

**PENGEMBANGAN SISTEM INFORMASI MANAJEMEN BEASISWA DI  
FAKULTAS MIPA UNIVERSITAS LAMPUNG MENGGUNAKAN  
GABUNGAN METODE AHP DAN TOPSIS**

**(Skripsi)**

**Oleh**

**FEBRASARI ALMANIA**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2018**

## ABSTRAK

### PENGEMBANGAN SISTEM INFORMASI MANAJEMEN BEASISWA DI FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM UNIVERSITAS LAMPUNG MENGGUNAKAN GABUNGAN METODE AHP DAN TOPSIS

Oleh

**FEBRASARI ALMANIA**

Para pengelola beasiswa di Fakultas MIPA Universitas Lampung selama ini kesulitan menyalin isi berkas pelamar ke dalam *spreadsheet*. Oleh karena itu, dikembangkan sistem informasi berbasis komputer yang mampu mempermudah kegiatan digitalisasi tersebut. Bersama metode AHP dan TOPSIS, sistem bahkan sanggup menghitung bobot-bobot kriteria bermodalkan intensitas dominasi suatu kriteria terhadap kriteria lain serta menyeleksi para pendaftar secara otomatis. Sistem dibangun dengan memanfaatkan *framework* Odoo lalu diuji menggunakan teknik *equivalence class partitioning*. Keluaran-keluaran sistem untuk 294 *test case* membuktikan bahwa sistem sukses memenuhi spesifikasi yang dicanangkan. Akurasi estimasi bobot pun terbilang baik, di mana selisih hasil kalkulasi sistem dengan manual sangat kecil. Melalui penyempurnaan lebih lanjut, sistem berpeluang besar menyumbang manfaat bagi aktivitas manajemen beasiswa di Fakultas MIPA Universitas Lampung.

Kata kunci: AHP, *Analytic Hierarchy Process*, manajemen beasiswa, MCDM, Odoo, pengambilan keputusan, *Technique for Order of Preference by Similarity to Ideal Solution*, TOPSIS.

## **ABSTRACT**

### **THE DEVELOPMENT OF COMPUTER-BASED SCHOLARSHIP MANAGEMENT SYSTEM AT FACULTY OF MATHEMATICS AND NATURAL SCIENCES OF UNIVERSITY OF LAMPUNG USING COMBINATION OF AHP AND TOPSIS**

**By**

**FEBRASARI ALMANIA**

The organizers of scholarships at the Faculty of Mathematics and Natural Sciences of Lampung University have been troubled by copying the contents of applicant's files into the spreadsheet. Therefore, a computer-based information system was developed to assist them in the digitalization. By utilizing AHP and TOPSIS method, the system can even calculate criteria weights based on the dominance of a criterion against the others as well as select the grantees automatically. The system was built using the Odoo framework and then tested using the equivalence class partitioning technique. Its outcomes for 294 test cases prove that the system meets the specification that had been set. The accuracy of weight estimation was fairly good, where the difference between system and manual result was very small. Through further refinement, the system is most likely to benefit people that involved in scholarship management activities at the Faculty of Mathematics and Natural Sciences, University of Lampung.

**Keywords:** AHP, Analytic Hierarchy Process, decision making, MCDM, Odoo, scholarship management, Technique for Order of Preference by Similarity to Ideal Solution, TOPSIS.

**PENGEMBANGAN SISTEM INFORMASI MANAJEMEN BEASISWA DI  
FAKULTAS MIPA UNIVERSITAS LAMPUNG MENGGUNAKAN  
GABUNGAN METODE AHP DAN TOPSIS**

Oleh

**FEBRASARI ALMANIA**

Skripsi

sebagai salah satu syarat untuk mencapai gelar  
**SARJANA KOMPUTER**

pada

**Jurusan Ilmu Komputer  
Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2018**

Judul Skripsi

: PENGEMBANGAN SISTEM INFORMASI  
MANAJEMEN BEASISWA DI FAKULTAS  
MIPA UNIVERSITAS LAMPUNG  
MENGUNAKAN GABUNGAN METODE  
AHP DAN TOPSIS

Nama Mahasiswa

: Febrasari Almania

Nomor Pokok Mahasiswa

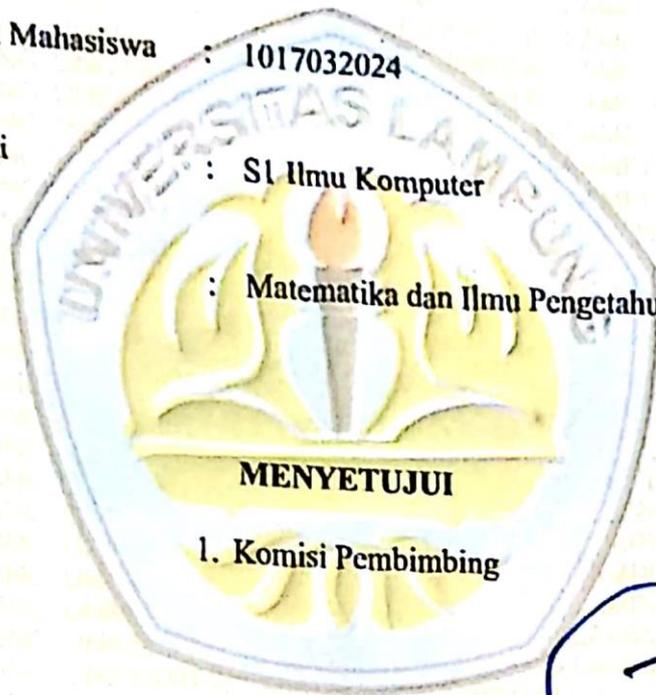
: 1017032024

Program Studi

: SI Ilmu Komputer

Fakultas

: Matematika dan Ilmu Pengetahuan Alam



*Dr. Ir. Kurnia Muludi*

**Dr. Ir. Kurnia Muludi, M.S.Sc.**  
NIP. 19640616 198902 1 001

*Aristoteles*

**Aristoteles, S.Si., M.Si.**  
NIP. 19810521 200604 1 002

2. Ketua Jurusan Ilmu Komputer  
FMIPA Universitas Lampung

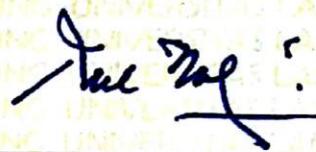
*Dr. Ir. Kurnia Muludi*

**Dr. Ir. Kurnia Muludi, M.S.Sc.**  
NIP. 19640616 198902 1 001

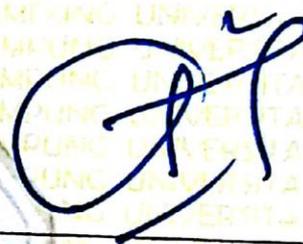
**MENGESAHKAN**

**1. Tim Penguji**

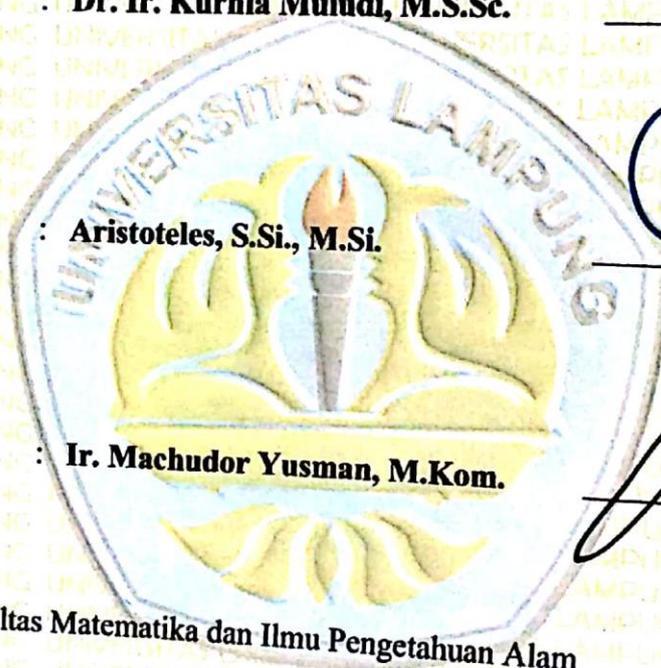
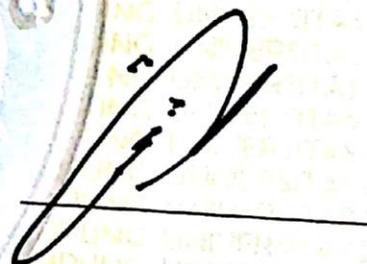
**Ketua : Dr. Ir. Kurnia Muludi, M.S.Sc.**



**Sekretaris : Aristoteles, S.Si., M.Si.**



**Penguji Utama : Ir. Machudor Yusman, M.Kom.**



**2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam**



**Prof. Warsito, S.Si., DEA, Ph.D.**  
NIP. 19710212 199512 1 001

**Tanggal Lulus Ujian Skripsi : 21 Desember 2017**

## PERNYATAAN ORISINALITAS

Saya, yang bertanda tangan di bawah ini.

Nama : Febrasari Almania  
NPM / NIK : 1017032024 / 1871014502930003  
Fakultas : Matematika dan Ilmu Pengetahuan Alam  
Program Studi : S1 Ilmu Komputer  
Alamat : Jalan Cengkeh Utara V/70 LK II Perumnas Way Halim,  
Bandarlampung 35141

Dengan ini menyatakan bahwa skripsi yang berjudul "**Pengembangan Sistem Informasi Manajemen Beasiswa di Fakultas MIPA Universitas Lampung Menggunakan Gabungan Metode AHP dan TOPSIS**" adalah hasil kerja saya sendiri dan bukan plagiat, baik untuk naskah laporan maupun kegiatan *programming* yang turut menjadi bagian darinya. Seluruh ide, materi, atau pendapat yang dikutip dari orang lain telah dicantumkan bersama dengan sumbernya menggunakan gaya penulisan referensi yang sesuai.

Apabila di kemudian hari, pernyataan ini terbukti tak sejalan dengan kenyataannya, saya bersedia menanggung sanksi yang dijatuhkan berdasarkan peraturan yang berlaku, termasuk pencabutan gelar Sarjana Komputer yang saya dapatkan.

Bandarlampung, 7 Desember 2017  
Yang menyatakan,



Febrasari Almania

## RIWAYAT HIDUP



Penulis merupakan anak sulung dari lima bersaudara yang lahir pada tanggal 4 Februari 1993 di Jakarta Selatan, dari pasangan Alman Susanto dan Maimunah. Penulis menghabiskan 5 tahun pertamanya di Bogor; sempat terdaftar sebagai murid kelas 1 di SD Negeri Tugu XI Bogor. Pada tahun 1999, ia dan keluarganya bertransmigrasi ke Bandarlampung.

Pendidikan yang dilalui penulis di Bumi Tapis Berseri, berturut-turut, yakni SD Negeri 1 Perumnas Way Halim (1999-2004), SMP Negeri 29 Bandar Lampung (2004-2007), dan SMA Negeri 2 Bandar Lampung (2007-2010). Melalui jalur SNMPTN (tes tertulis), penulis melanjutkan hidupnya selaku mahasiswa Jurusan Ilmu Komputer angkatan 2010. Di saat yang sama, beliau menerima beasiswa Bidik Misi yang berjalan selama 8 semester.

Praktikum-praktikum yang pernah diasuh penulis antara lain Pemrograman C++ (2012), Sistem Keamanan Komputer (2013), Pemrograman Mikroprosesor (2013), dan Pemrograman Berorientasi Objek (2013). Pada liburan semester 5, penulis melaksanakan Kerja Praktik (KP) di Jurusan Ilmu Komputer selaku *developer* fitur tambahan di Siakad Matahari. Setahun setelahnya, beliau menjalani Kuliah Kerja Nyata (KKN) di Kecamatan Padang Cermin, Kabupaten Pesawaran, berkelompok dengan 8 mahasiswa lainnya yang berasal dari Fakultas Ekonomi, ISIP, MIPA, dan Teknik. Dalam masa  $\pm$  30 hari, penulis bersama tim menyosialisasikan program Pos Pemberdayaan Keluarga (Posdaya), memberi pelatihan produksi *virgin coconut oil*

(VCO) dan pupuk kompos berbahan baku kulit kakao kepada warga, menggelar bimbingan belajar bagi murid-murid Sekolah Dasar, serta turut meramaikan kegiatan-kegiatan dusun di bulan Ramadhan.

Posisi penulis di HIMAKOM (Himpunan Mahasiswa Ilmu Komputer) masa bakti 2012/2013 adalah Anggota Divisi Informasi, Komunikasi, dan Media (Infokomedia). Kontribusi terbesar penulis selama amanah tersebut terjadi saat diselenggarakannya Pekan Raya Jurusan Ilmu Komputer I (2013), dimana ia membuat soal untuk Olimpiade Komputer Tingkat SMA/Sederajat Se-Provinsi Lampung. Di pertengahan tahun 2012, penulis meneruskan tugas kakak tingkatnya di Dekanat FMIPA sebagai *webmaster* situs profil fakultas. Tanggung jawab ini diembannya hingga 2014.

Penulis terlibat dalam beragam proyek, di antaranya:

1. Pengembangan Sistem Informasi Pajak Kendaraan Bermotor (PKB) milik Dinas Pendapatan Provinsi Lampung (2014).
2. Pembangunan jaringan laboratorium komputer Yayasan Pendidikan Surya Dharma Bandar Lampung (2014).
3. Pengembangan situs pemasaran produk keripik Asa-Cipto Roso (2015).
4. Pengembangan TEDS (*Tunas Enterprise Dealer System*) di PT. Tunas Dwipa Matra (Mei-Desember 2016).

Kepada segala makhluk  
yang gatal bertanya,  
**“Kapan lulus?”**



*Kupersembahkan skripsi ini spesial untuk kalian*

***Karena sesungguhnya sesudah kesulitan itu ada kemudahan***

***Sesungguhnya sesudah kesulitan itu ada kemudahan***

(Q.S. Al-Insyirah: 5-6)

***Dia-lah Dzat yang tidak pernah menyalahi janji***

(Q.S. Āli 'Imrōn: 9 | Q.S. Ar-Ra'd: 31 | Q.S. Ar-Rūm: 6 | Q.S. Az-Zumar: 20)

***Dan Dia-lah Sebaik-baik Pembuat Rencana***

(Q.S. Āli 'Imrōn: 54 | Q.S. Al-Anfāl: 30)

## SANWACANA

*Assalaamu'alaikum warahmatullaahi wabarakaatuh.*

Tak puas-puas penulis memuja dan memuji Allah *subhaanahu wa ta'aala*, yang tak pernah absen menganugerahkan barokah, rohmah, maghfiroh juga taufiq-Nya kepada hamba-Nya yang lemah dan aniaya ini, meskipun diri tak pernah meminta ataupun luput mensyukurinya. *Laa ilaaha illaa anta, subhaanaka, innii kuntu minazh zhoolimiin*. Shalawat beserta salam penulis tujukan untuk manusia terbaik, Rasulullah Muhammad *shallallaahu 'alaihi wa sallaam*, yang telah dengan gigih menyebarkan cahaya Islam ke penjuru alam semesta, mengangkat derajat manusia ke tingkat yang amat mulia. Semoga kita semua memperoleh *syafa'at* beliau di *yaumul akhir* kelak, dikumpulkan bersama keluarganya, sahabat-sahabatnya, dan *salafush shalih* yang menjadi teladan umat.

Skripsi yang berjudul “***Pengembangan Sistem Informasi Manajemen Beasiswa di Fakultas MIPA Universitas Lampung Menggunakan Gabungan Metode AHP dan TOPSIS***” ini disusun oleh penulis guna memenuhi syarat lulus sebagai Sarjana Komputer di Universitas Lampung. Memang pengerjaannya cukup lama, namun itu semua terbayar dengan aneka pengetahuan dan keahlian teknis yang berhasil diraih penulis. Selagi berproses, penulis juga belajar bermacam-macam *soft-skill*, seperti bersabar, berkomunikasi, menjaga kejujuran, pantang menyerah hingga terketuk

detik terakhir, melawan tantangan, kuadran prioritas, manajemen waktu, serta maksimalisasi doa dan tawakal.

Sepanjang masa penelitian, penulis menganalisis keadaan, menyusun pemecahan masalah, membangun solusi, lalu mengevaluasi sistem yang sudah dikembangkan. Hasil pengujian membuktikan bahwa Sistem Informasi Manajemen Beasiswa sukses memenuhi tugas-tugas yang diembankan di spesifikasi awal. Tetapi, masih ditemui banyak kekurangan di sana-sini, sehingga besar harapan penulis akan adanya para penerus yang kelak menyempurnakan sistem. Begitu raih manfaat yang dapat dipetik jika versi final dari sistem benar-benar diterapkan di lingkup FMIPA Universitas Lampung.

Berikut ini adalah semua insan yang telah membantu penulis dari awal kegiatan skripsi hingga selesainya laporan. Tak ada yang bisa penulis persembahkan selain ucapan terima kasih yang dalam dan do'a yang tulus. Semoga kalian semua dibalas oleh Allah *subhaanahu wa ta'aala* dengan kebaikan yang berlipat-lipat, ditolong di dunia, dan diselamatkan di akhirat. *Aamiin allaahumma aamiin*.

1. Kedua orang tua (**Alman Susanto** dan **Maimunah**) dan adik-adik (**Dina Nurul Amelia**, **Dini Nurul Amelina**, **Virghayanti Indah Pratiwi**, dan **Muhammad Najie Karamy**), serta segenap keluarga besar, yang tak lelah memantau perkembangan skripsi ini, tak jenuh menasihati, pun tak reda mendo'akan kebaikan untuk penulis. Allah balas kesabaran tiada ujung yang kalian jaga dengan gelar sarjana yang kita idam-idamkan. Penulis memohon maaf atas keegoisan yang selama ini penulis tampilkan, yang sering membuat kalian khawatir.

2. Bapak **Prof. Warsito, S.Si., DEA, Ph.D.** selaku Dekan FMIPA yang sudah mem-bubuhkan tanda tangannya di halaman pengesahan.
3. Bapak **Drs. Suratman, M.Sc.** selaku Wakil Dekan III (Bidang Kemahasiswaan dan Alumni) yang tempo lalu mengizinkan penulis mewawancarai anggota-anggota divisinya, dalam rangka analisis masalah.
4. Bapak **Dr. Ir. Kurnia Muludi, M.S.Sc.** dan Ibu **Astria Hijriani, S.Kom., M.Kom.** selaku Pembimbing I dan II, yang begitu toleran kepada penulis, juga atas perhatiannya dalam memberikan berbagai arahan sekaligus pencerahan.
5. Bapak **Aristoteles, S.Si., M.Si.** selaku Pembimbing II (pengganti). Meskipun masa bimbingan dengan beliau tergolong singkat, namun penulis amat tertolong dengan kelapangan-kelapangan yang beliau berikan.
6. Bapak **Ir. Machudor Yusman, M.Kom.** selaku Pembahas, yang telah menyampaikan sejumlah anjuran supaya sistem yang digodok penulis mendekati kenyataan di lapangan. Revisi penulisan yang berlimpah dari beliau kini menjadi bekal penulis dalam menciptakan karya-karya ilmiah selanjutnya.
7. Bapak **Didik Kurniawan, S.Si., M.T.** selaku Sekretaris Jurusan Ilmu Komputer, yang telah memudahkan segala urusan pelik. Bersemester-semester penulis dididik dalam mata kuliah pemrograman oleh beliau. Meski lebih deras dukanya dibanding sukanya, berkat binaan beliau, penulis mampu berdiri di atas kaki sendiri. Terlampau besar peran beliau bagi karier penulis. Semoga Allah *al-Kaafii* memberkahi hidup beliau dan keluarganya. *Aamiin.*
8. Ibu **Anie Rose Irawati, S.T., M.Cs.** selaku dosen Pembimbing Akademik (PA) penulis. Penulis ingat sekali bagaimana beliau mendorong penulis untuk menjadi lulusan tercepat se-angkatan  $\pm 3,5$  tahun yang lalu; namun apa daya yang terjadi

justru sebaliknya. Hormat penulis selalu untuk beliau. Maafkan pula pribadi penulis yang memang susah diatur.

9. Seluruh orang tua penulis di kampus dari awal NPM disandang hingga akhirnya ditanggalkan:

- Dr. Eng. Admi Syarif
- Drs. Rd. Irwan Adi Pribadi, M.Kom.
- Dwi Sakethi, S.Si., M.Kom.
- Rico Andrian, S.Si., M.Kom.
- Febi Eka Febriansyah, S.T., M.T.
- Tristiyanto, S.Kom., M.I.S., Ph.D.
- Ossy Dwi Endah Wulansari, M.T.
- Bambang Hermanto, S.Kom., M.Cs.
- Favorisen R. Lumbanraja, S.Si., M.Si., Ph.D.
- Prof. Dra. Wamiliana, M.A., Ph.D.
- Prof. Drs. Mustofa Usman, M.A., Ph.D.
- Drs. Rudi Ruswandi, M.Si.
- Dr. Khoirin Nisa, S.Si., M.Si.
- Dian Kurniasari, S.Si., M.Sc.
- Widiarti, S.Si., M.Si.
- Dr. Hardoko Insan Qudus, M.Si.
- Dra. Dwi Asmi, M.Si., Ph.D.
- Prof. Dr. Sutopo Hadi, S.Si., M.Sc.
- Ahmad Faisol, S.Si., M.Sc.
- Fitriani, S.Si., M.Sc.
- Subian Saidi, S.Si., M.Si.
- Drs. Kahfie Nazaruddin, M.Hum.
- Drs. Deddy Supriyadi, M.Pd.
- Drs. Tontowi, M.Si.
- Dr. Marta Dinata, M.Pd.
- Drs. Baharuddin, M.Pd.
- Drs. Ruswanto, M.Ag.
- Drs. Ediman Ginting, M.Si.
- Gurus Ahmad Pauzi, S.Si., M.T.
- Prof. Dr. I Gede Bagus Wiranata, S.H., M.H.
- Agung Wibawa, S.Sos., M.Si.
- Ir. Indriyanto, M.P.

Tiada imbalan yang sanggup penulis sandingkan dengan pengorbanan waktu dan tenaga yang beliau-beliau lakukan semasa mengajari penulis. Pengetahuan yang Bapak/Ibu tinggalkan sukses mentransformasi pola pikir dan karakter penulis; sungguh itu lebih berharga daripada harta pusaka apapun, sebagaimana yang dikatakan oleh *sayyidinaa* ‘Ali ibn Abi Thalib *rodhiyallaahu ‘anhu*,”*Ilmu itu lebih baik daripada harta, ilmu akan menjagamu sedangkan kamu-lah yang akan menjaga harta. Ilmu itu hakim sedangkan harta adalah yang dihakimi. Telah mati para penyimpan harta dan tersisa-lah para pemilik ilmu, walaupun diri-diri mereka telah tiada akan tetapi pribadi-pribadi mereka tetap ada pada*

*hati-hati manusia.”* Beserta niat yang lurus, semoga warisan Bapak/Ibu ini dicatat oleh Allah *al-Bashiir* sebagai aliran pahala yang takkan putus bahkan setelah mata tertutup. *Aamiin. A good teacher can inspire hope, ignite the imagination, and instill a love of learning* (Brad Henry).

10. Bapak **Eko Putranto, B.A.Sc., CISA, CISSP**, Bapak **Muhammad Alaydrus, S.Kom.**, dan Divisi Teknologi Informasi *main dealer* Honda Lampung yang sudah mengedukasi penulis serta menunjukkan batas atas yang perlu penulis dobrak demi karier yang lebih cemerlang. Penulis berikrar untuk memperbaiki etos kerjanya agar kelak pantas berkolaborasi dengan para ahli dan praktisi TI yang bersinar dan hebat-hebat di Nusantara, bahkan dunia.
11. Ibu **Eny Maryani, S.H., M.H.** dan Bapak **Slamet Riyanto**, selaku Kasub dan anggota Bagian Kemahasiswaan dan Alumni FMIPA Universitas Lampung, yang dengan ramah dan bersahabat memberikan informasi yang penulis perlukan untuk melengkapi skripsinya.
12. Ibu **Anita, A.Md.**, Ibu **Ade Nora Maela**, Mas **Nurkholis**, Mas **Ardi Nofalian**, dan para staf Dekanat FMIPA Universitas Lampung atas layanan-layanannya yang membaik dari hari ke hari. Sungguh tinggi optimisme penulis akan tergapainya kualitas manajerial fakultas berikut jurusan-jurusan di dalamnya yang benar-benar merefleksikan standar ISO 9001.
13. **Ade Oktaviani, S.T.**, seorang teman yang sudah penulis anggap seperti saudara sendiri, untuk segala dukungan dan andilnya yang tak tanggung-tanggung dalam segala hal. Semoga menara cahaya yang membuat iri para syuhada menjadi tempat persinggahan kita di dunia selanjutnya. *Aamiin yaa Raafi’*.

14. **Tri Yuni Susanti**, kawan perjuangan yang menghadapi beragam fenomena berlaka-liku, yang bersamanya penulis belajar juga mengajar, bersemangat juga menyemangati, berupaya juga mengupayakan, dan bermimpi juga memantik impian. Penulis pintakan pada Allah *ar-Rosyiid at-Tawwaab ash-Shobuur* agar Dia tak pernah meninggalkanmu walau sedetik.
15. Teman-teman **Ilmu Komputer Universitas Lampung 2010** yang kerap mengingatkan dan menanyai tak henti-henti terkait skripsi ini sehingga penulis terdorong secara moril untuk menyelesaikan apa yang ia mulai. Semoga kita semua dijumpakan kembali oleh Allah *Al-Jaami'* di masa depan dalam keadaan yang lebih baik daripada ketika saat berpisah. *Aamiin yaa Haadii yaa Mughniiy.*
16. **Muhammad Chairuddin, S.Kom.** yang sampai saat ini masih meminjami penulis *hard-disk*-nya tanpa (banyak) pamrih.
17. **Pita Utari Ningtyas, S.Kom., Fajar Sidik, S.Kom., dan Hermansyah** yang rela meramaikan seminar-seminar penulis juga memandu penulis perihal penyelesaian administrasi.
18. **Choiranti Efrina, S.Kom. dan Hartanto Tantriawan, S.Kom., M.Kom.,** sejawat-sejawat terdekat penulis dalam setiap tugas individu maupun kelompok. Semoga Allah *ar-Rohmaan ar-Rohiim* memberkahi rumah tangga kalian serta mengamanah-kan pada kalian sebanyak-banyaknya keturunan.
19. **M. Harry Haryono, S.Kom. dan Rian Pandu, S.Kom.,** sejawat-sejawat terdekat lainnya dalam hal perunduhan sinema-sinema kualitas terbaik. Meskipun apa yang kita seriusi ini hingga sekarang tak lunas disaksikan, semoga menjadi pengingat tentang betapa tak berfaedahnya pertemanan kita.

20. **Harjo Apkuanbo, S.Kom.** yang tak pernah bosan berbagi pengalaman serta pertolongan kepada penulis. Kisah-kisah yang kerap beliau bagikan kala berjumpa selalu memperluas dan memperdalam cakrawala pikiran dan hati penulis. *Jazaakallaahu khoiron katsir*. Biarlah Allah *tabaaroka wa ta'aala* yang tentukan sendiri balasan sepadan untuk beliau.
21. **Rahmat Widodo, S.Kom.** yang merupakan teman diskusi penulis tentang *Open Authorization (OAuth) 2.0*. Melalui skripsinya, ia lahirkan karya yang memang sangat dibutuhkan oleh almamater. Adik-adik sekalian patut meneladani kegigihan beliau.
22. **Amin Nurul Fatah** yang mengintroduksi penulis pada *software Snipping Tool*. Apa yang beliau perkenalkan sungguh bermanfaat bagi penulis saat mendokumentasikan antarmuka sistem; semoga dihitung oleh Allah *al-Wahhaab* sebagai amal jariyah. *Aamiin*.
23. Senior-senior dan junior-junior Ilmu Komputer sekalian yang turut memberi warna-warni dalam kehidupan perkuliahan penulis. Pengalaman ditatar dan menatar, bahu-membahu menyiapkan acara, pontang-panting menyusun soal-soal Olimpiade, mentraktir dan ditraktir, menitip dan dititipi, bercanda dan berkeluh kesah, tertawa dan menertawakan, serta membantu dan dibantu adalah sekelumit saja dari kenangan yang insya Allah akan penulis wariskan pada anak-cucu kelak. Mari kita harumkan nama almamater, Lampung, dan Indonesia dengan berbagai inovasi dan terobosan teknologi. HIMAKOM Semakin Jaya!
24. **Isti Nur Kumalasari, S.T.** dan **Sri Rizky, S.T.** yang *semestinya* di-wisuda setelah kami. Siapa yang sangka kalau waktu tiga tahun yang penulis

*dedikasikan* bersama kalian *cuma* berbalas *printer* dan *software* pengecek plagiat **#Self\_Sarcasm**. Bekerja dan berjodoh-lah segera; siapa yang tahu apakah kalian bisa merasakan keduanya hingga akhir hayat **#Sarcasm**. *Anyway*, tetap kudo'akan agar kalian diampuni dosa-dosanya dan dimaafkan kesalahan-kesalahannya oleh Allah *al-Ghoffaar al-Ghofuur*. *Aamiin*.

25. **Angga Wijaya, S.Si.**, alumni Jurusan Matematika Universitas Lampung 2010 yang bersedia membantu perihal pengujian sistem secara matematis. Semoga engkau ditunjuki oleh Tuhan titik terang saat berada dalam kegelapan, kemudian dituntun menuju kebahagiaan yang hakiki.
26. **Surtini Karlina Sari, S.Si., (M.Si.)**, alumni Jurusan Kimia Universitas Lampung 2010 yang sedang menyempurnakan studi pascasarjananya. Di sela-sela kesibukannya, beliau konsisten meringankan beban penulis dengan semangat 45+86. Semoga engkau segera dipasangkan oleh Allah *al-Waduud* dengan seseorang yang sanggup mendekatkanmu pada surga-Nya. *Aamiin*.
27. Adik-adik se-almamater: **Ima Kurnia, Adriyana Budiarti, Prastiana Tiara Pratiwi, Dias Isyadini, Reni Rukma Winarti, Ani Lailia, Wanda Gustina Utami, Endah Kurnia Setia Dewi, Desi Yanti, Desta Feranita, Dina Munjiati, Lili Adiningsih, Novenda Sartika, Nurul Dawati Adawiyah, Mita Yuliana, Synthia Mega Putri, Nabila Hasri Ainun, dan Silvia Rahayu Angraini**. Semoga Allah *al-'Aliim* mengokohkan sekaligus menggegaskan langkah kalian selama menimba ilmu dan mengasah keterampilan, sehingga tercapai sebuah titel yang tak sekedar titel; yang diridhoi Allah *'Azza wa Jallaa* pun mampu diaplikasikan untuk menjawab kebutuhan-kebutuhan masyarakat. *Aamiin*.

28. Mantan Presiden RI ke-6 **Prof. Dr. H. Susilo Bambang Yudhoyono, M.A.**, jajaran Kementerian Pendidikan dan Kebudayaan masa bakti 2009-2014, dan Pembantu Rektor III **Prof. Dr. Sunarto DM, S.H., M.H.**, beserta timnya atas peluncuran dan pengelolaan beasiswa Bidik Misi yang telah mengakomodasi semua pengeluaran penulis selama 8 semester.
29. Dan pihak-pihak lainnya yang turut berjasa sehingga laporan ini akhirnya rampung, yang sayangnya, tidak dapat disebutkan satu persatu.

Pepatah berkata tiada sesuatu yang sempurna di dunia; begitu pun skripsi ini. Penulis sangat menyadarinya, oleh karena itu, saran dan kritik yang membangun sangat dinantikan sebagai bahan perbaikan untuk tulisan-tulisan yang akan datang. Semoga buah keringat ini mencerahkan siapapun yang sedang dirundung kebuntuan, menjadi jawaban bagi semua orang yang sedang kesulitan, dan menenangkan hati-hati mahasiswa yang sedang galau. *Bertakwalah kepada Allah, Allah mengajarmu, dan Allah Maha Mengetahui segala sesuatu.*

*Assalaamu'alaikum warahmatullaahi wabarakaatuh.*

Bandarlampung, 7 Desember 2017

Febrasari Almania

## DAFTAR ISI

	Halaman
DAFTAR TABEL .....	xiv
DAFTAR GAMBAR.....	xvi
I. PENDAHULUAN .....	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	4
1.3. Batasan Masalah .....	4
1.4. Tujuan.....	5
1.5. Manfaat.....	5
II. TINJAUAN PUSTAKA.....	6
2.1. Sistem Pendukung Keputusan .....	6
2.2. Pengelolaan Beasiswa di FMIPA Universitas Lampung .....	8
2.3. <i>Multiple Criteria Decision Making</i> (MCDM).....	17
2.4. <i>Analytic Hierarchy Process</i> (AHP).....	21
2.4.1. Pengukuran Relatif.....	21
2.4.2. Struktur Hierarki .....	23
2.4.3. Teknis AHP .....	24
2.4.4. Fungsi AHP dalam Sistem Informasi.....	25
2.4.5. Sintesis Prioritas.....	27
2.4.6. Sintesis Prioritas Menggunakan Pendekatan.....	31
2.4.7. Validasi Matriks Perbandingan Berpasangan.....	32
2.5. <i>Technique for Order of Preference by Similarity to Ideal Solutions</i> (TOPSIS).....	33
2.5.1. Teknis TOPSIS .....	34
2.5.2. Peran Perbandingan Matematis dalam Metode TOPSIS.....	36

III. METODE PENELITIAN.....	39
3.1. Tempat dan Waktu Penelitian.....	39
3.2. Perangkat Pendukung Pengembangan Sistem.....	39
3.3. Metode Penelitian .....	40
3.3.1. Observasi dan wawancara .....	40
3.3.2. Studi literatur .....	42
3.3.3. Perancangan sistem .....	42
3.3.4. Pengembangan sistem .....	107
3.3.5. Pengujian sistem .....	110
3.3.6. Penarikan simpulan .....	111
3.3.7. Publikasi penelitian .....	111
IV. HASIL DAN PEMBAHASAN .....	112
4.1. Hasil Pengembangan Sistem.....	112
4.1.1. Laman <i>Login</i> dan Menu <i>Logout</i> .....	115
4.1.2. Laman Periode .....	117
4.1.3. Laman Master Beasiswa.....	123
4.1.4. Laman Pengguna.....	126
4.1.5. Laman Mahasiswa.....	133
4.1.6. Laman Pegawai.....	135
4.1.7. Laman Beasiswa .....	136
4.2. Hasil Pengujian Sistem.....	162
4.3. Pembahasan .....	164
V. SIMPULAN DAN SARAN .....	165
5.1. Simpulan.....	165
5.2. Saran.....	165
DAFTAR PUSTAKA .....	169

## DAFTAR TABEL

Tabel	Halaman
1. Makna Simbol-Simbol Diagram Alir.....	12
2. Ikhtisar Metode-Metode MCDM.....	18
3. Skala Dasar Angka Mutlak.....	22
4. Matriks Perbandingan Berpasangan Kriteria Beasiswa PPA.....	26
5. Pendekatan-Pendekatan AHP.....	31
6. <i>Random Consistency Index</i> .....	33
7. Daftar Skor Kriteria.....	36
8. Data Asli Alternatif.....	37
9. Nilai Alternatif setelah Dikonversi.....	37
10. Komparasi Skor IPK dari 2 Metode Konversi.....	38
11. Daftar Kriteria dalam Sistem Informasi Manajemen Beasiswa FMIPA.....	46
12. Daftar Poin untuk Kriteria <i>Prestasi</i> .....	101
13. Daftar Menu/Submenu Sistem Manajemen Beasiswa.....	117
14. Mekanisme Keamanan Odoo.....	121
15. <i>Access Control</i> Laman Periode.....	123
16. <i>Access Control</i> Laman Master Beasiswa.....	126
17. Kontrol Keamanan Laman Pengguna.....	133

18. Kontrol Keamanan Laman Beasiswa bagian Beasiswa .....	149
19. Kontrol Keamanan Laman Beasiswa bagian Pendaftaran .....	160
20. Perbandingan Bobot-Bobot Kriteria untuk Matriks I .....	163
21. Perbandingan Bobot-Bobot Kriteria untuk Matriks II .....	163
22. Perbandingan Bobot-Bobot Kriteria untuk Matriks III.....	163
23. Jumlah Kasus Uji per Blok Sistem .....	205
24. Rencana Pengujian Sistem Manajemen Beasiswa.....	206
25. Hasil Pengujian Sistem Manajemen Beasiswa.....	263

## DAFTAR GAMBAR

Gambar	Halaman
1. Komponen-Komponen DSS.....	7
2. Diagram Alir Pengelolaan Beasiswa di FMIPA Universitas Lampung.....	13
3. Contoh Struktur Hierarki.....	24
4. Struktur Hierarki Beasiswa PPA. ....	26
5. Diagram Alir Penelitian. ....	40
6. Diagram Alir Pengelolaan Beasiswa Setelah Implementasi Sistem.....	43
7. Ragam <i>Use-Case</i> di Blok <b>Umum</b> . ....	47
8. <i>Activity Diagram</i> Fungsi <i>Login</i> . ....	48
9. <i>Sequence Diagram</i> Fungsi <i>Login</i> .....	48
10. <i>Activity Diagram</i> Fungsi <i>Logout</i> . ....	49
11. <i>Sequence Diagram</i> Fungsi <i>Logout</i> .....	49
12. <i>Activity Diagram</i> Fungsi Pembuatan Akun.....	49
13. <i>Sequence Diagram</i> Fungsi Pembuatan Akun. ....	50
14. <i>Activity Diagram</i> Fungsi Pembacaan Data Akun. ....	51
15. <i>Sequence Diagram</i> Fungsi Pembacaan Data Akun. ....	51
16. <i>Activity Diagram</i> Fungsi Pengubahan Data Akun.....	52
17. <i>Sequence Diagram</i> Fungsi Pengubahan Data Akun.....	53

18. <i>Activity Diagram</i> Fungsi Penghapusan Akun.....	54
19. <i>Sequence Diagram</i> Fungsi Penghapusan Akun.....	55
20. <i>Activity Diagram</i> Fungsi Penonaktifan Akun.....	56
21. <i>Sequence Diagram</i> Fungsi Penonaktifan Akun.....	56
22. Ragam <i>Use-Case</i> di Blok <b>Periode</b> .....	57
23. <i>Activity Diagram</i> Fungsi Pembuatan Data Periode. ....	57
24. <i>Sequence Diagram</i> Fungsi Pembuatan Data Periode.....	58
25. <i>Activity Diagram</i> Fungsi Pembacaan Data Periode.....	59
26. <i>Sequence Diagram</i> Fungsi Pembacaan Data Periode. ....	59
27. <i>Activity Diagram</i> Fungsi Penghapusan Periode. ....	60
28. <i>Sequence Diagram</i> Fungsi Penghapusan Periode.....	60
29. Ragam <i>Use-Case</i> di Blok <b>Master Beasiswa</b> .....	61
30. <i>Activity Diagram</i> Fungsi Pembuatan Master Beasiswa. ....	62
31. <i>Sequence Diagram</i> Fungsi Pembuatan Master Beasiswa. ....	62
32. <i>Activity Diagram</i> Fungsi Pembacaan Data Master Beasiswa. ....	63
33. <i>Sequence Diagram</i> Fungsi Pembacaan Data Master Beasiswa.....	63
34. <i>Activity Diagram</i> Fungsi Pengubahan Data Master Beasiswa. ....	64
35. <i>Sequence Diagram</i> Fungsi Pengubahan Data Master Beasiswa. ....	64
36. <i>Activity Diagram</i> Fungsi Penghapusan Master Beasiswa.....	65
37. <i>Sequence Diagram</i> Fungsi Penghapusan Master Beasiswa. ....	66
38. <i>Use-Case</i> <b>Melihat Data Mahasiswa</b> .....	67

39. <i>Activity Diagram</i> Fungsi Pembacaan Data Mahasiswa. ....	67
40. <i>Sequence Diagram</i> Fungsi Pembacaan Data Mahasiswa.....	68
41. <i>Use-Case</i> <b>Melihat Data Pegawai</b> . ....	68
42. <i>Activity Diagram</i> Fungsi Pembacaan Data Pegawai.....	69
43. <i>Sequence Diagram</i> Fungsi Pembacaan Data Pegawai.....	69
44. Ragam <i>Use-Case</i> di Blok <b>Validator</b> . ....	70
45. <i>Activity Diagram</i> Fungsi Pembuatan Data Validator.....	71
46. <i>Sequence Diagram</i> Fungsi Pembuatan Data Validator.....	71
47. <i>Activity Diagram</i> Fungsi Pembacaan Data Validator. ....	72
48. <i>Sequence Diagram</i> Fungsi Pembacaan Data Validator. ....	72
49. <i>Activity Diagram</i> Fungsi Pengubahan Data Validator.....	72
50. <i>Activity Diagram</i> Fungsi Penghapusan Validator.....	73
51. <i>Sequence Diagram</i> Fungsi Penghapusan Validator.....	73
52. Ragam <i>Use-Case</i> di Blok <b>Kriteria</b> . ....	74
53. <i>Activity Diagram</i> Fungsi Pembacaan Data Kriteria.....	74
54. <i>Sequence Diagram</i> Fungsi Pembacaan Data Kriteria. ....	75
55. <i>Activity Diagram</i> Fungsi Pengubahan Data Kriteria. ....	76
56. <i>Sequence Diagram</i> Fungsi Pengubahan Data Kriteria.....	77
57. <i>Activity Diagram</i> Fungsi Perhitungan Bobot Kriteria. ....	78
58. <i>Sequence Diagram</i> Fungsi Perhitungan Bobot Kriteria.....	79
59. Ragam <i>Use-Case</i> di Blok <b>Beasiswa</b> . ....	80

60. <i>Activity Diagram</i> Fungsi Pembuatan Beasiswa.....	82
61. <i>Sequence Diagram</i> Fungsi Pembuatan Beasiswa. ....	83
62. <i>Activity Diagram</i> Fungsi Pembacaan Data Beasiswa. ....	84
63. <i>Sequence Diagram</i> Fungsi Pembacaan Data Beasiswa. ....	84
64. <i>Activity Diagram</i> Fungsi Pengubahan Data Beasiswa.....	85
65. <i>Sequence Diagram</i> Fungsi Pengubahan Data Beasiswa. ....	86
66. <i>Activity Diagram</i> Fungsi Penghapusan Beasiswa.....	87
67. <i>Sequence Diagram</i> Fungsi Penghapusan Beasiswa.....	88
68. <i>Activity Diagram</i> Fungsi Publikasi Beasiswa.....	89
69. <i>Sequence Diagram</i> Fungsi Publikasi Beasiswa.....	89
70. Ragam <i>Use-Case</i> di Blok <b>Pendaftaran</b> . ....	90
71. <i>Activity Diagram</i> Fungsi Pendaftaran Beasiswa.....	92
72. <i>Sequence Diagram</i> Fungsi Pendaftaran Beasiswa.....	93
73. <i>Activity Diagram</i> Fungsi Pembacaan Data Pendaftar.....	94
74. <i>Sequence Diagram</i> Fungsi Pembacaan Data Pendaftar. ....	94
75. <i>Activity Diagram</i> Fungsi Pengubahan Data Pendaftar.....	95
76. <i>Sequence Diagram</i> Fungsi Pengubahan Data Pendaftar.....	96
77. <i>Activity Diagram</i> Fungsi Pembatalan Pendaftaran. ....	97
78. <i>Sequence Diagram</i> Fungsi Pembatalan Pendaftaran. ....	97
79. <i>Activity Diagram</i> Fungsi Validasi Data Pendaftar.....	98
80. <i>Sequence Diagram</i> Fungsi Validasi Data Pendaftar.....	99

