



**SISTEM PENDUKUNG KEPUTUSAN PEMILIHAN MAJELIS JEMAAT  
GEREJA PROTESTAN DI INDONESIA BAGIAN BARAT “GETSEMANI”  
JEMBER MENGGUNAKAN METODE *SIMPLE MULTI ATTRIBUTE  
RATING TECHNIQUE* (SMART)**

**SKRIPSI**

Diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat  
untuk menyelesaikan pendidikan di Fakultas Ilmu Komputer Program Studi  
Sistem Informasi Universitas Jember dan mendapat gelar Sarjana Sistem  
Informasi

**HALAMAN JUDUL**

disusun oleh

**Yosafat Parulian Devinaldo**

**NIM 122410101061**

**PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS JEMBER**

**2019**

## **PERSEMBAHAN**

Skripsi ini saya persembahkan untuk :

1. Tuhan Yang Maha Esa yang senantiasa memberikan rahmat dan hidayah-Nya untuk mempermudah dan melancarkan dalam mengerjakan skripsi.
2. Kedua orang tua saya
3. Keluarga besar yang selalu mensupport dengan do'a dan dukungannya.
4. Sahabat – sahabatku dengan dukungan beserta doanya.
5. Guru – guru baik dari pendidikan formal maupun informal.
6. Gereja Protestan di Indonesia bagian Barat “GETSEMANI” Jember.
7. Almamater Fakultas Ilmu Komputer Universitas Jember.

## **MOTTO**

“Digowo urip nguripi, digowo mati nyegeri”

Syahadat Kencana Kawruh Jendra Hayuningrat

## PENYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Yosafat Parulian Devinaldo

NIM : 122410101061

Menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Sistem Pendukung Keputusan Pemilihan Majelis Jemaat Gereja Protestan di Indonesia Bagian Barat “GETSEMANI” Jember Menggunakan Metode *Simple Multiple Attribute Rating Technique* (SMART)” adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya, belum pernah diajukan pada institusi mana pun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

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

Jember, 20 Mei 2019

Yang menyatakan,

Yosafat Parulian Devinaldo

NIM 122410101061

## **SKRIPSI**

# **SISTEM PENDUKUNG KEPUTUSAN PEMILIHAN MAJELIS JEMAAT GEREJA PROTESTAN DI INDONESIA BAGIAN BARAT “GETSEMANI” JEMBER MENGGUNAKAN METODE *SIMPLE MULTI ATTRIBUTE RATING TECHNIQUE* (SMART)**

Oleh :

Yosafat Parulian Devinaldo

NIM 122410101061

Pembimbing

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

Dosen Pembimbing Pendamping : Windy Eka Yulia R, S. Kom, MT

## RINGKASAN

Gereja Indonesia di Indonesia bagian Barat “GETSEMANI” Jember adalah sebuah organisasi gereja yang tergabung dalam GPI yang beralamat di Jalan PB Sudirman no. 1 Jember dan memiliki sekitar lebih dari 450 jemaat. Untuk menunjang kegiatan gereja agar terlaksana dengan baik, maka dibentuk badan pelaksana dari jemaat yang disebut sebagai Majelis gereja. Jemaat yang dapat dipilih menjadi Majelis harus menjadi jemaat tetap GKT Jember dan memiliki kriteria sesuai dengan Alkitab seperti yang telah ditetapkan oleh Sinode GPIB, yaitu: 1) Memiliki moralitas yang baik.; 2) Memiliki karakter yang baik; 3) Memiliki kompetensi yang baik; 4) Pemahaman tentang ajaran Kristen yang baik; Pada pelaksanaannya, majelis jemaat masih dipilih secara manual oleh jemaat di mana semua jemaat aktif dapat terpilih menjadi majelis gereja dan kurang diperhatikannya kriteria dari Alkitab seperti yang disebutkan di atas.

Sistem penunjang keputusan pemilihan majelis dibuat untuk membantu memberikan rekomendasi pemilihan majelis berdasarkan kriteria-kriteria yang ditentukan dan mengurangi subjektifitas.

Pada penelitian ini sistem yang dibangun menggunakan metode *Simple Multiple Attribute Rating Technique* (SMART) yang dapat menggambarkan kesimpulan pasti dari data yang samar-samar dan ambigu dan tidak tepat. Hasil akhir dari penerapan metode SMART akan menghasilkan perankingan data alternatif untuk menentukan alternatif terbaik untuk dipilih sebagai majelis gereja.

## **PRAKATA**

Puji syukur kepada Tuhan Yang Maha Esa yang telah melimpahkan rahmat, karunia-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Sistem Pendukung Keputusan Pemilihan Majelis Jemaat Gereja Protestan di Indonesia Bagian Barat “GETSEMANI” Jember”. Skripsi ini disusun untuk melengkapi salah satu syarat menyelesaikan pendidikan Strata 1 (S1) pada Program Studi Sistem Informasi Fakultas Ilmu Komputer Universitas Jember.

Dalam proses penyusunan skripsi ini tidak terlepas dari bantuan berbagai pihak. Oleh karena itu dalam kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Thomas Mariyunus dan Sari Napitupulu selaku orang tua dari penulis yang telah memberikan support selalu dalam setiap perjalanan penulis.
2. Drs. Antonius Cahya P, M.App. Sc., Ph.D sebagai Dosen Pembimbing Utama dan Windi Eka Yulia Retnani, S.Kom., M.T. sebagai Dosen Pembimbing Pendamping yang telah meluangkan waktu, pikiran dan perhatian dalam penulisan skripsi ini;
3. Prof. Drs. Slamir, M. Com. Sc., Ph.D selaku Ketua Program Studi Sistem Informasi Universitas Jember;
4. Bopo Pinisepuh Budi Siswanto Wongsodjono, Ibu Piteraning Dyah, Deva Dhaharoe Dwi Kharisma, dan seluruh saudara - saudara YPAMJ;
5. Fathoni Wahyudi yang secara ikhlas meminjamkan laptopnya untuk saya gunakan dalam pengerjaan penulisan ini;
6. Sahabat seperjuangan saya dari TK sampai sekarang yaitu Yohanis Permadi Polowiwi yang selalu mensupport dalam segala kondisi;

7. UKMK ETALASE dan seluruh anggotanya sebagai tempat berproses dalam segala hal selama masa perkuliahan;
8. Rohaniawan, Majelis dan Jemaat GPIB “GETSEMANI” Jember yang telah membantu berjalannya penelitian ini;
9. Teman-teman yang telah membantu dan mensupport agar penulisan ini bisa terselesaikan khususnya Gatel, Aji, kawan – kawan “LINGKAR MERIMBUN” yaitu Vindri, Fahmy Adji, Mas Tata, Cak Dodon, Mas Caca, dan kawan FORMATION yang tergabung dalam grup “H-28 NO MUTUNG”
10. Kepada diri saya sendiri yang tetap tabah dalam 7 tahun perkuliahan ini;
11. Semua pihak yang tidak bisa penulis sebutkan satu per satu;

Penulis menyadari penulisan skripsi ini masih jauh dari kesempurnaan, oleh karena itu penulis mengharapkan kritik dan saran dari semua pihak. Akhir kata, penulis berharap semoga skripsi ini dapat memberikan manfaat bagi semua pihak.

Jember, 20 Mei 2019

Penulis



## DAFTAR ISI

HALAMAN JUDUL.....	i
PERSEMBAHAN .....	ii
MOTTO.....	ii
PENYATAAN .....	iii
SKRIPSI.....	iv
RINGKASAN .....	v
PRAKATA.....	vi
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xii
DAFTAR GAMBAR .....	xiv
DAFTAR LAMPIRAN.....	xvi
BAB 1. PENDAHULUAN .....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Tujuan.....	3
1.4 Batasan Masalah .....	4
BAB 2. TINJAUAN PUSTAKA .....	5
2.1 Penelitian Terdahulu.....	5
2.2 Metode Simple Multi Attribute Rating Technique (SMART) .....	6
2.3 Langkah Pengujian Metode SMART .....	7
BAB 3. METODOLOGI PENELITIAN.....	8
3.1 Jenis Penelitian .....	8
3.2 Tahapan Penelitian.....	8
3.2.1 Studi Literatur dan Jurnal.....	9
3.2.2 Wawancara.....	9
3.2.3 Pembuatan Sistem .....	10
3.2.4 Uji Efektivitas Metode SMART .....	13
BAB 4.PENGEMBANGAN SISTEM INFORMASI.....	19

4.1	Deskripsi Umum Sistem Informasi .....	19
4.2	Analisis Kebutuhan Sistem Informasi .....	19
4.2.1	Kebutuhan Fungsional .....	20
4.2.2	Kebutuhan Nonfungsional .....	20
4.2.3	Kebutuhan Antarmuka Pemakai .....	20
4.2.4	Kebutuhan Antarmuka Perangkat Keras .....	21
4.2.5	Kebutuhan Perangkat Lunak .....	21
4.3	Desain Sistem .....	21
4.3.1	<i>Bussiness Process</i> .....	21
4.3.2	<i>Use Case Diagram</i> .....	22
4.3.3	<i>Use Case Skenario</i> .....	25
4.3.4	<i>Activity Diagram</i> .....	29
4.3.5	<i>Sequence Diagram</i> .....	32
4.3.6	<i>Class Diagram</i> .....	36
4.3.7	<i>Entity Relationship Diagrams (ERD)</i> .....	37
4.4	Penulisan Implementasi Kode Program.....	38
4.4.1	Kode Program tambah_anggota() .....	38
4.4.2	Kode Program update_anggota() .....	45
4.5	Pengujian Sistem Informasi .....	52
BAB 5.HASIL DAN PEMBAHASAN.....		58
5.1	Hasil Pengembangan Sistem Pendukung Keputusan Pemilihan Majelis Jemaat Gereja Protestan di Indonesia bagian Barat “GETSEMANI” Jember .....	58
5.1.1	Tampilan Halaman Home .....	58
5.1.2	Tampilan Halaman Utama Admin .....	59
5.2	Hasil Penghitungan Metode SMART .....	65
5.2.1	Data Kriteria.....	65
5.2.2	Data Sub Kriteria.....	66
5.2.3	Penerapan Metode SMART .....	68
5.2.4	Perhitungan Menggunakan Aplikasi Excel Pemilihan Majelis Jemaat GPIB “GETSEMANI” Jember .....	69

5.3	Perbandingan Hasil Perhitungan Menggunakan Aplikasi Excel dan Perhitungan Menggunakan Metode SMART .....	71
5.4	Pembahasan Pada Sistem Pendukung Keputusan Pemilihan Majelis Jemaat GPIB “GETSEMANI” Jember .....	73
5.4.1	Fitur Sistem .....	73
5.4.2	Pembahasan <i>Software Development Life Cycle Model Waterfall</i> ... ..	74
5.4.3	Pengujian Sistem.....	75
BAB 6. PENUTUP.....		76
6.1	Kesimpulan.....	76
6.2	Saran .....	77
DAFTAR PUSTAKA .....		78
LAMPIRAN .....		79
LAMPIRAN A. SKENARIO.....		79
A.1	Skenario Login.....	79
A.2	Skenario View Data Jemaat .....	80
A.3	<i>Skenario Insert</i> Data Jemaat .....	82
A.4	<i>Skenario Update</i> Data Jemaat.....	85
A.5	<i>Skenario Delete</i> Data Jemaat .....	87
A.6	<i>Skenario View</i> Kriteria Majelis Jemaat.....	89
A.7	<i>Skenario Update</i> Kriteria Majelis Jemaat .....	90
A.8	<i>Skenario View</i> Sub Kriteria Majelis Jemaat.....	92
A.9	<i>Skenario Update</i> Sub Kriteria Majelis Jemaat.....	93
A.10	<i>Skenario Rangking</i> .....	95
LAMPIRAN B.1 ACTIVITY .....		98
B.1	Activity Login .....	98
B.2	Activity View Data Jemaat .....	98
B.3	Activity Insert Data Jemaat.....	99
B.4	Activity Update Data Jemaat .....	99
B.5	Activity Delete Data Jemaat.....	100
B.6	Activity View Kriteria Majelis Jemaat .....	100
B.7	Activity Update Kriteria Majelis Jemaat.....	101

B.8 Activity View Sub Kriteria Majelis Jemaat .....	101
B.9 Activity Update Sub Kriteria Majelis Jemaat .....	102
B.10 Activity Rangking .....	102
LAMPIRAN C. SEQUENCE .....	103
C.1 Sequence Login.....	103
C.2 Sequence <i>View</i> Data Jemaat.....	103
C.3 Sequence Insert Data Jemaat.....	104
C.4 Sequence Update Data Jemaat .....	104
C.5 Sequence Delete Data Jemaat .....	105
C.6 Sequence View Kriteria Majelis Jemaat .....	105
C.7 Sequence Update Kriteria Majelis Jemaat .....	106
C.8 Sequence View Sub Kriteria Majelis Jemaat .....	106
C.9 Sequence Update Sub Kriteria Majelis Jemaat .....	107
C.10 Sequence View Rangking .....	107
LAMPIRAN D. KODE PROGRAM .....	108
D.1 Kode Program Controller/anggota.php .....	108
D.2 Kode Program Controller/beranda.php .....	113
D.3 Kode Program Controller/kriteria.php .....	113
D.4 Kode Program Controller/landing.php.....	115
D.5 Kode Program Controller/rangking.php .....	116
D.6 Kode Program Controller/subkriteria.php .....	117
D.7 Kode Program Model/m_anggota.php.....	118
D.8 Kode Program Model/m_kriteria.php.....	120
D.9 Kode Program Model/m_login.php .....	121
D.10 Kode Program Model/m_subkriteria.php .....	121

## DAFTAR TABEL

Tabel 3.1 Daftar Kriteria Pemilihan Majelis Jemaat GPIB Getsemani Jember ....	13
Tabel 3.2 Pentuan Bobot Kriteria dan Normalisasi Bobot.....	14
Tabel 3.3 Sub Kriteria Moralitas yang Tinggi .....	15
Tabel 3.4 Sub Kriteria Karakter yang Baik.....	15
Tabel 3.5 Sub Kriteria Kompetensi yang Baik .....	15
Tabel 3.6 Sub Kriteria Kelengkapan Administrasi .....	16
Tabel 3.7 Sub Kriteria Kekristenan.....	16
Tabel 3.8 Nilai Alternatif .....	17
Tabel 3.9 Tabel Perhitungan Akhir .....	17
Tabel 3.10 Tabel Perankingan.....	18
Tabel 4.1 Definisi Aktor .....	24
Tabel 4.2 Definisi Usecase.....	24
Tabel 4.3 Hasil Pengujian Black Box .....	53
Tabel 5.1 Data Kriteria.....	65
Tabel 5.2 Sub Kriteria Moralitas yang Tinggi .....	66
Tabel 5.3 Sub Kriteria Kompetensi yang Baik .....	67
Tabel 5.4 Sub Kriteria Kelengkapan Administrasi .....	67
Tabel 5.5 Sub Kriteria Kekristenan.....	67
Tabel 5.6 Normalisasi Kriteria.....	68
Tabel 5.7 Perhitungan Manual Pada Percobaan Sampel Data Jemaat GPIB “GETSEMANI” Jember.....	69
Tabel 5.8 Perbandingan Perhitungan Manual dan Perhitungan Sistem .....	71

A.1 Skenario Login .....	79
A.2 Skenario View Data Jemaat .....	80
A.3 <i>Skenario Insert</i> Data Jemaat.....	82
A.4 <i>Skenario Update</i> Data Jemaat .....	85
A.5 Skenario <i>Delete</i> Data Jemaat.....	87
A.6 Skenario <i>View</i> Kriteria Majelis Jemaat .....	89
A.7 Skenario <i>Update</i> Kriteria Majelis Jemaat.....	90
A.8 Skenario <i>View</i> Sub Kriteria Majelis Jemaat .....	92
A.9 Skenario <i>Update</i> Sub Kriteria Majelis Jemaat .....	93
A.10 Skenario <i>Rangking</i> .....	95
D.1 Tabel Kode Program Controller/anggota.php .....	108
D.2 Tabel Kode Program Controller/beranda.php .....	113
D.3 Tabel Kode Program Controller/kriteria.php .....	113
D.4 Tabel Kode Program Controller/landing.php.....	115
D.5 Tabel Kode Program Controller/rangking.php.....	116
D.6 Tabel Kode Program Controller/subkriteria.php.....	117
D.7 Tabel Kode Program Model/m_anggota.php .....	118
D.8 Tabel Kode Program Model/m_kriteria.php .....	120
D.9 Tabel Kode Program Model/m_login.php .....	121
D.10 Tabel Kode Program Model/m_subkriteria.php.....	121

## DAFTAR GAMBAR

Gambar 3.1 Tahapan Penelitian .....	9
Gambar 3.2 Alur Perancangan Model Waterfall.....	10
Gambar 3.3 Flowchart metode SMART Pemilihan Majelis Jemaat GPIB “Getsemani” Jember.....	11
Gambar 4.1 Bussiness Process .....	22
Gambar 4. 2 Use Case Diagram .....	23
Gambar 4. 3 Class Diagram .....	37
Gambar 4.4 Entity Relationship Diagrams (ERD).....	38
Gambar 4. 5 Kode Program tambah_anggota() .....	45
Gambar 4.6 Kode Program update_anggota() .....	52
Gambar 5.1 Halaman Home.....	58
Gambar 5.2 Halaman Login .....	59
Gambar 5.3 Halaman Utama Admin.....	59
Gambar 5.4 Halaman Jemaat .....	60
Gambar 5.5 Halaman Kriteria .....	60
Gambar 5.6 Halaman Sub Kriteria.....	60
Gambar 5.7 Form Tambah Data Jemaat .....	61
Gambar 5.8 Form Update Data Jemaat .....	61
Gambar 5.9 Alert Hapus Data Jemaat.....	62
Gambar 5.10 Form Edit Kriteria .....	62
Gambar 5.11 Form Data Sub Kriteria .....	63
Gambar 5.12 Form Edit Data Sub Kriteria .....	63
Gambar 5.13 Alerts Hapus Data Sub Kriteria.....	64
Gambar 5. 14 Halaman Rangking per Sektor .....	64
Gambar 5.15 Halaman Detail Perangkingan.....	65

B.1 Activity Login .....	98
B.2 Activity View Data Jemaat.....	98
B.3 Activity Insert Data Jemaat .....	99
B.4 Activity Update Data Jemaat.....	99
B.5 Activity Delete Data Jemaat.....	100
B.6 Activity View Kriteria Majelis Jemaat.....	100
B.7 Activity Update Kriteria Majelis Jemaat.....	101
B.8 Activity View Sub Kriteria Majelis Jemaat .....	101
B.9 Activity Update Sub Kriteria Majelis Jemaat.....	102
B.10 Activity Rangking .....	102
C.1 Sequence Login .....	103
C.2 Sequence <i>View</i> Data Jemaat .....	103
C.3 Sequence Insert Data Jemaat .....	104
C.4 Sequence Update Data Jemaat.....	104
C.5 Sequence Delete Data Jemaat.....	105
C.6 Sequence View Kriteria Majelis Jemaat.....	105
C.7 Sequence Update Kriteria Majelis Jemaat.....	106
C.8 Sequence View Sub Kriteria Majelis Jemaat .....	106
C.9 Sequence Update Sub Kriteria Majelis Jemaat .....	107
C.10 Sequence View Rangking.....	107



## DAFTAR LAMPIRAN

A.1 Skenario Login .....	79
A.2 Skenario View Data Jemaat .....	80
A.3 <i>Skenario Insert</i> Data Jemaat.....	82
A.4 <i>Skenario Update</i> Data Jemaat .....	85
A.5 Skenario <i>Delete</i> Data Jemaat.....	87
A.6 Skenario <i>View</i> Kriteria Majelis Jemaat .....	89
A.7 Skenario Update Kriteria Majelis Jemaat.....	90
A.8 Skenario View Sub Kriteria Majelis Jemaat .....	92
A.9 Skenario <i>Update</i> Sub Kriteria Majelis Jemaat .....	93
A.10 Skenario Rangking .....	95
B.1 Activity Login .....	98
B.2 Activity View Data Jemaat.....	98
B.3 Activity Insert Data Jemaat .....	99
B.4 Activity Update Data Jemaat.....	99
B.5 Activity Delete Data Jemaat .....	100
B.6 Activity View Kriteria Majelis Jemaat.....	100
B.7 Activity Update Kriteria Majelis Jemaat .....	101
B.8 Activity View Sub Kriteria Majelis Jemaat .....	101

B.9 Activity Update Sub Kriteria Majelis Jemaat.....	102
B.10 Activity Rangking .....	102
C.1 Sequence Login .....	103
C.2 Sequence <i>View</i> Data Jemaat .....	103
C.3 Sequence Insert Data Jemaat .....	104
C.4 Sequence Update Data Jemaat.....	104
C.5 Sequence Delete Data Jemaat.....	105
C.6 Sequence View Kriteria Majelis Jemaat.....	105
C.7 Sequence Update Kriteria Majelis Jemaat.....	106
C.8 Sequence View Sub Kriteria Majelis Jemaat .....	106
C.9 Sequence Update Sub Kriteria Majelis Jemaat .....	107
C.10 Sequence View Rangking.....	107
D.1 Tabel Kode Program Controller/anggota.php .....	108
D.2 Tabel Kode Program Controller/beranda.php .....	113
D.3 Tabel Kode Program Controller/kriteria.php .....	113
D.4 Tabel Kode Program Controller/landing.php.....	115
D.5 Tabel Kode Program Controller/rangking.php.....	116
D.6 Tabel Kode Program Controller/subkriteria.php.....	117
D.7 Tabel Kode Program Model/m_anggota.php .....	118
D.8 Tabel Kode Program Model/m_kriteria.php .....	120
D.9 Tabel Kode Program Model/m_login.php .....	121
D.10 Tabel Kode Program Model/m_subkriteria.php.....	121

## **BAB 1. PENDAHULUAN**

### **1.1 Latar Belakang**

Gereja Protestan di Indonesia bagian Barat (disingkat GPIB) adalah persekutuan orang percaya Kristen Protestan di Indonesia dimana Tuhan Yesus Kristus menjadi dasar dan kepalanya. GPIB adalah bagian dari Gereja Protestan di Indonesia (GPI) yang telah ada terlebih dahulu pada tahun 1605 saat jaman Hindia Belanda. Gereja Protestan di Indonesia (GPI) adalah ibu dari 12 Gereja Bagian Mandiri di seluruh persada Nusantara, mulai dari Gereja Masehi Injili di Minahasa (GMIM), Gereja Protestan Maluku (GPM), Gereja Masehi Injili di Timor (GMIT), Gereja Protestan Indonesia di Donggala (GPIB), Gereja Protesta di Indonesia di Buol Tolitoli (GPIBT), Gereja Protestan Indonesia di Gorontalo (GPIG), Gereja Kristen Luwuk Banggai (GKLB), Gereja Protestan Indonesia di Papua (GPI Papua), Indonesia Evangelical Christian Church (IECC), Gereja Protestan di Banggai Kepulauan ( GPIBK), Gereja Masehi Injili di Talaud (GERMITA) dan Gereja Protestan di Indonesia bagian Barat (GPIB). Melalui Surat Keputusan Wakil Tinggi Kerajaan Belanda di Indonesia No. 2, tanggal 1 Desember 1948 akhirnya diputuskan dan disetujui untuk GPIB menjadi gereja mandiri keempat di lingkungan GPI.

GPIB Getsemani Jember beralamat di Jalan PB Sudirman no. 1 Jember dan memiliki sekitar lebih dari 450 jemaat. Dari banyaknya jemaat yang dimiliki GPIB Getsemani Jember, maka dibentuklah suatu konsep sektor pelayanan guna memetakan jemaat dan memudahkan dalam pelayanan. Terdapat 5 sektor pelayanan di GPIB Getsemani Jember, yaitu sektor Sion, sektor Zaitun, sektor Moria, sektor Torsina dan sektor Sinai.

Majelis Jemaat dibentuk untuk mengkoordinir segala kebutuhan dan kegiatan gereja agar dapat terlaksana dengan baik. Majelis jemaat terdiri dari Pendeta yang ditugaskan oleh Majelis Sinode di jemaat, Diaken dan Penatua yang dipilih dari dan oleh warga sidi menurut Tata Gereja GPIB Tahun 2010 Peraturan

Nomor 1 tentang Presbiter dan Tata cara Pemilihan Presbiter Pasal 3 dan Petunjuk Pelaksanaan Pemilihan Diaken dan Penatua yang ditetapkan oleh Majelis Sinode. Jumlah Diaken dan Penatua yang dipilih sebagai Fungsiaris Majelis Jemaat ditetapkan dalam Sidang Majelis Jemaat menurut kebutuhan GPIB Jemaat "GETSEMANI" sesuai dengan Peraturan Pemilihan Diaken dan Penatua GPIB yang berlaku. Masa tugas Fungsiaris Majelis Jemaat ditetapkan selama 5 (lima) tahun dan dapat dipilih kembali. Majelis Jemaat diwakili oleh Ketua dan Sekretaris Pelaksana Harian Majelis Jemaat.

