

**PENGEMBANGAN SISTEM INFORMASI BEASISWA MENGGUNAKAN  
METODE ITERATIF (STUDI KASUS: UNIVERSITAS BRAWIJAYA)**

**SKRIPSI**

Untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:

**DIONYSIUS BRIANANDA Y.**

**NIM: 125150400111041**



**PROGRAM STUDI SISTEM INFORMASI  
JURUSAN SISTEM INFORMASI  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA  
MALANG  
2016**

## PENGESAHAN

PENGEMBANGAN SISTEM INFORMASI BEASISWA MENGGUNAKAN METODE  
ITERATIF (STUDI KASUS: UNIVERSITAS BRAWIJAYA)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun Oleh :

DIONYSIUS BRIANANDA Y

NIM: 125150400111041

Skripsi ini telah diuji dan dinyatakan lulus pada  
12 Juni 2016

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Ismiarta Aknuranda, S.T., M.Sc., Ph.D.

NIK:2010067407191001

Satrio Agung W., S.Kom, M.Kom

NIK:198605212012121001

Mengetahui  
Ketua Jurusan Sistem Informasi

Herman Tolle, Dr. Eng., S.T, M.T

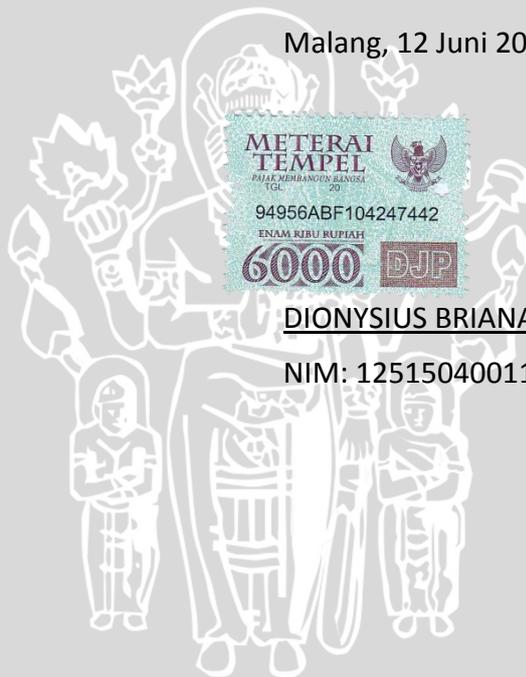
NIP: 197408232000121001

## PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 12 Juni 2016



DIONYSIUS BRIANANDA Y

NIM: 125150400111041

## KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadirat Tuhan, karena dengan rahmat, dan karunia-Nya penulis dapat menyelesaikan skripsi dengan judul **“Pengembangan Sistem Informasi Basiswa menggunakan metode iteratif studi kasus universitas brawijaya”**. Skripsi ini disusun sebagai salah satu persyaratan untuk memperoleh gelar Sarjana Komputer di Program Studi Informatika /Ilmu Komputer, Fakultas Ilmu Komputer Universitas Brawijaya. Melalui pengantar ini penulis mengucapkan banyak terima kasih karena dalam penulisan dan penyusunan laporan skripsi ini penulis mendapat bantuan dan dorongan baik secara langsung maupun tidak langsung dari berbagai pihak diantaranya:

1. Bapak Ismiarta Aknuranda, S.T., M.Sc., Ph.D., selaku dosen pembimbing I dan Bapak Satrio Agung W., S.Kom, M.Kom, selaku dosen pembimbing II yang telah sabar memberikan bimbingan, arahan, ilmu dan saran untuk laporan skripsi ini.
2. Orang tua atas segala dukungan, kasih sayang, perhatian dan kesabarannya, serta yang senantiasa selalu memberikan doa dan semangat demi terselesaikannya skripsi ini.
3. Bapak dan Ibu dosen serta seluruh staf Fakultas Ilmu Komputer (FILKOM) Universitas Brawijaya yang telah memberikan ilmunya serta arahan selama masa perkuliahan.
4. Andriani, Cicha, Voni, Gilrandy, Madés, Dimas, Faishal, Davi dan Teman-teman seperjuangan FILKOM angkatan 2012 yang tidak bisa disebutkan satu persatu yang ikut mendukung dan menyemangati selama proses pengerjaan ini.
5. Seluruh Keluarga Besar Fakultas Ilmu Komputer yang telah memberikan tempat untuk kuliah dan mencari pengalaman di Fakultas Ilmu Komputer Universitas Brawijaya.
6. Semua pihak yang tidak dapat penulis sebutkan satu per satu yang terlibat baik secara langsung maupun tidak langsung demi terselesaikannya skripsi ini. Semoga jasa dan amal baik mendapatkan balasan dari Tuhan YME.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna dan tidak terlepas dari kekurangan dan kesalahan karena keterbatasan materi dan pengetahuan yang dimiliki penulis. Oleh karena itu, segala kritik dan saran yang bersifat membangun penulis harapkan untuk penyempurnaan skripsi ini. Penulis juga berharap mudah-mudahan skripsi ini dapat bermanfaat bagi semua pihak.

Malang, 1 Juni 2016

Dionysius Briananda Y

## ABSTRAK

Pada setiap lembaga pendidikan khususnya universitas banyak sekali beasiswa yang ditawarkan. Banyaknya mahasiswa yang membutuhkan informasi beasiswa menjadi syarat sendiri bahwa tiap lembaga pendidikan harus menyediakan informasi beasiswa. Universitas Brawijaya (UB) memiliki situs web beasiswa yang hanya berisi informasi beasiswa, sedangkan untuk pendaftaran dilakukan secara manual yaitu mahasiswa pergi mengumpulkan berkas dan menyerahkan ke lembaga kemahasiswaan. Pencatatan penerima beasiswa dilakukan juga masih manual. Berdasarkan kondisi diatas dibutuhkan suatu sistem informasi beasiswa untuk UB yang tidak hanya menampilkan informasi namun juga menyediakan pendaftaran secara online. Tujuan penelitian ini adalah untuk mengetahui persyaratan fungsional dan non-fungsional , merancang, mengimplementasikan, dan menguji sistem informasi beasiswa UB. Pengembangan sistem yang dilakukan menggunakan metode iteratif pada tahap analisis, perancangan dan implementasi. -Hasil pengujian fungsionalitas pada sistem ini menunjukkan bahwa fungsi sistem yang diuji telah sesuai dengan spesifikasinya dan hasil pengujian kompatibilitas menunjukkan bahwa sistem mampu berjalan dengan baik pada browser Internet Explorer, Microsoft Edge, Mozilla Firefox, Google Chrome, Safari, Opera, browser iOS, dan browser Android.