81. <i>Activity Diagram</i> Fungsi Penolakan Data Pendaftar. ....	99
82. <i>Sequence Diagram</i> Fungsi Penolakan Data Pendaftar.....	100
83. Ragam <i>Use-Case</i> di Blok <b>Seleksi</b> .....	101
84. <i>Activity Diagram</i> Fungsi Seleksi Pendaftar.....	102
85. <i>Sequence Diagram</i> Fungsi Seleksi Pendaftar.....	103
86. <i>Activity Diagram</i> Fungsi Pembacaan Hasil Seleksi.....	103
87. <i>Sequence Diagram</i> Fungsi Pembacaan Hasil Seleksi.....	104
88. <i>Entity Relationship Diagram</i> (ERD) Sistem Manajemen Beasiswa.....	105
89. Metode Pengembangan Iteratif.....	108
90. Strategi Implementasi Iteratif.....	109
91. Pola Desain <i>Model-View-Controller</i> .....	113
92. Sampel <i>Model</i> Odoo.....	114
93. Hasil Realisasi <i>Model</i> di Basis Data.....	114
94. Sampel <i>View</i> Odoo.....	115
95. Tampilan Laman <i>Login</i> dan Menu <i>Logout</i> .....	116
96. <i>Login</i> Gagal.....	116
97. Tampilan Laman Periode (Tipe <i>List</i> ).....	118
98. Tampilan Laman Periode (Tipe <i>Form</i> ).....	118
99. Pesan Peringatan untuk Masukan Tahun yang Salah.....	119
100. Pesan Peringatan untuk Duplikasi Periode.....	119
101. Periode-Periode yang Inaktif.....	120

102. Penghapusan Periode Gagal.....	120
103. Kode-Kode Keamanan Level <i>Access Control</i> .....	122
104. Tampilan Laman Master Beasiswa (Tipe <i>List</i> ).....	123
105. Tampilan Laman Master Beasiswa (Tipe <i>Form</i> ).....	124
106. Pesan Peringatan untuk Duplikasi Master Beasiswa.....	124
107. Pesan Peringatan untuk Master Beasiswa tak Bernama.....	125
108. Struktur Tabel <i>res_users</i> .....	126
109. Tampilan Laman Pengguna (Tipe <i>List</i> ).....	127
110. Tampilan Laman Pengguna (Tipe <i>Form</i> ).....	127
111. Tampilan Seksi <b>Preferensi</b> .....	128
112. Pesan Peringatan untuk Duplikasi <i>Login</i> .....	128
113. Mengganti Status Pengguna Lewat Tombol <b>Aktif</b> .....	129
114. Pengguna-Pengguna yang Inaktif.....	129
115. Tampilan Laman Preferensi.....	130
116. Dialog Perubahan Kata Sandi.....	130
117. Perubahan Kata Sandi oleh Administrator: Pemilihan ID Pengguna.....	132
118. Perubahan Kata Sandi oleh Administrator.....	132
119. Tampilan Laman Mahasiswa (Tipe <i>List</i> ).....	134
120. Tampilan Laman Mahasiswa Khusus Administrator (Tipe <i>Form</i> ).....	134
121. Tampilan Laman Mahasiswa (Tipe <i>Form</i> ).....	134
122. Tampilan Laman Pegawai (Tipe <i>List</i> ).....	135

123. Tampilan Laman Pegawai (Tipe <i>Form</i> ).....	136
124. Tampilan Laman Pegawai Khusus Administrator (Tipe <i>Form</i> ).....	136
125. Tampilan Laman Beasiswa (Tipe <i>List</i> ).....	137
126. Tampilan Laman Beasiswa Khusus Mahasiswa (Tipe <i>List</i> ). ....	137
127. Tampilan Laman Beasiswa (Tipe <i>Form</i> ). ....	138
128. Tampilan Laman Beasiswa Khusus Mahasiswa (Tipe <i>Form</i> ). ....	138
129. Pesan Peringatan untuk Duplikasi Beasiswa.....	139
130. Pembuatan Master Beasiswa Melalui Laman Beasiswa.....	139
131. Pesan Peringatan untuk Penjadwalan yang Tak Logis. ....	140
132. Pesan Peringatan untuk Isian Tanggal yang Tak Benar.....	140
133. Alokasi Penerima Beasiswa per Jurusan.....	141
134. Modifikasi Data Beasiswa yang Berstatus <b>Menunggu Bobot</b> .....	142
135. <i>Field-Field</i> Kekuatan Relatif Antar Kriteria.....	143
136. <i>Field-Field</i> Bobot Kriteria. ....	143
137. Kalkulasi Bobot-Bobot Kriteria Gagal. ....	144
138. Panduan Pengisian <i>Field-Field</i> Kekuatan Relatif Antar Kriteria.....	144
139. Pesan Peringatan untuk Syarat Publikasi Beasiswa Tak Lengkap. ....	145
140. Penambahan Validator Beasiswa.....	146
141. Penambahan Validator Beasiswa Sukses.....	146
142. Penghapusan Validator Beasiswa.....	147
143. Laman Beasiswa yang Berstatus <b>Dipublikasikan</b> Khusus Administrator dan Tim Seleksi (Tipe <i>Form</i> ).....	147

144.	Tampilan Laman Pendaftaran (Tipe <i>List</i> ).	150
145.	Tampilan Laman Pendaftaran Khusus Mahasiswa (Tipe <i>List</i> ).	150
146.	Tampilan Laman Pendaftaran (Tipe <i>Form</i> ).	150
147.	Tampilan Laman Pendaftaran Khusus Mahasiswa (Tipe <i>Form</i> ).	151
148.	Memilih Beasiswa di Formulir Pendaftaran Beasiswa <i>Online</i> .	152
149.	Poin-Poin Lingkup Prestasi.	153
150.	Poin-Poin Titel Prestasi.	153
151.	Registrasi Gagal karena Sedang Melamar/Menerima Beasiswa Lain.	154
152.	Pembaruan Data Registrasi Gagal.	155
153.	<i>Field Alasan Penolakan</i> .	156
154.	Borang Registrasi yang Berstatus <b>Valid</b> .	156
155.	Borang Registrasi yang Berstatus <b>Ditolak</b> .	157
156.	Pesan Peringatan untuk Validasi oleh Pegawai yang Tak Berwenang.	157
157.	Borang Registrasi yang Berstatus <b>Dibatalkan</b> .	158
158.	Pesan Peringatan untuk Pembatalan oleh Bukan Pemilik Berkas.	158
159.	Borang Registrasi yang Berstatus <b>Terpilih</b> .	159
160.	Borang Registrasi yang Berstatus <b>Tereliminasi</b> .	159
161.	Halaman <b>Pendaftar</b> dari Beasiswa PPA 2018/2019.	161

## I. PENDAHULUAN

### 1.1. Latar Belakang

Beasiswa merupakan bantuan pendidikan yang diberikan kepada para pendidik maupun terdidik yang memenuhi kualifikasi tertentu. Wujudnya bisa berupa uang tunai, potongan biaya, ataupun barang-barang yang dapat menunjang kegiatan belajar. Tingginya urgensi pendidikan formal serta keterbatasan keadaan, terutama dalam hal dana untuk menempuhnya, menjadikan beasiswa sebuah komponen penting yang harus hadir dalam setiap institusi pendidikan dari segala jenjang. Para pihak yang menyelenggarakan beasiswa antara lain lembaga pendidik, pemerintah, komunitas, perusahaan negeri, juga swasta. Berbagai motivasi yang melatarbelakanginya, di antaranya: implementasi tugas negara atau pengabdian pada masyarakat, memberi hadiah (*reward*) sehingga para pelajar tetap atau bahkan terpacu semangat belajarnya, atau dalam rangka merekrut karyawan ahli.

Kualifikasi yang disodorkan oleh para penyedia beasiswa umumnya terkait dengan kemampuan finansial, prestasi akademik, serta prestasi non-akademik. Penghasilan keluarga per bulan, IPK terakhir, dan karier berorganisasi adalah contohnya. Bersama dengan standar minimalnya masing-masing, poin-poin kualifikasi tadi lazim disebut kriteria. " $IPK \geq 3,00$ " merupakan salah satu contoh kriteria.

Proses penentuan penerima beasiswa dapat diringkas menjadi beberapa tahapan: 1) pendaftaran, 2) penyeleksian calon penerima berdasarkan kriteria, 3) pemberian peringkat pada calon penerima, 4) pengumuman, dan 5) pelaporan, baik kepada penyedia beasiswa maupun pejabat berwenang lainnya. Pada tahap pertama, para pelamar beasiswa mengumpulkan data pribadi serta bukti kualifikasi diri yang relevan pada pihak penyeleksi. Selanjutnya, tim seleksi mengeliminasi berkas pelamar yang tidak lengkap ataupun tidak memenuhi kriteria. Peserta yang berhasil lolos dari tahap 2 kemudian diurutkan menurun; ia yang nilai akhirnya paling besar berada di posisi paling atas, dan sebaliknya. Nilai akhir ini diperoleh dari akumulasi bobot kriteria dikali nilai konversi per kriteria. Akhirnya, para pelamar dengan *ranking* tinggi dipilih, sebanyak kuota yang tersedia, untuk menjadi penerima beasiswa.

Langkah-langkah di atas diadaptasi pula oleh tim penyeleksi di FMIPA Universitas Lampung, namun pelaksanaannya tidak efisien. Setelah berkas para pendaftar yang masih berbentuk *hard copy* terkumpul, anggota tim harus menyalin isi berkas-berkas tersebut ke dalam perangkat lunak *spreadsheet*. Jadi, untuk memperoleh satu paket data, dua kelompok melakukan *input* secara terpisah. Mekanisme seperti ini tentunya boros sumber daya. Seiring dengan semakin banyaknya persediaan beasiswa serta semakin ramai mahasiswa yang terlibat di dalamnya, tenaga dan waktu tim seleksi pun semakin terkuras. Jika apa yang tertulis dalam formulir-formulir registrasi langsung terkonversi ke bentuk digital, produktivitas tim seleksi pasti meningkat.

Agar tahap pendaftaran tak lagi menyusahkan, diperlukan sebuah sistem informasi yang mampu diakses oleh semua mahasiswa, dimana pun dan kapan pun. Dengan begitu, tiap individu yang berkepentingan dengan beasiswa bisa mengisi sendiri data pribadi serta kualifikasi dirinya. Pihak fakultas tinggal menunggu lalu memvalidasi, apakah data yang dimasukkan sudah lengkap dan benar. Peserta yang datanya dinyatakan valid barulah diikutkan di tahap selanjutnya, yakni seleksi.

Hadirnya metode-metode pengambilan keputusan memungkinkan kasus-kasus berkriteria majemuk, termasuk penyeleksian penerima beasiswa, diselesaikan oleh komputer. Urgensinya kian meninggi bila mengingat segala *human error* yang bisa terjadi, yang berpotensi menurunkan atau merusak objektivitas hasil seleksi. Pastinya, tak ada pihak yang menginginkan hal seperti ini. Oleh karena itu, otomatisasi seleksi penerima beasiswa oleh sumber daya komputasi adalah ide yang patut direalisasikan.

*Analytic Hierarchy Process (AHP)* dan *Technique for Others Reference by Similarity to Ideal Solution (TOPSIS)* dianggap sebagai metode penanganan yang paling sesuai. Dengan AHP, dari tingkat kepentingan suatu kriteria terhadap kriteria lainnya, dapat ditentukan bobot tiap kriteria. Setelahnya, bobot-bobot ini dimanfaatkan dalam metode TOPSIS untuk menghitung perolehan nilai masing-masing pelamar. Tentu saja ada metode lain yang juga bisa diterapkan. Pada beberapa jurnal, metode AHP maupun TOPSIS bahkan dipakai sendiri-sendiri guna menuntaskan masalah seleksi penerima beasiswa ini. Namun, gabungan keduanya lebih dipilih agar algoritma proses lebih sederhana, tanpa mengurangi akurasi

keputusan akhir. Kondisi yang demikian mempermudah *developer* saat membangun sistem dengan bahasa pemrograman.

### **1.2. Rumusan Masalah**

Permasalahan yang ditemukan pada penelitian ini adalah: bagaimana mengimplementasikan sistem informasi berbasis komputer yang mampu menyeleksi penerima beasiswa serta mengelola data yang berhubungan dengan proses tersebut?

### **1.3. Batasan Masalah**

Batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Fase validasi data pelamar diserahkan sepenuhnya kepada tim seleksi. Sistem tinggal menerima masukan status “**valid**” atau “**tidak valid**”.
2. Olah data yang dilakukan sistem terbatas untuk menentukan peserta-peserta seleksi yang berhak mendapatkan beasiswa. Jadi, selain identitas pelamar terpilih serta nilai akhirnya, tidak ada lagi informasi yang dihasilkan.
3. Sistem tidak bertanggung jawab atas segala peristiwa yang terjadi setelah proses seleksi, yang untuk menanggulangnya, dibutuhkan seleksi ulang, misalnya pendaftar terpilih tak bersedia menerima beasiswa (mengundurkan diri) atau dikeluarkan oleh universitas.
4. Kriteria-kriteria yang menjadi standar kompetensi tiap beasiswa dicukupkan satu level saja.

#### 1.4. Tujuan

Tujuan dari penelitian ini adalah:

1. Mempermudah tugas perekaman data pribadi penerima beasiswa ke media penyimpanan fakultas.
2. Meningkatkan efisiensi sekaligus efektivitas penyeleksian penerima beasiswa-beasiswa yang berlaku di Fakultas MIPA.

#### 1.5. Manfaat

Manfaat penelitian, yang tak lain merupakan manfaat sistem informasi yang dibuat, dapat dilihat dari sejumlah perspektif:

1. Bagi penyelenggara beasiswa

Pencatatan data serta seleksi para calon penerima beasiswa terlaksana lebih mudah, lebih cepat, dan lebih akurat.

2. Bagi pelamar beasiswa

Keraguan dan keengganan mahasiswa FMIPA Universitas Lampung untuk mengejar beasiswa yang dibutuhkan (akibat faktor subjektivitas pemilihan) setidaknya menurun, karena adanya jaminan bahwa sistem memberikan beasiswa pada orang-orang yang pantas saja.

3. Bagi pengembang sistem

Penelitian ini menjadi kesempatan bagi *developer* untuk mengamalkan ilmu yang diperoleh selama kuliah, ajang mengasah keterampilan serta menumpuk pengalaman, juga menjadi jalan menuju gelar sarjana.

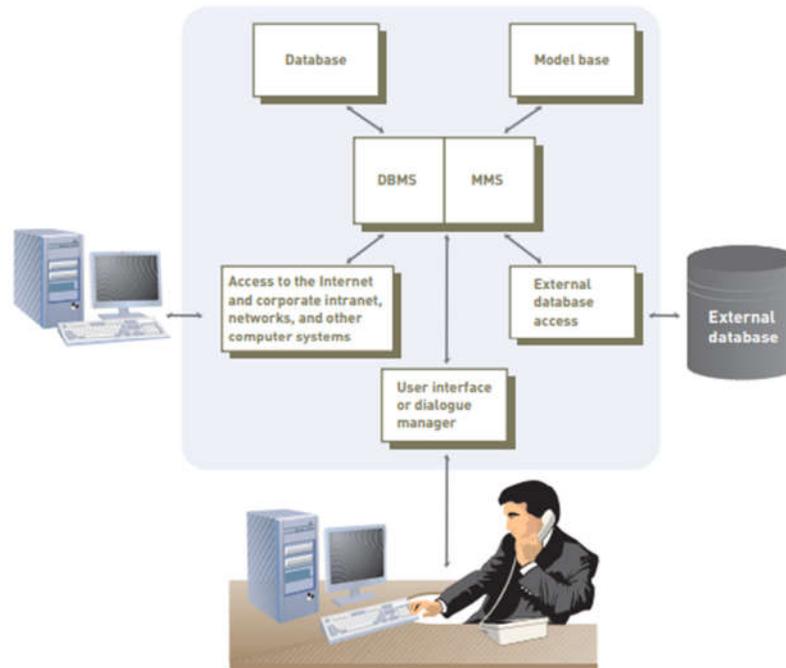
## II. TINJAUAN PUSTAKA

### 2.1. Sistem Pendukung Keputusan

Sistem-sistem komputasi yang umumnya dioperasikan oleh organisasi bisnis atau institusi nirlaba bisa diklasifikasikan dalam tiga golongan besar, yaitu sistem pemrosesan transaksi (*transaction processing system* – TPS), sistem manajemen informasi (*management information system* – MIS), dan sistem pendukung keputusan (*decision support system* – DSS). Deskripsi kerja TPS, MIS, dan DSS, berturut-turut, adalah merekam transaksi yang terjadi, mengelaborasi fakta yang profitabel dari data yang telah dikumpulkan TPS, dan membantu pengguna meramu kebijakan yang efektif. Meski sama-sama berujung pada pembentukan informasi, fokus MIS ialah menolong organisasi untuk “*do things right*”, sementara DSS menyokong eksekutif dalam “*do the right thing*” (Stair & Reynolds, 2012).

DSS biasanya diaplikasikan di ranah manajemen dan perencanaan bisnis, kesehatan, militer, juga area-area lainnya yang kerap timbul situasi-situasi kompleks di sana. Wright dan Sittig (2008) misalnya, dalam rangka menggalakkan pemanfaatan DSS klinis oleh publik, mengembangkan sebuah *framework* untuk memilih arsitektur yang paling sanggup menjawab tuntutan medis di lingkungan tertentu. ForestDSS CoP membangun berbagai DSS guna menyelenggarakan pelestarian hutan yang berkesinambungan. Zhang dan Babovic (2011) mengimplementasikan DSS anti terorisme di pelabuhan-pelabuhan di Selat Malaka.

Suatu DSS terdiri dari beberapa komponen dasar, di antaranya basis data, *model-base*, dan antarmuka (Druzdzet & Flynn, 2002; Stair & Reynolds, 2012). Hubungan ketiganya ditunjukkan oleh Gambar 1.



Gambar 1. Komponen-Komponen DSS (Stair & Reynolds, 2012).

Basis data menyimpan sejumlah besar data yang relevan dengan masalah, sedangkan *model-base* mentransformasi data tersebut menjadi informasi yang berfaedah bagi para pengambil keputusan. Analisis kuantitatif yang dilaksanakan oleh *model-base* juga bisa mencakup data eksternal, apabila diperlukan. Antarmuka sendiri merupakan representasi grafis dari sistem. Elemen-elemen komunikasi turut dianggap oleh Serifi, Randic, dan Dasic (2007) sebagai inti DSS karena *database* dan *model-base* membutuhkannya untuk mengakses dan mengirimkan data dari/ke *user* maupun sistem lain.

## **2.2. Pengelolaan Beasiswa di FMIPA Universitas Lampung**

Pengelola tingkat fakultas, yakni Wakil Dekan III (bidang Kemahasiswaan dan Alumni) bersama Subbag Kemahasiswaan & Alumni FMIPA, menindaklanjuti surat resmi dari Universitas kepada FMIPA tentang pembukaan suatu beasiswa dengan melangsungkan salah satu dari dua hal berikut, tergantung mandat yang tertulis di dalamnya:

1. Meneruskan informasi beasiswa kepada pengelola tingkat jurusan yang terdiri dari para dosen di jurusan.
2. Aksi pada poin 1 sekaligus memberi instruksi kepada pengelola tingkat jurusan untuk melakukan seleksi administratif—mengeliminasi mahasiswa/i yang tidak melengkapi persyaratan atau yang kapabilitasnya tak memenuhi standar.

Kewajiban masing-masing jurusan terhadap dua arahan tadi ialah:

1. Menyebarkan kabar perihal beasiswa kepada mahasiswa.
2. Tindakan pada poin 1 plus menyeleksi para pendaftar.

Salah satu beasiswa yang padanya diterapkan pola pertama: Beasiswa Karya Salemba Empat (KSE) dan Beasiswa DataPrint. Dalam kasus ini, tanggung jawab Perguruan Tinggi terbatas pada diseminasi informasi beasiswa. Pengurusan beasiswa, mulai dari fase pendaftaran hingga pengumuman, diserahkan sepenuhnya kepada penyedia beasiswa.

Paradigma kedua dipraktikkan pada Beasiswa Bank Indonesia (BI), Beasiswa Yayasan Pelayanan Kasih A&A Rachmat (YPKAAR), Beasiswa Peningkatan

Prestasi Akademik (PPA), dan Beasiswa Bantuan Belajar Mahasiswa (BBM) — fungsi Beasiswa BBM kini diambil alih oleh Bantuan Biaya Pendidikan Bidikmisi (Kemristekdikti, 2018). Di sini, pengelola tingkat fakultas berperan sebagai pengantar berkas-berkas pelamar ke universitas, sedangkan pengelola tingkat jurusan menghimpun berkas-berkas yang dimaksud sekaligus menyelenggarakan seleksi berskala jurusan. Penyuplai dua beasiswa pertama mencukupkan wewenang pihak Jurusan pada seleksi administratif, sementara dua sisanya menuntut pihak yang sama, melalui komando dari Wakil Dekan III, agar merangkap sebagai tim seleksi kompetitif — memberi peringkat kepada mahasiswa/i berdasarkan kualitas per individu terhadap kriteria-kriteria beasiswa. Hasilnya kemudian diajukan ke universitas untuk diadu kembali dalam seleksi akhir yang melibatkan seantero Unila. (Maryani & Riyanto, 2017)

Parameter-parameter seleksi kompetitif tingkat jurusan pada Beasiswa BBM dan PPA digariskan oleh Fakultas MIPA dan didistribusikan ke semua jurusan dalam bentuk lampiran Surat Keputusan (SK). Sampelnya diberikan di halaman 182 dan 183. Tampak bahwa kriteria-kriteria yang menjadi indikator penilaian para pelamar ialah *IPK*, *Jumlah SKS*, *Prestasi Kejuaraan*, *Jenjang Peran dalam Lembaga Kemahasiswaan (LK)*, *Penghasilan* serta *Jumlah Tanggungan Orang tua/Wali*. Bobot masing-masing kriteria merefleksikan gugus mahasiswa yang disasar. Beasiswa BBM misalnya; setengah bagian dari nilai pelamarnya digantungkan pada komponen *Penghasilan* dan *Jumlah Tanggungan Orang tua/Wali* karena visi beasiswa ini memang menyantuni para pelajar yang lemah secara finansial. Target Beasiswa PPA pun begitu, tetapi lebih spesifik lagi, yakni harus berprestasi. Oleh

karena itu, akumulasi porsi kriteria *IPK*, *Prestasi Kejuaraan*, dan *Jenjang Peran dalam Lembaga Kemahasiswaan (LK)* mencapai 90%.

Kriteria-kriteria yang dibebankan oleh suatu beasiswa umumnya berbeda dengan beasiswa lain; hal tersebut tercermin dari berkas-berkas yang diminta. Kartu Hasil Studi (KHS)/Transkrip dibutuhkan demi menegaskan nominal IP/IPK seseorang. Slip gaji plus Kartu Keluarga (KK), Surat Keterangan Tidak Mampu (SKTM) dari Kelurahan/Desa, atau Kartu Indonesia Pintar (KIP) (Bidikmisi, 2018) merupakan bukti ketidakberdayaan ekonomi suatu keluarga; sejumlah beasiswa bahkan lebih ketat dengan mewajibkan hadirnya dokumen-dokumen lain, seperti foto kondisi rumah, foto keluarga dengan formasi penuh, struk pembayaran rekening listrik, daftar pengeluaran keluarga per bulan, nomor keanggotaan BPJS Kesehatan, dan slip pembayaran Pajak Bumi dan Bangunan (PBB) (PPMB S1 Universitas Lampung, 2018). Prestasi-prestasi serta riwayat berorganisasi dijustifikasi dengan sertifikat juara dan surat keputusan (SK) tentang kepengurusan lembaga. Apabila tercantum batasan usia dalam kualifikasi beasiswa, pengecekannya bisa dilakukan via Kartu Tanda Penduduk (KTP). Demi mengonfirmasi bahwa pendaftar adalah mahasiswa aktif, diperlukan Kartu Tanda Mahasiswa (KTM), slip SPP terakhir, Kartu Rencana Studi (KRS), atau Surat Keterangan Mahasiswa Aktif. Universitas Lampung menerapkan kebijakan “Satu Mahasiswa Satu Beasiswa” sehingga tiap pelamar, dari beasiswa apapun tanpa terkecuali, harus mengantongi surat keterangan tidak sedang melamar/menerima beasiswa lain yang ditandatangani oleh Wakil Dekan III (Maryani & Riyanto, 2017).

Surat keterangan tidak sedang melamar/menerima beasiswa lain akan dikeluarkan jika NPM mahasiswa yang menghendaknya tidak tercatat di dalam SK-SK yang masih berlaku tentang para penerima beasiswa dan tidak dimuat pula di dalam surat pengantar dari berkas-berkas pelamar yang tengah diseleksi oleh pengelola tingkat universitas. Pihak Fakultas memanfaatkan fitur *Find* pada aplikasi *spreadsheet* untuk menyelesaikan tugas ini (Maryani & Riyanto, 2017). Konsekuensinya, mereka mesti menyalin semua nama yang tertera dalam bundel SK dan surat pengantar tadi ke dalam *spreadsheet*.

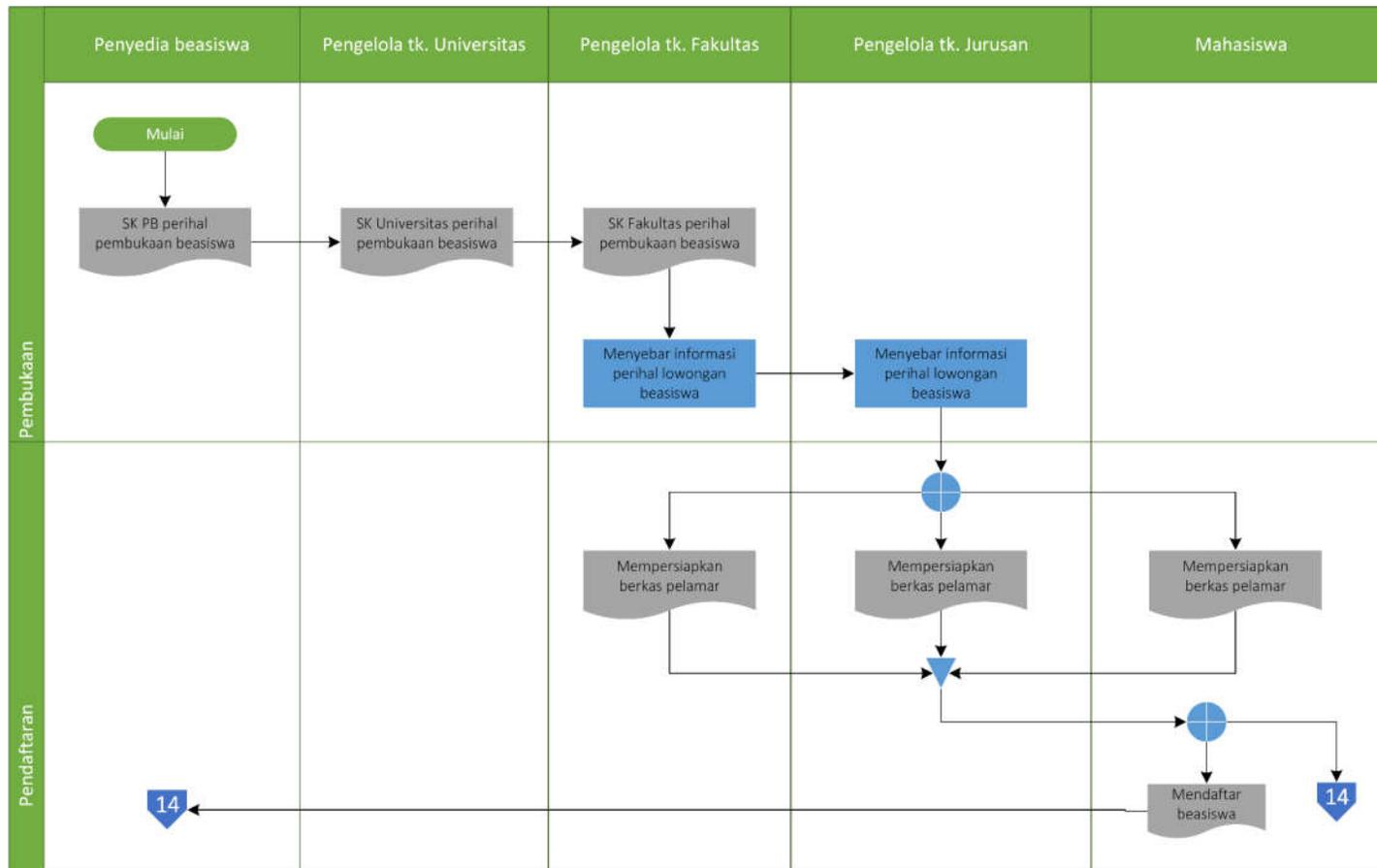
Selepas pendaftaran ditutup, pihak Jurusan menyelenggarakan seleksi administratif, yang kemudian disusul dengan seleksi kompetitif jika diperintahkan. Perhitungan nilai akhir pelamar pun dituntaskan dengan bantuan *spreadsheet*. Itu artinya beban pengelola tingkat jurusan kian menumpuk karena perlu melaksanakan digitalisasi data pelamar, yang mana rekat pada berkas *hard copy*. Demi menghindari pekerjaan tersebut, Sekretaris Jurusan Ilmu Komputer bersama dua mahasiswi bimbingannya mengembangkan Sistem Informasi Seleksi Penerimaan Beasiswa PPA dan BBM (Sakethi, 2015; Sari, 2015; Yunita, 2015). Seiring waktu, logika kedua sistem tak relevan lagi, sehingga kini proses registrasi daring ditangani oleh *Google Forms* (Kurniawan, 2018b). Jadi, selain mengisi formulir dengan pulpen, para pelamar dari Jurusan Ilmu Komputer juga memasukkan detail pendaftarannya pada borang *online*, jika tak ingin digugurkan (Kurniawan, 2018c). Sayangnya, inovasi semacam ini tidak terjadi di jurusan Biologi, Fisika, Kimia, dan Matematika. Para pengelola di sana masih berkuat dengan mode manual.

Paparan barusan dapat dipresentasikan dalam bentuk diagram alir (*flowchart*). Tiap komponen *flowchart* memiliki makna, yang mana dijabarkan oleh Tabel 1.

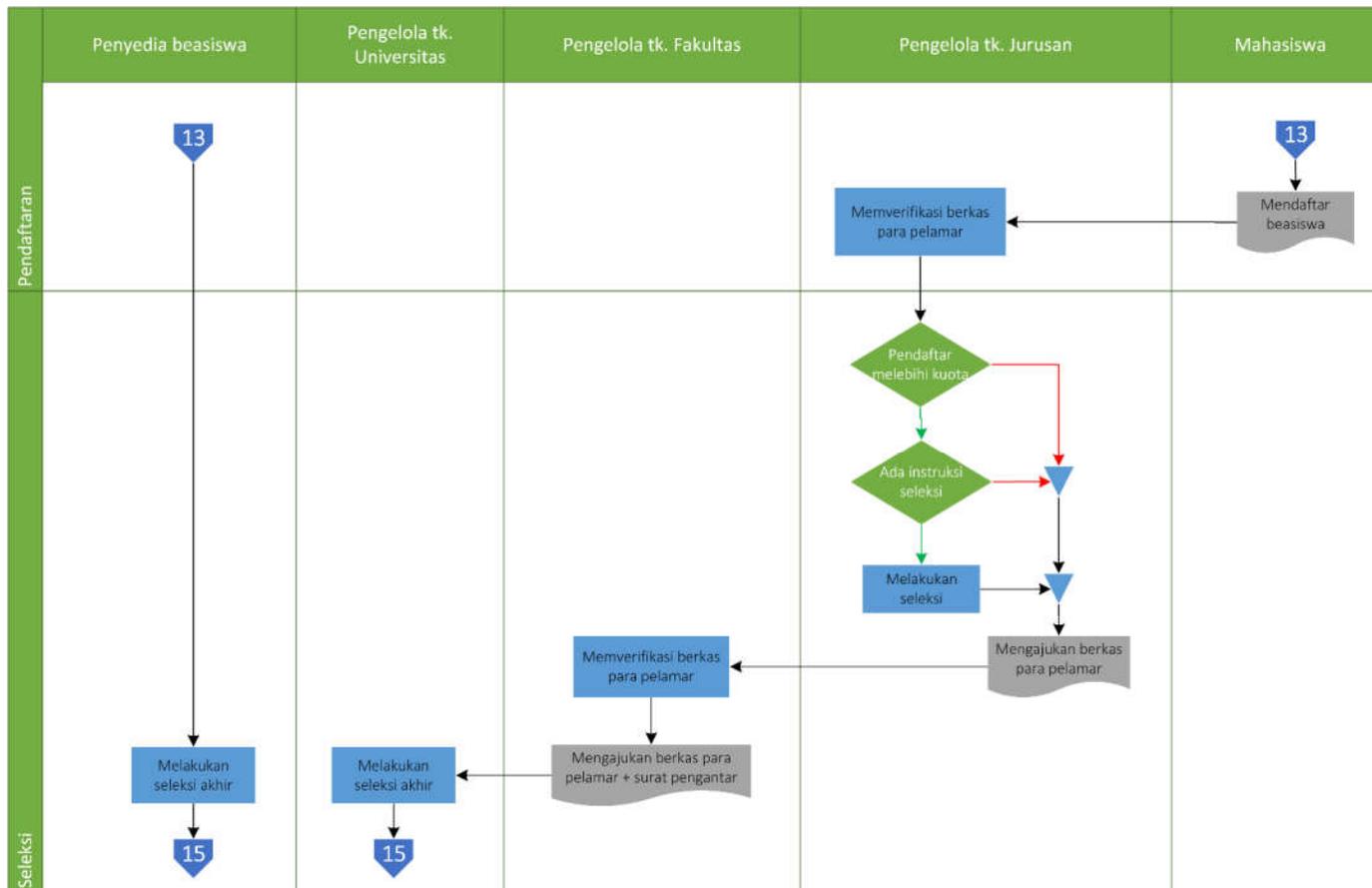
Tabel 1. Makna Simbol-Simbol Diagram Alir (Lucid Software Inc., 2018)

Simbol	Nama	Makna
	<i>Start/End</i>	Indikator mulai atau berakhirnya kegiatan yang menjadi objek diagram.
	<i>Process</i>	Menggambarkan proses, aksi, atau fungsi.
	<i>Document</i>	Melambungkan penerimaan dan/atau pemberian dokumen.
	<i>Decision</i>	Mengandung pertanyaan yang harus dijawab dengan <i>ya/tidak</i> atau <i>benar/salah</i> — pada Gambar 2, respon positif dan negatif dibedakan dengan anak panah hijau dan merah. Tiap jawaban memiliki konsekuensi masing-masing.
	<i>Off-page connector/Link</i>	Menghubungkan elemen-elemen <i>flowchart</i> yang ukurannya lebih dari satu halaman. Biasanya simbol ini disertakan dengan nomor halaman untuk mempermudah penelusuran.
	<i>Or</i>	Mengindikasikan bahwa aliran proses terbelah menjadi dua atau lebih.
	<i>Merge</i>	Menggabungkan beberapa jalur menjadi satu.

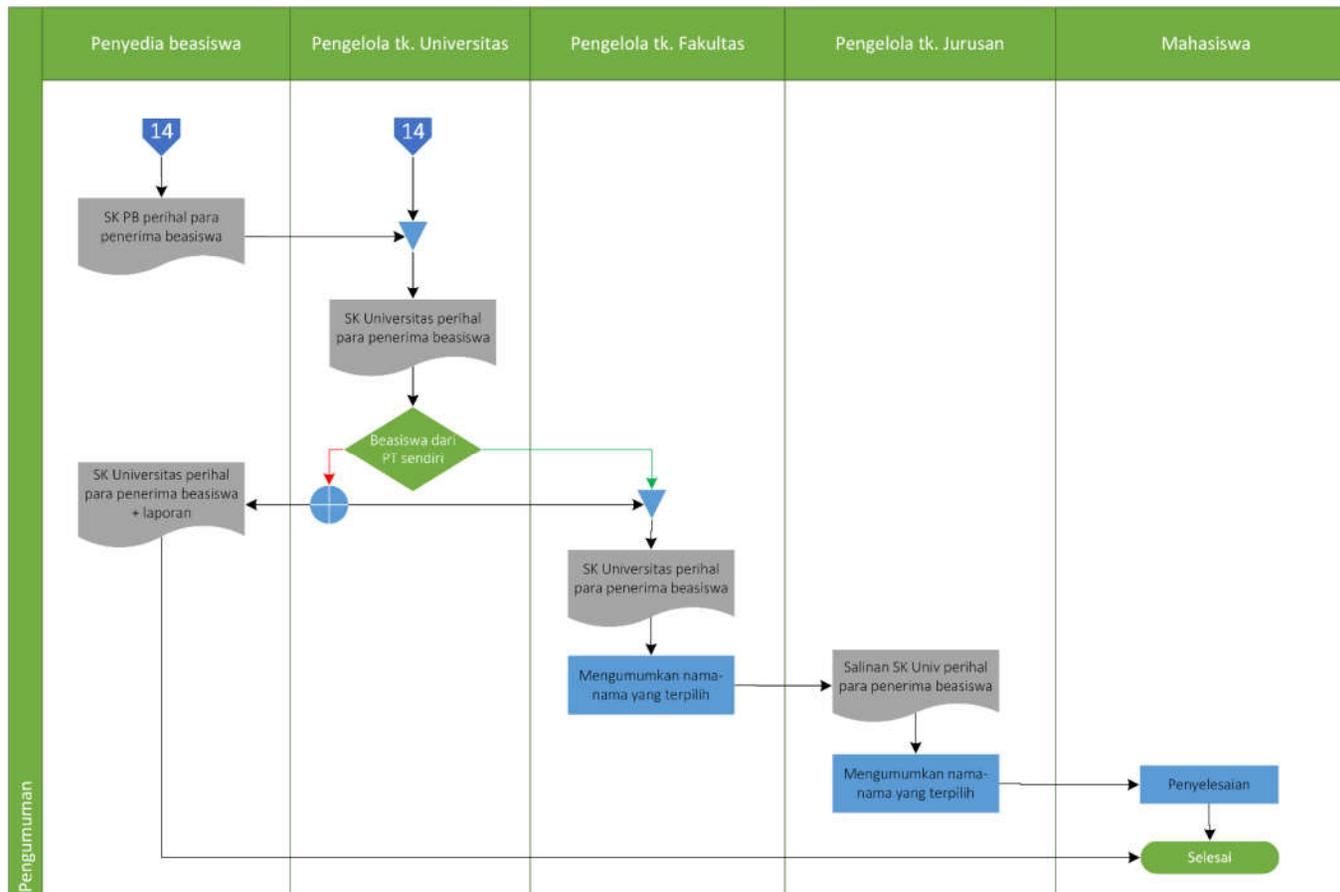
Diagram alirnya sendiri diekspos pada Gambar 2. Tampak bahwa aktivitas “**mempersiapkan berkas pelamar**” menjangkau tiga *stakeholder* secara serempak. Peralnya, pengurusan dokumen-dokumen pendaftaran bukan beban mahasiswa/i semata. Di balik beberapa syarat, misalnya surat keterangan layak menerima beasiswa dan surat pernyataan keabsahan dokumen, ada andil pihak Jurusan dan pengelola tingkat fakultas. Mengenai kegiatan “**memverifikasi berkas pelamar**”, hakikatnya adalah pengecekan kelengkapan dan validasi berkas.



Gambar 2. Diagram Alir Pengelolaan Beasiswa di FMIPA Universitas Lampung.



Gambar 2. (lanjutan)



Gambar 2. (lanjutan)

Di samping kapasitas pemohon terhadap kriteria beasiswa, sejumlah beasiswa seperti Beasiswa BI, Beasiswa YPKAAR, dan Beswan Djarum turut mempertimbangkan faktor-faktor lainnya dalam menentukan kelulusan seorang pelamar, misalnya keikutsertaan mahasiswa/i dalam agenda-agenda penyedia beasiswa dan keterpilihan sebelumnya —diterapkan pada para pelamar yang pernah menerima beasiswa bersangkutan pada tempo yang lalu— atau hasil ujian wawancara/tertulis. Penyuplai Beasiswa BI bahkan menekankan bahwa nama-nama kandidat yang dulu tidak aktif berpartisipasi dalam kegiatan Generasi Baru Indonesia (GenBI) akan dimasukkan dalam daftar hitam (*black list*), sehingga mereka tidak berpeluang untuk menjadi penerima pada periode kedua (Kurniawan, 2018a). Namun, perspektif- perspektif tadi berada di level universitas atau mungkin korporasi yang jenjangnya lebih tinggi dari fakultas, sehingga tak akan diintegrasikan ke dalam sistem.

Penetapan penerima beasiswa dikukuhkan melalui Surat Keputusan (SK) yang dicetuskan oleh pihak Universitas atau penyedia beasiswa, tergantung siapa yang menyeleksi. Dari mana pun asalnya, seluruh SK tersebut harus sampai ke tangan pengelola tingkat universitas demi keperluan pengumuman. SK kemudian diserahkan ke pihak Fakultas supaya salinannya bisa diteruskan ke pengelola tingkat jurusan. Kedua kubu lantas menyiarkan kabar dalam SK kepada segenap *civitas* akademik FMIPA via mading atau situs resmi fakultas dan jurusan. Jika tim seleksi akhir adalah universitas sementara beasiswa bersumber dari luar, maka universitas terikat untuk mengirimkan SK yang sama kepada penyuplai beasiswa (Kemristekdikti, 2018).

Proses “**Penyelesaian**” di Gambar 2 mengilustrasikan syarat-syarat lanjutan yang mesti dipenuhi oleh para penerima berikut tenggat waktunya, yang umumnya tertera pula di dalam SK. Bank Indonesia misalnya, menagih fotokopi KTP, KTM, buku rekening, dan formulir tes urine (Kurniawan, 2018d).

### **2.3. *Multiple Criteria Decision Making (MCDM)***

Seiring waktu berjalan, peradaban manusia dihadapkan pada momen-momen yang tidak hanya terpaku pada "ya" dan "tidak". Sebagai wujud dari dinamika kehidupan yang makin kompleks, pada beberapa aktivitas, suatu entitas tidak sepenuhnya baik atau sepenuhnya buruk. Ada kalanya sebuah komponen pendukung tidak boleh diperbanyak karena akan mengundang dampak yang tak diinginkan. Seringkali pula komponen-komponen penghambat tidak dapat dihilangkan atau diabaikan. Akhirnya, manusia ditinggalkan dengan sederetan skenario, yang harus ia pilih salah satunya agar pemenuhan tujuan/kebutuhan dari atau melalui aktivitas bersangkutan, meskipun tidak maksimal, setidaknya optimal.

Dalam menentukan skenario mana yang harus diikuti, manusia umumnya memanfaatkan intuisi. Sayangnya, ihwal yang dijumpai bertambah besar dan rumit, hingga naluri bukan lagi alat yang efektif untuk menanganinya. Pembelajaran guna menaklukkan kondisi seperti ini pun berjalan. Titik ujungnya adalah tumbuhnya disiplin ilmu baru yang diberi nama *Multiple Criteria Decision Making (MCDM)*.

Melalui MCDM, lahir berbagai metode yang menyokong prosedur pengambilan putusan dalam beragam konteks. Titel-titel berikut hanya sekelumit dari kuantitas

yang beredar, namun populer: *Multi-Attribute Utility Theory* (MAUT), *Analytic Hierarchy Process* (AHP), *Fuzzy Theory*, *Case-Based Reasoning* (CBR), *Data Envelopment Analysis* (DEA), *Simple Multi-Attribute Rating Technique* (SMART), *Goal Programming*, ELECTRE, PROMETHEE, *Simple Additive Weighting* (SAW), *Technique for Order of Preference by Similarity to Ideal Solution* (TOPSIS). Velasquez & Hester (2013) menyatakan bahwa tiap metode ini mempunyai keunggulan dan kekurangan, serta area penerapannya masing-masing. Uraian tentang ini ditunjukkan pada Tabel 2.