Jemaat yang dapat dipilih menjadi Majelis paling tidak sudah menjadi anggota jemaat gereja selama 1 tahun dan memiliki beberapa kriteria, yaitu :

1. Memiliki moralitas yang baik.
2. Memiliki karakter yang baik.
3. Memiliki kompetensi yang baik.
4. Pemahaman tentang ajaran Kristen yang baik.

Majelis Jemaat harus bisa menjadi penutan bagi sekitarnya serta memposisikan diri pada saat terjadi perselisihan diantara jemaat. Selain itu semua, menjadi Majelis Jemaat haruslah berlandaskan kasih dan panggilan dari Tuhan itu sendiri, bukan karena keegoisan dan duniawi semata.

Pemilihan Majelis Jemaat di GPIB "Getsemani" Jember masih memakai cara manual, di mana semua jemaat tetap bisa terpilih menjadi majelis gereja. Pemilihan manual tersebut berlangsung secara demokratis dengan menggunakan metode voting untuk setiap nama yang bisa dipilih, lalu nantinya akan keluar beberapa nama yang memenuhi syarat untuk menjadi Majelis Jemaat. Hal ini dirasa kurang efisien dan kurang tepat sasaran mengingat jumlah jemaat yang terus bertambah serta adanya kriteria-kriteria dari Alkitab yang diabaikan. Misal, ada seorang majelis gereja terpilih yang berselisih dengan jemaat lain di gereja tersebut. Akibatnya bukan damai yang dibawa, melainkan ketidak nyamanan dalam kegiatan gereja, padahal salah satu kewajiban majelis jemaat gereja adalah menyelesaikan segala perselisihan di antara sesama anggota jemaat dengan damai. Maka dari itu

diperlukan suatu sistem untuk menunjang dan membantu dalam pemilihan majelis jemaat agar hasil diinginkan dapat tercapai dengan maksimal.

Skripsi ini menggunakan metode SMART untuk membangun SPK Pemilihan Majelis Jemaat. Metode SMART dipilih karena algoritmanya cukup sederhana sehingga tidak membutuhkan proses komputasi yang kompleks. Hal ini sangat membantu untuk menyelesaikan suatu problem keputusan yang memiliki banyak atribut dan alternatif keputusan. Selain itu metode ini juga memiliki formulasi untuk menanggulangi konflik antar kriteria yang biasa terjadi dalam pengambilan keputusan dengan banyak kriteria. Berdasarkan masalah di atas, diperlukan suatu penelitian untuk merancang dan membangun sistem pendukung keputusan yang dapat digunakan untuk membantu memilih majelis jemaat di Gereja Protestan di Indonesia bagian Barat (GPIB). Sistem yang memungkinkan pengguna merancang penunjang keputusannya dengan mudah dan cepat.

## **1.2 Rumusan Masalah**

Berdasarkan permasalahan di atas, maka rumusan masalah dalam penelitian ini sebagai berikut:

1. Bagaimana metode SMART dapat digunakan dalam perhitungan pemilihan majelis jemaat di GPIB Getsemani Jember?
2. Bagaimana merancang dan membangun Sistem Pendukung Keputusan pemilihan majelis jemaat menggunakan metode SMART?

## **1.3 Tujuan**

Tujuan dari penelitian ini adalah:

1. Menentukan pemilihan Majelis Jemaat di GPIB Jember.
2. Membangun sistem penunjang keputusan yang dapat membantu GPIB Getsemani Jember dalam pengambilan keputusan untuk menentukan majelis baru.

#### **1.4 Batasan Masalah**

Penelitian dari objek yang dibahas agar tidak terjadi penyimpangan dalam proses pembuatan sistem maka, akan dipaparkan batasan masalah sebagai berikut:

1. Model perancangan yang akan digunakan adalah model *waterfall*.
2. Sistem ini hanya memberikan rekomendasi Majelis Jemaat yang sesuai dengan kriteria.
3. Perubahan nilai bobot kriteria hanya dapat dilakukan satu kali setiap pembaharuan data calon Majelis Jemaat.

## **BAB 2. TINJAUAN PUSTAKA**

### **2.1 Penelitian Terdahulu**

Metode SMART pernah digunakan dalam Sistem Pendukung Keputusan Berbasis Web Untuk Pemilihan Cafe (Novianti, 2016). Penelitian pemilihan cafe ini memberikan informasi mengenai café yang direkomendasikan kepada para konsumen. Sistem ini akan memberikan informasi tujuan cafe yang dapat dikunjungi oleh konsumen sesuai dengan kriteria yang diinputkan. Hal ini berhubungan dengan pemilihan anggota gereja yang sesuai dengan kriteria agar bisa dipilih menjadi Majelis Jemaat.

Metode SMART juga digunakan untuk membangun Sistem Pendukung Keputusan Penentuan Pemberian RASKIN untuk Masyarakat (Fauziah, 2015). Pada penelitian ini dipaparkan mengenai sistem pendukung keputusan yang mampu menampilkan perangkaan data penerima RASKIN sesuai keteria bobot yang sudah ditetapkan agar pembagian RASKIN tetap sasaran dan transparan. Hal ini berkaitan dengan perangkaan akhir untuk setiap anggota jemaat yang nantinya bisa terpilih menjadi Majelis Jemaat.

Sistem Penunjang Keputusan Pemilihan Majelis pernah dibangun dengan Metode *Fuzzy-Analytic Hierarchy Process* (F-AHP) (Utut, 2015). Pada penelitian ini, Sistem Penunjang Keputusan Pemilihan Majelis akan dibangun dengan metode SMART. Metode FAHP dan SMART ini bekerja dengan cara yang sama yaitu pembobotan suatu kriteria. Hanya saja dalam pengerjaannya metode SMART dianggap lebih sederhana karena pembobotan pada SMART menggunakan skala 0 sampai 1, sehingga mempermudah perhitungan dan perbandingan nilai pada masing-masing alternative. Dengan kriteria yang terbatas dan cenderung sederhana maka metode SMART dianggap lebih cocok untuk menangani problem tersebut, karena dalam penggunaan *Fuzzy-Analytic Hierarchy Process* mencakup sector yang lebih luas dan lebih kompleks.

## 2.2 Metode Simple Multi Attribute Rating Technique (SMART)

Simple Multi Attribute Rating Technique (SMART) merupakan metode perankingan yang banyak digunakan dalam sistem penunjang keputusan. SMART merupakan teknik pengambilan keputusan multi kriteria yang didasarkan pada teori bahwa setiap alternatif terdiri dari sejumlah kriteria yang memiliki nilai-nilai dan setiap kriteria memiliki bobot yang menggambarkan seberapa penting ia dibandingkan dengan kriteria lain. Pembobotan ini digunakan untuk menilai setiap alternatif agar diperoleh alternatif terbaik (Suryanto, Muhammad Safrizal, 2015). Teknik pengambilan keputusan multi kriteria ini digunakan untuk mendukung pengambilan keputusan dalam memilih beberapa alternatif terbaik dari beberapa alternatif yang diberikan. Alternatif yang dipilih harus sesuai dengan tujuan yang telah dirumuskan pada perancangan sistem penunjang keputusan yang dibuat. SMART lebih banyak digunakan karena kesederhanaanya dalam merespon kebutuhan pembuat keputusan dan caranya menganalisa respon. Analisa yang terlibat adalah transparan sehingga metode ini memberikan pemahaman masalah yang tinggi dan dapat diterima oleh pembuat keputusan. SMART menggunakan linear additive model untuk meramal nilai setiap alternatif. Kefleksibelan menjadi kelebihan dari metode SMART. Model fungsi utiliti linear yang digunakan oleh SMART dapat dilihat pada Persamaan 2.1

$$SMART = \sum_{j=1}^k w_j u_{ij} \dots\dots\dots 2.1$$

- j adalah nilai pembobotan kriteria ke- j dari k kriteria,
- u ij adalah nilai utility alternatif i pada kriteria j .
- Pemilihan keputusan adalah mengidentifikasi mana dari n alternatif yang mempunyai nilai fungsi terbesar.
- Nilai fungsi ini juga dapat digunakan untuk meranking n alternatif.



Menghitung nilai normalisasi bobot dapat dilihat pada Persamaan 2.2

$$Normalisasi = \frac{w_j}{\sum w_j} \dots\dots\dots 2.2$$

Keterangan:  $w_j$  : bobot suatu kriteria

- $\sum w_j$  : Total bobot semua kriteria
- $nw_j$  adalah normalisasi bobot kriteria ke- j

Menghitung nilai utiliti dapat dilihat pada Persamaan 2.3

$$\sum w_j = u_i(a_i) = 100 \frac{(C_{max} - C_{out\ i})}{(C_{max} - C_{min})} \% \dots\dots\dots 2.3$$

Keterangan:

$u_i(a_i)$  : nilai utility kriteria ke-i untuk kriteria ke-i

$C_{max}$  : nilai kriteria maksimal

$C_{min}$  : nilai kriteria minimal

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

### 2.3 Langkah Pengujian Metode SMART

Dalam penelitian digunakan 2 langkah pengujian untuk metode SMART. Yang pertama dengan membandingkan perhitungan sistem dengan manual. Pada langkah pertama ini digunakan *Microsoft Excel* dikarenakan sistem masih belum terbentuk.

Langkah kedua adalah dengan mencocokkan dan mengkonsultasikan perhitungan manual dengan pakar. Dalam hal ini yang dimaksud pakar adalah pihak dari Gereja Protetan di Indonesia bagian Barat (GPIB). Hal ini dilakukan untuk mengetahui apakah ada perbedaan hasil dari pihak Gereja Protetan di Indonesia bagian Barat (GPIB) dengan perhitungan menggunakan metode SMART dalam hasil akhir pemilihan majelis jemaat

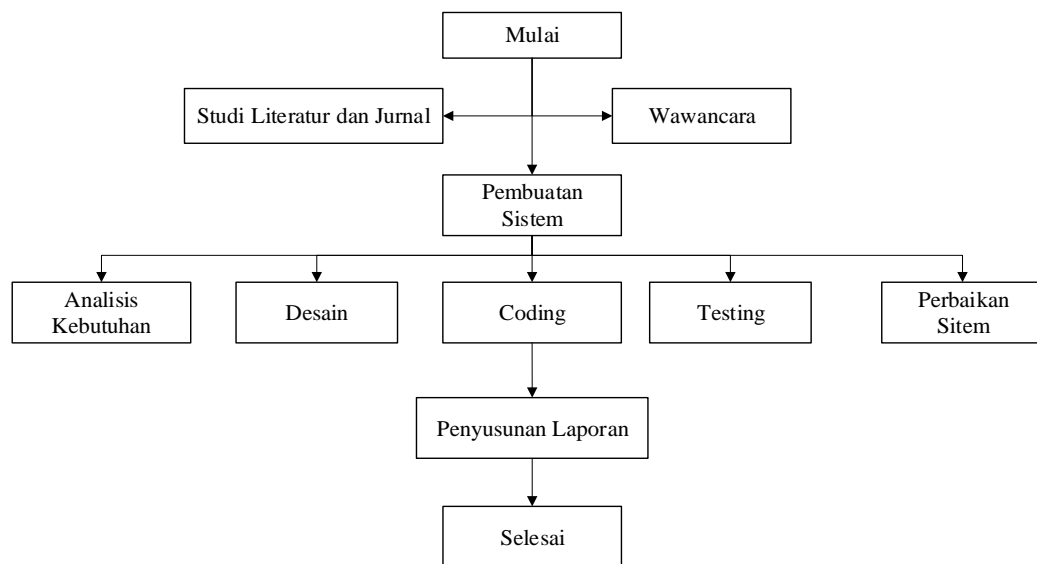
## **BAB 3. METODOLOGI PENELITIAN**

### **3.1 Jenis Penelitian**

Jenis penelitian yang digunakan pada penelitian ini merupakan jenis penelitian pengembangan, karena tujuan penelitian adalah untuk membangun sebuah sistem penunjang pengambilan keputusan. Penelitian ini bertujuan untuk mengembangkan proses yang masih manual menjadi sistem terkomputerisasi untuk menyempurnakan proses yang sudah ada dengan mengimplementasikan Simple Multi Attribute Technique Rating (SMART) sebagai metode penilaian untuk menghasilkan keluaran yang lebih akurat dan sesuai dengan yang dikehendaki.

### **3.2 Tahapan Penelitian**

Model pengembangan sistem yang digunakan dalam penelitian ini adalah teknik pengembangan sistem model waterfall. Model waterfall menerapkan proses dasar dari spesifikasi, pengembangan, validasi dan evolusi sebuah sistem berdasarkan pendekatan sistematis mulai dari analisis kebutuhan, perancangan sistem, implementasi sistem, pengujian sistem, dan lain sebagainya (Sommerville, 2011). Model pengembangan waterfall ini berdasarkan kebutuhan pengguna yang merupakan prioritas utama. Disebut dengan waterfall karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan. Sebagai contoh tahap desain harus menunggu selesainya tahap sebelumnya yaitu tahap analisis. Tahapan penelitian digambarkan pada Gambar 3.1.



**Gambar 3.1 Tahapan Penelitian**

### **3.2.1 Studi Literatur dan Jurnal**

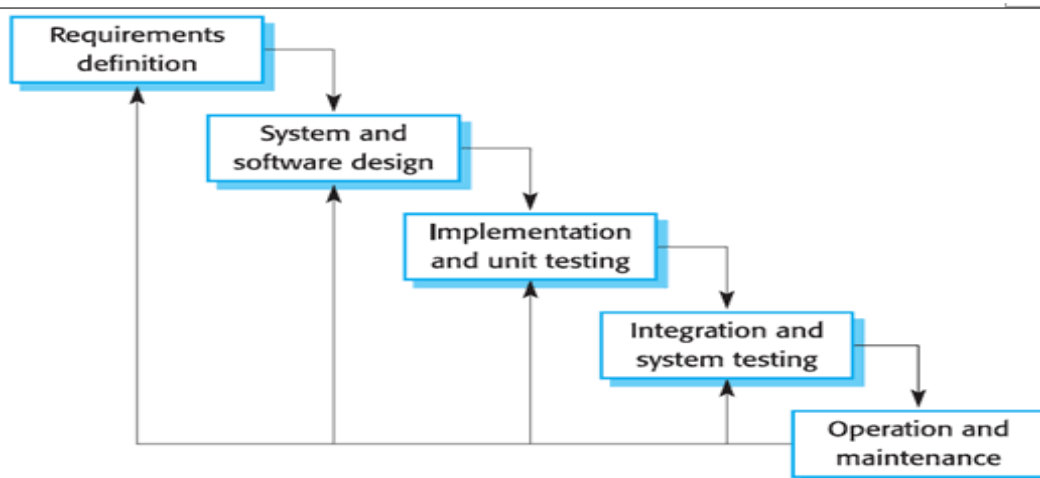
Studi literatur bertujuan untuk memperkuat pemilihan metode penelitian dan sebagai pembanding dengan penelitian terdahulu yang telah dilakukan. Jurnal yang digunakan adalah jurnal penelitian terdahulu yang berfungsi untuk menunjang data penelitian

### **3.2.2 Wawancara**

Wawancara dilakukan di Gereja Protestan di Indonesia Bagian Barat (GPIB). Dengan mewawancarai pakar diharapkan data yang dibutuhkan seperti daftar anggota jemaat dan kriteria – kriteria yang dibutuhkan agar dapat menjadi Majelis Jemaat dapat terpenuhi. Waktu wawancara dilakukan selama tiga bulan, dimulai bulan Maret 2018 sampai dengan bulan Juni 2018.

### 3.2.3 Pembuatan Sistem

Pembuatan sistem dibangun berdasarkan analisis dan pengumpulan data untuk membangun sistem yang sesuai dengan kebutuhan pada Sistem Pendukung Keputusan Pemilihan Majelis Jemaat Gereja GPIB Jember Menggunakan Metode Simple Multi Attribute Rating Technique (SMART). Model pengembangan yang digunakan dalam perancangan sistem ini yaitu metode *Waterfall*. Tahapan pengembangan perangkat lunak *Waterfall* dibedakan menjadi beberapa tahapan yang runtun mulai dari analisis kebutuhan, design sistem, Coding & Testing (Pressman, 2009). Alur perancangan model *waterfall* dapat dilihat pada Gambar 3.2.

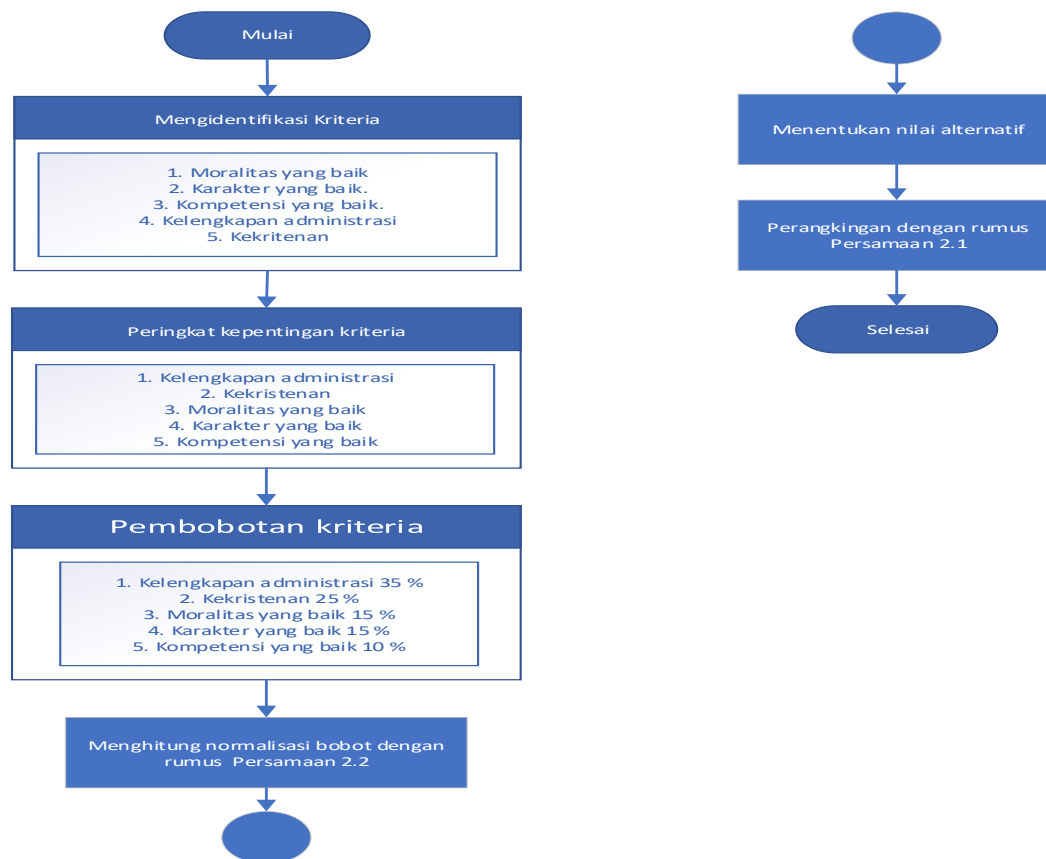


**Gambar 3.2** Alur Perancangan Model *Waterfall*

#### 3.2.3.1 Analisis Kebutuhan

Analisis kebutuhan sistem dalam penelitian ini adalah analisis data yang dibutuhkan dalam pembangunan sistem. Untuk memahami sifat program yang akan dibangun, maka harus memahami informasi yang dibutuhkan untuk perangkat lunak, fungsi yang diperlukan, alur, kinerja, dan interface dari program yang akan dibangun (Pressman, 2001).

Penelitian ini menggunakan metode SMART dalam pengerjaannya. Langkah pengerjaannya pertama dimulai dari mengidentifikasi kriteria yang nantinya akan diperingkat mulai dari kriteria yang paling utama sampai paling bawah. Kemudian kriteria tadi diberi nilai bobot untuk masing – masingnya yang lalu di normalisasi menggunakan Persamaan 2.2. Setelah normalisasi bobot kriteria selanjutnya akan ditentukan nilai alternatif tiap – tiap kriteria yang sesudah itu baru bias di buat perankingan menggunakan Persamaan 2.1. Setelah rangking didapatkan maka akan keluar data yang dibutuhkan. Langkah – langkah tersebut dapat dilihat pada Gambar 3.3



**Gambar 3.3 Flowchart metode SMART Pemilihan Majelis Jemaat GPIB “Getsemani” Jember**

Tahap ini meliputi tahap desain sistem dengan pembuatan diagram menggunakan toolsUML Visual Paradigm. Diagram-digram tersebut akan

digunakan sebagai acuan pembuatan sistem pada tahap implementasi. Diagram-diagram yang digunakan yang digunakan sebagai berikut:

1. Business Process digunakan untuk mendefinisikan aktifitas dan proses
2. Use Case Diagram digunakan untuk mendefinisikan fungsional sistem
3. Scenario digunakan untuk menjelaskan fitur sistem
4. Sequence Diagram digunakan untuk menunjukkan rangkaian pesan yang dikirim antar object juga interaksi antar object.
5. Class Diagram digunakan untuk menggambarkan struktur statis class dalam sistem.
6. Entity Realationship Diagram digunakan untuk menunjukkan relasi antar object

### **3.2.3.2 Implementasi Koding**

Implementasi merupakan tahapan dimana seluruh rancangan desain system yang telah dibuat diimplementasikan ke dalam bentuk program agar terbentuk Aplikasi yang sesuai. *Tools* yang digunakan pada penelitian ini adalah *Sublime Text* sebagai editor kode dengan Bahasa pemrograman HTML, CSS, dan PHP, dan Xampp sebagai manajemen database.

### **3.2.3.3 Pengujian Sistem**

Tahap pengujian sistem (testing) ini dilakukan setelah proses pembuatan sistem (coding) telah selesai dilakukan. Testing software adalah proses mengoperasikan software dalam suatu kondisi yang di kendalikan untuk memverifikasi apakah sistem telah berjalan sebagaimana telah ditetapkan menurut spesifikasi, mendeteksi error, dan validasi apakah spesifikasi yang telah ditetapkan sudah memenuhi keinginan atau kebutuhan dari pengguna yang sebenarnya (Romeo, 2003). Proses pengujian sistem dilakukan dengan black box. Metode pengujian black box dilakukan untuk mengetahui apakah ada kesalahan dalam fungsionalitas sistem yang telah dibangun.

### 3.2.3.4 Perawatan Sistem

Tahap perawatan sistem (maintenance) ini dilakukan setelah sistem selesai dibangun dan diuji. Karena sistem ini digunakan hanya pada saat Pemilihan Majelis Jemaat, maka perawatan dilakukan pada 3 bulan sebelum pemilihan dilakukan.

### 3.2.4 Uji Efektivitas Metode SMART

Uji efektivitas metode digunakan untuk menganalisis hasil dari penghitungan data dengan menggunakan metode SMART. Dari sini nantinya dapat diketahui apakah metode SMART dapat dipakai menjadi metode yang efektif untuk digunakan dalam pemilihan Majelis Jemaat di GPIB Getsemani Jember. Uji efektivitas ini dilakukan dengan beberapa langkah, sebagai berikut:

1. Penentuan Kriteria.

Pada tahapan ini mencantumkan penentuan kriteria dari pemilihan Majelis Jemaat di GPIB Getsemani Jember yang ditunjukkan pada Tabel 3.1

**Tabel 3.1 Daftar Kriteria Pemilihan Majelis Jemaat GPIB Getsemani Jember**

No.	Kriteria
1.	Moralitas yang tinggi
2.	Karakter yang baik
3.	Kompetensi yang baik
4.	Kelengkapan administrasi
5.	Kekristenan

## 2. Penentuan Bobot Kriteria dan Normalisasi Bobot.

Pada tahapan ini mencantumkan penentuan bobot dari tiap kriteria lalu dinormalisasikan dengan rumus Persamaan 2.2 yang ditunjukkan pada Tabel 3.2.

**Tabel 3.2 Pentuan Bobot Kriteria dan Normalisasi Bobot**

No.	Kriteria	Bobot	Normalisasi Bobot
1.	Memiliki moralitas yang tinggi	50	0,25
2.	Karakter yang baik serta tidak tempramen	40	0,2
3.	Kompetensi yang baik	50	0,25
4.	Kelengkapan administrasi	30	0,15
5.	Kekristenan	30	0.15

## 3. Penentuan sub Kriteria

Pada tahapan ini mencantumkan penentuan sub dari tiap kriteria yang ada. Tiap kriteria memiliki masing – masing sub nya dengan jumlah yang berbeda yang ditunjukkan pada Tabel 3.3, Tabel 3.4, Tabel 3.5, Tabel 3.6, dan Tab. Disini saya misalkan tiap kriteria memiliki 4 sub kriteria.