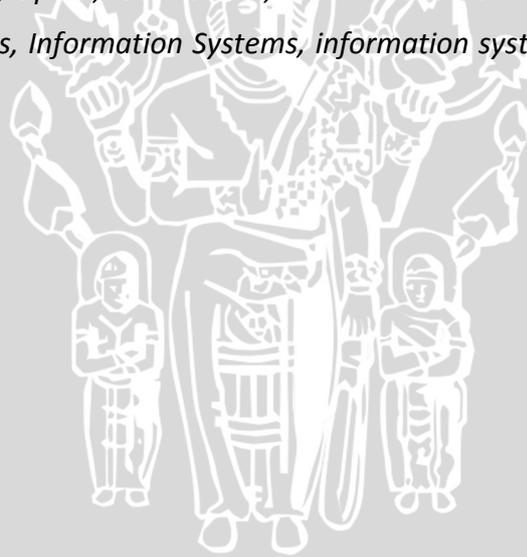
Kata kunci: Sistem informasi beasiswa, pengembangan sistem informasi, pendekatan iteratif.



## ABSTRACT

*In every educational institution in particular the university a lot of scholarships offered. The number of students who need information about the program is the requirement itself that each institution should provide information on scholarships. Universitas Brawijaya (UB) has a scholarship web sites that contain only information on scholarships, while registration is done manually, the students went to collect the files and handed to student organizations. Recording awardees do also still manual. Based on the above conditions needed an information system for UB scholarship that not only display information but also provide online registration. The purpose of this study was to determine the functional requirements and non-functional, design, implement, and test system UB scholarship information. Development of the system was performed using iterative methods in analysis, design and implementation. -Search Functionality testing on this system shows that the system functions were tested in accordance with the specifications and compatibility testing results show that the system is able to run well on the Internet Explorer browser, Microsoft Edge, Mozilla Firefox, Google Chrome, Safari, Opera, iOS browser, and Android browser.*

*Keywords: Scholarships, Information Systems, information systems development, an iterative approach.*



## DAFTAR ISI

<i>PENGESAHAN</i> .....	<i>ii</i>
<i>PERNYATAAN ORISINALITAS</i> .....	<i>iii</i>
<i>KATA PENGANTAR</i> .....	<i>iv</i>
<i>ABSTRAK</i> .....	<i>v</i>
<i>ABSTRACT</i> .....	<i>vi</i>
<i>DAFTAR ISI</i> .....	<i>vii</i>
<i>DAFTAR TABEL</i> .....	<i>xi</i>
<i>DAFTAR GAMBAR</i> .....	<i>xiii</i>
<i>DAFTAR LAMPIRAN</i> .....	<i>1</i>
<b>BAB 1 PENDAHULUAN</b> .....	<b>2</b>
1.1 Latar belakang.....	2
1.2 Rumusan masalah.....	3
1.3 Tujuan .....	3
1.4 Manfaat.....	3
1.5 Batasan masalah .....	3
1.6 Sistematika pembahasan.....	4
<b>BAB 2 LANDASAN KEPUSTAKAAN</b> .....	<b>5</b>
2.1 Kajian Pustaka .....	5
2.2 Dasar Teori.....	5
2.2.1 Beasiswa .....	5
2.2.2 Sistem Informasi.....	5
2.2.3 BPMN .....	8
2.2.4 Metode iteratif.....	9
2.2.4 MoSCoW.....	10
2.2.5 <i>Unified Modeling Language (UML)</i> .....	11
2.2.6 Pengujian Sistem.....	17
<b>BAB 3 METODOLOGI</b> .....	<b>19</b>
3.1 Analisis Persyaratan.....	19
3.2 Perancangan .....	20
3.3 Implementasi .....	20

3.4 Evaluasi Pengguna .....	20
3.5 Pengujian .....	20
3.6 Kesimpulan dan Saran .....	20
<b>BAB 4 ANALISIS PERSYARATAN SISTEM .....</b>	<b>21</b>
4.1 Analisis Persyaratan .....	21
4.1.1 Analisis Proses Bisnis .....	21
4.1.2 Proses Bisnis Saat Ini .....	21
4.1.3 Proses Bisnis Usulan .....	22
4.2 Pemetaan Proses Bisnis dengan Persyaratan Fungsional .....	23
4.2.1 Analisis Pemangku Kepentingan dan Pengguna .....	24
4.2.2 Key Stakeholder and <i>User Needs</i> .....	24
4.2.3 Pemetaan Proses Bisnis Terhadap Fitur Produk .....	26
4.2.4 Fitur Produk .....	26
4.2.5 Spesifikasi Persyaratan .....	27
4.3 Persyaratan Naratif .....	29
4.3.1 Deskripsi Aktor .....	30
4.3.2 Spesifikasi <i>Use Case</i> .....	30
4.4 Perancangan Diagram aktivitas .....	42
4.4.1 Mendaftar beasiswa .....	42
4.4.2 Melihat pengumuman .....	43
4.4.3 Menseleksi Partisipan .....	44
4.4.4 Mengelola data <i>user</i> .....	45
4.4.5 Mengelola data beasiswa .....	46
4.4.6 Mengelola data jurusan dan fakultas .....	47
4.4.7 Login .....	48
4.4.8 Mengatur profil .....	49
4.4.9 Membuat laporan .....	50
4.5 Pemodelan kelas .....	51
4.6 Perancangan Interaksi Antar Objek .....	55
4.6.1 Diagram <i>Sequence</i> Mengajukan Beasiswa .....	55
4.6.2 Diagram <i>Sequence</i> Menyeleksi Partisipan .....	57
4.6.3 Diagram <i>Sequence</i> Melihat Pengumuman .....	57



4.6.4 Diagram <i>Sequence</i> Mengelola <i>User</i> .....	58
4.6.5 Diagram <i>Sequence</i> Mengelola Beasiswa.....	60
4.6.6 Diagram <i>Sequence</i> Mengelola data fakultas dan jurusan.....	61
4.6.7 Diagram <i>Sequence</i> Login .....	63
4.6.8 Diagram <i>Sequence</i> Mengatur Profil .....	64
4.7 Data model.....	66
4.8 Perancangan Antarmuka .....	71
4.8.1 Login .....	71
4.8.2 Home Mahasiswa .....	72
4.8.3 Menu Beasiswa .....	72
4.8.4 halaman <i>Detail</i> Beasiswa Mahasiswa .....	73
4.8.5 Halaman Mendaftar Beasiswa .....	73
4.8.6 Halaman Profil.....	74
4.8.7 Halaman Ubah <i>Password</i> .....	74
4.8.8 Halaman Seleksi Partisipan .....	75
4.8.9 Halaman Membuat Laporan .....	75
4.8.10 Halaman Manajemen Data <i>User</i> .....	76
4.8.11 Halaman Menambah Beasiswa.....	76
<b>BAB 5 IMPLEMENTASI DAN PENGUJIAN .....</b>	<b>78</b>
5.1 Spesifikasi Sistem.....	78
5.2 Implementasi .....	78
5.2.1 Halaman Login.....	78
5.2.2 Halaman Home Mahasiswa .....	79
5.2.3 Halaman Beasiswa Mahasiswa .....	79
5.2.4 Halaman Laporan Mahasiswa .....	80
5.2.5 Halaman Profil Mahasiswa.....	80
5.2.6 Halaman Ubah Profil Mahasiswa .....	81
5.2.7 Halaman Ubah <i>Password</i> .....	81
5.2.8 Halaman <i>Detail</i> Beasiswa Mahasiswa .....	82
5.2.9 Halaman Mengajukan Beasiswa .....	82
5.2.10 Halaman Home Fakultas .....	83
5.2.11 Halaman Beasiswa Fakultas .....	83