Tabel 2. Ikhtisar Metode-Metode MCDM (Velasquez & Hester, 2013)

<b>Metode</b>	<b>Kelebihan</b>	<b>Kekurangan</b>	<b>Area Aplikasi</b>
<i>Multi-Attribute Utility Theory</i> (MAUT)	<ul style="list-style-type: none"> <li>➤ Memperhitungkan ketidakpastian.</li> <li>➤ Dapat menerima preferensi.</li> </ul>	<ul style="list-style-type: none"> <li>➤ Butuh banyak masukan.</li> <li>➤ Preferensi yang dimasukkan harus tepat.</li> </ul>	Ekonomi, keuangan, aktuarial, manajemen air, manajemen energi, pertanian.
<i>Analytic Hierarchy Process</i> (AHP)	<ul style="list-style-type: none"> <li>➤ Mudah digunakan.</li> <li>➤ Skala proses dapat berubah-ubah sesuai kebutuhan.</li> <li>➤ Struktur hierarki dapat diatur sedemikian rupa agar cocok dengan banyak masalah.</li> <li>➤ Tidak membutuhkan banyak data.</li> </ul>	<ul style="list-style-type: none"> <li>➤ Jika ada ketergantungan antara kriteria dan alternatif, akan terjadi inkonsistensi antara keputusan dan peringkat kriteria.</li> <li>➤ <i>Rank reversal</i>.</li> </ul>	Masalah-masalah yang terkait dengan performa, manajemen sumber daya, kebijakan dan strategi perusahaan, kebijakan publik, strategi politik, dan perencanaan.
<i>Case-Based Reasoning</i> (CBR)	<ul style="list-style-type: none"> <li>➤ Tidak butuh banyak data.</li> <li>➤ Pemeliharaan sedikit.</li> <li>➤ Hasil yang diberikan semakin membaik dari waktu ke waktu.</li> <li>➤ Dapat beradaptasi dengan perubahan yang terjadi di lingkungan.</li> </ul>	<ul style="list-style-type: none"> <li>➤ Sensitif pada data yang tak konsisten.</li> <li>➤ Butuh banyak kasus.</li> </ul>	Bisnis, asuransi kendaraan, kedokteran, dan desain teknik.
<i>Data Envelopment Analysis</i> (DEA)	<ul style="list-style-type: none"> <li>➤ Mampu menangani beberapa masukan dan keluaran sekaligus.</li> <li>➤ Efisiensi dapat dianalisis dan dikuantifikasi.</li> </ul>	<ul style="list-style-type: none"> <li>➤ Tidak menangani data yang tidak akurat.</li> <li>➤ Mengasumsikan semua masukan dan keluaran persis diketahui.</li> </ul>	Ekonomi, kedokteran, utilitas, keamanan jalan, pertanian, ritel, dan masalah-masalah bisnis.

Tabel 2. (lanjutan)

<b>Metode</b>	<b>Kelebihan</b>	<b>Kekurangan</b>	<b>Area Aplikasi</b>
<i>Fuzzy Set Theory</i>	<ul style="list-style-type: none"> <li>➤ Mampu menerima masukan yang tidak akurat.</li> <li>➤ Memperhitungkan informasi yang tidak lengkap.</li> </ul>	<ul style="list-style-type: none"> <li>➤ Sulit dikembangkan.</li> <li>➤ Membutuhkan banyak simulasi sebelum bisa digunakan.</li> </ul>	Teknik, ekonomi, lingkungan, sosial, kesehatan, dan manajemen.
<i>Simple Multi-Attribute Rating Technique (SMART)</i>	<ul style="list-style-type: none"> <li>➤ Sederhana.</li> <li>➤ Membolehkan segala jenis teknik penetapan bobot.</li> <li>➤ Pekerjaan pengambil keputusan semakin berkurang.</li> </ul>	Prosedurnya menyusahkan.	Lingkungan, konstruksi, transportasi dan logistik, militer, masalah produksi dan perakitan.
<i>Goal Programming</i>	<ul style="list-style-type: none"> <li>➤ Mampu menangani masalah-masalah berskala besar.</li> <li>➤ Dapat menghasilkan alternatif yang tak berhingga.</li> </ul>	Kemampuannya untuk menurunkan bobot koefisien biasanya perlu dikombinasikan dengan metode MCDM lain.	Perencanaan produksi, penjadwalan, perawatan kesehatan, seleksi portofolio, sistem distribusi, perencanaan energi, manajemen waduk air, manajemen satwa liar.
ELECTRE	Memperhitungkan ketidakpastian dan ketidakjelasan.	<ul style="list-style-type: none"> <li>➤ Proses dan hasil metode ini sulit dijelaskan dalam istilah awam.</li> <li>➤ Prinsip <i>outrank</i> menyebabkan kekuatan dan kelemahan masing-masing alternatif tidak diidentifikasi secara langsung.</li> </ul>	Energi, ekonomi, lingkungan, manajemen air, dan masalah transportasi.
PROMETHEE	<ul style="list-style-type: none"> <li>➤ Mudah digunakan.</li> <li>➤ Tidak membutuhkan asumsi kalau kriteria-kriteria itu berbanding.</li> </ul>	Tidak menyediakan metode yang jelas untuk menetapkan bobot.	Lingkungan, hidrologi, manajemen air, bisnis dan keuangan, kimia, logistik dan transportasi, produksi dan perakitan, energi, pertanian.
<i>Simple Additive Weighting (SAW)</i>	<ul style="list-style-type: none"> <li>➤ Kemampuan untuk mengompensasi kriteria.</li> <li>➤ Bersifat intuitif.</li> <li>➤ Kalkulasinya sederhana, tidak memerlukan program komputer yang kompleks.</li> </ul>	<ul style="list-style-type: none"> <li>➤ Estimasi yang dihasilkan tidak selalu mencerminkan keadaan sebenarnya.</li> <li>➤ Hasil bisa saja tidak logis.</li> </ul>	Manajemen air, bisnis, dan manajemen keuangan.

Tabel 2. (lanjutan)

<b>Metode</b>	<b>Kelebihan</b>	<b>Kekurangan</b>	<b>Area Aplikasi</b>
<i>Technique for Order Preferences by Similarity to Ideal Solutions</i> (TOPSIS)	<ul style="list-style-type: none"> <li>➤ Prosesnya sederhana.</li> <li>➤ Mudah digunakan dan diprogram.</li> <li>➤ Jumlah tahapan sama, terlepas dari jumlah atribut.</li> </ul>	<ul style="list-style-type: none"> <li>➤ Tidak mempertimbangkan korelasi antar atribut.</li> <li>➤ Sulit dalam hal penetapan bobot dan menjaga konsistensi keputusan.</li> </ul>	Manajemen dan logistik rantai suplai, teknik, sistem produksi, bisnis dan pemasaran, lingkungan, sumber daya manusia, dan manajemen sumber air.

Pemilihan AHP dan TOPSIS sebagai alat penuntasan masalah didasari oleh alasan-alasan berikut:

1. Penetapan penerima beasiswa termasuk dalam ranah pengelolaan sumber daya manusia. Tabel 2 menunjukkan bahwa AHP dan TOPSIS sangat cocok diterapkan pada perkara-perkara yang berkaitan dengan sumber daya.
2. AHP dan TOPSIS, bila diimplementasikan bersamaan, saling menutupi kekurangan masing-masing.
3. Kelemahan AHP sebenarnya tidak akan berlaku karena keadaan yang melingkupi penelitian menjamin hal tersebut. Namun, jika hanya mengandalkan AHP, kalkulasi yang tergelar akan alot ketika menyentuh fase pembobotan alternatif, akibat ukuran alternatif yang (biasanya) tinggi. Oleh karena itu, perlu digandeng TOPSIS yang kerangka kerjanya ajek, tidak bergantung pada total unsur yang terlibat. Modus operandi TOPSIS, secara natural, mampu membereskan sisa perhitungan tanpa kesusahan yang berarti.

Sementara itu, evaluasi pertalian atribut, determinasi bobot, dan pengecekan konsistensi keputusan adalah mutlak karakteristik AHP. Dengan demikian,

tampak jelas bahwa AHP merupakan jawaban atas kelemahan TOPSIS. Semua aspek tersebut akan menyempurnakan performa TOPSIS.

4. Manurung (2011) telah membuktikan bahwa gabungan AHP dan TOPSIS bisa digunakan untuk memecahkan masalah penyeleksian penerima beasiswa. Yusuf, Koniyo, dan Novian (2013) bahkan bertindak lebih jauh lagi dengan mendemonstrasikan betapa jituanya kombinasi AHP-TOPSIS, dibanding TOPSIS sendirian, dalam memprediksi peringkat kandidat penggenggam beasiswa BBM dan PPA di Universitas Negeri Gorontalo.

#### **2.4. *Analytic Hierarchy Process (AHP)***

##### **2.4.1. Pengukuran Relatif**

Selaku ketua tim penasihat ACDA (*Arms Control and Disarmament Agency*) untuk perdagangan senjata, Thomas Lorie Saaty terkendala sebuah masalah: bagaimana mengukur kuantitas faktor-faktor tak berwujud (*intangible factors*) ketika jumlahnya amat banyak dan tak mungkin diabaikan (Saaty, 2005). Selain itu, cara serta skala ukur masing-masing faktor yang tak sama membuat proses perbandingan antar faktor kian sulit (Saaty, 2005). Guna mengatasi keduanya, Saaty pun merumuskan teknik komparasi yang mampu mengakomodir segala perbedaan pengukuran, sekaligus menyatukan seluruh faktor, baik *tangible* maupun *intangible*, dalam satu konteks perbandingan yang adil. Teknik tersebut dikenal sampai sekarang dengan nama *Analytic Hierarchy Process* atau AHP.

AHP adalah metode pengambilan keputusan multi-kriteria yang menurunkan prioritas tiap alternatif dari kekuatan relatif satu kriteria terhadap kriteria lainnya.

Dikatakan *relatif* karena nilai kekuatan yang dimaksud bergantung pada seberapa penting kriteria tersebut bagi para pengambil keputusan. Misalnya, sebelum membeli mobil baru, terdapat sekian faktor yang perlu dipertimbangkan: harga, kapasitas penumpang, desain eksterior, dan desain interior. Seseorang dari kalangan menengah ke bawah akan menempatkan harga sebagai kriteria yang paling penting, mengingat dana yang ia miliki terbatas. Sementara itu, seseorang dari kalangan kaya raya lebih mengutamakan desain eksterior kendaraan karena tuntutan lingkungan yang serba mewah.

Menurut Vargas (2010), perbandingan relatif menggunakan AHP sebenarnya dapat dilakukan dengan banyak cara. Namun, skala yang diusulkan Saaty merupakan sarana yang paling sering dipakai (Vargas, 2010). Skala yang diperlihatkan oleh Tabel 3 ini menunjukkan seberapa dominan satu elemen terhadap elemen lainnya.

Tabel 3. Skala Dasar Angka Mutlak (Saaty, 1987)

<b>Intensitas Kepentingan</b>	<b>Definisi</b>	<b>Penjelasan</b>
1	<i>Equal importance</i>	Kedua aktivitas memberikan kontribusi yang besarnya seimbang kepada tujuan
2	<i>Weak or slight</i>	
3	<i>Moderate importance</i>	Kontribusi satu aktivitas terhadap tujuan sedikit mengungguli aktivitas yang lain
4	<i>Moderate plus</i>	
5	<i>Strong importance</i>	Kontribusi satu aktivitas kepada tujuan jelas unggul atas aktivitas yang lain
6	<i>Strong plus</i>	

Tabel 3. (lanjutan)

<b>Intensitas Kepentingan</b>	<b>Definisi</b>	<b>Penjelasan</b>
7	<i>Very strong or demonstrated importance</i>	Kontribusi satu aktivitas terhadap tujuan sangat mengungguli aktivitas yang lain; aktivitas tersebut sangat disukai, dominasinya terlihat dalam praktik
8	<i>Very very strong</i>	
9	<i>Extreme importance</i>	Kontribusi satu aktivitas terhadap tujuan mutlak mengungguli aktivitas yang lain
1.1 – 1.9		Jika kontribusi kedua aktivitas amat dekat

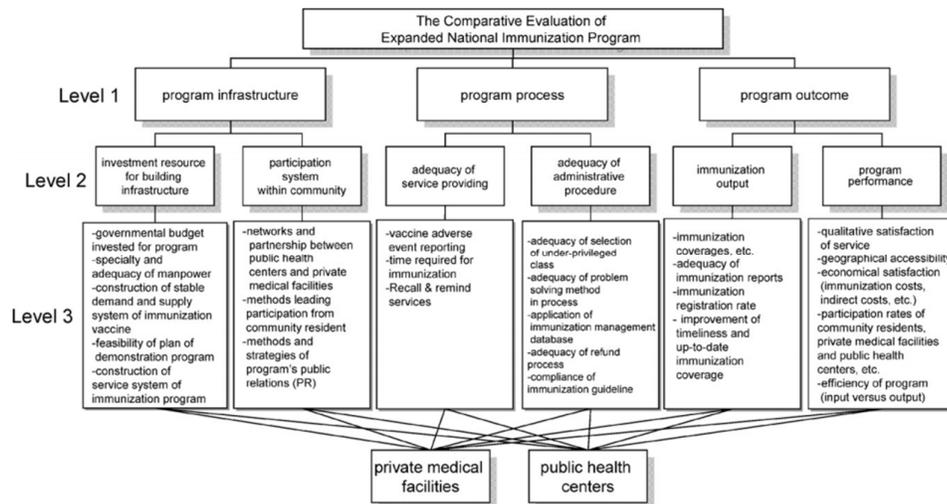
#### 2.4.2. Struktur Hierarki

Ada satu hal lagi yang menarik perhatian Saaty, yaitu: keputusan akan semakin sukar diambil dari suatu kasus seiring dengan meningkatnya kompleksitas kasus tersebut. Beliau lalu mempelajari berbagai solusi yang dilakukan manusia di masa lampau untuk menangani kejadian-kejadian kompleks, hingga ia sampai pada sebuah persamaan, yakni penstrukturan hierarki (Forman & Gass, 2001). Struktur ini, selain mempermudah proses penyelesaian masalah, juga membantu menyamakan persepsi dan/atau pola pikir para pemangku kepentingan. Mengingat manfaatnya yang begitu besar, Saaty akhirnya menerapkan struktur hierarki di AHP.

Konsep penstrukturan hierarki adalah memecah suatu masalah besar menjadi sub-sub masalah yang lebih kecil, yang kemudian sub-sub tersebut dikelompokkan ke dalam tingkatan-tingkatan atau level-level. Pada AHP, level teratas adalah tujuan yang hendak dicapai dengan diambilnya keputusan terbaik. Level kedua berisikan

kriteria-kriteria yang dipertimbangkan dalam proses pengambilan keputusan. Jika satu atau lebih kriteria tadi memiliki subkriteria, maka semua subkriteria tersebut ditempatkan di level yang lebih bawah lagi. Alternatif-alternatif keputusan yang dapat dipilih diletakkan di level terakhir. (Bhushan & Rai, 2004; Saaty, 1991)

Contoh pemakaian struktur hierarki ditunjukkan oleh Gambar 3. Shin, et al. (2009) menggunakannya untuk menentukan pelaksana agenda imunisasi nasional di Korea Selatan.



Gambar 3. Contoh Struktur Hierarki. (Shin, et al., 2009)

### 2.4.3. Teknis AHP

Garis besar pembentukan prioritas dengan metode AHP dijabarkan dalam poin-poin berikut:

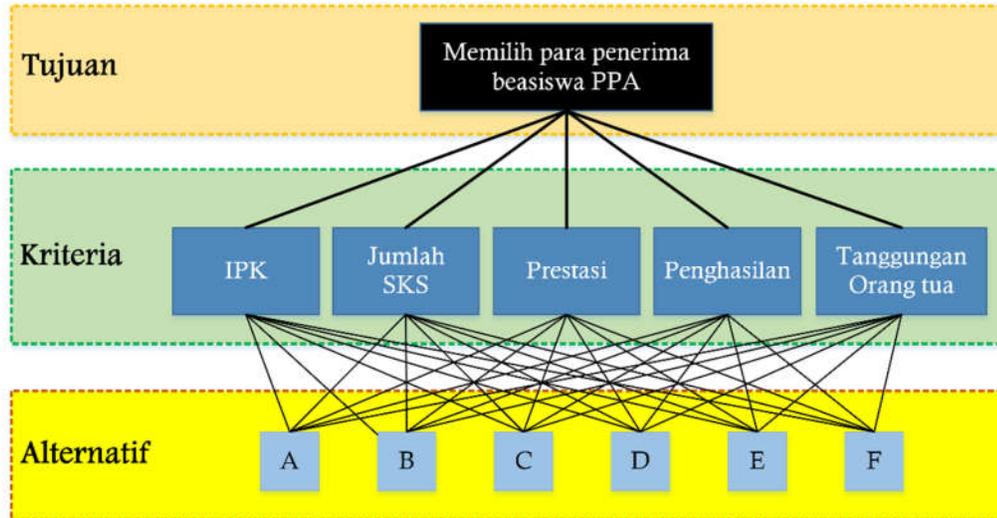
- Mendefinisikan masalah yang sedang dihadapi.
- Membuat hierarki keputusan. Jalan keluar untuk masalah yang ditemukan umumnya menjadi tujuan hierarki.

- c) Membuat matriks perbandingan berpasangan untuk tiap level dalam hierarki.
- d) Mengekstrak prioritas dari matriks perbandingan berpasangan. Prioritas sebuah elemen selanjutnya digunakan untuk menimbang prioritas elemen-elemen yang tepat berada di bawahnya. Setelah itu, seluruh nilai yang diperoleh alternatif dijumlahkan untuk mendapatkan prioritas global alternatif terhadap tujuan.  
(Saaty, 2008a)

#### 2.4.4. Fungsi AHP dalam Sistem Informasi

Satu atau beberapa kriteria diperhatikan lebih seksama pada suatu beasiswa. IPK, misalnya, sebagai tolok ukur kecerdasan, menjadi sorotan dalam beasiswa khusus mahasiswa intelek. Di beasiswa lain, kondisi finansial pelamar menyedot paling banyak animo karena fokusnya adalah membantu pelajar yang tak mampu secara ekonomi. Alaminya, tim penyeleksi masing-masing beasiswa akan memberikan bobot yang paling tinggi pada dua kriteria ini.

Ketika konteksnya sistem informasi berbasis komputer, tugas barusan diambil alih oleh AHP. Dengan AHP, preferensi penyedia beasiswa terhadap kriteria tertentu bisa dikonversi ke wujud bilangan kuantitatif. Masukan yang dibutuhkan adalah matriks yang tiap selnya diisi oleh skala dasar Saaty (lihat Tabel 3), yang disebut juga matriks perbandingan berpasangan. Contoh struktur hierarki beasiswa PPA dan matriks perbandingan berpasangannya dapat dilihat di Gambar 4 dan Tabel 4.



Gambar 4. Struktur Hierarki Beasiswa PPA.

Tabel 4. Matriks Perbandingan Berpasangan Kriteria Beasiswa PPA

	IPK	Jumlah SKS	Prestasi	Penghasilan	Tanggung Ortu
IPK	1	3	6	9	9
Jumlah SKS	$\frac{1}{3}$	1	5	3	3
Prestasi	$\frac{1}{6}$	$\frac{1}{5}$	1	2	2
Penghasilan	$\frac{1}{9}$	$\frac{1}{3}$	$\frac{1}{2}$	1	1
Tanggung Ortu	$\frac{1}{9}$	$\frac{1}{3}$	$\frac{1}{2}$	1	1

Baris pertama matriks pada Tabel 4 menunjukkan bahwa kriteria *IPK* 3 kali lebih penting daripada *Jumlah SKS*, 6 kali lebih penting daripada *Prestasi*, dan 9 kali lebih penting daripada *Penghasilan* dan *Jumlah Tanggungan Orang tua*. Adalah logis jika kemudian dikatakan, "Kriteria *Jumlah SKS*, *Prestasi*, *Penghasilan*, dan *Jumlah Tanggungan Orang tua*, berturut-turut,  $\frac{1}{3}$  kali,  $\frac{1}{6}$  kali,  $\frac{1}{9}$  kali, dan  $\frac{1}{9}$  kali lebih penting daripada *IPK*". Inilah asal-muasal semua nilai di kolom *IPK*. Setelah angka disematkan pada satu perbandingan, maka perbandingan

pasangannya mendapatkan resiprok dari angka tersebut (Bhushan & Rai, 2004; Saaty, 2003).

2.4.5. Sintesis Prioritas

Anggap  $n$  kriteria  $A_1, A_2, \dots, A_n$  saling dibandingkan dalam matriks  $A$ . Elemen-elemennya,  $a_{ij}$ , mengindikasikan keutamaan kriteria  $A_i$  terhadap  $A_j$ ,  $i, j = 1, \dots, n$ .

$$\begin{matrix} & A_1 & \cdots & A_n \\ A_1 & \left[ \begin{array}{ccc} 1 & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ 1/a_{1n} & \cdots & 1 \end{array} \right] & \cdots & \cdots \end{matrix} \dots\dots\dots (1)$$

Pertama-tama, asumsikan angka-angka yang terekam dalam matriks  $A$  berasal dari pengukuran yang presisi  $w_1, w_2, \dots, w_n$  (Saaty, 1990; Saaty, 2008b). Dalam peristiwa ini, relasi antara bobot  $w_i$  dan  $a_{ij}$  adalah  $\frac{w_i}{w_j} = a_{ij}$  atau  $w_i = w_j a_{ij}$  dengan

$i, j = 1, \dots, n$ , sehingga

$$\begin{aligned}
 a_{11} &= w_1/w_1, a_{12} = w_1/w_2, \dots, a_{1n} = w_1/w_n \\
 a_{21} &= w_2/w_1, a_{22} = w_2/w_2, \dots, a_{2n} = w_2/w_n \\
 a_{31} &= w_3/w_1, a_{32} = w_3/w_2, \dots, a_{3n} = w_3/w_n \\
 &\vdots \qquad \qquad \qquad \vdots \qquad \qquad \qquad \ddots \qquad \qquad \qquad \vdots \\
 a_{n1} &= w_n/w_1, a_{n2} = w_n/w_2, \dots, a_{nn} = w_n/w_n.
 \end{aligned}$$

Maka, matriks di atas berubah menjadi

$$\begin{matrix} & A_1 & \cdots & A_n \\ A_1 & \left[ \begin{array}{ccc} w_1/w_1 & \cdots & w_1/w_n \\ \vdots & \ddots & \vdots \\ w_n/w_1 & \cdots & w_n/w_n \end{array} \right] & \cdots & \cdots \end{matrix} \dots\dots\dots (2)$$

Matriks  $A$  pada butir (2) menyanggah properti transitif karena memenuhi syarat  $a_{ij} \cdot a_{jk} = a_{ik}$ ,  $i, j, k = 1, \dots, n$ . Matriks yang begitu disebut *konsisten* dan memiliki bentuk ideal (Alonso & Lamata, 2006; Saaty, 1990; Sinaga, 2010). Sayangnya, relasi barusan tidak akan bertahan pada kasus-kasus umum akibat dua alasan:

- a) Penakaran fisik tak pernah benar-benar tepat secara matematis; dibutuhkan ruang bagi deviasi.
- b) Dalam kegiatan pengambilan keputusan oleh manusia, kadar deviasi biasanya lebih besar.

Jika seluruh komponen baris  $i$  matriks  $A$  dikalikan, berturut-turut, dengan  $w_1, w_2, w_3, \dots, w_n$  akan dihasilkan sederet  $w_i$  yang identik.

$$\frac{w_i}{w_1} w_1 = w_i, \frac{w_i}{w_2} w_2 = w_i, \dots, \frac{w_i}{w_j} w_j = w_i, \dots, \frac{w_i}{w_n} w_n = w_i$$

Namun, lazimnya, baris yang diperoleh merepresentasikan nilai-nilai statistik di sekitar  $w_i$ . Maka itu, realistis rasanya bila  $w_i$  sama dengan rerata dari nilai-nilai ini.

$$w_i = \text{avg}(a_{i1}w_1, a_{i2}w_2, \dots, a_{in}w_n)$$

$$w_i = \frac{1}{n} \sum_{j=1}^n a_{ij}w_j \text{ atau } nw_i = \sum_{j=1}^n a_{ij}w_j \text{ dimana } i = 1, 2, \dots, n \dots\dots\dots(3)$$

Persamaan (3) yang masih terlalu “kaku” ternyata menyulitkan pencarian bobot  $w_1, w_2, \dots, w_n$ . Faktanya,  $a_{ij}$  cenderung dekat ke  $w_i/w_j$ , yang artinya ia merupakan perturbasi kecil dari rasio ini (Saaty, 2008b). Berubahnya  $a_{ij}$  memaksakan alterasi  $n$  menjadi  $c$  (Saaty, 1990, 2008b), menimbulkan solusi lain yang berpadanan dengan (3).

$$cw_i = \sum_{j=1}^n a_{ij}w_j \text{ dimana } i = 1, \dots, n \dots\dots\dots(4)$$

Penjelasan tentang perhitungan  $c$  dan  $w$  dimulai dengan mencantumkan formula solusi untuk keadaan ideal:

$$A\vec{w} = \begin{bmatrix} w_1/w_1 & \dots & w_1/w_n \\ \vdots & \ddots & \vdots \\ w_n/w_1 & \dots & w_n/w_n \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix} = n \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix} = n\vec{w} \text{ (Saaty, 1990).....(5)}$$

Tampak bahwa Persamaan (5) berpola sama dengan Teorema Perron-Frobenius untuk matriks bujur sangkar ber-entri positif (Alonso & Lamata, 2006).

$$A\vec{v} = \lambda\vec{v} \text{.....(6)}$$

dimana  $A$  = matriks positif berordo  $n \times n$ ,  
 $\vec{v}$  = *eigenvector* matriks  $A$ ,  
 dan  $\lambda$  = *eigenvalue* matriks  $A$ .

Persamaan (5) dapat diselesaikan dengan cara mengubahnya ke bentuk linear homogen  $(A - nI)\vec{w} = 0$ , dimana  $I$  merupakan matriks identitas berordo  $n$ . Persamaan linear homogen memiliki tak hingga solusi tak nol  $\vec{w}$  jika dan hanya jika matriks  $A - nI$  *singular* (tidak mempunyai invers). Syarat matriks yang seperti ini adalah determinannya sama dengan nol. (Kardi, 2011; Saaty, 1990; Sinaga, 2010)

$$A\vec{w} = n\vec{w}$$

$$A\vec{w} = nI\vec{w}$$

$$A\vec{w} - nI\vec{w} = 0$$

$$(A - nI)\vec{w} = 0 \text{.....(7)}$$

$$\det(A - nI) = 0 \text{.....(8)}$$

Setelah mengembangkan rumus determinan, selanjutnya diperoleh polinomial  $n^n + c_1 \cdot n^{n-1} + \dots + c_{n-1} \cdot n + c_n$ , dimana  $c_1, c_2, \dots, c_n$  adalah koefisien persamaan. Produk dari persamaan ini adalah  $m$  buah  $n$ , beberapa di antaranya identik, sedangkan beberapa lainnya bilangan kompleks. Akhirnya, didapat kumpulan  $\vec{w}$  sesudah semua nilai  $n$  tersebut disubstitusikan ke Persamaan (7). (Kardi, 2011)

Lantaran tiap baris matriks  $A$  merupakan kombinasi linear dari baris pertama, maka semua *eigenvalue*-nya, kecuali satu, sama dengan nol. Jumlah *eigenvalue* suatu matriks didapat dari jumlah elemen-elemen diagonal (*trace*) matriks tersebut. Dengan demikian, pada posisi ideal,  $n$  adalah *eigenvalue* matriks  $A$  (Alonso & Lamata, 2006; Saaty, 1990, 2005, 2008b); sedangkan  $\vec{w}$  menempati status *eigenvector*  $\vec{v}$ .

Sebaliknya, ketika  $A$  tak lagi konsisten namun tetap resiprok, *eigenvalue*-nya berganti menjadi  $c$  (lihat Persamaan 4) yang dilambangkan dengan notasi  $\lambda_{max}$ , menandakan *eigenvalue* terbesar dari matriks  $A$  (Bhushan & Rai, 2004; Saaty, 1990). Pembuktian lebih lanjut mendemonstrasikan bahwa prioritas  $w_1, w_2, \dots, w_n$  yang diinginkan terwakili pula oleh *eigenvector*  $\vec{v}$ , asalkan inkonsistensi keputusan dijaga dalam batas-batas tertentu (Saaty, 2003). Alhasil, demi terwujudnya kooperasi dengan lingkungan yang disesaki ketidakstabilan, Persamaan (5) pun dimodifikasi:

$$A\vec{w} = \lambda_{max}\vec{w} \text{ (Saaty, 1990, 2008b) } \dots\dots\dots (9)$$

dimana  $A$  = matriks positif resiprok berordo  $n \times n$ ,  
 $\vec{w}$  = prioritas kriteria/*eigenvector* matriks  $A$ ,  
dan  $\lambda_{max}$  = *eigenvalue* terbesar matriks  $A$ .

#### 2.4.6. Sintesis Prioritas Menggunakan Pendekatan

Di samping cara yang baru saja dibahas, yang untuk seterusnya disebut "AHP original", ada bentuk AHP lain yang eksis dan dipakai oleh berbagai entitas untuk menuntaskan perkara yang dihadapinya (Kunz, 2010). Sejarah munculnya bentuk-bentuk lain ini – yang selanjutnya dirujuk dengan kata "pendekatan" – tidak diketahui dengan pasti. Pendekatan-pendekatan tersebut terdaftar dalam Tabel 5.

Tabel 5. Pendekatan-Pendekatan AHP (Saaty, 1991)

Pendekatan	Kaidah Kalkulasi Prioritas ( $\bar{w}$ )	Keterangan
1	<p>a) Mengalikan elemen-elemen di baris yang sama. Hasil kali disimpan dalam variabel <math>m_1, m_2, \dots, m_n</math> dimana <math>n</math> adalah ordo matriks.</p> <p>b) Menghitung rata-rata geometrik dari hasil perkalian <math>p_1 = \sqrt[n]{m_1}, \dots, p_n = \sqrt[n]{m_n}</math>.</p> <p>c) Menormalisasi hasil pengakaran</p> $w_i = \frac{p_i}{\sum_{i=1}^n p_i}, \text{ dengan } i = 1, \dots, n.$	<p>Tidak dijamin akurat untuk perkara dengan <math>n &gt; 3</math></p> <p>Dipakai oleh Kunz (2010) dan Sinaga (2010).</p>
2	<p>a) Menormalisasi elemen-elemen di setiap kolom</p> $x'_{ij} = \frac{x_{ij}}{\sum_{i=1}^n x_{ij}}, \text{ dengan } i, j = 1, \dots, n$ <p>dan <math>n</math> adalah ordo matriks.</p> <p>b) Merata-rata bilangan-bilangan yang berada dalam satu baris</p> $w_i = \frac{\sum_{j=1}^n x'_{ij}}{n}, \text{ dengan } i = 1, \dots, n.$	<p>Dipakai oleh Kardi (2006) dan Vargas (2010).</p>
3	<p>a) Memangkatkan matriks perbandingan berpasangan dengan eksponen perdana <math>k = 1</math>.</p> <p>b) Menjumlahkan elemen-elemen di baris yang sama. Hasil penjumlahan disimpan dalam <math>s_1, s_2, \dots, s_n</math> dimana <math>n</math> adalah ordo matriks.</p>	<p>Sifat lain yang dimiliki oleh matriks konsisten ialah <math>A^k = n^{k-1}A</math>, <math>n</math> adalah ordo matriks <math>A</math>; jadi seluruh pangkat dari <math>A</math> pada dasarnya sama dengan <math>A</math>. Matriks yang inkonsisten gagal</p>

Tabel 5. (lanjutan)

Pendekatan	Kaidah Kalkulasi Prioritas ( $\bar{w}$ )	Keterangan
3	c) Menormalisasi hasil penjumlahan $w_i^{k=1} = \frac{s_i}{\sum_{i=1}^n s_i}$ , dengan $i = 1, \dots, n$ . d) Mengulangi langkah (a)-(c), kali ini $k = k + 1$ . e) Mengulangi langkah (d), sampai selisih antara $w_i^k$ dan $w_i^{k+1}$ tak melewati margin yang ditetapkan. (Saaty, 2008a)	mencapai kondisi di atas, oleh karena itu, perlu dipertimbangkan prioritas yang diturunkan langsung dari matriks itu sendiri, prioritas kedua yang diperoleh dari pangkat 2 matriks, dan seterusnya. Dengan memangkatkan matriks, <i>eigenvector</i> yang dihasilkan melukiskan transitivitas pada semua orde. (Saaty, 2003, 2008b)

Saaty (1991) memperingatkan, untuk aplikasi-aplikasi penting, pemakai AHP hendaknya memanfaatkan prosedur penurunan prioritas yang ke-3. Aproksimasi ke-1 dan ke-2, meski memberikan hasil yang dekat dengan *eigenvector* yang dicari, membawa beberapa masalah, salah satunya *rank reversal*.

#### 2.4.7. Validasi Matriks Perbandingan Berpasangan

Pekerjaan tak lantas usai setelah prioritas, atau bobot, tiap kriteria diketahui. Matriks perbandingan berpasangan yang bersangkutan perlu diuji untuk memastikan pembuatnya benar-benar memasukkan angka-angka yang mencerminkan dominasi relatif, bukan poin-poin acak. Hal ini dilakukan dengan membandingkan  $\lambda_{max}$  matriks dengan  $\lambda$  yang dicapai pada status ideal, yang mana adalah ordo matriks  $n$ .

$$CI = \frac{\lambda_{max} - n}{n - 1} \quad \text{dimana } CI = \text{Consistency Index.}$$

$$CR = \frac{CI}{RI} \quad \text{dimana } RI = \text{Random Consistency Index,} \\ \text{dan } CR = \text{Consistency Ratio.}$$

(Bhushan & Rai, 2004; Sinaga, 2010; Vargas, 2010).

Nominal  $\lambda_{max}$  diraih dengan menjumlahkan elemen-elemen yang sekolom kemudian mengalikan vektor yang terbentuk dengan  $\vec{w}$  (Saaty, 1991). Untuk matriks resiprokal positif, berlaku  $\lambda_{max} \geq n$  (Saaty, 2008b). *RI* sendiri merupakan *consistency index* rata-rata dari sampel berukuran 500, yang tersusun dari matriks-matriks resiprok yang digenerasi secara acak menggunakan skala  $1/9, 1/8, \dots, 1, \dots, 8, 9$  (Saaty, 1991). Tabel 6 memperlihatkan *RI* untuk matriks persegi ordo 1- 15.

Tabel 6. *Random Consistency Index* (Endah & Raharjo, 2003)

<i>n</i>	<i>RI</i>	<i>n</i>	<i>RI</i>	<i>n</i>	<i>RI</i>
1	0.00	6	1.24	11	1.51
2	0.00	7	1.32	12	1.48
3	0.58	8	1.41	13	1.56
4	0.9	9	1.45	14	1.57
5	1.12	10	1.49	15	1.59

Jika rasio  $CR \leq 0,1$  (10%), matriks dikatakan konsisten dan keputusan  $\vec{w}$  diterima. Sebaliknya, *CR* yang lebih dari itu menyiratkan terlalu banyak kontradiksi dalam matriks. Antisipasi untuk situasi terakhir adalah mengaji ulang matriks, lalu merevisi bobot-bobot yang dimuat oleh vektor  $\vec{w}$ .

## 2.5. *Technique for Order of Preference by Similarity to Ideal Solutions (TOPSIS)*

Ketenaran TOPSIS tak lepas dari konsep yang diangkatnya, yakni solusi optimal merupakan ia yang jaraknya paling dekat dengan solusi ideal positif sekaligus

paling jauh dari solusi ideal negatif. Konsep yang sederhana, namun amat jitu dalam memecahkan berbagai problem praktis. Pondasinya yang simpel tersebut kemudian menjadi akar bagi nilai plus TOPSIS lainnya, yaitu gampang dipahami dan komputasinya yang mudah lagi efisien. (Shalehah, 2014)

### 2.5.1. Teknis TOPSIS

a) Membuat matriks keputusan yang berisikan angka hasil konversi kompetensi tiap alternatif. Matriks ini berordo  $m$  baris dan  $n$  kolom.

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & \dots & x_{1n} \\ x_{21} & x_{22} & x_{23} & \dots & x_{2n} \\ x_{31} & x_{32} & x_{33} & \dots & x_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & x_{m3} & \dots & x_{mn} \end{bmatrix}$$

b) Menormalisasi matriks keputusan dengan persamaan berikut:

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{k=1}^m x_{kj}^2}} \text{ dimana } i = 1, 2, 3, \dots, m \\ \text{dan } j = 1, 2, 3, \dots, n$$

c) Mengalikan matriks ternormalisasi dengan bobot masing-masing kriteria yang diperoleh dari metode AHP ( $w_1, w_2, \dots, w_m$ ). Matriks yang dihasilkan selanjutnya disebut matriks ternormalisasi terbobot.

$$y_{ij} = w_j \times r_{ij} \text{ dimana } i = 1, 2, 3, \dots, m \\ \text{dan } j = 1, 2, 3, \dots, n$$

$$Y = [w_1 \quad w_2 \quad \dots \quad w_m] \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ r_{21} & r_{22} & \dots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{m1} & r_{m2} & \dots & r_{mn} \end{bmatrix}$$

$$= \begin{bmatrix} w_1 \times r_{11} & w_2 \times r_{12} & \dots & w_n \times r_{1n} \\ w_1 \times r_{21} & w_2 \times r_{22} & \dots & w_n \times r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_1 \times r_{m1} & w_2 \times r_{m2} & \dots & w_n \times r_{mn} \end{bmatrix}$$

d) Menentukan matriks solusi ideal positif  $A^+$  dan matriks solusi ideal negatif  $A^-$ .

Solusi ideal positif dan solusi ideal negatif, berturut-turut, didefinisikan sebagai sekumpulan nilai terbaik dan terburuk yang dapat dicapai untuk setiap kriteria.

Nilai-nilai yang dimaksud diambil dari data asli seluruh alternatif.

$$A^+ = (y_1^+, y_2^+, \dots, y_n^+)$$

$$A^- = (y_1^-, y_2^-, \dots, y_n^-)$$

dimana

$$y_j^+ = \begin{cases} \max(y_{ij}), & \text{jika } y_j \text{ adalah atribut keuntungan} \\ \min(y_{ij}), & \text{jika } y_j \text{ adalah atribut biaya} \end{cases}$$

$$y_j^- = \begin{cases} \min(y_{ij}), & \text{jika } y_j \text{ adalah atribut keuntungan} \\ \max(y_{ij}), & \text{jika } y_j \text{ adalah atribut biaya} \end{cases}$$

e) Menentukan jarak antara nilai tiap alternatif dengan solusi ideal positif dan solusi ideal negatif.

$$D_i^+ = \sqrt{\sum_{j=1}^n (y_{ij} - y_j^+)^2} \quad \text{dimana } i = 1, 2, 3, \dots, m$$

$$D_i^- = \sqrt{\sum_{j=1}^n (y_{ij} - y_j^-)^2} \quad \text{dimana } i = 1, 2, 3, \dots, m$$

f) Menghitung nilai preferensi setiap alternatif. Alternatif terpilih adalah ia yang nilai preferensinya tertinggi atau sekian besar tertinggi, tergantung selera pengambil keputusan.

$$V_i = \frac{D_i^-}{D_i^- + D_i^+} \quad \begin{array}{l} \text{dimana } 0 < V_i < 1 \\ \text{dan } i = 1, 2, 3, \dots, m \end{array}$$

(Shalehah, 2014; Wimatsari, Putra, & Buana, 2013)

### 2.5.2. Peran Perbandingan Matematis dalam Metode TOPSIS

Umumnya, angka-angka yang dimasukkan ke dalam matriks keputusan TOPSIS sudah berwujud hasil konversi dari data asli. Gejala ini diduga merupakan efek dari munculnya kriteria-kriteria yang bersifat kualitatif, misalnya prestasi dan riwayat organisasi. Supaya bisa diikutkan ke dalam matriks, kriteria yang demikian perlu dikuantifikasi dengan teknik penskalaan yang tepat. Agar adil, kriteria sisanya pun diskala memakai siasat yang sama.

Salah satu trik konversi yang dilakukan beberapa peneliti (Manurung, 2011; Y.I.E., Soebroto, & Regasari, 2013; Yusuf, Koniyo, & Novian, 2013) adalah menyematkan skor, yang dianggap cocok, pada tiap rentang nilai yang mungkin dari seluruh kriteria. Baik kadar skor maupun interval yang diwakilinya ini dipilih berdasarkan selera. Adapun contoh serta pemberdayaan skor-skor yang dimaksud, berturut-turut, ditampilkan oleh Tabel 7, Tabel 8, dan Tabel 9.

Tabel 7. Daftar Skor Kriteria (Manurung, 2011)

Kriteria	Data Awal	Data Konversi
IPK	2,75-3,00	1
	3,01-3,50	3
	3,51-4,00	5
Penghasilan	Di atas Rp 4.000.000	1
	Rp 2.100.000 – Rp 4.000.000	3
	Rp 0 – Rp 2.000.000	5
Jumlah Tanggungan	0-2 orang	1
	3-5 orang	3
	Di atas 5 orang	5
Semester	Di atas semester 6	1
	4-6	3
	0-3	5
Status Beasiswa	Mendapatkan lebih dari 4 beasiswa	1
	2-4	3
	0-2	5

Tabel 8. Data Asli Alternatif (Manurung, 2011)

NPM	Nama	IPK	Penghasilan	Jumlah tanggungan	Semester	Status beasiswa
061401078	Budi	3,2	Rp 1.000.000	2	9	2
061401089	Dewi	3,5	Rp 2.000.000	2	5	2
061401003	Hendri	2,75	Rp 1.500.000	4	5	1
061401091	Rani	3,9	Rp 3.000.000	1	2	0
061401077	Isti	2,1	Rp 1.000.000	3	9	2

Tabel 9. Nilai Alternatif setelah Dikonversi (Manurung, 2011)

NPM	Nama	IPK	Penghasilan	Jumlah tanggungan	Semester	Status beasiswa
061401078	Budi	3	5	1	1	3
061401089	Dewi	3	5	1	3	3
061401003	Hendri	1	5	3	3	5
061401091	Rani	5	3	1	5	5
061401077	Isti	1	5	3	1	3

Cara di atas digulirkan demi memenuhi asas keadilan, namun ternyata belum bisa dikatakan begitu. Perjuangan seorang mahasiswa mencapai IPK 3,00 boleh jadi lebih keras dibanding temannya yang mendulang 3,80; sayangnya, ia dihargai sama dengan mereka yang digit transkripnya tidak menyentuh kepala 3. Kerja para buruh yang berpenghasilan 2,1 juta/bulan jelas lebih kasar dan berbahaya dibanding pegawai kantoran yang diupahi 3 juta/bulan; akan tetapi keduanya dipatok dengan ponten yang setara. Pengeluaran keluarga ber-anak 5 dianggap seimbang dengan famili yang tanggungannya hanya 3 jiwa. Amat kentara bahwa komposisi semacam ini jauh dari kata *setimpal*.

Oleh karena itu, digunakan kiat lain yang lebih menjanjikan, yakni perbandingan matematis. Kaidah ini menjamin tak ada dua atau lebih bilangan berbeda yang

berujung pada poin serupa. Masukan yang dibutuhkan hanya nilai maksimum setiap kriteria serta skor terkait, yang otomatis menjadi skor tertinggi. Dua angka barisan nantinya diposisikan sebagai pembagi dan pengali nilai-nilai sisanya, demi menurunkan skor untuk nilai-nilai tersebut. Metode ini diilustrasikan pada Tabel 10, sekaligus dibandingkan dengan teknik dari Tabel 7.

Tabel 10. Komparasi Skor IPK dari 2 Metode Konversi

IPK	Metode 1: Tabel 7	Metode 2: Perbandingan	
	Hasil Konversi	Hasil Konversi	
4,00	5		5,0000
3,99	5	$3,99/4 \times 5$	4,9875
3,98	5	$3,98/4 \times 5$	4,9750
3,75	5	$3,75/4 \times 5$	4,6875
3,55	5	$3,55/4 \times 5$	4,4375
3,51	5	$3,51/4 \times 5$	4,3875
3,50	3	$3,50/4 \times 5$	4,3750
3,25	3	$3,25/4 \times 5$	4,0625
3,01	3	$3,01/4 \times 5$	3,7625
3,00	1	$3,00/4 \times 5$	3,7500
2,99	1	$2,99/4 \times 5$	3,7375
2,75	1	$2,75/4 \times 5$	3,4375
2,01	0	$2,01/4 \times 5$	2,5125
2,00	0	$2,00/4 \times 5$	2,5000
1,90	0	$1,90/4 \times 5$	2,3750
1,55	0	$1,55/4 \times 5$	1,9375

Digit-digit di sisi kanan Tabel 10 diperoleh melalui evaluasi menggunakan Tabel 7, sementara yang di sebelah kiri diraih lewat proses perbandingan. Dengan sekilas pandang, dapat disimpulkan bahwa metode perbandingan matematis menyuguhkan hasil yang amat proporsional.