**Tabel 3.3 Sub Kriteria Moralitas yang Tinggi**

No.	Sub kriteria	Nilai	Bobot
1.	sub 1	100	0,25
2.	sub 2	80	
3.	sub 3	60	
4.	sub 4	40	

**Tabel 3.4 Sub Kriteria Karakter yang Baik**

No.	Sub kriteria	Nilai	Bobot
1.	sub 1	100	0,25
2.	sub 2	80	
3.	sub 3	60	
4.	sub 4	40	

**Tabel 3.5 Sub Kriteria Kompetensi yang Baik**

No.	Sub kriteria	Nilai	Bobot
1.	sub 1	100	0,25
2.	sub 2	80	

Dilanjutkan

Lanjutan

3.	sub 3	60	
4.	sub 4	40	

**Tabel 3.6 Sub Kriteria Kelengkapan Administrasi**

No.	Sub kriteria	Nilai	Bobot
1.	sub 1	100	0,25
2.	sub 2	80	
3.	sub 3	60	
4.	sub 4	40	

**Tabel 3.7 Sub Kriteria Kekristenan**

No.	Sub kriteria	Nilai	Bobot
1.	sub 1	100	0,25
2.	sub 2	80	
3.	sub 3	60	
4.	sub 4	40	

#### 4. Memasukkan Nilai Alternatif

Pada tahapan ini mencantumkan penentuan nilai dari tiap calon Majelis Jemaat untuk setiap kriteria yang ditunjukkan pada tabel Tabel 3.8.

**Tabel 3.8 Nilai Alternatif**

No.	Alternatif	Kriteria				
		Moralitas yang tinggi	Karakter yang baik	Kompetensi yang baik	Kelengkapan Administrasi	Kekristenan
1.	Ridho	60	80	40	60	40
2.	Sandy	100	40	60	60	40
3.	Septa	40	40	100	80	60
4.	Ipat	40	60	80	60	100
5.	Shinta	40	80	100	60	60

#### 5. Perankingan.

Pada tahapan ini mencantumkan perankingan dari setiap calon berdasarkan semua perhitungan. Ranking ini menggunakan Persamaan 2.1, lalu diambil nilai yang paling tertinggi yang ditunjukkan pada Tabel 3.9 dan Tabel 3.10.

**Tabel 3.9 Tabel Perhitungan Akhir**

No.	Alternatif	Hasil
1.	Ridho	56
2.	Sandy	63
3.	Septa	64
4.	Ipat	66
5.	Shinta	69

Melalui data diatas maka nilai yang tertinggi adalah yang terpilih yaitu Shinta karena menempati urutan pertama dalam perankingan.

**Tabel 3.10 Tabel Perankingan**

No.	Alternatif	Hasil
1.	Shinta	69
2.	Ipat	66
3.	Septa	64
4.	Sandy	63
5.	Ridho	56

## **BAB 4.PENGEMBANGAN SISTEM INFORMASI**

### **4.1 Deskripsi Umum Sistem Informasi**

Sistem yang akan dibuat yaitu Sistem Pendukung Keputusan Pemilihan Majelis Jemaat Gereja Protestan di Indonesia bagian Barat “Getsemani” Jember Menggunakan Metode *Simple Multiple Attribute Rating Technique (SMART)* . Sistem ini dikembangkan dengan tujuan untuk mempermudah para *jemaat* yang akan memilih majelis jemaat gereja sesuai kriteria kebutuhan yang diinginkannya di gereja tersebut. Aktor yang terlibat dalam aplikasi ini yaitu *administrator* (pihak panitia penyelenggaraan pemilihan majelis jemaat) yang mempunyai tugas manajemen data calon majelis jemaat, menentukan kriteria dan sub kriteria majelis jemaat, serta menetapkan nilai setiap majelis jemaat dengan melakukan perhitungan oleh sistem. Kemudian *user* (jemaat gereja) yang dapat melihat hasil pemilihan majelis jemaat berdasarkan kebutuhan yang telah ditetapkan. Kemudian sistem akan memproses daftar calon majelis jemaat dan melanjutkan ke tahap selanjutnya yakni sistem akan menampilkan hasil perankingan berdasarkan data yang sudah terkumpul menurut kebutuhannya. Hasil dari pemilihan majelis jemaat tersebut dapat dilihat oleh *user* itu sendiri. Hasil akhir dari sistem ini yakni output berupa nama serta rangking yang diperoleh dari nilai pembobotan yang dilakukan oleh *administrator* serta perhitungan yang dilakukan sistem dengan rumus melalui metode SMART.

### **4.2 Analisis Kebutuhan Sistem Informasi**

Data yang didapatkan peneliti dari berbagai literatur dan hasil wawancara serta observasi selanjutnya dianalisis sehingga dapat memenuhi kebutuhan sistem yang didefinisikan dalam bentuk kebutuhan fungsional dan kebutuhan nonfungsional. Kebutuhan fungsional adalah kebutuhan yang menggambarkan proses yang berkaitan dengan fungsi dari sistem dan juga kebutuhan yang menjadi

karakteristik yang dimiliki sistem. Sedangkan kebutuhan nonfungsional adalah kebutuhan yang tidak terkait secara langsung dengan fungsi sistem.

#### **4.2.1 Kebutuhan Fungsional**

Kebutuhan fungsional dari sistem pendukung keputusan pemilihan Majelis Jemaat “Getsemani” Jember adalah sebagai berikut :

1. Sistem dapat menampilkan halaman login, data jemaat tetap, data majelis, bobot kriteria dan sub kriteria, kriteria majelis, data hasil perankingan.
2. Sistem dapat memajemen data jemaat, data majelis, data bobot kriteria dan data bobot sub kriteria dengan melakukan input, edit dan *delete*.
3. Sistem dapat melakukan perhitungan menggunakan metode SMART yang dapat menjadi alternatif pemilihan Majelis Jemaat “Getsemani” Jember.

#### **4.2.2 Kebutuhan Nonfungsional**

Kebutuhan nonfungsional dari sistem pendukung keputusan pemilihan Majelis Jemaat “Getsemani” Jember sebagai berikut :

1. Sistem dapat berjalan di berbagai *web browser*.
2. Tampilan sistem mudah dimengerti (*user friendly*).

#### **4.2.3 Kebutuhan Antarmuka Pemakai**

Pengguna berinteraksi dengan sistem menggunakan *web browser*. Pengguna mengolah data (*input, edit, delete*) menggunakan tombol-tombol yang tampil di *web browser*.

#### 4.2.4 Kebutuhan Antarmuka Perangkat Keras

Kebutuhan perangkat keras yang dapat digunakan untuk mengakses sistem penunjang keputusan ini, adalah :

1. PC (*Personal Computer*) atau laptop.
2. *Keyboard*.
3. *Mouse*.

#### 4.2.5 Kebutuhan Perangkat Lunak

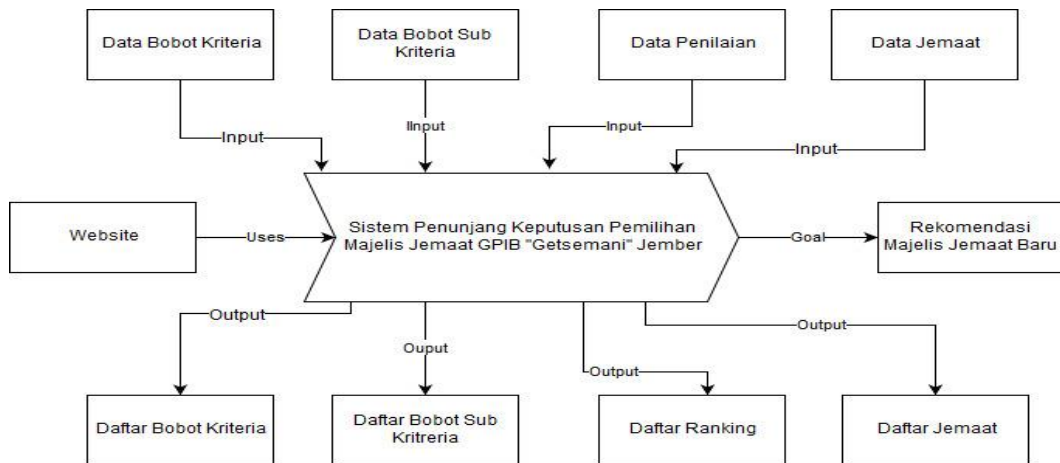
Kebutuhan perangkat lunak yang digunakan dalam sistem penunjang keputusan pemilihan Majelis Jemaat “Getsemani” Jember dirancang menggunakan bahasa pemrograman PHP dengan database *mySQL*.

### 4.3 Desain Sistem

Tahapan yang dilakukan setelah melakukan analisis kebutuhan sistem informasi yaitu tahap perencanaan pengembangan sistem informasi yang dapat digambarkan dengan desain sistem informasi. Desain sistem informasi ini meliputi elemen sistem informasi, *usecase diagram*, skenario, *activity diagram*, *sequence diagram*, *class diagram*, dan *entity relationship diagram* (ERD).

#### 4.3.1 *Bussiness Process*

*Bussines process* merupakan kumpulan kegiatan yang saling terkait dan berhubungan untuk menyelesaikan suatu masalah dan untuk mencapai tujuan tertentu. *Bussiness Process* dari sistem pendukung keputusan pemilihan Majelis Jemaat GPIB “Getsemani” Jember dapat dilihat pada Gambar 4.1



**Gambar 4.1** *Bussiness Process*

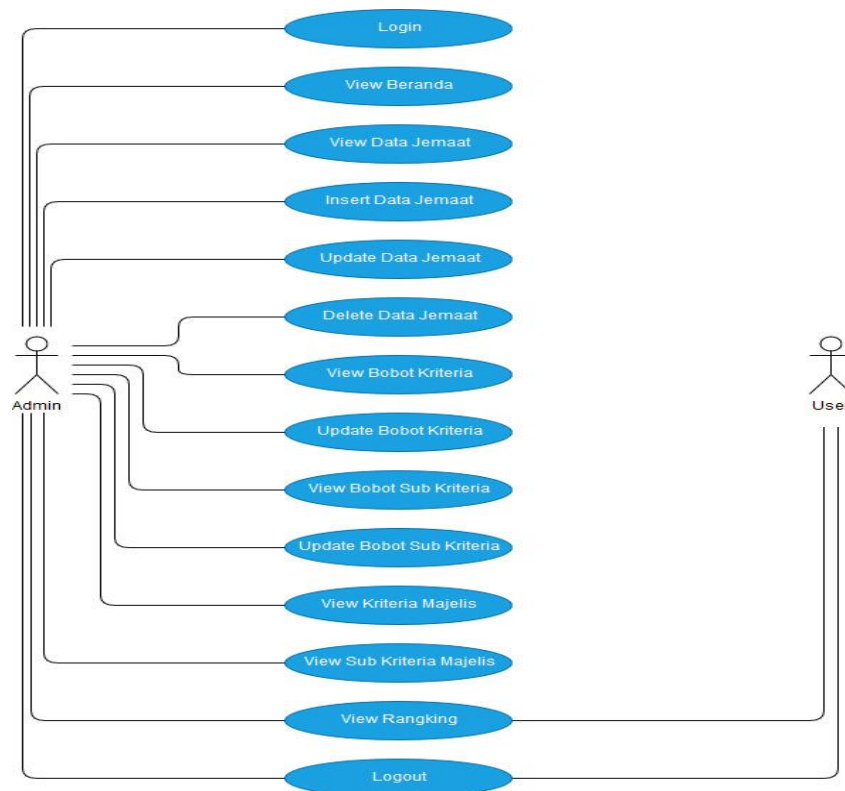
Pada gambar 4.1 menjelaskan bahwa Sistem Penunjang Keputusan Pemilihan Majelis Jemaat GPIB “Getsemani” Jember dengan Metode *SMART* terdiri dari inputan berupa data bobot kriteria, data bobot sub kriteria, data jemaat tetap, dan data penilaian kepada setiap jemaat. Data-data yang digunakan nantinya akan diolah menggunakan persamaan dalam metode *SMART* untuk mendapatkan rekomendasi Majelis Jemaat gereja baru yang sesuai dengan kriteria.

Sistem ini berbasis *web*. Tujuan dari sistem ini adalah menghasilkan rekomendasi Majelis Jemaat GPIB “Getsemani” Jember yang sesuai dengan kriteria yang ditetapkan.

#### 4.3.2 *Use Case Diagram*

*Use case diagram* digunakan untuk menggambarkan kegiatan yang dilakukan sistem dan aktor yang berjalan ketika melakukan suatu fungsi. *Use case diagram* dari sistem pendukung keputusan pemilihan Majelis dapat dilihat pada gambar 4.2





**Gambar 4. 2 Use Case Diagram**

Gambar 4.2 menunjukkan *use case diagram* sistem penunjang keputusan majelis jemaat GPIB “Getsemani” Jember yang terdiri dari 2 aktor.

Use case sistem penunjang keputusan majelis jemaat GPIB “Getsemani” Jember ini mempunyai penjelasan berupa tabel definisi aktor yang menggambarkan tugas-tugas aktor dalam mengoperasikan sistem tersebut dapat dilihat pada Tabel 4.1 Definisi Aktor. Penjelasan lainnya yaitu disebut definisi use case yang menggambarkan fungsionalitas dapat dilihat pada Tabel 4.2 Definisi Use Case.

**Tabel 4.1 Definisi Aktor**

No.		Aktor	Deskripsi
1.		Admin	Aktor Admin pada sistem informasi ini memiliki hak akses untuk: <ol style="list-style-type: none"> <li>1. Mengelola data calon Majelis Jemaat</li> <li>2. Mengelola data kriteria Majelis Jemaat</li> <li>3. Mengelola data sub kriteria Majelis Jemaat</li> <li>4. Melihat data kriteria Majelis Jemaat</li> <li>5. Melihat data sub kriteria Majelis Jemaat</li> <li>6. Melihat ranking Majelis Jemaat terpilih</li> <li>7. Melihat hasil perhitungan pemilihan Majelis Jemaat</li> </ol>
2.		User	Aktor User pada sistem informasi ini memiliki hak akses untuk: <ol style="list-style-type: none"> <li>1. Melihat ranking Majelis Jemaat terpilih</li> </ol>

**Tabel 4.2 Definisi Usecase**

No.	Usecase	Deskripsi
	Login	Menggambarkan proses autentifikasi pengguna yang digunakan untuk masuk ke sistem informasi.
2	View Beranda	Berfungsi untuk menampilkan halaman utama sistem
3	View Data Jemaat	Berfungsi untuk menampilkan halaman data jemaat.

Dilanjutkan

Lanjutan

4	<i>Insert</i> Data Jemaat	Berfungsi untuk menambah data jemaat baru
5	<i>Update</i> Data Jemaat	Berfungsi untuk mengubah data jemaat
6	<i>Delete</i> Data Jemaat	Berfungsi untuk menghapus data jemaat
7	<i>View</i> Bobot Kriteria	Berfungsi untuk menampilkan data hasil penghitungan bobot kriteria dengan metode SMART
8	<i>Update</i> Bobot Kriteria	Berfungsi untuk meng- <i>update</i> bobot kriteria
9	<i>View</i> Bobot Sub Kriteria	Berfungsi untuk menampilkan data penghitungan bobot sub kriteria dengan metode SMART
10	<i>Update</i> Bobot Sub Kriteria	Berfungsi untuk meng- <i>update</i> bobot sub kriteria
11	<i>View</i> Kriteria Majelis	Berfungsi untuk menampilkan informasi mengenai kriteria-kriteria majelis
12	<i>View</i> Sub Kriteria Majelis	Berfungsi untuk menampilkan informasi mengenai sub kriteria mengenai majelis
13	<i>View</i> Rangking	Berfungsi untuk menampilkan data ranking penilaian setelah diproses dengan metode SMART
14	<i>Logout</i>	Berfungsi untuk keluar dari hak akses admin/operator menjadi user

#### 4.3.3 Use Case Skenario

Skenario sistem berfungsi untuk menjelaskan alur dari sebuah sistem serta alur alternatif yang dilakukan oleh para aktor yang menggunakan sistem ini. Skenario sistem sesuai dengan yang ada pada *Use case diagram* seperti pada Gambar 4.2.

#### **4.3.3.1 Skenario *Login***

*Use Case* Skenario *Login* merupakan alur dari aktor dan sistem jika akan masuk dalam sistem. Aktor pertama kali masuk ke halaman utama untuk melakukan *Login* agar dapat mengakses sistem. Setelah aktor sudah melakukan *Login* dengan memasukkan *username* dan *password* maka sistem akan merekasi dengan membuka halaman admin. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif skenario *Login* ditunjukkan pada lampiran A.1

#### **4.3.3.2 Skenario *View Data Jemaat***

*Use case* skenario *view* data jemaat menampilkan gambaran mengenai bagaimana aktor dan sistem bekerja untuk menampilkan data jemaat. Aktor yang akan melihat data jemaat lebih dahulu harus masuk ke halaman jemaat. Sistem akan menampilkan halaman data jemaat ketika aktor sudah ke halaman data jemaat. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif *view* data jemaat ditunjukkan pada lampiran A.3

#### **4.3.3.3 Skenario *Insert Data Jemaat***

*Use case* skenario *insert* data jemaat menampilkan gambaran mengenai bagaimana aktor dan sistem bekerja untuk menambahkan data jemaat. Aktor yang akan menambahkan data jemaat lebih dahulu harus masuk ke halaman jemaat. Sistem akan menampilkan halaman data jemaat, lalu aktor bisa menambahkan data jemaat dengan memilih menu tambah jemaat. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif *insert* data jemaat ditunjukkan pada lampiran A.4

#### **4.3.3.4 Skenario *Update Data Jemaat***

*Use case* skenario *update* data jemaat menampilkan gambaran mengenai bagaimana aktor dan sistem bekerja untuk pembaharuan data jemaat. Aktor yang akan memperbaharui data jemaat lebih dahulu harus masuk ke halaman jemaat. Sistem akan menampilkan halaman data jemaat, lalu aktor bisa memperbaharui data jemaat dengan memilih menu edit. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif *update* data jemaat ditunjukkan pada lampiran A.5

#### **4.3.3.5 Skenario *Delete Data Jemaat***

*Use case* skenario *delete* data jemaat menampilkan gambaran mengenai bagaimana aktor dan sistem bekerja untuk menampulkan data jemaat. Aktor yang akan menghapus data jemaat lebih dahulu harus masuk ke halaman jemaat. Sistem akan menampilkan halaman data jemaat, lalu aktor bisa menghapus data jemaat dengan memilih menu delete. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif *delete* data jemaat ditunjukkan pada lampiran A.6

#### **4.3.3.6 Skenario *View Kriteria Majelis Jemaat***

*Use case* skenario *view* kriteria Majelis Jemaat menampilkan gambaran mengenai bagaimana aktor dan sistem bekerja untuk menampilkan informasi tentang kriteria Majelis Jemaat yang dibutuhkan. Aktor dapat melihat kriteria majelis jemaat dengan memilih halaman kriteria terlebih dahulu, lalu sistem akan menampilkan halaman kriteria majelis jemaat. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif *view* kriteria majelis jemaat ditunjukkan pada lampiran A.7

#### **4.3.3.7 Skenario *Update* Kriteria Majelis Jemaat**

*Use case* skenario *update* bobot kriteria menampilkan gambaran mengenai bagaimana aktor dan sistem bekerja untuk mengubah informasi tentang kriteria Majelis Jemaat. Aktor yang akan memperbaharui kriteria majelis jemaat lebih dahulu harus masuk ke halaman kriteria. Sistem akan menampilkan halaman kriteria, lalu aktor bisa memperbaharui kriteria majelis jemaat dengan memilih menu edit. *Use case* skenario *update* kriteria Majelis Jemaat dapat dilihat pada lampiran A.8

#### **4.3.3.8 Skenario *View* Sub Kriteria Majelis Jemaat**

*Use case* skenario *view* kriteria Majelis Jemaat menampilkan gambaran mengenai bagaimana aktor dan sistem bekerja untuk menampilkan informasi tentang kriteria Majelis Jemaat yang dibutuhkan. Aktor dapat melihat sub kriteria majelis jemaat dengan memilih halaman kriteria terlebih dahulu, lalu sistem akan menampilkan halaman sub kriteria majelis jemaat. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif *view* sub kriteria majelis jemaat ditunjukkan pada lampiran A.9

#### **4.3.3.9 Skenario *Update* Sub Kriteria Majelis Jemaat**

*Use case* skenario *update* bobot sub kriteria menampilkan gambaran mengenai bagaimana aktor dan sistem bekerja untuk mengubah informasi tentang kriteria Majelis Jemaat. Aktor yang akan memperbaharui sub kriteria majelis jemaat lebih dahulu harus masuk ke halaman sub kriteria. Sistem akan menampilkan halaman kriteria, lalu aktor bisa memperbaharui sub kriteria majelis jemaat dengan memilih menu edit. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif *update* sub kriteria majelis jemaat ditunjukkan pada lampiran A.10

#### **4.3.3.10 Skenario View Ranking**

*Use case* skenario *view ranking* menampilkan gambaran mengenai bagaimana aktor dan sistem bekerja untuk menampilkan halaman *ranking* untuk rekomendasi Majelis Jemaat baru. Aktor yang akan melihat *ranking* pemilihan majelis jemaat lebih dahulu harus masuk ke halaman *ranking* lalu sistem akan menampilkan halaman *ranking* . Penjelasan urutan aksi aktor dan reaksi sistem pada skenario utama dan skenario alternatif *view ranking* ditunjukkan pada lampiran A.11

#### **4.3.4 Activity Diagram**

*Activity diagram* merupakan diagram yang menggambarkan semua proses yang terjadi pada sistem pendukung keputusan Pemilihan Majelis Jemaat GPIB “Getsemani” Jember.

##### **4.3.4.1 Activity Diagram Login**

*Activity diagram Login* menampilkan gambaran proses mengenai bagaimana aktor mengakses sistem informasi dengan menggunakan *username* dan *password* yang telah disediakan serta masuk sesuai hak akses yang ada. Jika aktor salah memasukkan *username* atau *password*, maka sistem akan memunculkan pop up “This field is required”. Penjelasan proses pada *activity diagram Login* ditunjukkan pada lampiran B.1

##### **4.3.4.2 Activity Diagram View Data Jemaat**

*Activity diagram view data jemaat* menampilkan gambaran proses mengenai bagaimana aktor melihat data jemaat. Aktor dapat melakukan proses *view* dengan memilih menu jemaat yang kemudian sistem akan mengalihkan ke halaman data

jemaat. Penjelasan proses pada *activity diagram Logout* ditunjukkan pada lampiran B.3

#### **4.3.4.3 Activity Diagram Insert Data Jemaat**

*Activity diagram insert* data jemaat menampilkan gambaran proses mengenai bagaimana aktor memasukkan data jemaat. Aktor dapat melakukan proses *insert* dengan memilih menu jemaat yang kemudian sistem akan mengalihkan ke halaman data jemaat. Aktor memilih tombol tambah jemaat yang kemudian memunculkan form data jemaat. Aktor mengisi form data jemaat, kemudian klik tombol simpan untuk menyimpan data jemaat ke dalam database. Penjelasan proses pada *activity diagram Insert* data jemaat ditunjukkan pada lampiran B.4

#### **4.3.4.4 Activity Diagram Update Data Jemaat**

*Activity diagram update* data jemaat menampilkan gambaran proses mengenai bagaimana aktor memperbaharui data jemaat. Aktor dapat melakukan proses *update* dengan memilih menu jemaat yang kemudian sistem akan mengalihkan ke halaman data jemaat. Aktor memilih tombol edit yang kemudian memunculkan form data jemaat. Aktor kemudian mengisi form data jemaat dengan data baru, kemudian klik tombol simpan untuk menyimpan data jemaat yang baru ke dalam database. Penjelasan proses pada *activity diagram update* data jemaat ditunjukkan pada lampiran B.5

#### **4.3.4.5 Activity Diagram Delete Data Jemaat**

*Activity diagram delete* data jemaat menampilkan gambaran proses mengenai bagaimana aktor menghapus data jemaat. Aktor dapat melakukan proses *delete* dengan memilih menu jemaat yang kemudian sistem akan mengalihkan ke



halaman data jemaat. Aktor memilih tombol delete yang kemudian sistem akan menghapus data jemaat yang telah dipilih. Penjelasan proses pada *activity diagram delete* data jemaat ditunjukkan pada lampiran B.6

#### **4.3.4.6 Activity Diagram View Kriteria Majelis Jemaat**

*Activity diagram view* kriteria majelis jemaat menampilkan gambaran proses mengenai bagaimana aktor melihat kriteria majelis jemaat. Aktor dapat melakukan proses *view* dengan memilih menu kriteria yang kemudian sistem akan mengalihkan ke halaman kriteria. Penjelasan proses pada *activity diagram view* kriteria majelis jemaat ditunjukkan pada lampiran B.7

#### **4.3.4.7 Activity Diagram Update Kriteria Majelis Jemaat**

*Activity diagram update* kriteria majelis jemaat menampilkan gambaran proses mengenai bagaimana aktor memperbaharui kriteria majelis jemaat. Aktor dapat melakukan proses *update* dengan memilih menu kriteria yang kemudian sistem akan mengalihkan ke halaman kriteria. Aktor memilih tombol edit yang kemudian memunculkan form kriteria. Aktor kemudian mengisi form kriteria dengan data baru, kemudian klik tombol simpan untuk menyimpan kriteria yang baru ke dalam database. Penjelasan proses pada *activity diagram update* kriteria majelis jemaat ditunjukkan pada lampiran B.8

#### **4.3.4.8 Activity Diagram View Sub Kriteria Majelis Jemaat**

*Activity diagram view* sub kriteria majelis jemaat menampilkan gambaran proses mengenai bagaimana aktor melihat sub kriteria majelis jemaat. Aktor dapat melakukan proses *view* dengan memilih menu sub kriteria yang kemudian sistem akan mengalihkan ke halaman sub kriteria. Penjelasan proses pada *activity diagram view* sub kriteria majelis jemaat ditunjukkan pada lampiran B.9

#### **4.3.4.9 Activity Diagram Update Sub Kriteria Majelis Jemaat**

*Activity diagram update* sub kriteria majelis jemaat menampilkan gambaran proses mengenai bagaimana aktor memperbaharui sub kriteria majelis jemaat. Aktor dapat melakukan proses *update* dengan memilih menu sub kriteria yang kemudian sistem akan mengalihkan ke halaman sub kriteria. Aktor memilih tombol edit yang kemudian memunculkan form sub kriteria. Aktor kemudian mengisi form sub kriteria dengan data baru, kemudian klik tombol simpan untuk menyimpan sub kriteria yang baru ke dalam database. Penjelasan proses pada *activity diagram update* sub kriteria majelis jemaat ditunjukkan pada lampiran B.10

#### **4.3.4.10 Activity Diagram View Rangking**

*Activity diagram view rangking* menampilkan gambaran proses mengenai bagaimana aktor melihat rangking akhir dari pemilihan Majelis Jemaat seperti yang ditunjukkan pada lampiran B. Aktor dapat melakukan proses *view* dengan memilih menu rangking yang kemudian sistem akan mengalihkan ke halaman rangking. Penjelasan proses pada *activity diagram update* sub kriteria majelis jemaat ditunjukkan pada lampiran B.11

#### **4.3.5 Sequence Diagram**

*Sequence diagram* merupakan diagram yang menggambarkan interaksi antar kelas yang dilakukan sesuai dengan alur masing-masing. *Sequence diagram* merupakan desain pengembangan yang mendekati proses pengkodean, maka dari itu di dalam *sequence diagram* bahasa yang dituliskan merupakan bahasa pengkodean, dimana bahasa pengkodean tersebut menjelaskan jalannya alur dalam fitur secara berurutan. *Sequence diagram* dalam sistem informasi ini antara lain sebagai berikut.

