



**SISTEM INFORMASI REKOMENDASI PENERIMA PROGRAM  
INDONESIA PINTAR (PIP) MENGGUNAKAN METODE *SIMPLE  
ADDITIVE WEIGHTING (SAW) DAN BORDA***

**SKRIPSI**

**Oleh**

**Mohammad Nahdhiana Hasmi**

**NIM 132410101056**

**FAKULTAS ILMU KOMPUTER  
PROGRAM STUDI SISTEM INFORMASI  
UNIVERSITAS JEMBER  
2019**



**SISTEM INFORMASI REKOMENDASI PENERIMA PROGRAM  
INDONESIA PINTAR (PIP) MENGGUNAKAN METODE *SIMPLE  
ADDITIVE WEIGHTING (SAW) DAN BORDA***

**SKRIPSI**

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

**Oleh**

**Mohammad Nahdhiana Hasmi**

**NIM 132410101056**

**FAKULTAS ILMU KOMPUTER  
PROGRAM STUDI SISTEM INFORMASI  
UNIVERSITAS JEMBER  
2019**

## PERSEMBAHAN

Skripsi ini saya persembahkan untuk :

1. Allah SWT yang senantiasa memberikan rahmat dan hidayah-Nya untuk mempermudah dan melancarkan dalam mengerjakan skripsi.
2. Ibunda tercinta Sri Utami dan Ayahanda Alm. Mustakim.
3. Kedua kakak perempuanku Fikriyatus Shofia dan Ulul Hikmah.
4. Keluarga besarku di rumah.
5. Guru – guruku baik dari pendidikan formal maupun informal.
6. Almamater Program Studi Sistem Informasi Universitas Jember.

**MOTTO**

“Jangan berputus asa! Tuhan tidak akan membebani seseorang kecuali yang sesuai dengan kemampuannya.”



## PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Mohammad Nahdhiana Hasmi

NIM : 132410101056

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Sistem Informasi Rekomendasi Penerima Program Indonesia Pintar (PIP) menggunakan Metode *Simple Additive Weighting* dan *Borda*” 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, 28 Januari 2019

Yang Menyatakan

Mohammad Nahdhiana Hasmi

**SKRIPSI**

**SISTEM INFORMASI REKOMENDASI PENERIMA PROGRAM  
INDONESIA PINTAR (PIP) MENGGUNAKAN METODE *SIMPLE  
ADDITIVE WEIGHTING (SAW) DAN BORDA***

**Oleh :**

**Mohammad Nahdhiana Hasmi**

**NIM 132410101056**

Pembimbing

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

Dosen Pembimbing Pendamping : Muhamad Arief Hidayat, S.Kom., M.Kom.

## PENGESAHAN PEMBIMBING

Skripsi berjudul “Sistem Informasi Rekomendasi Penerima Program Indonesia Pintar (PIP) menggunakan Metode *Simple additive Weighting* dan *Borda*”, telah diuji dan disahkan pada:

hari, tanggal :

tempat : Prodi Sistem Informasi Fakultas Ilmu Komputer  
Universitas Jember

Disetujui oleh:

Pembimbing I,

Pembimbing II,

Drs. Antonius Cahya P, M.App., Sc., Ph.D    Muhamad Arief Hidayat, S.Kom., M.Kom

NIP. 196909281993021001

NIP. 198403052010122002

## PENGESAHAN PENGUJI

Skripsi berjudul “Sistem Informasi Rekomendasi Penerima Program Indonesia Pintar (PIP) menggunakan Metode *Simple additive Weighting* dan *Borda*”, telah diuji dan disahkan pada:

hari, tanggal :

tempat : Prodi Sistem Informasi Fakultas Ilmu Komputer  
Universitas Jember

Tim Penguji :

Penguji I,

Penguji II,

Nelly Oktavia A, S.Si., MT  
NIP. 198410242009122008

Diksy Media F, S.Kom., M.Kom  
NIP. 760016853

Mengesahkan  
Dekan Fakultas Ilmu Komputer

Prof. Dr. H. Saiful Bukhori, ST., M.Kom  
NIP. 196811131994121001



## RINGKASAN

Sistem Informasi Rekomendasi Penerima Program Indonesia Pintar menggunakan Metode *Simple Additive Weights* dan *Borda* Mohammad Nahdhiana Hasmi, 132410101056; 2019, 139 HALAMAN; Fakultas Ilmu Komputer; Program Studi Sistem Informasi Universitas Jember.

Program Indonesia Pintar (PIP) merupakan penyempurnaan dari program Bantuan Siswa Miskin (BSM) dalam mendukung siswa dari keluarga miskin untuk memperoleh pendidikan. Bantuan Program Indonesia Pintar (PIP) tidak langsung diberikan begitu saja kepada para pelajar. Terdapat beberapa kriteria yang harus dipenuhi oleh pelajar agar bisa mendapatkan bantuan tersebut. Selain kriteria yang harus dipenuhi, terdapat pula pembatasan kuota bagi calon penerima Program Indonesia Pintar di masing-masing sekolah sehingga diperlukan adanya seleksi.

Sistem informasi yang akan dibangun pada penelitian ini bertujuan untuk membantu pihak sekolah dalam proses seleksi penerima PIP. Sistem informasi ini dapat menyeleksi daftar siswa yang mengajukan diri untuk mendapatkan PIP menggunakan metode *Simple Additive Weights* dan *Borda*. Hasil keluaran yang diberikan oleh sistem informasi yang dibangun berupa daftar siswa-siswa yang direkomendasikan untuk mendapatkan PIP.

## PRAKATA

Puji syukur kehadirat Allah SWT atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul “Sistem Informasi Rekomendasi Penerima Program Indonesia Pintar (PIP) menggunakan Metode *Simple Additive Weighting* dan *Borda*”. Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan Strata Satu (S1) pada Program Studi Sistem Informasi Universitas Jember.

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

1. Prof. Dr. H. Saiful Bukhori, S.T., M.Kom., selaku Dekan Fakultas Ilmu Komputer Universitas Jember;
2. Nelly Oktavia Adiwijaya, S.Si, M.T., selaku Dosen Pembimbing Akademik, yang telah mendampingi penulis sebagai mahasiswa;
3. Drs. Antonius Cahya P, M.App., Sc., Ph.D selaku Dosen Pembimbing Utama dan M. Arief Hidayat, S.Kom, M.Kom., selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi;
4. Seluruh Bapak dan Ibu dosen beserta staf karyawan di Program Studi Sistem Informasi Universitas Jember;
5. Ibunda tercinta Sri Utami dan Ayahanda Alm. Mustakim yang selalu mendukung dan mendoakan penulis.
6. Kedua kakak perempuanku Fikriyatus Shofia dan Ulul Hikmah.
7. Segenap keluarga besar penulis di rumah.
8. Teman-teman seperjuangan Intention angkatan 2013.
9. Teman-Teman Program Studi Sistem Informasi disemua angkatan.
10. Semua pihak yang tidak dapat disebutkan satu persatu.

Dengan harapan bahwa penelitian ini nantinya akan terus berlanjut dan berkembang kelak, penulis juga menerima segala kritik dan saran dari semua pihak

demi kesempurnaan skripsi ini. Akhirnya penulis berharap, semoga skripsi ini dapat bermanfaat.

Jember, Januari 2019

Penulis



**DAFTAR ISI**

SKRIPSI .....	i
PERSEMBAHAN .....	ii
MOTTO .....	iii
PERNYATAAN .....	iv
PENGESAHAN PEMBIMBING.....	vi
PENGESAHAN PENGUJI .....	vii
RINGKASAN .....	viii
PRAKATA .....	ix
DAFTAR ISI .....	xi
DAFTAR TABEL .....	xv
DAFTAR GAMBAR .....	xvi
BAB 1. PENDAHULUAN .....	1
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	2
1.3. Tujuan .....	2
1.4. Manfaat .....	2
1.5. Batasan Masalah .....	3
1.6. Sistematika Penulisan .....	4
BAB 2. TINJAUAN PUSTAKA .....	5
2.1. Penelitian Terdahulu .....	5
2.2. Program Indonesia Pintar (PIP) .....	5
2.3. SDLC Waterfall .....	7
2.4. Metode Simple Additive Weighting (SAW) .....	9

2.3.1. Langkah-Langkah Perhitungan Menggunakan Metode <i>Simple Additive Weighting (SAW)</i> .....	9
2.4. Metode <i>Borda</i> .....	10
BAB 3. METODOLOGI PENELITIAN.....	11
3.1. Jenis Penelitian .....	11
3.2. Pengembangan Sistem .....	11
3.2.1. Analisis Kebutuhan .....	11
3.2.2. Desain Sistem .....	12
3.2.3. Pengkodean .....	13
3.2.4. Pengujian .....	13
3.3. Uji Penerapan Metode <i>Simple Additive Weighting (SAW)</i> dan <i>Borda</i> . 14	
BAB 4. PENGEMBANGAN SISTEM .....	15
4.1. Analisis Kebutuhan Sistem .....	15
4.1.1. Kebutuhan Fungsional dan Non-Fungsional Sistem .....	17
4.2. Desain Sistem .....	17
4.2.1. <i>Use Case</i> Diagram.....	18
4.2.2. Skenario <i>Use Case</i> .....	20
4.2.3. <i>Sequence</i> Diagram .....	29
4.2.4. <i>Activity Diagram</i> .....	31
4.2.5. <i>Class Diagram</i> .....	38
4.2.6. <i>Entity Relationship Diagram (ERD)</i> .....	39
4.3. Implementasi Perancangan .....	39
4.4. Pengujian Sistem .....	40
4.4.3. Pengujian <i>Black Box</i> .....	40
BAB 5. HASIL DAN PEMBAHASAN.....	42

5.1. Hasil Implementasi <i>Coding</i> pada Sistem Informasi Rekomendasi Penerima Program Indonesia Pintar (PIP) .....	42
5.1.1. Tampilan Halaman Depan.....	42
5.1.2. Tampilan Halaman <i>Login</i> .....	43
5.1.3. Tampilan Halaman <i>Dashboard</i> .....	43
5.1.4. Halaman Pengguna.....	45
5.1.5. Halaman Pendaftar .....	46
5.1.6. Halaman Kriteria .....	46
5.1.7. Halaman Kuota Pendaftar .....	47
5.1.8. Halaman Ranking Pendaftar .....	47
5.1.9. Halaman Validasi .....	48
5.1.10. Halaman Hasil Rekomendasi .....	49
5.2. Penerapan Metode <i>Simple Additive Weighting</i> dan <i>Borda</i> pada Proses Seleksi Siswa Penerima Program Indonesia Pintar (PIP) .....	50
5.2.1. Penerapan Metode <i>Simple Additive Weighting</i> .....	50
5.2.2. Penerapan Metode <i>Borda</i> .....	51
5.3. Pengujian Perhitungan Metode <i>Simple Additive Weighting</i> dan <i>Borda</i> pada Sistem Informasi Rekomendasi Penerima Program Indonesia Pintar (PIP) .....	53
5.3.1. Perhitungan Manual .....	53
5.3.2. Perhitungan Sistem .....	61
5.3.3. Uji Akurasi Penerapan Metode <i>Simple Additive Weighting</i> dan <i>Borda</i> pada Proses Seleksi Penerima Program Indonesia Pintar .....	63
5.4. Pengaruh Pemberian Bobot Kriteria Metode <i>Simple Additive Weighting</i> .....	66

5.5. Pembahasan Penerapan Metode <i>Simple Additive Weighting</i> pada Seleksi Penerima Program Indonesia Pintar .....	68
5.6. Pembahasan Pengembangan Sistem Informasi Rekomendasi Penerima Program Indonesia Pintar .....	69
5.6.1. Pembahasan Tahap Analisis Kebutuhan .....	69
5.6.2. Pembahasan Tahap Desain Sistem .....	70
5.6.3. Pembahasan Tahap Implementasi Perancangan.....	71
5.6.4. Pembahasan Tahap Pengujian .....	71
BAB 6. PENUTUP .....	73
6.1. Kesimpulan .....	73
6.2. Saran .....	73
DAFTAR PUSTAKA .....	74
Lampiran .....	76

**DAFTAR TABEL**

Tabel 4.1. Data Kriteria .....	15
Tabel 4.2. Pilihan Kriteria .....	16
Tabel 4.3. Definisi Aktor .....	19
Tabel 4.4. Definisi Use Case .....	19
Tabel 4.5. Skenario Use Case Menentukan Jumlah Kuota .....	23
Tabel 4.6. Skenario Use Case Guru Bk Mengelola Ranking Pendaftar.....	24
Tabel 4.7. Skenario Use Case Bendahara Mengelola Ranking Pendaftar .....	25
Tabel 4.8. Skenario Use Case Validasi Hasil Rekomendasi .....	27
Tabel 4.9. Skenario Use Case Melihat Hasil Rekomendasi .....	29
Tabel 4.10. Pengujian Black Box Fitur Menentukan Jumlah Kuota.....	40
Tabel 5.1. Data dan Nilai Pendaftar .....	54
Tabel 5.2. Nilai Normalisasi Data Pendaftar .....	55
Tabel 5.3. Hasil Perkalian Nilai Normalisasi dengan Bobot Kriteria .....	57
Tabel 5.4. Hasil Akhir Perhitungan Metode SAW .....	58
Tabel 5.5. Ranking Guru BK dan Bendahara .....	60
Tabel 5.6. Borda Point dan Borda Count .....	61
Tabel 5.7. Hasil Rekomendasi Berdasarkan Metode SAW .....	64
Tabel 5.8. Hasil Rekomendasi Berdasarkan Metode SAW dan Borda .....	65



## DAFTAR GAMBAR

Gambar 2.1. Tahapan SDLC Model Waterfall .....	8
Gambar 4.1. Use Case Diagram .....	18
Gambar 4.2. Activity Diagram Menentukan Jumlah Kuota .....	34
Gambar 4.3. Activity Diagram (Guru BK) Mengelola Data Ranking .....	35
Gambar 4.4. Activity Diagram (Bendahara) Mengelola Data Ranking .....	36
Gambar 4.5. Activity Diagram Validasi Hasil Rekomendasi .....	37
Gambar 4.6. Activity Diagram Pengguna Melihat Hasil Rekomendasi .....	38
Gambar 4.7. Entity Relationship Diagram (ERD) .....	39
Gambar 5.1. Tampilan Halaman Depan .....	42
Gambar 5.2. Tampilan Halaman Login .....	43
Gambar 5.4. Tampilan Halaman Dashboard BK .....	44
Gambar 5.3. Tampilan Halaman Dashboard Admin .....	44
Gambar 5.5. Tampilan Halaman Dashboard Bendahara .....	45
Gambar 5.6. Tampilan Halaman Dashboard Ketua .....	45
Gambar 5.7. Tampilan Halaman Pengguna .....	46
Gambar 5.8. Tampilan Halaman Pendaftar .....	46
Gambar 5.9. Tampilan Halaman Kriteria .....	47
Gambar 5.10. Tampilan Halaman Kuota Pendaftar .....	47
Gambar 5.11. Tampilan Halaman Ranking Pendaftar (BK) .....	48
Gambar 5.12. Tampilan Halaman Ranking Pendaftar (Bendahara) .....	48
Gambar 5.13. Tampilan Halaman Validasi .....	49
Gambar 5.14. Tampilan Halaman Hasil Rekomendasi .....	49
Gambar 5.15. Penerapan Metode Simple Additive Weighting .....	50
Gambar 5.16. Penerapan Metode Borda .....	52
Gambar 5.17. Hasil Perhitungan SAW Sistem .....	62
Gambar 5.18. Hasil Perhitungan Borda Sistem .....	63
Gambar 5.19. Bobot Awal .....	67
Gambar 5.20. Hasil Awal .....	67
Gambar 5.21. Bobot Setelah Diubah .....	68

Gambar 5.22. Hasil Setelah Perubahan Bobot ..... 68



## **BAB 1. PENDAHULUAN**

### **1.1. Latar Belakang**

Pendidikan merupakan hak bagi setiap warga negara Indonesia sesuai dengan yang tertuang pada Undang-Undang Dasar Negara Republik Indonesia Tahun 1945 pasal 31 ayat 1 (Rubianto, 2016). Demi mendukung hal ini pemerintah mengadakan program wajib belajar 12 tahun. Namun, biaya yang diperlukan untuk menempuh pendidikan 12 tahun tidaklah sedikit, sehingga masih banyak anak-anak usia sekolah yang tidak bersekolah maupun putus sekolah karena kekurangan biaya. Oleh sebab itu, pemerintah senantiasa memberikan bantuan salah satunya adalah Program Indonesia Pintar (PIP) di sekolah mulai dari tingkat SD sampai dengan tingkat SMA sederajat.

Program Indonesia Pintar (PIP) merupakan penyempurnaan dari program Bantuan Siswa Miskin (BSM) dalam mendukung siswa dari keluarga miskin untuk memperoleh pendidikan. Bantuan ini diberikan oleh pemerintah dalam bentuk sejumlah uang tunai pada anak usia sekolah (6-21 tahun) yang berasal dari keluarga miskin dan rentan miskin sesuai dengan kriteria yang telah ditetapkan. Kriteria-kriteria tersebut meliputi: memiliki Kartu Indonesia Pintar (KIP)/Kartu Keluarga Sejahtera (KKS)/Kartu Perlindungan Sosial (KPS), siswa yang orang tuanya terdaftar sebagai peserta PKH (Program Keluarga Harapan), dan sebagainya (Petunjuk Teknis Program Indonesia Pintar (PIP), 2017).

Terdapat dua penyelenggara Program Indonesia Pintar yaitu Kementerian Pendidikan dan Kebudayaan (Kemendikbud) serta Kementerian Agama (Kemenag). Namun, dalam penyelenggaraan PIP terdapat pembatasan kuota dikarenakan adanya keterbatasan dana yang disediakan. Oleh sebab itu, untuk mendapatkan PIP para siswa yang mendaftar akan diseleksi terlebih dahulu. Pada tahap seleksi inilah terkadang masih ditemukan masalah, salah satunya adalah masih adanya penerima PIP yang kurang tepat sasaran. Hal ini terjadi karena terdapat banyak kriteria yang harus dipenuhi sehingga menyebabkan adanya kesulitan dalam memprioritaskan siswa calon penerima Program Indonesia Pintar. (P., Hamdana, & Fahreza, 2018). Oleh karena itu, diperlukan suatu sistem

informasi yang bisa membantu merekomendasikan siswa yang diprioritaskan untuk mendapatkan PIP.

Sistem informasi rekomendasi penerima Program Indonesia Pintar (PIP) yang akan dibangun pada penelitian ini digunakan untuk membantu merekomendasikan penerima Program Indonesia Pintar (PIP) sesuai dengan kriteria-kriteria yang telah ditentukan. Metode yang digunakan untuk membantu merekomendasikan siswa yang diprioritaskan menerima Program Indonesia Pintar (PIP) adalah *Simple Additive Weighting* (SAW) dan *Borda*. Metode *Simple Additive Weighting* (SAW) digunakan untuk meranking siswa pendaftar PIP berdasarkan bobot masing-masing kriteria, kemudian data tersebut akan dipilih (*voting*) menggunakan metode *Borda* oleh pihak-pihak yang terkait sehingga menghasilkan luaran berupa daftar siswa yang direkomendasikan mendapatkan Program Indonesia Pintar (PIP).

## 1.2. Rumusan Masalah

Berdasarkan permasalahan yang diuraikan pada latar belakang, maka rumusan masalah dalam penelitian ini adalah bagaimana cara menentukan siswa yang diprioritaskan untuk menerima bantuan Program Indonesia Pintar?

## 1.3. Tujuan

Tujuan dari penelitian ini adalah menggunakan Metode *Simple Additive Weighting* dan *Borda* untuk menentukan siswa yang diprioritaskan menerima bantuan Program Indonesia Pintar.

## 1.4. Manfaat

Penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

### 1. Manfaat bagi Peneliti

Dapat melatih kemampuan dan menerapkan ilmu pengetahuan yang telah diperoleh di Program Studi Sistem Informasi untuk membantu perusahaan atau masyarakat umum.

### 2. Manfaat bagi objek penelitian

Penelitian ini dapat membantu merekomendasikan siswa yang berhak mendapatkan Program Indonesia Pintar (PIP) berdasarkan kriteria-kriteria yang telah ditentukan.

### 1.5. Batasan Masalah

Agar tidak terjadi penyimpangan dalam proses penelitian dan pembangunan Sistem Informasi Rekomendasi Penerima Program Indonesia Pintar (PIP) menggunakan Metode *SimpleAdditiveWeighting* (SAW) dan *Borda* maka ditetapkan beberapa batasan permasalahan. Adapun batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Sistem informasi yang dibangun memberikan luaran berupa daftar siswa yang direkomendasikan menerima Program Indonesia Pintar (PIP).
2. Metode yang digunakan untuk merekomendasikan siswa yang berhak mendapatkan Program Indonesia Pintar (PIP) adalah *SimpleAdditive Weighting* (SAW) dan *Borda*
3. Kriteria yang digunakan meliputi: kepemilikan kartu, kepemilikan kendaraan, pekerjaan ayah, pekerjaan ibu, penghasilan orang tua, dan jumlah saudara.

## 1.6. Sistematika Penulisan

Adapun sistematika penulisan skripsi ini adalah:

### 1. Pendahuluan

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

### 2. Tinjauan Pustaka

Bab ini memaparkan tinjauan tentang landasan materi yang berkaitan dengan masalah dalam penelitian.

### 3. Metodologi Penelitian

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

### 4. Perancangan Sistem

Bab ini berisi uraian tentang perancangan desain sistem. Perancangan sistem dimulai dari analisis kebutuhan fungsional dan non-fungsional sistem, kemudian merancang *businessproces, use case diagram, scenario, sequence diagram, activity diagram, class diagram, relationship diagram (ERD)*.

### 5. Hasil dan Pembahasan

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

### 6. Penutup

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

## BAB 2. TINJAUAN PUSTAKA

Pada bagian ini dipaparkan tinjauan yang berkaitan dengan masalah yang dibahas serta kajian teori yang berkaitan dengan masalah.

### 2.1. Penelitian Terdahulu

Penggunaan Metode *Simple Additive Weighting* menentukan siswa yang berhak mendapatkan bantuan sudah pernah dilakukan. Salah satunya oleh Oktovantua TP Butar Butar pada April 2015 dengan judul “Sistem Pendukung Keputusan Rekomendasi Penerima Bantuan Siswa Miskin (BSM) dengan Metode *Simple Additive Weighting* (SAW)”. Penelitian ini membahas tentang penerapan Metode *Simple Additive Weighting* membantu panitia seleksi BSM dalam menentukan siswa yang berhak memperoleh Bantuan Siswa Miskin. Kriteria yang digunakan pada penelitian ini antara lain: semester, penghasilan orang tua, rata-rata nilai semester, jumlah tanggungan orang tua, nilai ekstra, dan prestasi (Butar Butar, 2015). Hasil dari penelitian yang dilakukan tersebut adalah sistem pendukung keputusan yang bisa membantu pihak sekolah untuk menyeleksi siswa yang berhak mendapatkan Bantuan Siswa Miskin.

Penelitian berikutnya dengan judul “*Group Decision Support System* Pembelian Rumah dengan menggunakan Analytical Hierarchy Process (AHP) dan Borda” dilakukan oleh Standy Oei pada Mei 2013. Penelitian ini membahas tentang penerapan Metode AHP dan Borda untuk membantu suatu keluarga menentukan pilihan rumah yang akan dibeli. Kriteria yang digunakan pada penelitian ini antara lain: lokasi, harga, fasilitas, dan *development* (Oei, 2013). Hasil dari penelitian ini adalah sistem pendukung keputusan grup yang bisa menggabungkan hasil perhitungan persepsi yang berbeda dari masing-masing anggota keluarga calon pembeli rumah.

### 2.2. Program Indonesia Pintar (PIP)

Program Indonesia Pintar (PIP) adalah penyempurnaan dari program Bantuan Siswa Miskin (BSM). Program Indonesia Pintar merupakan pemberian bantuan tunai pendidikan kepada anak usia sekolah (umur 6-21 tahun) yang berasal

dari keluarga pemegang Kartu Keluarga Sejahtera (KKS) atau yang memenuhi kriteria yang telah ditetapkan sebelumnya. Program ini ditandai dengan pemberian Kartu Indonesia Pintar (KIP). KIP berfungsi sebagai identitas/penanda untuk mendapatkan manfaat Program Indonesia Pintar dan hal ini hanya akan diperoleh apabila anak pemegang KIP tersebut mendaftar di sekolah/madrasah, pondok pesantren, Kelompok Belajar (Kejar Paket A/B/C), lembaga pelatihan atau kursus (Petunjuk Teknis Program Indonesia Pintar (PIP), 2017).

Terdapat dua penyelenggara Program Indonesia Pintar yaitu Kementerian Pendidikan dan Kebudayaan (Kemendikbud) dan Kementerian Agama (Kemenag). Penyelenggara yang pertama yaitu melalui Kementerian Pendidikan dan Kebudayaan (Kemendikbud) jalur ini diperuntukkan bagi siswa-siswi yang bersekolah di sekolah umum (SD, SMP, dan SMA), kejar paket, atau lembaga pelatihan/kursus. Sedangkan jalur yang kedua melalui Kementerian Agama (Kemenag) jalur ini diperuntukkan bagi siswa-siswi yang bersekolah di madrasah (MI, MTs, dan MA) atau pondok pesantren.