5.2.12 Halaman Laporan Fakultas dan Universitas.....	84
5.2.13 Halaman <i>Detail</i> Beasiswa Fakultas.....	84
5.2.14 Halaman Seleksi Partisipan .....	85
5.2.15 Halaman <i>Detail</i> Beasiswa Laporan Fakultas .....	85
5.2.16 Halaman Profil Fakultas .....	86
5.2.17 Halaman Home Universitas.....	87
5.2.18 Halaman Beasiswa Universitas .....	87
5.2.19 Halaman <i>Detail</i> Beasiswa Universitas.....	88
5.2.20 Halaman Manajemen Data <i>User</i> .....	88
5.2.21 Halaman Manajemen Data Fakultas dan Jurusan .....	89
5.2.22 Halaman Tambah Beasiswa .....	90
5.2.23 Halaman Ubah Beasiswa.....	91
5.2.24 Halaman Ubah <i>User</i> .....	92
5.2.25 Halaman Tambah Fakultas.....	93
5.2.26 Halaman Tambah Jurusan.....	93
5.2.27 Halaman Ubah Fakultas .....	94
5.2.28 Halaman Ubah Jurusan .....	94
5.2.29 Halaman Profil Universitas.....	95
5.2.30 Halaman Ubah Profil Universitas.....	95
5.2.31 Halaman <i>Detail</i> Laporan Universitas.....	96
5.3 Rencana Pengujian.....	96
5.4 Pengujian <i>Black-Box</i> .....	98
5.5 Pengujian <i>Compatibility</i> .....	106
5.6 Analisis Pengujian .....	107
<b>BAB 6 PENUTUP .....</b>	<b>109</b>
6.1 Kesimpulan.....	109
6.2 Saran .....	109
<b>DAFTAR PUSTAKA.....</b>	<b>110</b>
<i>Lampiran Proses iterasi</i> .....	112



## DAFTAR TABEL

<b>Tabel 2.1</b> Notasi use case diagram (Satzinger, 2011).....	12
<b>Tabel 2.2</b> Notasi Aktivitas Diagram (Satzinger, 2011).....	13
<b>Tabel 2.3</b> Notasi sequence Diagram (Satzinger, 2011).....	14
<b>Tabel 2.4</b> Notasi class diagram (Satzinger, 2011) .....	15
<b>Tabel 4.1</b> Kelebihan dan Kekurangan Proses Bisnis Saat Ini.....	22
<b>Tabel 4.2</b> Perbandingan sistem informasi saat ini dengan sistem usulan.....	23
<b>Tabel 4.3</b> Aktivitas pada BPMN .....	23
<b>Tabel 4.4</b> Tipe pemangku kepentingan.....	24
<b>Tabel 4.5</b> Key Stakeholder Needs iterasi ke-2 .....	24
<b>Tabel 4.6</b> Fitur Produk.....	26
<b>Tabel 4.7</b> Fitur Produk.....	27
<b>Tabel 4.7</b> Persyaratan Fungsional iterasi ke-2.....	28
<b>Tabel 4.8</b> Persyaratan Non Fungsional iterasi ke-2 .....	29
<b>Tabel 4.9</b> Deskripsi aktor.....	30
<b>Tabel 4.10</b> Penambahan kelas pada iterasi ke 1 .....	52
<b>Tabel 4.11</b> Penambahan kelas pada iterasi ke 2 .....	52
<b>Tabel 4.12</b> Kelas userController .....	53
<b>Tabel 4.13</b> Kelas loginController .....	53
<b>Tabel 4.14</b> Kelas profilController.....	54
<b>Tabel 4.15</b> Kelas reportController .....	54
<b>Tabel 4.16</b> Kelas beasiswaController .....	54
<b>Tabel 4.17</b> Kelas fakultasController .....	54
<b>Tabel 4.18</b> Penjelasan tabel beasiswa .....	67
<b>Tabel 4.19</b> Penjelasan tabel biodata .....	67
<b>Tabel 4.20</b> Penjelasan tabel partisipan.....	68
<b>Tabel 4.21</b> Penjelasan tabel user .....	69
<b>Tabel 4.22</b> Penjelasan tabel pengumuman .....	70
<b>Tabel 4.23</b> Penjelasan tabel distribusi .....	70
<b>Tabel 4.24</b> Penjelasan tabel fakultas .....	70
<b>Tabel 4.25</b> Penjelasan tabel jurusan .....	71

<b>Tabel 4.26</b> Penjelasan tabel pendidikan_terakhir .....	71
<b>Tabel 5.1</b> Rencana pengujian.....	96
<b>Tabel 5.2</b> Hasil pengujian black-box .....	98
<b>Tabel 5.3</b> Kalkulasi hasil pengujian kompabilitas .....	107
<b>Tabel A.1</b> Key Stakeholder Needs iterasi ke-1 .....	112
<b>Tabel A.2</b> Tabel persyaratan fungsional iterasi ke-1 .....	113
<b>Tabel A.3</b> Persyaratan Non Fungsional iterasi ke-1.....	114
<b>Tabel A.4</b> Penambahan Key Stakeholder dan user needs iterasi ke-1.....	116
<b>Tabel A.5</b> Persyaratan fungsional iterasi ke-1 .....	116
<b>Tabel A.5</b> Persyaratan fungsional iterasi ke-1(lanjutan) .....	117
<b>Tabel A.6</b> Penambahan kelas pada iterasi ke 1 .....	117
<b>Tabel A.7</b> Persyaratan Non Fungsional iterasi ke-2.....	118
<b>Tabel A.8</b> Penambahan kelas pada iterasi ke-2.....	118