#### **4.3.5.1 Sequence Diagram Login**

*Sequence diagram Login* merupakan diagram yang menggambarkan alur *user* dan *admin* untuk melakukan *login* ke dalam sistem pendukung keputusan pemilihan Majelis Jemaat dengan memasukkan *username* dan *password* yang sudah ada, kemudian sistem akan melakukan autentikasi dengan database. Aktor dapat mengakses sistem jika data yang dimasukkan dinyatakan valid oleh sistem. Interaksi antar kelas dan *function* yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.1

#### **4.3.5.2 Sequence Diagram View Data Jemaat**

*Sequence diagram view data jemaat* merupakan diagram yang menggambarkan alur untuk melihat data jemaat dengan klik menu data jemaat. Sistem akan menjalankan *function* yang berfungsi untuk mengalihkan aktor ke halaman jemaat sehingga aktor bisa melihat data jemaat yang tersedia. Interaksi antar kelas dan *function* yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.3

#### **4.3.5.3 Sequence Diagram Insert Data Jemaat**

*Sequence diagram insert data jemaat* merupakan diagram yang menggambarkan alur untuk menambahkan data jemaat dengan klik tambah jemaat pada halaman jemaat. Sistem akan menjalankan *function* yang berfungsi untuk memunculkan form untuk menambahkan data jemaat baru, kemudian aktor mengisi data jemaat baru. Data jemaat yang baru kemudian akan disimpan ke dalam database dengan klik simpan, sistem akan menyimpan data jemaat yang baru ke database. Aktor akan kembali ke halaman jemaat dan data jemaat baru akan muncul pada halaman jemaat. Interaksi antar kelas dan *function* yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.4

#### **4.3.5.4 Sequence Diagram Update Data Jemaat**

*Sequence diagram update* data jemaat merupakan diagram yang menggambarkan alur untuk memperbaharui data jemaat dengan klik tombol edit pada table di halaman jemaat. Sistem akan menjalankan *function* yang berfungsi untuk memunculkan form untuk memperbaharui data jemaat baru, kemudian aktor melakukan pembaharuan pada data jemaat yang telah dipilih. Data jemaat yang telah diperbaharui akan disimpan ke dalam database dengan klik simpan, sistem akan menyimpan data jemaat yang sudah diperbaharui ke dalam database. Aktor akan kembali ke halaman jemaat dan data jemaat yang sudah diperbaharui akan muncul pada halaman jemaat. Interaksi antar kelas dan *function* yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.5

#### **4.3.5.5 Sequence Diagram Delete Data Jemaat**

*Sequence diagram delete* data jemaat merupakan diagram yang menggambarkan alur untuk menghapus data jemaat dengan klik tombol delete. Sistem akan menjalankan *function* yang berfungsi untuk menghapus data jemaat yang dipilih. Data jemaat akan dihapus jika aktor mengklik tombol “OK” dan sistem akan secara otomatis menghapus data jemaat pada database. Aktor akan kembali ke halaman jemaat dan data jemaat yang sudah dihapus akan hilang di halaman jemaat. Interaksi antar kelas dan *function* yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.6

#### **4.3.5.6 Sequence Diagram View Kriteria Majelis Jemaat**

*Sequence diagram view* bobot kriteria merupakan diagram yang menggambarkan alur untuk menampilkan data kriteria majelis jemaat dengan klik menu kriteria. Sistem akan menjalankan *function* yang berfungsi untuk

mengalihkan aktor ke halaman kriteria majelis jemaat sehingga aktor bisa melihat kriteria majelis jemaat yang ada. Interaksi antar kelas dan *function* yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.7

#### **4.3.5.7 Sequence Diagram Update Kriteria Majelis Jemaat**

*Sequence diagram update* bobot kriteria merupakan diagram yang menggambarkan alur untuk memperbaharui data kriteria majelis jemaat dengan klik tombol. Sistem akan menjalankan *function* yang berfungsi untuk memunculkan form untuk memperbaharui kriteria majelis jemaat baru, kemudian aktor melakukan pembaharuan pada kriteria majelis jemaat yang telah dipilih. Kriteria majelis jemaat yang telah diperbaharui akan disimpan ke dalam database dengan klik simpan, sistem akan menyimpan kriteria jemaat yang sudah diperbaharui ke dalam database. Aktor akan kembali ke halaman kriteria dan kriteria jemaat yang sudah diperbaharui akan muncul pada halaman kriteria. Interaksi antar kelas dan *function* yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.8

#### **4.3.5.8 Sequence Diagram View Sub Kriteria Majelis Jemaat**

*Sequence diagram view* bobot sub kriteria merupakan diagram yang menggambarkan alur untuk menampilkan data sub kriteria majelis jemaat dengan klik menu sub kriteria. Sistem akan menjalankan *function* yang berfungsi untuk mengalihkan aktor ke halaman sub kriteria majelis jemaat sehingga aktor bisa melihat sub kriteria majelis jemaat yang ada. Interaksi antar kelas dan *function* yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.9

#### **4.3.5.9 Sequence Diagram Update Sub Kriteria Majelis Jemaat**

*Sequence diagram update* bobot sub kriteria merupakan diagram yang menggambarkan alur untuk memperbaharui data sub kriteria majelis jemaat dengan

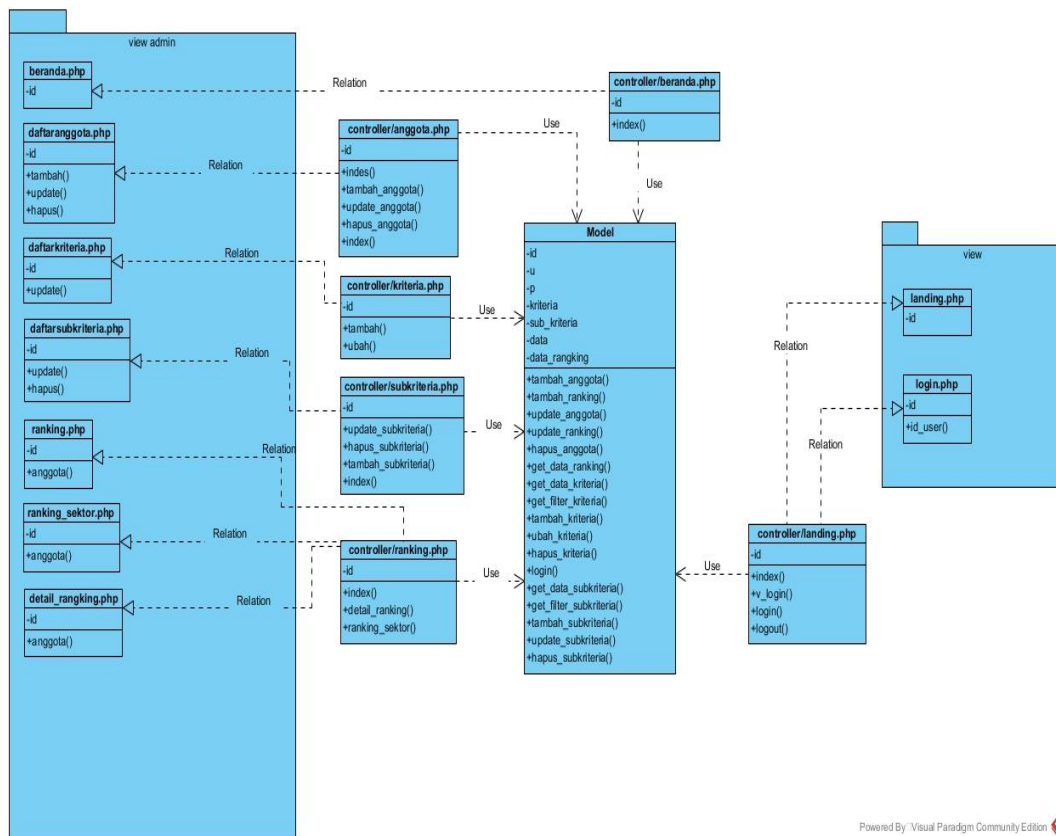
klik tombol. Sistem akan menjalankan *function* yang berfungsi untuk memunculkan form untuk memperbaharui sub kriteria majelis jemaat baru, kemudian aktor melakukan pembaharuan pada sub kriteria majelis jemaat yang telah dipilih. Sub Kriteria majelis jemaat yang telah diperbaharui akan disimpan ke dalam database dengan klik simpan, sistem akan menyimpan sub kriteria jemaat yang sudah diperbaharui ke dalam database. Aktor akan kembali ke halaman sub kriteria dan sub kriteria jemaat yang sudah diperbaharui akan muncul pada halaman sub kriteria. Interaksi antar kelas dan *function* yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.10

#### **4.3.5.10 Sequence Diagram View Rangking**

*Sequence diagram view ranking* merupakan diagram yang menggambarkan alur untuk menampilkan urutan nilai berdasarkan penilaian terhadap setiap kriteria dan sub kriteria yang sudah ditentukan dengan klik menu rangking. Sistem akan menjalankan *function* yang berfungsi untuk mengalihkan aktor ke halaman rangking sehingga aktor bisa melihat rangkin pemilihan majelis jemaat. Interaksi antar kelas dan *function* yang digunakan dalam proses pengkodean selanjutnya dapat dilihat pada lampiran C.11

#### **4.3.6 Class Diagram**

*Class diagram* merupakan diagram untuk menampilkan kelas-kelas maupun paket-paket yang ada pada suatu sistem informasi yang nantinya digunakan. Dalam paradigma OOP (*Object Oriented Programming*) terdapat 3 jenis kelas yaitu *model*, *view* dan *controller*. *Class diagram* yang digunakan Sistem Pendukung Keputusan Pemilihan Majelis Jemaat Gereja Protestan di Indonesia Bagian Barat "GETSEMANI" Jember Menggunakan Metode *Simple Multi Attribute Rating Technique* (SMART) dapat dilihat pada Gambar 4.3.

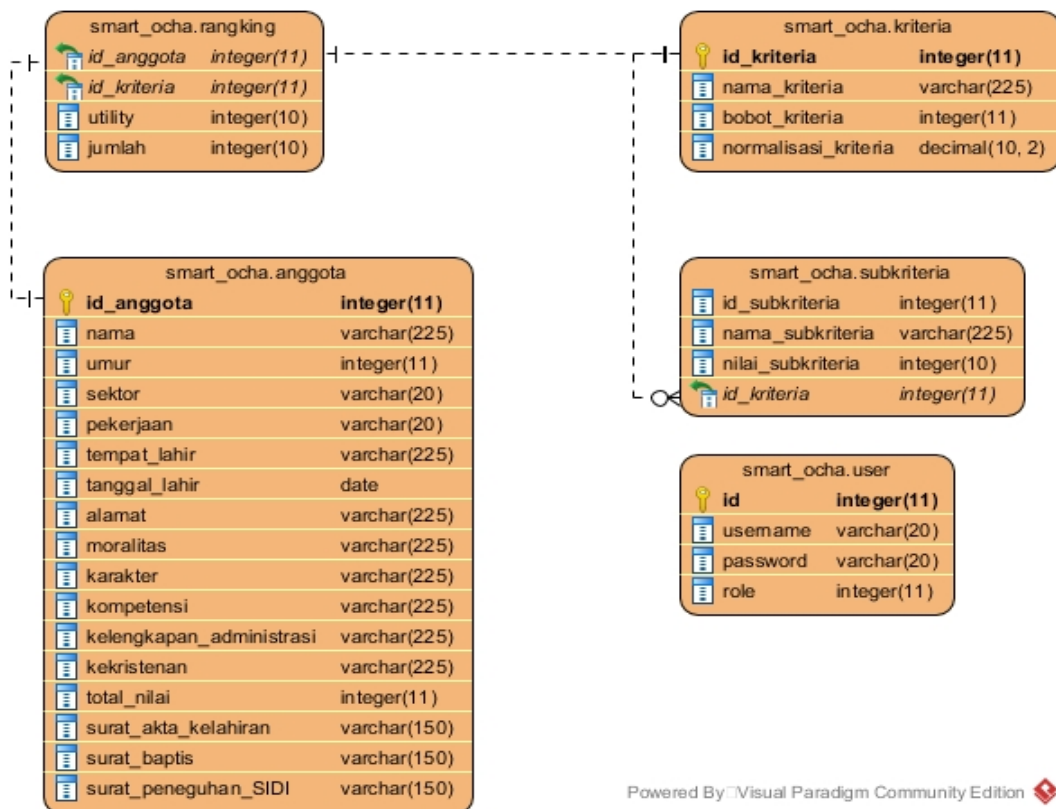


**Gambar 4. 3 Class Diagram**

#### 4.3.7 Entity Relationship Diagrams (ERD)

*Entity Relationship Diagram* (ERD) merupakan gambaran hubungan antar data dalam basis data yang memiliki relasi. ERD yang Sistem Pendukung Keputusan

Pemilihan Majelis Jemaat Gereja Protestan di Indonesia Bagian Barat “GETSEMANI” Jember Menggunakan Metode *Simple Multi Attribute Rating Technique* (SMART) dapat dilihat pada Gambar 4.4



Gambar 4.4 Entity Relationship Diagrams (ERD)

## 4.4 Penulisan Implementasi Kode Program

### 4.4.1 Kode Program tambah\_anggota()

Pada *Models* model terdapat *method* `tambah_anggota()` yang berfungsi untuk menambahkan data jemaat yang secara otomatis melakukan perhitungan mempunyai kriteria dan sub kriteria yang berbeda-beda. *Method* tersebut dapat dilihat pada gambar 4.5.

```
public function tambah_anggota(){

    //konfigurasi upload file

    $nama = $this->input->post('nama');
```



```

$umur = $this->input->post('umur');

$pekerjaan = $this->input->post('pekerjaan');

$tempat_lahir = $this->input->post('tempat_lahir');

$tanggal_lahir = $this->input->post('tanggal_lahir');

$surat_akta = $this->input->post('surat_akta');

$surat_baptis = $this->input->post('surat_baptis');

$surat_peneguhan = $this->input->post('surat_peneguhan');

$alamat = $this->input->post('alamat');

$alamat = $this->input->post('alamat');

$alamat = $this->input->post('alamat');

$alamat = $this->input->post('alamat');

$sektor = $this->input->post('sektor');

//-----ambil bobot dan normalisasi masing-
masing kriteria-----

//Kriteria Moralitas yang tinggi

$normalisasi_moralitas = $this->M_kriteria->get_filter_kriteria(1)->row()-
>normalisasi_kriteria;

$utility_moralitas = $this->M_subkriteria->get_filter_subkriteria($this-
>input->post('kriteria_1'))->row()->nilai_subkriteria;

```

```
$jumlah_moralitas = $normalisasi_moralitas * $utility_moralitas;
```

#### **//Kriteria Karakter yang baik**

```
$normalisasi_karakter = $this->M_kriteria->get_filter_kriteria(2)->row()->normalisasi_kriteria;
```

```
$utility_karakter = $this->M_subkriteria->get_filter_subkriteria($this->input->post('kriteria_2'))->row()->nilai_subkriteria;
```

```
$jumlah_karakter = $normalisasi_karakter * $utility_karakter;
```

#### **//Kriteria Kompetensi yang baik**

```
$normalisasi_kompetensi = $this->M_kriteria->get_filter_kriteria(3)->row()->normalisasi_kriteria;
```

```
$utility_kompetensi = $this->M_subkriteria->get_filter_subkriteria($this->input->post('kriteria_3'))->row()->nilai_subkriteria;
```

```
$jumlah_kompetensi = $normalisasi_kompetensi * $utility_kompetensi;
```

#### **//Kriteria Kelengkapan administrasi**

```
$normalisasi_administrasi = $this->M_kriteria->get_filter_kriteria(4)->row()->normalisasi_kriteria;
```

```
$utility_administrasi = $this->M_subkriteria->get_filter_subkriteria($this->input->post('kriteria_4'))->row()->nilai_subkriteria;
```

```
$jumlah_administrasi = $normalisasi_administrasi * $utility_administrasi;
```

### **//Kriteria Kekristenan**

```
$normalisasi_kekristenan = $this->M_kriteria->get_filter_kriteria(5)->row()->normalisasi_kriteria;
```

```
$utility_kekristenan = $this->M_subkriteria->get_filter_subkriteria($this->input->post('kriteria_5'))->row()->nilai_subkriteria;
```

```
$jumlah_kekristenan = $normalisasi_kekristenan * $utility_kekristenan;
```

### **//Hitung Total Nilai**

```
$total_nilai = $jumlah_moralitas + $jumlah_karakter + $jumlah_kompetensi + $jumlah_administrasi + $jumlah_kekristenan;
```

```
if ($upload != NULL) { //jika ada file yang diupload
```

```
    $surat_akta_kelahiran = $data_upload['file_name'];
```

```
}else{
```

```
    $surat_akta_kelahiran = NULL;
```

```
}
```

```
if ($upload1 != NULL) { //jika ada file yang diupload
```

```
$surat_baptis = $data_upload1['file_name'];

}else{

    $surat_baptis = NULL;

}

if ($upload2 != NULL) { //jika ada file yang diupload

    $surat_peneguhan_SIDI = $data_upload2['file_name'];

}else{

    $surat_peneguhan_SIDI = NULL;

}

$data = array(

    'nama' => $nama,

    'umur' => $umur,

    'pekerjaan' => $pekerjaan,

    'tempat_lahir' => $tempat_lahir,

    'tanggal_lahir' => $tanggal_lahir,

    'alamat' => $alamat,

    'sektor' => $sektor,
```

```

        'moralitas' => $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_1'))->row()->id_subkriteria,

        'karakter' => $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_2'))->row()->id_subkriteria,

        'kompetensi' => $this->M_subkriteria->get_filter_subkriteria($this-
>input->post('kriteria_3'))->row()->id_subkriteria,

        'kelengkapan_administrasi' => $this->M_subkriteria-
>get_filter_subkriteria($this->input->post('kriteria_4'))->row()->id_subkriteria,

        'kekristenan' => $this->M_subkriteria->get_filter_subkriteria($this-
>input->post('kriteria_5'))->row()->id_subkriteria,

        'total_nilai' => $total_nilai,

        'surat_akta_kelahiran' => $surat_akta_kelahiran,

        'surat_baptis' => $surat_baptis,

        'surat_peneguhan_SIDI' => $surat_peneguhan_SIDI,

    );

//simpan data anggota

    $this->M_anggota->tambah_anggota($data);

    $id_anggota = $this->db->insert_id();

//simpan detail perhitungan skor ke tabel ranking

```

```
$jumlah[1] = $jumlah_moralitas;

$jumlah[2] = $jumlah_karakter;

$jumlah[3] = $jumlah_kompetensi;

$jumlah[4] = $jumlah_administrasi;

$jumlah[5] = $jumlah_kekristenan;

$utility[1] = $utility_moralitas;

$utility[2] = $utility_karakter;

$utility[3] = $utility_kompetensi;

$utility[4] = $utility_administrasi;

$utility[5] = $utility_kekristenan;

$kriteria = $this->M_kriteria->get_data_kriteria()->result();

$i = 1;

foreach ($kriteria as $kr){

    $data_ranking = array(

        'id_anggota' => $id_anggota,

        'id_kriteria' => $kr->id_kriteria,

        'utility' => $utility[$i],
```

```

        'jumlah' => $jumlah[$i]

    );

    $this->M_anggota->tambah_ranking($data_ranking);

    $i++;

}

$this->session->set_flashdata('success', 'Data Berhasil Ditambah.');
```

```

    redirect('anggota');

}

```

**Gambar 4. 5 Kode Program tambah\_anggota()**

#### 4.4.2 Kode Program update\_anggota()

Pada *Models* model terdapat *method* `update_anggota()` yang berfungsi untuk mengedit data jemaat yang secara otomatis melakukan perhitungan mempunyai kriteria dan sub kriteria yang berbeda-beda. *Method* tersebut dapat dilihat pada gambar 4.6.

```

public function update_anggota(){

    //konfigurasi upload file

    $config['upload_path']    = './upload/';

    $config['allowed_types']  = 'jpg|png|jpeg';

    $config['overwrite']      = true;

```

```
$this->load->library('upload', $config);

$upload = $this->upload->do_upload('surat_akta');

$data_upload = $this->upload->data();

$upload1 = $this->upload->do_upload('surat_baptis');

$data_upload1 = $this->upload->data();

$upload2 = $this->upload->do_upload('surat_peneguhan');

$data_upload2 = $this->upload->data();

$id_anggota = $this->input->post('id_anggota');

$nama = $this->input->post('nama');

$umur = $this->input->post('umur');

$pekerjaan = $this->input->post('pekerjaan');

$tempat_lahir = $this->input->post('tempat_lahir');

$tanggal_lahir = $this->input->post('tanggal_lahir');

$salamat = $this->input->post('alamat');

$sektor = $this->input->post('sektor');
```



**//-----ambil bobot dan normalisasi masing-masing kriteria-----**

**//Kriteria Moralitas yang tinggi**

\$normalisasi\_moralitas = \$this->M\_kriteria->get\_filter\_kriteria(1)->row()->normalisasi\_kriteria;

\$utility\_moralitas = \$this->M\_subkriteria->get\_filter\_subkriteria(\$this->input->post('kriteria\_1'))->row()->nilai\_subkriteria;

\$jumlah\_moralitas = \$normalisasi\_moralitas \* \$utility\_moralitas;

**//Kriteria Karakter yang baik**

\$normalisasi\_karakter = \$this->M\_kriteria->get\_filter\_kriteria(2)->row()->normalisasi\_kriteria;

\$utility\_karakter = \$this->M\_subkriteria->get\_filter\_subkriteria(\$this->input->post('kriteria\_2'))->row()->nilai\_subkriteria;

\$jumlah\_karakter = \$normalisasi\_karakter \* \$utility\_karakter;

**//Kriteria Kompetensi yang baik**

\$normalisasi\_kompetensi = \$this->M\_kriteria->get\_filter\_kriteria(3)->row()->normalisasi\_kriteria;

```

    $utility_kompetensi          =          $this->M_subkriteria-
>get_filter_subkriteria($this->input->post('kriteria_3'))->row()-
>nilai_subkriteria;

```

```

    $jumlah_kompetensi          =          $normalisasi_kompetensi      *
$utility_kompetensi;

```

### **//Kriteria Kelengkapan administrasi**

```

    $normalisasi_administrasi = $this->M_kriteria->get_filter_kriteria(4)-
>row()->normalisasi_kriteria;

```

```

    $utility_administrasi      =          $this->M_subkriteria-
>get_filter_subkriteria($this->input->post('kriteria_4'))->row()-
>nilai_subkriteria;

```

```

    $jumlah_administrasi      =          $normalisasi_administrasi      *
$utility_administrasi;

```

### **//Kriteria Kekristenan**

```

    $normalisasi_kekristenan = $this->M_kriteria->get_filter_kriteria(5)-
>row()->normalisasi_kriteria;

```

```

    $utility_kekristenan      =          $this->M_subkriteria-
>get_filter_subkriteria($this->input->post('kriteria_5'))->row()-
>nilai_subkriteria;