Sasaran PIP di madrasah adalah para siswa di MI (Madrasah Ibtidaiyah), MTs (Madrasah Tsanawiyah), dan MA (Madrasah Aliyah) baik negeri maupun swasta yang telah memenuhi kriteria yang telah ditetapkan. Kriteria-kriteria yang harus dipenuhi agar seorang siswa bisa mendapatkan Program Indonesia Pintar antara lain (Petunjuk Teknis Program Indonesia Pintar (PIP), 2017):

- a. Siswa penerima Kartu Indonesia Pintar (PIP).
- b. Siswa yang berasal dari keluarga penerima Kartu Sejahtera/Kartu Perlindungan Sosial (KKS/KPS) atau peserta Program Keluarga Harapan (PKH) tetapi belum memiliki Kartu Indonesia Pintar.
- c. Apabila masih terdapat sisa kuota dan anggaran, dapat diberikan kepada siswa madrasah yang orang tuanya tidak mampu berdasarkan data yang ada di *Educational Management Information System* yang dikirim Kementerian Agama Pusat dan dibuktikan dengan Surat Keterangan Tidak Mampu (SKTM) dari Desa/Kelurahan.
- d. Siswa dari keluarga tidak mampu yang berasal dari Provinsi Papua dan Papua Barat dapat diprioritaskan menerima manfaat PIP tanpa memiliki



KIP/KKS/KPS atau peserta program PKH dibuktikan dengan SKRTM/SKTM/SKMM dari Kelurahan/Desa/Madrasah.

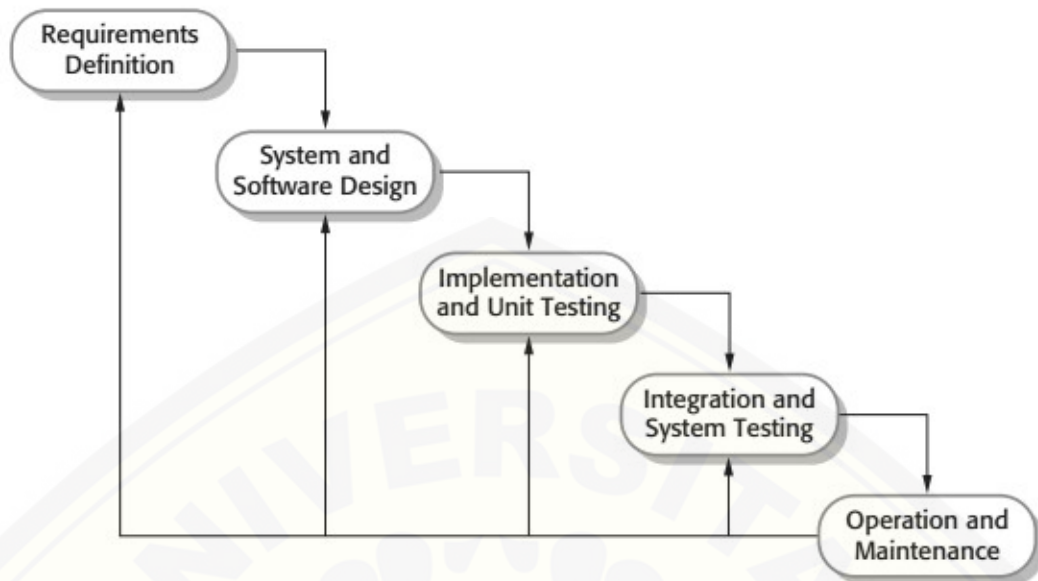
- e. Berada pada usia sekolah yakni 6 – 21 tahun. Bagi anak usia sekolah (6 – 21 tahun) penerima KIP yang tidak terdaftar di madrasah (putus sekolah) untuk mendapatkan manfaat Program Indonesia Pintar harus mendaftarkan diri kembali ke madrasah terlebih dahulu.

Besar bantuan PIP yang diberikan berbeda pada setiap jenjangnya. Bagi setiap siswa yang berhak mendapatkan PIP sesuai kriteria yang telah ditentukan bantuan yang diterima adalah sebesar (Petunjuk Teknis Program Indonesia Pintar (PIP), 2017):

1. untuk siswa Madrasah Ibtidaiyah:
  - a. Rp. 225.000, /siswa/semester, atau
  - b. Rp. 450.000,-/siswa/tahun
2. untuk siswa Madrasah Tsanawiyah:
  - a. Rp. 375.000,-/siswa/semester, atau
  - b. Rp. 750.000,-/siswa/tahun
3. untuk siswa Madrasah Aliyah:
  - a. Rp. 500.000,-/siswa/semester, atau
  - b. Rp. 1000.000,-/siswa/tahun

### 2.3. SDLC Waterfall

SDLC (*System Development Life Cycle*) merupakan salah satu metode untuk mengembangkan perangkat lunak/sistem yang mempunyai ciri khas berupa pengerjaan setiap fase harus diselesaikan terlebih dahulu sebelum melanjutkan ke fase berikutnya. Hal ini menghasilkan pengerjaan fase yang lebih fokus dan maksimal dikarenakan tidak adanya pengerjaan secara paralel. (Nugraha, Syarif, & Dharmawan, 2018), Rossa A.S dan Shalahuddin (2013 dalam Kaunen dan Arizona 2017:106) menjelaskan bahwa Metode *Waterfall* merupakan model pengembangan perangkat lunak dengan pendekatan alur hidup secara terurut dimulai dari tahap analisis, pengkodean, pengujian, dan tahap pendukung/support. Tahapan SDLC Model *Waterfall* lebih jelasnya bisa dilihat pada gambar 2.1.



Gambar 2.1. Tahapan SDLC Model *Waterfall*

Sumber: Sommerville (2011:30 dalam Rosyidah dkk., 2015:45)

Adapun tahapan Metode *Waterfall* diuraikan sebagai berikut (Dermawan & Hartini, 2017):

1. Analisis Kebutuhan  
Tahap ini dimaksudkan untuk menspesifikasikan kebutuhan perangkat lunak seperti apa yang dibutuhkan oleh user.
2. Desain  
Tahap desain ini menerjemahkan kebutuhan perangkat lunak yang diperoleh dari tahap analisis kebutuhan menjadi representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.
3. Pembuatan Kode Program  
Pembuatan kode program merupakan tahap untuk mengimplementasikan desain yang telah dibuat ke dalam bahasa pemrograman. Hasil dari tahap ini adalah aplikasi/*software* sesuai dengan desain yang telah dibuat.
4. Pengujian  
Pengujian ini bertujuan untuk mengetahui apakah keluaran sistem yang telah dibuat sudah sesuai dengan kriteria kebutuhan dari pengguna serta untuk meminimalisir terjadinya *error*
5. Pemeliharaan

Ketika perangkat lunak sudah diserahkan ke *user*, masih dimungkinkan terjadinya kesalahan atau terdapat *error*. Tahap pemeliharaan berfungsi untuk mengatasi *error* yang mungkin masih terjadi yang tidak ditemukan ketika tahap pengujian dilakukan.

## 2.4. Metode Simple Additive Weighting (SAW)

Metode *Simple Additive Weighting* dikenal juga sebagai metode penjumlahan berbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut (Fisburn, 1967). Metode SAW mengharuskan pembuat keputusan untuk menentukan bobot pada setiap atribut. Skor total untuk alternatif diperoleh dengan cara menjumlahkan seluruh hasil perkalian antara rating (yang dapat dibandingkan lintas atribut) dan bobot tiap atribut (Butar Butar, 2015). Metode SAW membutuhkan normalisasi matriks keputusan ( $X$ ) kesuatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada (T., Yanosma, & Anggriani, 2016).

### 2.3.1 Langkah-Langkah Perhitungan Menggunakan Metode *Simple Additive Weighting* (SAW)

Penyelesaian permasalahan pengambilan keputusan menggunakan metode *Simple Additive Weighting* harus dilakukan dalam beberapa langkah (Butar Butar, 2015), yaitu:

1. Menentukan kriteria-kriteria yang akan dijadikan acuan dalam pengambilan keputusan, yaitu  $C_i$ .
2. Menentukan rating kecocokan setiap alternatif pada setiap kriteria.
3. Membuat matriks keputusan berdasarkan kriteria ( $C_i$ ), kemudian melakukan normalisasi matriks berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut keuntungan maupun atribut biaya) sehingga diperoleh matriks ternormalisasi  $R$ .
4. Hasil akhir diperoleh dari proses perankingan yaitu penjumlahan dari perkalian matriks ternormalisasi  $R$  dengan vektor bobot sehingga diperoleh nilai terbesar yang dipilih sebagai alternatif terbaik ( $A_i$ ) sebagai solusi.

Persamaan yang digunakan untuk melakukan normalisasi tersebut adalah:

$$r_{ij} = \begin{cases} \frac{X_{ij}}{\text{MAX}_i(X_{ij})} & \text{Jika } j \text{ adalah atribut keuntungan} \\ \frac{\text{MIN}_i(X_{ij})}{X_{ij}} & \text{Jika } j \text{ adalah atribut biaya} \end{cases} \quad (1)$$

Dimana:

- $r_{ij}$  = rating kinerja ternormalisasi.  
 $\text{Max}_{ij}$  = nilai maksimum dari setiap baris dan kolom.  
 $\text{Min}_{ij}$  = nilai minimum dari setiap baris dan kolom.  
 $X_{ij}$  = baris dan kolom dari matriks.

Dengan  $r_{ij}$  adalah rating kinerja ternormalisasi dari alternatif  $A_i$  pada atribut  $C_j$ ;  $i=1,2,\dots,m$  dan  $j=1,2,\dots,n$ . Nilai preferensi untuk setiap alternatif ( $V_i$ ) diberikan sebagai:

$$V_i = \sum_{j=1}^n W_j R_{ij} \quad (2)$$

Keterangan :

- $V_i$  : Rangking untuk setiap alternatif.  
 $W_j$  : Nilai bobot dari setiap kriteria.  
 $R_{ij}$  : Nilai rating kinerja ternormalisasi.

Nilai  $V_i$  yang lebih besar mengindikasikan bahwa alternatif  $A_i$  lebih terpilih.

#### 2.4. Metode *Borda*

*Borda* merupakan suatu metode *voting* yang digunakan pada pengambilan keputusan kelompok untuk pemilihan *single winner* maupun *multiple winner*. *Borda* menentukan pemenang dengan memberikan sejumlah nilai tertentu untuk masing-masing alternatif. Selanjutnya, pemenang akan ditentukan oleh banyaknya jumlah nilai yang dikumpulkan alternatif. (Sari, Santoso, & Ernawati, 2014)

Langkah-langkah untuk perhitungan dengan metode *borda* (Oei, 2013), yaitu:

1. Setiap pengambil keputusan memberikan  $n-1$  untuk alternatif pilihan pertama, nilai  $n-2$  untuk alternatif pilihan kedua, ..., dan nilai 0 untuk alternatif pilihan terakhir.
2. Alternatif dengan nilai total tertinggi adalah pemenangnya.

### BAB 3. METODOLOGI PENELITIAN

Bab ini menjelaskan tentang gambaran tahapan yang sistematis yang dilakukan untuk menganalisa data untuk menjawab perumusan masalah sehingga dapat mencapai tujuan sebenarnya dari penelitian. Pada metodologi penelitian akan dijelaskan tentang tahapan dari penelitian membangun Sistem Informasi Rekomendasi Penerima Program Indonesia Pintar (PIP) Menggunakan Metode *Simple Additive Weighing* dan *Borda*

#### 3.1. Jenis Penelitian

Jenis penelitian yang dilakukan merupakan penelitian kuantitatif. Penelitian ini termasuk ke dalam penelitian kuantitatif karena di dalamnya terdapat proses perhitungan. Proses perhitungan yang ada pada penelitian ini adalah proses perekomendasi siswa berhak mendapatkan PIP atau tidak menggunakan Metode SAW dan *Borda*

#### 3.2. Pengembangan Sistem

Penelitian mengenai pembangunan sistem informasi rekomendasi penerima Program Indonesia Pintar (PIP) ini menggunakan pendekatan *System Development Lifecycle* (SDLC) model *Waterfall* Model *Waterfall* memiliki tahapan yang sistematis yaitu tahap analisis kebutuhan, desain sistem, penulisan kode program, pengujian program, dan penerapan serta pemeliharaan.

##### 3.2.1 Analisis Kebutuhan

Tahapan analisis kebutuhan pada penelitian yang dilakukan dimulai dengan tahap pengumpulan data meliputi observasi dan wawancara. Tahapan selanjutnya yaitu tahap pengolahan data dan gambaran umum sistem yang akan dibangun.

1. Tahap Pengumpulan Data.
  - a. Observasi.

Observasi merupakan cara pengumpulan data dengan mengadakan pengamatan langsung terhadap objek yang diteliti dan mengadakan pencatatan secara sistematis dalam suatu periode tertentu. Observasi bertujuan untuk

mendapatkan data berdasarkan kondisi objek dilapangan secara akurat dan variabel-variabel yang berpengaruh terhadap objek yang diteliti.

b. Wawancara.

Wawancara merupakan cara pengumpulan data dengan mengadakan tanya jawab terkait objek penelitian terhadap narasumber terkait.

2. Tahap Pengolahan Data

Tahap pengolahan data dimulai dengan menelaah data secara keseluruhan yang telah diperoleh dari tahap pengumpulan data. Data yang telah terkumpul digunakan untuk mengidentifikasi kebutuhan fungsional dan kebutuhan non-fungsional sistem.

3. Gambaran Umum Sistem

Sistem yang akan dibangun merupakan sistem informasi rekomendasi penerima Program Indonesia Pintar (PIP) menggunakan Metode *Simple Additive Weighting* (SAW) dan *Borda* Sistem ini berbasis *web* dan dibangun untuk membantu kegiatan seleksi siswa penerima PIP. Metode *Simple Additive Weighting* (SAW) digunakan untuk meranking siswa pendaftar PIP berdasarkan bobot masing-masing kriteria, kemudian data tersebut akan dipilih (*voting*) menggunakan metode *Borda* oleh pihak-pihak yang terkait sehingga menghasilkan luaran berupa daftar siswa yang direkomendasikan mendapatkan Program Indonesia Pintar (PIP).

### 3.2.2 Desain Sistem

Proses desain sistem yang akan dibangun menggunakan *Unified Modeling Language* (UML). UML digunakan karena sudah menggunakan konsep *Object Oriented Design* yang tentunya akan sangat memudahkan *developer* untuk membangun sebuah sistem. Diagram UML yang akan digunakan antara lain:

a. *Use Case Diagram*

*Use case diagram* menggambarkan fungsionalitas dari sebuah sistem dan interaksi antara *user* dengan sistem untuk melakukan suatu pekerjaan tertentu. *use case diagram* juga menggambarkan hak akses dari aktor.

### b. Scenario

*Scenario* digunakan untuk menjelaskan fitur yang ada pada *use case diagram*. Scenario menjelaskan secara detail aksi yang dilakukan oleh aktor dan reaksi sistem.

### c. Sequence Diagram

*SequenceDiagram* (diagram urutan) adalah suatu diagram yang memperlihatkan atau menampilkan interaksi-interaksi antar objek didalam sistem yang disusun pada sebuah urutan atau rangkaian waktu. Interaksi antar objek tersebut termasuk pengguna, *display*, dan sebagainya berupa pesan atau *message*

### d. Activity Diagram

*Activity Diagram* digunakan untuk menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir.

### e. Class Diagram

*Class Diagram* menggambarkan struktur dan deskripsi *class package* dan objek beserta hubungan satu sama lain seperti pewarisan, asosiasi, dan lain-lain.

### f. Entity Relationship Diagram

*Entity Relationship Diagram* menggambarkan struktur *database* yang akan dibangun pada sistem.

## 3.2.3 Pengkodean

Tahapan ini merupakan tahapan untuk mengimplementasikan desain yang sudah dirancang ke dalam kode program. Pada penelitian ini sistem dibangun berbasis web menggunakan bahasa pemrograman PHP ( *Hypertext Preprocessor* dengan *framework laravel* serta *MariaDB* sebagai pengelola basis datanya.

## 3.2.4 Pengujian

Setelah tahap pengkodean selesai, tahap pengujian dilakukan guna mengetahui apakah terdapat *error* pada sistem yang dibuat. Pengujian yang dilakukan dalam penelitian ini adalah *black box testing* *Black box testing* merupakan pengujian yang menitik beratkan pada uji fungsionalitas dari program yang dibuat. Pengujian ini digunakan untuk menemukan ketidaksesuaian program dengan kebutuhan fungsional maupun non-fungsional. Hal yang perlu dilakukan

dalam pengujian ini adalah menguji *interface* dari program untuk memastikan suatu masukan di proses oleh sistem dengan benar dan menghasilkan luaran yang sesuai dengan perancangan.

### 3.3. Uji Penerapan Metode *Simple Additive Weighting (SAW)* dan *Borda*

Penerapan metode *Simple Additive Weighting (SAW)* dan *Borda* pada kegiatan seleksi penerima Program Indonesia Pintar (PIP) akan diuji menggunakan pengujian akurasi. Rumus yang digunakan untuk menghitung akurasi tersebut adalah (Handayani & S., 2014):

$$Akurasi = \frac{\text{Jumlah Data Akurat}}{\text{Jumlah Data yang diuji}} \quad (1)$$



## BAB 4. PENGEMBANGAN SISTEM

Bab ini akan membahas tentang pengembangan sistem informasi rekomendasi penerima Program Indonesia Pintar (PIP) menggunakan metode *Simple Additive Weighting* (SAW) dan *Borda*. Tahap pengembangan sistem dilaksanakan berdasarkan metode *waterfall* dimulai dari analisis kebutuhan fungsional dan non-fungsional sistem, pembuatan desain sistem, penulisan kode program, kemudian pengujian sistem.

### 4.1. Analisis Kebutuhan Sistem

Tahapan awal dari model pengembangan sistem *waterfall* yang dilakukan adalah tahapan analisis. Tahapan analisis ini dilakukan untuk memperoleh kebutuhan-kebutuhan dari sistem yang dibangun, baik berupa kebutuhan fungsional maupun kebutuhan non-fungsional. Dimana hasil analisis tersebut sangat mempengaruhi fungsionalitas sistem yang dibangun untuk dapat digunakan sesuai dengan fungsi dan kebutuhan pengguna. Adapun data yang terkumpul dari proses analisis kebutuhan adalah data kriteria penerima PIP dan data siswa. Berikut ini beberapa data yang diperoleh dari hasil analisis kebutuhan.

#### 1. Data Kriteria

Pada tabel 4.1 akan dijelaskan mengenai kriteria yang digunakan untuk merekomendasikan penerima Program Indonesia Pintar.

Tabel 4.1. Data Kriteria

No.	Kriteria	Bobot
1.	Kepemilikan Kartu	30
2.	Kepemilikan Kendaraan	15
3.	Pekerjaan Ayah	10
4.	Pekerjaan Ibu	10
5.	Penghasilan Keluarga	15
6.	Jumlah Saudara	20

## 2. Data Pilihan Kriteria

Pada tabel 4.2 akan dijelaskan mengenai pilihan masing-masing kriteria yang digunakan untuk merekomendasikan penerima program Indonesia Pintar (PIP). Terdapat 28 data pilihan kriteria.

Tabel 4.2. Pilihan Kriteria

No.	Pilihan	Nilai
1	Memiliki KIP/KPS atau merupakan anggota PKH	100
2	Memiliki SKTM	90
3	Tidak Memiliki KIP/KPS dan bukan merupakan anggota PKH	1
4	Mobil atau Mobil dan Motor	1
5	Motor Lebih Dari 1	30
6	Motor	80
7	Pensiunan/Almarhum	100
8	Buruh atau Tukang atau Serabutan	90
9	Petani atau Peternak	75
10	Pegawai Swasta	25
11	Pedagang atau Pengusaha atau Wiraswasta	15
12	PNS	1
13	Pensiunan/Almarhum	100
14	Ibu Rumah Tangga	90
15	Pegawai Swasta	25
16	Pedagang atau Pengusaha atau Wiraswasta	15
17	Petani/Peternak	75
18	PNS	1
19	<= Rp. 500.000,-	100
20	Rp. 500.001,- Sampai Rp. 1.000.000,-	75
21	Rp.1.000.001,- Sampai Rp.2.000.000,-	50
22	Rp. 2.000.001,- Sampai Rp. 3.000.000,-	25
23	Lebih dari Rp. 3.000.001,-	1
24	4 Atau Lebih	100
25	3 saudara	75
26	2 saudara	<b>50</b>
27	1 saudara	25
28	Anak tunggal	1

### 3. Data Siswa

Data siswa yang akan digunakan dalam penelitian ini adalah data 20 siswa yang diambil dari kelas 7 dan kelas 8 Madrasah Abc dengan rincian 10 penerima PIP dan 10 bukan penerima PIP.

#### 4.1.1 Kebutuhan Fungsional dan Non-Fungsional Sistem

Kebutuhan fungsional sistem merupakan fitur-fitur inti yang harus dipenuhi oleh sistem agar sistem mampu difungsikan sesuai dengan tujuan dan kebutuhan pengguna. Kebutuhan fungsional dari sistem ini adalah:

1. Sistem mampu mengelola data pengguna meliputi mengubah dan melihat data pengguna.
2. Sistem mampu mengelola data siswa meliputi menambah, mengubah, dan melihat data siswa.
3. Sistem mampu mengelola data pendaftar meliputi menambah dan melihat data pendaftar, menentukan kuota penerima Program Indonesia Pintar, dan memberi ranking data pendaftar.
4. Sistem mampu mengelola data kriteria yang digunakan untuk merekomendasikan penerima Program Indonesia Pintar meliputi menambah, mengaktifkan, menonaktifkan, serta melihat data kriteria.
5. Sistem mampu menampilkan data siswa hasil rekomendasi penerima Program Indonesia Pintar menggunakan Metode *Simple Additive Weighting* (SAW) dan *Borda*

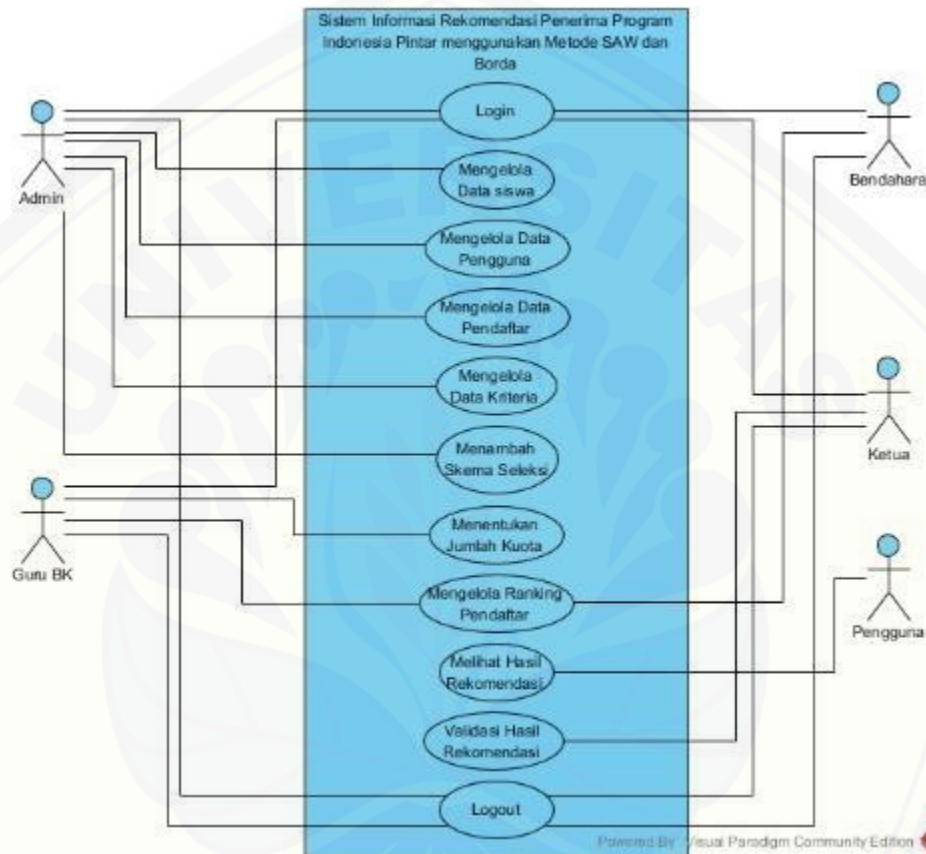
Sedangkan kebutuhan non-fungsional sistem pada penelitian ini adalah tampilan aplikasi yang sederhana dan sistem menggunakan *username* serta *password* untuk membatasi hak akses kepada sistem. (Sumber: hasil wawancara)

#### 4.2. Desain Sistem

Tahapan yang dilakukan setelah analisis kebutuhan sistem adalah tahap perencanaan pembangunan sistem yang digambarkan dengan desain sistem. Desain sistem dalam penelitian ini meliputi: *use case diagram*, *use case scenario*, *activity diagram*, *sequence diagram*, *class diagram*, dan *entity relationship diagram*.

### 4.2.1. Use Case Diagram

*Use case* diagram merupakan *blueprint* sistem yang menggambarkan fitur dan aktor yang dapat mengakses fitur sistem tersebut. *Use case diagram* pada sistem informasi rekomendasi penerima Program Indonesia Pintar ini dapat dilihat pada gambar 4.1.



Gambar 4.1. Use Case Diagram

Penjelasan mengenai definisi aktor dan definisi *use case* dalam *use case* diagram pada gambar 4.1 akan dijelaskan di bawah ini.

#### 1. Definisi Aktor

Definisi aktor merupakan penjelasan tentang aktor-aktor sebagai pengguna dari sistem informasi rekomendasi penerima Program Indonesia Pintar yang akan dibangun. Terdapat 4 aktor seperti yang dijelaskan pada tabel 4.3.

Tabel 4.3. Definisi Aktor

No.	Aktor	Definisi Tugas
1.	Admin	Mengelola data sistem secara keseluruhan meliputi data pengguna, data siswa, data pendaftar, menambah skema seleksi, serta data kriteria
2.	Guru BK	Guru BK memiliki hak akses menentukan jumlah kuota seleksi, memberi ranking pendaftar
3.	Bendahara	Bendahara memiliki hak akses untuk memberi ranking pendaftar
4.	Ketua	Ketua memiliki hak akses untuk memvalidasi hasil rekomendasi penerima PIP
5.	Pengguna	Pengguna memiliki hak akses untuk melihat hasil rekomendasi penerima Program Indonesia Pintar

## 2. Definisi *Use Case*

Definisi *use case* merupakan penjelasan dari masing-masing *use case* atau fitur-fitur dari sistem informasi rekomendasi penerima Program Indonesia Pintar yang akan dibangun. Definisi *use case* pada sistem ini dijelaskan pada tabel 4.4.