### III. METODE PENELITIAN

#### 3.1. Tempat dan Waktu Penelitian

Penelitian ini diinisiasi sejak pertengahan tahun 2017, di Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA) Universitas Lampung.

#### 3.2. Perangkat Pendukung Pengembangan Sistem

Perangkat keras yang digunakan selama pembuatan sistem adalah *notebook* Acer Aspire E1-470 dengan spesifikasi:

1. Prosesor Intel® Core™ i3-3217U, yang terdiri dari dua inti pemrosesan (*dual core*) berfrekuensi 1.80 GHz, *cache* L1 32 KB, *cache* L2 256 KB, dan *cache* L3 3MB.
2. *Random Access Memory* (RAM) Corsair 8192 MB DDR3L SODIMM PC3-12800 (1600 MHz).
3. *Hard Disk Drive* (HDD) Hitachi HTS54755 500 GB.

Perangkat lunak yang dimanfaatkan dalam penelitian antara lain:

1. Linux Mint Debian Edition (LMDE) 2 “Betsy” 64-bit.
2. Odo Community 10 beserta seluruh dependensinya.
3. Microsoft Windows 7 Ultimate Edition 64-bit SP1.
4. *Browser* Mozilla Firefox dan Google Chrome.
5. Python 2.7.9.

- |                     |                           |
|---------------------|---------------------------|
| 6. Postgre SQL 9.3. | 8. Atom 1.22.0.           |
| 7. PhpPgAdmin 5.1.  | 9. StarUML 2.8.0.         |
| 8. SchemaSpy 5.0.0. | 10. Microsoft Visio 2013. |

### 3.3. Metode Penelitian

Gambar 5 menyajikan tahapan-tahapan apa saja yang dilakukan selama penelitian berlangsung, yang mana diawali dengan observasi, wawancara, dan studi literatur, kemudian diteruskan dengan siklus hidup pengembangan sistem (*system development life cycle*), lalu akhirnya ditutup dengan menarik kesimpulan serta publikasi laporan.



Gambar 5. Diagram Alir Penelitian

#### 3.3.1. Observasi dan wawancara

Tujuan yang hendak dicapai dari tahap ini adalah mendeteksi masalah yang sedang dihadapi oleh mahasiswa FMIPA Universitas Lampung. Masalah inilah yang nantinya menjadi pendorong digelarnya suatu penelitian.

Observasi sebenarnya telah dijalankan secara “tak sengaja” sejak tahun 2011, dimana kala itu tak sedikit keluhan yang terdengar dari rekan-rekan mahasiswa perihal kurang adilnya hasil seleksi beasiswa-beasiswa di FMIPA. Beberapa dari

mereka bahkan berputus asa di awal, sehingga enggan mencoba peruntungan pada periode berikutnya. Hingga tiba masanya penelitian ini dirintis, tepatnya tahun 2017, problem yang sama belum terpecahkan. Memang ada sejumlah solusi yang dicetuskan, namun semuanya berpusat pada satu jenis beasiswa saja. Memanajementi struktur yang demikian tentu sulit, mengingat berlimpahnya total beasiswa yang berlaku di FMIPA.

Wawancara diadakan di Gedung Dekanat FMIPA untuk memahami tata cara pemilihan penerima beasiswa. Melalui kegiatan yang ditujukan hanya pada anggota tim seleksi ini, dijumpai persoalan lainnya, yakni pengetikan manual data pelamar. Tugas seperti ini cukup merepotkan dan harus ditunaikan berulang-ulang, sehingga beliau mengharapkan hadirnya sebuah mekanisme dimana pengisian data langsung ditangani oleh mahasiswa.

Kalkulasi nilai akhir pelamar diawali dengan mengonversi data kriterianya ke skor yang berpadanan, seperti yang ditunjukkan di halaman 182, 183, dan 195. Skor-skor tersebut lantas dikalikan dengan bobot masing-masing kriteria kemudian dijumlahkan. Berdasarkan hasil penjumlahan ini, para pelamar diurutkan menurun per jurusan. Selama kuota jurusannya belum terpenuhi, mahasiswa-mahasiswa yang berada di peringkat atas terpilih sebagai penerima beasiswa; sebaliknya, sisanya akan dieliminasi.

Selepas wawancara, segala aktivitas yang terjadi semenjak pendaftaran beasiswa dibuka sampai hasil seleksi diumumkan plus pihak-pihak yang terlibat

dideskripsikan oleh Gambar 2. Skema dalam gambar disebut juga *proses bisnis*, yang menjadi panduan bagi *developer* untuk membangun sistem yang relevan dengan kondisi di lapangan. Di samping diagram alir, format proses bisnis lainnya ialah *activity diagram* dan *sequence diagram* (Hamilton & Miles, 2006).

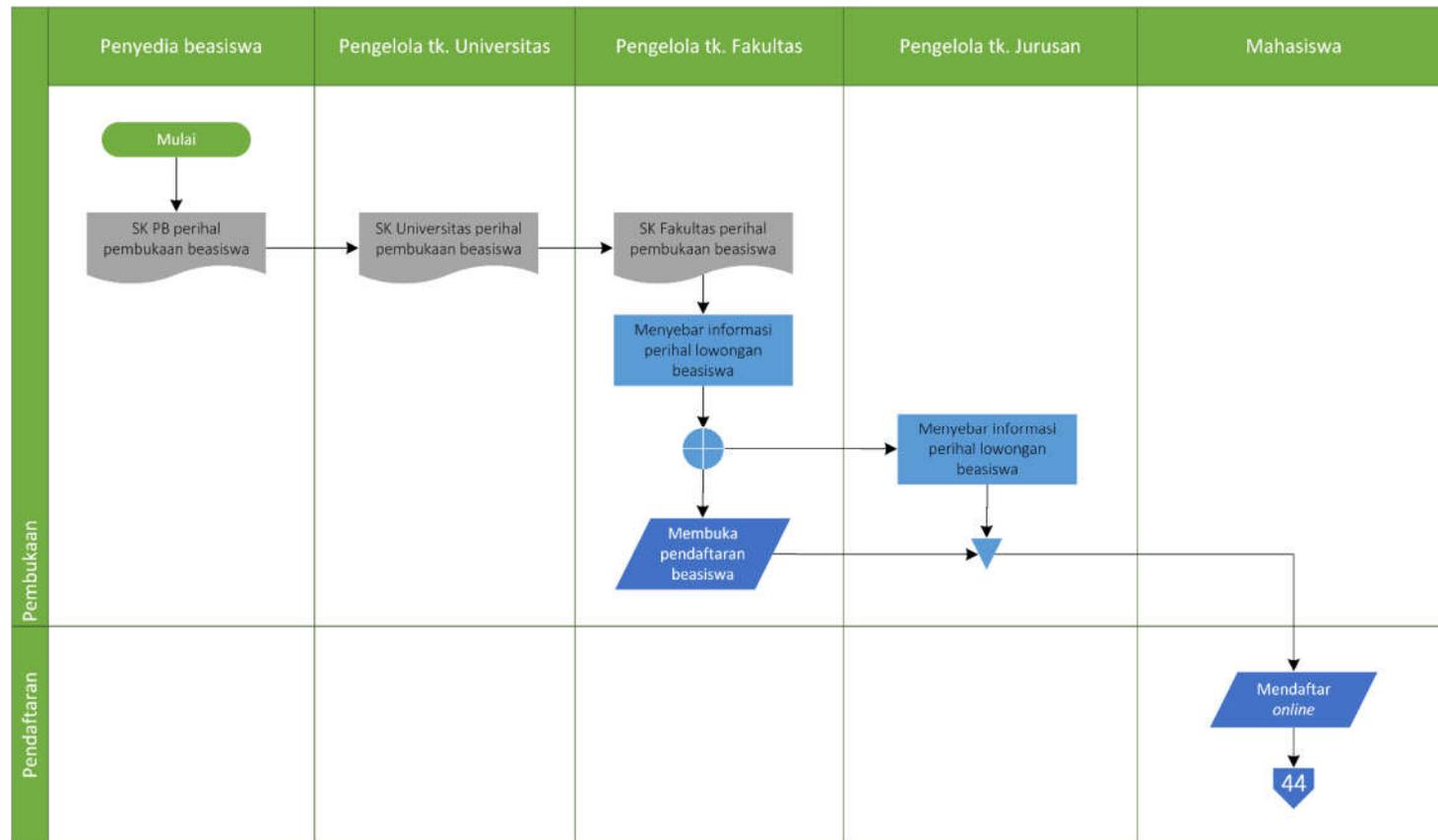
### 3.3.2. Studi literatur

Sasaran dari tahap ini adalah memperoleh wawasan tentang riset-riset sebelumnya yang sehaluan dengan topik penelitian. Sumber-sumber yang bisa dirujuk amat bervariasi, mulai dari jurnal, buku, karya ilmiah, artikel, hingga video tutorial. Peneliti sendiri memanfaatkan studi literatur guna mempelajari konsep matematis serta praktik AHP dan TOPSIS.

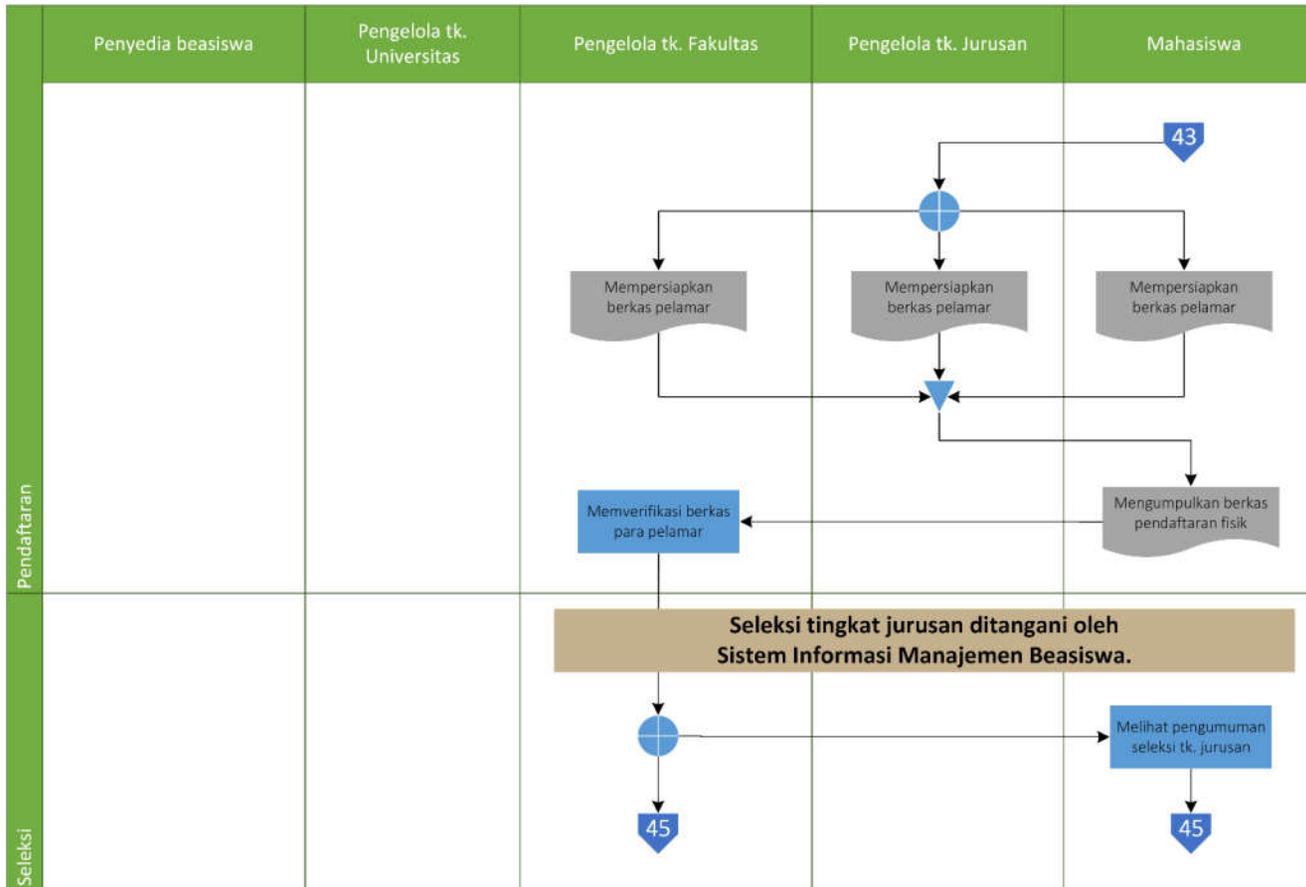
### 3.3.3. Perancangan sistem

Sistem Manajemen Beasiswa FMIPA Universitas Lampung jatuh pada kategori DSS, mengingat keluarannya berupa keputusan tentang siapa-siapa saja yang berhak menerima beasiswa. *Model-base* sistem diisi oleh algoritma AHP dan TOPSIS. Platform-nya berbasis *web* dan dapat dijangkau via internet, sehingga pihak-pihak yang berkepentingan bisa berinteraksi dengan sistem dimanapun dan kapanpun.

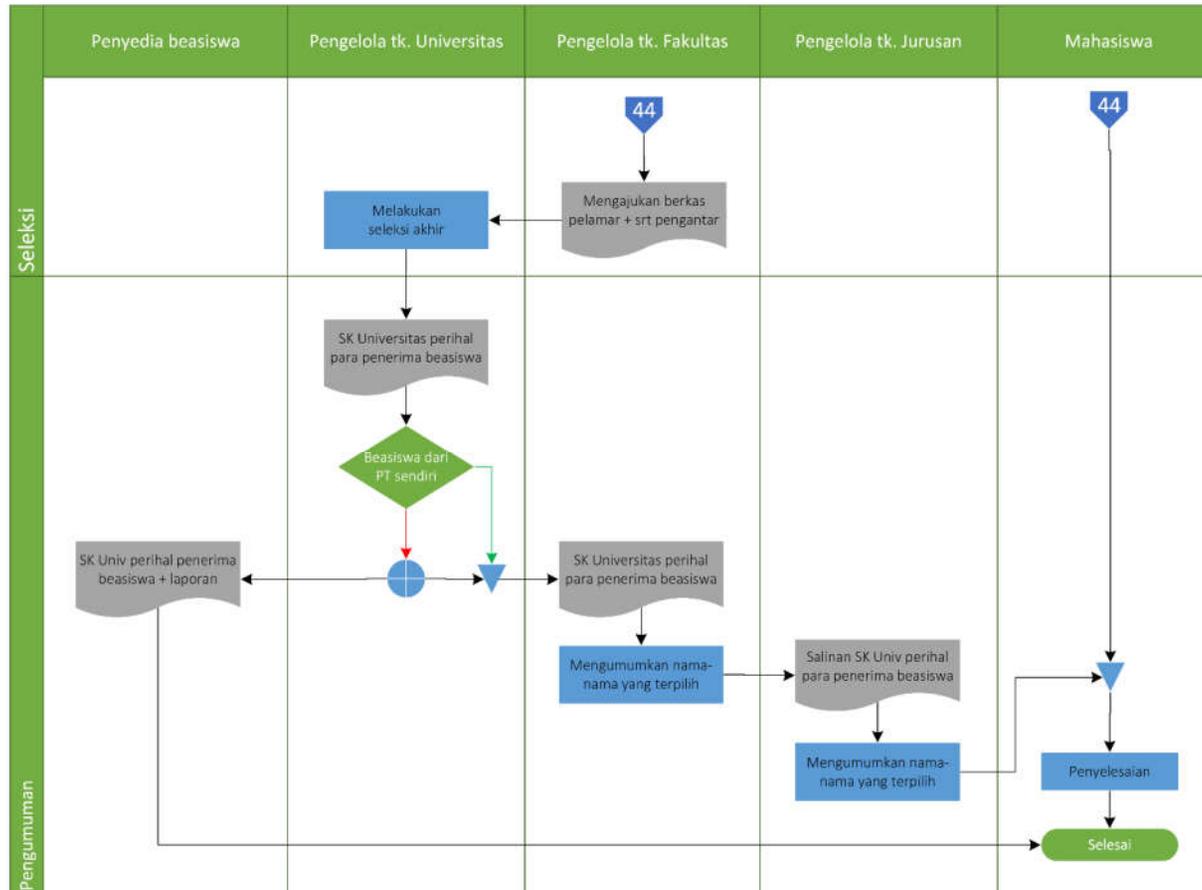
Sesudah sistem diimplementasikan, tata kelola beasiswa yang dideskripsikan di Gambar 2 diharapkan berganti, mengikuti apa yang disuguhkan oleh Gambar 6. Andil pihak Jurusan akan berkurang drastis hingga tertinggal satu peran saja, yaitu diseminasi info-info beasiswa serta persiapan berkas pelamar. Kewajiban verifikasi berkas kelak diambil sepenuhnya oleh Fakultas.



Gambar 6. Diagram Alir Pengelolaan Beasiswa Setelah Implementasi Sistem.



Gambar 6. (lanjutan)



Gambar 6. (lanjutan)

Dari Gambar 6, dapat ditarik dua jenis pengguna sistem, yakni 1) **mahasiswa** selaku pelamar dan 2) pengelola tingkat fakultas yang selanjutnya dipanggil dengan istilah **tim seleksi** atau **tim penyeleksi**. **Administrator** juga dibutuhkan demi menangani berbagai prosedur manajemen *user*. Akibat berada di luar cakupan sistem, tak ditemui lagi dalam *flowchart* alur untuk beasiswa yang penyelenggaraannya murni dipegang oleh penyedia beasiswa.

Beasiswa-beasiswa yang dikelola oleh tim penyeleksi FMIPA mensyaratkan kriteria yang berbeda-beda. Namun, akibat keterbatasan-keterbatasan yang muncul, diasumsikan semua beasiswa mengenakan kriteria-kriteria yang sama, yaitu *IPK*, *Semester*, *Pendapatan* dan *Jumlah Tanggungan Orang tua/Wali*, serta *Prestasi*. Keterangan selengkapnya perihal kriteria ditunjukkan oleh Tabel 11.

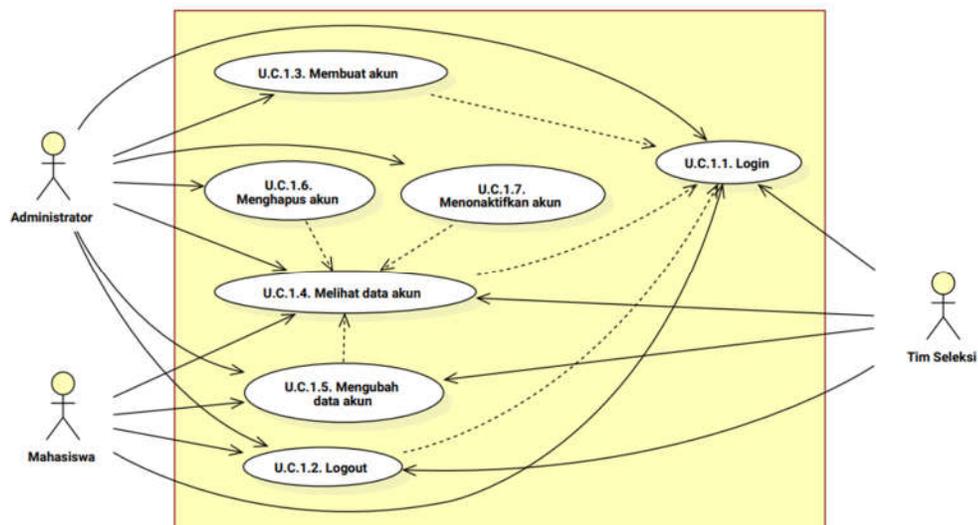
Tabel 11. Daftar Kriteria dalam Sistem Informasi Manajemen Beasiswa FMIPA

<b>Kriteria</b>	<b>Keterangan</b>
<i>IPK</i>	Semakin tinggi IPK dan semesternya, semakin tinggi nilai akhir si pelamar. Batas minimum ditetapkan khusus untuk kedua kriteria ini, karena beberapa beasiswa mengharuskan para pelamarnya memiliki IPK minimal 3,00 atau berada di semester 3 ke atas. Pada kriteria semester juga dipatok batas maksimum, karena umumnya beasiswa tidak lagi diberikan kepada para pelamar yang masa studinya sudah melewati 8 semester.
<i>Semester</i>	
<i>Pendapatan Ortu/Wali</i>	Semakin tinggi pendapatan keluarga, maka semakin rendah nilai akhir pelamar.
<i>Jumlah Tanggungan Ortu/Wali</i>	Serupa dengan IPK dan semester, semakin tinggi jumlah tanggungan orang tua, semakin tinggi pula nilai akhir pelamar.
<i>Prestasi</i>	Semakin tinggi prestasi pelamar, tentu semakin tinggi nilai akhirnya. Nominal kriteria ini perlu diperinci melalui parameter-parameter berikut: <ol style="list-style-type: none"> <li>1. Ruang lingkup: internasional, nasional, regional (gabungan beberapa provinsi), provinsi, kabupaten/kota.</li> <li>2. Titel: juara 1, juara 2, juara 3, juara harapan 1, juara harapan 2, juara harapan 3.</li> </ol>

Seluruh proses pada Gambar 6 diterjemahkan sebagai fungsi dalam sistem komputer lewat 37 buah *use case* yang dikelompokkan dalam 10 blok: **Umum**, **Periode**, **Master Beasiswa**, **Mahasiswa**, **Pegawai**, **Validator**, **Kriteria**, **Beasiswa**, **Pendaftaran**, dan **Seleksi**. Spesifikasi per *use case* diuraikan dalam *activity diagram* dan *sequence diagram*. *Activity diagram*<sup>1</sup> memfokuskan perhatian pada aktivitas-aktivitas yang menyusun sebuah fungsi, sedangkan *sequence diagram* menekankan pada kronologi aktivitas-aktivitas tersebut.

### 3.3.3.1. Umum

Gambar 7 menampilkan ragam *use case* yang termasuk dalam blok **Umum**, di antaranya **login**, **logout**, **membuat akun pengguna (user)**, **melihat data akun**, **mengubah data akun**, **menghapus akun**, dan **menonaktifkan akun**.

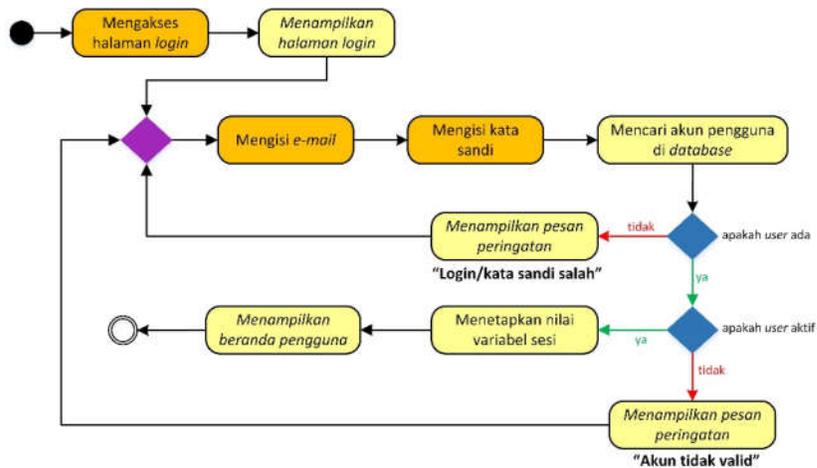


Gambar 7. Ragam *Use-Case* di Blok **Umum**.

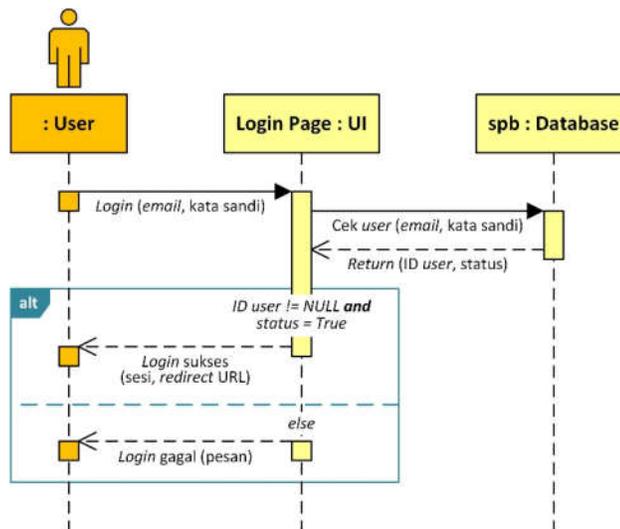
<sup>1</sup> Keterangan untuk *activity diagram*: Proses yang berbalok jingga diselenggarakan oleh *user*, sedangkan yang berbalok kuning oleh sistem.

a. *Login dan logout*

Segala operasi yang ditawarkan oleh sistem dapat dilakukan setelah pengguna berhasil *login*. *Activity* dan *sequence diagram* untuk fungsi *login* ditunjukkan oleh Gambar 8 dan Gambar 9, sedangkan *activity* dan *sequence diagram* untuk fungsi *logout* ditunjukkan oleh Gambar 10 dan Gambar 11.



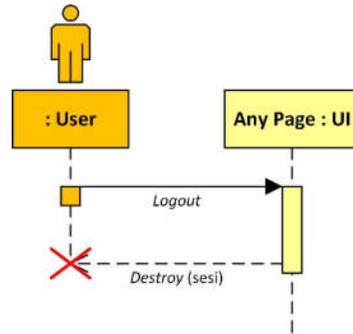
Gambar 8. *Activity Diagram Fungsi Login.*



Gambar 9. *Sequence Diagram Fungsi Login.*



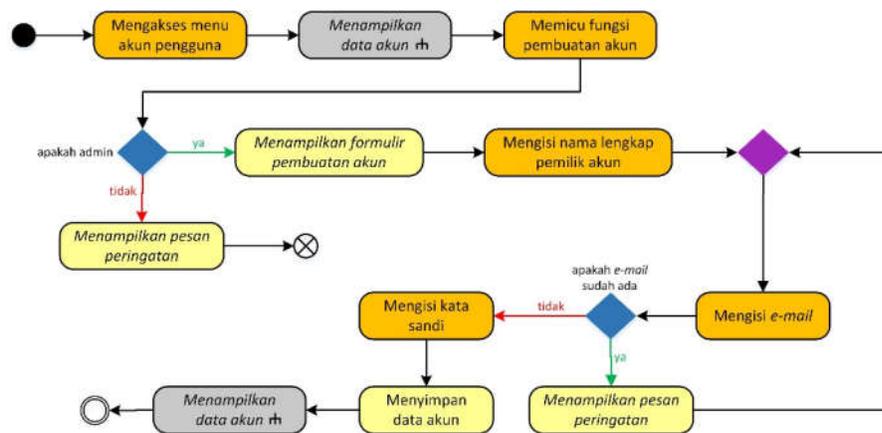
Gambar 10. *Activity Diagram Fungsi Logout.*



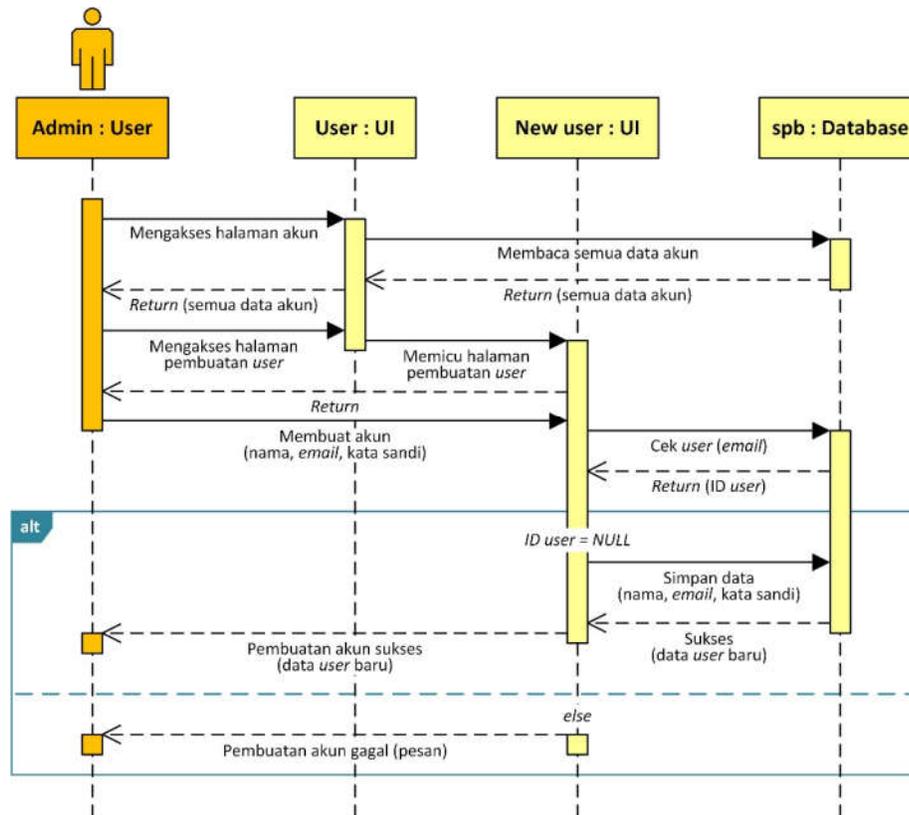
Gambar 11. *Sequence Diagram Fungsi Logout.*

### b. Membuat akun

Demi menghindari tumbuhnya akun-akun ganda, fungsi ini sepenuhnya berada di tangan administrator. Alasan yang sama menjadi dasar mengapa tak ada menu *Register* atau *Sign Up* pada sistem. Gambar 12 dan Gambar 13 memperlihatkan garis besar fungsi pembuatan akun.



Gambar 12. *Activity Diagram Fungsi Pembuatan Akun.*



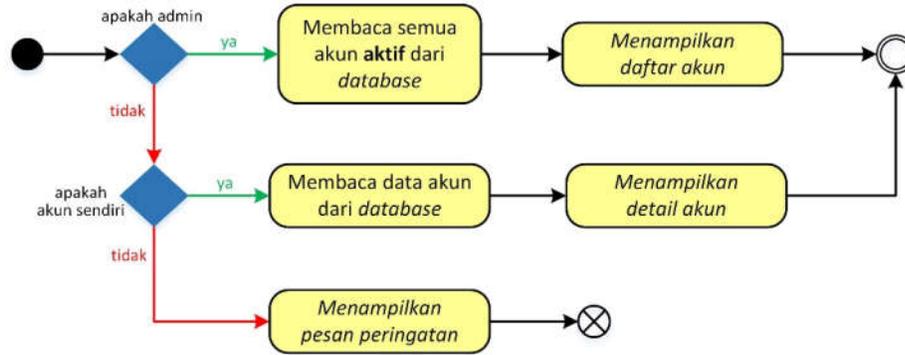
Gambar 13. *Sequence Diagram* Fungsi Pembuatan Akun.

Pembuatan akun gagal jika ditemui dua pengguna dengan nama surel atau nama *user* yang sama, karena dua atribut ini merupakan pembeda antara satu pengguna dengan pengguna lainnya.

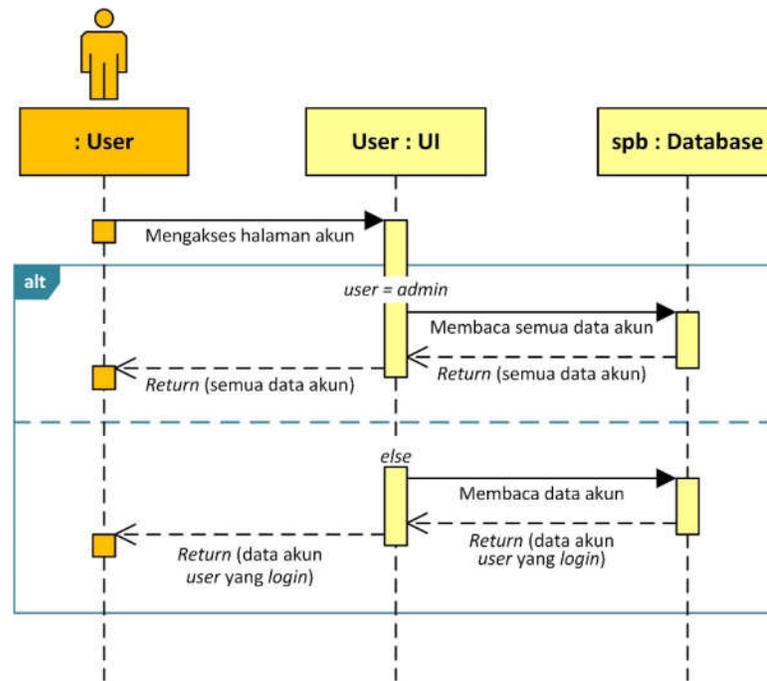
### c. *Melihat data akun*

Ketiga jenis aktor sanggup mengeksekusi fungsi ini, namun ada variasi terkait data yang boleh diakses. Bila pengguna adalah administrator, maka ia dapat melihat detail seluruh akun. Pengguna selain itu dicukupkan sampai akun miliknya saja.

Gambar 14 dan Gambar 15 mendeskripsikan fungsi pembacaan data akun secara singkat.



Gambar 14. *Activity Diagram* Fungsi Pembacaan Data Akun.

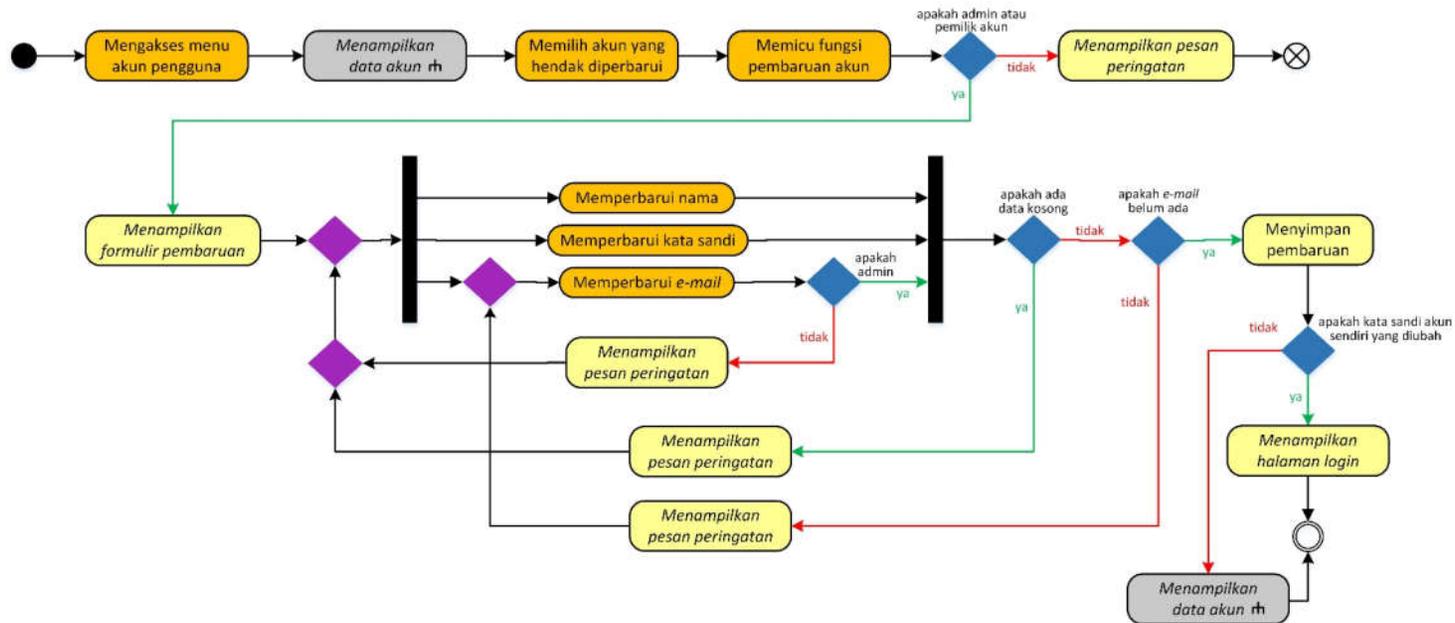


Gambar 15. *Sequence Diagram* Fungsi Pembacaan Data Akun.

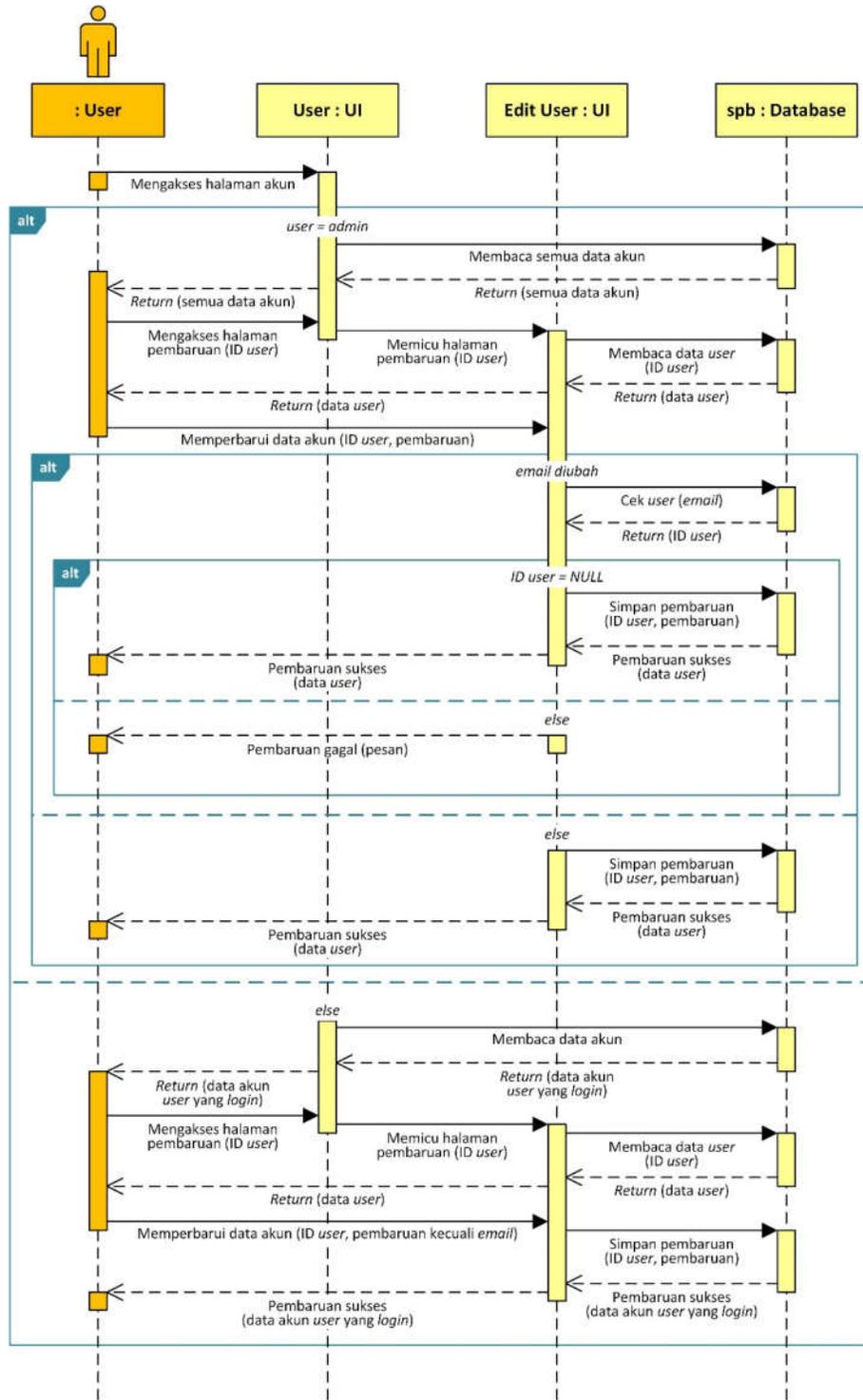
#### d. *Mengubah data akun*

Serupa dengan *use case* sebelumnya, semua aktor sistem dapat menjalankan fungsi ini, namun ada variasi terkait data yang boleh diubah. Bila pengguna adalah administrator, maka ia dapat mengubah detail seluruh akun, termasuk surel atau

*username*-nya, selama nilai kedua atribut tersebut tetap unik di dalam *database*. Pengguna selain administrator hanya diizinkan mengubah nama lengkap dan kata sandi miliknya. Gambar 16 dan Gambar 17 menampilkan *activity diagram* dan *sequence diagram* untuk fungsi perubahan data akun.



Gambar 16. *Activity Diagram* Fungsi Perubahan Data Akun.

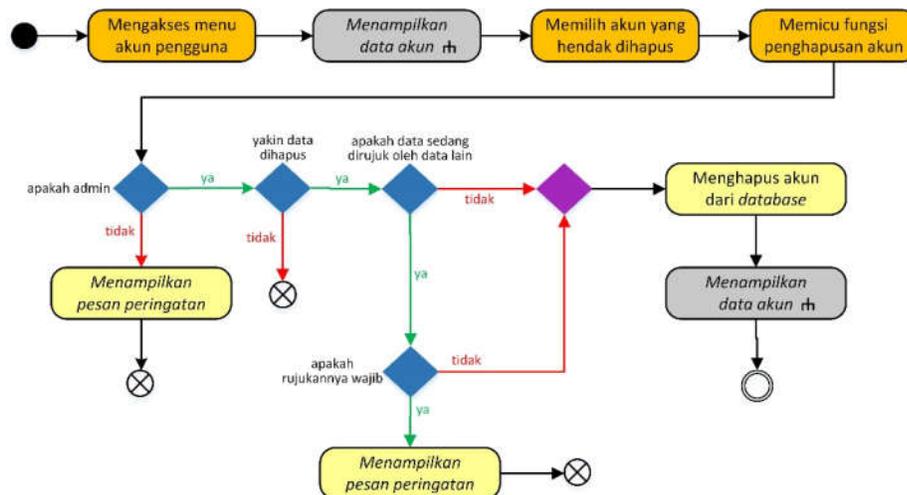


Gambar 17. Sequence Diagram Fungsi Perubahan Data Akun.

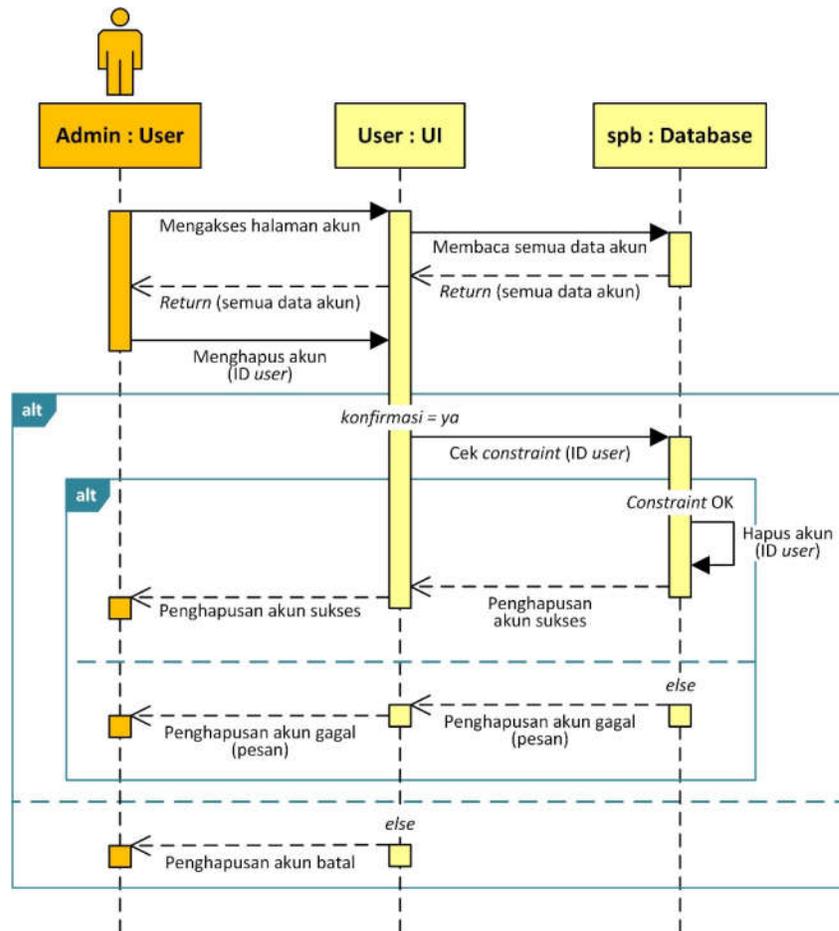
e. *Menghapus akun*

Fungsi kali ini dikhususkan untuk administrator. Kalau akun administrator satu-satunya hendak dihapus, maka akun admin pengganti harus tercatat dahulu di sistem sebelum akun yang lama dihilangkan. Seandainya, suatu akun tetap bertahan akibat satu dan lain hal, misalnya karena sedang dirujuk oleh suatu kolom yang tidak boleh dibiarkan **NULL**, alternatifnya adalah melaksanakan *use case* menonaktifkan akun.