```

```

    $jumlah_kekristenan      =          $normalisasi_kekristenan      *
$utility_kekristenan;

```

**//Hitung Total Nilai**

```
$total_nilai = $jumlah_moralitas + $jumlah_karakter +  
$jumlah_kompetensi + $jumlah_administrasi + $jumlah_kekristenan;
```

```
if ($upload != NULL) { //jika ada file yang diupload
```

```
    $surat_akta_kelahiran = $data_upload['file_name'];
```

```
}else{
```

```
    $surat_akta_kelahiran = $this->input->post('surat_akta_lama');
```

```
}
```

```
if ($upload1 != NULL) { //jika ada file yang diupload
```

```
    $surat_baptis = $data_upload1['file_name'];
```

```
}else{
```

```
    $surat_baptis = $this->input->post('surat_baptis_lama');
```

```
}
```

```
if ($upload2 != NULL) { //jika ada file yang diupload
```

```
    $surat_peneguhan_SIDI = $data_upload2['file_name'];
```

```
}else{
```

```
        $surat_peneguhan_SIDI          =          $this->input-
>post('surat_peneguhan_lama');

    }

    $data = array(

        'nama' => $nama,

        'umur' => $umur,

        'pekerjaan' => $pekerjaan,

        'tempat_lahir' => $tempat_lahir,

        'tanggal_lahir' => $tanggal_lahir,

        'alamat' => $alamat,

        'sektor' => $sektor,

        'moralitas' => $this->M_subkriteria->get_filter_subkriteria($this-
>input->post('kriteria_1'))->row()->id_subkriteria,

        'karakter' => $this->M_subkriteria->get_filter_subkriteria($this-
>input->post('kriteria_2'))->row()->id_subkriteria,

        'kompetensi' => $this->M_subkriteria->get_filter_subkriteria($this-
>input->post('kriteria_3'))->row()->id_subkriteria,

        'kelengkapan_administrasi'     =>          $this->M_subkriteria-
>get_filter_subkriteria($this->input->post('kriteria_4'))->row()-
>id_subkriteria,
```

```
'kekristenan' => $this->M_subkriteria->get_filter_subkriteria($this->input->post('kriteria_5'))->row()->id_subkriteria,  
  
'total_nilai' => $total_nilai,  
  
'surat_akta_kelahiran' => $surat_akta_kelahiran,  
  
'surat_baptis' => $surat_baptis,  
  
'surat_peneguhan_SIDI' => $surat_peneguhan_SIDI,  
  
);
```

**//simpan data anggota**

```
$status1 = $this->M_anggota->update_anggota($id_anggota, $data);
```

**//simpan detail perhitungan skor ke tabel ranking**

```
$jumlah[1] = $jumlah_moralitas;
```

```
$jumlah[2] = $jumlah_karakter;
```

```
$jumlah[3] = $jumlah_kompetensi;
```

```
$jumlah[4] = $jumlah_administrasi;
```

```
$jumlah[5] = $jumlah_kekristenan;
```

```
$utility[1] = $utility_moralitas;
```

```
$utility[2] = $utility_karakter;
```

```
$utility[3] = $utility_kompetensi;

$utility[4] = $utility_administrasi;

$utility[5] = $utility_kekristenan;

$ kriteria = $this->M_kriteria->get_data_kriteria()->result();

$i = 1;

foreach ($kriteria as $kr){

    $data_ranking = array(

        'utility' => $utility[$i],

        'jumlah' => $jumlah[$i]

    );

    $status = $this->M_anggota->update_ranking($id_anggota, $kr->id_kriteria, $data_ranking);

    $i++;

}
```

**Gambar 4.6 Kode Program update\_anggota()**

#### **4.5 Pengujian Sistem Informasi**

Pengujian *black box* berfungsi untuk menguji sistem informasi dari sisi fungsionalitas sistem informasi dengan tujuan mengetahui apakah fitur-fitur, inputan dan keluaran sistem informasi sesuai dengan kebutuhan pengguna atau tidak. Fitur-fitur yang diujikan mencakup fitur melihat, menambah, mengubah

dan menghapus data serta keluaran yang diharapkan dari sistem informasi saat fitur tersebut dijalankan. Hasil pengujian *black box* pada fitur hasil peramalan dapat dilihat pada Tabel 4.3.

**Tabel 4.3 Hasil Pengujian Black Box**

No.	Fitur	Aksi	Hasil	Keterangan
1.	Login	Klik tombol home (pojok kanan atas) kemudian pilih <i>login</i>	Menampilkan form <i>login</i> yang berisi <i>username</i> dan <i>password</i>	Berhasil
		Mengisi form <i>login</i> kemudian klik tombol <i>sign in</i>	Menampilkan halaman utama admin	Berhasil
		Klik tombol Masuk namun <i>username</i> atau <i>password</i> tidak sesuai	Menampilkan pesan " <i>This field is required</i> "	Berhasil
2.	Data Jemaat	Klik menu data Jemaat	Menampilkan halaman awal data laptop	Berhasil
		Mengisi text file "cari"	Mencari data yang dimasukkan	Berhasil
		Klik tambah jemaat	Menampilkan form data laptop	

Dilanjutkan

Lanjutkan

		Mengisi form data jemaat kemudian klik simpan data	Data berhasil tersimpan ke database kemudian kembali ke halaman awal data jemaat	Berhasil
		Klik edit data	Menampilkan form data jemaat	Berhasil
		Mengedit data jemaat yang sudah ada pada form dan klik simpan	Data berhasil diedit serta tersimpan ke database kemudian kembali ke halaman awal data jemaat	Berhasil
		Klik hapus data	Menampilkan pesan “yakin akan menghapus?”	Berhasil
		Klik OK	Data berhasil dihapus kemudian kembali ke halaman awal data laptop	Berhasil
		Klik cancel	Kembali ke halaman awal data laptop	Berhasil

Dilanjutkan



Lanjutkan

		Klik next/previous	Mengganti page pada data laptop	Berhasil
3.	Kriteria	Klik menu kriteria	Menampilkan halaman awal data kriteria	Berhasil
		Mengisi text file "cari"	Mencari data yang dimasukkan	Berhasil
		Klik edit data	Menampilkan form edit data kriteria	Berhasil
		Mengedit data kriteria yang sudah ada pada form kemudian klik simpan data	Data berhasil diedit serta tersimpan ke databas kemudian kembali k halaman awal data kriteria	Berhasil
		Klik next/previous	Mengganti page pada data laptop	Berhasil
4.	Sub Kriteria	Klik menu sub kriteria	Menampilkan halaman awal data sub kriteria	Berhasil
		Mengisi text fike "cari"	Mencari data yang dimasukkan	Berhasil

Dilanjutkan

Lanjutan

		Klik tambah kriteria	Menampilkan form tambah data sub kriteria	Berhasil
		Mengisi form tambah kriteria sub kriteria kemudian klik simpan data	Data berhasil tersimpan ke database kemudian kembali ke halaman awal data sub kriteria	Berhasil
		Klik edit data	Menampilkan form edit data kriteria	Berhasil
		Mengedit data sub kriteria yang sudah ada pada form kemudian klik simpan data	Data berhasil diedit serta tersimpan ke databas kemudian kembali k halaman awal data kriteria	Berhasil
		Klik hapus data	Menampilkan pesan “yakin akan menghapus?”	Berhasil
		Klik OK	Data berhasil dihapus kemudian kembali ke halaman awal data laptop	Berhasil

Dilanjutkan

## Lanjutan

		Klik cancel	Kembali ke halaman awal data laptop	Berhasil
		Klik next / previous	Mengganti page pada data laptop	Berhasil
5.	Rangking	Memilih dropdown “pilih salah satu sector”	menampilkan pilihan sektor	Berhasil
		Klik Rangking per Sektor pada sector yang sudah dipilih sebelumnya	Memunculkan halaman majelis jemaat terpilih sesuai sektornya	Berhasil
		Klik Kembali	Menampilkan halaman awal rangking	Berhasil
		Klik seacrh	Menampilkan detail setiap data jemaat	Berhasil
6.	Logout	Klik keluar	Menampilkan halaman home	Berhasil

## BAB 6. PENUTUP

### 6.1 Kesimpulan

Kesimpulan yang bias didapat dari penelitian ini adalah sebagai berikut :

1. Metode *Simple Multi Atribut Rating Technique* (SMART) diterapkan dalam beberapa hal yang mempengaruhi hasil perhitungan yakni bobot kriteria dan nilai sub kriteria yang dimasukkan *admin* sesuai kebutuhan. Semakin besar nilai yang dimasukkan, maka nilai akhir yang diperoleh akan semakin besar. Semakin besar nilai akhir yang dimiliki, maka semakin bagus juga kriteria jemaat tersebut untuk menjadi majelis jemaat. Penerapan metode *Simple Multi Atribut Rating Technique* (SMART) pada sistem pendukung keputusan pemilihan Majelis Jemaat GPIB “GETSEMANI” Jember, guna membantu jemaat GPIB “GETSEMANI” Jember memilih Majelis Jemaat yang sesuai kebutuhan secara objektif dan efisien.
2. Perancangan dan pembangunan sistem pendukung keputusan pemilihan Majelis Jemaat GPIB “GETSEMANI” Jember menggunakan metode *Simple Multi Atribut Rating Technique* menggunakan SDLC model *waterfall*. Proses yang terjadi dalam pembangunan sistem menggunakan model *waterfall* adalah, model ini mengharuskan semua tahap dalam pembangunan sistem dilakukan secara urut dari tahap analisis kebutuhan, desain, implementasi, pengujian hingga pemeliharaan sistem. Setiap tahap yang ada harus dilakukan dengan benar dan direncanakan secara matang, karena jika terdapat tahap yang salah atau tidak sesuai maka akan menimbulkan kesalahan pada tahap-tahap selanjutnya, hambatan yang dialami oleh peneliti dalam melakukan penelitian adalah ketika telah sampai pada tahap implementasi terdapat kesalahan atau perbaikan dari tester pada tahap desain, sehingga peneliti harus mengulang dari tahap awal yakni dari tahap analisis data hingga ke tahap desain sehingga membutuhkan waktu yang lama dalam proses pengerjaannya. Namun karena sistem pendukung keputusan pemilihan

Majelis sJemaat GPIB “GETSEMANI” Jember tergolong sistem yang berskala kecil, maka hambatan tersebut tidak begitu menjadi masalah besar.

## 6.2 Saran

Beberapa saran yang ditujukan untuk memberikan masukan yang lebih baik pada penelitian selanjutnya yaitu:

1. Pengembangan sistem pendukung keputusan pemilihan Majelis Jemaat GPIB “GETSEMANI” Jember yang akan dikembangkan selanjutnya diharapkan jemaat dapat menambahkan atau mengubah datanya dirinya sendiri beserta mengisi nilai sub kriteria sebagai jemaat yang nantinya bisa dipilih menjadi majelis jemaat tanpa merubah kode pemrograman. Di karenakan kritria dan sub kriteria yang dipakai dalam sistem ini sangat minim, akhirnya dalam perhitungan hasil akhir rekomendasi nilai yang dihasilkan mempunyai selisih yang hampir mirip. Disamping itu penambahan kriteria dan sub kriteria juga dapat menambahkan opsi pilihan jemaat dalam melakukan pemilihan.
2. Sistem pendukung keputusan pemilihan Majelis Jemaat GPIB “GETSEMANI” Jember yang akan dikembangkan selanjutnya diharapkan terdapat fitur pemilihan *online* agar memudahkan jemaat yang Namanya terdaftar sebagai warga jemaat GPIB “GETSEMANI” Jember namun bertempat tinggal di luar kota untuk sementara dapat secara langsung memilih jemaat yang direkomendasikan untuk menjadi majelis jemaat.

## DAFTAR PUSTAKA

- Fauziah. (2015). Sistem Pendukung Keputusan Penentuan Pemberian RASKIN untuk Masyarakat Menggunakan Metode SMART.
- Novianti, D. (2016). Sistem Pendukung Keputusan Berbasis Web Untuk Pemilihan Cafe Menggunakan Metode Smart (Simple Multi Attribute Rating Technique).
- Pressman, R. S. (2001). *Software Engineering: A Practitioner's Approach, Fifth Ed.* New York: McGraw-Hill Book Company.
- Romeo. (2003). *Testing dan Implementasi Sistem.* Surabaya: STIKOM.
- Suryanto, Muhammad Safrizal. (2015). Sistem Pendukung Keputusan Pemilihan Karyawan Teladan dengan Metode SMART (Simple Multi Attribute Rating Technique). *Jurnal CoreIT*, 26-27.
- Utut, A. (2015). Sistem Penunjang Keputusan Pemilihan Majelis Gereja Kristus Tuhan Jember dengan Metode Fuzzy-Analytic Hierarchy Process (F-AHP).
- Yunitarin, R. (2013). Sistem Pendukung Keputusan Pemilihan Peniar Radio Terbaik Menggunakan Metode Simple Multi Attribute Rating Technique (SMART).

## LAMPIRAN

### LAMPIRAN A. SKENARIO

#### A.1 Skenario Login

<b>Nomor Usecase</b>	<b>USC-01</b>
<b>Nama</b>	<i>Login</i>
<b>Aktor</b>	<b>Admin</b>
<b>Deskripsi Singkat</b>	<b>Aktor membuka halaman <i>Login</i> untuk autentifikasi hak akses sistem informasi</b>
<b>Prekondisi</b>	<b>Aktor membuka halaman <i>Login</i></b>
<b>Prakondisi</b>	<b>Aktor berhasil masuk ke sistem informasi</b>
<b>Flow Events</b>	
<b>Skenario Normal : <i>Login</i></b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Membuka sistem informasi Pemilihan Majelis Jemaat pada <i>browser</i>	
	2. Menampilkan halaman utama yang berisi :  - <i>Login</i>

	<ul style="list-style-type: none"> <li>- <i>Home</i></li> <li>- <i>Contact</i></li> </ul>
3. Pilih menu <i>Login</i>	
	<p>4. Menampilkan halaman <i>Login</i> yang mempunyai form dengan atribut:</p> <ul style="list-style-type: none"> <li>- <i>Username</i></li> <li>- <i>Password</i></li> </ul>
5. Mengisi form <i>Login</i>	
6. Klik tombol <i>Sign In</i>	
	7. Menampilkan halaman Dashboard untuk hak akses Admin
<b>Skenario Alternatif : <i>Username</i> dan <i>password</i> tidak sesuai</b>	
8. Klik tombol <i>Sign In</i>	
	9. Menampilkan halaman utama

## A.2 Skenario View Data Jemaat

<b>Nomor Usecase</b>	<b>USC-02</b>
<b>Nama</b>	<b><i>View Data Jemaat</i></b>
<b>Aktor</b>	<b>Admin</b>
<b>Deskripsi Singkat</b>	<b>Aktor melihat data jemaat</b>
<b>Prekondisi</b>	<b>Aktor memilih menu data jemaat</b>



<b>Prakondisi</b>	<b>Aktor berhasil masuk halaman data jemaat</b>
<b>Flow Events</b>	
<b>Skenario Normal : <i>View Data Jemaat</i></b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Klik menu data jemaat	
	<p>2. Menampilkan halaman data jemaat yang berisi :</p> <ul style="list-style-type: none"> <li>• Tombol tambah data</li> <li>• <i>Field</i> cari</li> <li>• Tabel data jemaat yang berisi : <ul style="list-style-type: none"> <li>- ID</li> <li>- Nama</li> <li>- Umur</li> <li>- Pekerjaan</li> <li>- Sektor</li> <li>- TTL</li> <li>- Alamat</li> <li>- Surat Akta Lahir</li> <li>- Surat Baptis</li> <li>- Surat Penegunah SIDI</li> </ul> </li> </ul>

	- Aksi yang berisi tombol <i>Edit</i> dan <i>Delete</i>
--	---

### A.3 Skenario Insert Data Jemaat

<b>Nomor Usecase</b>	<b>USC-03</b>
<b>Nama</b>	<b><i>Insert Data Jemaat</i></b>
<b>Aktor</b>	<b>Admin</b>
<b>Deskripsi Singkat</b>	<b>Aktor menambahkan data jemaat</b>
<b>Prekondisi</b>	<b>Aktor memilih menu data jemaat</b>
<b>Prakondisi</b>	<b>Aktor berhasil menambahkan data jemaat</b>
<b>Flow Events</b>	
<b>Skenario Normal : <i>Insert Data Jemaat</i></b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Klik menu data jemaat	
	<p>2. Menampilkan halaman data jemaat yang berisi :</p> <ul style="list-style-type: none"> <li>• Tombol tambah data</li> <li>• <i>Field</i> cari</li> <li>• Tabel data jemaat yang berisi : <ul style="list-style-type: none"> <li>- ID</li> </ul> </li> </ul>

	<ul style="list-style-type: none"> <li>- Nama</li> <li>- Umur</li> <li>- Pekerjaan</li> <li>- Sektor</li> <li>- TTL</li> <li>- Alamat</li> <li>- Surat Akta Lahir</li> <li>- Surat Baptis</li> <li>- Surat Penegunah SIDI</li> <li>- Aksi yang berisi tombol <i>Edit dan Delete</i></li> </ul>
3. Klik Tombol tambah data	
	<p>3. Menampilkan form tambah data jemaat yang berisi :</p> <ul style="list-style-type: none"> <li>- Nama</li> <li>- Umur</li> <li>- Pekerjaan</li> <li>- Tempat Lahir</li> <li>- Tanggal Lahir</li> <li>- Alamat</li> <li>- Sektor</li> </ul>

	<ul style="list-style-type: none"> <li>- Kriteria Moralitas yang tinggi</li> <li>- Kriteria Karakter yang baik</li> <li>- Kriteria Kompetensi yang baik</li> <li>- Kriteria Kelengkapan administrasi</li> <li>- Kriteria Kekristenan</li> <li>- Surat Akta Kelahiran</li> <li>- Surat Baptis</li> <li>- Surat Peneguhan SIDI</li> <li>- Tombol simpan dan tutup</li> </ul>
4. Mengisi form tambah data jemaat, kemudian mengklik tombol simpan	
	5. Menyimpan data ke dalam <i>database</i>
	6. Menampilkan halaman stok laptop dengan data baru yang telah ditambahkan ke tabel.
<b>Skenario Alternatif : Form Kriteria tidak terisi</b>	
1. Klik tombol Simpan Data	
	2. Menampilkan spam “ <i>Please select an item in the list</i> ”

#### A.4 Skenario Update Data Jemaat

<b>Nomor Usecase</b>	<b>USC-04</b>
<b>Nama</b>	<b><i>Update Data Jemaat</i></b>
<b>Aktor</b>	<b>Admin</b>
<b>Deskripsi Singkat</b>	<b>Aktor memperbaharui data jemaat</b>
<b>Prekondisi</b>	<b>Aktor memilih menu data jemaat</b>
<b>Prakondisi</b>	<b>Aktor berhasil memperbaharui data jemaat</b>
<b>Flow Events</b>	
<b>Skenario Normal : <i>Update Data Jemaat</i></b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Klik menu data jemaat	
	<p>2. Menampilkan halaman data jemaat yang berisi :</p> <ul style="list-style-type: none"> <li>• Tombol tambah data</li> <li>• <i>Field</i> cari</li> <li>• Tabel data jemaat yang berisi : <ul style="list-style-type: none"> <li>- ID</li> <li>- Nama</li> <li>- Umur</li> <li>- Pekerjaan</li> </ul> </li> </ul>

	<ul style="list-style-type: none"> <li>- Sektor</li> <li>- TTL</li> <li>- Alamat</li> <li>- Surat Akta Lahir</li> <li>- Surat Baptis</li> <li>- Surat Penegunah SIDI</li> <li>- Aksi yang berisi tombol <i>Edit</i> dan <i>Delete</i></li> </ul>
3. Klik Tombol <i>Edit</i> data pada salah satu data yang akan diubah	
	4. Menampilkan halaman data jemaat dengan form tambah data stok laptop telah terisi data yang akan diubah.
5. Mengubah data pada form tambah data jemaat, kemudian klik tombol Simpan Data	
	6. Menyimpan perubahan data ke dalam <i>database</i>
	7. Menampilkan halaman data jemaat dengan data yang telah diubah
<b>Skenario Alternatif : Form Kriteria tidak terisi</b>	
1. Klik tombol Simpan Data	

	2. Menampilkan spam “ <i>Please select an item in the list</i> ”
--	--

#### A.5 Skenario *Delete Data Jemaat*

<b>Nomor Usecase</b>	<b>USC-05</b>
<b>Nama</b>	<b><i>Delete Data Jemaat</i></b>
<b>Aktor</b>	<b>Admin</b>
<b>Deskripsi Singkat</b>	<b>Aktor menghapus data jemaat</b>
<b>Prekondisi</b>	<b>Aktor memilih menu data jemaat</b>
<b>Prakondisi</b>	<b>Aktor berhasil menghapus data jemaat</b>
<b>Flow Events</b>	
<b>Skenario Normal : <i>Delete Data Jemaat</i></b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
3. Klik menu data jemaat	
	<p>4. Menampilkan halaman data jemaat yang berisi :</p> <ul style="list-style-type: none"> <li>• Tombol tambah data</li> <li>• <i>Field</i> cari</li> <li>• Tabel data jemaat yang berisi : <ul style="list-style-type: none"> <li>- ID</li> <li>- Nama</li> </ul> </li> </ul>

	<ul style="list-style-type: none"> <li>- Umur</li> <li>- Pekerjaan</li> <li>- Sektor</li> <li>- TTL</li> <li>- Alamat</li> <li>- Surat Akta Lahir</li> <li>- Surat Baptis</li> <li>- Surat Penegunah SIDI</li> <li>- Aksi yang berisi tombol <i>Edit</i> dan <i>Delete</i></li> </ul>
3. Klik Tombol <i>Delete</i>	
	4. Menampilkan spam “Apakah Anda yakin ingin menghapus data ini?”
5. Klik tombol “OK”	
	6. Menghapus data pada <i>database</i>
	7. Menampilkan halaman data hasil perhitungan data telah dihapus.
<b>1. Skenario Alternatif : Batal menghapus data stok laptop</b>	
1. Klik tombol <i>delete</i> salah satu data yang akan dihapus	



	2. Menampilkan spam “Apakah Anda yakin ingin menghapus data ini?”
5. Klik tombol <i>cancel</i>	
	6. Kembali ke halaman data jemaat

#### A.6 Skenario *View* Kriteria Majelis Jemaat

<b>Nomor Usecase</b>	<b>USC-06</b>
<b>Nama</b>	<b>Melihat Data Kriteria Majelis Jemaat</b>
<b>Aktor</b>	<b>Admin</b>
<b>Deskripsi Singkat</b>	<b>Aktor melihat data Kriteria Majelis Jemaat</b>
<b>Prekondisi</b>	<b>Aktor memilih menu Kriteria Majelis Jemaat</b>
<b>Prakondisi</b>	<b>Aktor berhasil melihat Kriteria Majelis Jemaat</b>
<b>Flow Events</b>	
<b>Skenario Normal : Melihat Data Kriteria Laptop</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Klik menu Kriteria Majelis Jemaat	
	2. Menampilkan halaman Kriteria Majelis Jemaat yang meliputi :

	<ul style="list-style-type: none"> <li>• <i>Field</i> cari</li> <li>• Tabel data Kriteria Majelis Jemaat yang berisi: <ul style="list-style-type: none"> <li>- ID</li> <li>- Nama kriteria</li> <li>- Bobot (%)</li> <li>- Normalisasi</li> <li>- Aksi berupa tombol <i>edit</i></li> </ul> </li> </ul>
--	---

#### A.7 Skenario Update Kriteria Majelis Jemaat

<b>Nomor Usecase</b>	<b>USC-07</b>
<b>Nama</b>	<b>Mengubah Data Kriteria Majelis Jemaat</b>
<b>Aktor</b>	<b>Admin</b>
<b>Deskripsi Singkat</b>	<b>Aktor mengubah data Kriteria Majelis Jemaat</b>
<b>Prekondisi</b>	<b>Aktor mengubah menu Kriteria Majelis Jemaat</b>
<b>Prakondisi</b>	<b>Aktor berhasil mengubah Kriteria Majelis Jemaat</b>
<b>Flow Events</b>	
<b>Skenario Normal : Mengubah Data Kriteria Laptop</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>

7. Klik menu Kriteria Majelis Jemaat	
	<p>8. Menampilkan halaman Kriteria Majelis Jemaat yang meliputi :</p> <ul style="list-style-type: none"> <li>• <i>Field</i> cari</li> <li>• Tabel data Kriteria Majelis Jemaat yang berisi: <ul style="list-style-type: none"> <li>- ID</li> <li>- Nama kriteria</li> <li>- Bobot (%)</li> <li>- Normalisasi</li> <li>- Aksi berupa tombol <i>edit</i></li> </ul> </li> </ul>
9. Klik tombol <i>edit</i> salah satu data yang akan diubah	
	<p>10. Menampilkan halaman data Kriteria Majelis Jemaat yang berisi :</p> <ul style="list-style-type: none"> <li>- Nama Kriteria</li> <li>- Bobot Kriteria</li> <li>- Tombol Simpan dan Tutup</li> </ul>
11. Mengubah data pada form tambah data Kriteria laptop, kemudian klik tombol Simpan	