Tabel 4.4. Definisi Use Case

No.	<i>Use case</i>	Deskripsi
1.	<i>Login</i>	Menggambarkan proses autentifikasi masuk ke sistem
2.	Mengelola data pengguna	Menggambarkan proses melihat dan mengubah data pengguna
3.	Mengelola data siswa	Menggambarkan proses melihat, menambah, dan mengubah data siswa
4.	Mengelola data pendaftar	Menggambarkan proses melihat dan menambah data pendaftar

Dilanjutkan

Lanjutan

No.	<i>Use case</i>	Deskripsi
5.	Mengelola data kriteria	Menggambarkan proses melihat, menambah, dan mengaktifkan atau menonaktifkan data kriteria
6.	Menambah skema seleksi	Menggambarkan proses menambahkan data skema seleksi
7.	Menentukan jumlah kuota	Menggambarkan proses menentukan jumlah kuota penerima Program Indonesia Pintar
8.	Mengelola ranking pendaftar	Menggambarkan proses melihat dan mengubah data ranking pendaftar
9.	Validasi hasil rekomendasi	Menggambarkan proses memvalidasi hasil rekomendasi penerima PIP
10.	Melihat hasil rekomendasi	Menggambarkan proses melihat hasil rekomendasi penerima Program Indonesia Pintar

#### 4.2.2 Skenario *Use Case*

Skenario *use case* digunakan untuk menjelaskan alur dari sebuah sistem serta alur alternatif yang dilakukan oleh aktor yang menggunakan sistem informasi rekomendasi penerima Program Indonesia Pintar. Skenario *use case* sistem sesuai dengan *use case* program pada gambar 4.2.

##### 4.2.2.1. Skenario *Use Case Login*

Skenario *use case* login digunakan untuk menjabarkan alur dari proses *login* ke dalam sistem serta alur alternatif yang dilakukan oleh aktor yang menggunakan sistem. Aktor yang terdapat dalam skenario *use case* penelitian ini adalah *admin*, guru bk, bendahara, ketua, serta pengguna. Prekondisi dari skenario *use case login* adalah masing-masing aktor harus memiliki *e-mail* dan *password* yang telah terdaftar dalam sistem. Alur alternatif yang ada pada skenario ini adalah apabila *e-mail* dan *password* yang dimasukkan aktor salah serta apabila *field* pada form *login* ada yang kosong. Penjelasan lebih lengkap mengenai urutan aksi aktor

dan reaksi sistem pada skenario normal dan skenario alternatif skenario *use case login* terdapat pada lampiran A.1.

#### 4.2.2.2. Skenario *Use Case* Mengelola Data Pengguna

Skenario *use case* mengelola data pengguna digunakan untuk menjabarkan alur dari proses mengelola data pengguna didalam sistem serta alur alternatif yang digunakan oleh aktor yang menggunakan sistem. Aktor dari skenario *use case* mengelola data pengguna ini adalah admin. Prekondisi dari skenario *use case* ini adalah admin berhasil login kedalam sistem, sedangkan prakondisinya adalah data pengguna berhasil diubah. Alur alternatif yang terdapat pada skenario *use case* mengelola data pengguna ini ialah ketika mengklik tombol batal serta apabila *field* pada form ada yang kosong. Penjelasan lebih lengkap mengenai urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif skenario *use case* mengelola data pengguna terdapat pada lampiran A.2.

#### 4.2.2.3. Skenario *Use Case* Mengelola Data Siswa

Skenario *use case* mengelola data siswa digunakan untuk menjabarkan alur dari proses mengelola data siswa didalam sistem serta alur alternatif yang digunakan oleh aktor yang menggunakan sistem. Aktor dari skenario *use case* mengelola data siswa ini adalah admin. Prekondisi dari skenario *use case* ini adalah admin berhasil login kedalam sistem, sedangkan prakondisinya adalah data siswa berhasil diubah atau ditambah. Alur alternatif yang terdapat pada skenario *use case* mengelola data siswa ini ialah ketika mengklik tombol batal, apabila *field* pada form ada yang kosong, serta siswa sudah terdaftar. Penjelasan lebih lengkap mengenai urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif skenario *use case* mengelola data siswa terdapat pada lampiran A.3.

#### 4.2.2.4. Skenario *Use Case* Mengelola Data Pendaftar

Skenario *use case* mengelola data pendaftar digunakan untuk menjabarkan alur dari proses mengelola data pendaftar didalam sistem serta alur alternatif yang digunakan oleh aktor yang menggunakan sistem. Aktor dari skenario *use case* mengelola data pendaftar ini adalah admin. Prekondisi dari skenario *use case* ini adalah admin berhasil login kedalam sistem, sedangkan prakondisinya adalah data pendaftar berhasil ditambah. Alur alternatif yang terdapat pada skenario *use case*

mengelola data siswa ini ialah ketika *field* pada form ada yang kosong serta pendaftar sudah ada. Penjelasan lebih lengkap mengenai urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif skenario *use case* mengelola data pendaftar terdapat pada lampiran A.4.

#### **4.2.2.5. Skenario Use Case Mengelola Data Kriteria**

Skenario *use case* mengelola data kriteria digunakan untuk menjabarkan alur dari proses mengelola data pendaftar didalam sistem serta alur alternatif yang digunakan oleh aktor yang menggunakan sistem. Aktor dari skenario *use case* mengelola data kriteria ini adalah admin. Prekondisi dari skenario *use case* ini adalah admin berhasil login kedalam sistem, sedangkan prakondisinya adalah data kriteria berhasil dinonaktifkan/dinonaktifkan serta data kriteria berhasil ditambah. Alur alternatif yang terdapat pada skenario *use case* mengelola data kriteria ini ialah ketika *field* pada form ada yang kosong. Penjelasan lebih lengkap mengenai urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif skenario *use case* mengelola data kriteria terdapat pada lampiran A.5.

#### **4.2.2.6. Skenario Use Case Menambah Skema Seleksi**

Skenario *use case* menambah skema seleksi digunakan untuk menjabarkan alur dari proses menambah skema seleksi oleh aktor yang menggunakan sistem. Aktor dari skenario *use case* menambah skema seleksi ini adalah admin. Prekondisi dari skenario *use case* menambah skema seleksi ini adalah admin berhasil *login* kedalam sistem. Sedangkan prakondisinya adalah admin berhasil menambahkan skema seleksi. Penjelasan lebih lengkap mengenai skenario *use case* menambah skema seleksi bisa dilihat pada lampiran A.6.

#### **4.2.2.7. Skenario Use Case Menentukan Jumlah Kuota**

Skenario *use case* menentukan jumlah kuota digunakan untuk menjabarkan alur dari proses menentukan jumlah kuota didalam sistem serta alur alternatif yang digunakan oleh aktor yang menggunakan sistem. Aktor dari skenario *use case* mengelola data kriteria ini adalah guru bk. Prekondisi dari skenario *use case* ini adalah guru bk berhasil login kedalam sistem, sedangkan prakondisinya adalah data pendaftar sejumlah kuota siap diranking. Alur alternatif yang terdapat pada skenario *use case* mengelola data kriteria ini ialah ketika *field* pada form ada yang kosong



serta jumlah pendaftar belum memenuhi kuota. Penjelasan lebih lengkap mengenai urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif skenario *use case* menentukan jumlah kuota bisa dilihat pada tabel 4.5.

Tabel 4.5. Skenario Use Case Menentukan Jumlah Kuota

<b>Nama Use Case</b>	Menentukan Jumlah Kuota
<b>Aktor</b>	Guru BK
<b>Deskripsi Singkat</b>	Guru BK akan menentukan jumlah kuota
<b>Prekondisi</b>	Guru BK berhasil login ke dalam sistem
<b>Prakondisi</b>	Data pendaftar sejumlah kuota siap diranking
<b>Flow Events</b>	
<b>Skenario Normal : menentukan jumlah kuota</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Mengklik menu pendaftar	
	2. Menampilkan halaman skema seleksi yang berisi form skema seleksi dengan rincian: → Skema seleksi → Tombol <i>submit</i>
3. Memilih skema seleksi	
4. Mengklik tombol submit	
	5. Menampilkan halaman data pendaftar yang berisi tabel data pendaftar yang telah diinputkan beserta nilai perhitungan metode SAW dan tombol seleksi
6. Mengklik tombol seleksi	
	7. Menampilkan <i>moda</i> seleksi borda dengan <i>field</i> kuota penerima tipe <i>number</i> dan tombol proses ke seleksi
8. Mengisi jumlah kuota	
9. Mengklik tombol proses ke seleksi	
	10. Menampilkan halaman seleksi bk yang berisi tabel data pendaftar sejumlah kuota yang dimasukkan beserta dengan <i>dropdown</i> ranking
<b>Skenario Alternatif : isian data tidak lengkap</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
9a. Mengklik tombol proses ke seleksi	
	10a. <i>moda</i> seleksi borda menghilang

Dilanjutkan

Lanjutan

<b>Skenario Alternatif : jumlah pendaftar kurang dari kuota</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
9b. Mengklik tombol proses ke seleksi	
	10b. Menampilkan <i>flash warning</i> “Data Pendaftar Belum Cukup untuk Memenuhi Kuota! Silakan Hubungi Admin!”

#### 4.2.2.8. Skenario *Use Case* Mengelola Ranking Pendaftar

Skenario *use case* mengelola ranking pendaftar digunakan untuk menjabarkan alur dari proses mengelola ranking pendaftar didalam sistem serta alur alternatif yang digunakan oleh aktor yang menggunakan sistem. Aktor dari skenario *use case* mengelola ranking pendaftar ini adalah guru bk dan bendahara. Prekondisi dari skenario *use case* ini adalah guru bk dan bendahara berhasil login kedalam sistem, sedangkan prakondisinya adalah data pendaftar berhasil diberi ranking. Alur alternatif yang terdapat pada skenario *use case* mengelola ranking pendaftar ini ialah ketika *field* pada form ada yang kosong serta jumlah pendaftar belum memenuhi kuota. Penjelasan lebih lengkap mengenai urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif skenario *use case* mengelola ranking pendaftar bisa dilihat pada tabel 4.6 dan 4.7.

Tabel 4.6. Skenario Use Case Guru Bk Mengelola Ranking Pendaftar

<b>Nama Use Case</b>	Mengelola Ranking Pendaftar
<b>Aktor</b>	Guru BK
<b>Deskripsi Singkat</b>	Guru BK akan memberi ranking pendaftar
<b>Prekondisi</b>	Guru BK berhasil login ke dalam sistem
<b>Prakondisi</b>	Data pendaftar berhasil diberi ranking
<b>Flow Events</b>	
<b>Skenario Normal : melihat ranking pendaftar</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Mengklik menu ranking	
	2. Menampilkan halaman skema seleksi yang berisi form skema seleksi dengan rincian: → Skema seleksi → Tombol <i>submit</i>
3. Memilih skema seleksi	
4. Mengklik tombol submit	

Dilanjutkan

Lanjutan

<b>Flow Events</b>	
<b>Skenario Normal : melihat ranking pendaftar</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
	5. Menampilkan halaman data pendaftar yang berisi tabel data pendaftar sejumlah kuota periode tersebut terurut berdasarkan nilai metode SAW beserta <i>ranking</i> dan tombol ubah data
<b>Skenario Normal : mengubah ranking pendaftar</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
6. Mengklik tombol ubah data	
	7. Menampilkan halaman tambah ranking pendaftar yang berisi tabel pendaftar
8. Mengisi ranking pendaftar (pilih dari dropdown)	
	9. Menghapus pilihan ranking yang telah dipilih
10. Mengklik tombol <i>submit</i>	
	11. Menyimpan ke database
	12. Menampilkan halaman skema seleksi pendaftar dengan <i>flash notification</i> "berhasil memasukkan ranking pendaftar"
<b>Skenario Alternatif : isian data tidak lengkap</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
10a. Mengklik tombol <i>submit</i>	
	11a. Menampilkan span " <i>please fill out this field</i> "

Tabel 4.7. Skenario Use Case Bendahara Mengelola Ranking Pendaftar

<b>Nama Use Case</b>	Mengelola Ranking Pendaftar
<b>Aktor</b>	Bendahara
<b>Deskripsi Singkat</b>	Bendahara akan memberi ranking pendaftar
<b>Prekondisi</b>	Bendahara berhasil login ke dalam sistem
<b>Prakondisi</b>	Data pendaftar berhasil diberi ranking

Dilanjutkan

Lanjutan

<b>Flow Events</b>	
<b>Skenario Normal : melihat ranking pendaftar</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Mengklik menu ranking	
	2. Menampilkan halaman skema seleksi yang berisi form skema seleksi dengan rincian: → Skema seleksi → Tombol <i>submit</i>
3. Memilih skema seleksi	
4. Mengklik tombol submit	
<b>Skenario Normal : melihat ranking pendaftar</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
	5. Menampilkan halaman data pendaftar yang berisi tabel data pendaftar sejumlah kuota periode tersebut terurut berdasarkan nilai metode SAW beserta <i>ranking</i> dan tombol ubah data
<b>Skenario Normal : mengubah ranking pendaftar</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
6. Mengklik tombol ubah data	
	7. Menampilkan halaman tambah ranking pendaftar yang berisi table pendaftar serta <i>dropdown</i> ranking
8. Mengisi ranking pendaftar (pilih dari dropdown)	
	9. Mengganti nilai ranking sesuai dengan yang telah dipilih
10. Mengklik tombol <i>submit</i>	
	11. Menyimpan ke database
	12. Menampilkan halaman skema seleksi pendaftar dengan <i>flash notification</i> “Berhasil memasukkan ranking pendaftar”
<b>Skenario Alternatif : isian data tidak lengkap</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
10a. Mengklik tombol <i>submit</i>	
	11a. Menampilkan span “ <i>please fill out this field</i> ”

#### 4.2.2.9. Skenario *Use Case* Validasi Hasil Rekomendasi

Skenario *use case* validasi hasil rekomendasi digunakan untuk menjabarkan alur dari proses validasi hasil rekomendasi didalam sistem serta alur alternatif yang digunakan oleh aktor yang menggunakan sistem. Aktor dari skenario *use case* validasi hasil rekomendasi ini adalah ketua. Prekondisi dari skenario *use case* adalah ketua berhasil login kedalam sistem, sedangkan prakondisinya adalah data hasil rekomendasi penerima PIP berhasil divalidasi. Alur alternatif pada skenario *use case* validasi hasil rekomendasi ini ialah ketika ranking dari guru bk atau bendahara belum dimasukkan serta belum ada data. Penjelasan lebih lengkap mengenai urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif skenario *use case* validasi hasil rekomendasi bisa dilihat pada tabel 4.8.

Tabel 4.8. Skenario Use Case Validasi Hasil Rekomendasi

<b>Nama Use Case</b>	Validasi Hasil Rekomendasi
<b>Aktor</b>	Ketua
<b>Deskripsi Singkat</b>	Ketua akan memvalidasi hasil rekomendasi bk dan bendahara
<b>Prekondisi</b>	Ketua berhasil login ke dalam sistem
<b>Prakondisi</b>	Data hasil rekomendasi penerima PIP berhasil divalidasi
<b>Flow Events</b>	
<b>Skenario Normal : memvalidasi hasil rekomendasi</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Mengklik menu validasi	
	2. Menampilkan halaman skema seleksi yang berisi form skema seleksi dengan rincian: → Skema seleksi → Tombol <i>submit</i>
3. Memilih skema seleksi	
4. Mengklik tombol submit	

Dilanjutkan

Lanjutan

<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
	5. Menampilkan halaman validasi hasil rekomendasi yang berisi tabel data pendaftar yang telah direkomendasikan oleh bk dan bendahara dengan rincian nama, kelas, kriteria, poin borda, keterangan, serta tombol setuju semua bila data belum divalidasi
6. Mengklik Tombol Setujui Semua	
	7. Menyimpan ke database
	8. Menampilkan halaman validasi hasil rekomendasi yang berisi tabel data pendaftar yang telah direkomendasikan oleh bk dan bendahara dengan rincian nama, kelas, kriteria, poin borda, keterangan telah disetujui
<b>Skenario Alternatif : Ranking BK Belum dimasukkan</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
4a. Mengklik tombol submit	
	5a. Menampilkan halaman error "Ranking BK Belum Dimasukkan"
<b>Skenario Alternatif : Ranking Bendahara Belum dimasukkan</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
5b. Mengklik tombol submit	
	5b. Menampilkan halaman error "Ranking Bendahara Belum Dimasukkan"
<b>Skenario Alternatif : Belum Ada Data</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
4c. Mengklik tombol submit	
	5c. Menampilkan halaman validasi hasil rekomendasi dengan tulisan "Belum Ada Data untuk Periode Ini, Hubungi BK/Admin untuk Input Data"

#### 4.2.2.10. Skenario *Use Case* Melihat Hasil Rekomendasi

Skenario *use case* melihat hasil rekomendasi digunakan untuk menjabarkan alur dari proses melihat hasil rekomendasi didalam sistem serta alur alternatif yang digunakan oleh aktor yang menggunakan sistem. Aktor dari skenario *use case* melihat hasil rekomendasi ini adalah dan pengguna. Penjelasan lebih lengkap mengenai urutan aksi aktor dan reaksi sistem pada skenario *use case* melihat hasil rekomendasi bisa dilihat pada tabel 4.9.

Tabel 4.9. Skenario Use Case Melihat Hasil Rekomendasi

<b>Nama Use Case</b>	Melihat Hasil Rekomendasi
<b>Aktor</b>	Pengguna
<b>Deskripsi Singkat</b>	Pengguna akan melihat hasil rekomendasi
<b>Flow Events</b>	
<b>Skenario Normal : melihat hasil rekomendasi</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Mengklik menu hasil	2. Menampilkan halaman periode dan skema pengajuan yang berisi form periode dan skema pengajuan dengan rincian: → Skema seleksi → Periode pengajuan → Tombol <i>submit</i>
3. Memilih skema seleksi dan periode pengajuan	
4. Mengklik tombol cari	5. Menampilkan halaman hasil rekomendasi yang berisi table pendaftar yang disetujui menerima beasiswa dengan rincian nis, nama dan kelas, tombol <i>back</i> , serta <i>side menu</i> Home Sirpintar

#### 4.2.3. Sequence Diagram

*SequenceDiagram* merupakan diagram yang digunakan untuk menunjukkan interaksi yang terjadi antar objek didalam sistem yang disusun pada sebuah urutan dan rangkaian waktu. Sequence Diagram pada sistem informasi yang dibangun pada penelitian ini meliputi *sequence diagram* login, mengelola data pengguna, mengelola data siswa, mengelola data pendaftar, mengelola data kriteria, menentukan jumlah kuota, mengelola ranking pendaftar, validasi hasil rekomendasi, dan *sequence diagram* melihat hasil rekomendasi.

##### 4.2.3.1. Sequence Diagram Login

*Sequence diagram login* digunakan untuk menggambarkan skenario *use case login* secara teknis. *Sequence diagram login* class dan *method* yang digunakan dalam proses *login Sequence diagram login* dapat dilihat pada lampiran B.1.

#### 4.2.3.2. **Sequence Diagram Mengelola Data Pengguna**

*Sequence diagram* mengelola data pengguna digunakan untuk menggambarkan skenario *use case* mengelola data pengguna secara teknis. *Sequence diagram* mengelola data pengguna dapat dilihat pada lampiran B.2.

#### 4.2.3.3. **Sequence Diagram Mengelola Data Siswa**

*Sequence diagram* mengelola data siswa digunakan untuk menggambarkan skenario *use case* mengelola data siswa secara teknis. *Sequence diagram* mengelola data siswa berisi *class* dan *method* yang digunakan dalam proses mengelola data siswa. *Sequence diagram* mengelola data siswa diawali dengan admin yang telah login mengklik menu siswa untuk melihat data siswa yang telah ditambahkan. *Sequence diagram* mengelola data siswa dapat dilihat pada lampiran B.3.

#### 4.2.3.4. **Sequence Diagram Mengelola Data Pendaftar**

*Sequence diagram* mengelola data pendaftar digunakan untuk menggambarkan skenario *use case* mengelola data pendaftar secara teknis. *Sequence diagram* mengelola data pendaftar berisi *class* dan *method* yang digunakan dalam proses mengelola data pendaftar. *Sequence diagram* mengelola data pendaftar dapat dilihat pada lampiran B.4.

#### 4.2.3.5. **Sequence Diagram Mengelola Data Kriteria**

*Sequence diagram* mengelola data kriteria digunakan untuk menggambarkan skenario *use case* mengelola data kriteria secara teknis. *Sequence diagram* mengelola data kriteria berisi *class* dan *method* yang digunakan dalam proses mengelola data kriteria. Pada *sequence diagram* ini digambarkan bagaimana admin bisa melakukan pengelolaan pada data kriteria berupa melihat, mengubah, menambah, dan mengaktifkan/menonaktifkan data kriteria seleksi. *Sequence diagram* mengelola data kriteria dapat dilihat pada lampiran B.5.

#### 4.2.3.6. **Sequence Diagram Menambah Skema Seleksi**

*Sequence diagram* menambah skema seleksi digunakan untuk menggambarkan skenario *use case* menambah skema seleksi secara teknis. *Sequence diagram* menambah skema seleksi lebih jelasnya bisa dilihat pada lampiran B.6.



#### 4.2.3.7. **Sequence Diagram Menentukan Jumlah Kuota**

*Sequence diagram* menentukan jumlah kuota digunakan untuk menggambarkan skenario *use case* menentukan jumlah kuota secara teknis. *Sequence diagram* menentukan jumlah kuota dapat dilihat pada lampiran B.7.

#### 4.2.3.8. **Sequence Diagram Mengelola Data Ranking**

*Sequence diagram* mengelola data ranking digunakan untuk menggambarkan skenario *use case* mengelola data ranking secara teknis. *Sequence diagram* mengelola data ranking berisi *class* dan *method* yang digunakan dalam proses mengelola data ranking. *Sequence diagram* mengelola data ranking dapat dilihat pada lampiran B.8.

#### 4.2.3.9. **Sequence Diagram Validasi Hasil Rekomendasi**

*Sequence diagram* validasi hasil rekomendasi digunakan untuk menggambarkan skenario *use case* validasi hasil rekomendasi secara teknis. *Sequence diagram* validasi hasil rekomendasi berisi *class* dan *method* yang digunakan dalam proses validasi hasil rekomendasi. *Sequence diagram* validasi hasil rekomendasi ini diawali dengan ketua yang telah login ke dalam sistem mengklik menu validasi. *Sequence diagram* validasi hasil rekomendasi dapat dilihat pada lampiran B.9.

#### 4.2.3.10. **Sequence Diagram Melihat Hasil Rekomendasi**

*Sequence diagram* validasi hasil rekomendasi digunakan untuk menggambarkan skenario *use case* validasi hasil rekomendasi secara teknis. *Sequence diagram* validasi hasil rekomendasi berisi *class* dan *method* yang digunakan dalam proses validasi hasil rekomendasi. *Sequence diagram* validasi hasil rekomendasi dapat dilihat pada Lampiran B.10.

#### 4.2.4 **Activity Diagram**

*Activity diagram* pada penelitian ini menggambarkan alur aktivitas pada sistem informasi rekomendasi penerima Program Indonesia Pintar yang dibangun.

##### 4.2.4.1. **Activity Diagram Login**

*Activity diagram login* menggambarkan alur aktivitas *login* pada sistem informasi rekomendasi penerima Program Indonesia Pintar. Alur pada *activity* ini dimulai dengan aktor mengklik tombol masuk kemudian sistem menampilkan

halaman login. Selanjutnya, aktor mengisi *e-mail* dan *password* serta klik tombol login. Jika login gagal maka sistem akan menampilkan pesan error, jika isian data tidak lengkap maka menampilkan span *please fill out this, field*, jika login berhasil maka sistem menampilkan halaman dashboard. *Activity* diagram login selengkapnya dapat dilihat pada lampiran C.1.

#### **4.2.4.2. Activity Diagram Mengelola Data Pengguna**

*Activity* diagram mengelola data pengguna menggambarkan alur aktivitas manajemen data pengguna pada sistem informasi rekomendasi penerima Program Indonesia Pintar. Alur pada *activity* ini dimulai dengan aktor memilih menu pengguna kemudian sistem menampilkan halaman data pengguna. Selanjutnya, aktor bisa memilih untuk mengubah data pengguna. *Activity* diagram mengelola data pengguna selengkapnya dapat dilihat pada lampiran C.2.

#### **4.2.4.3. Activity Diagram Mengelola Data Siswa**

*Activity* diagram mengelola data siswa menggambarkan alur aktivitas manajemen data siswa pada sistem informasi rekomendasi penerima Program Indonesia Pintar. Alur pada *activity* ini dimulai dengan aktor memilih menu data siswa kemudian sistem menampilkan halaman data siswa. Selanjutnya, aktor bisa memilih untuk menambah atau mengubah data siswa. *Activity* diagram mengelola data siswa selengkapnya dapat dilihat pada lampiran C.3.

#### **4.2.4.4. Activity Diagram Mengelola Data Pendaftar**

*Activity* diagram mengelola data pendaftar menggambarkan alur aktivitas manajemen data pendaftar pada sistem informasi rekomendasi penerima Program Indonesia Pintar. Alur pada *activity* ini dimulai dengan aktor memilih menu pendaftar kemudian sistem menampilkan halaman skema seleksi data pendaftar. Selanjutnya, aktor bisa memilih untuk melihat atau menambah data pendaftar. *Activity* diagram mengelola data pendaftar selengkapnya dapat dilihat pada lampiran C.4.