## DAFTAR GAMBAR

<b>Gambar 2.1</b> Model Sistem Informasi Manajemen (McLeod dan Schell, 2008).....	7
<b>Gambar 2.2</b> Notasi BPMN flow object (OMG, 2013).....	8
<b>Gambar 2.3</b> Notasi BPMN data (OMG, 2013).....	8
<b>Gambar 2.4</b> Notasi BPMN connecting objects (OMG, 2013).....	9
<b>Gambar 2.5</b> Notasi BPMN swimlanes (OMG, 2013).....	9
<b>Gambar 2.6</b> Notasi BPMN artifacts (OMG, 2013).....	9
<b>Gambar 2.7</b> Iterative Process Flow (Pressman, 2010).....	10
<b>Gambar 2.8</b> Simbol class diagram (Satzinger, 2011).....	15
<b>Gambar 2.9</b> Tipe Pengujian Compatibility (Guru99, 2016).....	18
<b>Gambar 3.1</b> Diagram Alur Penelitian.....	19
<b>Gambar 4.1</b> Proses Bisnis Saat Ini.....	21
<b>Gambar 4.2</b> Proses Bisnis.....	22
<b>Gambar 4.3</b> Use Case Diagram.....	30
<b>Gambar 4.4</b> Aktivitas mendaftar beasiswa.....	43
<b>Gambar 4.5</b> Aktivitas Melihat pengumuman.....	44
<b>Gambar 4.6</b> Aktivitas Menseleksi Partisipan.....	45
<b>Gambar 4.7</b> Aktivitas Mengelola data user.....	46
<b>Gambar 4.8</b> Aktivitas Mengelola data beasiswa.....	47
<b>Gambar 4.9</b> Aktivitas Mengelola data jurusan.....	48
<b>Gambar 4.10</b> Login.....	49
<b>Gambar 4.11</b> Aktivitas Mengatur profil.....	50
<b>Gambar 4.12</b> Aktivitas Membuat laporan.....	51
<b>Gambar 4.13</b> Kelas Diagram.....	53
<b>Gambar 4.14</b> Sequence mengajukan beasiswa.....	56
<b>Gambar 4.15</b> Sequence Menyeleksi partisipan.....	57
<b>Gambar 4.16</b> Sequence Melihat pengumuman.....	58
<b>Gambar 4.17</b> Sequence mengelola user.....	59
<b>Gambar 4.18</b> Sequence mengelola beasiswa.....	61
<b>Gambar 4.19</b> Sequence mengelola data jurusan dan fakultas bagian 1.....	62
<b>Gambar 4.20</b> Sequence mengelola data jurusan dan fakultas bagian 2.....	63

<b>Gambar 4.20</b> Sequence login .....	64
<b>Gambar 4.21</b> Sequence mengatur profil.....	65
<b>Gambar 4.22</b> Data model .....	66
<b>Gambar 4.23</b> Perancangan antarmuka login .....	72
<b>Gambar 4.24</b> Perancangan antarmuka home mahasiswa.....	72
<b>Gambar 4.25</b> Perancangan antarmuka menu beasiswa .....	73
<b>Gambar 4.26</b> Perancangan antarmuka detail beasiswa mahasiswa.....	73
<b>Gambar 4.27</b> Perancangan antarmuka mendaftar beasiswa .....	74
<b>Gambar 4.28</b> Perancangan antarmuka halaman profil .....	74
<b>Gambar 4.29</b> Perancangan antarmuka halaman ubah password .....	75
<b>Gambar 4.30</b> Perancangan antarmuka halaman seleksi partisipan .....	75
<b>Gambar 4.31</b> Perancangan antarmuka membuat laporan.....	76
<b>Gambar 4.32</b> Perancangan antarmuka manajemen data user .....	76
<b>Gambar 4.33</b> Perancangan antarmuka menambah beasiswa.....	77
<b>Gambar 5.1</b> Antarmuka login .....	79
<b>Gambar 5.2</b> Antarmuka halaman home mahasiswa.....	79
<b>Gambar 5.3</b> Antarmuka halaman beasiswa mahasiswa .....	80
<b>Gambar 5.4</b> Antarmuka halaman laporan mahasiswa .....	80
<b>Gambar 5.5</b> Antarmuka halaman profil mahasiswa .....	81
<b>Gambar 5.6</b> Antarmuka halaman ubah profil mahasiswa .....	81
<b>Gambar 5.7</b> Antarmuka halaman ubah password .....	82
<b>Gambar 5.8</b> Antarmuka halaman detail beasiswa mahasiswa.....	82
<b>Gambar 5.9</b> Antarmuka halaman mengajukan beasiswa.....	83
<b>Gambar 5.10</b> Antarmuka halaman home fakultas.....	83
<b>Gambar 5.11</b> Antarmuka halaman beasiswa fakultas .....	84
<b>Gambar 5.12</b> Antarmuka halaman laporan fakultas dan universitas.....	84
<b>Gambar 5.13</b> Antarmuka halaman detail beasiswa fakultas.....	85
<b>Gambar 5.14</b> Antarmuka halaman seleksi partisipan .....	85
<b>Gambar 5.15</b> Antarmuka halaman detail beasiswa laporan fakultas bagian 1 ..	86
<b>Gambar 5.16</b> Antarmuka halaman detail beasiswa laporan fakultas bagian 2 ..	86
<b>Gambar 5.17</b> Antarmuka halaman profil fakultas .....	87
<b>Gambar 5.18</b> Antarmuka halaman home universitas .....	87



<b>Gambar 5.19</b> Antarmuka halaman beasiswa universitas .....	88
<b>Gambar 5.20</b> Antarmuka halaman detail beasiswa universitas .....	88
<b>Gambar 5.21</b> Antarmuka halaman manajemen data user .....	89
<b>Gambar 5.22</b> Antarmuka halaman manajemen data fakultas dan jurusan bagian 1.....	89
<b>Gambar 5.23</b> Antarmuka halaman manajemen data fakultas dan jurusan bagian 2.....	90
<b>Gambar 5.24</b> Antarmuka halaman tambah beasiswa .....	91
<b>Gambar 5.25</b> Antarmuka halaman ubah beasiswa.....	92
<b>Gambar 5.26</b> Antarmuka halaman ubah user.....	93
<b>Gambar 5.27</b> Antarmuka halaman tambah fakultas .....	93
<b>Gambar 5.28</b> Antarmuka halaman tambah jurusan .....	94
<b>Gambar 5.29</b> Antarmuka halaman ubah fakultas.....	94
<b>Gambar 5.30</b> Antarmuka halaman ubah jurusan.....	95
<b>Gambar 5.31</b> Antarmuka halaman profil universitas.....	95
<b>Gambar 5.32</b> Antarmuka halaman ubah profil universitas.....	96
<b>Gambar 5.33</b> Antarmuka halaman detail laporan universitas.....	96
<b>Gambar 5.34</b> Hasil pengujian compabilitas menggunakan sortsite .....	107
<b>Gambar A.1</b> Use case.....	115



## DAFTAR LAMPIRAN

<b>A.1 Proses Iterasi ke-1</b> .....	<b>112</b>
<b>A.2 Proses Iterasi ke-2</b> .....	<b>117</b>



## BAB 1 PENDAHULUAN

### 1.1 Latar belakang

Beasiswa merupakan pembiayaan yang bersumber dari perusahaan swasta, kedutaan, universitas, serta lembaga pendidik atau peneliti, atau juga dari kantor tempat bekerja, yang karena prestasinya seorang karyawan (atau seseorang) dapat diberikan kesempatan untuk meningkatkan kapasitas sumber daya manusianya melalui pendidikan (Ananta, 2014).