	12. Menyimpan perubahan data ke dalam <i>database</i>
	13. Menampilkan halaman data Kriteria Majelis Jemaat dengan data yang telah diubah
<b>Skenario Alternatif : Isian form tidak sesuai</b>	
1. Klik tombol Simpan	
	2. Menampilkan span " <i>Please fill out this field</i> "

#### A.8 Skenario View Sub Kriteria Majelis Jemaat

<b>Nomor Usecase</b>	<b>USC-08</b>
<b>Nama</b>	<b>Melihat Data Sub Kriteria Majelis Jemaat</b>
<b>Aktor</b>	<b>Admin</b>
<b>Deskripsi Singkat</b>	<b>Aktor melihat Data Sub Kriteria Majelis Jemaat</b>
<b>Prekondisi</b>	<b>Aktor memilih menu Sub Kriteria Majelis Jemaat</b>
<b>Prakondisi</b>	<b>Aktor berhasil melihat Sub Kriteria Majelis Jemaat</b>
<b>Flow Events</b>	
<b>Skenario Normal : Melihat Data Kriteria Laptop</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>

14. Klik menu Sub Kriteria Majelis Jemaat	
	<p>15. Menampilkan halaman Sub Kriteria Majelis Jemaat yang meliputi :</p> <ul style="list-style-type: none"> <li>• <i>Field</i> cari</li> <li>• Tabel data sub Kriteria Majelis Jemaat yang berisi: <ul style="list-style-type: none"> <li>- ID</li> <li>- Nama kriteria</li> <li>- Nama Sub Kriteria</li> <li>- Nilai</li> </ul> </li> <li>- Aksi berupa tombol <i>edit</i> dan <i>delete</i></li> </ul>

#### A.9 Skenario *Update* Sub Kriteria Majelis Jemaat

<b>Nomor Usecase</b>	<b>USC-09</b>
<b>Nama</b>	<b>Mengubah Data Kriteria Majelis Jemaat</b>
<b>Aktor</b>	<b>Admin</b>
<b>Deskripsi Singkat</b>	<b>Aktor mengubah data Kriteria Majelis Jemaat</b>
<b>Prekondisi</b>	<b>Aktor mengubah menu Kriteria Majelis Jemaat</b>

<b>Prakondisi</b>	<b>Aktor berhasil mengubah Kriteria Majelis Jemaat</b>
<b>Flow Events</b>	
<b>Skenario Normal : Mengubah Data Kriteria Laptop</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
16. Klik menu Sub Kriteria Majelis Jemaat	
	<p>17. Menampilkan halaman Sub Kriteria Majelis Jemaat yang meliputi :</p> <ul style="list-style-type: none"> <li>• <i>Field</i> cari</li> <li>• Tabel data sub Kriteria Majelis Jemaat yang berisi: <ul style="list-style-type: none"> <li>- ID</li> <li>- Nama kriteria</li> <li>- Nama Sub Kriteria</li> <li>- Nilai</li> <li>- Aksi berupa tombol <i>edit</i> dan <i>delete</i></li> </ul> </li> </ul>
18. Klik tombol <i>edit</i> salah satu data yang akan diubah	
	19. Menampilkan halaman data Sub Kriteria Majelis Jemaat yang berisi :

	<ul style="list-style-type: none"> <li>- Nama Sub Kriteria</li> <li>- Nilai Sub Kriteria</li> <li>- ID Kriteria</li> <li>- Tombol Simpan dan Tutup</li> </ul>
20. Mengubah data pada form tambah data Sub Kriteria laptop, kemudian klik tombol Simpan	
	21. Menyimpan perubahan data ke dalam <i>database</i>
	22. Menampilkan halaman data Sub Kriteria Majelis Jemaat dengan data yang telah diubah
<b>Skenario Alternatif : Isian form tidak sesuai</b>	
1. Klik tombol Simpan	
	2. Menampilkan span " <i>Please fill out this field</i> "

#### A.10 Skenario Rangking

<b>Nomor Usecase</b>	<b>USC-010</b>
<b>Nama</b>	<b>Melihat Data Rangking</b>
<b>Aktor</b>	<b>Admin</b>
<b>Deskripsi Singkat</b>	<b>Aktor melihat data Rangking</b>
<b>Prekondisi</b>	<b>Aktor memilih menu Rangking</b>

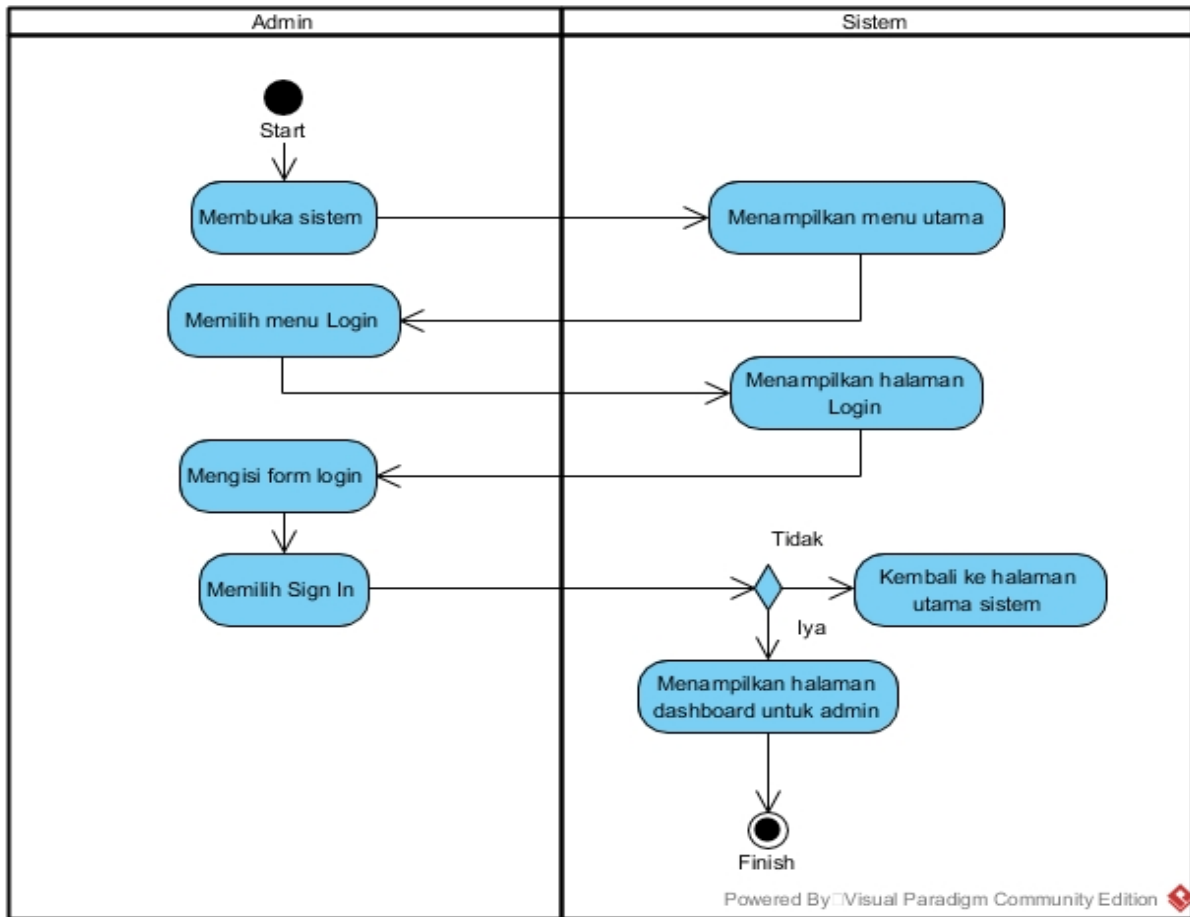
Prakondisi	Aktor berhasil melihat Rangking
<b>Flow Events</b>	
<b>Skenario Normal : Melihat Data Rangking</b>	
Aksi Aktor	Reaksi Sistem
1. Klik menu Rangking	
	<p>2. Menampilkan halaman rangking yang meliputi :</p> <ul style="list-style-type: none"> <li>• Tombol Rangking per sektor</li> <li>• <i>Field search</i></li> <li>• Tabel rangking majelis jemaat yang meliputi : <ul style="list-style-type: none"> <li>- <i>Rank</i></li> <li>- Nama</li> <li>- Moralitas</li> <li>- Karakter</li> <li>- Kompetensi</li> <li>- Administrasi</li> <li>- Kekristenan</li> <li>- Total Nilai</li> <li>- Tombil <i>detail</i></li> </ul> </li> </ul>



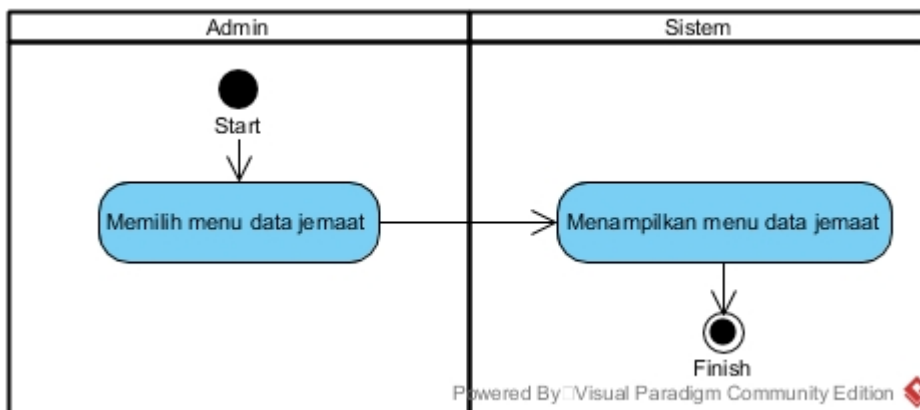
3. Memilih salah satu sector dan mengklik Ranging per sektor	
	<p>4. Menampilkan halaman ranging salah satu sector yang meliputi :</p> <ul style="list-style-type: none"><li>• Tombol kembali</li><li>• <i>Field search</i></li><li>• Tabel ranging majelis jemaat per sektor yang meliputi :<ul style="list-style-type: none"><li>- <i>Rank</i></li><li>- Nama</li><li>- Moralitas</li><li>- Karakter</li><li>- Kompetensi</li><li>- Administrasi</li><li>- Kekristenan</li><li>- Total Nilai</li></ul></li></ul>

## LAMPIRAN B.1 ACTIVITY

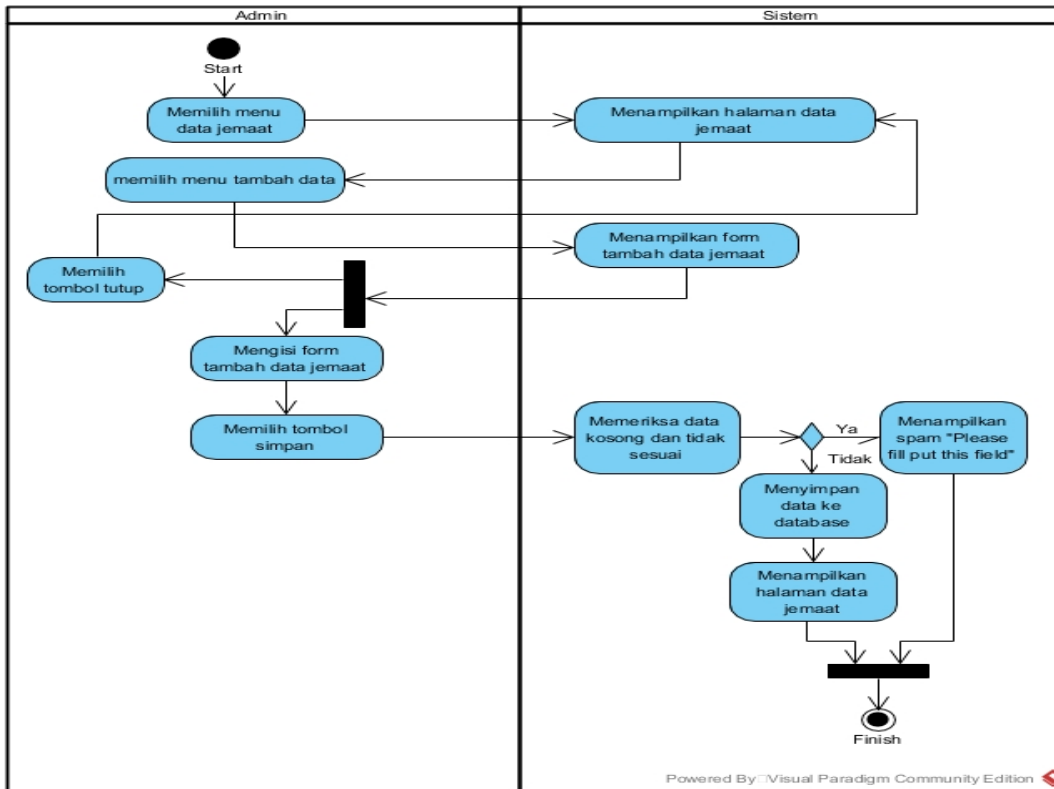
### B.1 Activity Login



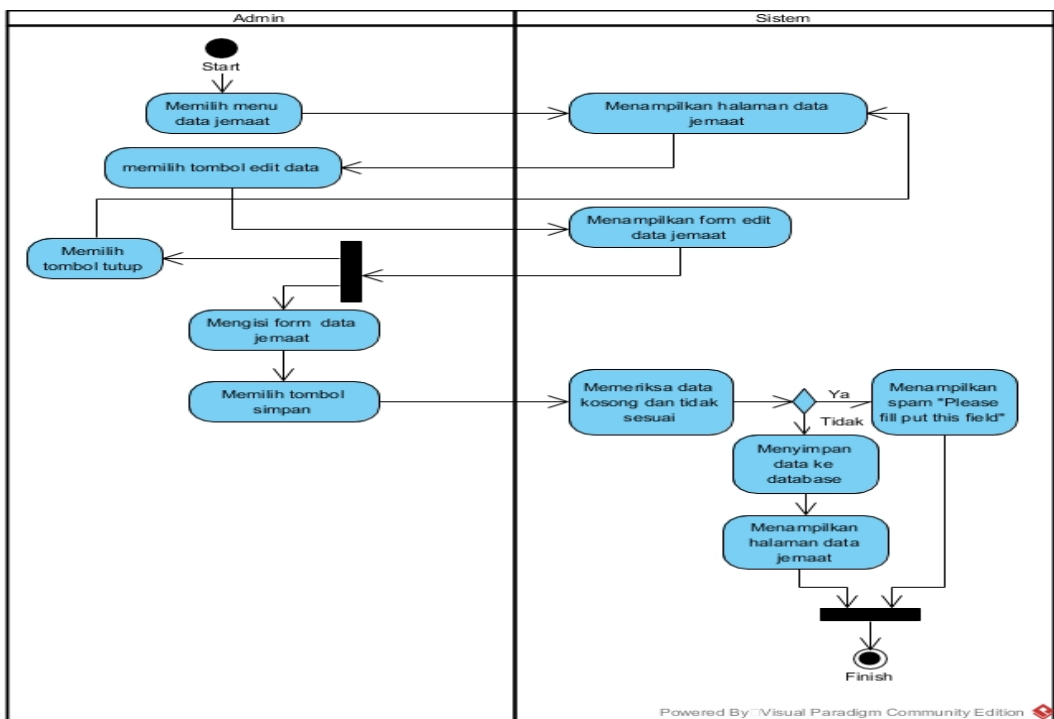
### B.2 Activity View Data Jemaat



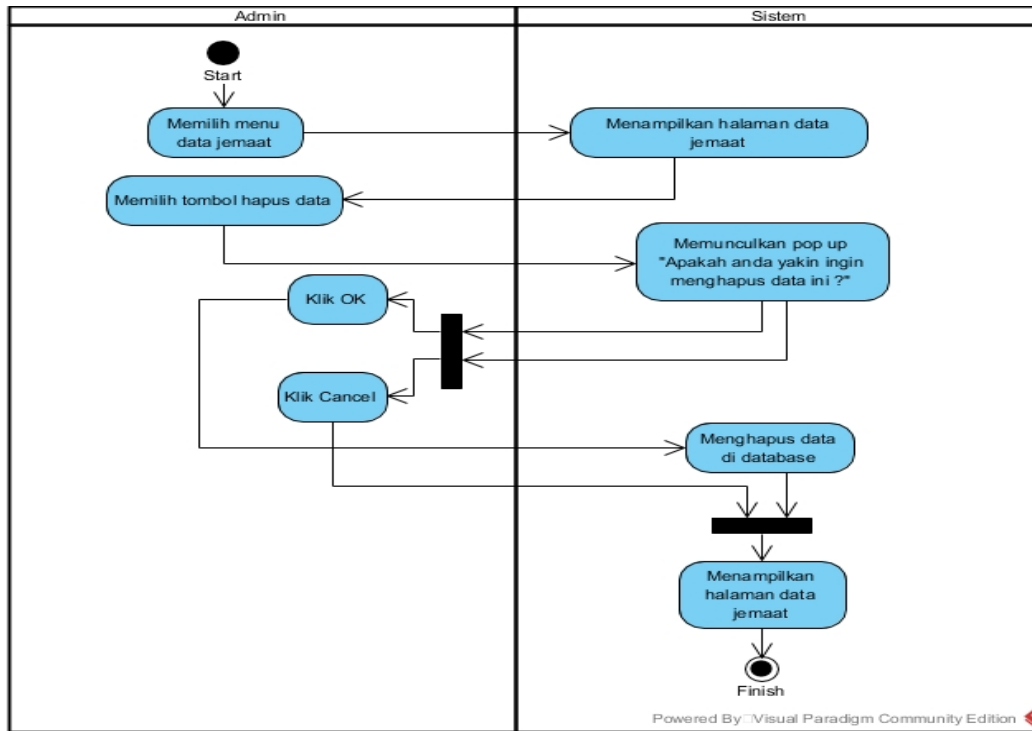
### B.3 Activity Insert Data Jemaat



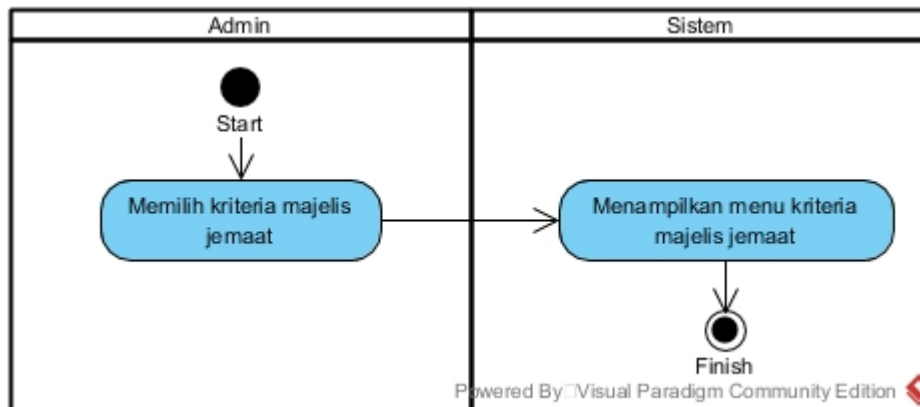
### B.4 Activity Update Data Jemaat



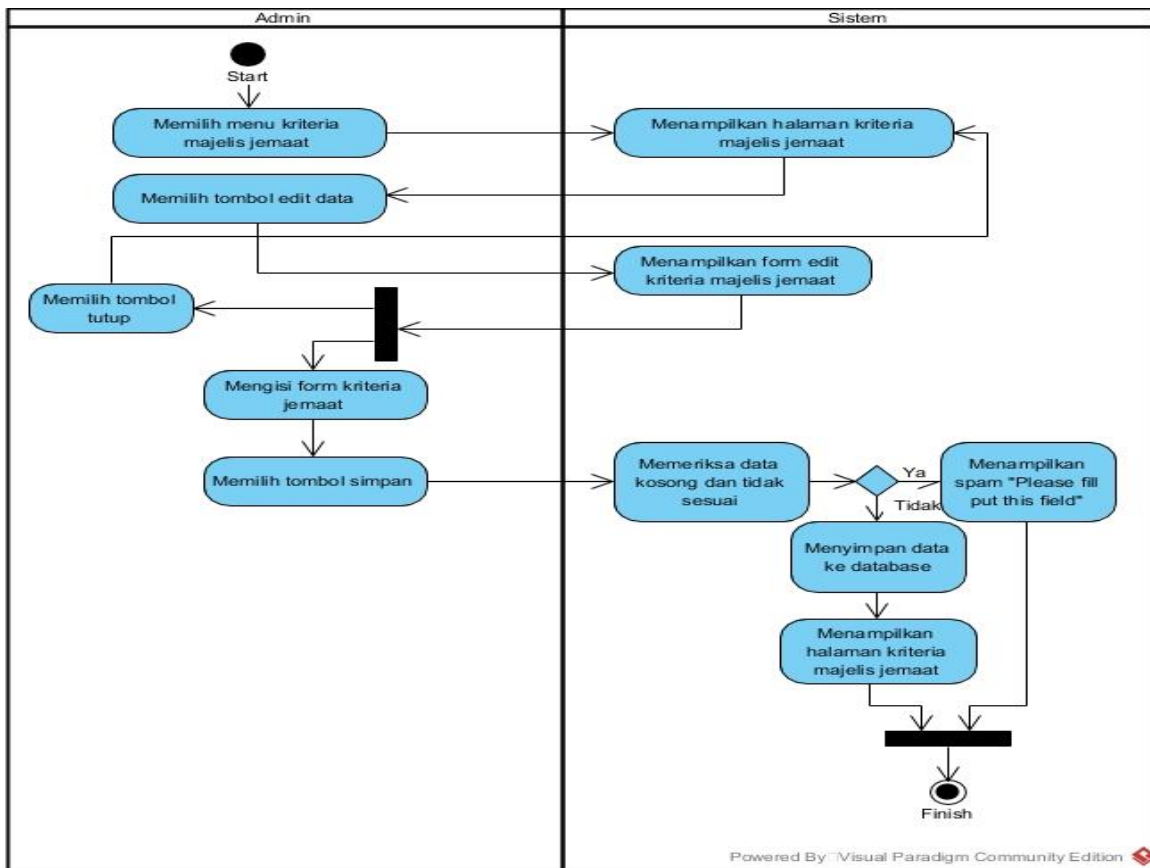
### B.5 Activity Delete Data Jemaat



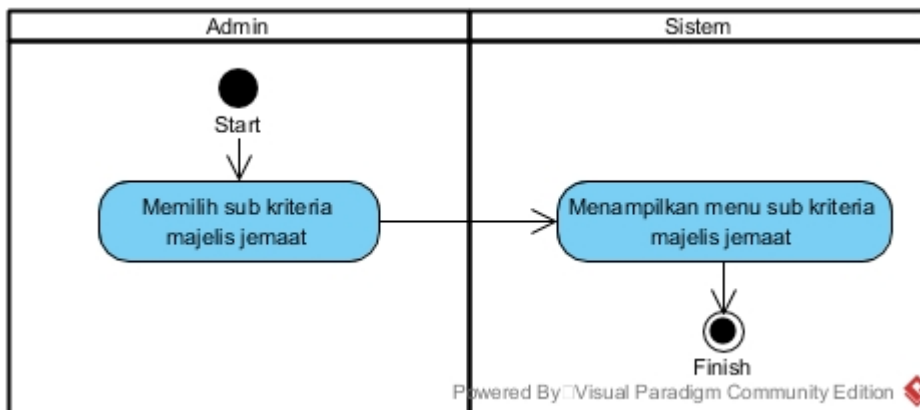
### B.6 Activity View Kriteria Majelis Jemaat



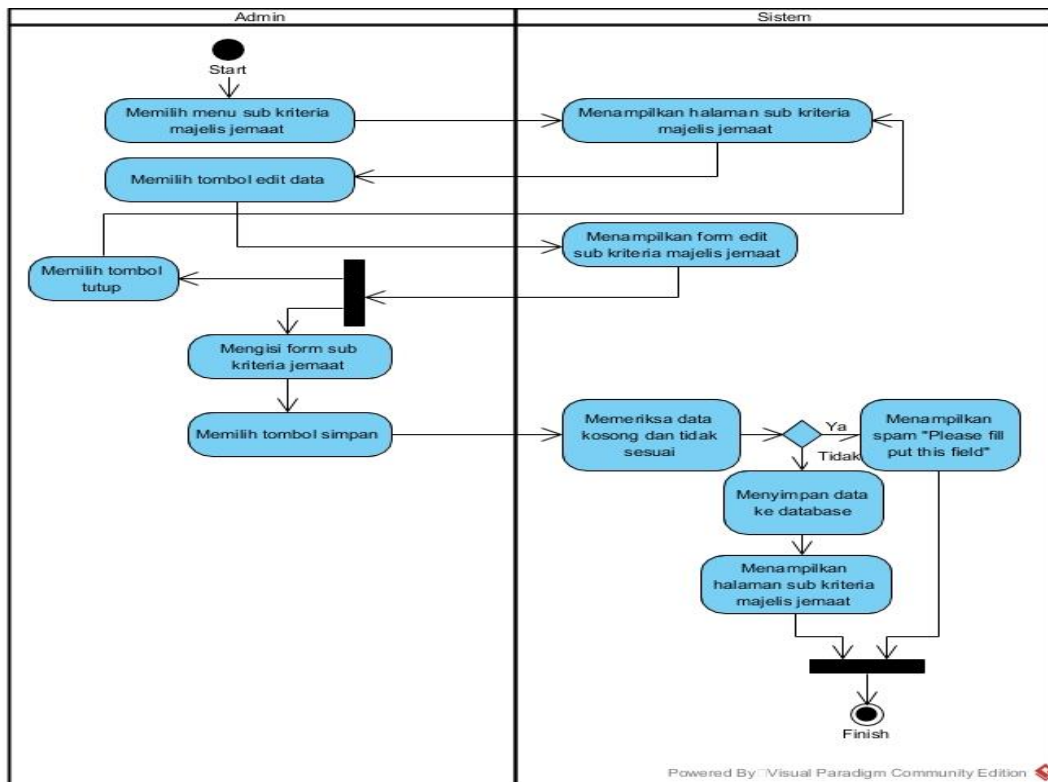
### B.7 Activity Update Kriteria Majelis Jemaat