#### **4.2.4.5. Activity Diagram Mengelola Data Kriteria**

*Activity* diagram mengelola data kriteria menggambarkan alur aktivitas manajemen data kriteria pada sistem informasi rekomendasi penerima Program Indonesia Pintar. Alur pada *activity* ini dimulai dengan aktor memilih menu kriteria

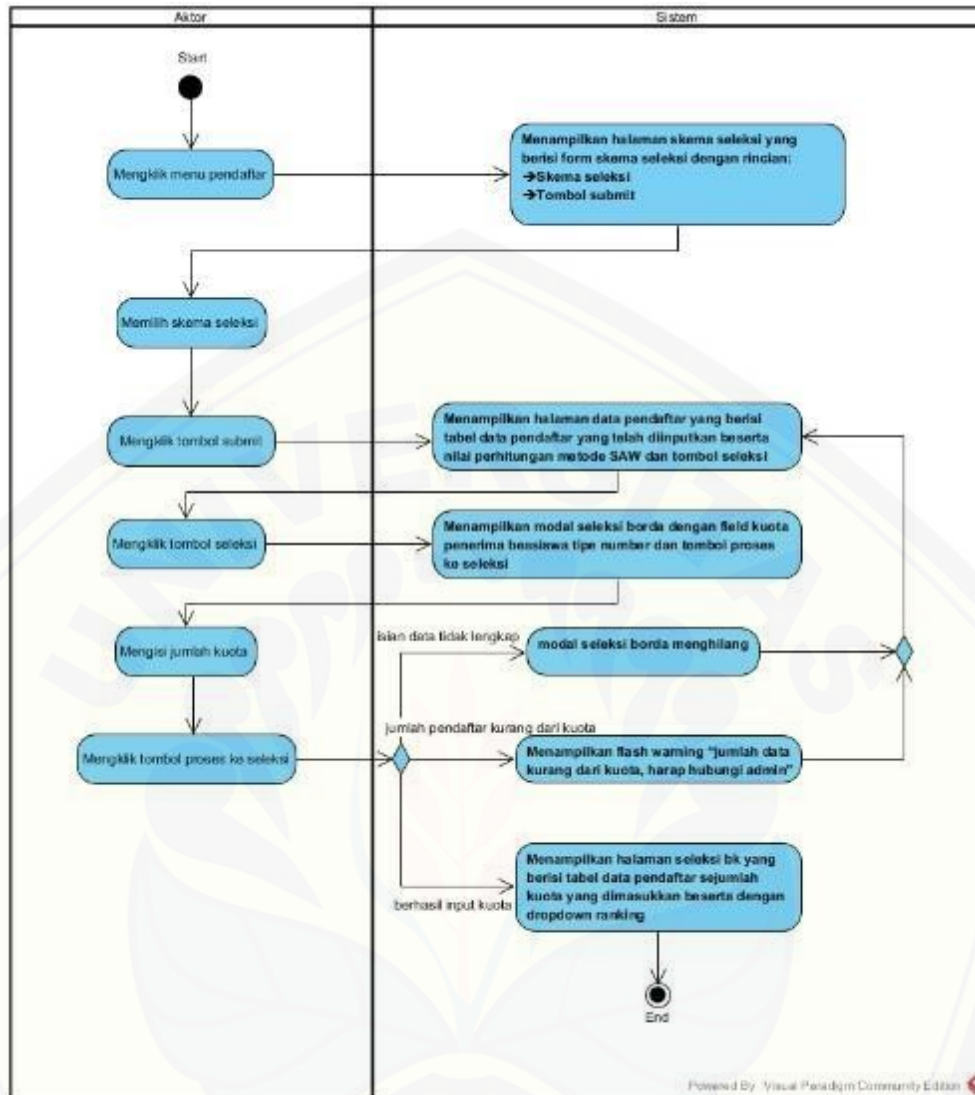
kemudian sistem menampilkan halaman data kriteria. Selanjutnya, aktor bisa memilih untuk menambah data kriteria, mengaktifkan maupun menonaktifkan data kriteria, atau melihat detail data kriteria. *Activity* diagram mengelola data kriteria selengkapnya dapat dilihat pada lampiran C.5.

#### **4.2.4.6. Activity Diagram Menambah Skema Seleksi**

*Activity* diagram menambah skema seleksi menggambarkan alur aktivitas menambah data skema seleksi pada sistem informasi rekomendasi penerima Program Indonesia Pintar. Alur pada *activity* ini dimulai dengan aktor memilih menu tambah skema kemudian sistem menampilkan halaman tambah skema seleksi. Selanjutnya, aktor bisa menambah data skema seleksi *Activity* diagram mengelola data kriteria selengkapnya dapat dilihat pada lampiran C.6.

#### **4.2.4.7. Activity Diagram Menentukan Jumlah Kuota**

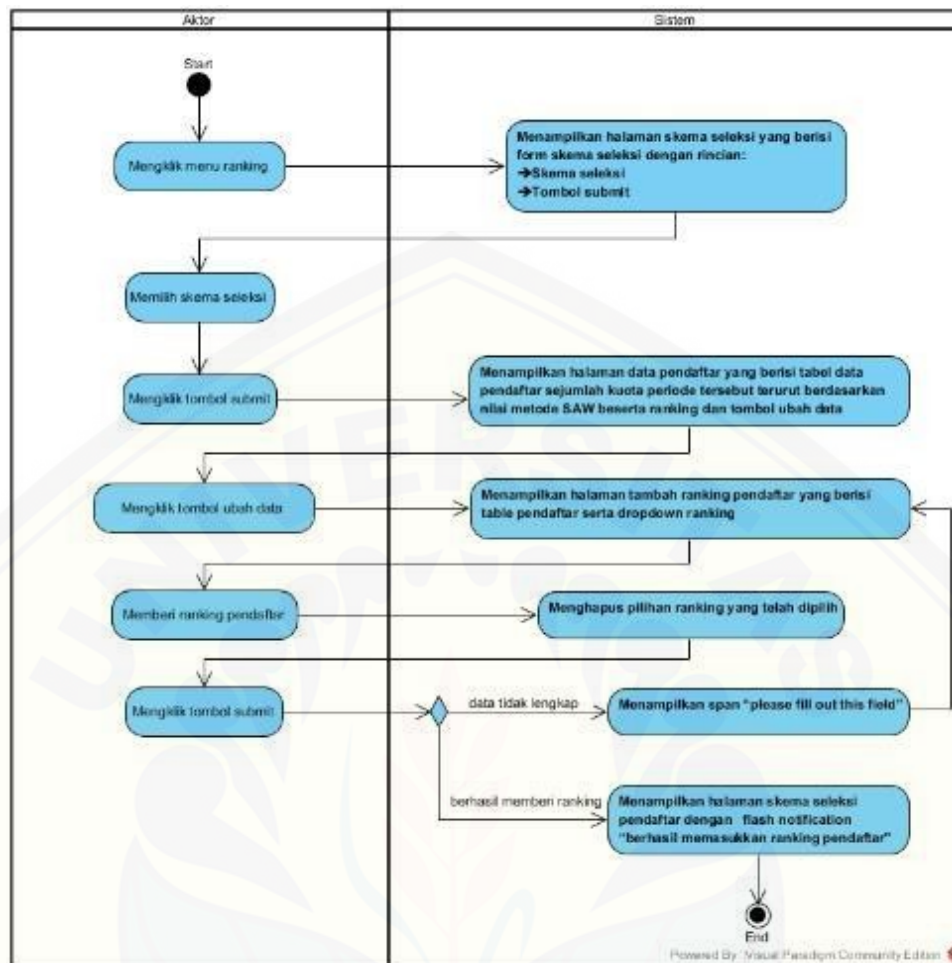
*Activity* diagram menentukan jumlah kuota menggambarkan alur aktivitas menentukan jumlah kuota pada sistem informasi rekomendasi penerima Program Indonesia Pintar. Alur pada *activity* ini dimulai dengan aktor memilih menu pendaftar kemudian sistem menampilkan halaman data pendaftar. Selanjutnya, aktor dapat mengklik tombol seleksi untuk memunculkan *modal* seleksi *board* guna menentukan jumlah kuota. *Activity* diagram menentukan jumlah kuota selengkapnya dapat dilihat pada gambar 4.2.



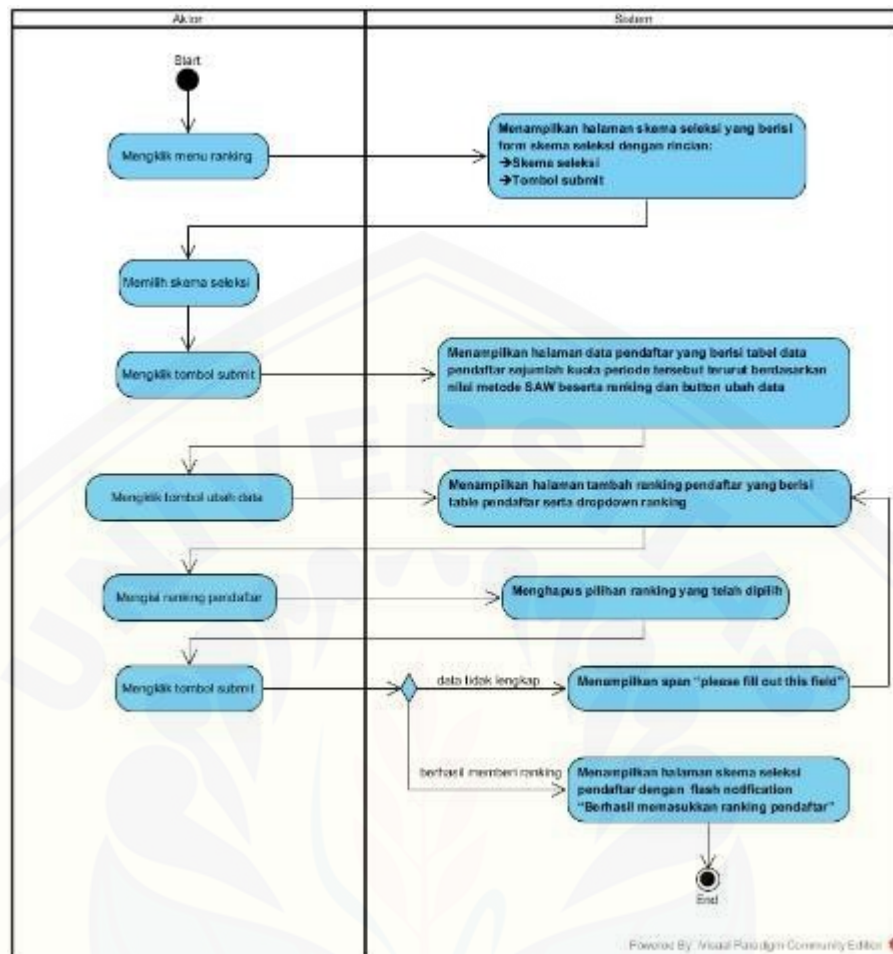
Gambar 4.2. Activity Diagram Menentukan Jumlah Kuota

#### 4.2.4.8. Activity Diagram Mengelola Data Ranking

*Activity* diagram mengelola data ranking menggambarkan alur aktivitas manajemen data ranking pada sistem informasi rekomendasi penerima Program Indonesia Pintar. Alur pada *activity* ini dimulai dengan aktor memilih menu ranking kemudian sistem menampilkan halaman data ranking. Selanjutnya, aktor dapat mengklik *floating button* ranking untuk memunculkan halaman beri atau ubah ranking guna memberi maupun mengubah data ranking. *Activity* diagram mengelola data ranking selengkapnya dapat dilihat pada gambar 4.3 dan 4.4.



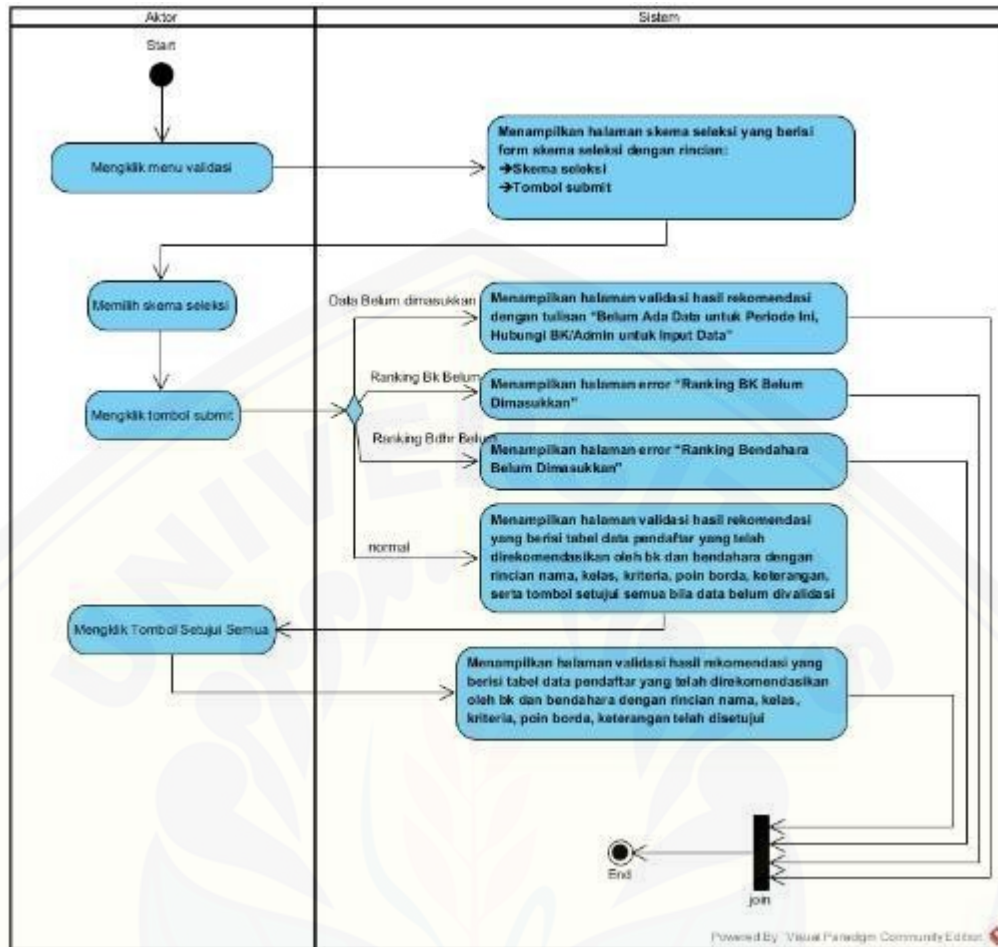
Gambar 4.3. Activity Diagram (Guru BK) Mengelola Data Ranking



Gambar 4.4. Activity Diagram (Bendahara) Mengelola Data Ranking

#### 4.2.4.9. Activity Diagram Validasi Hasil Rekomendasi

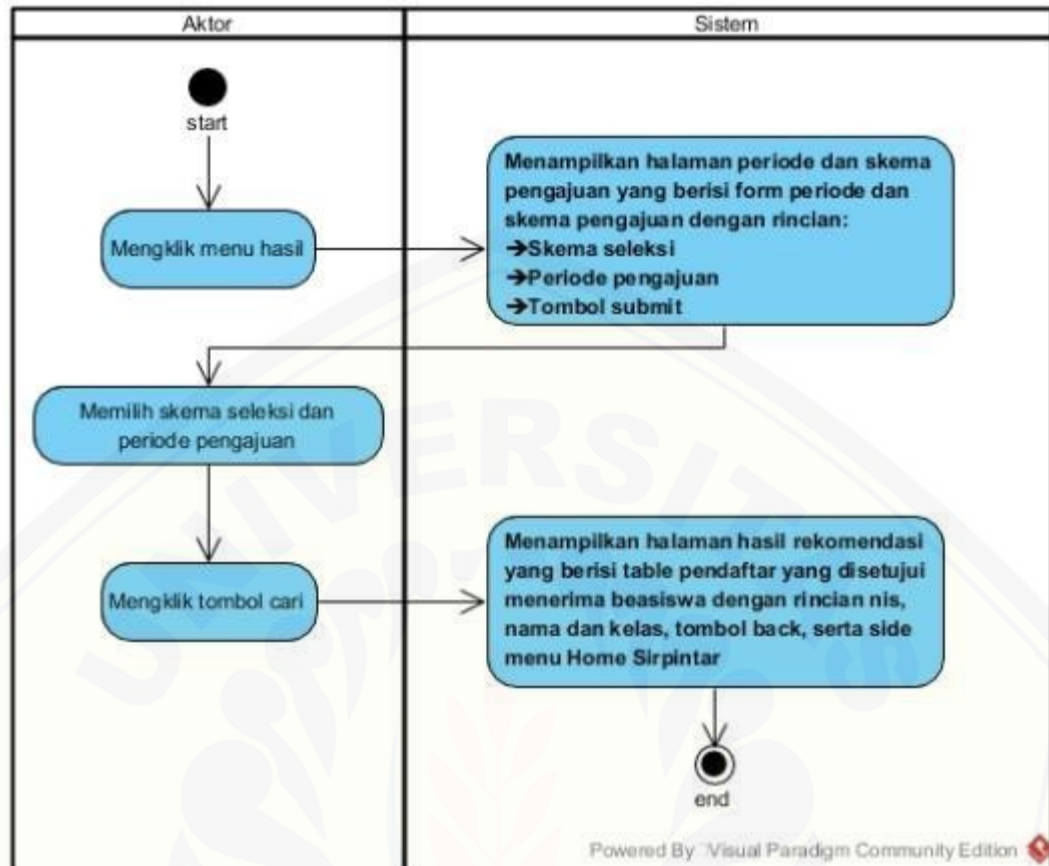
*Activity* diagram validasi hasil rekomendasi menggambarkan alur aktivitas validasi hasil rekomendasi pada sistem informasi rekomendasi penerima Program Indonesia Pintar. Alur pada *activity* ini dimulai dengan aktor memilih menu validasi penerima PIP kemudian sistem menampilkan halaman skema seleksi, setelah skema dipilih sistem kemudian menampilkan halaman validasi hasil rekomendasi. Selanjutnya, aktor dapat mengklik tombol setuju semua untuk menyetujui hasil rekomendasi. *Activity* diagram validasi hasil rekomendasi selengkapnya dapat dilihat pada gambar 4.5.



Gambar 4.5. Activity Diagram Validasi Hasil Rekomendasi

#### 4.2.4.10 Activity Diagram Melihat Hasil Rekomendasi

*Activity* diagram melihat hasil rekomendasi menggambarkan alur aktivitas melihat hasil rekomendasi pada sistem informasi rekomendasi penerima Program Indonesia Pintar. Alur pada *activity* ini dimulai dengan aktor memilih menu hasil rekomendasi kemudian sistem menampilkan halaman pilih periode serta skema seleksi. *Activity* diagram melihat hasil rekomendasiselengkapnya dapat dilihat pada gambar 4.6.



Gambar 4.6. Activity Diagram Pengguna Melihat Hasil Rekomendasi

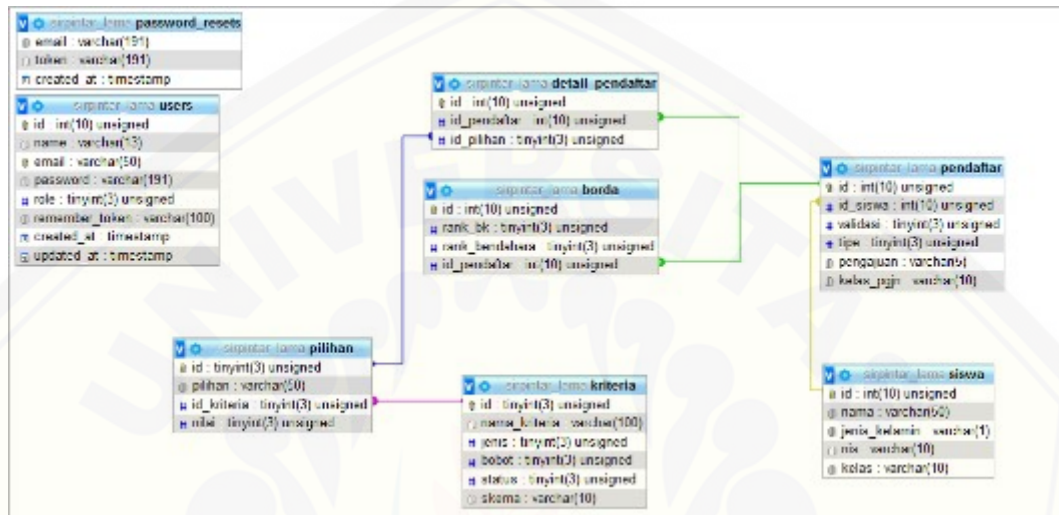
#### 4.2.5. Class Diagram

*Class diagram* menggambarkan tentang relasi antar *class* dalam sistem. Relasi terjadi antara *controller* dan *view*. Keterkaitan dan ketergantungan antar *class* dalam sistem bisa dilihat pada Lampiran D.



#### 4.2.6 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) pada sistem informasi rekomendasi penerima Program Indonesia Pintar ini menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai relasi. ERD sistem ditunjukkan pada gambar 4.7.



Gambar 4.7. Entity Relationship Diagram (ERD)

#### 4.3. Implementasi Perancangan

Tahap selanjutnya setelah tahap desain sistem pada penelitian ini adalah tahap pengimplementasian desain perancangan kedalam bahasa pemrograman. Bahasa pemrograman yang digunakan adalah Bahasa Pemrograman PHP dengan menggunakan *FrameworkLaravel*. Tahap implementasi perancangan ini menjelaskan tentang fitur-fitur yang terdapat pada sistem informasi rekomendasi penerima Program Indonesia Pintar. Fitur-fitur tersebut meliputi pengelolaan data pengguna, pengelolaan data siswa, menambah skema seleksi, pengelolaan data pendaftar, pengelolaan ranking pendaftar, pengelolaan data kriteria, melihat data hasil rekomendasi, validasi hasil rekomendasi, serta menentukan jumlah kuota penerima PIP pada sistem informasi rekomendasi penerima Program Indonesia Pintar.

#### 4.4. Pengujian Sistem

Pengujian dilakukan untuk mengevaluasi sistem yang telah dibuat. Proses pengujian pada sistem informasi yang dibangun pada penelitian ini menggunakan pengujian *black box*.

##### 4.4.3 Pengujian *Black Box*

Pengujian *Black Box* digunakan untuk menguji sistem dari segi spesifikasi fungsional sistem dengan tujuan mengetahui apakah fungsi-fungsi *input*, dan keluaran sistem sesuai dengan spesifikasi yang dibutuhkan oleh pengguna. Hasil pengujian *Black Box* untuk menentukan jumlah kuota bisa dilihat pada tabel 4.10.

Tabel 4.10. Pengujian Black Box Fitur Menentukan Jumlah Kuota

No	Fitur	Aksi	Hasil	Keterangan
1.	Menentukan jumlah kuota	Klik menu pendaftar	Menampilkan halaman skema seleksi	Berhasil
		Pilih skema seleksi lalu klik tombol submit	Menampilkan halaman data pendaftar yang berisi tabel data pendaftar yang telah diinputkan beserta nilai perhitungan Metode SAW dan tombol seleksi	Berhasil
		Klik tombol seleksi	Menampilkan <i>modal</i> seleksi borda dengan <i>field</i> kuota penerima tipe <i>number</i> dan tombol proses ke seleksi	Berhasil

Dilanjutkan

Lanjutan

No	Fitur	Aksi	Hasil	Keterangan
1.	Menentukan jumlah kuota	Klik tombol proses ke seleksi	Menampilkan halaman seleksi bk yang berisi tabel data pendaftar sejumlah kuota yang dimasukkan beserta dengan <i>dropdown ranking</i>	Berhasil
		Klik tombol proses ke seleksi saat isian data tidak lengkap	<i>modal</i> seleksi borda menghilang	Berhasil
		Klik tombol proses ke seleksi saat jumlah pendaftar kurang dari kuota	Menampilkan <i>flash warning</i> Data Pendaftar Belum Cukup untuk Memenuhi Kuota! Silakan Hubungi Admin!”	Berhasil

## BAB 6. PENUTUP

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

### 6.1. Kesimpulan

Kesimpulan dari hasil penelitian ini adalah penerapan Metode *Simple Additive Weighting* untuk menentukan prioritas siswa penerima Program Indonesia Pintar menghasilkan tingkat akurasi sebesar 90%. Penerapan Metode *Borda* untuk menentukan prioritas siswa penerima bantuan PIP tidak cocok. Hal ini dikarenakan Metode *Borda* tidak mampu menentukan rangking atau prioritas siswa penerima PIP dengan baik ketika ada 2 siswa atau lebih yang memiliki nilai *Borda Count* yang sama. Selain itu, penerapan Metode *Borda* lebih memakan waktu dikarenakan setiap pembuat keputusan diharuskan untuk memberikan rangking kepada masing-masing kandidat.

### 6.2. Saran

Saran penulis untuk pengembangan lebih lanjut penelitian ini adalah menggabungkan Metode *Simple Additive Weighting* dengan metode *voting* lainnya yang sama-sama menghasilkan *multiple winners*. Hal ini dikarenakan penerima Program Indonesia Pintar tidak hanya 1 siswa saja.