Pada setiap lembaga pendidikan khususnya universitas banyak sekali beasiswa yang ditawarkan kepada mahasiswa yang berprestasi maupun yang kurang mampu. Untuk mendapatkan beasiswa tersebut siswa harus mengikuti aturan-aturan yang telah ditetapkan. Oleh sebab itu tidak semua yang mendaftarkan diri sebagai calon penerima beasiswa tersebut akan diterima, hanya yang memenuhi kriteria-kriteria saja yang akan memperoleh beasiswa tersebut (Putra dan Hardiyanti, 2011). Selain itu umumnya mahasiswa yang sebelumnya telah mendapatkan beasiswa tidak diperkenankan untuk mengajukan beasiswa yang lain. Dengan demikian, diharapkan pembagian beasiswa dapat sesuai dengan tujuan.

Semakin banyaknya minat untuk mendaftar dalam sebuah kampus juga semakin menyulitkan tim seleksi penerima beasiswa untuk melakukan pengelompokan beasiswa dan seleksi calon penerima beasiswa terbaik, sehingga akan menjadi pekerjaan yang tidak mudah bagi tim seleksi. Bahkan pengaturan data beasiswa dapat menghabiskan waktu berminggu-minggu sebelum akhirnya dilakukan seleksi.

Berbagai segi harus diperhitungkan untuk melakukan seleksi peserta dan meminimalisasi tingkat kesalahan tim seleksi dalam menentukan peserta yang dianggap layak untuk menerima beasiswa. Oleh karena, itu diperlukan sistem yang dapat memberi kemudahan tim seleksi dalam mengelompokkan dan menyeleksi data. Selain itu, tim seleksi beasiswa tidak perlu lagi memilah-milah berkas-berkas beasiswa karena sistem akan memilahnya sendiri sesuai dengan beasiswa yang ditawarkan.

Berdasarkan hasil observasi dan wawancara dengan staf kemahasiswaan dan beberapa mahasiswa, situs web beasiswa.ub.ac.id hanya dapat menampilkan informasi mengenai beasiswa dan fitur login untuk mahasiswa PPA-BBP saja. Sebagai tambahan informasi, situs web beasiswa Universitas Brawijaya (UB) dibandingkan juga dengan situs web beasiswa Universitas Indonesia (UI). Situs web beasiswa UB hanya menampilkan informasi mengenai pengumuman beasiswa dalam bentuk berita, sedangkan untuk melakukan pendaftaran beasiswa masih dilakukan dengan cara konvensional, yaitu dengan penyerahan berkas ke bagian kemahasiswaan. Sedangkan situs web beasiswa UI tidak hanya menampilkan informasi mengenai pengumuman beasiswa, namun mahasiswa

dapat pula melakukan pendaftaran secara online dengan mengunggah berkas-berkas yang dibutuhkan sesuai dengan persyaratan beasiswa yang diminta. Selain itu, mahasiswa juga dapat mengikuti informasi seberapa banyak pendaftar dan seberapa banyak pendaftar yang akan diterima. Selain memudahkan mahasiswa untuk melengkapi berkas dengan cara *online*, memudahkan juga pihak fakultas untuk melakukan pengelompokan data dan seleksi beasiswa.

Berdasarkan masalah di atas dibutuhkan sistem informasi yang memiliki kemampuan untuk dapat menampilkan informasi mengenai beasiswa yang tersedia, pendaftaran secara online dan mempermudah seleksi berkas. Di tugas akhir ini dilakukan pengembangan sistem informasi beasiswa UB yang memiliki kemampuan tersebut. Sistem ini dikembangkan secara iteratif. Metode iteratif merupakan metode pengembangan sistem, yang di dalamnya setiap tahapan/fase pengembangan sistem dilaksanakan secara berulang-ulang sampai mendapatkan hasil yang diinginkan (Neni dan Uluwiya, 2011). Metode iteratif melibatkan calon pengguna untuk setiap iterasinya pada pengembangan sehingga mereka dapat memberikan masukan lebih awal dalam membangun sistem dan hal ini memperkecil risiko kegagalan sistem akibat ketidaksesuaian dengan kebutuhan pengguna (Sommerville, 2010).

## 1.2 Rumusan masalah

1. Apakah persyaratan fungsional dan non-fungsional sistem informasi beasiswa UB yang diusulkan?
2. Bagaimanakah rancangan sistem informasi beasiswa UB tersebut?
3. Bagaimanakah hasil pengujian sistem informasi beasiswa UB tersebut?

## 1.3 Tujuan

1. Menganalisis dan menyusun persyaratan fungsional dan non-fungsional sistem informasi beasiswa UB.
2. Merancang dan mengimplementasikan sistem informasi beasiswa UB dengan menggunakan metode iteratif.
3. Menguji sistem informasi beasiswa UB.

## 1.4 Manfaat

Manfaat penelitian dari penelitian yang dilakukan akan memudahkan mahasiswa dalam melakukan akses informasi beasiswa, selain itu memudahkan juga mahasiswa dalam melakukan pendaftaran beasiswa. Sistem informasi beasiswa tidak hanya memudahkan mahasiswa, namun juga akan memudahkan tim penyeleksi beasiswa.

## 1.5 Batasan masalah

Batasan masalah pada penelitian ini adalah:

1. Studi kasus penelitian dilakukan pada Universitas Brawijaya.

2. Penelitian ini menggunakan metode iteratif.
3. Sistem informasi beasiswa yang dirancang berbasis web.

## 1.6 Sistematika pembahasan

Sistematika penulisan dalam penelitian ini dengan susunan sebagai berikut:

### 1. BAB I PENDAHULUAN

Bab I berisi latar belakang, rumusan masalah, tujuan, manfaat dan batasan masalah, serta sistematika pembahasan.