Gambar 18 dan Gambar 19 memuat garis besar fungsi penghapusan akun.



Gambar 18. *Activity Diagram* Fungsi Penghapusan Akun.

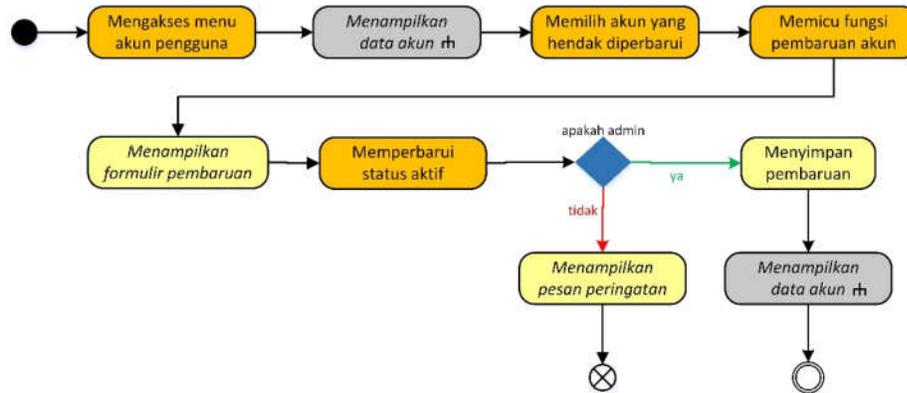


Gambar 19. *Sequence Diagram* Fungsi Penghapusan Akun.

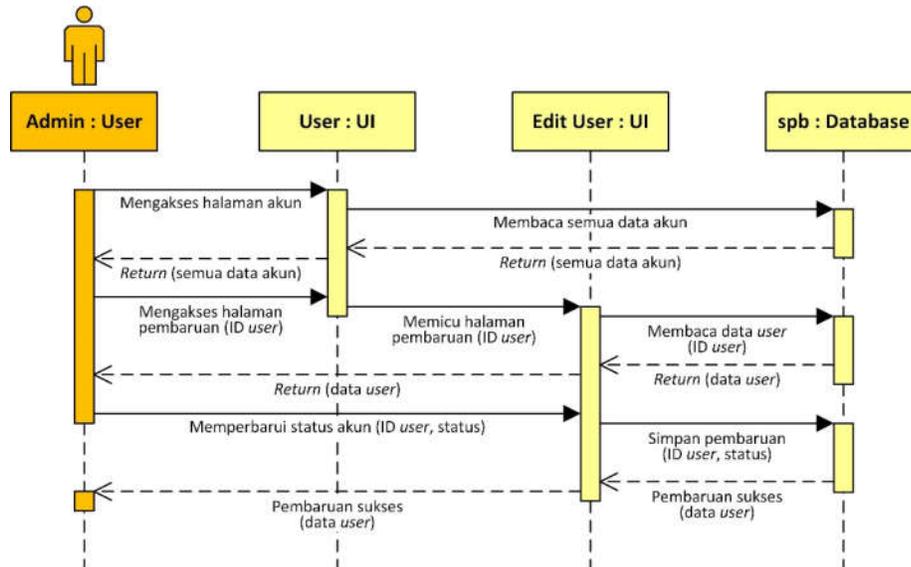
#### f. *Menonaktifkan akun*

*Use case* ini diaplikasikan oleh administrator ketika pemilik akun tidak berurusan dengan Fakultas MIPA Universitas Lampung secara sementara, misalnya sedang menjalani cuti. Pengguna yang non-aktif tidak akan bisa *login* ke dalam sistem walaupun kombinasi surel/nama *user* dan kata sandi yang ia masukkan benar.

Gambar 20 dan Gambar 21 mempresentasikan fungsi penonaktifan akun.



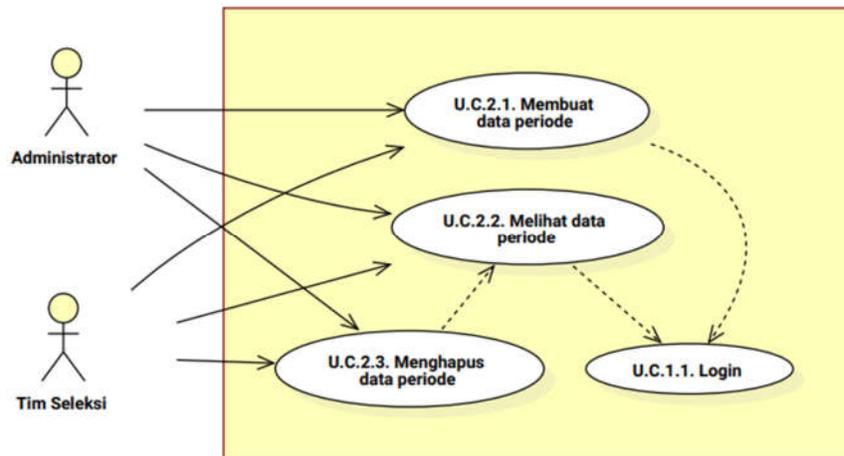
Gambar 20. *Activity Diagram* Fungsi Penonaktifan Akun.



Gambar 21. *Sequence Diagram* Fungsi Penonaktifan Akun.

### 3.3.3.2. Periode

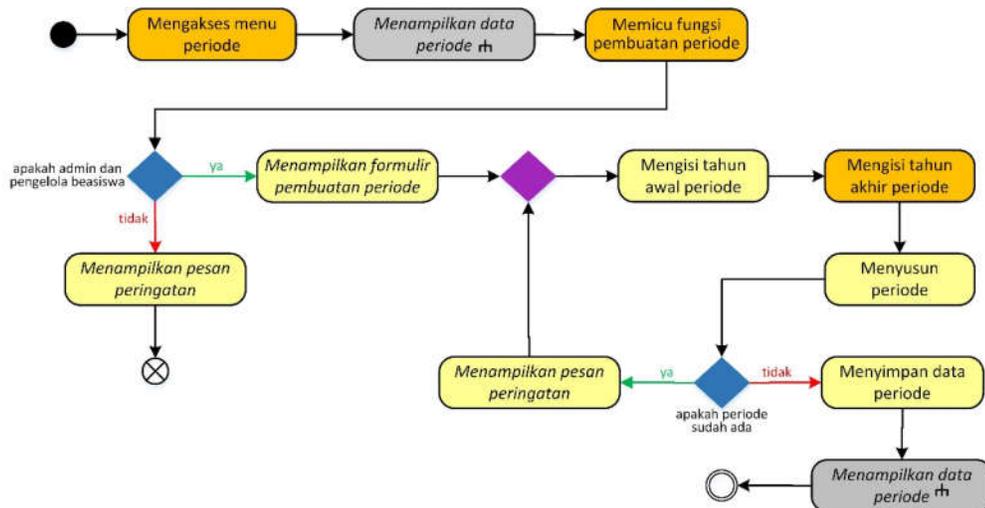
Gambar 22 menampilkan ragam *use case* yang termasuk dalam blok **Periode**, di antaranya **membuat data periode**, **melihat data periode**, dan **menghapus periode**. Administrator dan tim seleksi mampu melakukan semua fungsi di blok ini.



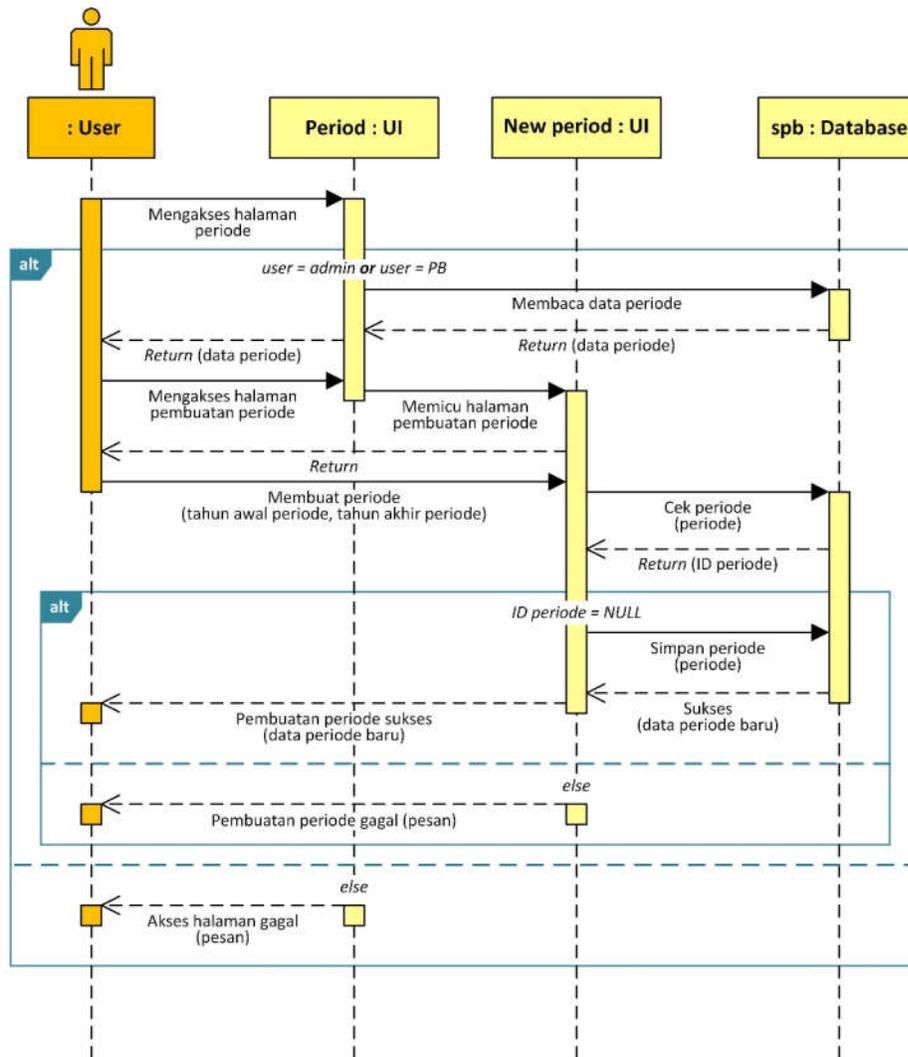
Gambar 22. Ragam *Use-Case* di Blok **Periode**.

a. *Membuat data periode*

*Activity* dan *sequence diagram* untuk fungsi pembuatan data periode ditunjukkan oleh Gambar 23 dan Gambar 24. Pembuatan periode tidak akan sukses jika ada dua periode yang cakupannya sama.



Gambar 23. *Activity Diagram* Fungsi Pembuatan Data Periode.

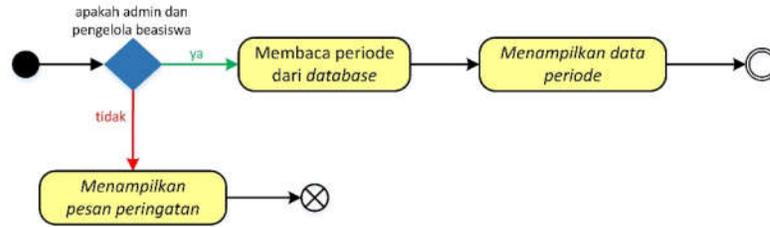


Gambar 24. *Sequence Diagram* Fungsi Pembuatan Data Periode.

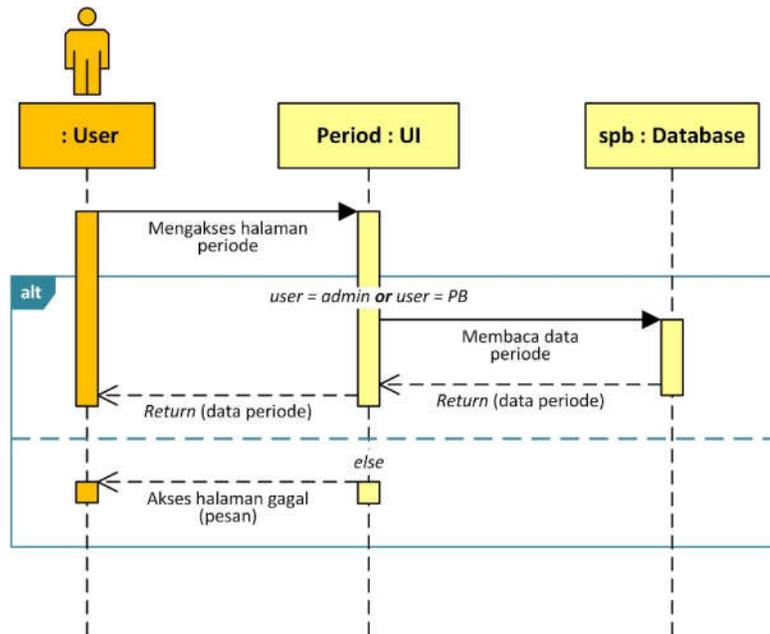
Periode bersama master beasiswa selaku ciri-ciri suatu beasiswa harus eksklusif. Keduanya membedakan satu beasiswa dengan beasiswa lainnya.

**b. *Melihat data periode***

Gambar 25 dan Gambar 26 mendeskripsikan fungsi pembacaan data periode.



Gambar 25. *Activity Diagram* Fungsi Pembacaan Data Periode.

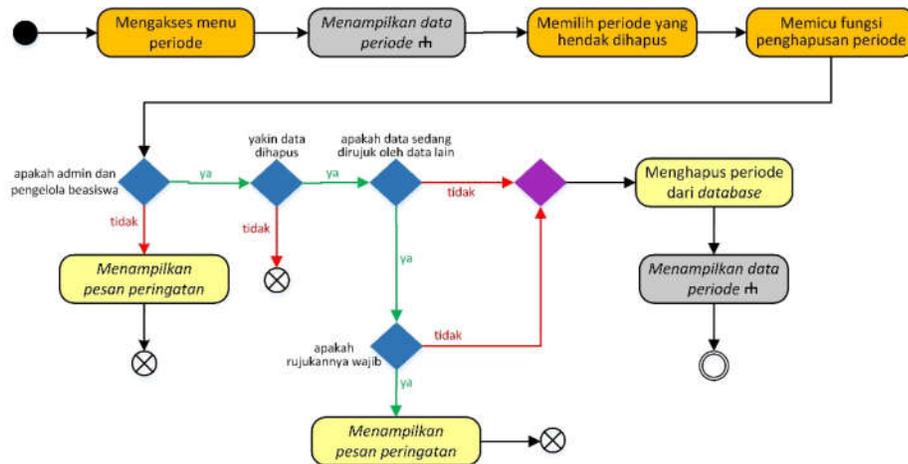


Gambar 26. *Sequence Diagram* Fungsi Pembacaan Data Periode.

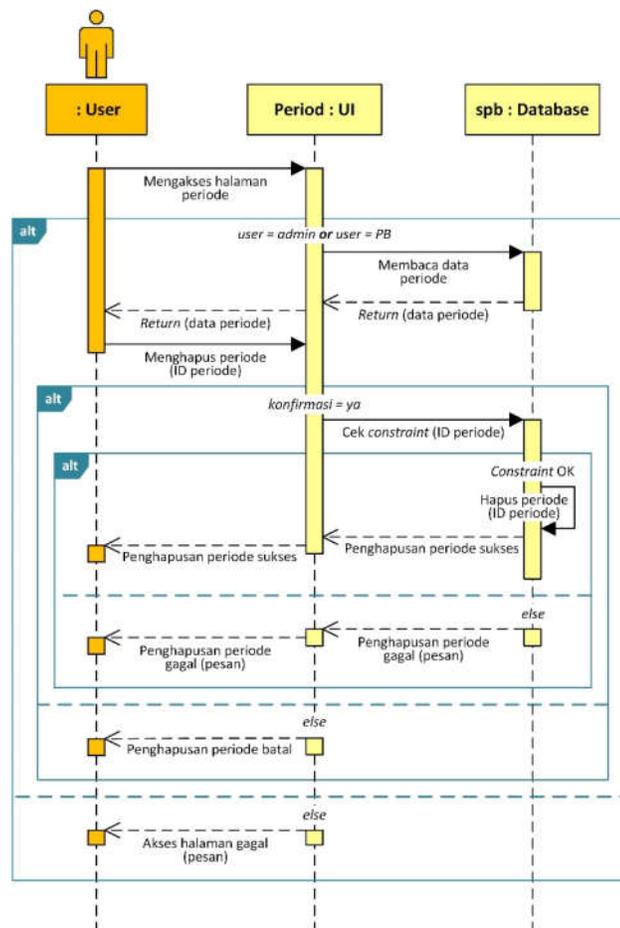
**c. Menghapus periode**

Suatu periode dapat dihapus bila belum ada satu pun beasiswa yang memakainya.

Gambar 27 dan Gambar 28 menampilkan *activity diagram* dan *sequence diagram* untuk fungsi penghapusan data periode.



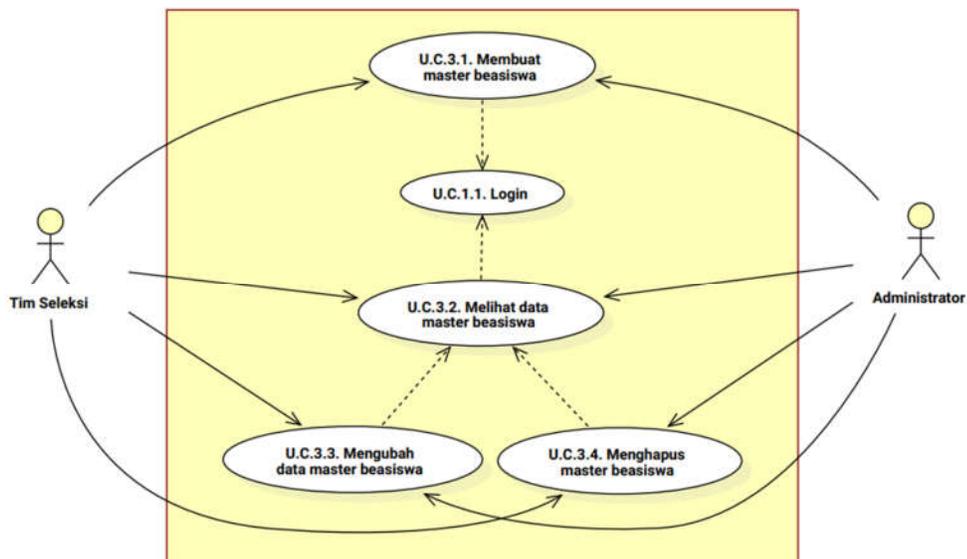
Gambar 27. Activity Diagram Fungsi Penghapusan Periode.



Gambar 28. Sequence Diagram Fungsi Penghapusan Periode.

### 3.3.3.3. Master Beasiswa

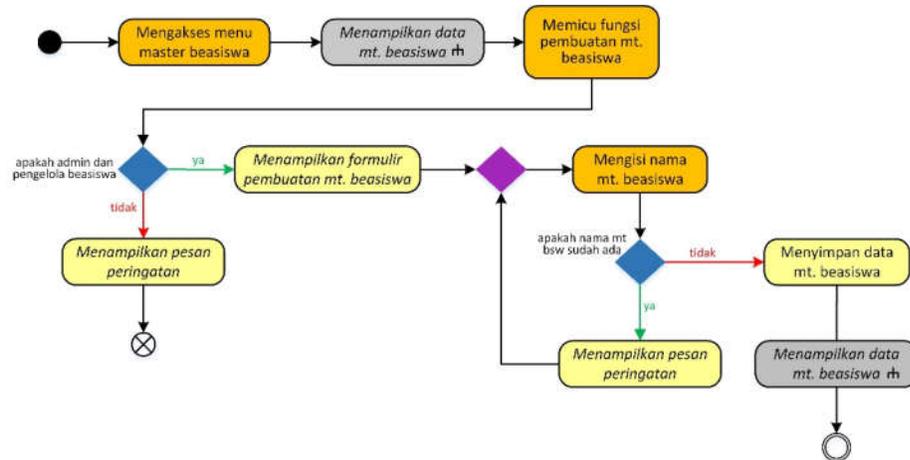
Gambar 29 menampilkan ragam *use case* yang termasuk dalam blok **Master Beasiswa**, di antaranya **membuat master beasiswa**, **melihat data master beasiswa**, **mengubah data master beasiswa**, dan **menghapus master beasiswa**. Segenap fungsi di blok ini hanya dapat dilakukan oleh administrator dan tim seleksi.



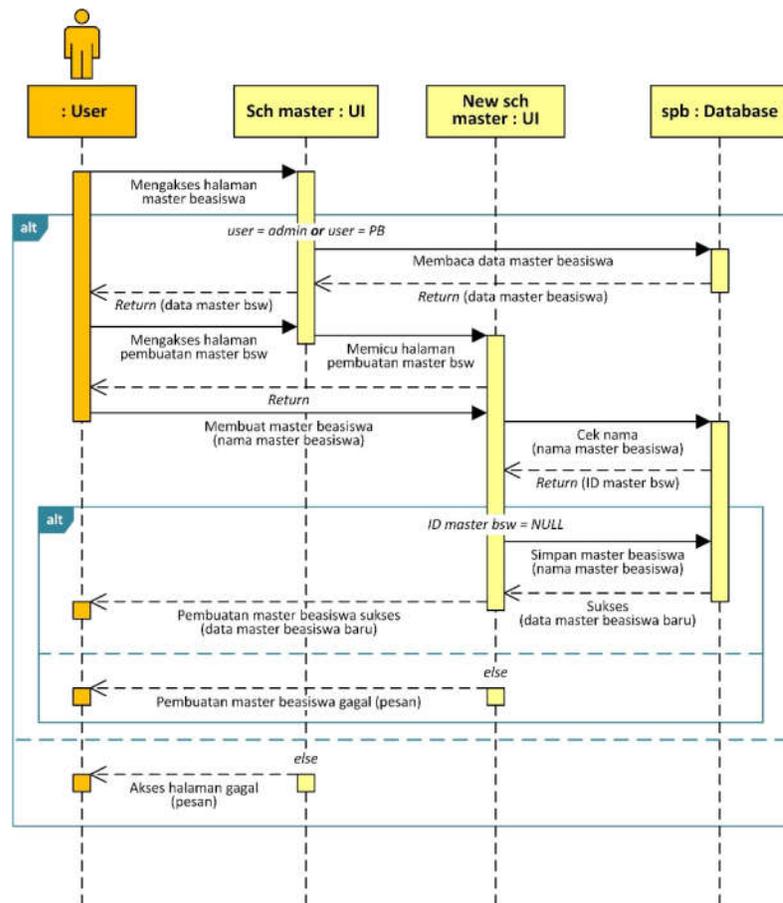
Gambar 29. Ragam *Use-Case* di Blok **Master Beasiswa**.

#### a. *Membuat master beasiswa*

*Activity* dan *sequence diagram* untuk fungsi pembuatan master beasiswa ditunjukkan oleh Gambar 30 dan Gambar 31. Pembuatan master beasiswa akan gagal jika terwujud dua master dengan nama serupa.



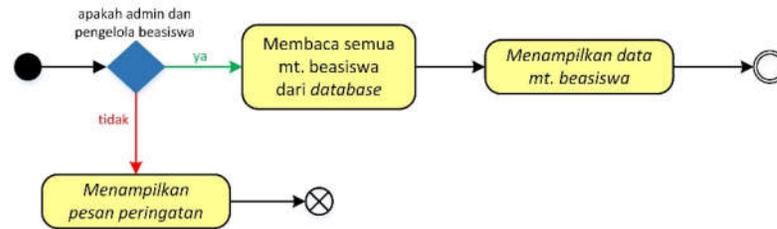
Gambar 30. Activity Diagram Fungsi Pembuatan Master Beasiswa.



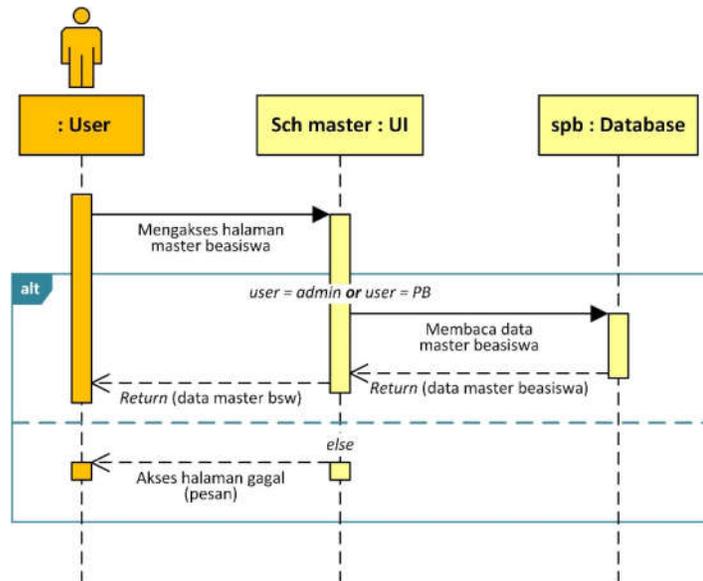
Gambar 31. Sequence Diagram Fungsi Pembuatan Master Beasiswa.

**b. Melihat data master beasiswa**

Gambar 32 dan Gambar 33 mendeskripsikan fungsi pembacaan master beasiswa.



Gambar 32. *Activity Diagram* Fungsi Pembacaan Data Master Beasiswa.

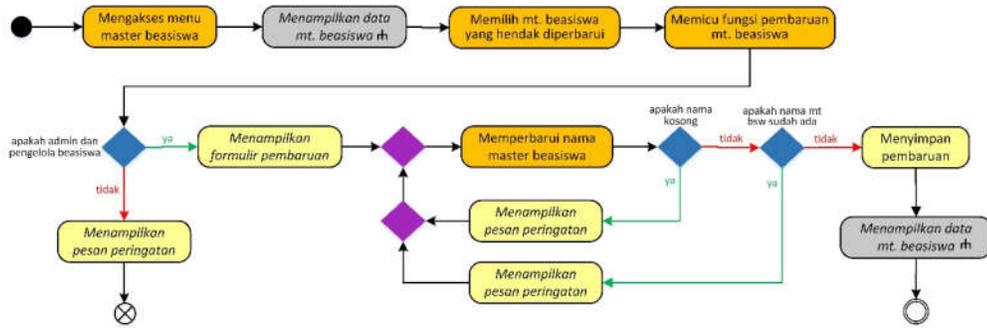


Gambar 33. *Sequence Diagram* Fungsi Pembacaan Data Master Beasiswa.

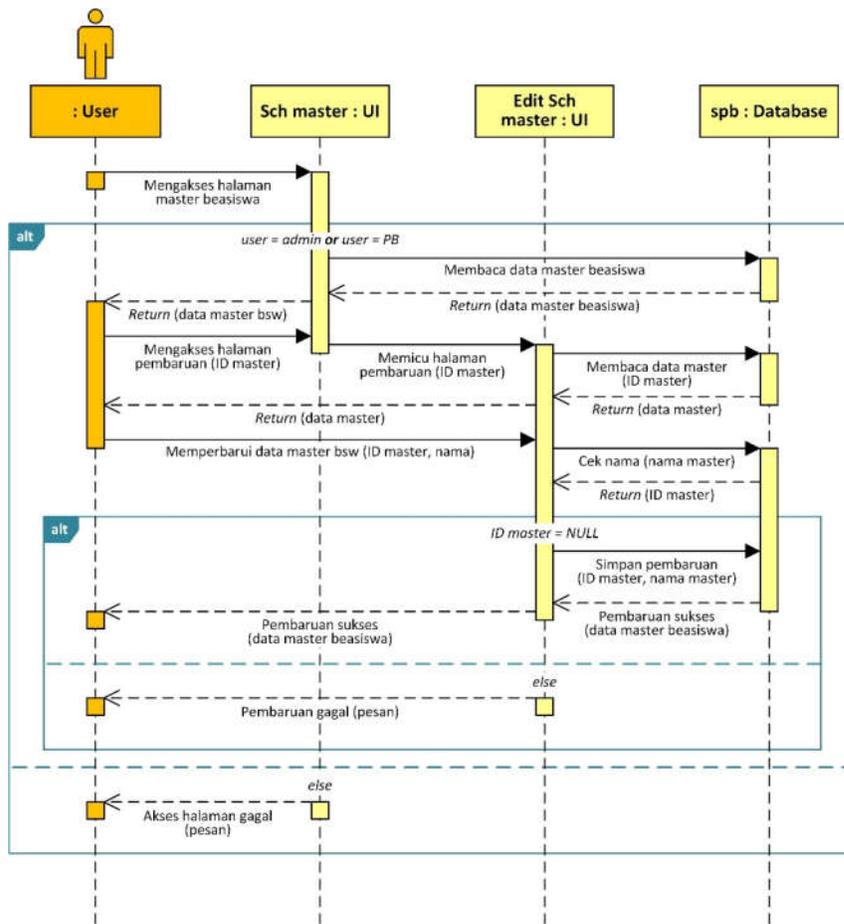
**c. Mengubah data master beasiswa**

Nama master beasiswa bisa diubah asalkan nilainya tetap khas di dalam *database*.

Gambar 34 dan Gambar 35 menampilkan *activity* dan *sequence diagram* untuk fungsi perubahan data master beasiswa.



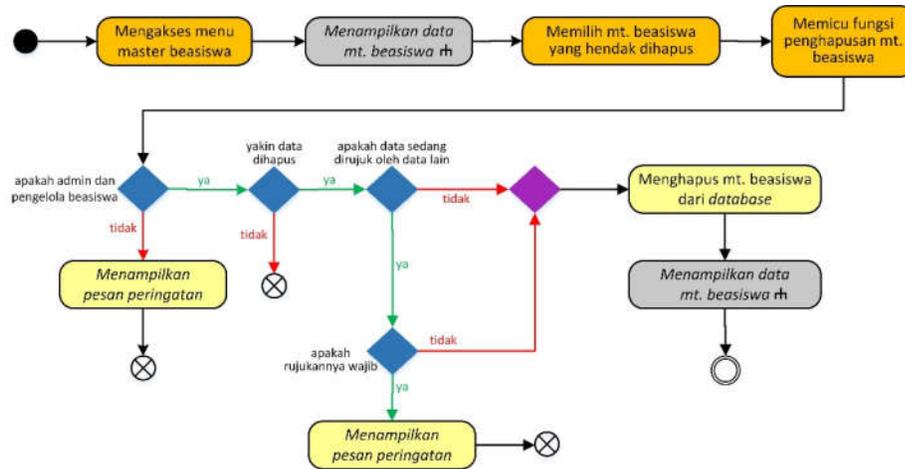
Gambar 34. Activity Diagram Fungsi Pengubahan Data Master Beasiswa.



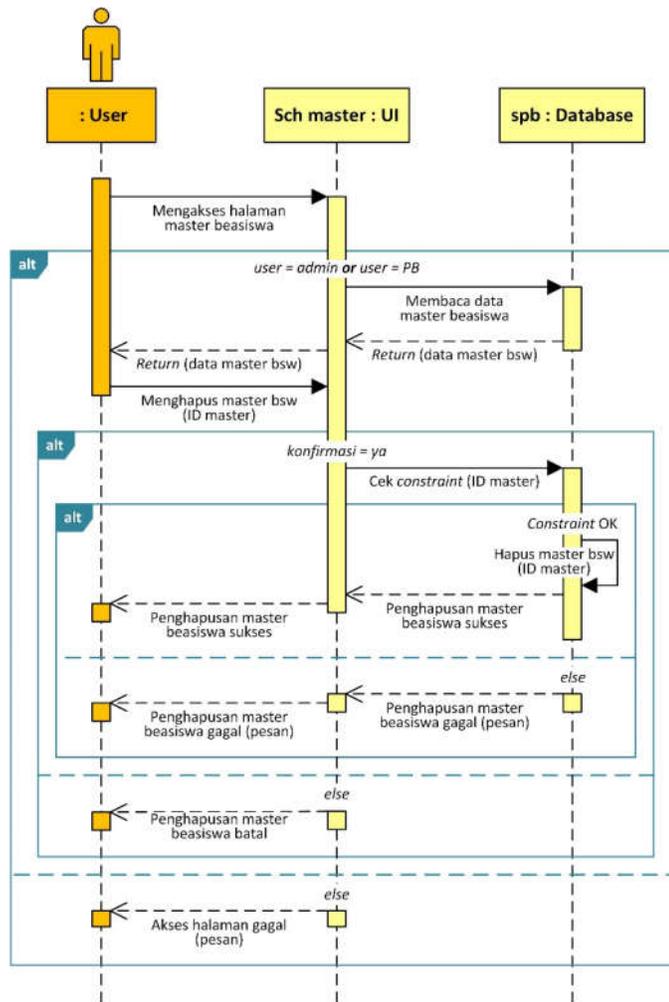
Gambar 35. Sequence Diagram Fungsi Pengubahan Data Master Beasiswa.

*d. Menghapus master beasiswa*

Suatu master beasiswa dapat dihapus bila belum ada satu pun beasiswa yang menggunakannya. Gambar 36 dan Gambar 37 memperlihatkan garis besar fungsi penghapusan master beasiswa.



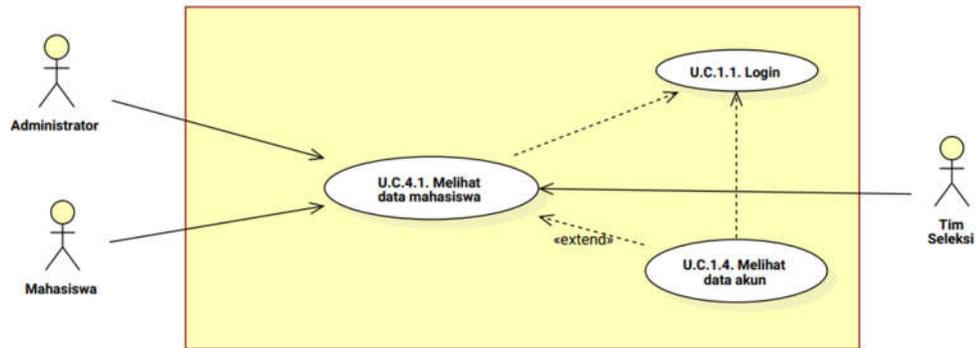
Gambar 36. Activity Diagram Fungsi Penghapusan Master Beasiswa.



Gambar 37. *Sequence Diagram* Fungsi Penghapusan Master Beasiswa.

#### 3.3.3.4. Mahasiswa

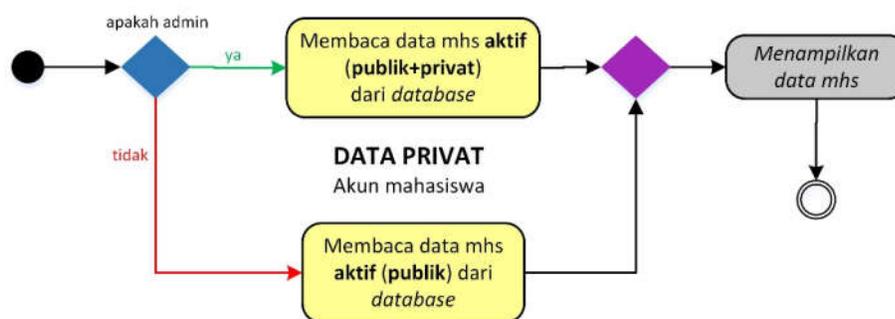
Supaya ruang lingkup sistem tidak melebar, blok **Mahasiswa** hanya mendefinisikan *use case* **melihat data mahasiswa**, seperti yang ditampilkan oleh Gambar 38. Masing-masing mahasiswa, selaku pendaftar beasiswa, mempunyai akun yang bisa digunakan untuk *login* ke dalam sistem.



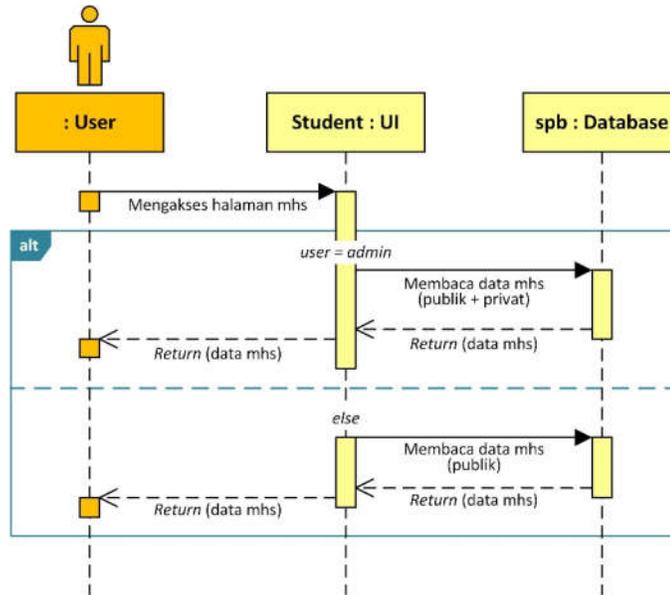
Gambar 38. *Use-Case* **Melihat Data Mahasiswa**.

Meskipun seluruh aktor sistem sanggup memicu fungsi ini, namun ada variasi terkait data yang boleh diakses. Data privat, semacam nama akun, hanya dapat dilihat oleh administrator. Pengguna lainnya dicukupkan hingga data publik saja, seperti nama lengkap, NPM, atau jurusan.

*Activity* dan *sequence diagram* untuk fungsi pembacaan data mahasiswa ditunjukkan oleh Gambar 39 dan Gambar 40.



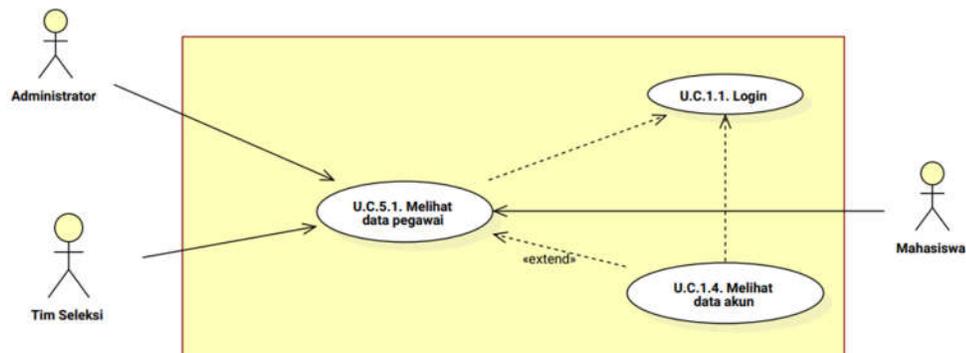
Gambar 39. *Activity Diagram* Fungsi Pembacaan Data Mahasiswa.



Gambar 40. *Sequence Diagram* Fungsi Pembacaan Data Mahasiswa.

### 3.3.3.5. Pegawai

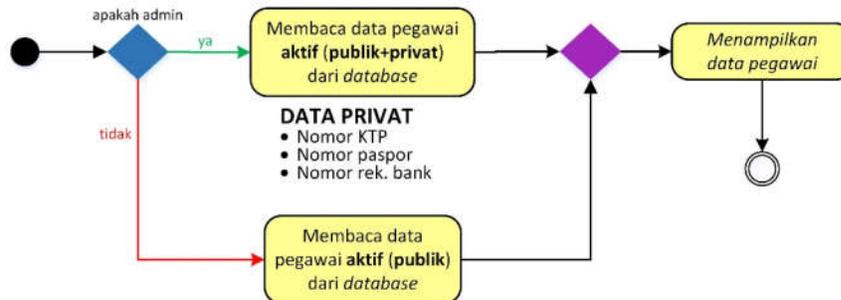
Setali tiga uang dengan blok **Mahasiswa**, demi menjaga agar jangkauan sistem tidak terlalu luas, blok **Pegawai** hanya mendefinisikan *use case* **melihat data pegawai**, seperti yang ditampilkan oleh Gambar 41. Pegawai-pegawai yang bertanggung jawab sebagai anggota tim seleksi memiliki akun yang bisa digunakan untuk *login* ke dalam sistem.



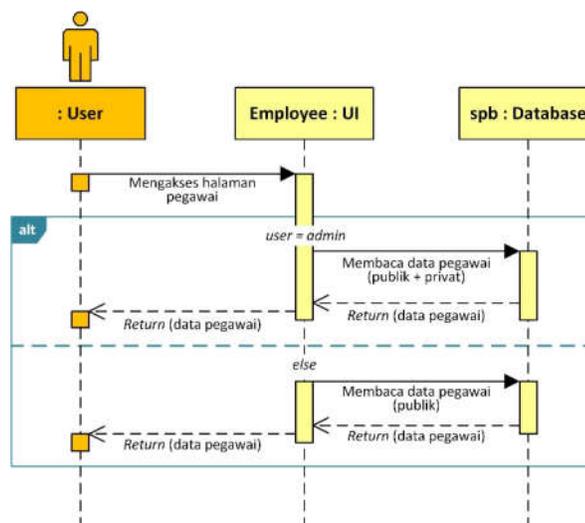
Gambar 41. *Use-Case* **Melihat Data Pegawai**.

Meskipun seluruh aktor sistem mampu memicu fungsi ini, namun ada variasi terkait data yang boleh diakses. Data privat, semacam nomor KTP, nomor paspor, dan nomor rekening bank, hanya dapat dilihat oleh administrator. Pengguna lainnya dicukupkan sampai data publik saja, seperti nama lengkap, nomor pegawai, alamat kantor, posisi, dan divisi kerja.

*Activity* dan *sequence diagram* untuk fungsi pembacaan data pegawai ditunjukkan oleh Gambar 42 dan Gambar 43.



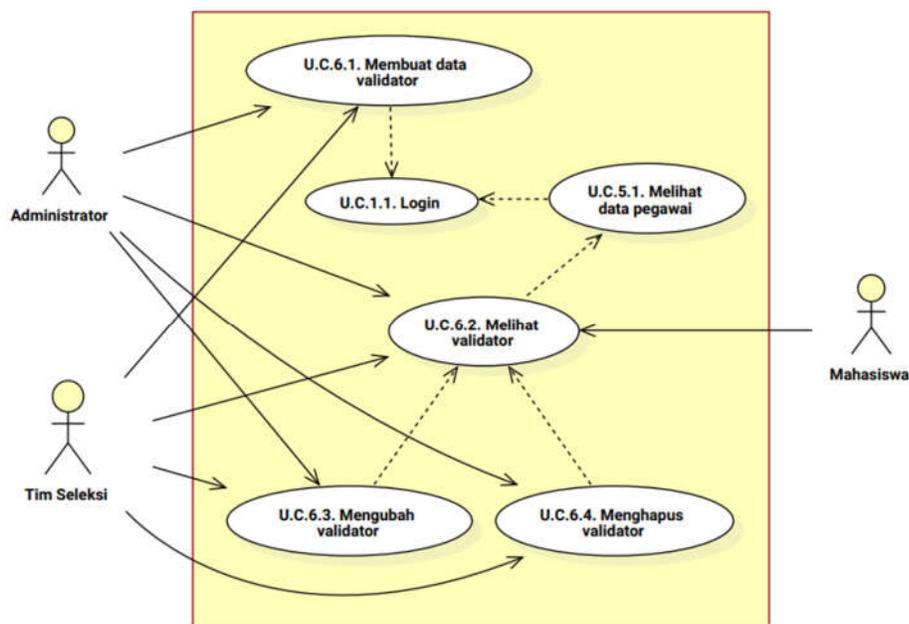
Gambar 42. *Activity Diagram* Fungsi Pembacaan Data Pegawai.



Gambar 43. *Sequence Diagram* Fungsi Pembacaan Data Pegawai.