### DAFTAR PUSTAKA

- Butar Butar, O. T. (2015). Sistem Pendukung Keputusan Rekomendasi Penerima Bantuan Siswa Miskin (BSM) dengan Metode Simple Additive Weighting (SAW) (Studi Kasus : SMPN 2 Tarabintang). *Pelita Informatika Budi Darma, Volume : IX, Nomor: 3, April 2015*.
- Dermawan, J., & Hartini, S. (2017). Implementasi Model Waterfall pada Pengembangan Sistem Informasi Perhitungan Nilai Mata Pelajaran Berbasis Web pada Sekolah Dasar Al-Azhar Syifa Budi Jatibening. *Paradigma Vol. 19 No.42-147*.
- Firmansyah, Y., & Jamilah. (2018). Implementasi SDLC Waterfall dalam Pembuatan Game Edukasi Perjuangan Indonesia "Historia" menggunakan RPG Maker Mv Berbasis Android. *Jurnal Khatulistiwa Informatika, Vol. VI No. 2178-185*.
- Handayani, B., & S., R. (2014). Perbandingan Metode AHP-SAW Dengan FMCDM-SAW Pada Pemberian Pinjaman Modal Usaha Pertanian. *JUTISI Vol. 3, No. 3, Desember, 2014*.
- Kaunen, & Arizona, N. D. (2017). Aplikasi Pengolahan Data Anggaran Pendapatan dan Belanja Desa (APBDES) Pada Kantor Desa Bakau Kecamatan Jawai Berbasis Web. *Cybernetics, Vol.01, No.02119*.
- Nugraha, W., Syarif, M., & Dharmawan, W. S. (2018). Penerapan Metode SDLC Waterfall dalam Sistem Informasi Inventory Barang Berbasis Desktop. *JUSIM (Jurnal Sistem Informasi Musirawas) Vol 03 No.01*
- Oei, S. (2013). Group Decision Support System Untuk Pembelian Rumah dengan Menggunakan Analytical Hierarchy Process (AHP) dan BORDA. *Seminar Nasional Informatika 2013 (semnasIN 2013)*
- P., D. K., Hamdana, E. N., & Fahreza, D. D. (2018). Sistem Pendukung Keputusan Prioritas Calon Penerima Program Indonesia Pintar pada Siswa Sekolah Menengah Pertama menggunakan Metode Topsis. *Jurnal Informatika Polinema*101-108.
- Petunjuk Teknis Program Indonesia Pintar (PIP)*. Jakarta: Direktorat Jenderal Pendidikan Islam Kementerian Agama RI.

- Rosyidah, B., Widiyaningtiyas, T., & Herwanto, H. W. (2015). Desain dan Implementasi Sistem Informasi Administrasi Siswa Berbasis Web di SMK TI Global Ponorogo. *TEKNO Jurnal Teknologi Elektro dan Kejuruan*, 50.
- Rubianto, D. T. (2016). Sistem Pendukung Keputusan Pemberian Beasiswa Menggunakan Metode Simple Additive Weighting (SAW). *Artikel Skripsi Universitas Nusantara PGRI Kediri*
- Sari, R. P., Santoso, A. J., & Ernawati. (2014). Perancangan Sistem Pendukung Keputusan Kelompok Metode Topsis dan Borda untuk Evaluasi Kegiatan Penanganan Infrastruktur Jalan. *Seminar Nasional Teknologi Informasi dan Komunikasi 2014 (SENTIKA 2014)*, 29.
- T., A. j., Yanosma, D., & Anggriani, K. (2016). Implementasi Metode K-Nearest Neighbor (KNN) dan Simple Additive Weighting (SAW) dalam Pengambilan Keputusan Seleksi Penerimaan Anggota Paskibraka (Studi Kasus : Dinas Pemuda dan Olahraga Provinsi Bengkulu). *Jurnal Pseudocode, Volume III Nomor 2, September 2016*
- Wulansari, D. J. (2017). Sistem Pendukung Keputusan Untuk Menentukan Penerima Kartu Indonesia Pintar Menggunakan Metode Simple Additive Weighting.

## Lampiran

A. Skenario *Use Case*A.1. Skenario *Use Case Login*Tabel A.1. Skenario *Use Case Login* Admin

<b>Nama Use Case</b>	<i>Login</i>
<b>Aktor</b>	<i>Admin</i>
<b>Deskripsi Singkat</b>	<i>Admin</i> akan mengakses aplikasi
<b>Prekondisi</b>	<i>e-mail</i> dan <i>password</i> yang akan digunakan untuk <i>login</i>
<b>Prakondisi</b>	Berhasil <i>login</i>
<b>Flow Events</b>	
<b>Skenario Normal : <i>login</i></b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Membuka halaman website	
	2. Menampilkan <i>landing page</i> website beserta tombol masuk
3. Mengklik tombol masuk	
	4. Menampilkan halaman login dengan form berisi <i>field</i> <i>e-mail</i> dan password, tombol login, serta <i>forgot password</i>
5. Mengisi <i>form e-mail</i> dan <i>password</i>	
6. Mengklik tombol login	
	7. Menampilkan halaman dashboard admin
<b>Skenario Alternatif : <i>e-mail</i> atau <i>password</i> salah</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
6a. Mengklik tombol login	
	7a. Menampilkan pesan teks “ <b><i>These credentials do not match our records.</i></b> ”
<b>Skenario Alternatif : isian data tidak lengkap</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
6b. Mengklik tombol login	
	7b. Menampilkan span “ <i>please fill out this field</i> ”

Tabel A.2. Skenario *Use Case Login* Guru Bk

<b>Nama Use Case</b>	<i>Login</i>
<b>Aktor</b>	BK
<b>Deskripsi Singkat</b>	BK akan mengakses aplikasi
<b>Prekondisi</b>	<i>e-mail</i> dan <i>password</i> yang akan digunakan untuk <i>login</i>
<b>Prakondisi</b>	Berhasil <i>login</i>
<b>Flow Events</b>	
<b>Skenario Normal : <i>login</i></b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Membuka halaman website	
	2. Menampilkan landing page website beserta tombol masuk
3. Mengklik tombol masuk	
	4. Menampilkan halaman login dengan form berisi <i>field</i> <i>e-mail</i> dan <i>password</i> , tombol <i>login</i> , serta <i>forgot password</i>
5. Mengisi <i>form</i> <i>e-mail</i> dan <i>password</i>	
6. Mengklik tombol <i>login</i>	
	7. Menampilkan halaman <i>dashboard</i> BK
<b>Skenario Alternatif : <i>e-mail</i> atau <i>password</i> salah</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
6a. Mengklik tombol <i>login</i>	
	7a. Menampilkan pesan teks “ <b><i>These credentials do not match our records.</i></b> ”
<b>Skenario Alternatif : isian data tidak lengkap</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
6b. Mengklik tombol <i>login</i>	
	7b. Menampilkan span “ <i>please fill out this field</i> ”

Tabel A.3. Skenario *Use Case Login* Bendahara

<b>Nama Use Case</b>	<i>Login</i>
<b>Aktor</b>	<i>Bendahara</i>
<b>Deskripsi Singkat</b>	<i>Bendahara</i> akan mengakses aplikasi
<b>Prekondisi</b>	<i>e-mail</i> dan <i>password</i> yang akan digunakan untuk <i>login</i>
<b>Prakondisi</b>	Berhasil <i>login</i>
<b>Flow Events</b>	
<b>Skenario Normal : <i>login</i></b>	



Aksi Aktor	Reaksi Sistem
1. Membuka halaman website	
	2. Menampilkan landing page website beserta tombol masuk
3. Mengklik tombol masuk	
	4. Menampilkan halaman login dengan form berisi <i>field</i> -mail dan password, tombol login, serta <i>forgot password</i>
5. Mengisi form <i>e-mail</i> dan <i>password</i>	
6. Mengklik tombol login	
	7. Menampilkan halaman dashboard bendahara
<b>Skenario Alternatif : e-mail atau password salah</b>	
Aksi Aktor	Aksi Sistem
6a. Mengklik tombol login	
	7a. Menampilkan pesan teks “ <b><i>These credentials do not match our records.</i></b> ”
<b>Skenario Alternatif : isian data tidak lengkap</b>	
Aksi Aktor	Aksi Sistem
6b. Mengklik tombol login	
	7b. Menampilkan span “ <i>please fill out this field</i> ”

Tabel A.4. Skenario *Use Case Login* Ketua

<b>Nama Use Case</b>	<i>Login</i>
<b>Aktor</b>	Ketua
<b>Deskripsi Singkat</b>	Ketua akan mengakses aplikasi
<b>Prekondisi</b>	<i>e-mail</i> dan <i>password</i> yang akan digunakan untuk <i>login</i>
<b>Prakondisi</b>	Berhasil <i>login</i>
<b>Flow Events</b>	
<b>Skenario Normal : <i>login</i></b>	
Aksi Aktor	Reaksi Sistem
1. Membuka halaman website	
	2. Menampilkan landing page website beserta tombol masuk
3. Mengklik tombol masuk	
	4. Menampilkan halaman login dengan form berisi <i>field</i> -mail dan password, tombol login, serta <i>forgot password</i>
5. Mengisi <i>form e-mail</i> dan <i>password</i>	

6. Mengklik tombol login	
	7. Menampilkan halaman <i>dashboard</i> Ketua
<b>Skenario Alternatif : e-mail atau password salah</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
6a. Mengklik tombol login	
	7a. Menampilkan pesan teks “ <b><i>These credentials do not match our records.</i></b> ”
<b>Skenario Alternatif : isian data tidak lengkap</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
6b. Mengklik tombol login	
	7b. Menampilkan span “ <i>please fill out this field</i> ”

**A.2. Skenario Use Case Mengelola Data Pengguna**

Tabel A.5. Skenario Use Case Mengelola Data Pengguna

<b>Nama Use Case</b>	Mengelola Data Pengguna
<b>Aktor</b>	Admin
<b>Deskripsi Singkat</b>	Admin akan mengelola data Pengguna
<b>Prekondisi</b>	Admin berhasil login kedalam sistem
<b>Prakondisi</b>	Data pengguna berhasil diubah
<b>Flow Events</b>	
<b>Skenario Normal : melihat data pengguna</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Mengklik menu pengguna	
	2. Menampilkan halaman data pengguna meliputi tabel data pengguna dengan rincian nama pengguna, alamat <i>e-mail</i> , dan tombol edit pengguna
<b>Skenario Normal : mengubah data pengguna</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
3. Mengklik tombol <i>edit</i> pengguna	
	4. Menampilkan halaman form edit pengguna yang berisi nama, e-mail, <i>textfield</i> pengguna, dan tombol submit
5. Mengisi form <i>edit</i> pengguna	
6. Mengklik tombol <i>submit</i>	
	7. Menyimpan ke database
	8. Menampilkan halaman data pengguna disertai <i>flash message</i> “data pengguna berhasil diubah”
<b>Skenario Alternatif : Batal mengubah data pengguna</b>	

Aksi Aktor	Aksi Sistem
6a. Mengklik tombol batal	
	7a. Menampilkan halaman data pengguna meliputi tabel data pengguna dengan rincian nama pengguna, alamat <i>e-mail</i> kole, dan tombol edit pengguna
<b>Skenario Alternatif : isian data tidak lengkap</b>	
Aksi Aktor	Aksi Sistem
6b. Mengklik tombol submit	
	7b. Menampilkan span " <i>please fill out this field</i> "

### A.3. Skenario Use Case Mengelola Data Siswa

Tabel A.6. Skenario Use Case Mengelola Data Siswa

<b>Nama Use Case</b>	Mengelola Data Siswa
<b>Aktor</b>	<i>Admin</i>
<b>Deskripsi Singkat</b>	<i>Admin</i> akan mengelola data siswa
<b>Prekondisi</b>	<i>Admin</i> berhasil login ke dalam sistem
<b>Prakondisi</b>	1. Data siswa berhasil ditambah 2. Data siswa berhasil diubah
<b>Flow Events</b>	
<b>Skenario Normal : melihat data siswa</b>	
Aksi Aktor	Reaksi Sistem
1. Mengklik menu siswa	
	2. Menampilkan halaman data siswa yang berisi tabel data siswa yang telah diinputkan beserta <i>tombol edit</i> siswa dan <i>floating button</i>
<b>Skenario Normal : menambah data siswa</b>	
Aksi Aktor	Reaksi Sistem
3. Mengarahkan kursor ke <i>floating button</i>	
	4. Menampilkan <i>floating button</i> naikkan kelas dan <i>floating button</i> tambah siswa
5. Mengklik <i>floatingbutton</i> tambah siswa	
	6. Menampilkan halaman tambah siswa yang berisi form tambah siswa meliputi nis/nisn, nama, jenis kelamin, kelas
7. Mengisi form tambah siswa	
8. Mengklik tombol <i>submit</i>	
	9. Menyimpan ke database

	10. Menampilkan halaman data siswa dengan <i>flash notification</i> “berhasil menambahkan siswa”
<b>Skenario Alternatif : batal menambah data siswa</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
8a. Mengklik tombol batal	
	9a. Menampilkan halaman tambah siswa yang berisi form tambah siswa meliputi nis/nisn, nama, jenis kelamin, kelas
<b>Skenario Alternatif : isian data tidak lengkap</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
8b. Mengklik tombol <i>submit</i>	
	9b. Menampilkan span “ <i>please fill out this field</i> ”
<b>Skenario Alternatif : data siswa sudah ada</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
8c. Mengklik tombol <i>submit</i>	
	9c. Mengecek nama dan nis/nisn ke database
	10c. Menampilkan halaman tambah siswa dengan <i>flash warning</i> “siswa sudah terdaftar”
<b>Skenario Normal : mengubah siswa</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
11. Mengklik <i>tombol edit</i> siswa	
	12. Menampilkan halaman <i>edit</i> siswa yang berisi form edit siswa meliputi nis/nisn, nama, jenis kelamin, kelas
13. Mengisi form <i>edit</i> siswa	
14. Mengklik tombol <i>submit</i>	
	15. Menyimpan ke database
	16. Menampilkan halaman data siswa dengan <i>flash notification</i> “Berhasil Update Data Siswa”
<b>Skenario Alternatif : batal mengubah data siswa</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
14a. Mengklik tombol batal	
	15a. Menampilkan halaman tambah siswa yang berisi form tambah siswa meliputi nis/nisn, nama, jenis kelamin, kelas
<b>Skenario Alternatif : isian data tidak lengkap</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
14b. Mengklik tombol <i>submit</i>	

	15b. Menampilkan span “ <i>please fill out this field</i> ”
<b>Skenario Alternatif : data siswa sudah ada</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
14c. Mengklik tombol <i>submit</i>	
	15c. Mengecek nama dan nis/nisn ke database
	16c. Menampilkan halaman edit siswa dengan <i>flash warning</i> “siswa sudah terdaftar”
<b>Skenario Normal : kenaikan kelas</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
17. Mengarahkan kursor ke <i>floating button</i>	
	18. Menampilkan <i>floating button</i> naikan kelas dan <i>floating button</i> tambah siswa
19. Mengklik <i>floating button</i> naikan kelas	
	20. Menambahkan 1 pada semua data kelas saat ini
	21. Menyimpan ke database
	22. Menampilkan halaman data siswa dengan <i>flash notification</i> “Kenaikan Kelas Selesai Diproses!”

#### A.4. Skenario Use Case Mengelola Data Pendaftar

Tabel A.7. Skenario Use Case Mengelola Data Pendaftar

<b>Nama Use Case</b>	Mengelola Data Pendaftar
<b>Aktor</b>	<i>Admin</i>
<b>Deskripsi Singkat</b>	<i>Admin</i> akan mengelola data pendaftar
<b>Prekondisi</b>	<i>Admin</i> berhasil login ke dalam sistem
<b>Prakondisi</b>	3. Data pendaftar berhasil ditambah 4. Data pendaftar berhasil diubah
<b>Flow Events</b>	
<b>Skenario Normal : melihat data pendaftar</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Mengklik menu pendaftar	
	2. Menampilkan halaman skema seleksi yang berisi form skema seleksi dengan rincian: → Skema seleksi → Tombol <i>submit</i>
3. Memilih skema seleksi	

4. Mengklik tombol submit	
	5. Menampilkan halaman data pendaftar yang berisi tabel data pendaftar dengan rincian: → nama pendaftar → kriteria-kriteria skema seleksi → <i>floating button</i> tambah pendaftar
<b>Skenario Normal : menambah data pendaftar</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
6. Mengklik <i>floating button</i> tambah pendaftar	
	7. Menampilkan halaman tambah pendaftar yang berisi form tambah pendaftar meliputi <i>dropdown</i> nama siswa beserta kelas, <i>dropdown</i> pengajuan (tahun dan semester) dan <i>dropdown</i> kriteria-kriteria skema seleksi yang digunakan dalam seleksi, tombol submit.
8. Mengisi form tambah pendaftar	
9. Mengklik tombol <i>submit</i>	
	10. Menyimpan ke database
	11. Menampilkan halaman data pendaftar yang berisi tabel data pendaftar dengan rincian nama pendaftar dan kriteria-kriteria skema seleksi yang telah diinputkan, <i>floating button</i> tambah pendaftar, dan <i>flash notification</i> "berhasil menambahkan pendaftar"
<b>Skenario Alternatif : isian data tidak lengkap</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
9a. Mengklik tombol <i>submit</i>	
	10a. Menampilkan span "Data Wajib Dipilih!"

#### A.5. Skenario Use Case Mengelola Data Kriteria

Tabel A.8. Skenario Use Case Mengelola Data Kriteria

<b>Nama Use Case</b>	Mengelola Data Kriteria
<b>Aktor</b>	<i>Admin</i>
<b>Deskripsi Singkat</b>	<i>Admin</i> akan mengelola data kriteria
<b>Prekondisi</b>	<i>Admin</i> berhasil login ke dalam sistem
<b>Prakondisi</b>	1. Data kriteria berhasil ditambah 2. Data kriteria berhasil diubah
<b>Flow Events</b>	

<b>Skenario Normal : melihat kriteria</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Mengklik menu kriteria	
	2. Menampilkan halaman data kriteria yang berisi tabel data kriteria dengan rincian nomor, nama kriteria, bobot kriteria, status kriteria, tombol detail kriteria, dan <i>floating button</i> tambah kriteria
3. Mengklik tombol detail kriteria	
	4. Menampilkan halaman detail kriteria yang berisi tabel detail kriteria dengan rincian nama opsi kriteria dan nilai serta tombol <i>back</i>
5. Mengklik tombol <i>back</i>	
	6. Menampilkan halaman data kriteria yang berisi tabel data kriteria dengan rincian nomor, nama kriteria, bobot kriteria, status kriteria, tombol detail kriteria, dan <i>floating button</i> tambah kriteria
<b>Skenario Normal : menonaktifkan kriteria</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
7. Mengklik tombol nonaktifkan kriteria	
	8. Menonaktifkan kriteria yang terpilih.
	9. Menampilkan halaman data kriteria yang berisi tabel data kriteria dengan rincian nomor, nama kriteria, bobot kriteria, status kriteria, tombol detail kriteria, dan <i>floating button</i> tambah kriteria
<b>Skenario Normal : mengaktifkan kriteria</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
10. Mengklik tombol aktifkan kriteria	
	11. Mengaktifkan kriteria yang terpilih.
	12. Menampilkan halaman data kriteria yang berisi tabel data kriteria dengan rincian nomor, nama kriteria, bobot kriteria,

	status kriteria, tombol detail kriteria, dan <i>floating button</i> tambah kriteria
--	---

### A.6. Skenario *Use Case* Menambah Skema Seleksi

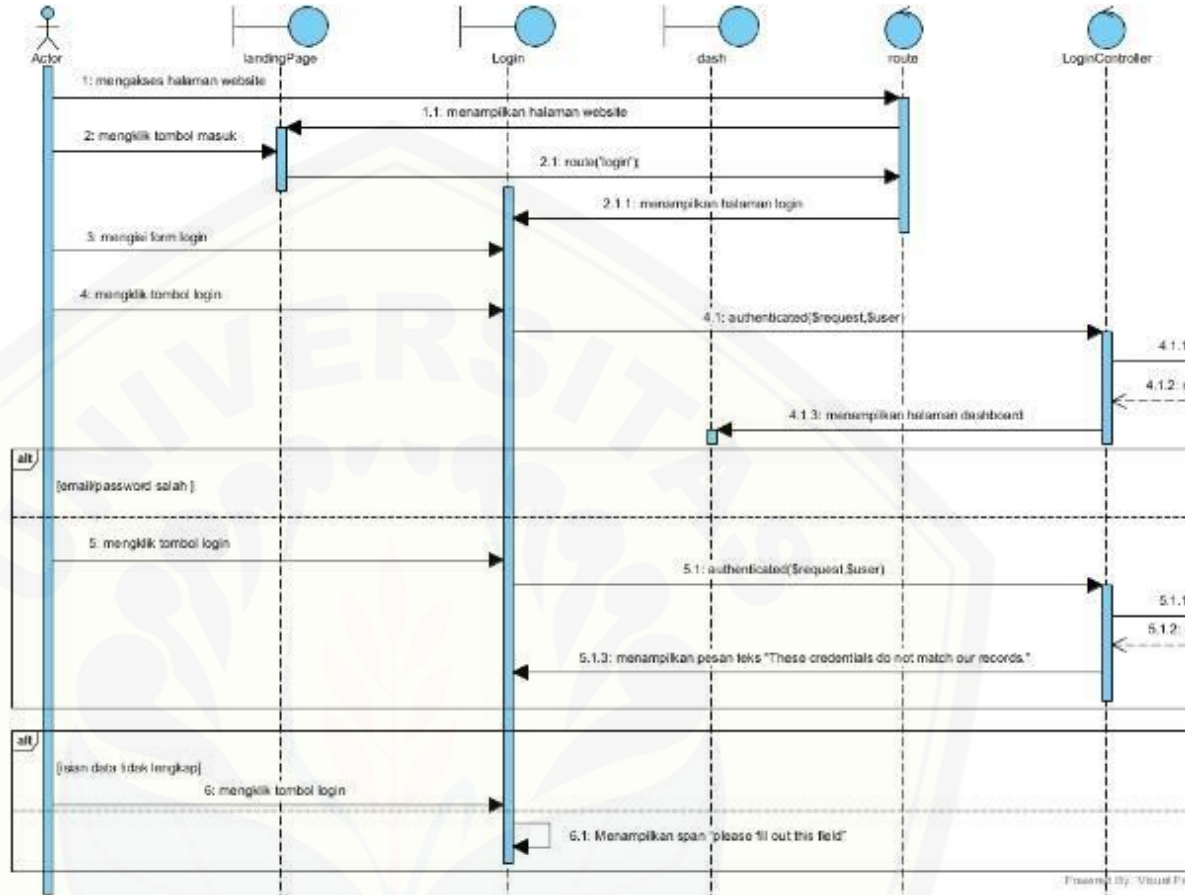
Tabel A.9. Skenario Use Case Menambah Skema Seleksi

<b>Nama <i>Use Case</i></b>	Menambah Skema Seleksi
<b>Aktor</b>	<i>Admin</i>
<b>Deskripsi Singkat</b>	<i>Admin</i> akan menambah skema seleksi
<b>Prekondisi</b>	<i>Admin</i> berhasil login ke dalam sistem
<b>Prakondisi</b>	Skema seleksi berhasil ditambah
<b>Flow Events</b>	
<b>Skenario Normal : menambah skema seleksi</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Mengklik menu skema	
	2. Menampilkan halaman tambah skema yang berisi <i>form</i> tambah skema meliputi nama skema dan tombol submit
3. Mengisi form tambah skema	
4. Mengklik tombol <i>submit</i>	
	5. Menyimpan ke database
	6. Menampilkan halaman tambah skema dengan <i>flash notification</i> "berhasil menambahkan skema seleksi"
<b>Skenario Alternatif : isian data tidak lengkap</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
4a. Mengklik tombol <i>submit</i>	
	5a. Menampilkan span " <i>please fill out this field</i> "
<b>Skenario Alternatif : data skema sudah ada</b>	
<b>Aksi Aktor</b>	<b>Aksi Sistem</b>
4b. Mengklik tombol <i>submit</i>	
	5b. Mengecek data skema ke database
	6b. Menampilkan halaman tambah skema dengan <i>flash warning</i> "skema sudah ada"