### 2. BAB II LANDASAN KEPUSTAKAAN

Bab ini akan dibahas tentang teori yang berkaitan dengan topik yang akan dibahas, temuan, dan/atau bahan penelitian sebelumnya yang diperoleh dari berbagai sumber dijadikan dasar teori yang mendasari perancangan, pembuatan prototipe dan evaluasi sistem.

### 3. BAB III METODOLOGI

Bab ini berisi metode-metode atau langkah-langkah yang akan digunakan dalam penelitian.

### 4. BAB IV ANALISIS PERSYARATAN SISTEM

Bab ini membahas mengenai proses bisnis yang sedang berjalan serta proses bisnis usulan, selain itu membahas mengenai spesifikasi persyaratan sistem terkait dengan analisis pemangku kepentingan, persyaratan fungsional dan non fungsional, pemodelan interaksi, pemodelan data dan desain antarmuka.

### 5. BAB V IMPLEMENTASI DAN PENGUJIAN

Bab ini membahas mengenai Spesifikasi sistem, hasil implementasi dan hasil pengujian.

### 6. BAB VI PENUTUP

Bab ini memuat kesimpulan yang diperoleh dari perancangan sistem serta saran yang diberikan untuk pengembangan sistem lebih lanjut.

## BAB 2 LANDASAN KEPUSTAKAAN

### 2.1 Kajian Pustaka

Kajian pustaka pada bab ini membahas mengenai penggunaan metode iteratif dalam pengembangan perangkat lunak di penelitian-penelitian sebelumnya. Pada penelitian mengenai perancangan dan pembangunan sistem informasi keuangan pada sekolah St. Agatha, metode pengembangan sistem yang digunakan adalah metode iterasi, agar sistem yang dibangun dapat dilakukan secara bertahap dan berulang-ulang dan apabila ditemukan kesalahan dapat langsung diperbaiki (Tenardi dan Agustina, 2013).

Penelitian kedua yang dijadikan referensi adalah penelitian mengenai perancangan dan pembuatan sistem informasi pemakaian *sparepart* mesin *packing* pada PT. XYZ. Metode pengembangan sistem yang digunakan adalah metode iteratif. Analisis dengan metode iterasi merupakan proses yang berulang dan berlanjut secara terus menerus dan saling menyusul. Dengan menerapkan sistem informasi yang telah dibangun dapat memenuhi kebutuhan perusahaan (Elizabeth dan Darmawan, 2015).

### 2.2 Dasar Teori

#### 2.2.1 Beasiswa

Beasiswa adalah pemberian berupa bantuan keuangan yang diberikan kepada perorangan yang bertujuan untuk digunakan demi keberlangsungan pendidikan yang ditempuh (Agus, 2009). Adapun menurut Kamus Besar Bahasa Indonesia Pengertian Beasiswa adalah Tunjangan yang diberikan kepada pelajar atau mahasiswa sebagai bantuan biaya belajar. Beasiswa diberikan kepada mahasiswa sesuai kriteria yang diminta penyandang dana, baik program studinya maupun jenjang pendidikan mahasiswa. Waktu pemberian beasiswa lamanya berkisar sampai dengan 12 bulan, bahkan dapat diperpanjang sampai selesai kuliah, minimal diberikan kepada mahasiswa yang duduk di semester 3. Maksud diberikannya beasiswa menurut Agus(2009) adalah untuk membantu membiayai pendidikan mahasiswa dari keluarga yang kurang/tidak mampu secara ekonomi. Dengan berbagai tujuan diantaranya, mengurangi mahasiswa yang putus kuliah, membantu mahasiswa dalam menyelesaikan pendidikannya tepat waktu, dan membantu mahasiswa menetapkan kariernya sejak awal.

#### 2.2.2 Sistem Informasi

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan (Jogiyanto, 2005).

Menurut Ardiyanto (2011), sistem informasi merupakan sekumpulan komponen pembentuk sistem yang mempunyai keterkaitan antara satu komponen dengan komponen lainnya yang bertujuan menghasilkan suatu informasi dalam suatu bidang tertentu. Dalam sistem informasi diperlukannya klasifikasi alur informasi, hal ini disebabkan keanekaragaman kebutuhan akan suatu informasi oleh pengguna informasi. Kriteria dari sistem informasi antara lain, fleksibel, efektif dan efisien. Sistem informasi berbasis komputer (*Computer based information system*) terdiri dari komponen-komponen sebagai berikut:

- a. Perangkat keras (*hardware*), merupakan komponen untuk melengkapi kegiatan memasukkan data, memproses data dan keluaran data.
- b. Perangkat lunak (*software*), merupakan program dan instruksi yang diberikan ke komputer untuk menjalankan sistem.
- c. Basis data (*database*), merupakan sekumpulan data dan informasi yang diorganisasikan sedemikian rupa sehingga mudah diakses pengguna sistem informasi.
- d. Telekomunikasi, merupakan sebuah sistem yang menghubungkan anantara pengguna sistem dengan sistem komputer secara bersama-sama ke dalam suatu jaringan kerja yang efektif.
- e. Manusia (*human*), merupakan personel dari sistem informasi, meliputi manajer, analis, *programmer*, dan *operator*, serta bertanggung jawab terhadap perawatan sistem.
- f. Prosedur, merupakan tata cara yang meliputi strategi, kebijakan, metode dan peraturan-peraturan dalam menggunakan sistem informasi.

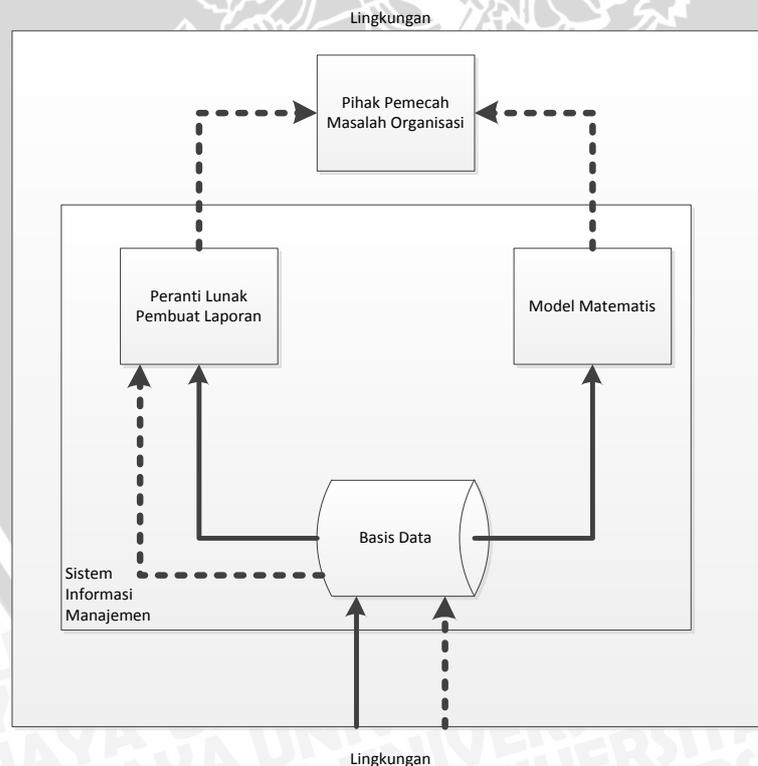
Sistem informasi manajemen (*management information system*) merupakan penerapan sistem informasi didalam organisasi untuk mendukung informasi-informasi yang dibutuhkan oleh semua tingkatan manajemen (Jogiyanto, 2005). Sedangkan menurut Ismail (2004), sistem informasi manajemen merupakan serangkaian sub-sistem informasi yang menyeluruh dan terkoordinasi yang secara rasional mampu mentransformasikan data sehingga menjadi informasi dengan berbagai cara guna meningkatkan produktivitas yang sesuai dengan gaya dan sifat manajer.

