->load->view('pilih_guru', $this->data);
}
}
```

```

function tempat_guru($snip, $sid_sekolah, $sid_jenis_guru){
    $logika = $this->m_guru->update_butuh_guru($snip, $sid_sekolah, $sid_jenis_guru);
    if($logika){
        redirect('c_guru/penempatan');
    }else{
        redirect('c_guru/penempatan');
    }
}
function hasil_penempatan(){
    $data_guru = $this->m_guru->select_detail_guru($this->session->userdata('username'));
    $this->data['sekolah'] = $this->m_sekolah->select_sekolah($data_guru['penempatan']);
    $this->data['guru'] = $data_guru;
    $this->load->view('hasil_penempatan', $this->data);
}
}

```

#### 4. c\_kriteria

Fungsi ini merupakan penulisan kode program yang ada pada controller c\_kriteria yang ada pada sistem SIPENTARU.

```

k?php
if (!defined('BASEPATH'))
    exit('No direct script access allowed');

class c_kriteria extends CI_Controller {

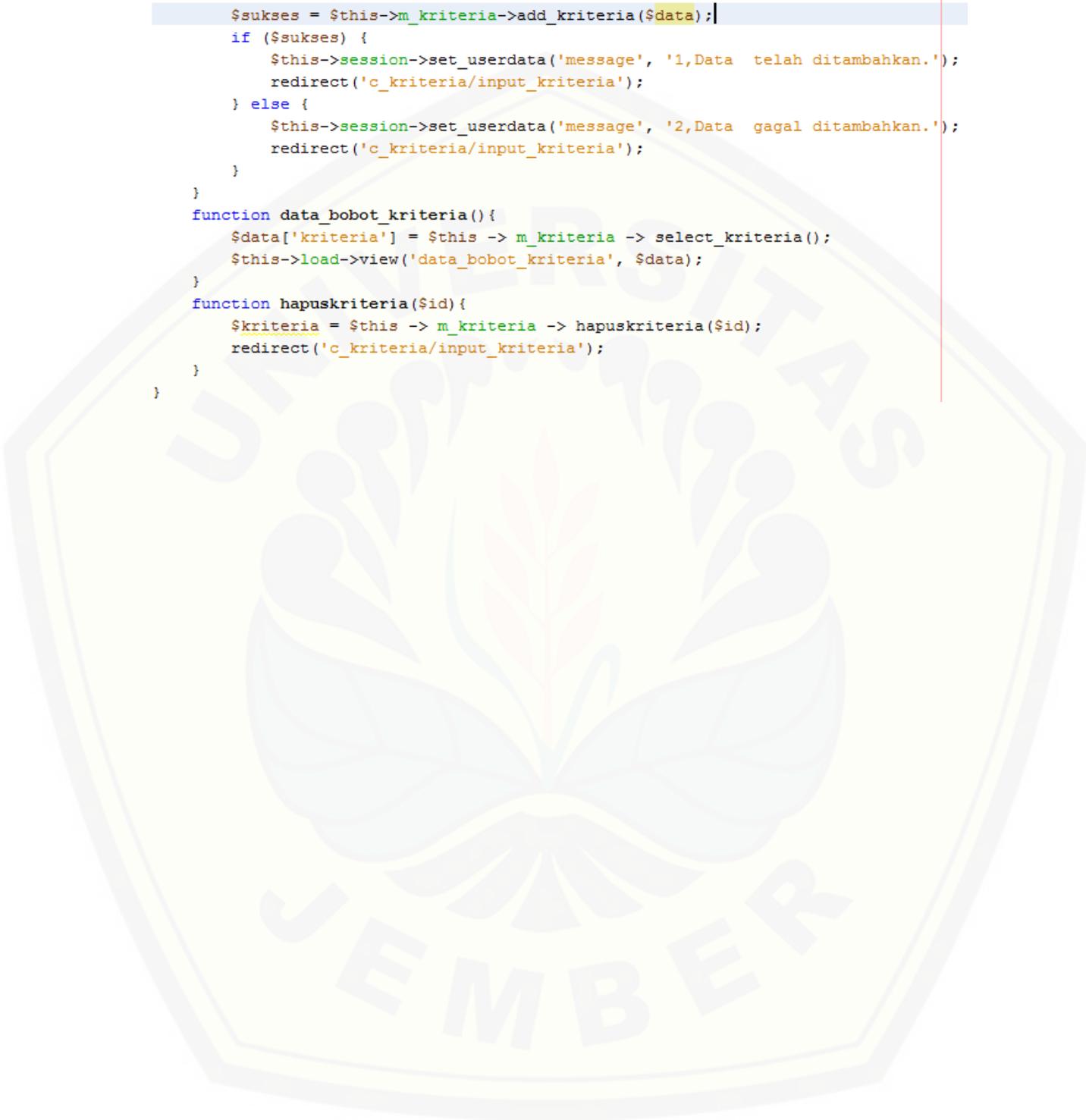
    function __construct() {
        parent::__construct();
        $this->load->model('m_atribut');
        $this->load->model('m_kriteria');
    }

    function input_kriteria() {
        $data['atribut'] = $this->m_atribut->select_atribut();
        $data['kriteria'] = $this->m_kriteria->select_kriteria();
        $this->load->view('input_kriteria', $data);
    }

    function add_kriteria() {
        $nama = $this->input->post('nama_kriteria');
        $data = array(
            'nama_kriteria' => $nama
        );
    }
}

```

```
$sukses = $this->m_kriteria->add_kriteria($data);  
if ($sukses) {  
    $this->session->set_userdata('message', '1,Data telah ditambahkan.');
```

The background of the page features a large, semi-transparent watermark of the Universitas Jember logo. The logo is a shield-shaped emblem with a central floral motif and the text 'UNIVERSITAS JEMBER' around the perimeter.

```
    redirect('c_kriteria/input_kriteria');  
} else {  
    $this->session->set_userdata('message', '2,Data gagal ditambahkan.');
```

```
    redirect('c_kriteria/input_kriteria');  
}  
}  
function data_bobot_kriteria(){  
    $data['kriteria'] = $this -> m_kriteria -> select_kriteria();  
    $this->load->view('data_bobot_kriteria', $data);  
}  
function hapuskriteria($id){  
    $kriteria = $this -> m_kriteria -> hapuskriteria($id);  
    redirect('c_kriteria/input_kriteria');
```

```
    }  
}
```

## 5. c\_atribut

Fungsi ini merupakan penulisan kode program yang ada pada controller c\_atribut yang ada pada sistem SIPENTARU.

```
<?php

if (!defined('BASEPATH'))
    exit('No direct script access allowed');
class C_atribut extends CI_Controller {
    function __construct() {
        parent::__construct();
        $this->load->model('m_atribut');
    }
    function input_atribut() {
        $data['aspek'] = $this->m_atribut->select_atribut();
        $this->load->view('input_atribut', $data);
    }
    function add_atribut() {
        $data = array(
            'id_aspek' => $this->input->post('id_aspek'),
            'nama_aspek' => $this->input->post('nama_aspek'),
            'presentase' => $this->input->post('presentase')
        );
        $logic = $this->m_atribut->add_atribut($data);
        if ($logic) {
            redirect('c_atribut/input_atribut');
        }
    }
    function ubah_atribut(){
        $presentase = $this->input->post('persen');
        $id = $this->input->post('idatribut');
        $this->m_atribut->ubah_atribut($presentase, $id);
    }
}
```