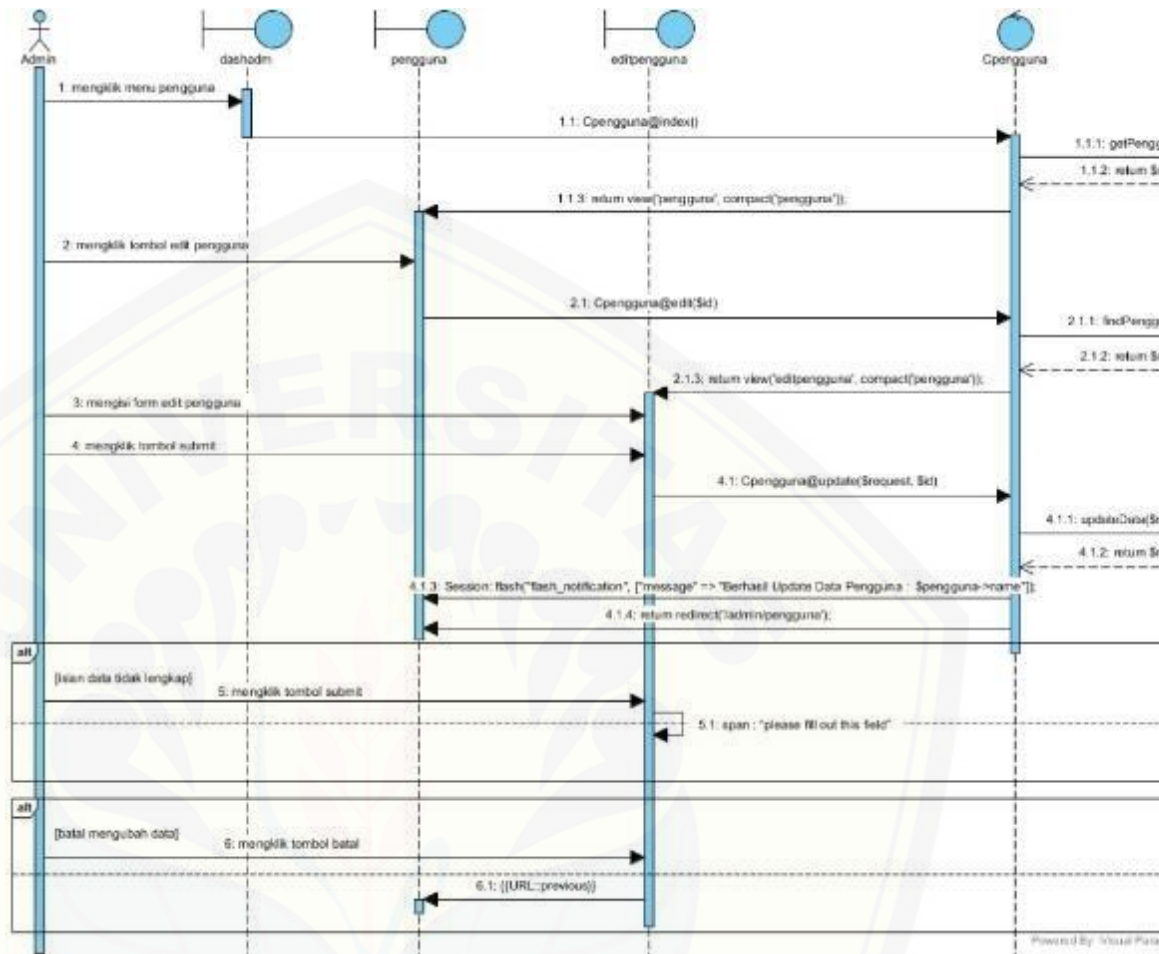
**B. Sequence Diagram**

**B.1. Sequence Diagram Login**



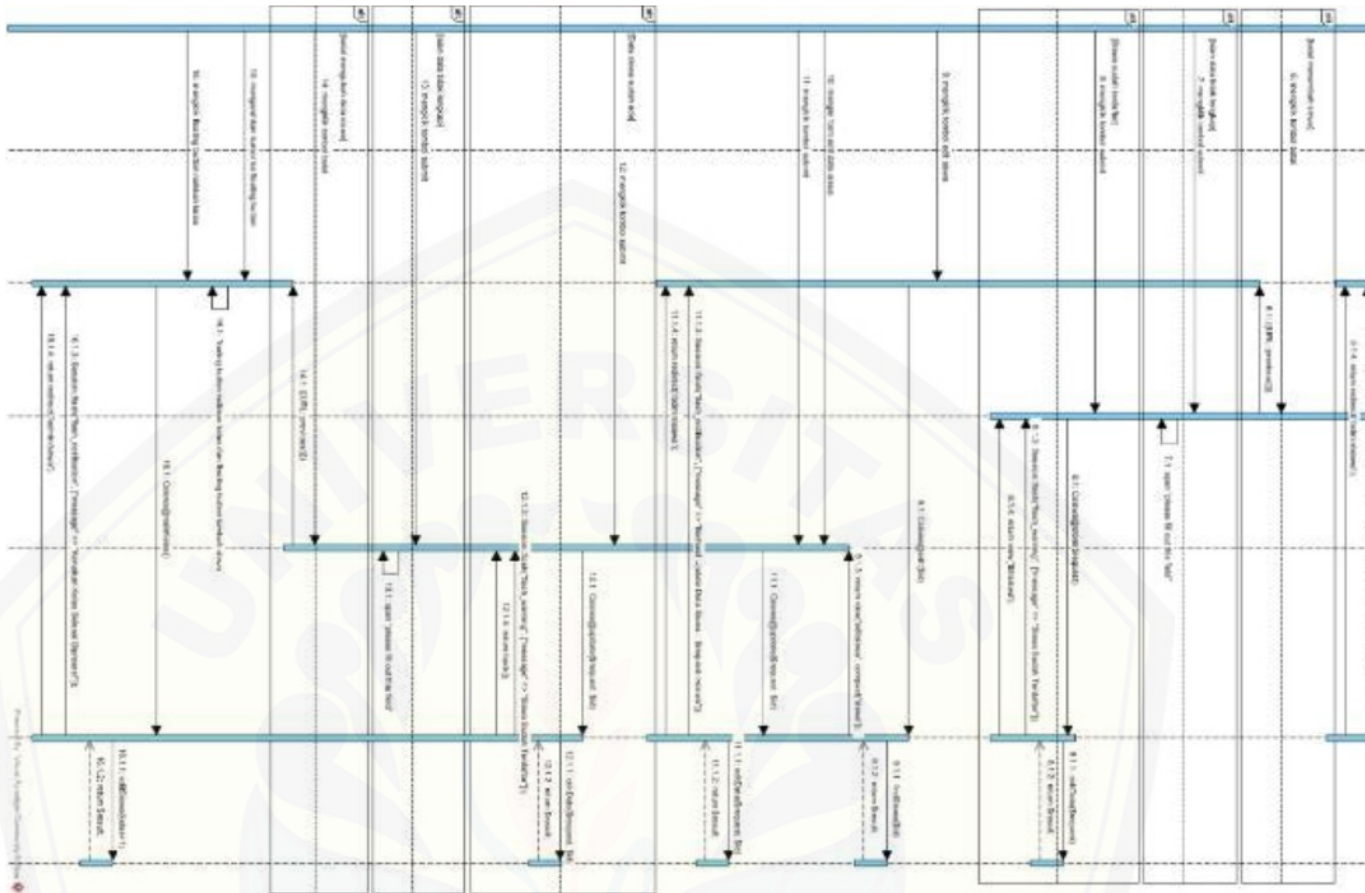
Gambar B.1. Sequence Diagram Login

**B.2. Sequence Diagram Mengelola Data Pengguna**



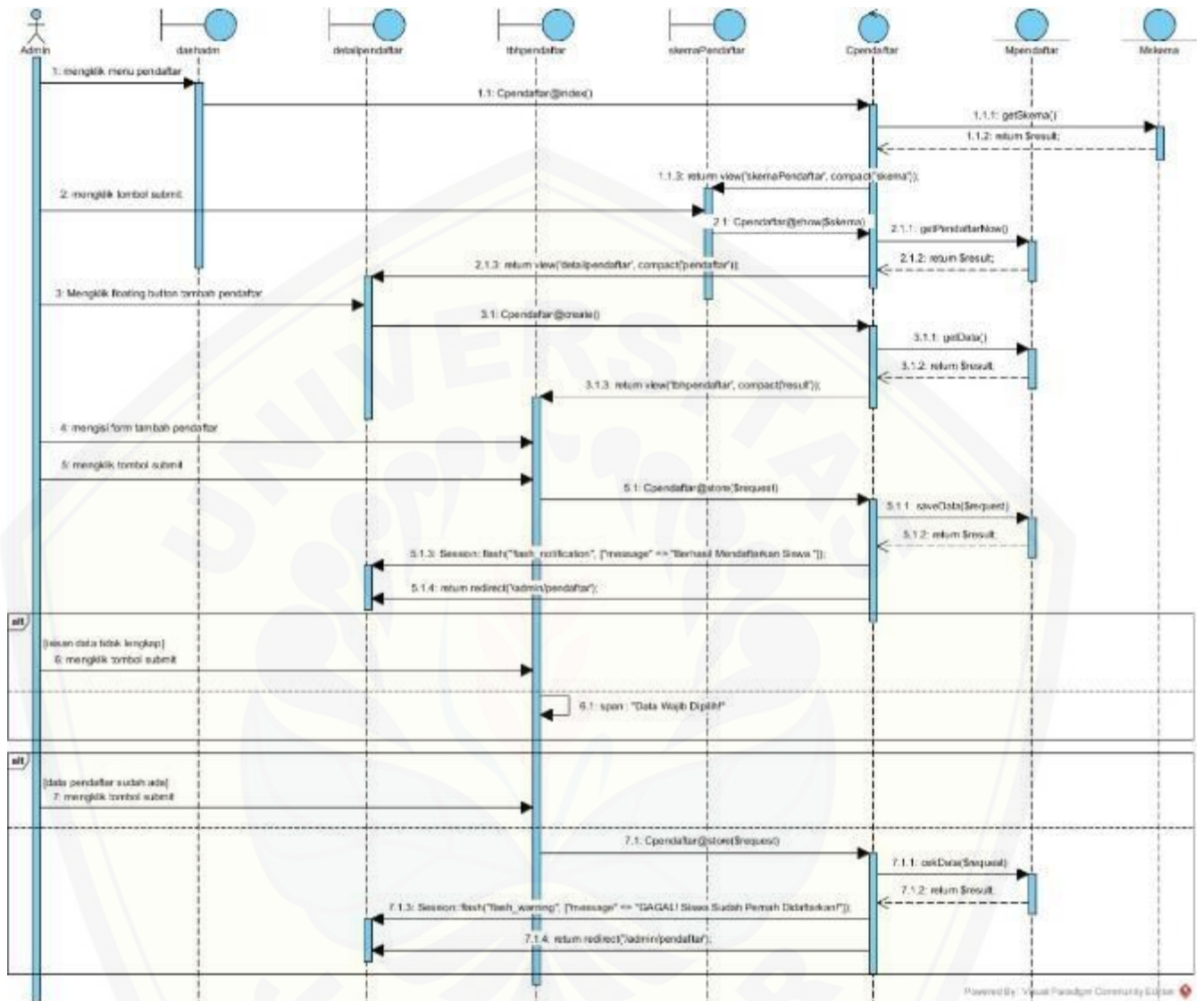
Gambar B.2. Sequence Diagram Mengelola Data Pengguna

**B.3. Sequence Diagram Mengelola Data Siswa**



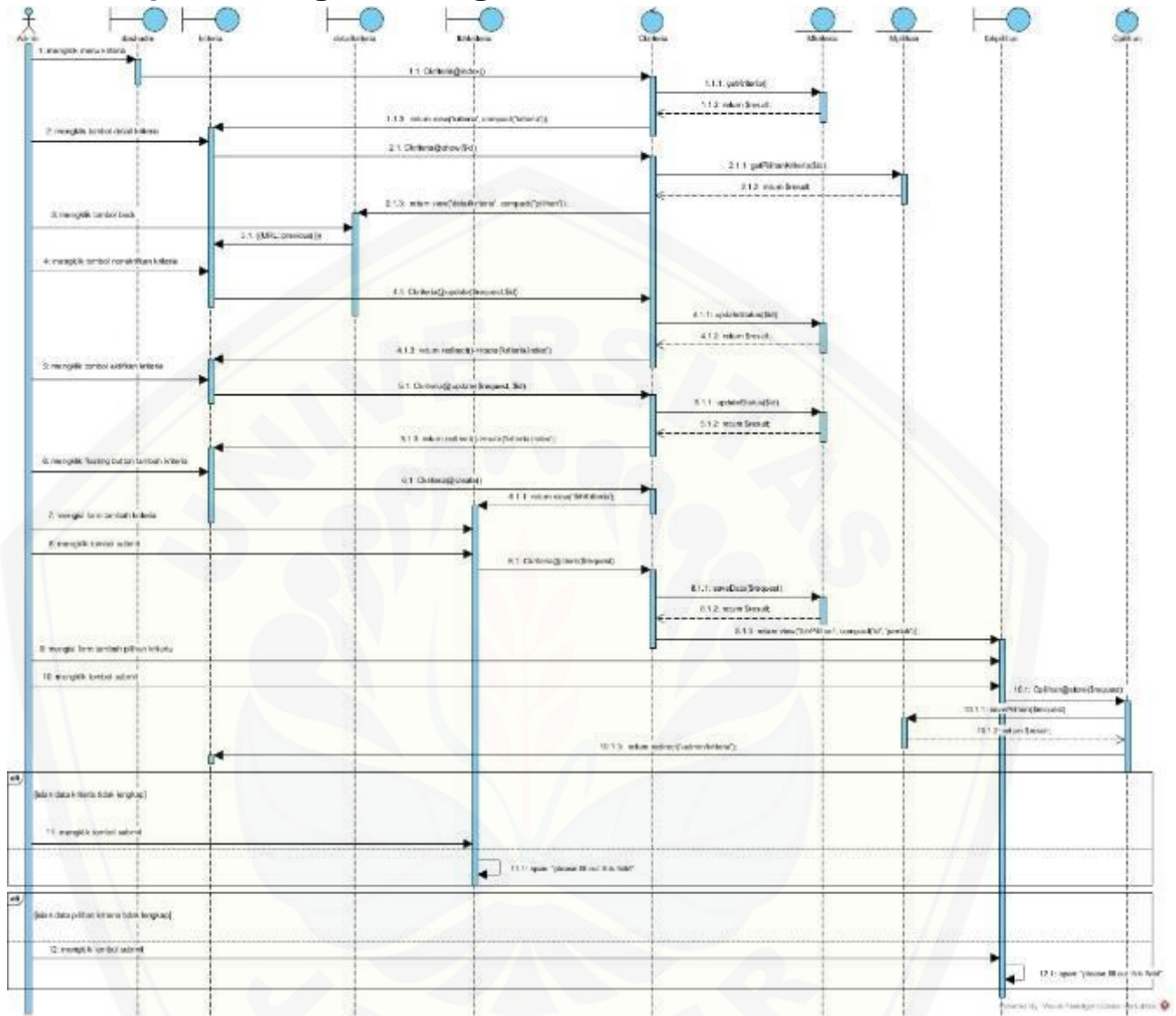
Gambar B.3. Sequence Diagram Mengelola Data Siswa

**B.4. Sequence Diagram Mengelola Data Pendaftar**



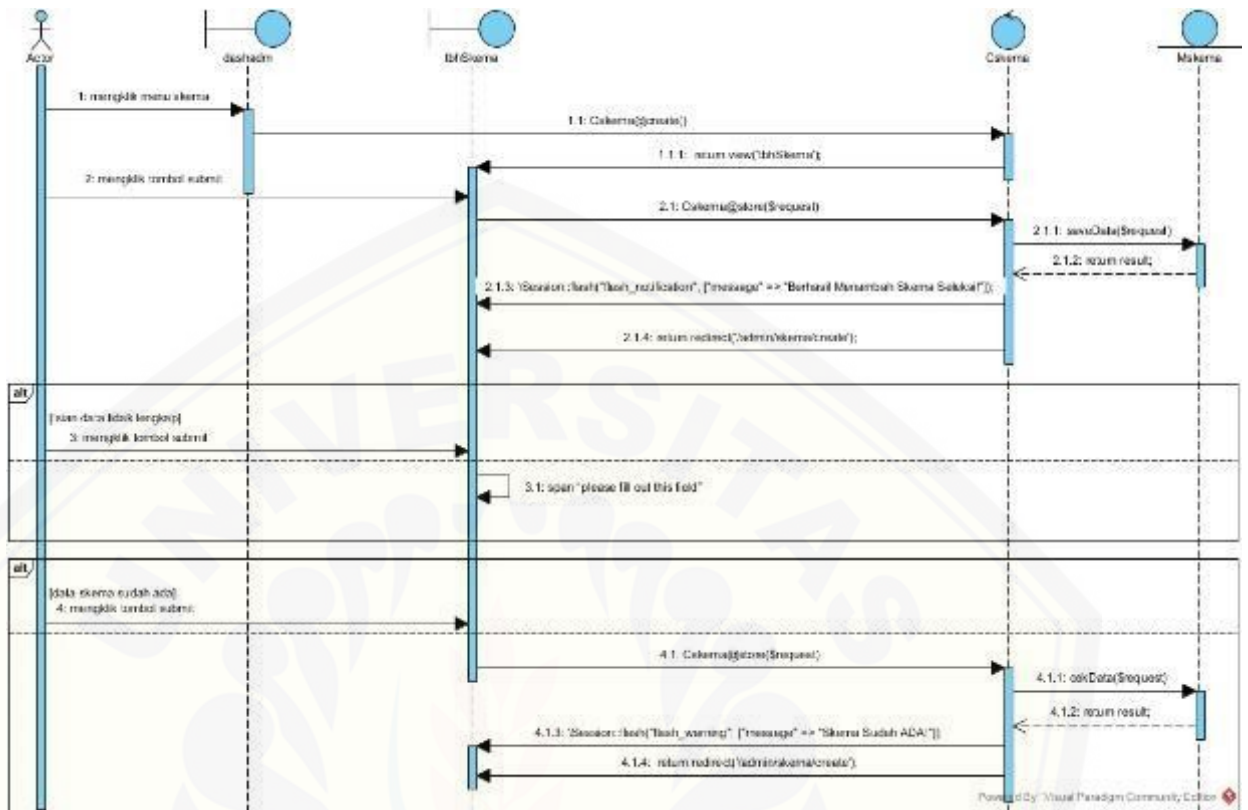
Gambar B.4. Sequence Diagram Mengelola Data Pendaftar

### B.5. Sequence Diagram Mengelola Data Kriteria



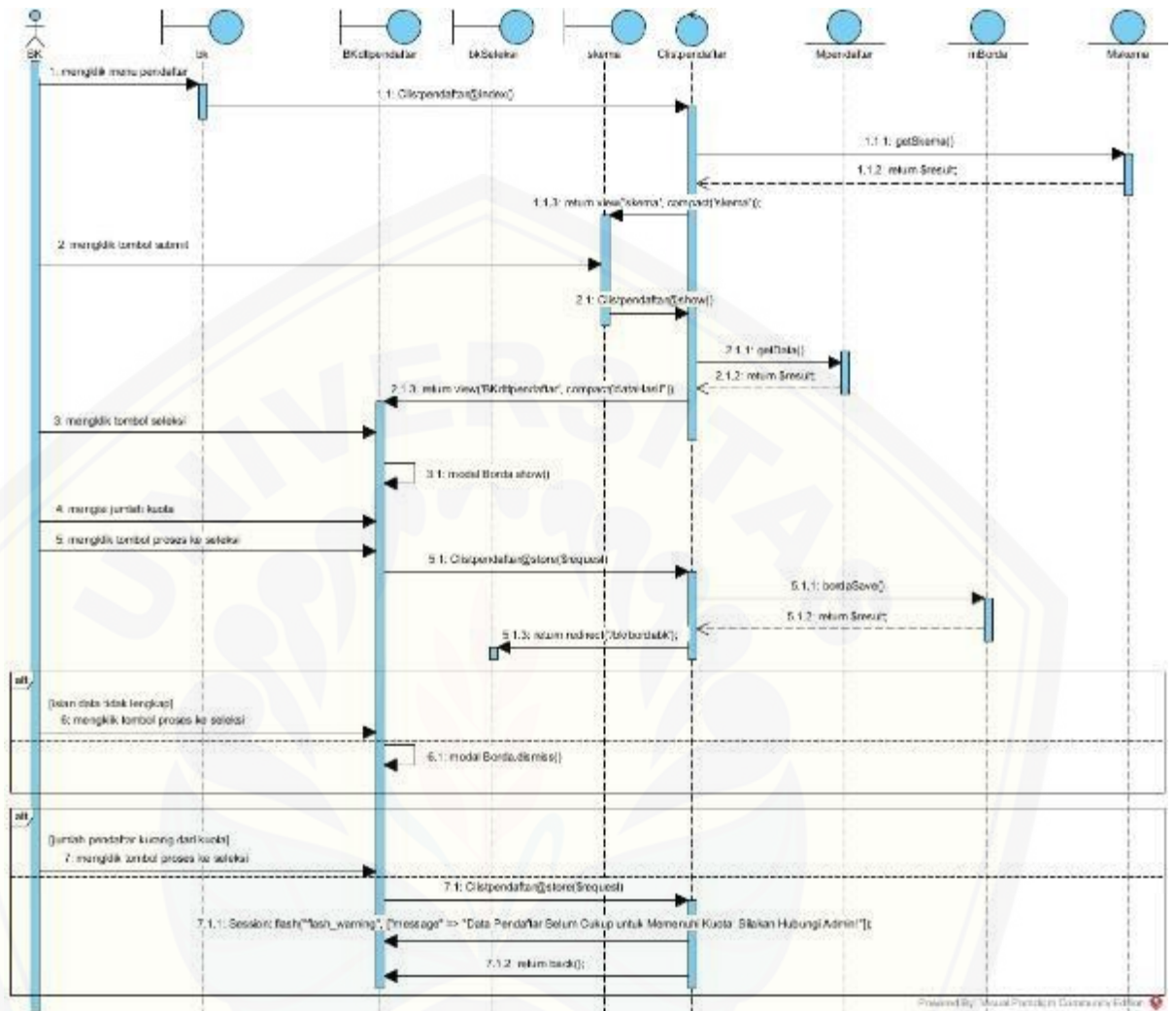
Gambar B.5. Sequence Diagram Mengelola Data Kriteria

**B.6. Sequence Diagram Menambah Skema Seleksi**



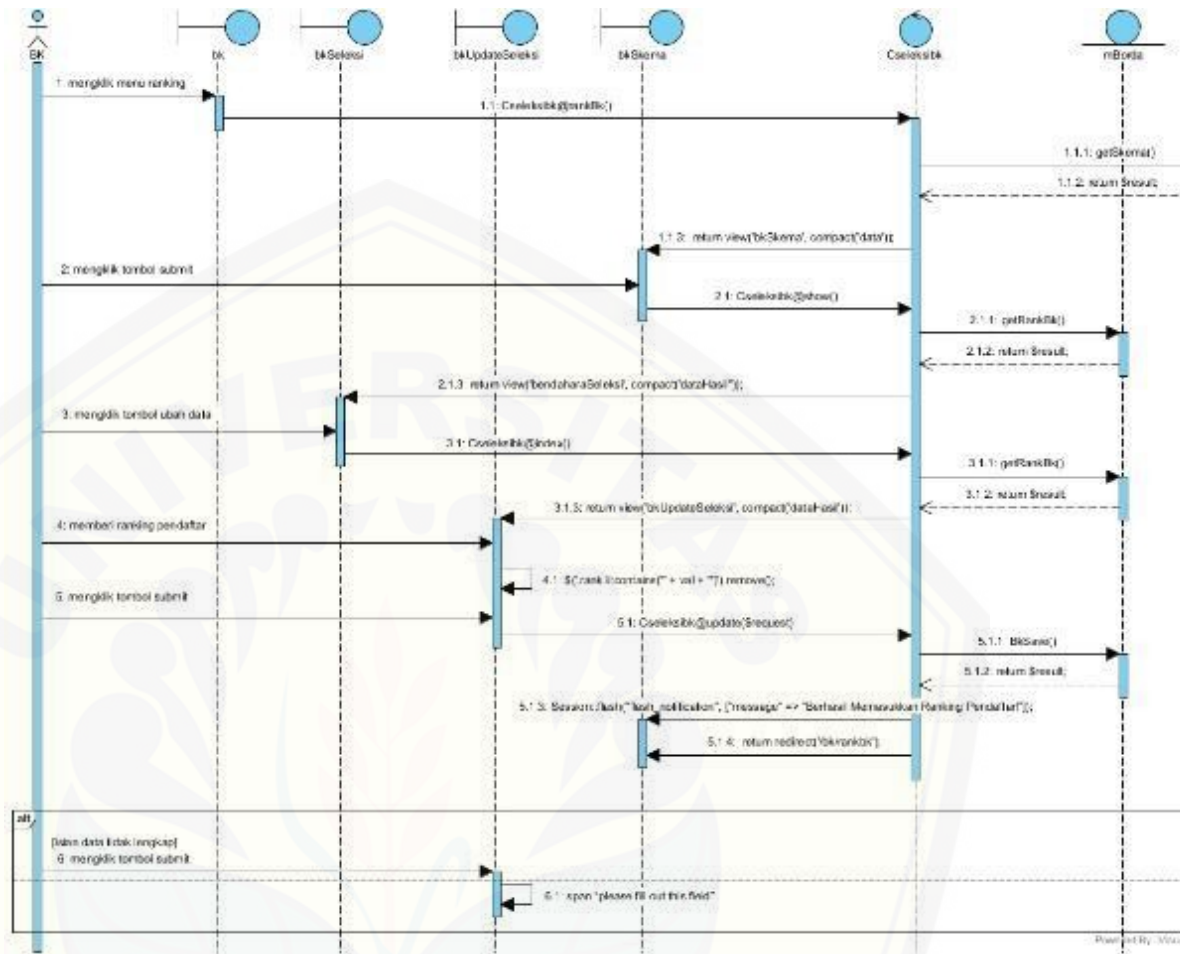
Gambar B.6. Sequence Diagram Menambah Skema Seleksi

**B.7. Sequence Diagram Menentukan Jumlah Kuota**



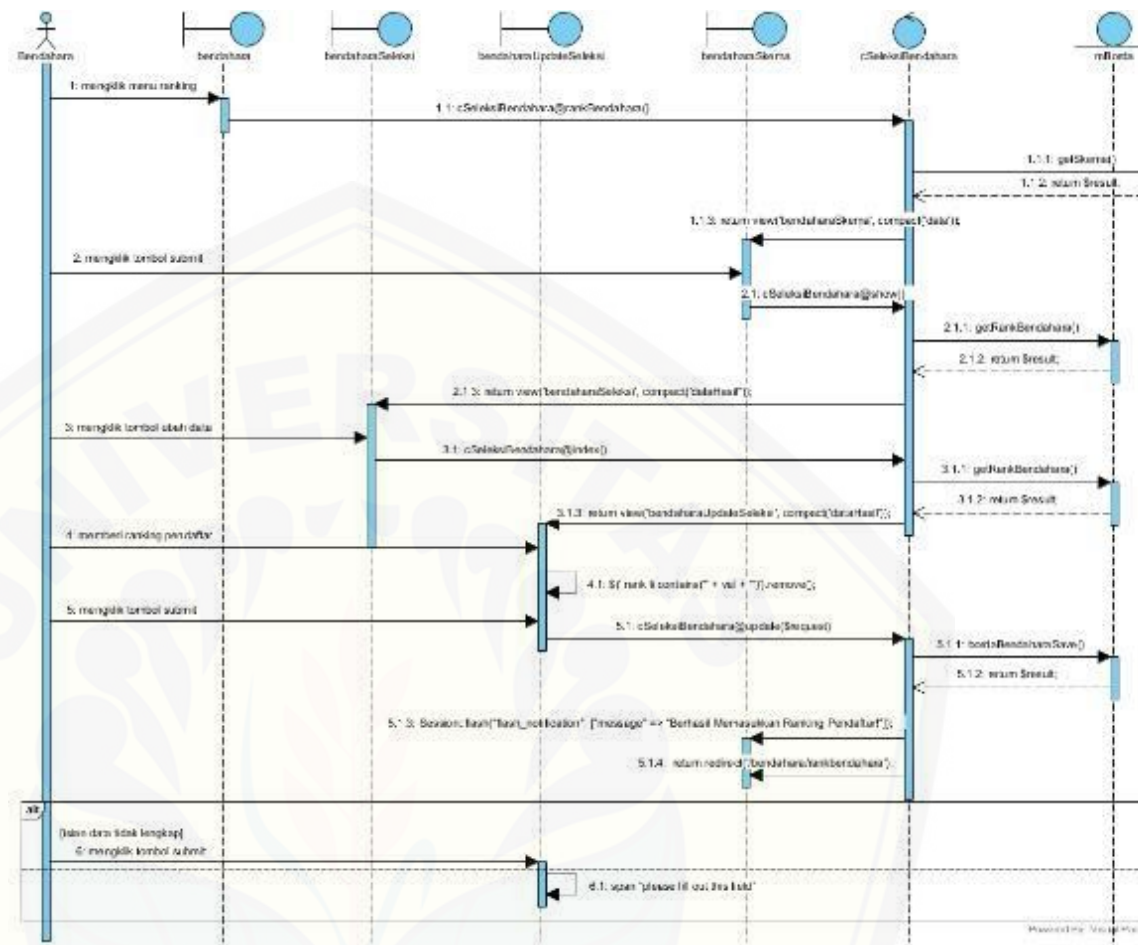
Gambar B.7. Sequence Diagram Menentukan Jumlah Kuota