Sistem informasi manajemen adalah kunci dari suatu bidang yang menekankan personal manajemen yang dapat memproses dan mengolah data menjadi suatu bentuk informasi yang dapat digunakan dalam mendukung keputusan dengan melewati suatu prosedur kerja (aturan kerja) yang telah ditetapkan. Sistem informasi manajemen merupakan sistem yang menyediakan informasi yang digunakan untuk mendukung operasi, manajemen serta pengambilan keputusan sebuah organisasi. Jenis-jenis sistem informasi manajemen selanjutnya dispesialisasikan jenisnya menurut kebutuhan dari organisasi penggunanya. Sistem informasi yang digunakan di instansi pemerintah, pendidikan, rumah sakit atau perusahaan memiliki fungsi yang berbeda (Widyanti, 2006).

Sistem informasi manajemen secara umum dapat dikatakan sebagai sebuah sistem manusia dan mesin yang terintegrasi dalam menyediakan informasi guna mendukung fungsi operasi manajemen dan penentuan alternatif

tindakan dalam sebuah organisasi sistem tersebut. Dalam operasinya, sistem informasi manajemen menggunakan perangkat keras (*hardware*), perangkat lunak (*software*), prosedur, model manajemen dan keputusan serta sebuah terminal data. Sistem informasi manajemen sebagai suatu kumpulan manusia dan sumber modal di dalam suatu organisasi bertanggung jawab untuk pengumpulan dan pengolahan data sewaktu menghasilkan informasi yang berguna untuk setiap hierarki manajemen dalam perencanaan dan pengendalian kegiatan-kegiatan organisasi. Tujuan dari suatu sistem informasi manajemen adalah memberikan informasi untuk pembuatan keputusan dalam merencanakan, memulai, mengatur dan mengendalikan operasi sub-sistem dari perusahaan/organisasi dan juga untuk memberikan perusahaan sebuah sinergi dalam prosesnya (Gaol, 2008).

Para pengguna sistem informasi manajemen biasanya terdiri dari entitas-entitas organisasi *formal*-perusahaan atau sub-unit anak perusahaan. Informasi yang diberikan oleh sistem informasi manajemen menjelaskan perusahaan atau salah satu sistem utamanya dilihat dari apa yang telah terjadi di masa lalu, apa yang terjadi dan apa yang kemungkinan akan terjadi di masa depan. *Output* informasi yang dihasilkan akan digunakan oleh pihak-pihak yang akan memecahkan masalah (baik itu manajer maupun kalangan profesional) dalam mengambil keputusan guna memecahkan masalah perusahaan (McLeod dan Schell, 2008).



**Gambar 2.1** Model Sistem Informasi Manajemen (McLeod dan Schell, 2008)

Keterangan:

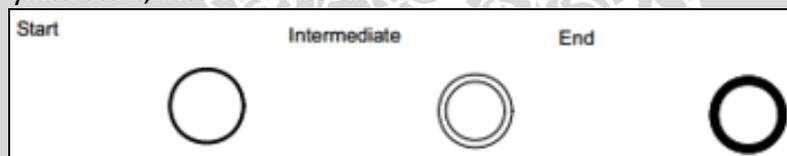
Data   
 Informasi

Pada **Gambar 2.1** terlihat bahwa pada basis data memuat data yang diberikan oleh sistem pemrosesan transaksi. Selain itu, baik data maupun informasi dimasukkan dari lingkungan. Lingkungan menjadi terlibat ketika perusahaan berinteraksi dengan organisasi-organisasi lain, seperti pemasok, untuk membentuk suatu sistem informasi antarorganisasi (*interorganizational information system*) (McLeod dan Schell, 2008).

### 2.2.3 BPMN

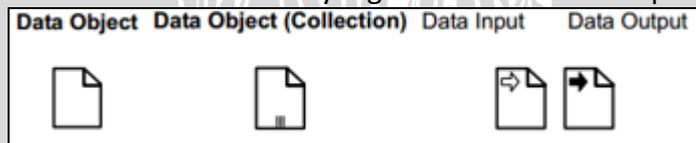
Menurut OMG (2013) Tujuan dari BPMN adalah untuk memberikan notasi yang mudah dipahami oleh semua pengguna bisnis, mulai dari bisnis analis yang membuat konsep awal dari sebuah proses, ke para pengembang teknis yang bertanggung jawab untuk mengimplementasikan suatu teknologi dari proses-proses tersebut. Dengan demikian, BPMN menciptakan suatu jembatan standar antara desain bisnis proses dan implementasi proses. Terdapat lima dasar kategori elemen yang digunakan, diantaranya :

1. Gambar 2.2 menggambarkan Flow Object yang terdiri dari event, activities dan gateways. Event menggambarkan sesuatu yang terjadi selama berlangsungnya proses bisnis. Event-event ini mempengaruhi aliran proses dan biasanya memiliki penyebab (trigger) atau hasil (result). Terdapat 3 tipe event yaitu start, intermediate dan end.



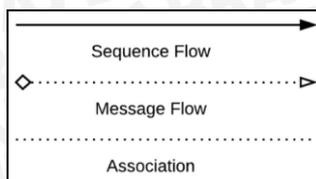
**Gambar 2.2** Notasi BPMN flow object (OMG, 2013)

2. Gambar 2.3 menggambarkan data yang terdiri dari data objects, data inputs, data outputs dan data stores. Data object memberikan informasi tentang kegiatan apa yang perlu dilakukan atau apa yang perlu dihasilkan. Data object dapat mewakili objek tunggal atau koleksi dari objek. Data input dan data output memberikan informasi yang sama untuk sebuah proses.