### 3.3.3.6. Validator

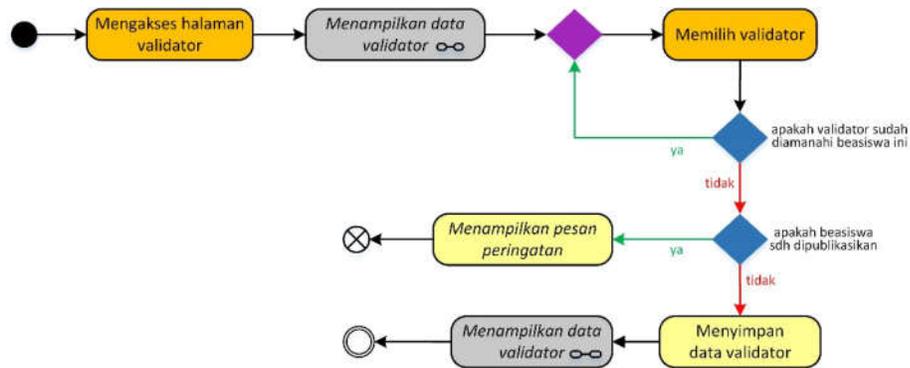
Validator ialah anggota tim seleksi yang bertugas memvalidasi/menolak arsip pendaftaran para pelamar. Gambar 44 menampilkan ragam *use case* yang termasuk dalam blok ini, di antaranya **membuat data validator**, **melihat validator**, **mengubah validator**, dan **menghapus validator**. Pengaturan akses untuk semua *use case* di blok **Validator** mengikuti blok **Beasiswa**.



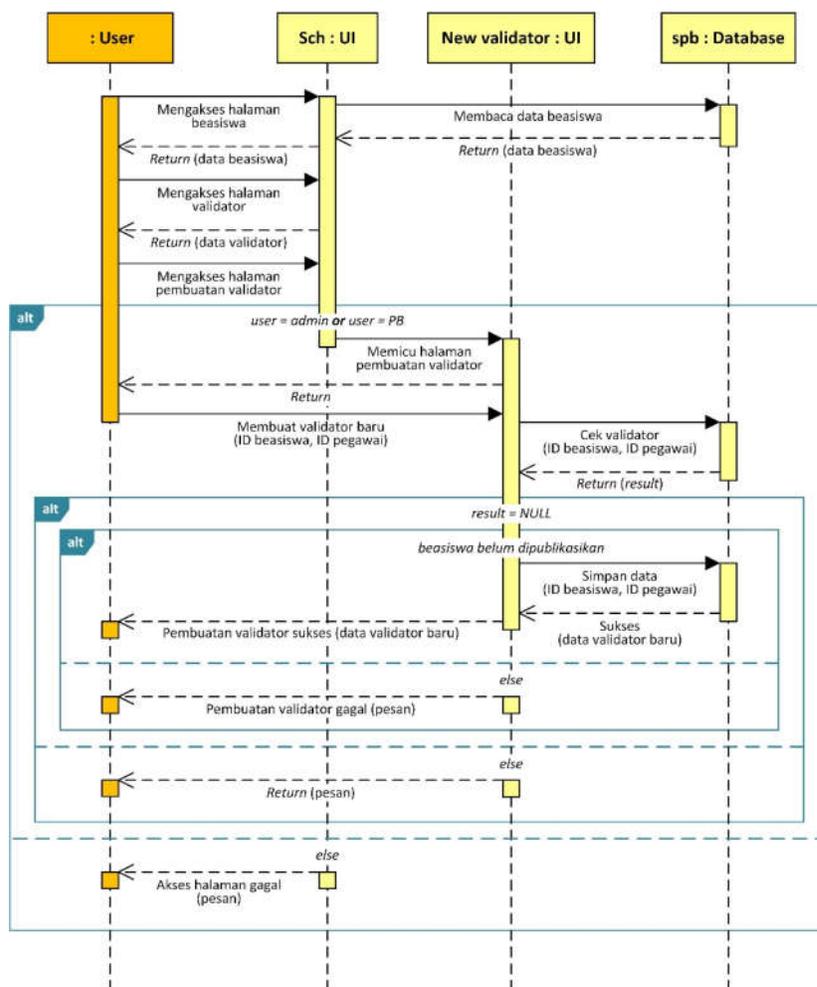
Gambar 44. Ragam *Use-Case* di Blok **Validator**.

#### a. *Membuat data validator*

Seorang pegawai dapat diamanahi sebagai validator beberapa beasiswa sekaligus, sementara satu beasiswa boleh diurus oleh lebih dari satu validator. Validator tiap beasiswa bisa terus ditambah jika beasiswa yang bersangkutan belum dipublikasikan. *Activity* dan *sequence diagram* untuk fungsi pembuatan data validator ditunjukkan oleh Gambar 45 dan Gambar 46.



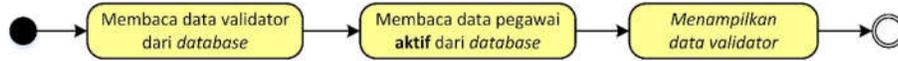
Gambar 45. Activity Diagram Fungsi Pembuatan Data Validator.



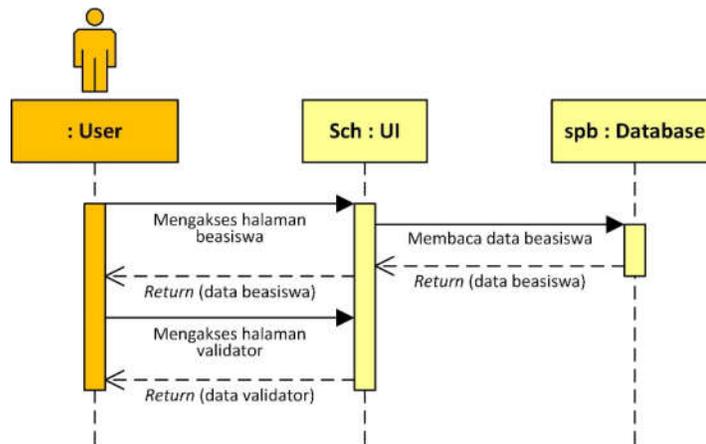
Gambar 46. Sequence Diagram Fungsi Pembuatan Data Validator.

**b. Melihat validator**

Gambar 47 dan Gambar 48 mendeskripsikan fungsi pembacaan data validator.



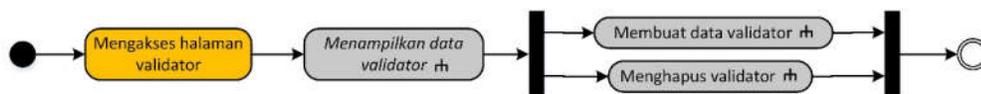
Gambar 47. *Activity Diagram* Fungsi Pembacaan Data Validator.



Gambar 48. *Sequence Diagram* Fungsi Pembacaan Data Validator.

**c. Mengubah validator**

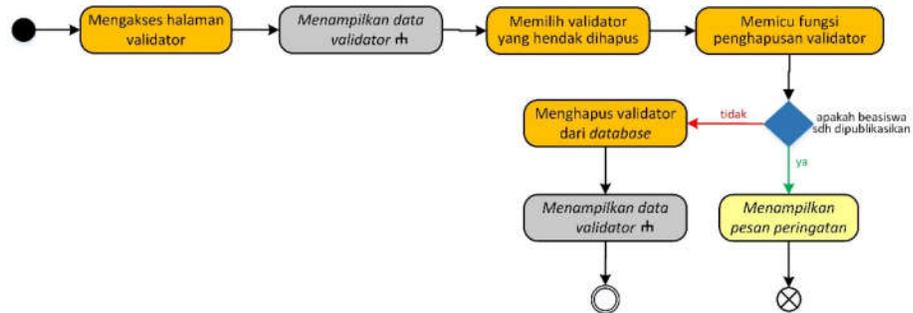
Fungsi ini sejatinya merupakan gabungan dari *use case* **Membuat data validator** dan **Menghapus validator**. Gambar 49 menampilkan *activity diagram* untuk fungsi perubahan data validator. *Sequence diagram*-nya diwakili oleh Gambar 46 dan Gambar 51.



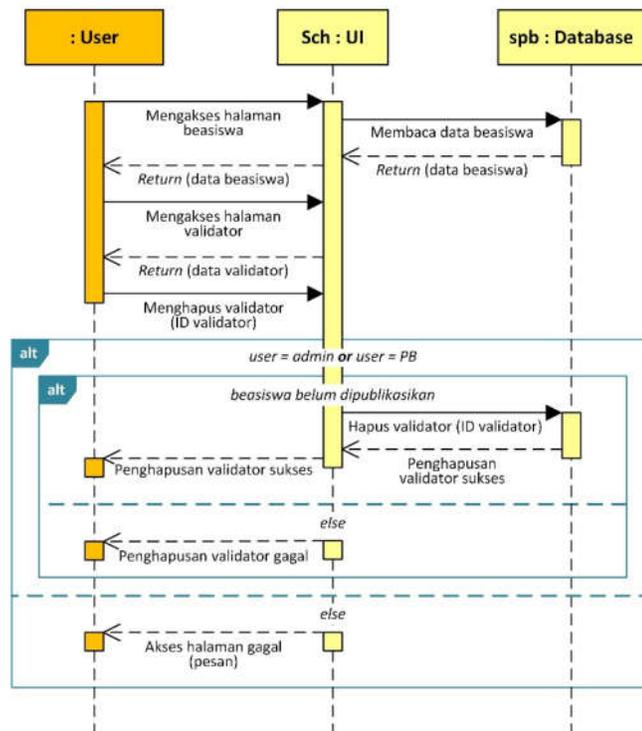
Gambar 49. *Activity Diagram* Fungsi Perubahan Data Validator.

d. *Menghapus validator*

Tampuk validator yang dipegang oleh seorang pegawai bisa dicabut selama beasiswa yang bersangkutan belum dipublikasikan. Gambar 50 dan Gambar 51 memperlihatkan garis besar fungsi penghapusan validator.



Gambar 50. *Activity Diagram* Fungsi Penghapusan Validator.

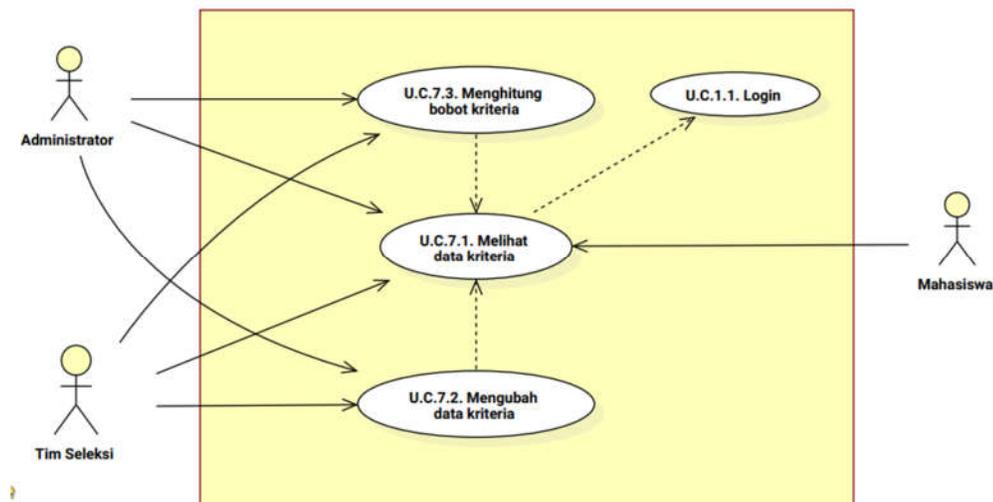


Gambar 51. *Sequence Diagram* Fungsi Penghapusan Validator.

### 3.3.3.7. Kriteria

Gambar 52 menampilkan ragam *use case* yang termasuk dalam blok **Kriteria**, di antaranya **melihat data kriteria**, **mengubah data kriteria**, dan **menghitung bobot kriteria**.

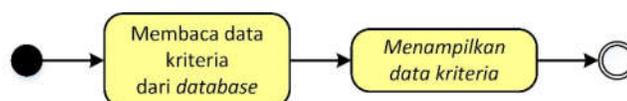
Selaras dengan blok **Validator**, akses ke semua *use case* di blok ini mengikuti blok **Beasiswa**, karena keduanya adalah unit-unit yang menyatu bersama beasiswa.



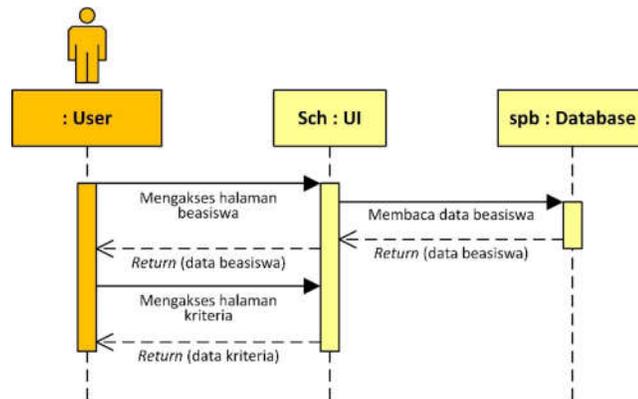
Gambar 52. Ragam *Use-Case* di Blok **Kriteria**.

#### a. *Melihat data kriteria*

*Activity* dan *sequence diagram* untuk fungsi pembacaan data kriteria ditunjukkan oleh Gambar 53 dan Gambar 54.



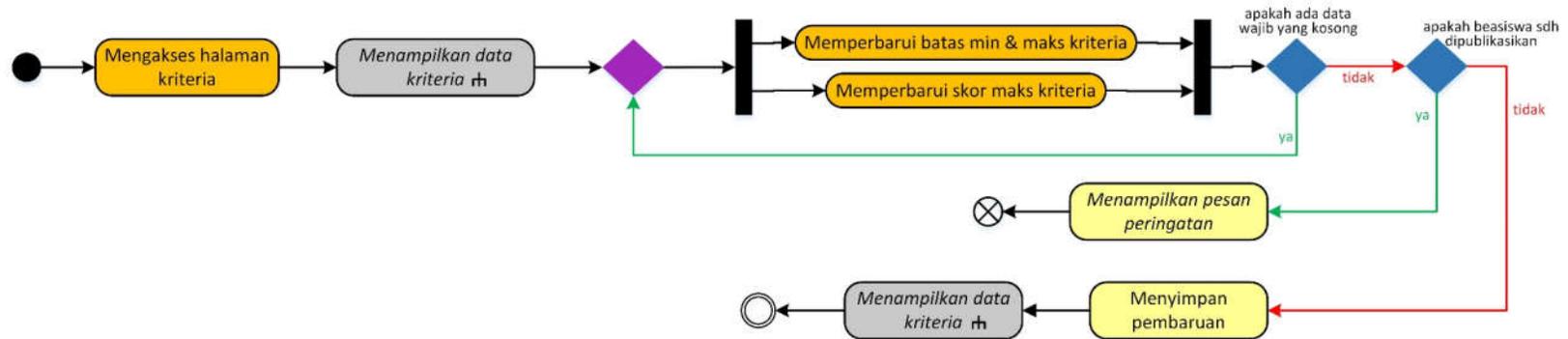
Gambar 53. *Activity Diagram* Fungsi Pembacaan Data Kriteria.



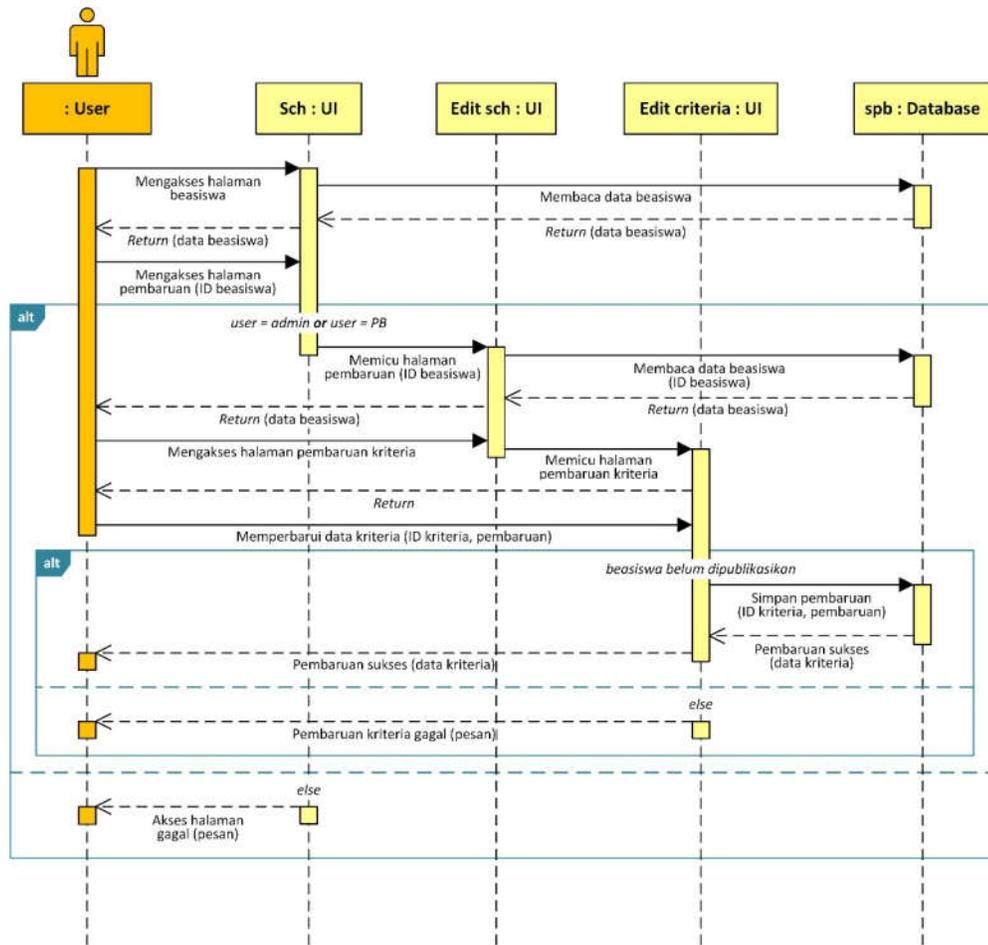
Gambar 54. *Sequence Diagram* Fungsi Pembacaan Data Kriteria.

**b. Mengubah data kriteria**

Data kriteria yang bisa diubah adalah batas minimum *IPK* serta batas minimum dan maksimum *Semester*. Pembaruan dapat dilakukan jika beasiswa belum dipublikasikan. Gambar 55 dan Gambar 56 mendeskripsikan fungsi perubahan data kriteria.



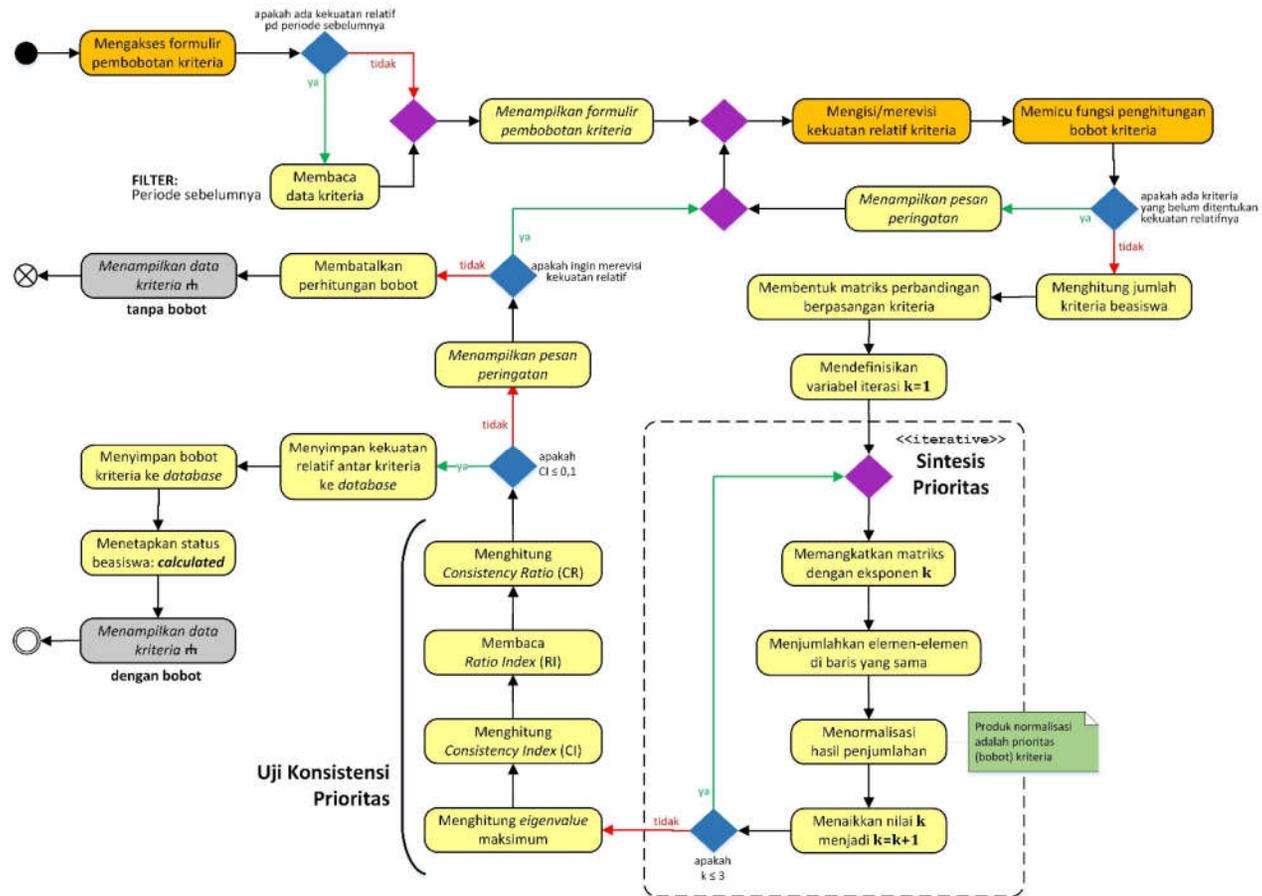
Gambar 55. Activity Diagram Fungsi Perubahan Data Kriteria.



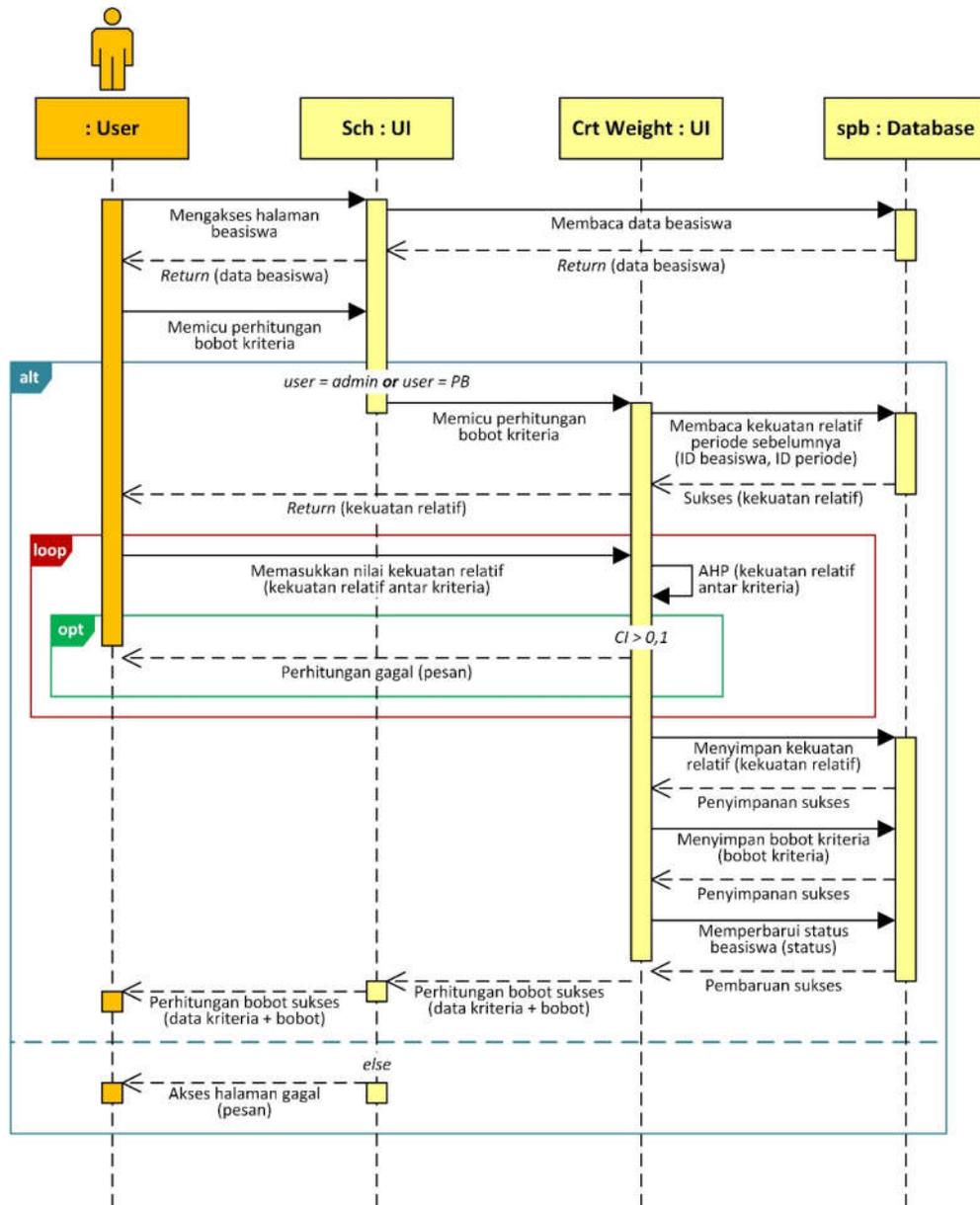
Gambar 56. *Sequence Diagram* Fungsi Pengubahan Data Kriteria.

### c. Menghitung bobot kriteria

Gambar 57 dan Gambar 58 menampilkan *activity diagram* dan *sequence diagram* untuk fungsi estimasi bobot kriteria. Sebelum prosedur perhitungan dieksekusi, kekuatan relatif antar kriteria harus diisi dahulu oleh pengguna yang berwenang. Supaya *user* terhindar dari memasukkan bilangan-bilangan yang persis berulang-ulang, sistem menyuguhkan kadar dominasi pada periode yang lampau secara otomatis. Bobot yang diturunkan dari metode *Analytic Hierarchy Process* selanjutnya disimpan di dalam *database*, kemudian status beasiswa diperbarui.



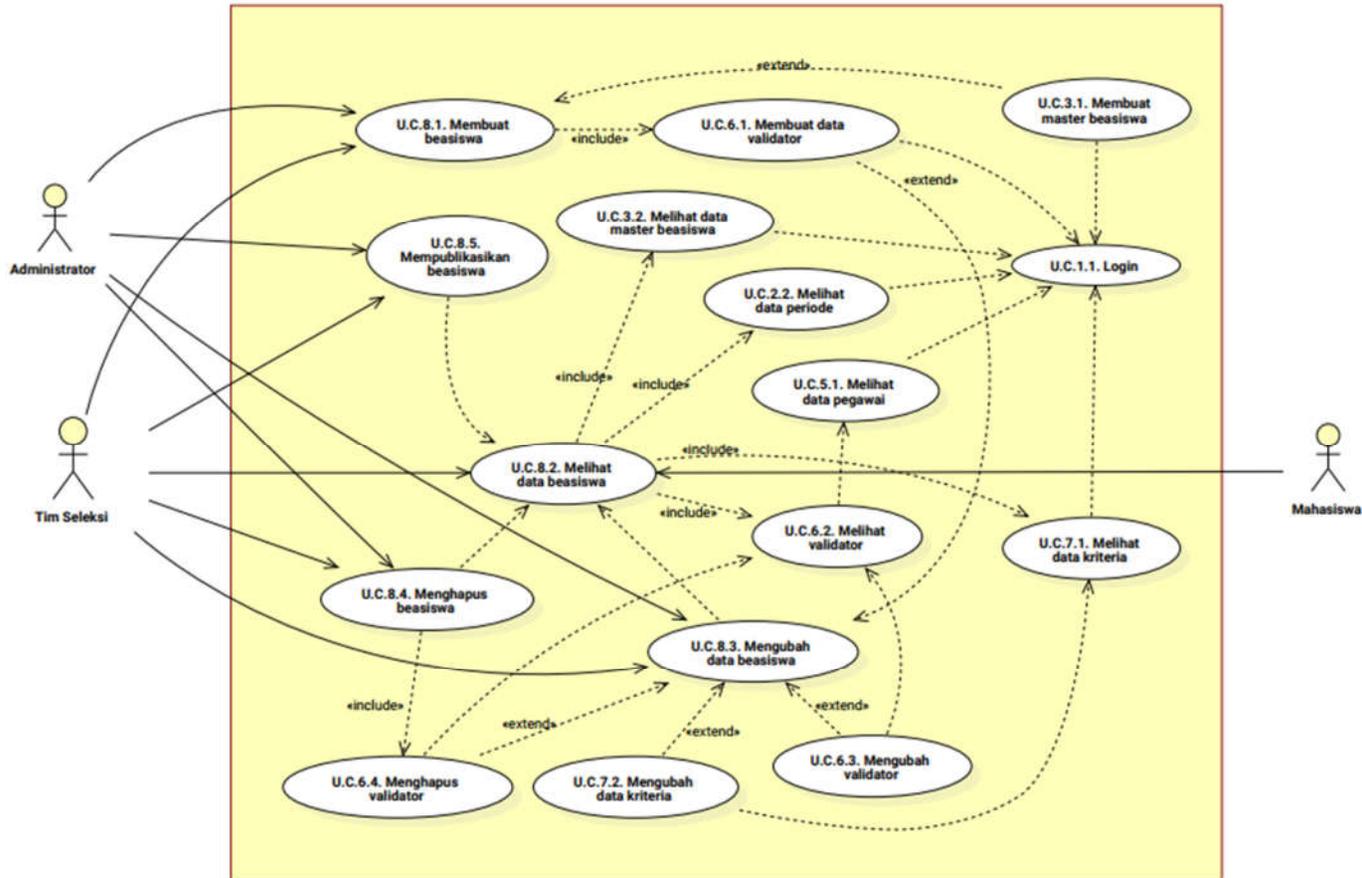
Gambar 57. Activity Diagram Fungsi Perhitungan Bobot Kriteria.



Gambar 58. *Sequence Diagram* Fungsi Perhitungan Bobot Kriteria.

### 3.3.3.8. Beasiswa

Gambar 59 menampilkan ragam *use case* yang termasuk dalam blok **Beasiswa**, di antaranya **membuat beasiswa**, **melihat data beasiswa**, **mengubah data beasiswa**, **menghapus beasiswa**, dan **mempublikasikan beasiswa**.



Gambar 59. Ragam *Use-Case* di Blok **Beasiswa**.

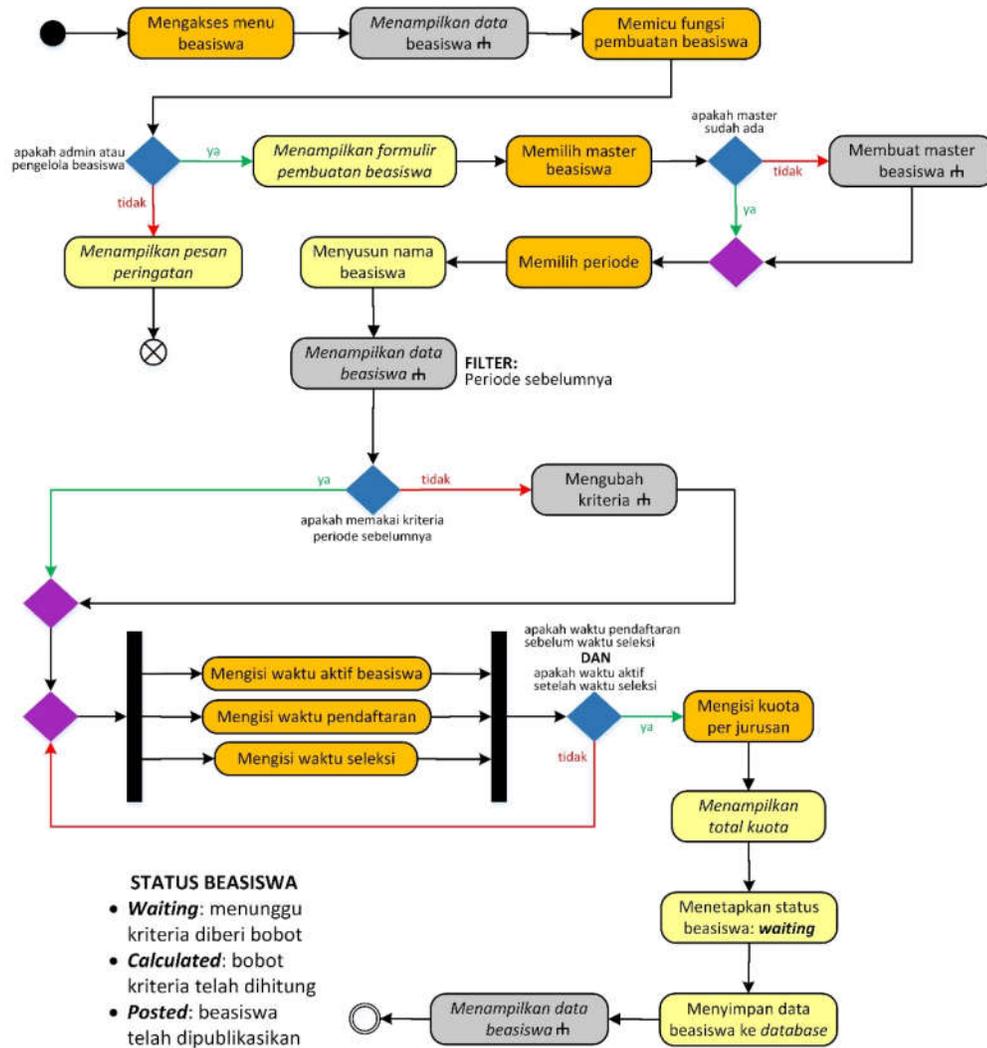
*a. Membuat beasiswa*

Masukan yang dibutuhkan oleh fungsi ini ialah master beasiswa, periode, kriteria-kriteria, waktu aktif beasiswa, rentang waktu pendaftaran, waktu seleksi, dan kuota jurusan. Sebelum disimpan ke dalam *database*, dilakukan pengecekan tanggal agar limit akhir registrasi senantiasa lebih kecil daripada waktu seleksi dan beasiswa hanya berlaku selepas seleksi selesai. Selain itu, total kuota juga dihitung dan turut direkam.

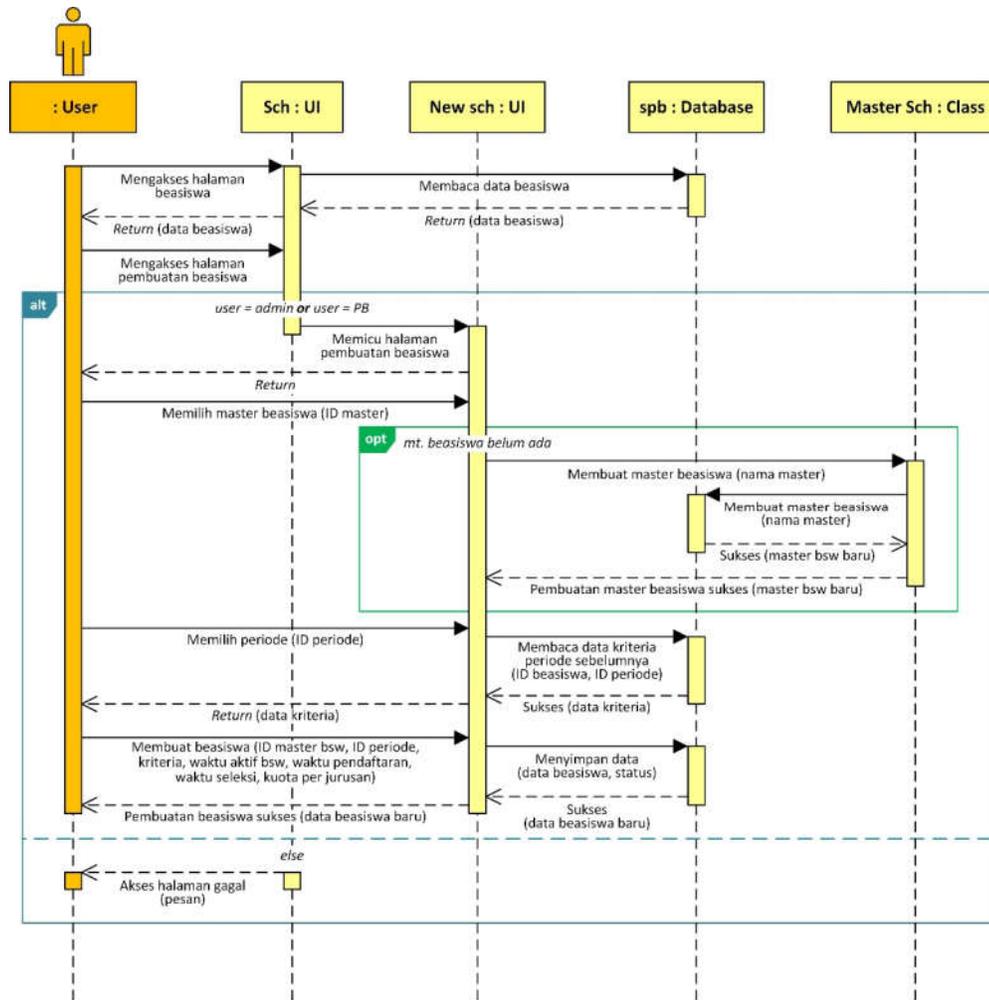
Guna mempermudah proses pembuatan beasiswa, pasca master beasiswa dan periode di-*input*-kan, sistem mencari beasiswa bermaster sama pada periode sebelumnya, kemudian menampilkan data kriterianya pada beasiswa yang baru. Kualifikasi suatu beasiswa yang biasanya jarang berubah dari tahun ke tahun mendorong munculnya ide ini. Jika ternyata detail kriteria yang dikehendaki berbeda dengan yang ditayangkan oleh sistem, administrator atau tim seleksi tinggal merevisinya hingga sesuai dengan keinginan.

Beasiswa yang baru saja dibuat dikenai status **waiting**, yakni menunggu dihitungnya bobot kriteria. Bila bobot yang dimaksud sudah dikalkulasi, statusnya naik menjadi **calculated**. Status terakhir adalah **posted** yang berarti beasiswa telah dipublikasikan. Beasiswa pada fase ini dapat dilihat lalu didaftari oleh mahasiswa kalau waktu pendaftarannya tiba.

*Activity* dan *sequence diagram* untuk fungsi pembuatan beasiswa ditunjukkan oleh Gambar 60 dan Gambar 61.



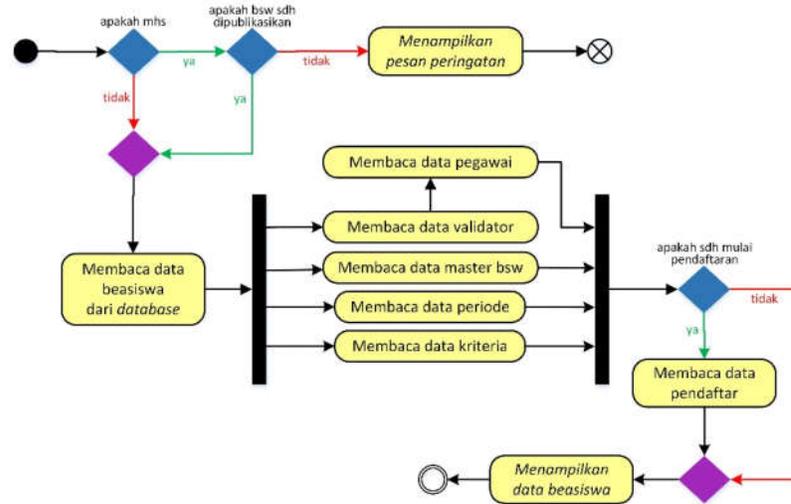
Gambar 60. Activity Diagram Fungsi Pembuatan Beasiswa.



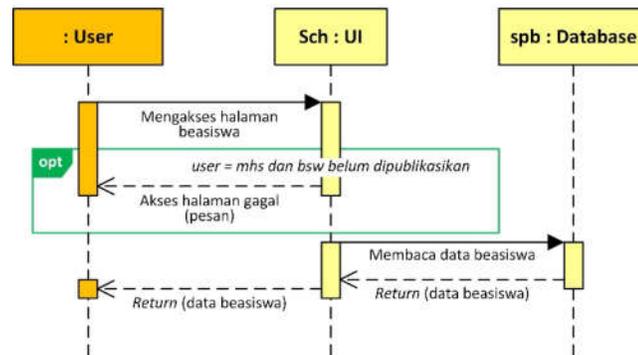
Gambar 61. *Sequence Diagram* Fungsi Pembuatan Beasiswa.

**b. Melihat data beasiswa**

Data kriteria dan validator merupakan bagian dari data beasiswa, sehingga keduanya akan selalu diikutsertakan apabila aktor memicu fungsi ini. Informasi tentang para pendaftar pun akan disajikan seandainya ada mahasiswa yang pernah melamar beasiswa terkait. Gambar 62 dan Gambar 63 mendeskripsikan fungsi pembacaan data beasiswa secara ringkas.



Gambar 62. Activity Diagram Fungsi Pembacaan Data Beasiswa.

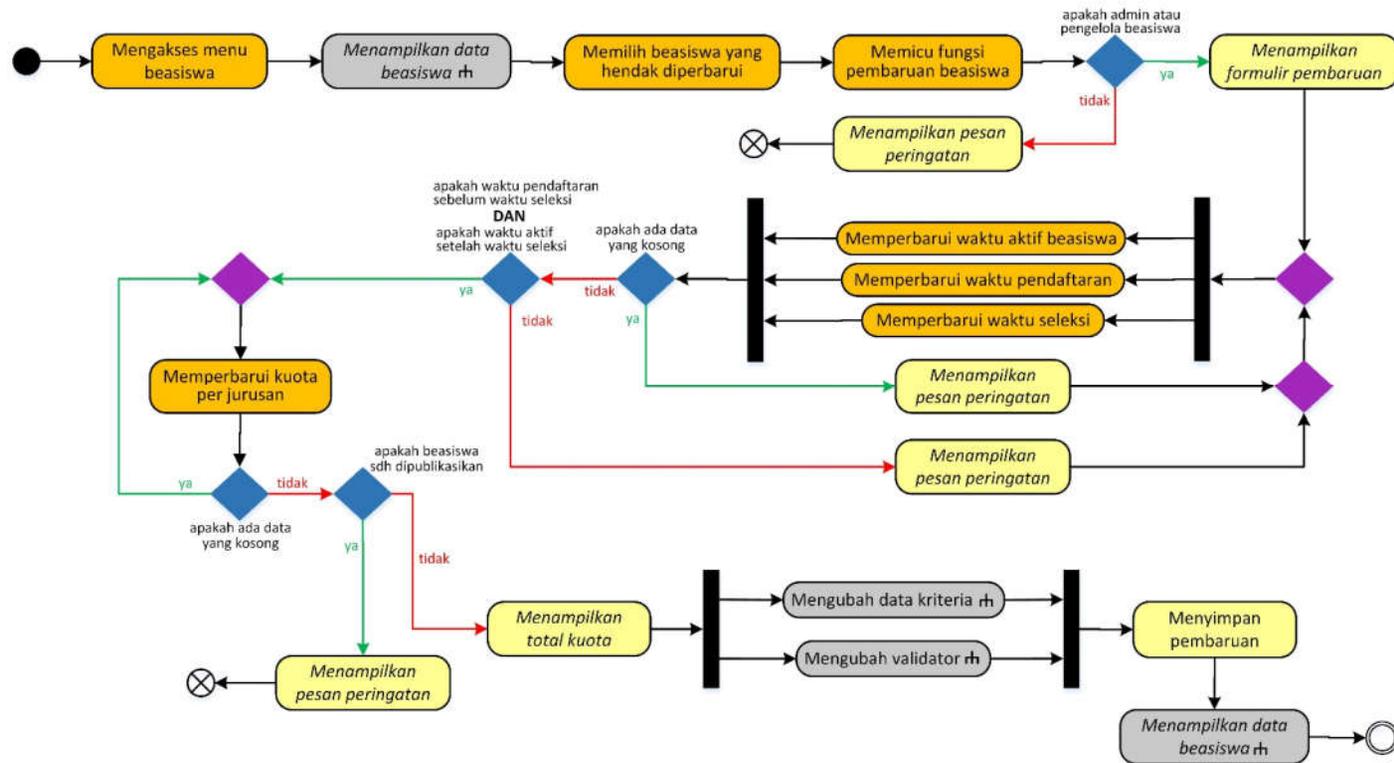


Gambar 63. Sequence Diagram Fungsi Pembacaan Data Beasiswa.