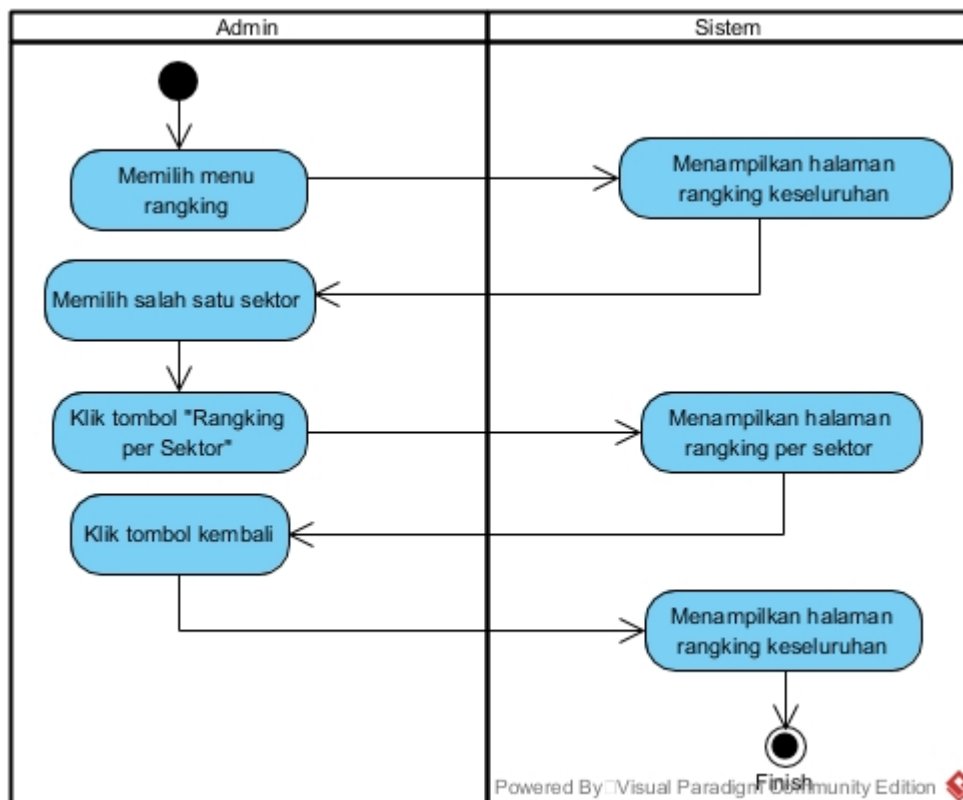
### B.8 Activity View Sub Kriteria Majelis Jemaat



### B.9 Activity Update Sub Kriteria Majelis Jemaat

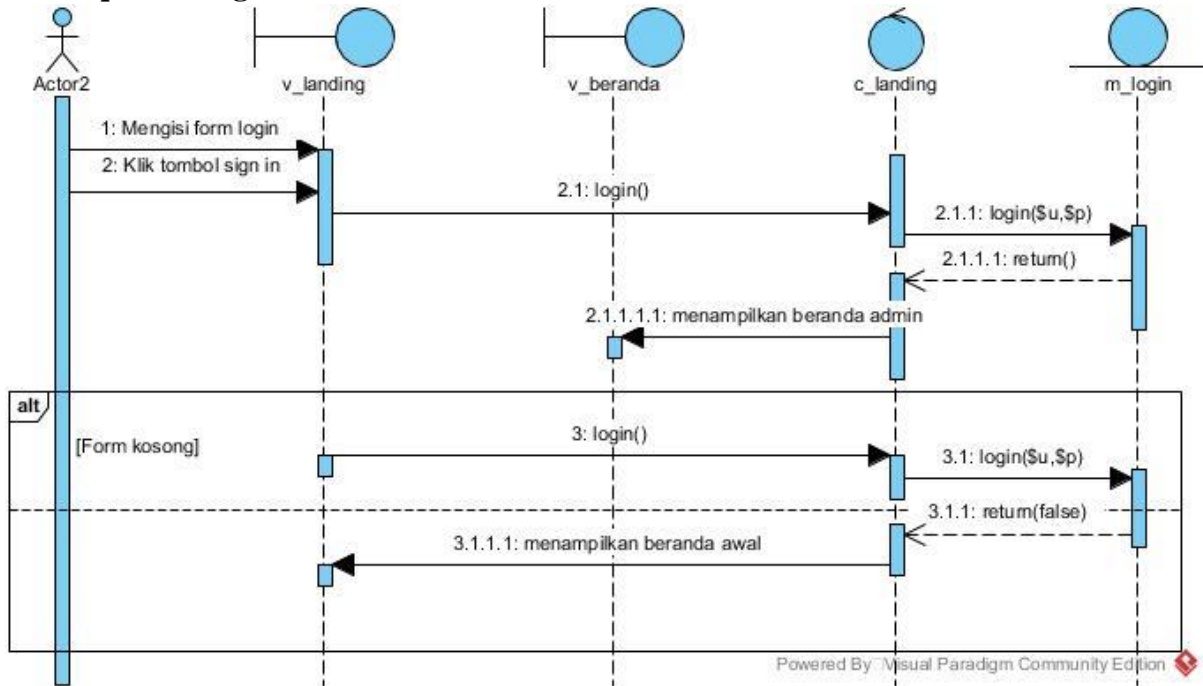


### B.10 Activity Rangking

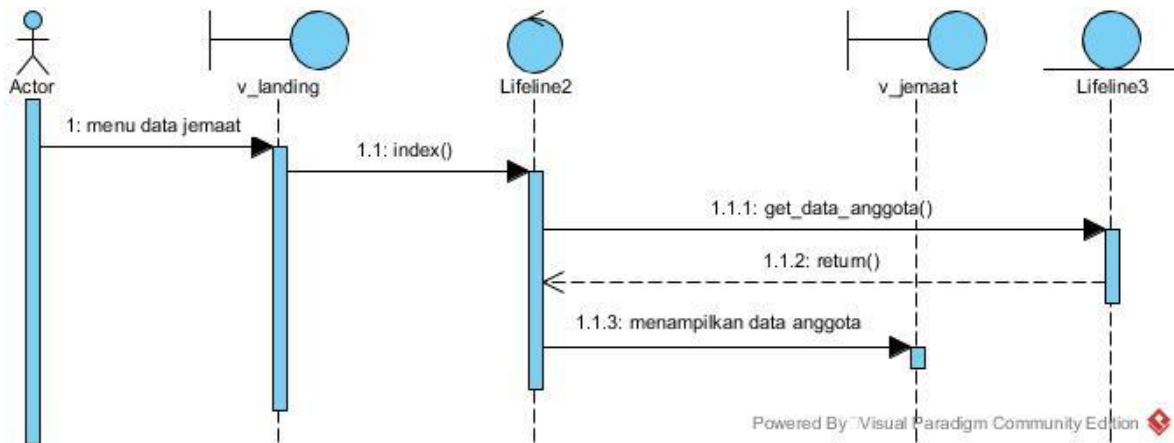


## LAMPIRAN C. SEQUENCE

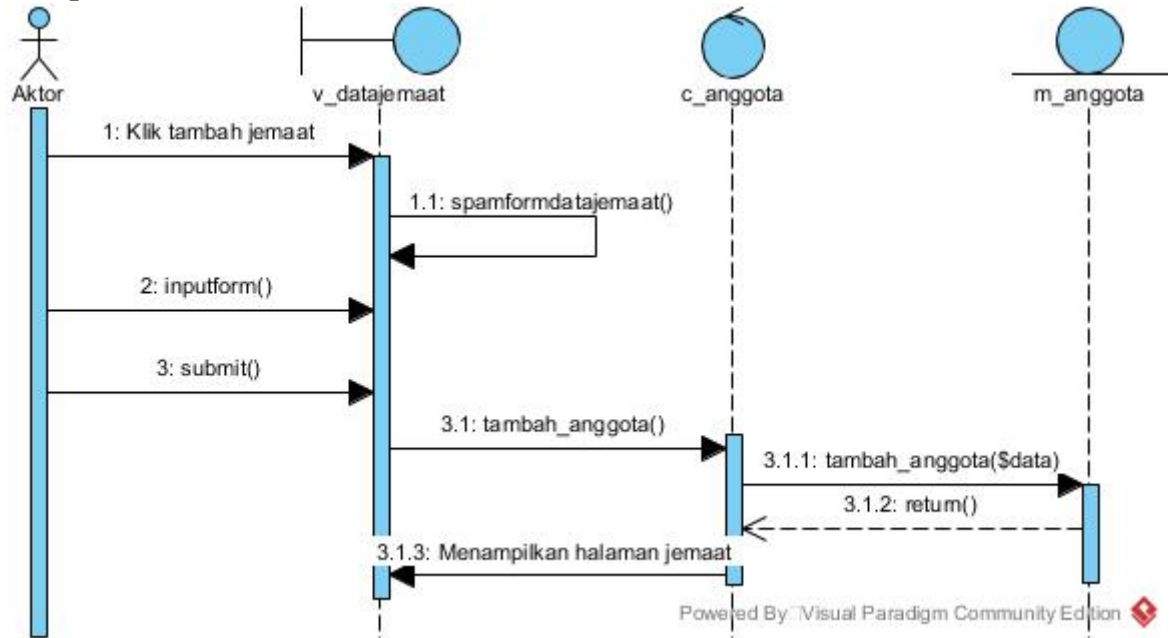
### C.1 Sequence Login



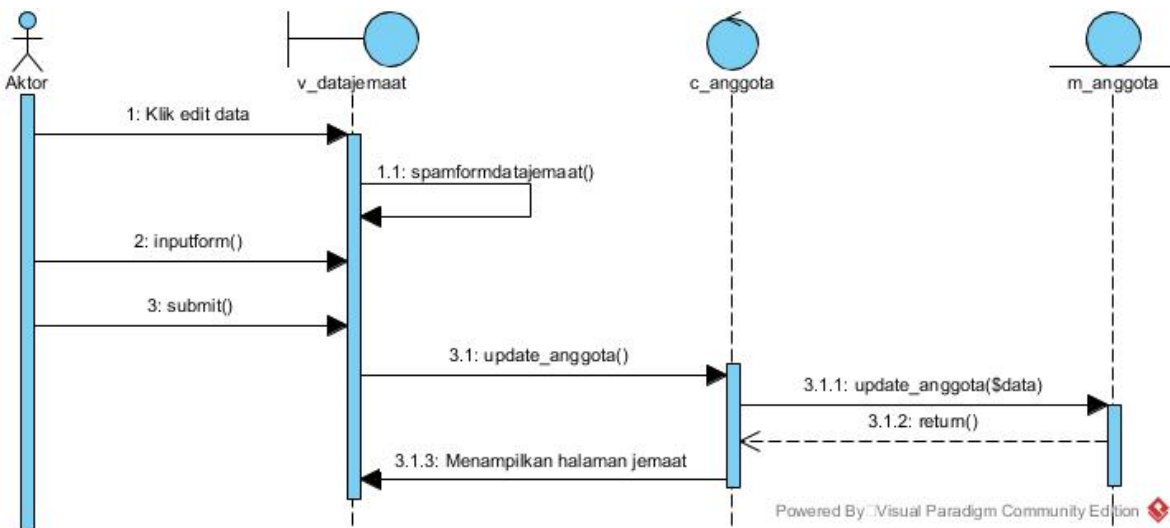
### C.2 Sequence View Data Jemaat



### C.3 Sequence Insert Data Jemaat

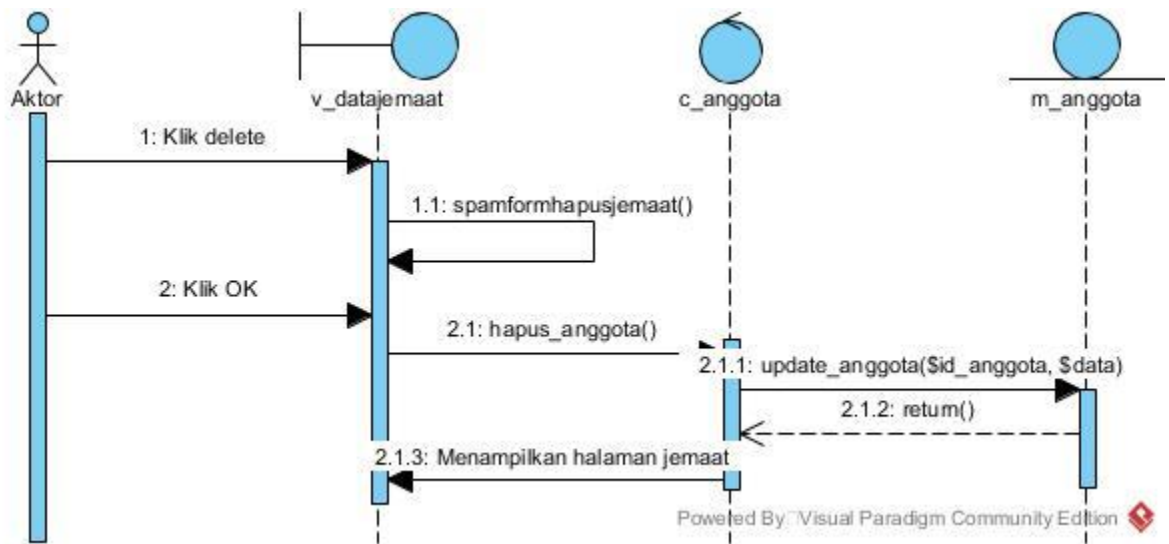


### C.4 Sequence Update Data Jemaat

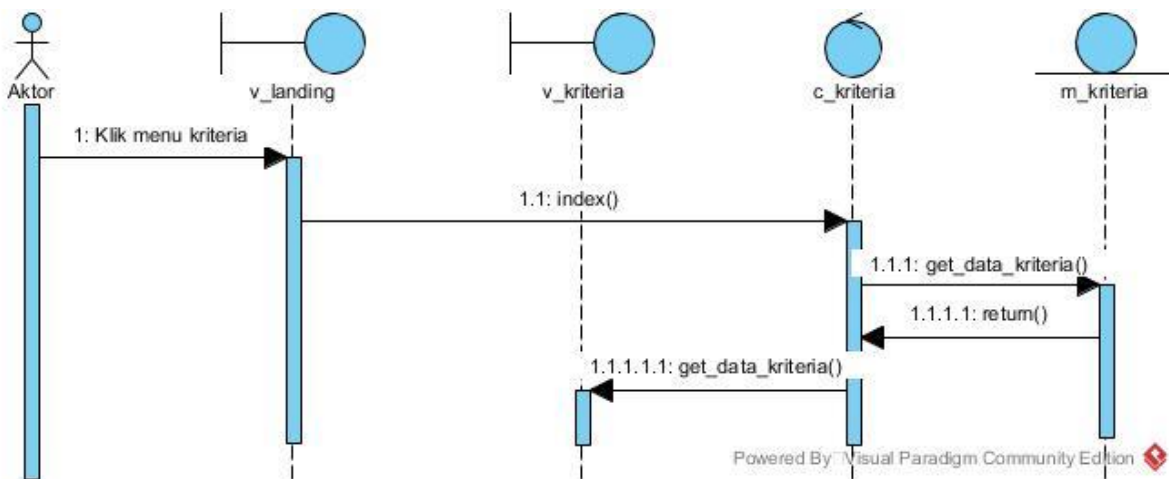




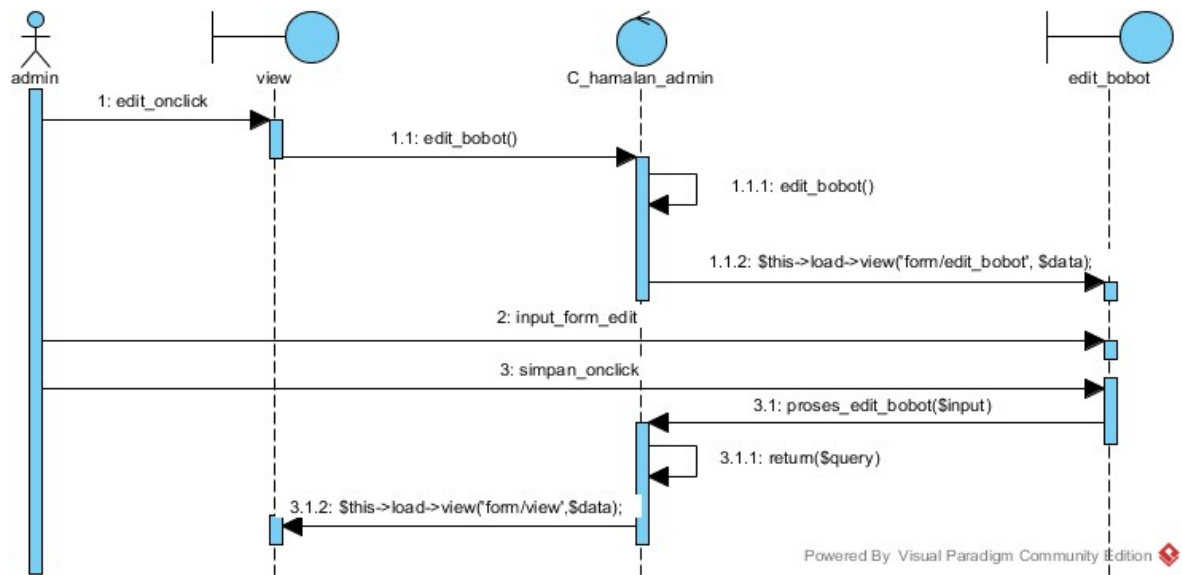
### C.5 Sequence Delete Data Jemaat



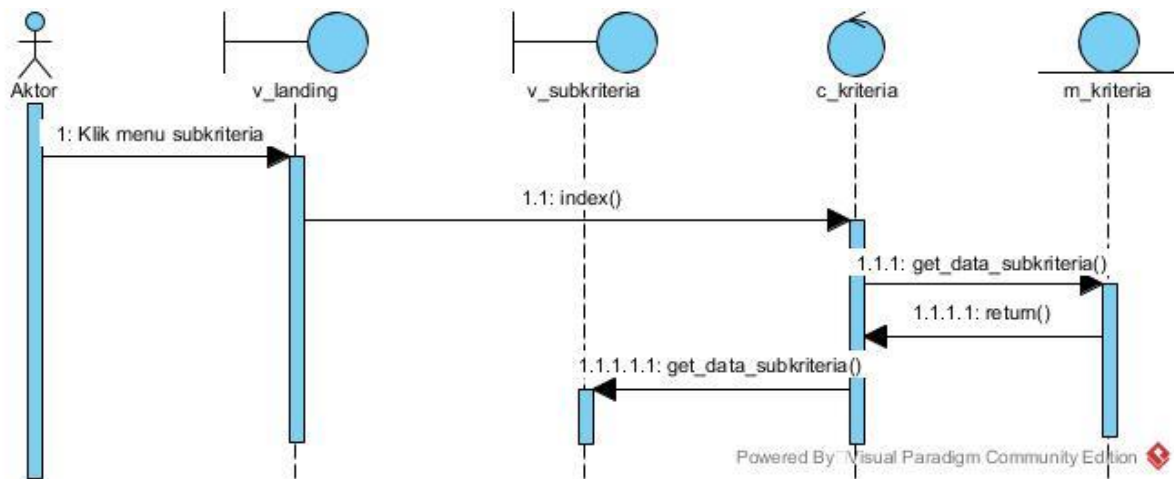
### C.6 Sequence View Kriteria Majelis Jemaat



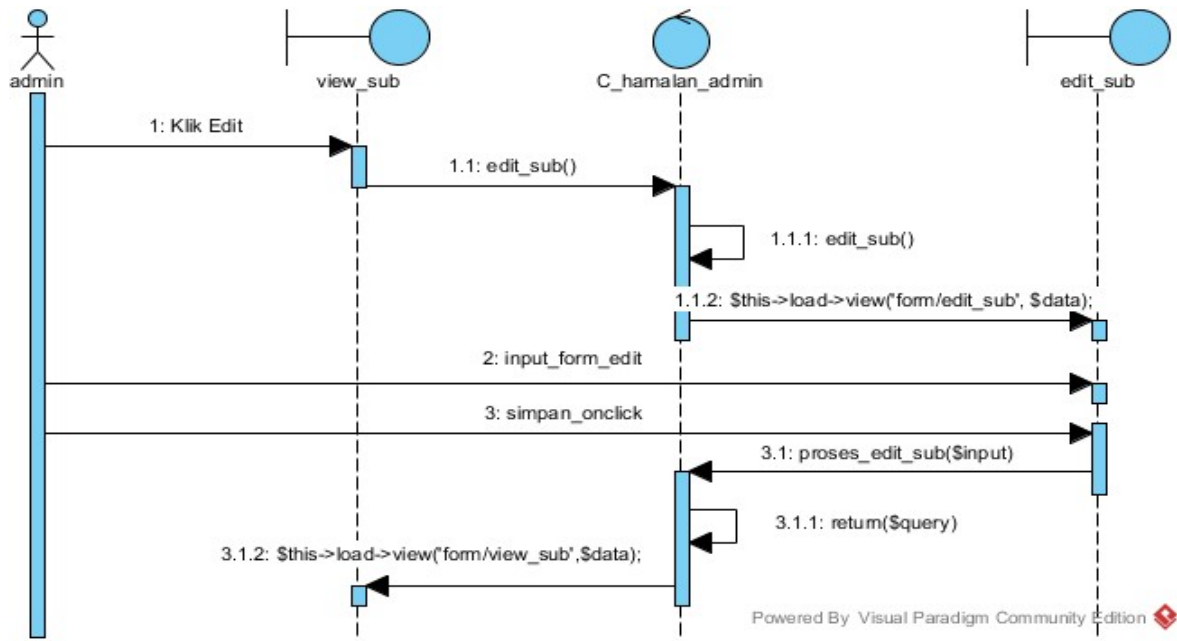
### C.7 Sequence Update Kriteria Majelis Jemaat



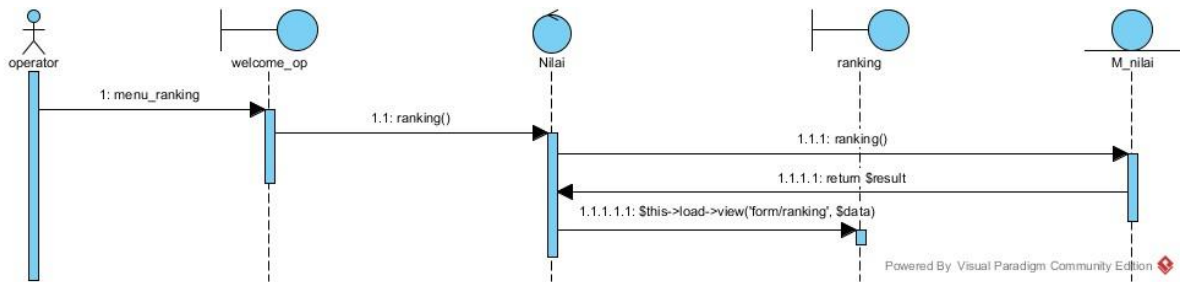
### C.8 Sequence View Sub Kriteria Majelis Jemaat