**B.8. Sequence Diagram Mengelola Data Ranking**



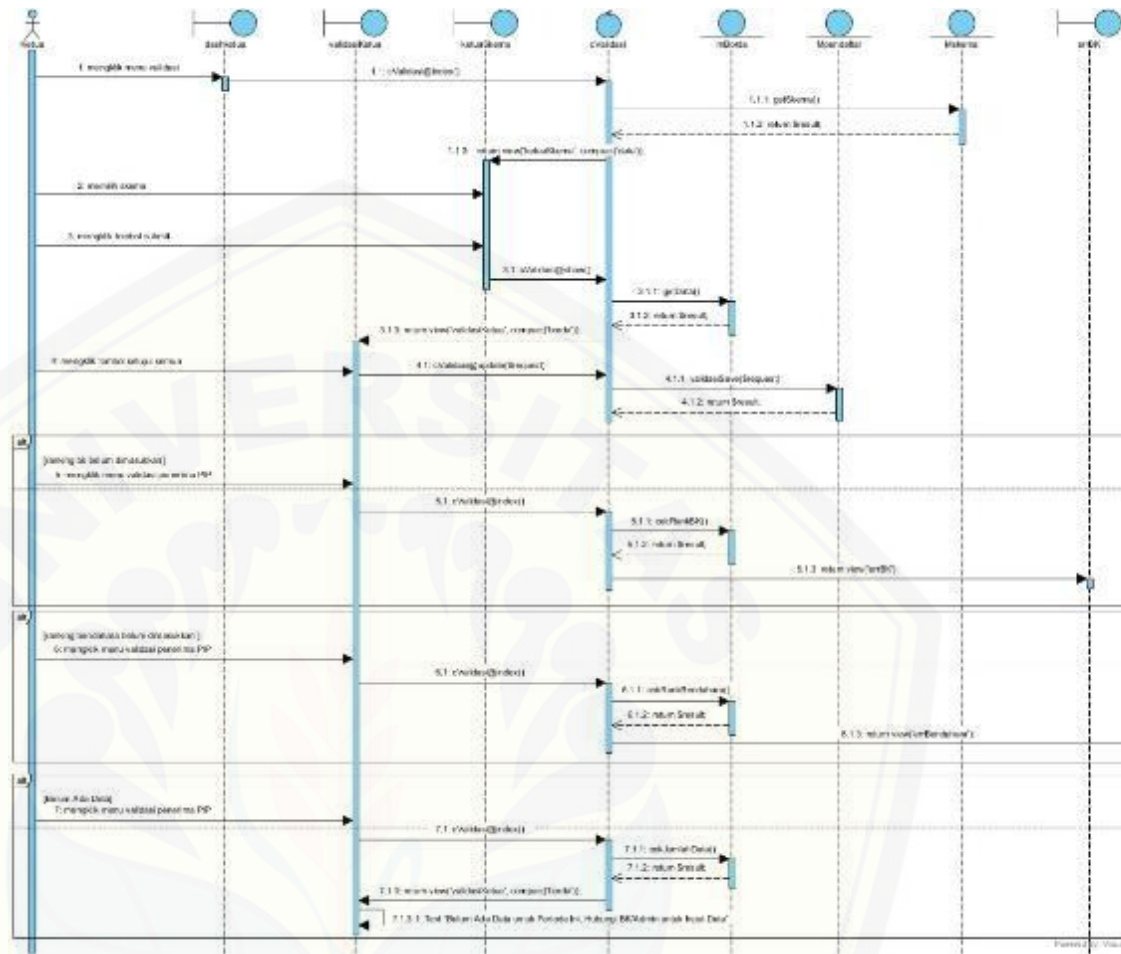
Gambar B.8. Sequence Diagram Mengelola Data Ranking





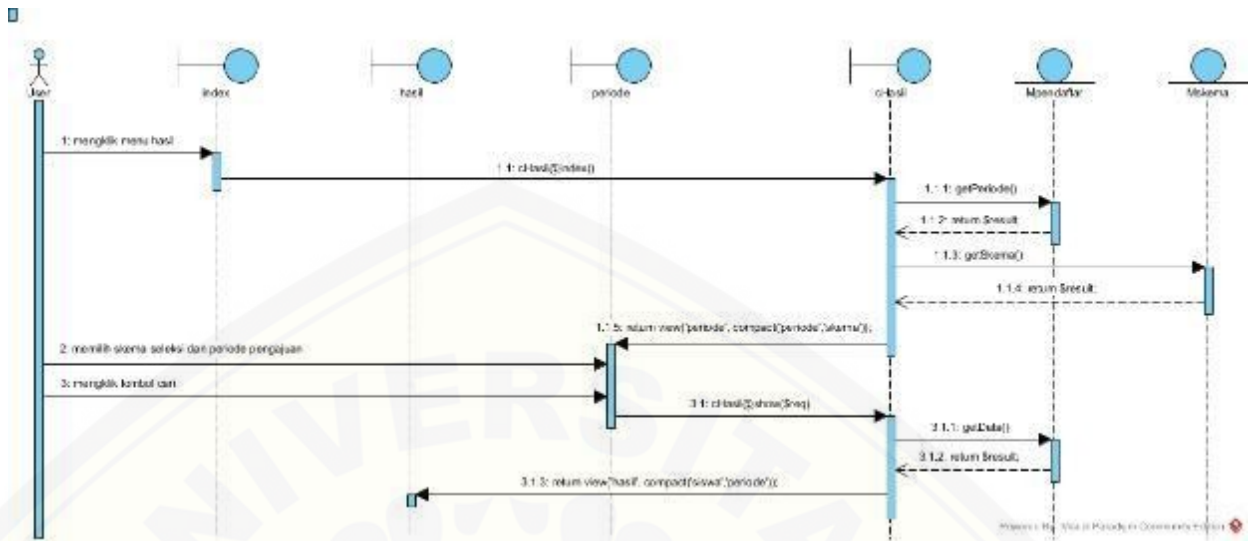
Gambar B.9. Sequence Diagram Bendahara Mengelola Data Ranking

**B.9. Sequence Diagram Validasi Hasil Rekomendasi**



Gambar B.10. Sequence Diagram Validasi Hasil Rekomendasi

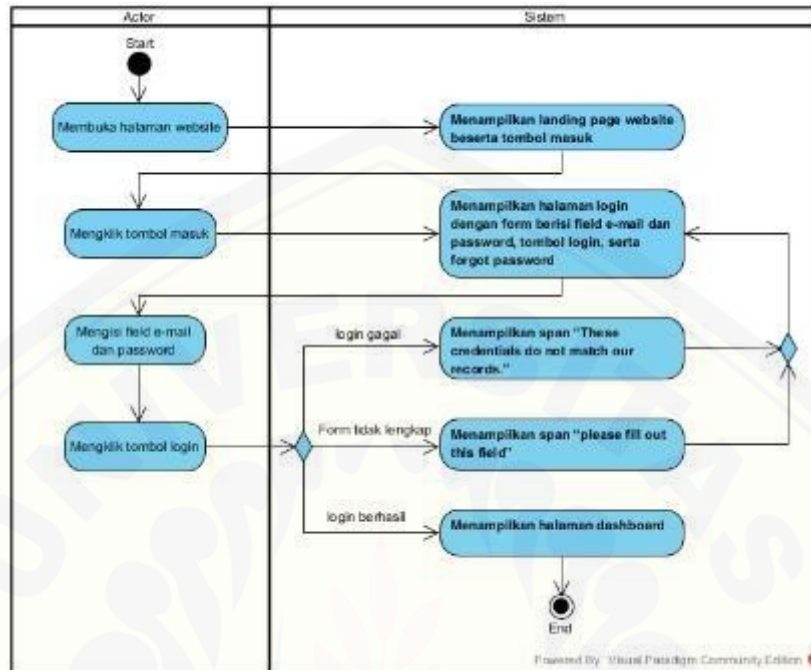
**B.10. Sequence Diagram Melihat Hasil Rekomendasi**



Gambar B.11. Sequence Diagram Melihat Hasil Rekomendasi

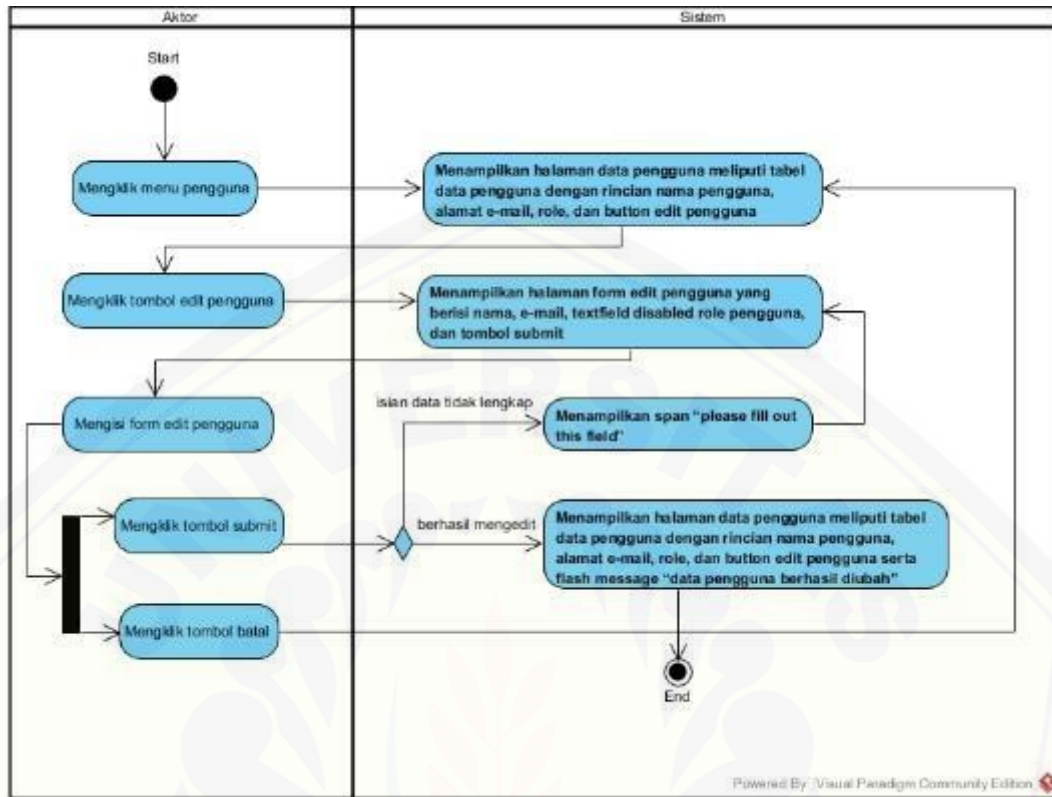
## C. Activity Diagram

### C.1. Activity Diagram Login



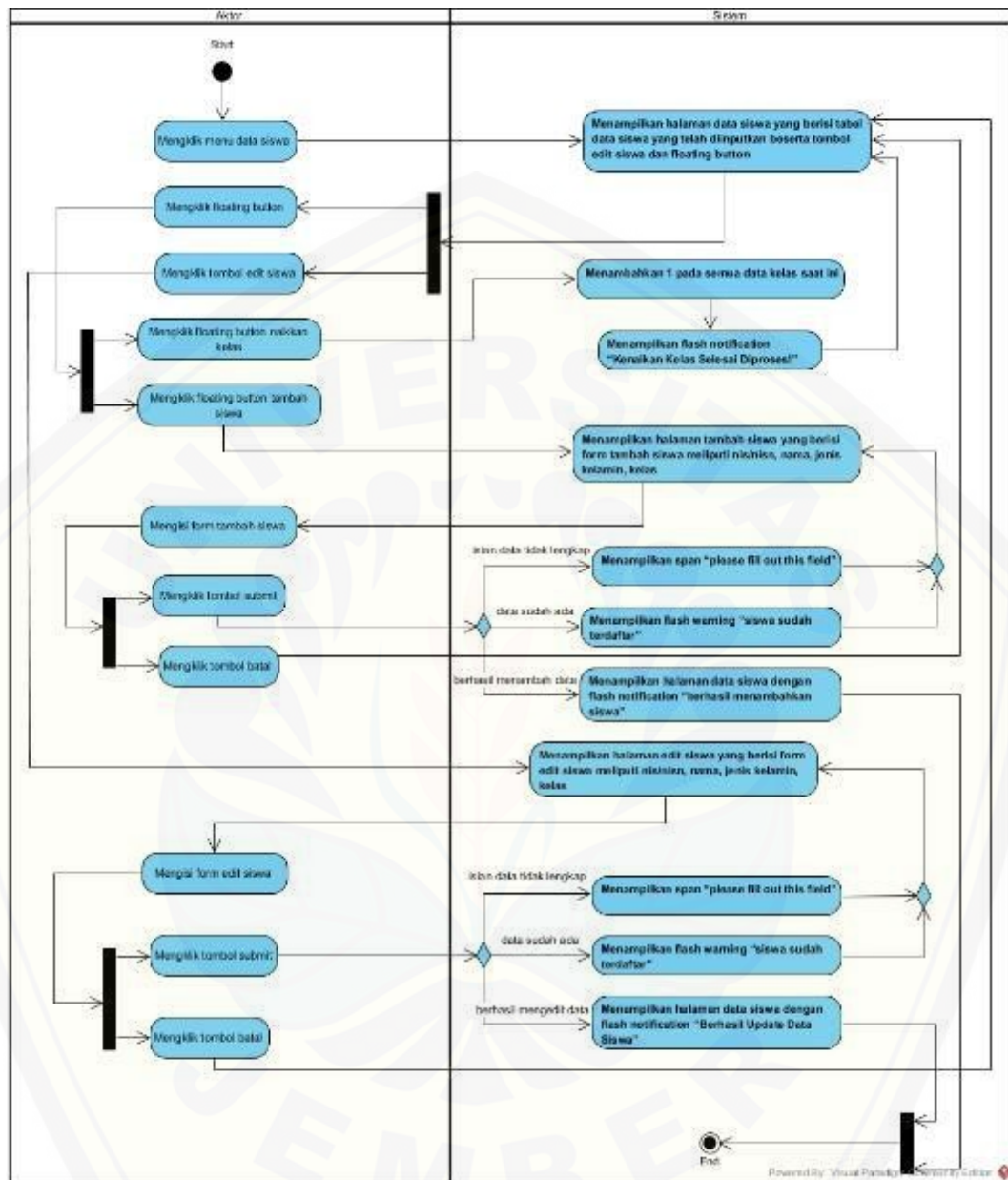
Gambar C.1. Activity Diagram Login

**C.2. Activity Diagram Mengelola Data Pengguna**



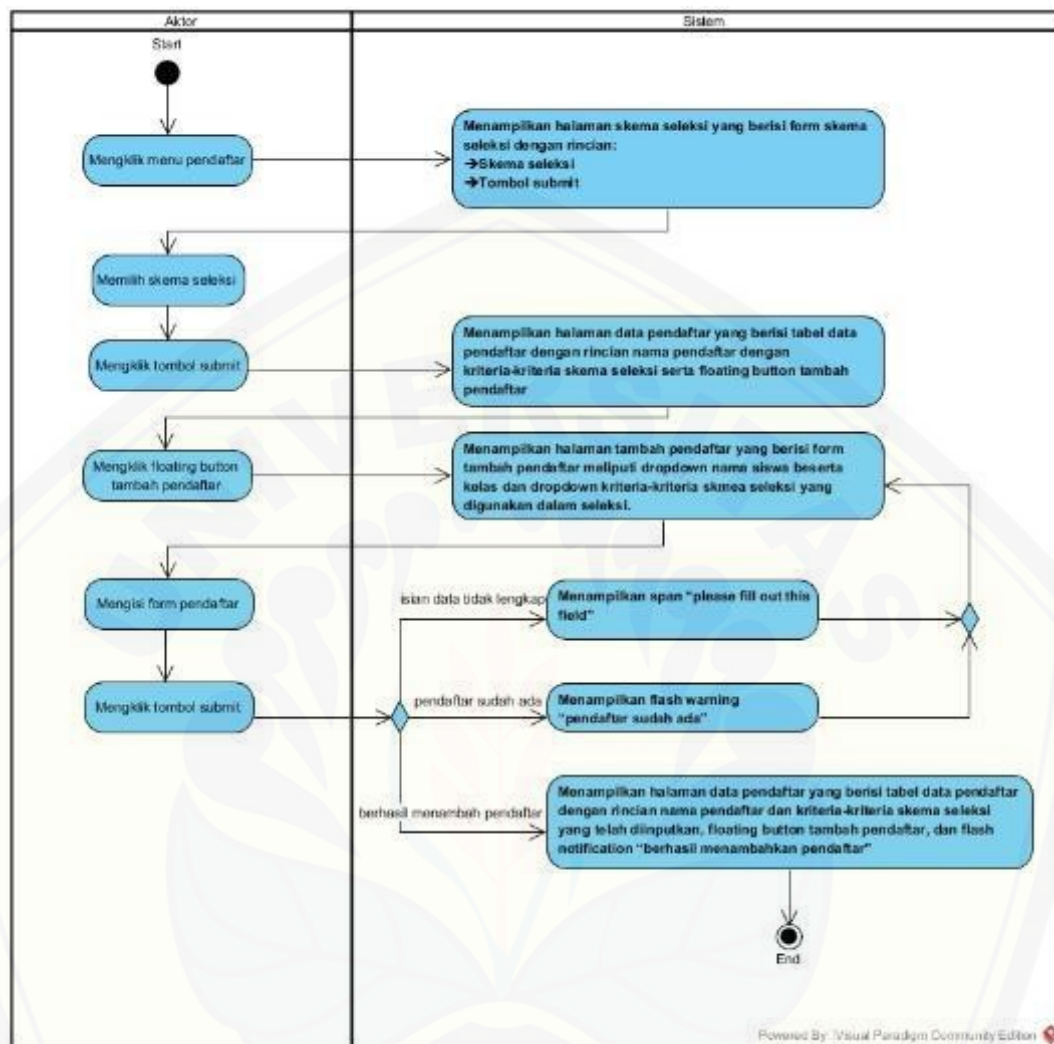
Gambar C.2. Activity Diagram Mengelola Data Pengguna

### C.3. Activity Diagram Mengelola Data Siswa



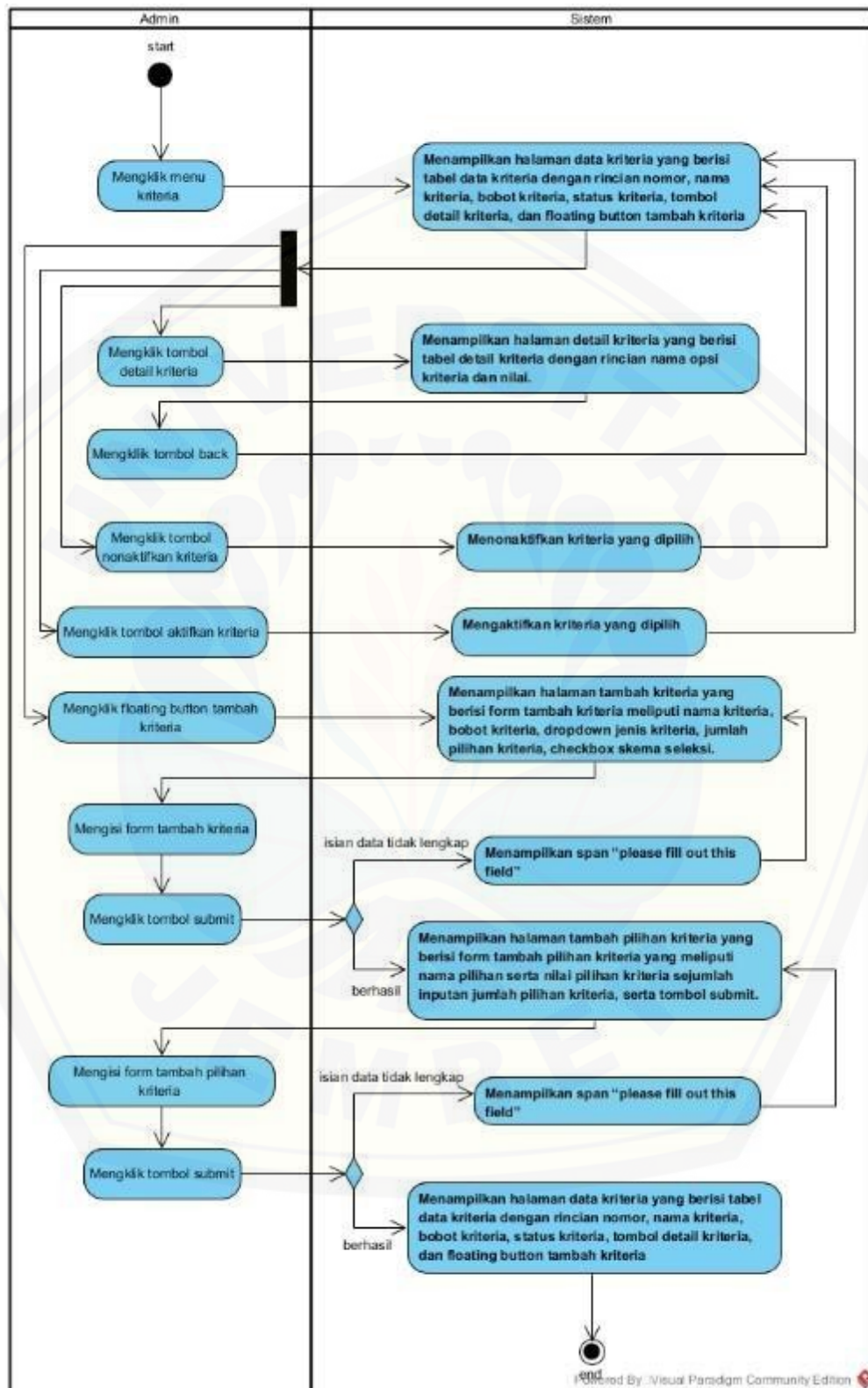
Gambar C.3. Activity Diagram Mengelola Data Siswa

### C.4. Activity Diagram Mengelola Data Pendaftar



Gambar C.4. Activity Diagram Mengelola Data Pendaftar

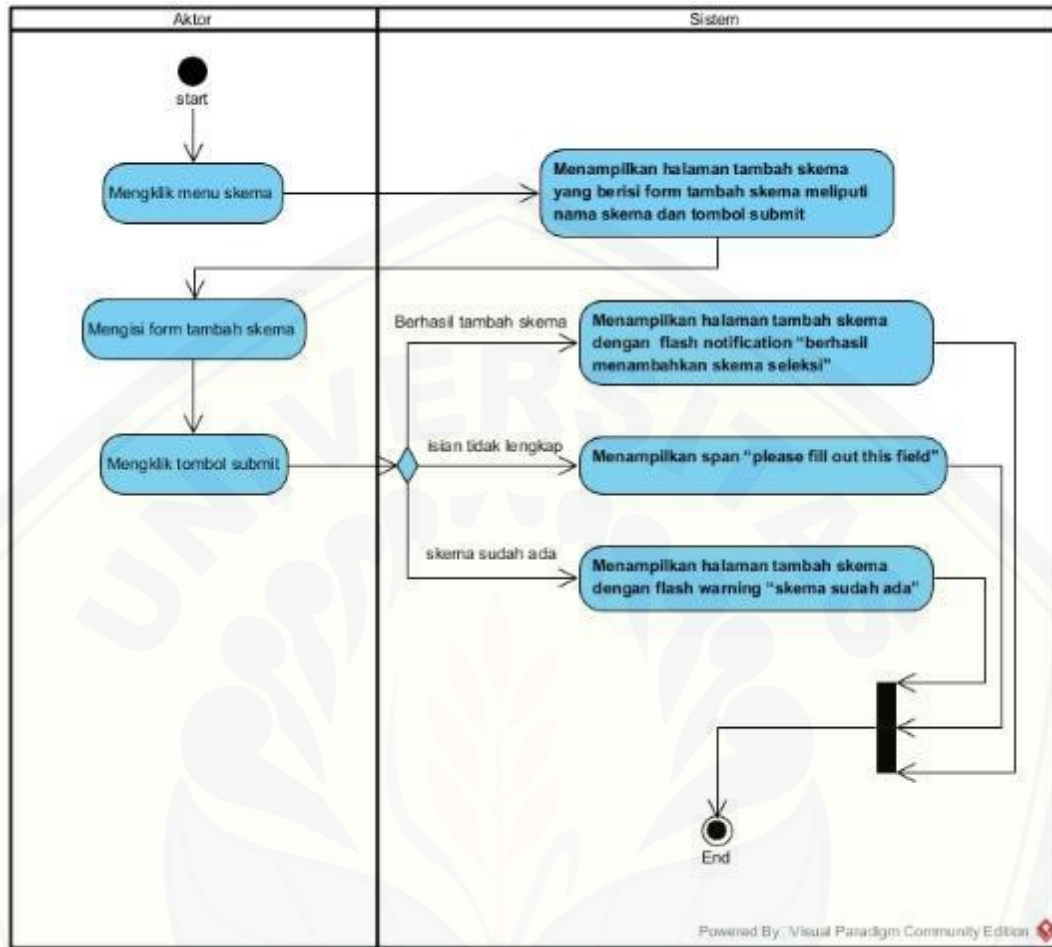
**C.5. Activity Diagram Mengelola Data Kriteria**