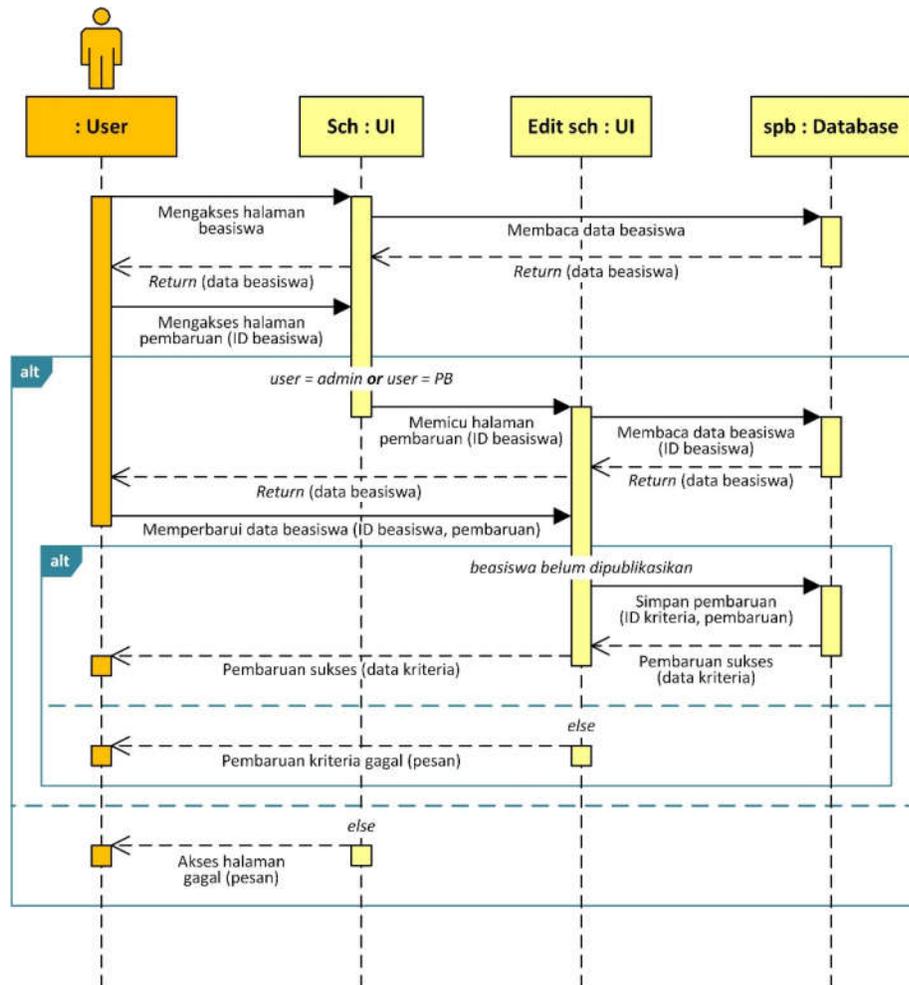
### c. Mengubah data beasiswa

Tepat sesudah dibuat, baik master maupun periode beasiswa tidak bisa diganti lagi, sehingga tersisa kriteria, waktu aktif, waktu registrasi, waktu seleksi, dan kuota jurusan yang masih mungkin dimodifikasi. Pembaruan pada segenap *field* tadi direkam ke *database* jika beasiswa belum dipublikasikan. Pengguna yang berhak mengeksekusi fungsi ini adalah administrator dan tim seleksi.

Gambar 64 dan Gambar 65 menampilkan *activity diagram* dan *sequence diagram* untuk fungsi perubahan data beasiswa.



Gambar 64. *Activity Diagram* Fungsi Perubahan Data Beasiswa.

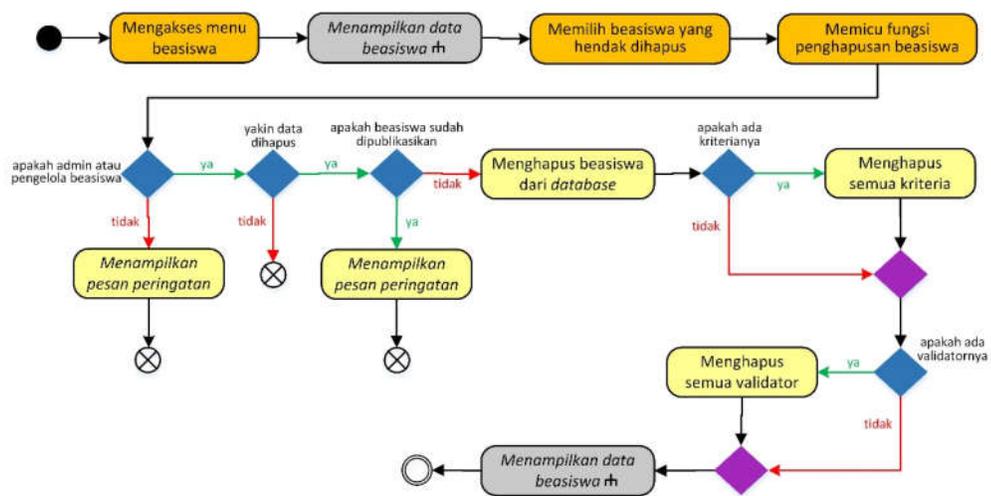


Gambar 65. *Sequence Diagram* Fungsi Perubahan Data Beasiswa.

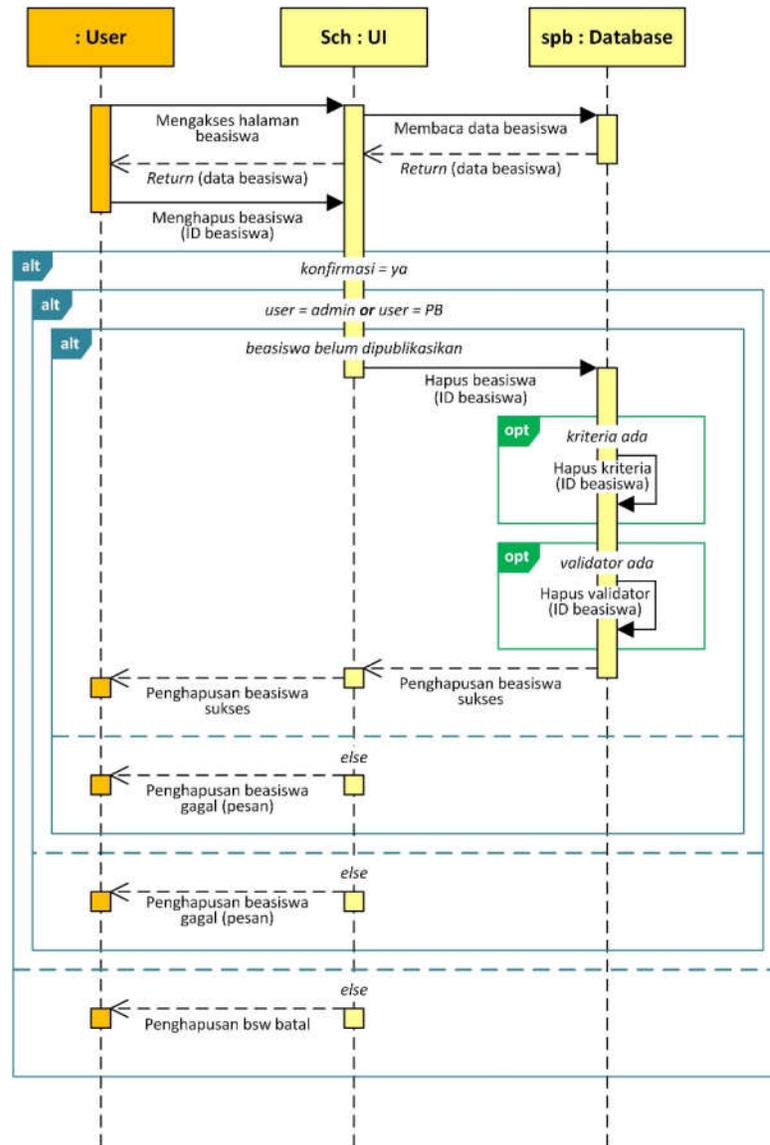
**d. Menghapus beasiswa**

Kondisi yang harus terpenuhi supaya suatu beasiswa dapat terhapus ialah belum dipublikasikan. *Use case* ini turut menghilangkan data kriteria berikut validator yang mengiringi beasiswa bersangkutan. Aktor yang berhak menjalankannya hanya administrator dan tim seleksi.

Gambar 66 dan Gambar 67 memperlihatkan garis besar fungsi penghapusan beasiswa.



Gambar 66. Activity Diagram Fungsi Penghapusan Beasiswa.



Gambar 67. *Sequence Diagram* Fungsi Penghapusan Beasiswa.

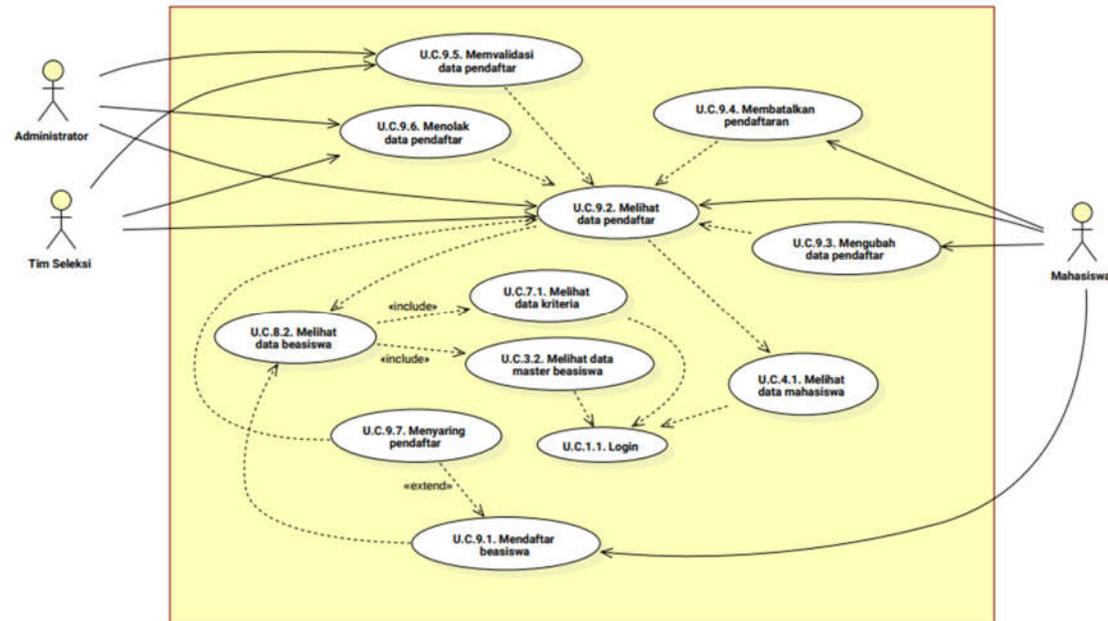
*e. Mempublikasikan beasiswa*

Fungsi ini dioperasikan oleh administrator atau tim seleksi. Agar status beasiswa sukses beralih menjadi **posted**, validatornya wajib ada dan bobot kriteria telah terhitung. Gambar 68 dan Gambar 69 mempresentasikan fungsi publikasi beasiswa.



### 3.3.3.9. Pendaftaran

Gambar 70 menampilkan ragam *use case* yang termasuk dalam blok **Pendaftaran**, di antaranya **menyaring pendaftar**, **mendaftar beasiswa**, **melihat data pendaftar**, **mengubah data pendaftar**, **membatalkan pendaftaran**, **memvalidasi data pendaftar**, dan **menolak data pendaftar**.



Gambar 70. Ragam *Use-Case* di Blok **Pendaftaran**.

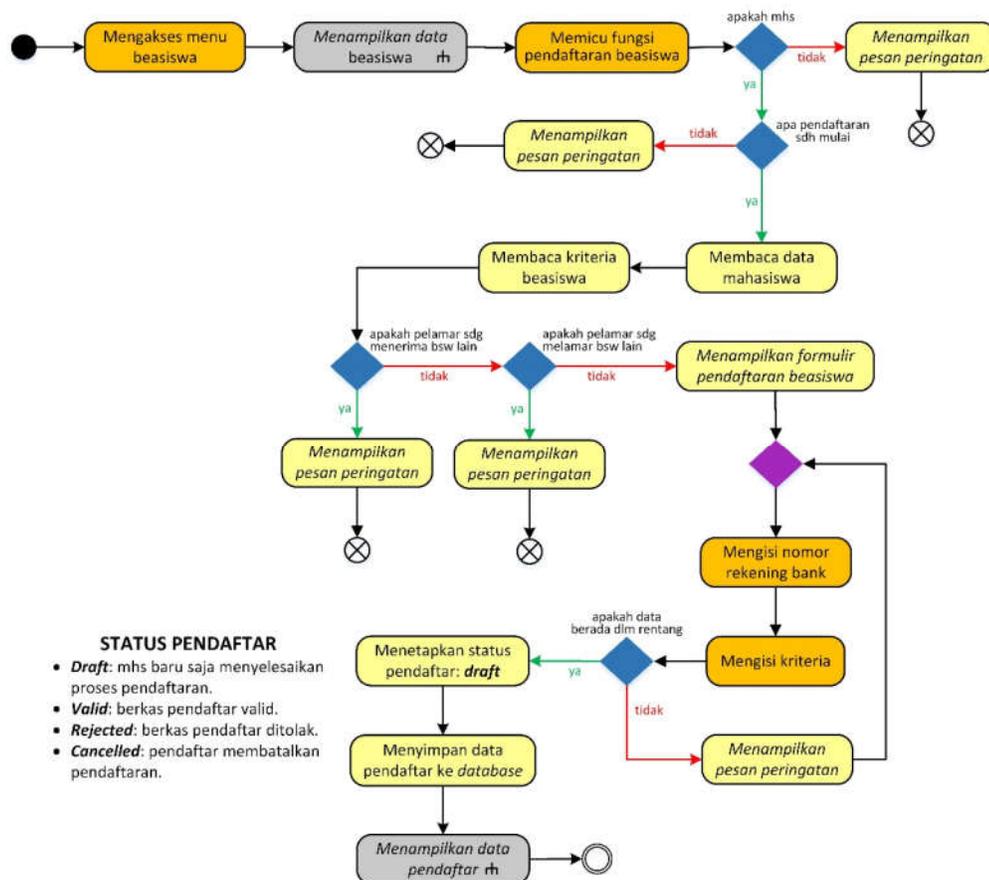
*a. Menyaring pendaftar*

Fungsi **menyaring pendaftar** sesungguhnya merupakan bagian dari fungsi **mendaftar beasiswa**. Pekerjaan yang diemban oleh *use case* ini adalah mengecek apakah mahasiswa yang hendak mendaftar tidak sedang melamar atau menerima beasiswa lain, kemudian membukakan formulir registrasi untuk mereka yang sesuai dengan filter saja. Kebijakan yang demikian diterapkan oleh pengelola beasiswa di FMIPA Universitas Lampung agar mahasiswa/i di fakultasnya memperoleh tepat satu beasiswa.

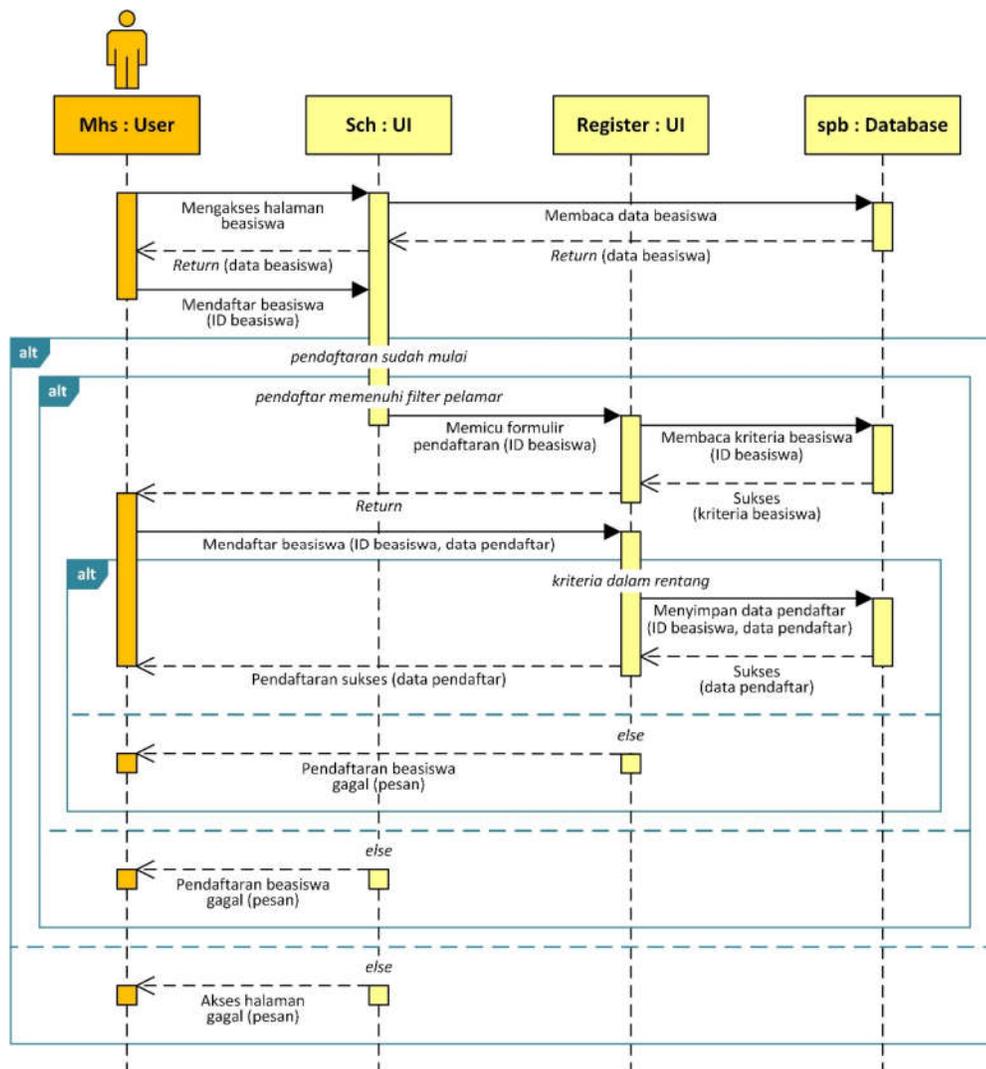
*b. Mendaftar beasiswa*

Fungsi yang diperuntukkan bagi mahasiswa ini dapat dijalankan jika waktunya sudah tiba. Data pelamar tidak akan dimasukkan ke *database* bila nilai IPK dan semesternya tidak berada dalam rentang yang telah ditetapkan. Prosedur pendaftaran yang berhasil ditandai dengan status **draft**.

*Activity* dan *sequence diagram* untuk fungsi pendaftaran beasiswa ditunjukkan oleh Gambar 71 dan Gambar 72.



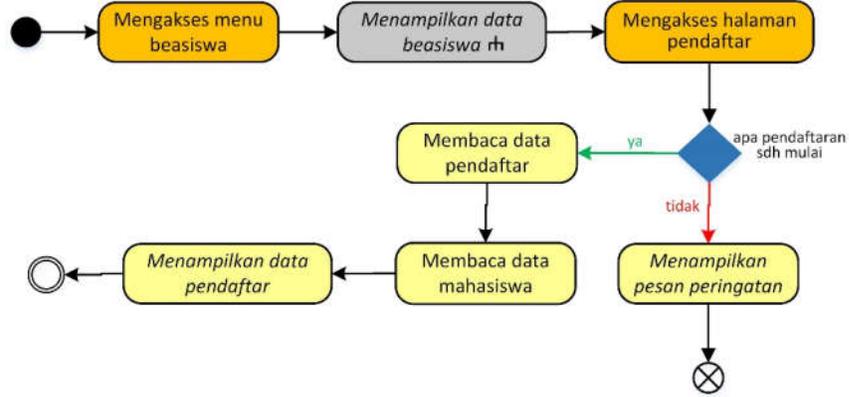
Gambar 71. Activity Diagram Fungsi Pendaftaran Beasiswa.



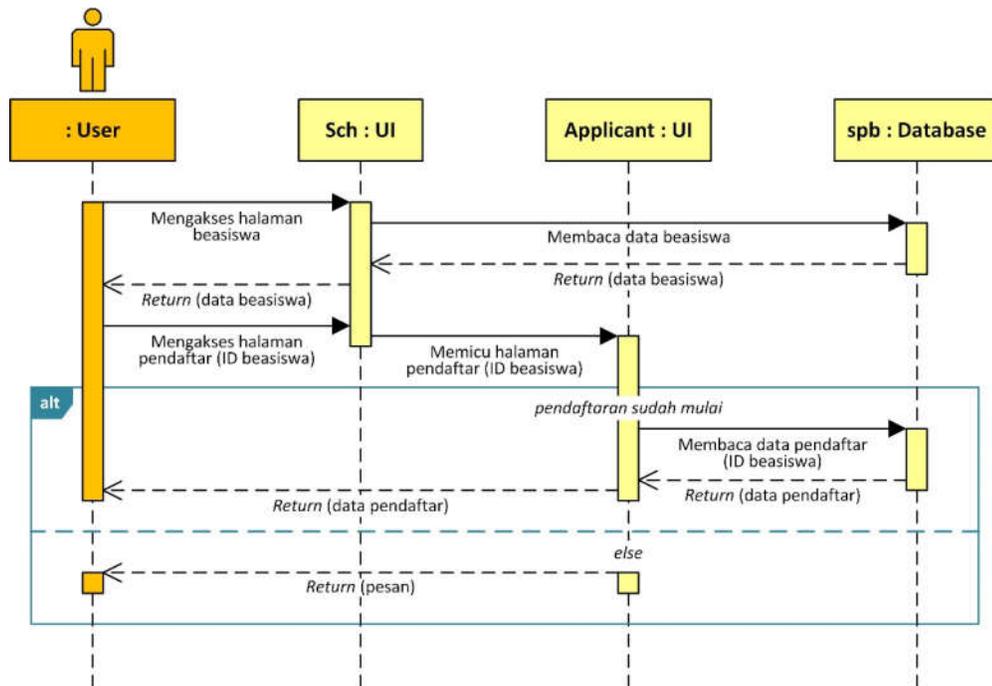
Gambar 72. Sequence Diagram Fungsi Pendaftaran Beasiswa.

c. *Melihat data pendaftar*

Gambar 73 dan Gambar 74 mendeskripsikan fungsi pembacaan data pendaftar.



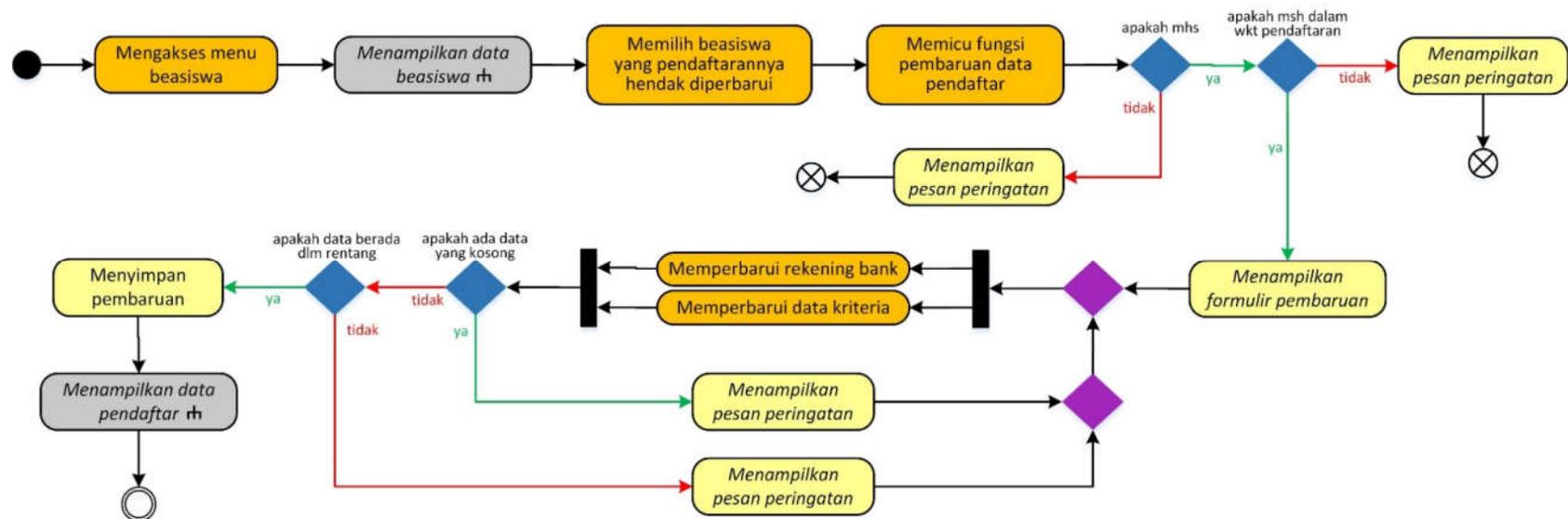
Gambar 73. *Activity Diagram* Fungsi Pembacaan Data Pendaftar.



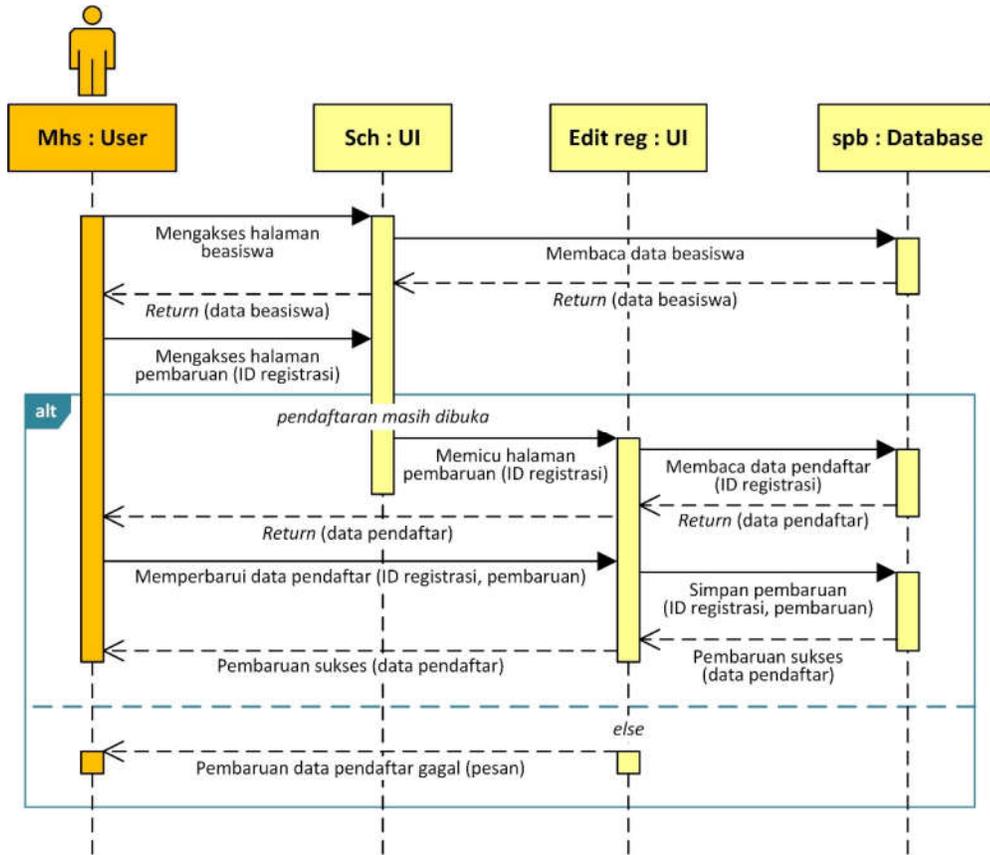
Gambar 74. *Sequence Diagram* Fungsi Pembacaan Data Pendaftar.

d. *Mengubah data pendaftar*

Mahasiswa, selaku aktor satu-satunya fungsi ini, bisa memperbarui detail pendaftarannya sendiri selama batas akhir pendaftaran belum terlewati. Gambar 75 dan Gambar 76 menampakkan *activity* dan *sequence diagram* untuk fungsi pengubahan data pendaftar.



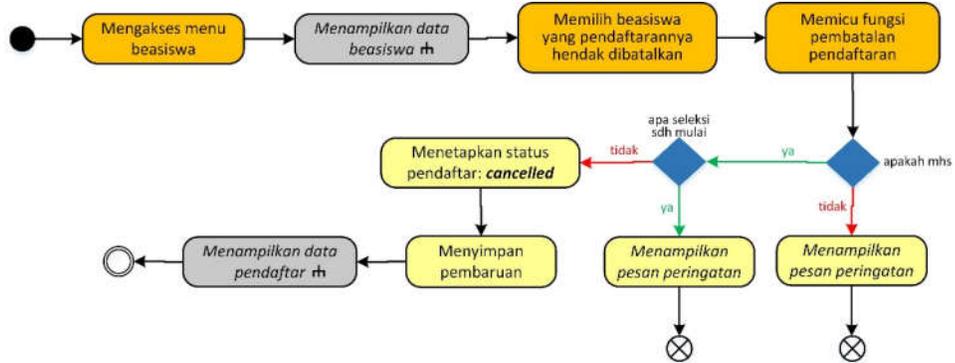
Gambar 75. *Activity Diagram* Fungsi Pengubahan Data Pendaftar.



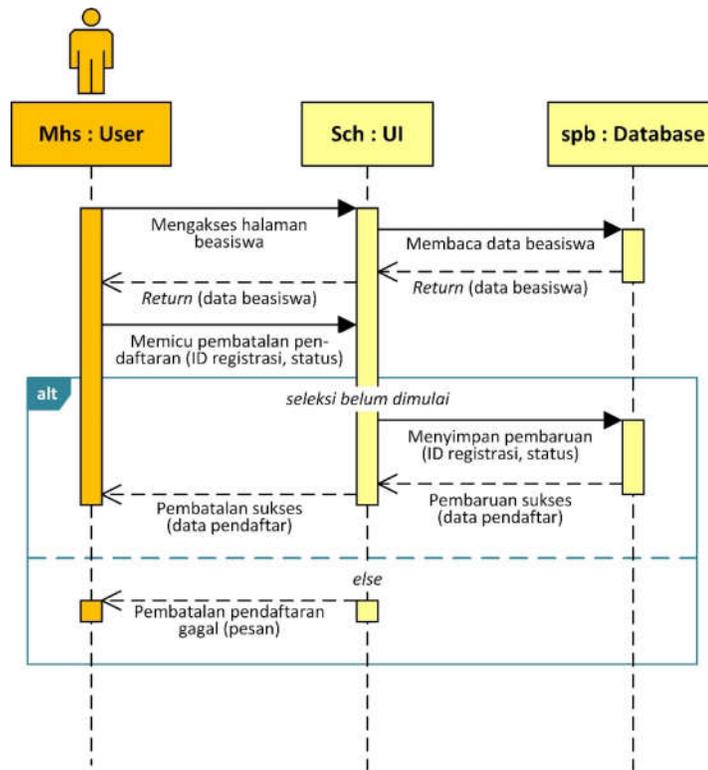
Gambar 76. Sequence Diagram Fungsi Pengubahan Data Pendaftar.

*e. Membatalkan pendaftaran*

*Use case* ini ada untuk mengantisipasi para pelamar yang tak ingin berlanjut ke proses seleksi. Aksi pembatalan memodifikasi status pendaftar menjadi **cancelled**. Gambar 77 dan Gambar 78 memperlihatkan garis besar fungsi pembatalan pendaftaran.



Gambar 77. Activity Diagram Fungsi Pembatalan Pendaftaran.



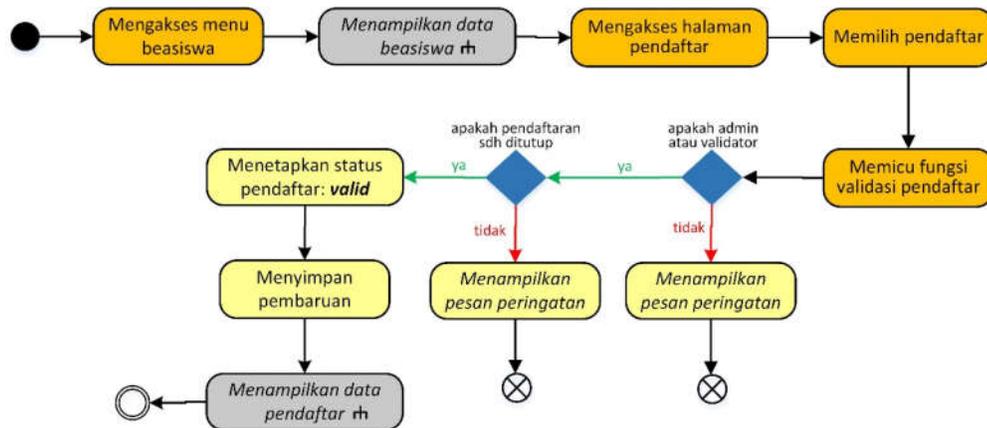
Gambar 78. Sequence Diagram Fungsi Pembatalan Pendaftaran.

*f. Memvalidasi dan menolak data pendaftar*

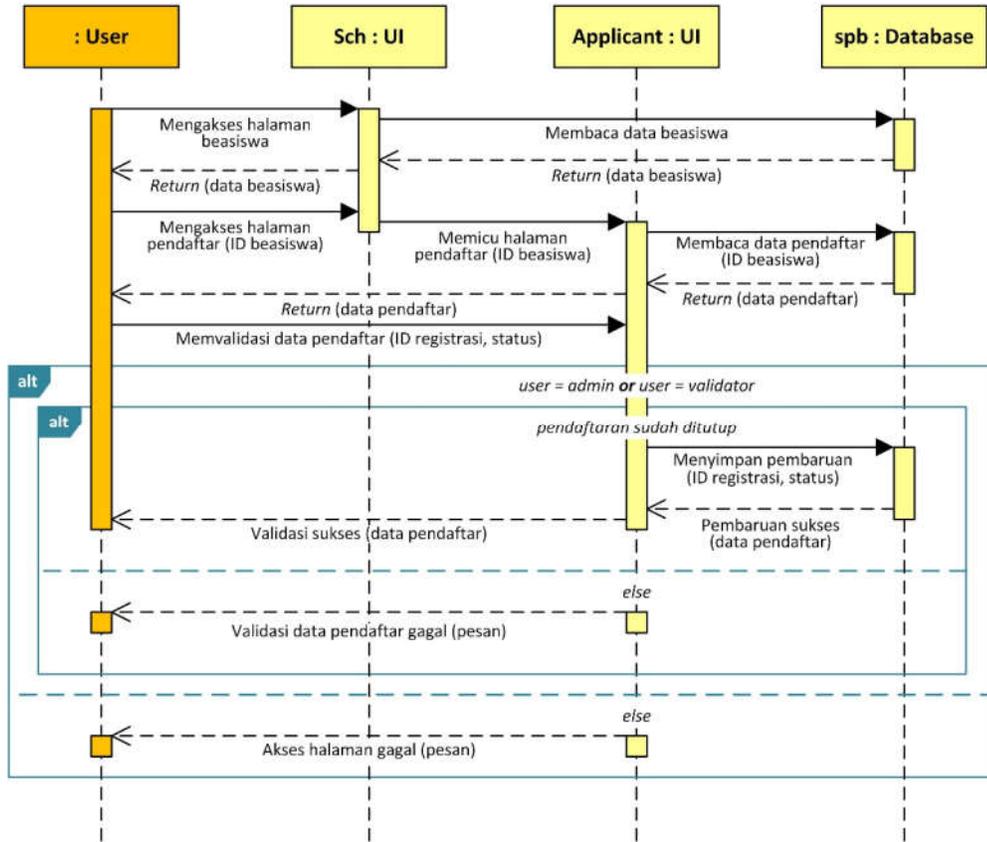
Data pelamar dianggap valid jika berkas pendaftarannya lengkap dan segala yang tercatat di formulir registrasinya sinkron dengan berkas tersebut. Di luar dua ihwal

itu, data yang diajukan berhak untuk ditolak. Pengaktifan masing-masing *use case* berakibat pada beralihnya status pendaftar menjadi, berturut-turut, **valid** dan **rejected**. Status pertama membuat mahasiswa yang bersangkutan disertakan dalam proses seleksi, sementara yang kedua berlaku sebaliknya.

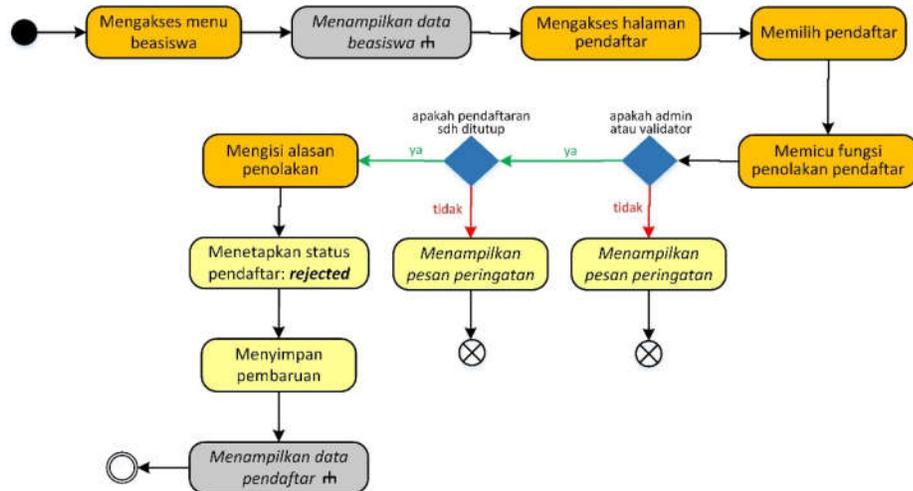
Dua fungsi ini menjadi wewenang validator yang ditunjuk dan dapat dilangsungkan kalau pendaftaran sudah ditutup. Operasi selengkapnya divisualisasikan oleh *activity* dan *sequence diagram*, yakni Gambar 79 dan Gambar 80 untuk fungsi validasi data pendaftar serta Gambar 81 dan Gambar 82 untuk fungsi penolakan data pendaftar.



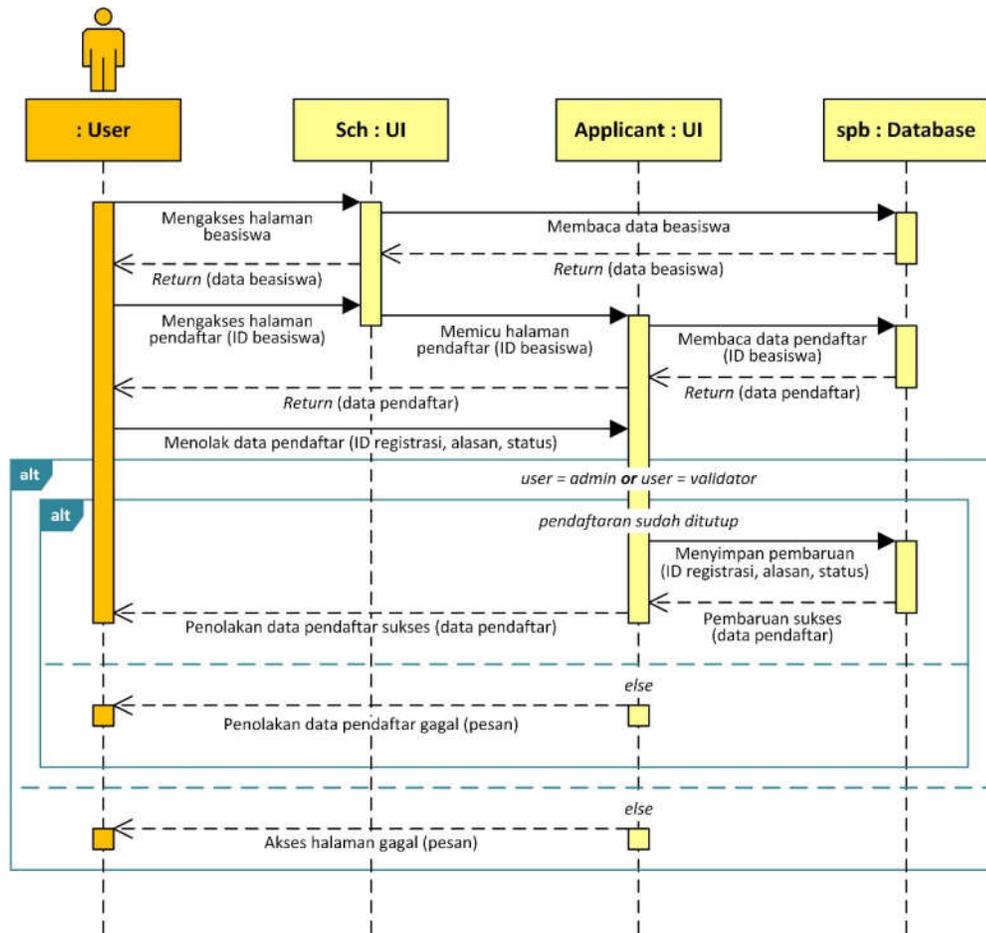
Gambar 79. Activity Diagram Fungsi Validasi Data Pendaftar.



Gambar 80. Sequence Diagram Fungsi Validasi Data Pendaftar.



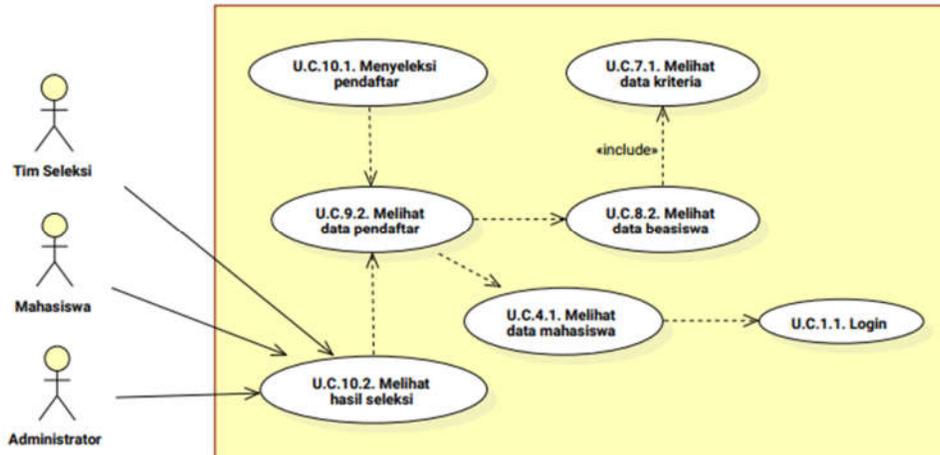
Gambar 81. Activity Diagram Fungsi Penolakan Data Pendaftar.



Gambar 82. *Sequence Diagram* Fungsi Penolakan Data Pendaftar.

### 3.3.3.10. Seleksi

Gambar 83 menampilkan ragam *use case* yang termasuk dalam blok ini, di antaranya **menyeleksi pendaftar** dan **melihat hasil seleksi**. Kontras dengan tandemnya yang mampu dieksekusi oleh siapapun yang *login*, *use case menyeleksi pendaftar* tidak didedikasikan untuk aktor manapun karena pelaksanaannya dikendalikan oleh waktu.



Gambar 83. Ragam *Use-Case* di Blok **Seleksi**.