### C.9 Sequence Update Sub Kriteria Majelis Jemaat



### C.10 Sequence View Rangka



## LAMPIRAN D. KODE PROGRAM

### D.1 Tabel Kode Program Controller/anggota.php

```
class anggota extends CI_Controller
{
    public function __construct(){
        parent::__construct();
        $this->load->model('M_anggota');
        $this->load->model('M_kriteria');
        $this->load->model('M_subkriteria');
        if (!$this->session->userdata('role')){
            redirect('/');
        }
    }

    public function index(){
        $data['anggota'] = $this->M_anggota->get_data_anggota()->result();
        $data['kriteria'] = $this->M_kriteria->get_data_kriteria()->result();
        $this->load->view('Admin/header.php', $data);
        $this->load->view('Admin/daftaranggota.php');
        $this->load->view('Admin/footer.php');
    }

    public function tambah_anggota(){

        $nama = $this->input->post('nama');
        $umur = $this->input->post('umur');
        $pekerjaan = $this->input->post('pekerjaan');
        $tempat_lahir = $this->input->post('tempat_lahir');
        $tanggal_lahir = $this->input->post('tanggal_lahir');
        $alamat = $this->input->post('alamat');
        $sektor = $this->input->post('sektor');
        $upload = $this->input->post('surat_akta');
        $upload1 = $this->input->post('surat_baptis');
        $upload2 = $this->input->post('surat_peneguhan');

        //-----ambil bobot dan normalisasi masing-masing kriteria----
        -----
        //Kriteria Moralitas yang tinggi
        $normalisasi_moralitas = $this->M_kriteria->get_filter_kriteria(1)->row()-
        >normalisasi_kriteria;
        $utility_moralitas = $this->M_subkriteria->get_filter_subkriteria($this->input-
        >post('kriteria_1'))->row()->nilai_subkriteria;
        $jumlah_moralitas = $normalisasi_moralitas * $utility_moralitas;

        //Kriteria Karakter yang baik
        $normalisasi_karakter = $this->M_kriteria->get_filter_kriteria(2)->row()-
        >normalisasi_kriteria;
```

```

$utility_karakter = $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_2'))->row()->nilai_subkriteria;
$jumlah_karakter = $normalisasi_karakter * $utility_karakter;

//Kriteria Kompetensi yang baik
$normalisasi_kompetensi = $this->M_kriteria->get_filter_kriteria(3)->row()-
>normalisasi_kriteria;
$utility_kompetensi = $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_3'))->row()->nilai_subkriteria;
$jumlah_kompetensi = $normalisasi_kompetensi * $utility_kompetensi;

//Kriteria Kelengkapan administrasi
$normalisasi_administrasi = $this->M_kriteria->get_filter_kriteria(4)->row()-
>normalisasi_kriteria;
$utility_administrasi = $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_4'))->row()->nilai_subkriteria;
$jumlah_administrasi = $normalisasi_administrasi * $utility_administrasi;

//Kriteria Kekristenan
$normalisasi_kekristenan = $this->M_kriteria->get_filter_kriteria(5)->row()-
>normalisasi_kriteria;
$utility_kekristenan = $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_5'))->row()->nilai_subkriteria;
$jumlah_kekristenan = $normalisasi_kekristenan * $utility_kekristenan;

//Hitung Total Nilai
$total_nilai = $jumlah_moralitas + $jumlah_karakter + $jumlah_kompetensi +
$jumlah_administrasi + $jumlah_kekristenan;

$data = array(
    'nama' => $nama,
    'umur' => $umur,
    'pekerjaan' => $pekerjaan,
    'tempat_lahir' => $tempat_lahir,
    'tanggal_lahir' => $tanggal_lahir,
    'alamat' => $alamat,
    'sektor' => $sektor,
    'moralitas' => $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_1'))->row()->id_subkriteria,
    'karakter' => $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_2'))->row()->id_subkriteria,
    'kompetensi' => $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_3'))->row()->id_subkriteria,
    'kelengkapan_administrasi' => $this->M_subkriteria->get_filter_subkriteria($this-
>input->post('kriteria_4'))->row()->id_subkriteria,
    'kekristenan' => $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_5'))->row()->id_subkriteria,
    'total_nilai' => $total_nilai,
    'surat_akta_kelahiran' => $upload,
    'surat_baptis' => $upload1,

```

```

        'surat_peneguhan_SIDI' => $upload2,
    );

    //simpan data anggota
    $this->M_anggota->tambah_anggota($data);
    $id_anggota = $this->db->insert_id();

    //simpan detail perhitungan skor ke tabel ranking
    $jumlah[1] = $jumlah_moralitas;
    $jumlah[2] = $jumlah_karakter;
    $jumlah[3] = $jumlah_kompetensi;
    $jumlah[4] = $jumlah_administrasi;
    $jumlah[5] = $jumlah_kekristenan;

    $utility[1] = $utility_moralitas;
    $utility[2] = $utility_karakter;
    $utility[3] = $utility_kompetensi;
    $utility[4] = $utility_administrasi;
    $utility[5] = $utility_kekristenan;

    $kriteria = $this->M_kriteria->get_data_kriteria()->result();
    $i = 1;
    foreach ($kriteria as $kr){
        $data_ranking = array(
            'id_anggota' => $id_anggota,
            'id_kriteria' => $kr->id_kriteria,
            'utility' => $utility[$i],
            'jumlah' => $jumlah[$i]
        );
        $this->M_anggota->tambah_ranking($data_ranking);
        $i++;
    }
    $this->session->set_flashdata('success', 'Data Berhasil Ditambah. ');
    redirect('anggota');
}

public function update_anggota(){
    $id_anggota = $this->input->post('id_anggota');
    $nama = $this->input->post('nama');
    $umur = $this->input->post('umur');
    $pekerjaan = $this->input->post('pekerjaan');
    $tempat_lahir = $this->input->post('tempat_lahir');
    $tanggal_lahir = $this->input->post('tanggal_lahir');
    $alamat = $this->input->post('alamat');
    $sektor = $this->input->post('sektor');
    $upload = $this->input->post('surat_akta');
    $upload1 = $this->input->post('surat_baptis');
    $upload2 = $this->input->post('surat_peneguhan');
}

```

```

//-----ambil bobot dan normalisasi masing-masing kriteria----
-----
//Kriteria Moralitas yang tinggi
$normalisasi_moralitas = $this->M_kriteria->get_filter_kriteria(1)->row()-
>normalisasi_kriteria;
$utility_moralitas = $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_1'))->row()->nilai_subkriteria;
$jumlah_moralitas = $normalisasi_moralitas * $utility_moralitas;

//Kriteria Karakter yang baik
$normalisasi_karakter = $this->M_kriteria->get_filter_kriteria(2)->row()-
>normalisasi_kriteria;
$utility_karakter = $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_2'))->row()->nilai_subkriteria;
$jumlah_karakter = $normalisasi_karakter * $utility_karakter;

//Kriteria Kompetensi yang baik
$normalisasi_kompetensi = $this->M_kriteria->get_filter_kriteria(3)->row()-
>normalisasi_kriteria;
$utility_kompetensi = $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_3'))->row()->nilai_subkriteria;
$jumlah_kompetensi = $normalisasi_kompetensi * $utility_kompetensi;

//Kriteria Kelengkapan administrasi
$normalisasi_administrasi = $this->M_kriteria->get_filter_kriteria(4)->row()-
>normalisasi_kriteria;
$utility_administrasi = $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_4'))->row()->nilai_subkriteria;
$jumlah_administrasi = $normalisasi_administrasi * $utility_administrasi;

//Kriteria Kekristenan
$normalisasi_kekristenan = $this->M_kriteria->get_filter_kriteria(5)->row()-
>normalisasi_kriteria;
$utility_kekristenan = $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_5'))->row()->nilai_subkriteria;
$jumlah_kekristenan = $normalisasi_kekristenan * $utility_kekristenan;

//Hitung Total Nilai
$total_nilai = $jumlah_moralitas + $jumlah_karakter + $jumlah_kompetensi +
$jumlah_administrasi + $jumlah_kekristenan;

$data = array(
    'nama' => $nama,
    'umur' => $umur,
    'pekerjaan' => $pekerjaan,
    'tempat_lahir' => $tempat_lahir,
    'tanggal_lahir' => $tanggal_lahir,
    'alamat' => $alamat,
    'sektor' => $sektor,

```

```

        'moralitas' => $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_1'))->row()->id_subkriteria,
        'karakter' => $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_2'))->row()->id_subkriteria,
        'kompetensi' => $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_3'))->row()->id_subkriteria,
        'kelengkapan_administrasi' => $this->M_subkriteria->get_filter_subkriteria($this-
>input->post('kriteria_4'))->row()->id_subkriteria,
        'kekristenan' => $this->M_subkriteria->get_filter_subkriteria($this->input-
>post('kriteria_5'))->row()->id_subkriteria,
        'total_nilai' => $total_nilai,
        'surat_akta_kelahiran' => $upload,
        'surat_baptis' => $upload1,
        'surat_peneguhan_SIDI' => $upload2,
    );

    //simpan data anggota
    $status1 = $this->M_anggota->update_anggota($id_anggota, $data);

    //simpan detail perhitungan skor ke tabel ranking
    $jumlah[1] = $jumlah_moralitas;
    $jumlah[2] = $jumlah_karakter;
    $jumlah[3] = $jumlah_kompetensi;
    $jumlah[4] = $jumlah_administrasi;
    $jumlah[5] = $jumlah_kekristenan;

    $utility[1] = $utility_moralitas;
    $utility[2] = $utility_karakter;
    $utility[3] = $utility_kompetensi;
    $utility[4] = $utility_administrasi;
    $utility[5] = $utility_kekristenan;

    $kriteria = $this->M_kriteria->get_data_kriteria()->result();
    $i = 1;
    foreach ($kriteria as $kr){
        $data_ranking = array(
            'utility' => $utility[$i],
            'jumlah' => $jumlah[$i]
        );
        $status = $this->M_anggota->update_ranking($id_anggota, $kr->id_kriteria,
$data_ranking);
        $i++;
    }

    $this->session->set_flashdata('success', 'Data Berhasil Diubah. ');
    redirect('anggota');
}

function hapus_anggota($id_anggota){
    $this->M_anggota->hapus_anggota($id_anggota);
}

```



```

    $this->session->set_flashdata('success', 'Data Berhasil Dihapus. ');
    redirect('anggota');
}
}

```

## D.2 Tabel Kode Program Controller/beranda.php

```

class beranda extends CI_Controller
{
    public function __construct(){
        parent::__construct();
        $this->load->model('M_login');
        $this->load->model('M_anggota');
        if (!$this->session->userdata('role')){
            redirect('/');
        }
    }

    public function index(){
        $data['anggota'] = $this->M_anggota->get_data_anggota()->result();
        $this->load->view('Admin/header.php',$data);
        $this->load->view('Admin/beranda.php');
        $this->load->view('Admin/footer.php');
    }
}

```

## D.3 Tabel Kode Program Controller/kriteria.php

```

class kriteria extends CI_Controller
{
    public function __construct(){
        parent::__construct();
        $this->load->model('M_kriteria');

        if (!$this->session->userdata('role')){
            redirect('/');
        }
    }

    public function index(){
        $data['kriteria'] = $this->M_kriteria->get_data_kriteria()->result();
        $this->load->view('Admin/header.php', $data);
        $this->load->view('Admin/daftarkriteria.php');
        $this->load->view('Admin/footer.php');
    }

    public function tambah_kriteria(){
        $nama_kriteria = $this->input->post('nama_kriteria');
        $bobot_kriteria = $this->input->post('bobot_kriteria');
    }
}

```

```

//hitung normalisasi
$jumlah_bobot = $this->db->select('sum(bobot_kriteria) as jumlah')->get('kriteria')->row()->jumlah;

if ($jumlah_bobot + $bobot_kriteria > 100){
    $this->session->set_flashdata('error', 'Jumlah Bobot Melebihi 100. Gagal
Menambah Data. ');
    redirect('kriteria');
} else {

    $normalisasi = $bobot_kriteria / $jumlah_bobot;

    $data = array(
        'nama_kriteria' => $nama_kriteria,
        'bobot_kriteria' => $bobot_kriteria,
        'normalisasi_kriteria' => $normalisasi,
    );

    $this->M_kriteria->tambah_kriteria($data);
    $this->session->set_flashdata('success', 'Berhasil Menambah Data. ');
    redirect('kriteria');
}
}

public function ubah_kriteria(){
    $id_kriteria = $this->input->post('id_kriteria');
    $nama_kriteria = $this->input->post('nama_kriteria');
    $bobot_lama = $this->input->post('bobot_lama');
    $bobot_kriteria = $this->input->post('bobot_kriteria');

    //ambil jumlah bobot lama
    $jumlah_bobot = $this->db->select('sum(bobot_kriteria) as jumlah')->get('kriteria')->row()->jumlah;

    if ($jumlah_bobot - $bobot_lama + $bobot_kriteria > 100){
        $this->session->set_flashdata('error', 'Jumlah Bobot Melebihi 100. Gagal
Mengubah Data. ');
        redirect('kriteria');
    } else {
        //hitung normalisasi
        $jumlah_normal = $jumlah_bobot - $bobot_lama + $bobot_kriteria;
        $normalisasi = $bobot_kriteria / $jumlah_normal;

        $data = array(
            'nama_kriteria' => $nama_kriteria,
            'bobot_kriteria' => $bobot_kriteria,
            'normalisasi_kriteria' => $normalisasi,
            'jumlah_normal' => $jumlah_normal,
        );
    }
}

```

```

        $this->M_kriteria->ubah_kriteria($data, $id_kriteria);
        $this->session->set_flashdata('success', 'Data Berhasil Diubah. ');
        redirect('kriteria');
    }
}

public function hapus_kriteria($id_kriteria){
    $this->M_kriteria->hapus_kriteria($id_kriteria);
    $this->session->set_flashdata('success', 'Data Berhasil Dihapus. ');
    redirect('kriteria');
}
}

```

#### D.4 Tabel Kode Program Controller/landing.php

```

class landing extends CI_Controller
{
    public function __construct(){
        parent::__construct();
        $this->load->model('M_login');
        $this->load->model('M_anggota');
        if ($this->session->userdata('id_user') != NULL){
            redirect('beranda');
        }
    }

    public function index(){
        $sektor = $this->input->post('sektor');
        $data['anggota'] = $this->M_anggota->get_sektor_anggota($sektor)-
>result();
        $data['sektor'] = $sektor;
        $this->load->view('landing.php', $data);
    }

    public function v_login(){
        $this->load->view('login.php');
    }

    public function login(){
        $u = $this->input->post('username');
        $p = $this->input->post('password');

        $b = $this->M_login->login($u, $p);
        if($b){
            $this->session->set_flashdata('success', 'Login Berhasil !');
            redirect(site_url(). 'beranda');
        }
        else{
            $this->session->set_flashdata('error', 'Username atau password salah!');

```

```

        redirect('/');
    }
}

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

```

### D.5 Tabel Kode Program Controller/rangking.php

```

class ranking extends CI_Controller
{
    public function __construct(){
        parent::__construct();
        $this->load->model('M_anggota');
        $this->load->model('M_kriteria');
        $this->load->model('M_subkriteria');
        if (!$this->session->userdata('role')){
            redirect('/');
        }
    }

    public function index(){
        $data['anggota'] = $this->M_anggota->get_data_anggota()->result();
        $this->load->view('Admin/header.php', $data);
        $this->load->view('Admin/rangking.php');
        $this->load->view('Admin/footer.php');
    }

    public function detail_ranking($id_anggota){
        $data['anggota'] = $this->M_anggota->get_detail_anggota($id_anggota)-
>result();
        $data['rangking'] = $this->M_anggota->get_data_ranking($id_anggota)-
>result();
        $data['subkriteria'] = $this->M_subkriteria-
>get_data_subkriteria($id_anggota)->result();
        $this->load->view('Admin/header.php', $data);
        $this->load->view('Admin/detail_ranking.php');
        $this->load->view('Admin/footer.php');
    }

    public function ranking_sektor(){
        $sektor = $this->input->post('sektor');
        $data['anggota'] = $this->M_anggota->get_sektor_anggota($sektor)-
>result();
        $data['sektor'] = $sektor;
        $this->load->view('Admin/header.php', $data);
        $this->load->view('Admin/rangking_sektor.php');
    }
}

```

```

        $this->load->view('Admin/footer.php');
    }
}

```

## D.6 Tabel Kode Program Controller/subkriteria.php

```

class subkriteria extends CI_Controller
{
    public function __construct(){
        parent::__construct();
        $this->load->model('M_subkriteria');
        $this->load->model('M_kriteria');

        if (!$this->session->userdata('role')){
            redirect('/');
        }
    }

    public function index(){
        $data['subkriteria'] = $this->M_subkriteria->get_data_subkriteria()->result();
        $data['kriteria'] = $this->M_kriteria->get_data_kriteria()->result();
        $this->load->view('Admin/header.php', $data);
        $this->load->view('Admin/daftarsubkriteria.php');
        $this->load->view('Admin/footer.php');
    }

    public function tambah_subkriteria(){
        $nama = $this->input->post('nama_subkriteria');
        $nilai = $this->input->post('nilai_subkriteria');
        $kriteria = $this->input->post('id_kriteria');

        $data = array(
            'nama_subkriteria' => $nama,
            'nilai_subkriteria' => $nilai,
            'id_kriteria' => $kriteria,
        );

        $this->M_subkriteria->tambah_subkriteria($data);
        $this->session->set_flashdata('success', 'Berhasil Menambah Data. ');
        redirect('subkriteria');
    }

    public function update_subkriteria(){
        $id_subkriteria = $this->input->post('id_subkriteria');
        $nama = $this->input->post('nama_subkriteria');
        $nilai = $this->input->post('nilai_subkriteria');
        $kriteria = $this->input->post('id_kriteria');

        $data = array(
            'nama_subkriteria' => $nama,

```

```

        'nilai_subkriteria' => $nilai,
        'id_kriteria' => $kriteria,
    );

    $this->M_subkriteria->update_subkriteria($id_subkriteria, $data);
    $this->session->set_flashdata('success', 'Berhasil Mengubah Data. ');
    redirect('subkriteria');
}

public function hapus_subkriteria($id_subkriteria){
    $this->M_subkriteria->hapus_subkriteria($id_subkriteria);
    $this->session->set_flashdata('success', 'Data Berhasil Dihapus. ');
    redirect('subkriteria');
}
}

```

#### D.7 Tabel Kode Program Model/m\_anggota.php

```

class M_anggota extends CI_Model
{
    public function get_data_anggota(){
        $this->db->select('anggota.*, moralitas.nama_subkriteria as
nama_moralitas, karakter.nama_subkriteria as nama_karakter,
kompetensi.nama_subkriteria as nama_kompetensi, administrasi.nama_subkriteria as
nama_administrasi, kekristenan.nama_subkriteria as nama_kekristenan,
moralitas.nilai_subkriteria as nilai_moralitas, karakter.nilai_subkriteria as nilai_karakter,
kompetensi.nilai_subkriteria as nilai_kompetensi, administrasi.nilai_subkriteria as
nilai_administrasi, kekristenan.nilai_subkriteria as nilai_kekristenan');
        $this->db->join('subkriteria moralitas','moralitas.id_subkriteria =
anggota.moralitas','left');
        $this->db->join('subkriteria karakter','karakter.id_subkriteria =
anggota.karakter','left');
        $this->db->join('subkriteria kompetensi','kompetensi.id_subkriteria =
anggota.kompetensi','left');
        $this->db->join('subkriteria administrasi','administrasi.id_subkriteria =
anggota.kelengkapan_administrasi','left');
        $this->db->join('subkriteria kekristenan','kekristenan.id_subkriteria =
anggota.kekristenan','left');
        $this->db->order_by('total_nilai', 'DESC');
        return $this->db->get('anggota');
    }

    public function get_detail_anggota($id_anggota){
        $this->db->select('anggota.*, moralitas.nama_subkriteria as
nama_moralitas, karakter.nama_subkriteria as nama_karakter,
kompetensi.nama_subkriteria as nama_kompetensi, administrasi.nama_subkriteria as
nama_administrasi, kekristenan.nama_subkriteria as nama_kekristenan,
moralitas.nilai_subkriteria as nilai_moralitas, karakter.nilai_subkriteria as nilai_karakter,
kompetensi.nilai_subkriteria as nilai_kompetensi, administrasi.nilai_subkriteria as
nilai_administrasi, kekristenan.nilai_subkriteria as nilai_kekristenan');
    }
}

```

```

        $this->db->join('subkriteria moralitas','moralitas.id_subkriteria =
anggota.moralitas','left');
        $this->db->join('subkriteria karakter','karakter.id_subkriteria =
anggota.karakter','left');
        $this->db->join('subkriteria kompetensi','kompetensi.id_subkriteria =
anggota.kompetensi','left');
        $this->db->join('subkriteria administrasi','administrasi.id_subkriteria =
anggota.kelengkapan_administrasi','left');
        $this->db->join('subkriteria kekristenan','kekristenan.id_subkriteria =
anggota.kekristenan','left');
        $this->db->where('id_anggota', $id_anggota);
        $this->db->order_by('total_nilai', 'DESC');
        return $this->db->get('anggota');
    }

    public function get_sektor_anggota($sektor){
        $this->db->select('anggota.*, moralitas.nama_subkriteria as
nama_moralitas, karakter.nama_subkriteria as nama_karakter,
kompetensi.nama_subkriteria as nama_kompetensi, administrasi.nama_subkriteria as
nama_administrasi, kekristenan.nama_subkriteria as nama_kekristenan,
moralitas.nilai_subkriteria as nilai_moralitas, karakter.nilai_subkriteria as nilai_karakter,
kompetensi.nilai_subkriteria as nilai_kompetensi, administrasi.nilai_subkriteria as
nilai_administrasi, kekristenan.nilai_subkriteria as nilai_kekristenan');
        $this->db->join('subkriteria moralitas','moralitas.id_subkriteria =
anggota.moralitas','left');
        $this->db->join('subkriteria karakter','karakter.id_subkriteria =
anggota.karakter','left');
        $this->db->join('subkriteria kompetensi','kompetensi.id_subkriteria =
anggota.kompetensi','left');
        $this->db->join('subkriteria administrasi','administrasi.id_subkriteria =
anggota.kelengkapan_administrasi','left');
        $this->db->join('subkriteria kekristenan','kekristenan.id_subkriteria =
anggota.kekristenan','left');
        $this->db->where('sektor', $sektor);
        $this->db->limit(7);
        $this->db->order_by('total_nilai', 'DESC');
        return $this->db->get('anggota');
    }

    public function tambah_anggota($data){
        $this->db->insert('anggota', $data);
    }

    public function tambah_ranking($data_ranking){
        $this->db->insert('ranking', $data_ranking);
    }

    public function update_anggota($id_anggota, $data){
        $this->db->where('id_anggota', $id_anggota);
        $this->db->update('anggota', $data);
    }

```

```

}

public function update_ranking($id_anggota, $id_kriteria, $data_ranking){
    $this->db->where('id_anggota', $id_anggota);
    $this->db->where('id_kriteria', $id_kriteria);
    $status = $this->db->update('ranking', $data_ranking);
    return $status;
}

public function hapus_anggota($id_anggota){
    $this->db->where('id_anggota', $id_anggota);
    $this->db->delete('anggota');
}

    public function get_data_ranking($id_anggota){
        $this->db->join('anggota', 'anggota.id_anggota = ranking.id_anggota');
        $this->db->join('kriteria', 'ranking.id_kriteria = kriteria.id_kriteria');
        $this->db->where('ranking.id_anggota', $id_anggota);
        return $this->db->get('ranking');
    }
}

```

#### D.8 Tabel Kode Program Model/m\_kriteria.php

```

class M_kriteria extends CI_Model
{
    public function get_data_kriteria(){
        return $this->db->get('kriteria');
    }

    public function get_filter_kriteria($id_kriteria){
        return $this->db->get_where('kriteria', array('id_kriteria' => $id_kriteria));
    }

    public function tambah_kriteria($data){
        $this->db->insert('kriteria', $data);
    }

    public function ubah_kriteria($data, $id_kriteria){
        $this->db->where('id_kriteria', $id_kriteria);
        $this->db->update('kriteria', $data);
    }

    public function hapus_kriteria($id_kriteria){
        $this->db->where('id_kriteria', $id_kriteria);
        $this->db->delete('kriteria');
    }
}

```



**D.9 Tabel Kode Program Model/m\_login.php**

```

class M_login extends CI_Model
{
    public function login($u,$p){
        $this->db->from('user');
        $this->db->where('username',$u);
        $this->db->where('password',$p);
        $a = $this->db->get();

        if($a->num_rows() == 1){
            $data=$a->result_array();
            $this->session->set_userdata('username',$data[0]['username']);
            $this->session->set_userdata('role',$data[0]['role']);

            return true;
        }
        else{
            return false;
        }
    }

    public function get_data(){
        return $this->db->get('anggota');
    }
}

```

**D.10 Tabel Kode Program Model/m\_subkriteria.php**

```

class M_subkriteria extends CI_Model
{
    public function get_data_subkriteria(){
        $this->db->join('kriteria', 'kriteria.id_kriteria = subkriteria.id_kriteria');
        return $this->db->get('subkriteria');
    }

    public function get_filter_subkriteria($id_subkriteria){
        return $this->db->get_where('subkriteria', array('id_subkriteria' => $id_subkriteria));
    }

    public function tambah_subkriteria($data){
        $this->db->insert('subkriteria', $data);
    }

    public function update_subkriteria($id_subkriteria, $data){
        $this->db->where('id_subkriteria', $id_subkriteria);
        $this->db->update('subkriteria', $data);
    }

    public function hapus_subkriteria($id_subkriteria){
        $this->db->where('id_subkriteria', $id_subkriteria);
    }
}

```

```
$this->db->delete('subkriteria');  
}  
}
```