Gambar C.5. Activity Diagram Mengelola Data Kriteria

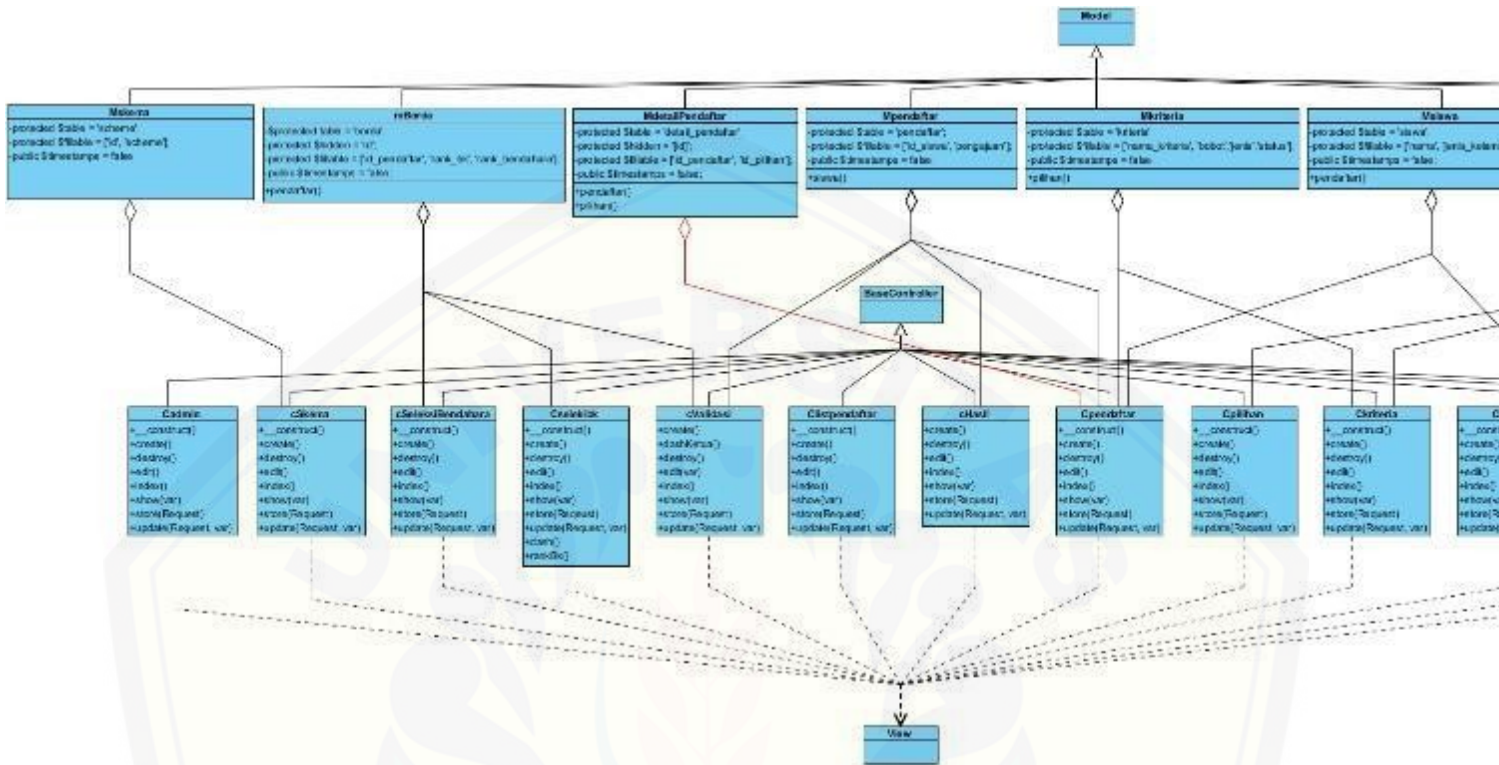


**C.6. Activity Diagram Menambah Data Skema**



Gambar C.6. Activity Diagram Menambah Data Skema

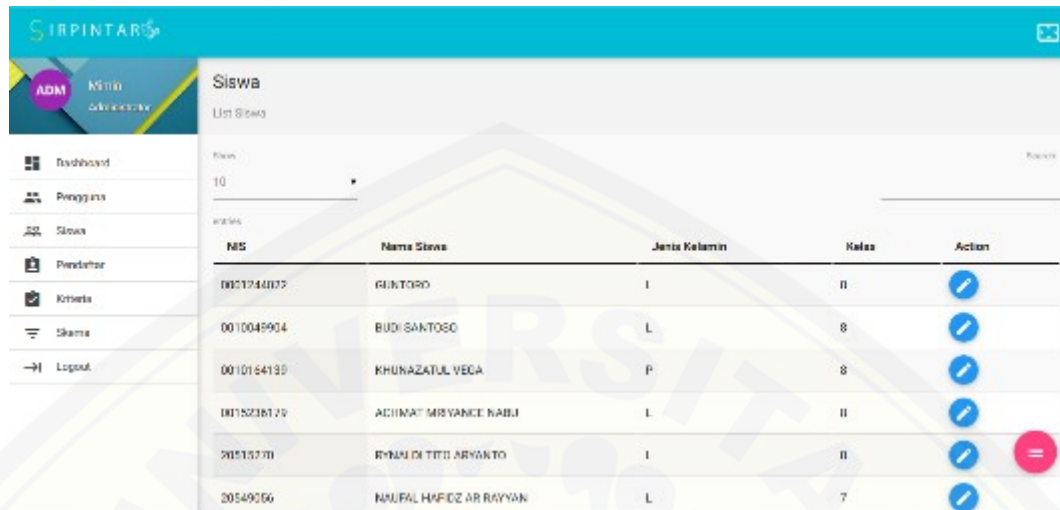
D. Class Diagram



Gambar D.1. Class Diagram

## E. Tampilan Sistem

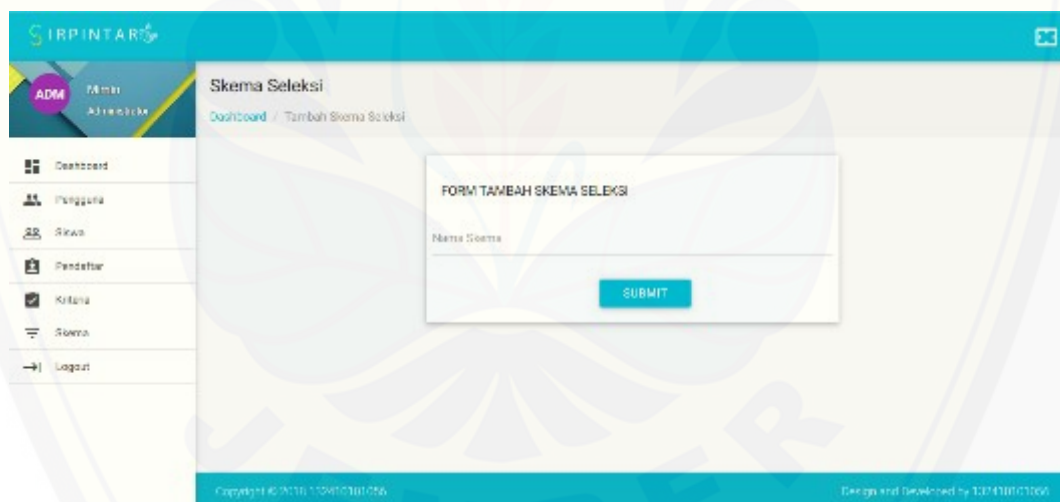
### D.1. Halaman Menu Siswa



NIS	Nama Siswa	Jenis Kelamin	Kelas	Action
0001744037	RIJUNTORO	I	II	
0010049904	BUDI SANTOSO	L	8	
0010164130	KHUNAZATUL VECA	P	8	
0015238120	ACHMAT MIZYANCE NABILI	L	II	
2001515770	RHYNDI TITO ARIYANTO	I	II	
20049006	NAUFAL HAFIDZ AR RAYWAN	L	7	

Gambar D.1. Halaman Menu Siswa

### D.2. Halaman Menu Skema



FORM TAMBAH SKEMA SELEKSI

Nama Skema

SUBMIT

Gambar D.2. Halaman Menu Skema

### D.3. Halaman Pilih Skema

IRPINTAR

BK BK IK

Skema Seleksi

Dashboard / Skema Seleksi

SKEMA SELEKSI

Program Indonesia Pintar

SUBMIT

Copyright © 2018 132410101056

Design and Developed by 132410101056

Gambar D.3. Halaman Pilih Skema

### D.4. Halaman Pilih Periode dan Skema

IRPINTAR

Utami Pengajuan Utami

Hasil Rekomendasi

Hasil Rekomendasi

PERIODE DAN SKEMA PENGAJUAN

Pilih Periode Pengajuan

Program Indonesia Pintar

CARI

Copyright © 2018 132410101056

Design and Developed by 132410101056

Gambar D.4. Halaman Pilih Periode dan Skema

### D.5. Hasil Perhitungan SAW

No	Nama	Status	Motor	Pegawai Swasta	Petani/Peternak	Nilai	Jumlah	Ranking
8	MUHAMMAD BINTANG MAULANA 7	Memili SKTM	Motor	Pegawai Swasta	Ibu Rumah Tangga	Rp. 500.000,-	2 Saudara	37.84561111
9	SAGUS RIDWAN 7	Memili KIP/KPS dan Menupakan Anggota PKH	Motor	Buruh/Tukang/Senibulan	Petani/Peternak	Rp. 500.000,-	1 Saudara	25.93194444
10	HAUFAL HADDFAR RAYHAN 7	Tidak Memili KIP/KPS dan Bukan Anggota PKH	Mobil atau Motor	PKS	Pedagang/Pengusaha/Wirausaha	Rp. 2.000.000,- Samud Rp. 1.000.000,-	1 Saudara	31.55665555
11	ARBA ATUS ANIN 7	Tidak Memili KIP/KPS dan Bukan Anggota PKH	Motor	Petani/Peternak	Ibu Rumah Tangga	Rp. 500.000,-	4 Abu Luthi	20.83194444

No	Nama	Status	Motor	Pegawai Swasta	Petani/Peternak	Nilai	Jumlah	Ranking
12	MUHAMMAD MAHARRI ABDILLAH 8	Tidak Memili KIP/KPS dan Bukan Anggota PKH	Motor	Perdagangan/Almarhum	Ibu Rumah Tangga	Rp. 500.000,-	3 Saudara	15.63081111
13	GUNTORO - II	Tidak Memili KIP/KPS dan Bukan Anggota PKH	Motor Lebih Dari 1	Pedagang/Pengusaha/Wirausaha	Ibu Rumah Tangga	Rp. 1.000.000,- Samud Rp. 2.000.000,-	2 Saudara	11.87777777
14	M. ALWI ZANZAMI 7	Tidak Memili KIP/KPS dan Bukan Anggota PKH	Motor	Pegawai Swasta	Pegawai Swasta	Rp. 500.000,- Samud Rp. 1.000.000,-	2 Saudara	11.8075
15	ANGGIRMA ALUA 7	Tidak Memili KIP/KPS dan Bukan Anggota PKH	Motor	Pedagang/Pengusaha/Wirausaha	Petani/Peternak	Rp. 500.000,-	2 Saudara	11.8075

No	Nama	Status	Motor	Profesi	Alamat	Role	Count	Value
16	DAVID CAJARI BIANTO - 7	Tidak Memiliki KIP/KPS dan Bukan Anggota PKI	Motor	Pedagang/Pengusaha/Wiraswasta	Ibu Rumah Tangga	RP. 500.000,-	2 Saudara	11.415277777
17	ROYALDI TITO ARWAKTO - 8	Tidak Memiliki KIP/KPS dan Bukan Anggota PKI	Motor	Pedagang/Pengusaha/Wiraswasta	Pensiunan/Amarham	Rp. 2.000.000,- Sampai Rp 3.000.000,-	1 Saudara	6.854166666
18	AHMAD INGGI - 7	Tidak Memiliki KIP/KPS dan Bukan Anggota PKI	Motor	Pegawai Swasta	Pegawai Swasta	Rp. 500.000,- Sampai Rp 1.000.000,-	1 Saudara	6.4875
19	AHMAD DAVID MAFANI - 7	Tidak Memiliki KIP/KPS dan Bukan Anggota PKI	Motor	Pedagang/Pengusaha/Wiraswasta	Ibu Rumah Tangga	RP. 500.000,-	1 Saudara	6.415277777
20	ADITYA BAGAS SAPITRA - 7	Tidak Memiliki KIP/KPS dan Bukan Anggota PKI	Motor	Buruh/Tukang/Serabutan	Pedagang/Pengusaha/Wiraswasta	RP. 500.000,-	1 Saudara	6.416277777

Gambar D.5. Hasil Perhitungan SAW

**F. Pengujian Black Box**

Tabel E.1. Pengujian Black Box

No	Fitur	Aksi	Hasil	Keterangan
1.	Mengelola data pengguna	Klik menu pengguna	Menampilkan halaman data pengguna meliputi tabel data pengguna dengan rincian nama pengguna, alamat <i>e-mail</i> , role, dan tombol edit pengguna	Berhasil
		Klik tombol <i>edit</i> pengguna	Menampilkan halaman form edit pengguna yang berisi nama, e-mail, <i>textfield disabled</i>	Berhasil

			<i>role</i> pengguna, dan tombol submit	
		Klik tombol <i>submit</i> pada <i>form</i> edit pengguna	Menyimpan ke database lalu menampilkan halaman data pengguna	Berhasil
		Klik tombol batal pada <i>form</i> edit pengguna	Menampilkan halaman data pengguna	Berhasil
		Klik tombol <i>submit</i> pada <i>form</i> edit pengguna saat isian <i>form</i> tidak lengkap	Menampilkan span " <i>please fill out this field</i> "	Berhasil
2.	Mengelola data siswa	Klik menu siswa	Menampilkan halaman data siswa yang berisi tabel data siswa yang telah diinputkan beserta tombol edit siswa dan <i>floating button</i>	Berhasil
		Arahkan kursor ke <i>floating button</i>	Menampilkan <i>floating button</i> naikkan kelas dan <i>floating button</i> tambah siswa	Berhasil
		Klik <i>floating button</i> tambah siswa	Menampilkan halaman tambah siswa yang berisi form tambah siswa meliputi nis/nisn,	Berhasil

			nama, jenis kelamin, kelas	
		Klik tombol submit pada <i>form</i> tambah siswa	Menyimpan ke database lalu menampilkan halaman data siswa dengan <i>flash notification</i> "berhasil menambahkan siswa"	Berhasil
		Klik tombol batal pada <i>form</i> tambah siswa	menampilkan halaman data siswa	Berhasil
		Klik tombol submit pada <i>form</i> tambah siswa saat isian data tidak lengkap	Menampilkan span " <i>please fill out this field</i> "	Berhasil
		Klik tombol submit pada <i>form</i> tambah siswa saat data siswa yang dimasukkan sudah ada di database	Menampilkan halaman tambah siswa dengan <i>flash warning</i> "siswa sudah terdaftar"	Berhasil
		Klik tombol <i>edit</i> siswa	Menampilkan halaman <i>edit</i> siswa yang berisi form edit siswa meliputi nis/nisn, nama, jenis kelamin, kelas	Berhasil



		Klik tombol submit pada <i>form edit</i> siswa	Menyimpan ke database lalu menampilkan halaman data siswa dengan <i>flash notification</i> "Berhasil Update Data Siswa"	Berhasil
		Klik tombol batal pada <i>form edit</i> siswa	Menampilkan halaman tambah siswa yang berisi form tambah siswa meliputi nis/nisn, nama, jenis kelamin, kelas	Berhasil
		Klik tombol submit pada <i>form edit</i> siswa saat isian data tidak lengkap	Menampilkan span " <i>please fill out this field</i> "	Berhasil
		Klik tombol submit pada <i>form edit</i> siswa saat data yang dimasukkan sudah ada di database	Menampilkan halaman edit siswa dengan <i>flash warning</i> "siswa sudah terdaftar"	Berhasil
		Arahkan kursor ke <i>floating button</i>	Menampilkan <i>floating button</i> naikkan kelas dan <i>floating button</i> tambah siswa	Berhasil

		Mengklik <i>floating button</i> naikkan kelas	Menambahkan 1 pada semua data kelas saat ini didatabase lalu menampilkan halaman data siswa dengan <i>flash notification</i> "Kenaikan Kelas Selesai Diproses!"	Berhasil
3.	Mengelola data pendaftar	Klik menu pendaftar	Menampilkan halaman skema seleksi dengan pilihan skema seleksi yang telah dimasukkan	Berhasil
		Pilih skema seleksi lalu klik tombol submit	Menampilkan halaman data pendaftar yang berisi tabel data pendaftar dengan rincian nama pendaftar dan kriteria-kriteria PIP serta <i>floating button</i> tambah pendaftar	Berhasil
		Klik <i>floating button</i> tambah pendaftar	Menampilkan halaman tambah pendaftar yang berisi form tambah pendaftar meliputi nama siswa beserta kelas, <i>dropdown</i> dan <i>dropdown</i> kriteria-kriteria PIP yang digunakan dalam seleksi.	Berhasil

		Klik tombol submit pada <i>form</i> tambah pendaftar	Menyimpan ke database lalu menampilkan halaman data pendaftar yang berisi tabel data pendaftar dengan rincian nama pendaftar dan kriteria-kriteria PIP yang telah diinputkan, <i>floating button</i> tambah pendaftar, dan <i>flash notification</i> “berhasil menambahkan pendaftar”	Berhasil
		Klik tombol submit pada <i>form</i> tambah pendaftar saat isian data tidak lengkap	Menampilkan span “Data Wajib Dipilih!”	Berhasil
		Klik tombol submit pada <i>form</i> tambah pendaftar saat data sudah ada di database	Menampilkan halaman tambah pendaftar dengan <i>flash warning</i> “pendaftar sudah ada”	Berhasil
4.	Mengelola data kriteria	Klik menu kriteria	Menampilkan halaman data kriteria yang berisi tabel data kriteria dengan rincian nomor, nama kriteria, bobot	Berhasil

			kriteria, status kriteria, tombol detail kriteria, dan <i>floating button</i> tambah kriteria	
		Klik tombol detail kriteria	Menampilkan halaman detail kriteria yang berisi tabel detail kriteria dengan rincian nama opsi kriteria dan nilai serta tombol <i>back</i>	Berhasil
		Klik tombol <i>back</i>	Menampilkan halaman data kriteria yang berisi tabel data kriteria dengan rincian nomor, nama kriteria, bobot kriteria, status kriteria, tombol detail kriteria, dan <i>floating button</i> tambah kriteria	Berhasil
		Klik tombol nonaktifkan kriteria	Menonaktifkan kriteria yang terpilih lalu menampilkan halaman data kriteria yang berisi tabel data kriteria dengan rincian nomor, nama kriteria, bobot kriteria, status kriteria, tombol detail kriteria, dan <i>floating button</i> tambah kriteria	Berhasil

		Klik tombol aktifkan kriteria	Mengaktifkan kriteria yang dipilih lalu menampilkan halaman data kriteria yang berisi tabel data kriteria dengan rincian nomor, nama kriteria, bobot kriteria, status kriteria, tombol detail kriteria, dan <i>floating button</i> tambah kriteria	Berhasil
		Klik <i>floating button</i> tambah kriteria	Menampilkan halaman tambah kriteria yang berisi form tambah kriteria meliputi nama kriteria, bobot kriteria, jenis kriteria jumlah pilihan kriteria, serta <i>checkbox</i> kema seleksi.	Berhasil
		Klik tombol submit pada <i>form</i> tambah kriteria	Menyimpan ke database lalu menampilkan halaman tambah pilihan kriteria yang berisi form tambah pilihan kriteria yang meliputi nama pilihan serta nilai pilihan kriteria sejumlah inputan	Berhasil

			jumlah pilihan kriteria, serta tombol submit.	
		Klik tombol submit pada <i>form</i> tambah pilihan kriteria	Menyimpan ke database lalu menampilkan halaman data kriteria yang berisi tabel data kriteria dengan rincian nomor, nama kriteria, bobot kriteria, status kriteria, tombol detail kriteria, dan <i>floating button</i> tambah kriteria	Berhasil
		Klik tombol submit pada <i>form</i> tambah kriteria saat isian data tidak lengkap	Menampilkan span "please fill out this field"	Berhasil
		Klik tombol submit pada <i>form</i> tambah pilihan kriteria saat isian data tidak lengkap	Menampilkan span "please fill out this field"	Berhasil
5.	Menambah skema seleksi	Klik menu skema	Menampilkan halaman tambah skema dengan <i>form</i> tambah skema yang berisi <i>fields</i> skema seleksi serta tombol submit	Berhasil

		Klik tombol submit	Menampilkan <i>flash message</i> “Berhasil menambahkan skema seleksi”	Berhasil
		Klik tombol submit saat isian data sama dengan didatabse	Menampilkan <i>flash warning</i> Skema Sudah ADA!”	Berhasil
		Klik tombol submit saat isian data tidak lengkap	Menampilkan span “please fill out this field”	Berhasil
6.	Mengelola ranking pendaftar (bendahara)	Klik menu ranking	Menampilkan halaman skema seleksi	Berhasil
		Pilih skema seleksi lalu klik tombol submit	Menampilkan halaman data pendaftar yang berisi tabel data pendaftar sejumlah kuota periode tersebut terurut berdasarkan nilai Metode SAW beserta <i>ranking</i> dan tombol ubah data	Berhasil
		Klik tombol ubah data	Menampilkan halaman tambah ranking pendaftar yang berisi	Berhasil

			table pendaftar serta <i>dropdown</i> ranking	
		Isi ranking pendaftar (pilih dari <i>dropdown</i> )	Menghapus pilihan <i>ranking</i> yang telah dipilih	Berhasil
		Klik tombol submit	Menyimpan ke database lalu menampilkan halaman data ranking pendaftar dengan <i>flash notification</i> “berhasil memasukkan ranking pendaftar”	Berhasil
		Mengklik tombol <i>submit</i> saat isian data tidak lengkap	Menampilkan span “ <i>please fill out this field</i> ”	Gagal menampilkan span
7.	Mengelola ranking pendaftar (BK)	Klik menu ranking	Menampilkan halaman skema seleksi	Berhasil
		Pilih skema seleksi lalu klik tombol submit	Menampilkan halaman data pendaftar yang berisi tabel data pendaftar sejumlah kuota periode tersebut terurut berdasarkan nilai Metode SAW beserta <i>ranking</i> dan tombol ubah data	Berhasil



		Klik tombol ubah data	Menampilkan halaman tambah ranking pendaftar yang berisi table pendaftar serta <i>dropdown ranking</i>	Berhasil
		Isi ranking pendaftar (pilih dari <i>dropdown</i> )	Menghapus pilihan <i>ranking</i> yang telah dipilih	Berhasil
		Klik tombol submit	Menyimpan ke database lalu menampilkan halaman data ranking pendaftar dengan <i>flash notification</i> "berhasil memasukkan ranking pendaftar"	Berhasil
		Mengklik tombol <i>submit</i> saat isian data tidak lengkap	Menampilkan span " <i>please fill out this field</i> "	Gagal menampilkan span
8.	Validasi hasil rekomendasi	Klik menu validasi	Menampilkan halaman skema seleksi	Berhasil
		Pilih skema seleksi lalu klik tombol submit	Menampilkan halaman validasi hasil rekomendasi yang berisi tabel data pendaftar yang telah direkomendasikan oleh guru bk dan bendahara dengan rincian nama,	Berhasil

			kelas, kriteria, poin borda, keterangan, serta tombol setuju semua bila data belum divalidasi	
		Klik tombol setuju semua	Menampilkan halaman validasi hasil rekomendasi yang berisi tabel data pendaftar yang telah direkomendasikan oleh bk dan bendahara dengan rincian nama, kelas, kriteria, poin borda, keterangan telah disetujui	Berhasil
		Klik tombol submit saat ranking bk belum dimasukkan	Menampilkan halaman error "Ranking BK Belum Dimasukkan"	Berhasil
		Klik tombol submit saat ranking bendahara belum dimasukkan	Menampilkan halaman error "Ranking Bendahara Belum Dimasukkan"	Berhasil
		Klik menu validasi saat belum ada data	Menampilkan halaman validasi hasil rekomendasi dengan	Berhasil

			tulisan “Belum Ada Data untuk Periode Ini, Hubungi BK/Admin untuk Input Data”	
9.	Melihat hasil rekomendasi (pengguna umum)	Klik menu hasil	Menampilkan halaman pilih periode rekomendasi dengan rincian <i>drop-down</i> periode pengajuan, skema seleksi, tombol cari, serta <i>side menu</i> Home Sirpintar	Berhasil
		Pilih periode pengajuan dan skema seleksi lalu klik cari	Menampilkan halaman hasil rekomendasi yang berisi table pendaftar yang disetujui menerima PIP dengan rincian nama dan kelas, tombol <i>back</i> serta <i>side menu</i> Home Sirpintar	Berhasil
10.	Login	Buka halaman website	Menampilkan <i>landing page website</i> beserta tombol masuk	Berhasil
		Klik tombol masuk	Menampilkan halaman login dengan form berisi <i>field</i> e-mail dan password, tombol login, serta <i>forgot password</i>	Berhasil
		Isi <i>form</i> lalu klik tombol login	Menampilkan halaman dashboard	Berhasil

		Klik tombol login saat <i>e-mail</i> atau <i>password</i> salah	Menampilkan pesan teks “ <b><i>These credentials do not match our records.</i></b> ”	Berhasil
		Klik tombol login saat isian data tidak lengkap	Menampilkan span “ <i>please fill out this field</i> ”	Berhasil