*a. Menyeleksi pendaftar*

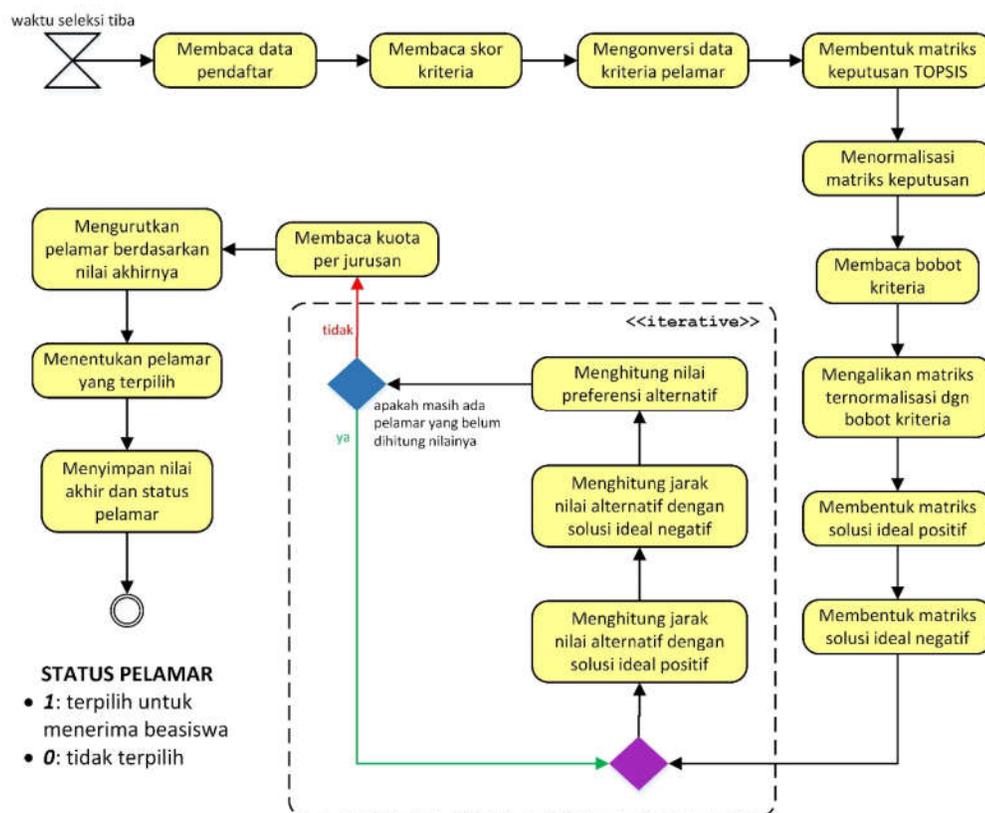
Ketika sampai saatnya seleksi, sistem mengimplementasikan algoritma TOPSIS. Pertama-tama, data segenap pelamar diubah menjadi skor-skor yang setimbang menggunakan metode perbandingan matematis, kemudian disusun dalam bentuk matriks. Khusus untuk kriteria *Prestasi* yang bersifat kualitatif, sebelum konversi dilakukan, tingkatan-tingkatan yang ada diwakilkan dahulu dengan poin-poin tertentu, sebagaimana yang ditunjukkan pada Tabel 12. Total prestasi seorang pendaftar berasal dari penambahan seluruh poin yang dikumpulkan olehnya.

Tabel 12. Daftar Poin untuk Kriteria *Prestasi*

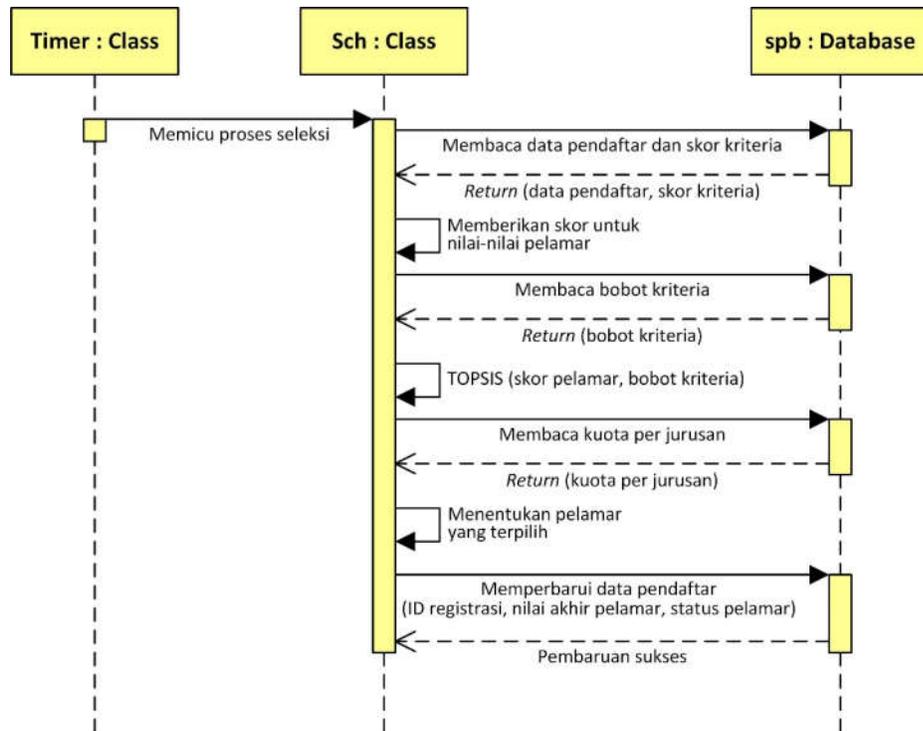
Ruang lingkup		Poin	Titel		Poin
1.	Internasional	24	1.	Juara I	11
4.	Nasional	18	2.	Juara II	9
5.	Regional (gabungan provinsi)	12	3.	Juara III	7
6.	Provinsi	6	4.	Juara Harapan I	5
7.	Kabupaten/kota	1	5.	Juara Harapan II	3
			6.	Juara Harapan III	1

Pada kenyataannya, hanya diambil satu prestasi tertinggi untuk mewakili raihan *Prestasi* pelamar; begitu juga dengan *Riwayat Berorganisasi*. Sistem melakukan yang lebih baik lagi dengan menghitung semuanya, memakai kaidah yang dimuat oleh Tabel 12. Usaha seseorang untuk menyabet sekian titel juara dalam sejumlah kompetisi Provinsi mungkin saja sama dengan jerih payah individu yang memiliki sebuah sertifikat berskala nasional. Oleh karena itu, tak ada alasan logis untuk mempertimbangkan satu, sementara yang lain dinafikan.

Gambar 84 dan Gambar 85 memperlihatkan *activity* dan *sequence diagram* untuk fungsi seleksi pendaftar.



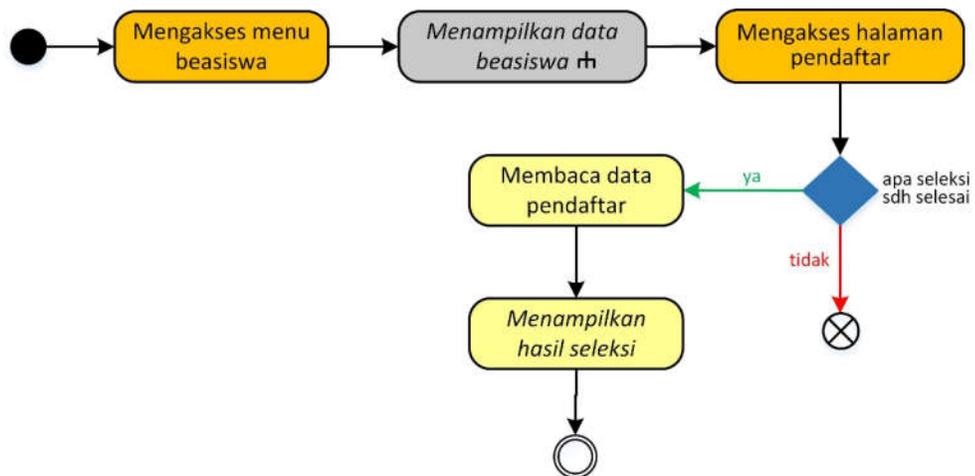
Gambar 84. *Activity Diagram* Fungsi Seleksi Pendaftar.



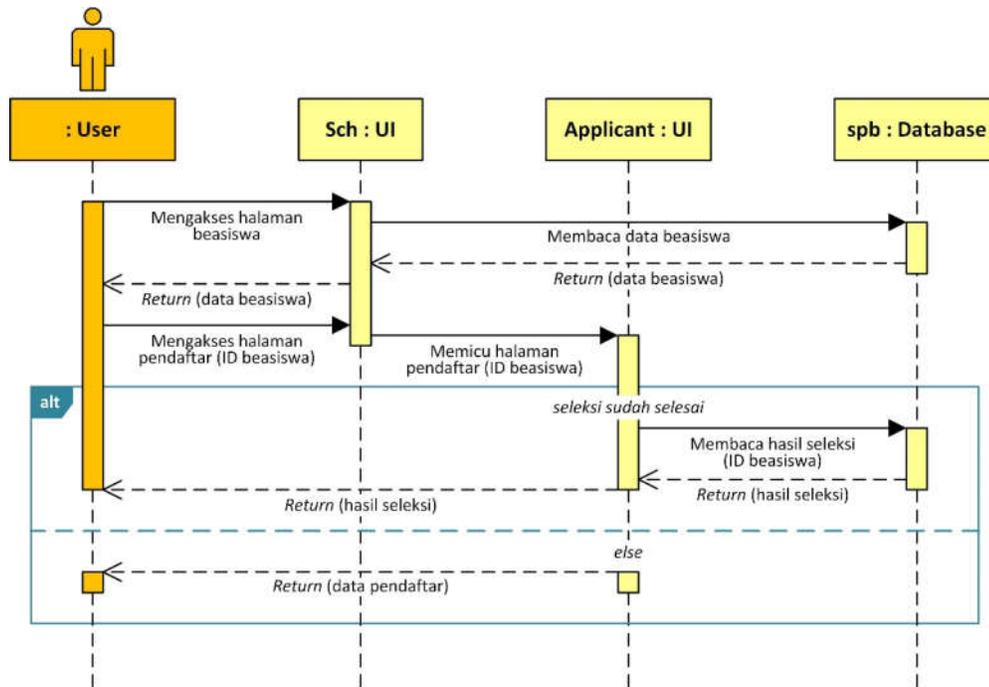
Gambar 85. *Sequence Diagram* Fungsi Seleksi Pendaftar.

b. *Melihat hasil seleksi*

Gambar 86 dan Gambar 87 mendeskripsikan fungsi pembacaan hasil seleksi.



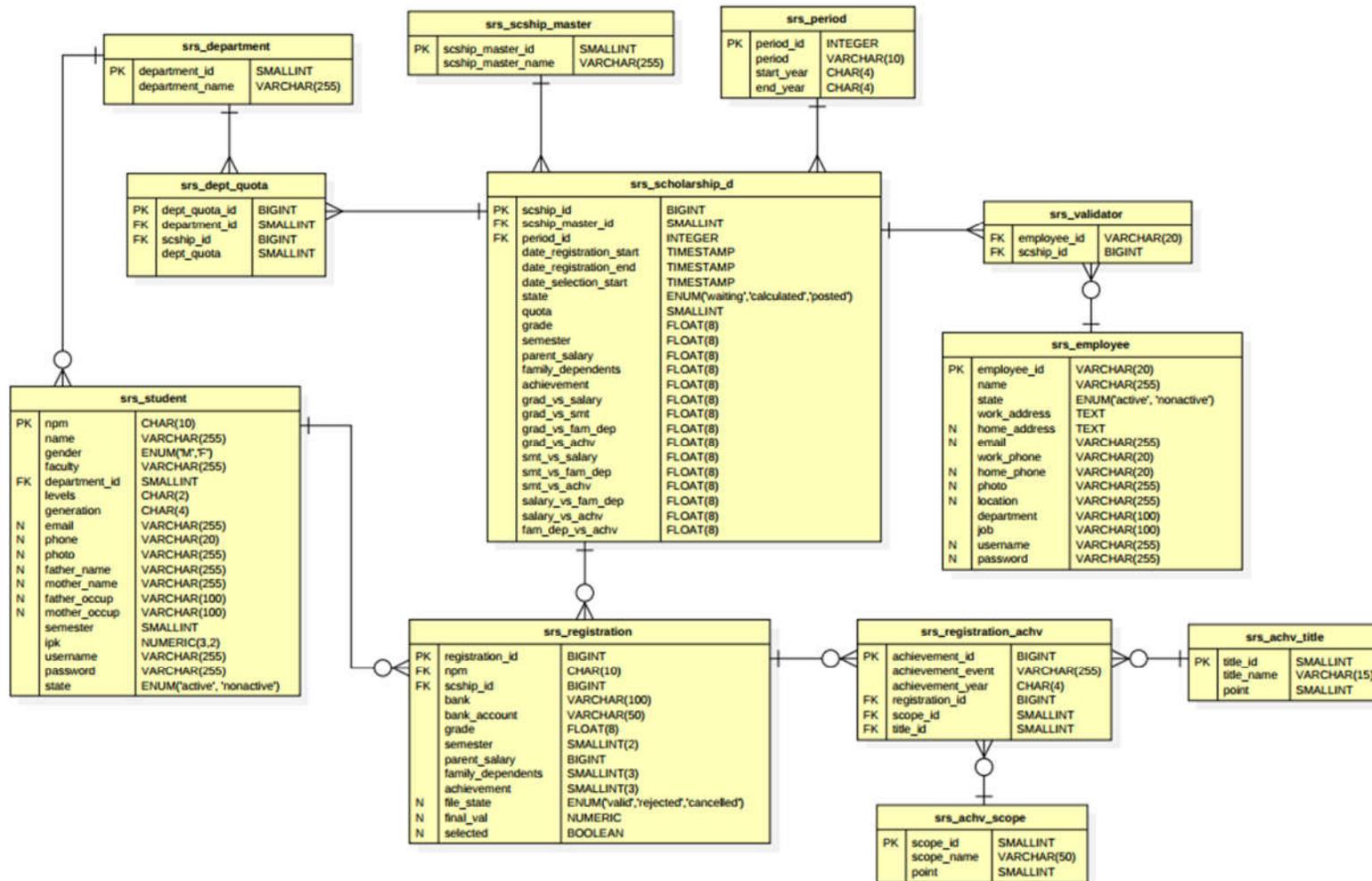
Gambar 86. *Activity Diagram* Fungsi Pembacaan Hasil Seleksi.



Gambar 87. *Sequence Diagram* Fungsi Pembacaan Hasil Seleksi.

### 3.3.3.11. Entity Relationship Diagram

Mengacu pada sekian kebutuhan fungsional yang sudah dirumuskan, dapat didesain sebuah *entity relationship diagram* (ERD) yang mendokumentasikan struktur data sistem. Sesuai dengan namanya, ERD adalah representasi grafis dari objek-objek yang berkepentingan, yang disebut *entitas*, dan pertalian di antara mereka (Mannino, 2005). Gambar 88 menampilkan ERD dari Sistem Manajemen Beasiswa, dimana entitas dilambangkan dengan tabel yang kolom-kolomnya diistilahkan sebagai *atribut*. *Flag* PK mengisyaratkan bahwa atribut yang bersangkutan merupakan *primary key*, yaitu pengenalan unik bagi suatu entitas, sedangkan *flag* N menandakan bahwa atribut itu boleh NULL.



Gambar 88. Entity Relationship Diagram (ERD) Sistem Manajemen Beasiswa

Atribut-atribut yang menyanggah *flag* FK, akronim dari *foreign key*, mengindikasikan simbol hubungan antar entitas. `srs_student.department_id` misalnya, eksis selaku penjelmaan relasi *many-to-one* dari `srs_student` dan `srs_department`. Seorang mahasiswa/i bernaung di bawah tepat satu jurusan, sementara jurusan yang dimaksud dihuni oleh banyak mahasiswa. Menurut logika ini, tabel `srs_student` perlu mengakomodasi 1 kolom ekstra untuk mencatat jurusan masing-masing mahasiswa.

Sebelas relasi lainnya diterangkan melalui paparan berikut:

1. Suatu beasiswa digelar secara anual, sedangkan dalam 1 tahun ditawarkan beberapa beasiswa. Relasi *many-to-many* antara entitas **beasiswa** dan **periode**, yang dicerminkan berturut-turut oleh `srs_scship_master` dan `srs_period`, melahirkan tabel `srs_scholarship_d`. Tabel baru ini memuat spesifikasi semua beasiswa yang hadir dalam tiap kurunnya, juga *identifier* entitas-entitas “induk”-nya, yakni `scship_master_id` dan `period_id`.
2. Dalam penyelenggaraannya, sebuah beasiswa selalu mengalokasikan kuota penerima per jurusan. Ikatan yang terjalin lagi-lagi *many-to-many*, melibatkan `srs_department` dan `srs_scholarship_d`, memunculkan `srs_dept_quota`. Keduanya menyumbangkan *key* `department_id` dan `scship_id`.
3. Asosiasi *many-to-many* juga menyambungkan `srs_scholarship_d` dan `srs_employee`; berpijak pada alasan yang telah dibahas di subbab 3.3.3.6. Akibatnya, timbul `srs_validator`, entitas yang dikatakan *lemah* karena tanpa *primary key* (Mannino, 2005).

4. Sepanjang masa studinya, mahasiswa/i boleh mendaftarkan dirinya di sejumlah beasiswa, dimana satu beasiswa bisa dilamar oleh banyak individu. Manifestasi dari premis tadi ialah `srs_registration`, yang mempertautkan `srs_student` dan `srs_scholarship_d`.
5. Seorang pendaftar sangat mungkin mendulang prestasi yang berlimpah. Situasi begini jelas tak tertampung dalam sebaris *record* di `srs_registration`. Itu berarti entitas lain harus disiapkan untuk rincinya; yang mana menjadi sebab adanya `srs_registration_achv`. Mengenai `srs_achv_scope` dan `srs_achv_title`, keduanya bertindak sebagai perekam poin-poin cakupan dan titel prestasi yang sebelumnya diekspos di Tabel 12. Nominal angka yang dikoleksi pelamar disimpan pada kolom `srs_registration.achievement`.

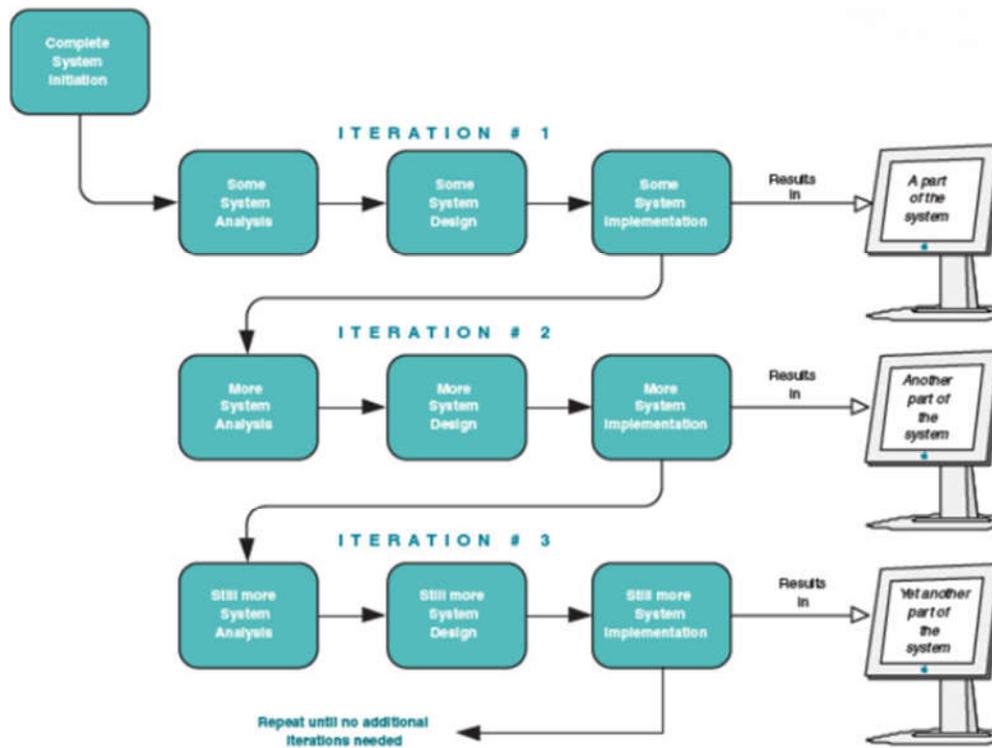
Agar sistem tetap terjaga dalam ruang lingkungannya, *field-field* di bawah ini, yang seharusnya merupakan *foreign key*, diberi tipe *string*:

<b>srs_student</b>	<b>srs_employee</b>
faculty	department
levels	job
father_occup	
mother_occup	

#### 3.3.4. Pengembangan sistem

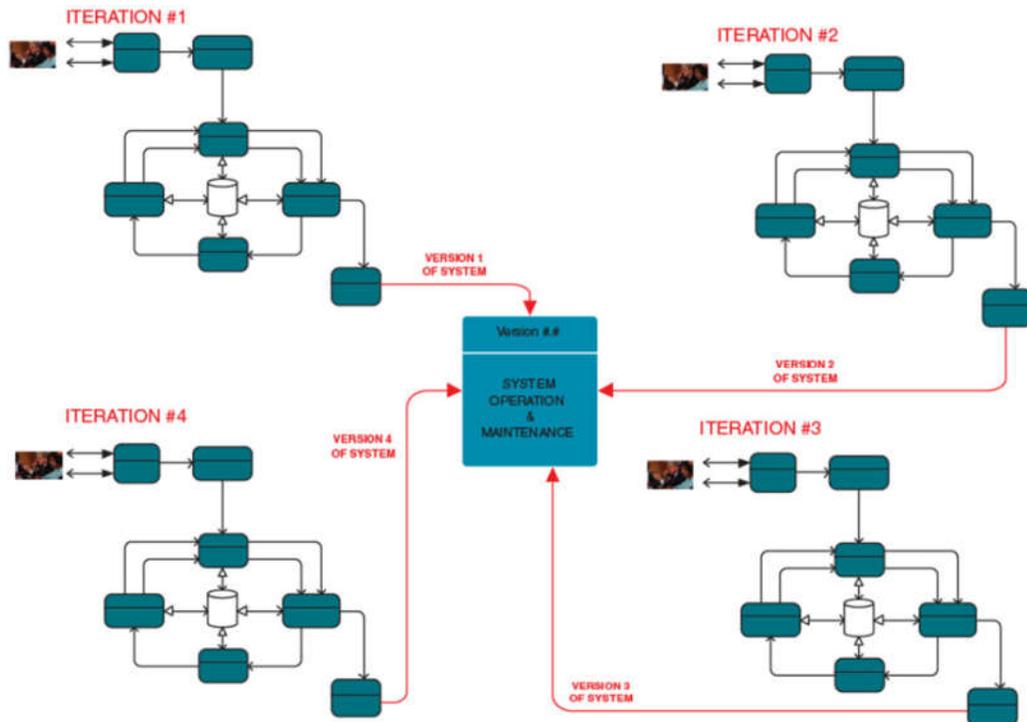
Kegiatan konstruksi sistem mengikuti metode pengembangan iteratif/inkremental. Menurut *Centers for Medicare & Medicaid Services* (2008), kasus yang paling cocok untuk diterapkan metode iteratif adalah sistem informasi berbasis web. Kontras dengan pendekatan *waterfall* yang mengharuskan setiap fase disempurnakan dulu sebelum meneruskan ke fase selanjutnya, metode ini justru memperbolehkan analisis, desain, dan implementasi sebagian kecil dari sistem. Sesudahnya, ketiga tahap barusan diulangi lagi guna mewujudkan potongan-

potongan sistem yang lain. Hingga sistem selesai diproduksi, terjadi puluhan atau bahkan ratusan iterasi, seperti yang diilustrasikan oleh Gambar 89.



Gambar 89. Metode Pengembangan Iteratif (Whitten & Bentley, 2007).

Besar iterasi yang telah dilalui ini dicerminkan oleh nomor versi sistem, yang mana dipergakan pada Gambar 90.



Gambar 90. Strategi Implementasi Iteratif (Whitten & Bentley, 2007).

Sistem Manajemen Beasiswa dikodekan di atas Odoo 10 *Community Edition*. Odoo merupakan platform *open source* bagi aplikasi-aplikasi bisnis berbasis web yang didistribusikan di bawah lisensi AGPL (library/odoo, tanpa tanggal; Reis, 2016). Odoo ditulis dengan bahasa Python serta disokong oleh DBMS Postgre SQL.

Fungsi utama Odoo adalah *Enterprise Resource Planning* (ERP). Oleh karena itu, dalam setiap rilisnya, selalu diikutsertakan seperangkat *software* terintegrasi yang tugas-tugasnya mencakup seluruh area bisnis, di antaranya *Sales*, *CRM*, *Accounting*, *Human Resources*, *Purchase*, *Inventory*, *Email Marketing*, *e-Commerce*, *Survey*, dan *Website Builder*. Fungsi lainnya, yang akan diberdayakan dalam penelitian, adalah *framework* pemrograman. Menggunakan fitur ini,

*developer* dapat memodifikasi aplikasi yang tersedia dan menciptakan aplikasi baru.

### 3.3.5. Pengujian sistem

Pengujian diselenggarakan ketika sistem sedang diprogram untuk membuktikan bahwa sistem bekerja sesuai dengan *use-case-use-case*-nya. Kategori tes yang diimplementasikan adalah *black box* tipe *functional*. Memanfaatkan teknik *Equivalence Class Partitioning*, domain *input* dibagi ke dalam beberapa kelas, yang mana dari tiap-tiapnya diambil suatu nilai untuk dicobakan ke sistem. Bila nihil kekurangan yang terdeteksi, maka nilai-nilai lain dari kelas tersebut tidak akan menangkap apapun, sehingga sistem dinyatakan lulus. Sisi positif dari cara semacam ini ialah menghemat jumlah kasus uji yang mesti dilaksanakan. (Burnstein, 2003)

Meskipun begitu, demi simplifikasi lebih lanjut, rencana *testing* yang disusun tidak selalu mengecek semua *field*. Pada formulir pembuatan atau pendaftaran beasiswa yang ramai dengan *field*, pengujian sekumpulan *field* yang bersifat sama dilimpahkan ke salah satu di antara mereka. Sampelnya adalah skenario 94, dimana pemeriksaan medan wajib terhadap isian **NULL** didelegasikan pada *field* **tanggal seleksi**. Jika sistem mengeluarkan pesan *error* yang diharapkan, maka sebelas *field* sisanya dianggap membuahkan efek serupa.

Kasus-kasus uji sistem dilampirkan pada Tabel 24. Di sana termuat 283 kasus uji yang merupakan pengembangan dari 213 kelas.

Selain kasus-kasus di atas, perlu juga dilakukan komparasi antara bobot-bobot kriteria hasil estimasi sistem dan hasil hitung manual. Pasalnya, algoritma AHP yang dipraktikkan sistem tidak mencontoh teori perhitungan yang baku, tetapi merefleksikan kaidah aproksimasi. Tiga problem di bawah ini diusulkan untuk memeriksa seberapa dekat *output* sistem dengan nilai sebenarnya. Masing-masing masalah berbentuk matriks bujur sangkar berordo lima, mengingat ada lima kriteria yang disorot dalam penelitian.

$$\begin{array}{ccc}
 \text{Matriks I} & \text{Matriks II} & \text{Matriks III} \\
 \left[ \begin{array}{ccccc} 1 & 5 & 9 & 9 & 3 \\ 1/5 & 1 & 4 & 4 & 1/3 \\ 1/9 & 1/4 & 1 & 1 & 1/6 \\ 1/9 & 1/4 & 1 & 1 & 1/6 \\ 1/3 & 3 & 6 & 6 & 1 \end{array} \right] & \left[ \begin{array}{ccccc} 1 & 1/5 & 1/8 & 1/7 & 2 \\ 5 & 1 & 1/3 & 1/2 & 7 \\ 8 & 3 & 1 & 2 & 9 \\ 7 & 2 & 1/2 & 1 & 9 \\ 1/2 & 1/7 & 1/9 & 1/9 & 1 \end{array} \right] & \left[ \begin{array}{ccccc} 1 & 1/4 & 2/3 & 2/3 & 5/2 \\ 4 & 1 & 3 & 3 & 6 \\ 3/2 & 1/3 & 1 & 1 & 4 \\ 3/2 & 1/3 & 1 & 1 & 4 \\ 2/5 & 1/6 & 1/4 & 1/4 & 1 \end{array} \right]
 \end{array}$$

### 3.3.6. Penarikan simpulan

Simpulan merupakan jawaban atas rumusan masalah yang telah didefinisikan di awal riset (lihat halaman 4). Informasi ini diperoleh dari analisis hasil uji sistem.

### 3.3.7. Publikasi penelitian

Sepanjang perjalanannya, seluk-beluk penelitian dibukukan kemudian dicetak. Salinannya dihibahkan kepada universitas untuk disebarakan di perpustakaan dan diunggah ke repositori jurnal.

## V. SIMPULAN DAN SARAN

### 5.1. Simpulan

Berdasarkan hasil dari pengujian sistem yang telah dilakukan, dapat ditarik simpulan sebagai berikut:

1. Sistem Informasi Manajemen Beasiswa FMIPA Universitas Lampung berhasil dikembangkan dengan memanfaatkan *framework* Odoo.
2. Sistem Informasi Manajemen Beasiswa FMIPA Universitas Lampung sukses mencapai tujuannya, yakni merekam data pelamar beasiswa dan menyeleksi secara otomatis.
3. Akurasi perhitungan bobot-bobot kriteria oleh sistem terbilang baik.
4. Terdeteksi satu kekurangan sistem, yaitu kriteria-kriteria antara satu beasiswa dengan yang lain diasumsikan sama. Dengan demikian, sistem yang dibangun belum mampu menjawab tantangan yang ada secara ril.

### 5.2. Saran

Poin-poin di bawah ini dapat dipertimbangkan untuk diwujudkan pada penelitian selanjutnya, demi penyempurnaan Sistem Informasi Manajemen Beasiswa:

1. Sistem hendaknya sanggup merekam varian kriteria tiap beasiswa serta menampilkan formulir registrasi yang dinamis (berisikan kriteria-kriteria tadi).
2. Ruang lingkup sistem diperluas hingga mencakup seluruh universitas.

3. Pangkalan data sistem hendaknya diintegrasikan dengan *database* utama universitas menggunakan solusi-solusi teknologi yang beredar supaya konsistensi data selalu terjaga.
4. Basis yang dipakai untuk menetapkan apakah seorang mahasiswa sedang melamar/menerima beasiswa mesti diperkaya lagi. Jadi, faktor-faktor lain dihadirkan di samping tanggal aktif beasiswa, misalnya total dana yang telah disalurkan penyedia beasiswa ke rekening penerima, semester yang kini tengah dijalani, IP semester lalu, dan status aktif mahasiswa yang bersangkutan (terkait cuti/kelulusan).
5. *Experience* pengguna ketika melakukan registrasi dapat diperbaiki dengan menonaktifkan tombol **Buat** pada laman Pendaftaran bagi mahasiswa/i yang sedang melamar/menerima beasiswa; tak perlu repot mengisi borang *online* untuk sekedar mengetahui bahwa dirinya tidak diperkenankan mendaftar. Pola yang sama juga boleh diterapkan pada button **Batal**, **Validasi**, dan **Tolak**.
6. Implementasi fitur rekomendasi dosen Pembimbing Akademik dalam sistem.
7. Sistem belum sanggup membuahakan keputusan ketika beberapa pelamar mendulang nilai akhir yang sama, sehingga dibutuhkan algoritma tambahan guna mengantisipasi keadaan ini.
8. Proses seleksi yang diselenggarakan oleh sistem harus mencakup para peserta perpanjangan, tidak hanya para pendaftar baru.
9. Aplikasi seluler untuk sistem juga perlu dikembangkan agar sistem mampu mengakomodasi para pengguna perangkat *mobile*. Konsekuensi dari butir ini ialah API (*Application Programming Interface*) sistem harus dibuat, yang berperan sebagai jembatan antara sumber daya sistem dengan aplikasi.

10. Uji-uji non-fungsional hendaknya diselenggarakan untuk memaksimalkan performa, ketahanan, fleksibilitas, dan skalabilitas sistem.
11. Ada baiknya disusun sebuah mekanisme, dimana validasi kualifikasi para pelamar beasiswa dilakukan langsung dari hasil pindai (*scan*) berkas-berkas terkait yang diunggah ke media penyimpanan sistem.
12. Selain seleksi penerima beasiswa, AHP dan TOPSIS bisa diberdayakan untuk menyelesaikan masalah-masalah lain yang berkenaan dengan pendidikan, seperti penentuan biaya UKT (Uang Kuliah Tunggal), pemilihan Mahasiswa Berprestasi, atau perekrutan pegawai.

# DAFTAR PUSTAKA

## DAFTAR PUSTAKA

- Alonso, J. A., & Lamata, M. T. (2006). Consistency in The Analytic Hierarchy Process: A New Approach. *International Journal of Uncertainty, Fuzziness, and Knowledge-Based Systems*, 14(4), 445-459.
- Bhushan, N., & Rai, K. (2004). Strategic Decision Making: Applying The Analytic Hierarchy Process. (1). London: Springer-Verlag. doi:10.1007/b97668
- Bidikmisi. (2018). *Apa saja syarat pendaftaran bidikmisi 2018?* Diambil dari Bidikmisi - Beranda - Ristekdikti: <https://bidikmisi.belmawa.ristekdikti.go.id/petunjuk/2>
- Burnstein, I. (2003). Practical Software Testing: A Process-Oriented Approach. (1). New York: Springer-Verlag. doi:10.1007/b97392
- Centers for Medicare & Medicaid Services. (2008). *Selecting A Development Approach*. Diambil dari XLC Process Overview: <https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/XLC/Downloads/SelectingDevelopmentApproach.pdf>
- Curry, E., & Grace, P. (2008). Flexible-Self Management Using the Model-View-Controller Pattern. *IEEE Software*, 25(3), 84-90. doi:10.1109/MS.2008.60
- Druzdzel, M. J., & Flynn, R. R. (2002). Decision Support Systems. Pittsburgh. Diambil dari <https://www.pitt.edu/~druzdzel/psfiles/dss.pdf>
- Endah, D., & Raharjo, H. (2003). Evaluating Relationship of Consistency Ratio and Number of Alternatives on Rank Reversal. *Proceedings of the 7th International Symposium on The AHP (2003)*, (hal. 207-216). Bandung.
- Forman, E. H., & Gass, S. I. (2001). The Analytic Hierarchy Process – An Exposition. *Operations Research*, 49(4), 469-486. doi:10.1287/opre.49.4.469.11231

- Hamilton, K., & Miles, R. (2006). *Learning UML 2.0*. O'Reilly. Diambil dari [http://portal.unimap.edu.my/portal/page/portal30/Lecturer%20Notes/KEJURUTERAAN\\_KOMPUTER/Semester%202%20Sidang%20Akademik%2020112012/EKT420%20Software%20Engineering/E-Books/UML/Learning%20UML%202.0%20%5B2006%5D.pdf](http://portal.unimap.edu.my/portal/page/portal30/Lecturer%20Notes/KEJURUTERAAN_KOMPUTER/Semester%202%20Sidang%20Akademik%2020112012/EKT420%20Software%20Engineering/E-Books/UML/Learning%20UML%202.0%20%5B2006%5D.pdf)
- Kardi, T. (2006). *Priority Vector*. Diambil dari Kardi Teknomo's Page: <http://people.revoledu.com/kardi/tutorial/AHP/Priority%20Vector.htm>
- Kardi, T. (2011). *Eigenvalue and Eigenvector*. Diambil dari Kardi Teknomo's Page: <http://people.revoledu.com/kardi/tutorial/LinearAlgebra/EigenValueEigenVector.html>
- Kemristekdikti. (2018). *Panduan Beasiswa Peningkatan Prestasi Akademik (PPA) Tahun 2018*. Diambil dari Kopertis III | Laman Resmi Kopertis Wilayah III Jakarta: <http://kopertis3.or.id/v5/wp-content/uploads/Panduan-Beasiswa-PPA-tahun-2018.pdf>
- Kunz, J. (2010, Maret). *The Analytic Hierarchy Process (AHP)*. Idaho. Diambil dari <http://www.researchgate.net>
- Kurniawan, D. (2018a). *Beasiswa Bank Indonesia tahun 2018*. Diambil dari Jurusan Ilmu Komputer FMIPA: <http://ilkom.unila.ac.id/?p=3028>
- Kurniawan, D. (2018b). *Form Pendaftaran Beasiswa PPA 2018*. Diambil dari Jurusan Ilmu Komputer FMIPA: [http://ilkom.unila.ac.id/?page\\_id=3108](http://ilkom.unila.ac.id/?page_id=3108)
- Kurniawan, D. (2018c). *Seleksi Calon Penerima Beasiswa PPA Tahun 2018*. Diambil dari Jurusan Ilmu Komputer FMIPA: <http://ilkom.unila.ac.id/?p=3106>
- Kurniawan, D. (2018d). *Pencairan Penerima Beasiswa Bank Indonesia*. Diambil dari Jurusan Ilmu Komputer FMIPA: <http://ilkom.unila.ac.id/?p=3290>
- library/odoo*. (tanpa tanggal). Diambil dari Docker Hub: <https://hub.docker.com/r/library/odoo/>
- Lucid Software Inc. (2018). *Flowchart Symbols and Notation*. Diambil dari Lucidchart: <https://www.lucidchart.com/pages/flowchart-symbols-meaning-explained>

- Mannino, M. V. (2005). *Database Design, Application Development, and Administration*. (3). New York: McGraw-Hill, Inc.
- Manurung, P. (2011). *Sistem Pendukung Keputusan Seleksi Penerimaan Beasiswa dengan Metode AHP dan TOPSIS (Studi Kasus: FMIPA USU)*. Universitas Sumatera Utara, Ilmu Komputer, Medan. Diambil dari USU Repository: <http://repository.usu.ac.id/handle/123456789/21657>
- Maryani, E., & Riyanto, S. (2017). *Pengelolaan Beasiswa di FMIPA Universitas Lampung*. (F. Almanian, Pewawancara) Bandar Lampung.
- PPMB S1 Universitas Lampung. (2018). *Berkas yang Harus Dibawa pada Waktu Wawancara*. Diambil dari PMPAP: [http://pmpap.unila.ac.id/berkas\\_2018.pdf](http://pmpap.unila.ac.id/berkas_2018.pdf)
- Reis, D. (2016). *Odoo 10 Development Essentials*. Birmingham, United Kingdom: Packt Publishing.
- Saaty, T. L. (1987). The Analytic Hierarchy Process - What It is and How It is Used. *Mathematical Modelling*, 9(3-5), 161-176.
- Saaty, T. L. (1990). How to make a decision: The Analytic Hierarchy Process. *European Journal of Operational Research*, 48, 9-26.
- Saaty, T. L. (1991). Some Mathematical Concepts of The Analytic Hierarchy Process. *Behaviormetrika*, 29, 1-9.
- Saaty, T. L. (2003). Decision Making with The AHP: Why is The Principal Eigenvector Necessary. *European Journal of Operational Research*, 145, 85-91.
- Saaty, T. L. (2005). The Analytic Hierarchy And Analytic Network Processes For The Measurement of Intangible Criteria And For Decision Making. Dalam J. Figueira, S. Greco, & M. Ehrgott, *Multiple Criteria Decision Analysis: State of The Art Surveys* (hal. 345-405). doi:10.1007/0-387-23081-5\_9
- Saaty, T. L. (2008a). Decision Making with The Analytic Hierarchy Process. *International Journal of Services Sciences*, 1(1), 83-98. doi:10.1504/IJSSci.2008.01759

- Saaty, T. L. (2008b). Relative Measurement and Its Generalization in Decision Making: Why Pairwise Comparisons are Central in Mathematics for The Measurement of Intangible Factors (The Analytic Hierarchy/Network Process). *RACSAM - Revista de la Real Academia de Ciencias Exactas, Fisicas y Naturales. Serie A. Matematicas*, 102(2), 251–318. doi:10.1007/BF03191825
- Sakethi, D. (2015). *Pemasukan Data Pendaftar Beasiswa PPA dan PA-BBM 2015*. Diambil dari Jurusan Ilmu Komputer FMIPA: <http://ilkom.unila.ac.id/?p=1298>
- Sari, N. (2015). Pengembangan Sistem Informasi Pendaftaran Beasiswa Peningkatan Prestasi Akademik (PPA) di Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung dengan Menerapkan Algoritma Sorting Quick Sort dan Selection Sort. Skripsi, Universitas Lampung, Ilmu Komputer, Bandar Lampung. Diambil dari <http://digilib.unila.ac.id/id/eprint/11877>
- Security in Odoo*. (2017). Diambil dari Odoo 10.0 Documentation: <https://www.odoo.com/documentation/10.0/reference/security.html>
- Serifi, V., Randic, S., & Dasic, P. (2007). Examples of Software Tools for Decision Support System (DSS) Based on Modern Technologies. *Proceedings of the International Conference of Quality, Management, Environment, and Engineering (ICQME-2007)*, (hal. 283-291). Budva.
- Shalehah, A. (2014). Penerapan TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) untuk Seleksi Penerimaan Beasiswa (Studi Kasus Pendaftar Beasiswa dari Fakultas MIPA di Universitas Brawijaya). *Jurnal Mahasiswa Statistik Universitas Brawijaya*, 2(5), 341-344.
- Shin, T., Kim, C.-B., Ahn, Y.-H., Kim, H.-Y., Cha, B. H., Uh, Y., . . . Go, U.-Y. (2009). The Comparative Evaluation of Expanded National Immunization Policies in Korea using An Analytic Hierarchy Process. *Vaccine*, 27, 792-802.
- Sinaga, J. (2010). Penerapan Analytic Hierarchy Process (AHP) dalam Pemilihan Perusahaan Badan Usaha Milik Negara (BUMN) sebagai Tempat Kerja Mahasiswa Universitas Sumatera Utara (USU). Universitas Sumatera Utara, Ilmu Komputer, Medan. Diambil dari USU Repository: <http://repository.usu.ac.id/handle/123456789/14119>

- Stair, R. M., & Reynolds, G. W. (2012). *Fundamentals of Information Systems*. (6). Course Technology, Cengage Learning.
- Vargas, R. V. (2010). Using The Analytic Hierarchy Process (AHP) to Select and Prioritize Projects in A Portfolio. Diambil dari ResearchGate: <http://www.researchgate.net>
- Velasquez, M., & Hester, P. T. (2013). An Analysis of Multi-Criteria Decision Making Methods. *International Journal of Operations Research*, 10(2), 56-66.
- Whitten, J. L., & Bentley, L. D. (2007). *Systems Analysis and Design Methods*. (7). New York: McGraw Hill, Inc.
- Wimatsari, G. A., Putra, I. K., & Buana, P. W. (2013). Multi-Attribute Decision Making Scholarship Selection Using A Modified Fuzzy TOPSIS. *International Journal of Computer Science*, 10(1), 309-317.
- Wright, A., & Sittig, D. F. (2008). A Framework and Model for Evaluating Clinical Decision Support Architectures. *Journal of Biomedical Informatics*, 41, 982-990. doi:10.1016/j.jbi.2008.03.009
- Y.I.E., B., Soebroto, A. A., & Regasari, R. (2013). Sistem Pendukung Keputusan Penentuan Kelayakan Pengisian Bibit Ayam Broiler di Kandang Peternak Menggunakan Metode AHP dan TOPSIS. *Repositori Jurnal Mahasiswa PTHK Universitas Brawijaya*, 2(10).
- Yunita. (2015). Pengembangan Sistem Informasi Seleksi Penerimaan Beasiswa Bantuan Belajar Mahasiswa (BBM) di Jurusan Ilmu Komputer FMIPA Universitas Lampung Menerapkan Metode Merge Sort dan Insertion Sort. Skripsi, Universitas Lampung, Ilmu Komputer, Bandar Lampung. Diambil dari <http://digilib.unila.ac.id/id/eprint/12896>
- Yusuf, A. A., Koniyo, M. H., & Novian, D. (2013). Analisis Perbandingan Metode Gabungan AHP dan TOPSIS dengan Metode TOPSIS. *KIM Fakultas Teknik*, 1(1). Diambil dari <http://kim.ung.ac.id>
- Zhang, S. X., & Babovic, V. (2011). An Evolutionary Real Options Framework for The Design and Management of Projects and Systems with Complex Real Options and Exercising Conditions. *Decision Support Systems*, 51, 119-129. doi:10.1016/j.dss.2010.12.001