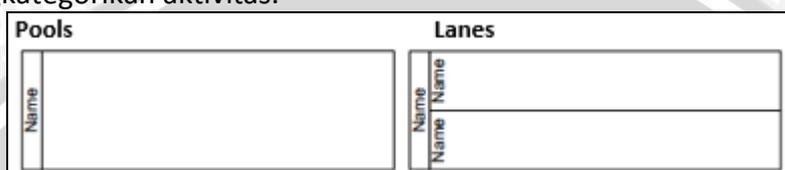
**Gambar 2.3** Notasi BPMN data (OMG, 2013)

3. Gambar 2.4 menggambarkan connecting objects yang terdiri dari sequence flows, messages flows data associations. Connecting object merupakan elemen yang menghubungkan flow object. Sequence flow digunakan untuk menunjukkan urutan yang kegiatan akan dilakukan dalam sebuah proses. Message flow digunakan untuk menunjukkan aliran pesan antara dua entitas yang siap untuk mengirim dan menerima. Association digunakan untuk asosiasi data, informasi dan artefak dengan aliran beda.



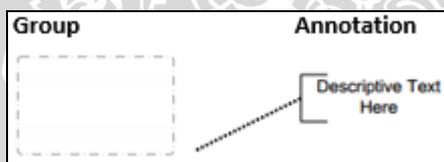
**Gambar 2.4** Notasi BPMN connecting objects (OMG, 2013)

- Gambar 2.5 menggambarkan swimlanes yang terdiri dari pools dan lanes. Pool yang mewakili partisipan dalam sebuah proses dan lane merupakan sub-bagian dalam sebuah pool dan akan menambah panjang dari pool baik vertikal maupun horizontal. Lanes digunakan untuk mengatur dan mengkategorikan aktivitas.



**Gambar 2.5** Notasi BPMN swimlanes (OMG, 2013)

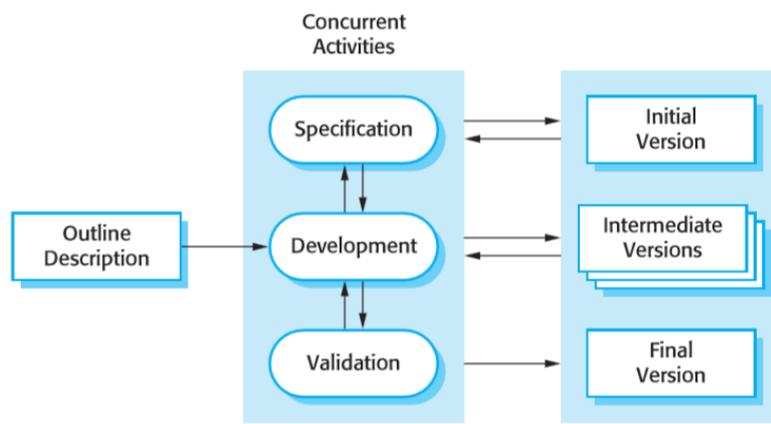
- Gambar 2.6 menggambarkan Artifacts yang terdiri dari group dan annotation. Group digunakan untuk tujuan dokumentasi atau analisis. Annotation merupakan mekanisme untuk pemodel memberikan informasi teks tambahan untuk pembaca dari diagram BPMN.



**Gambar 2.6** Notasi BPMN artifacts (OMG, 2013)

### 2.2.4 Metode iteratif

Metode iteratif merupakan metode dimana setiap tahapan/fase pengembangan sistem dilaksanakan secara berulang-ulang sampai mendapatkan hasil yang diinginkan (Neni dan Uluwiya, 2011). Gambar 2.7 menjelaskan proses yang akan diadaptasi pada skripsi ini yaitu *incremental development* yang terjadi iterasi dan menghasilkan *increment*. *Outline Description* adalah kerangka secara umumnya sedangkan pada *concurrent activities* terjadi proses iterasi yang menghasilkan sebuah *increment* yang akan diuji dan hasil *increment* berikutnya akan digabungkan dengan *increment* sebelumnya dan akan diuji lagi begitu seterusnya hingga proses iterasi berhenti.



**Gambar 2.7** Incremental development (Sommerville, 2010)

#### 2.4.2 Kelebihan dan kekurangan metode Iteratif

Pendekatan iteratif memiliki sejumlah keuntungan (Sommerville, 2010):

1. Pengguna bisa mendapatkan prioritas tinggi lebih awal sehingga dapat mengantisipasi *software failure*.
2. Pengguna lebih memanfaatkan atau berkomitmen dalam menggunakan sistem, karena pengguna ikut berperan langsung.

Kekurangan pendekatan iterative yaitu (Sommerville, 2010):

1. Proses *increment* harus dilakukan dalam porsi yang kecil dimana pemetaan porsi-porsi tersebut sulit dilakukan dalam pembagian yang pas.
2. Menganalisa kelas yang umum dipakai cukup sulit, karena tidak bisa dilihat dari awal apa yang akan terjadi pada *increment* mendatang.

#### 2.2.4 MoSCoW

MoSCoW adalah teknik prioritas untuk membantu memahami dan mengelola prioritas pekerjaan suatu proyek (DSDM Consortium, 2014). Aturan MoSCoW memiliki 4 prioritas yaitu :

1. Must Have (Mo) : Semua fitur yang diklasifikasikan dalam kelompok harus diimplementasikan, jika tidak disampaikan maka sistem tidak bekerja.
2. Should Have (S) : Fitur ini menjadi prioritas bagi sistem, namun dapat diabaikan apabila kendala waktu sangat beresiko.
3. Could Have (Co) : Fitur ini bisa ada namun juga dapat diabaikan karena tidak memengaruhi tuntutan bisnis.
4. Won't Have This Time (W) : Persyaratan ini akan dipenuhi bila terdapat waktu sisa dalam pengembangan sistem atau pada pengembangan selanjutnya.

### 2.2.5 Unified Modeling Language (UML)

*Unified Modeling Language* (UML) merupakan sistem arsitektur yang bekerja dalam OOAD (*Object-Oriented Analysis/Design*) dengan satu bahasa yang konsisten untuk menentukan, visualisasi, mengkonstruksi, dan mendokumentasikan *artifact* (sepotong informasi yang digunakan atau dihasilkan dalam suatu proses rekayasa software, dapat berupa model, deskripsi, atau software) yang terdapat dalam sistem software. UML merupakan bahasa pemodelan yang paling sukses dari tiga metode OO yang telah ada sebelumnya, yaitu Booch, OMT (*Object Modeling Technique*), dan OOSE (*Object-Oriented Software Engineering*) (Satzinger, 2011).

- **Tujuan UML diantaranya adalah:**

1. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.
2. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
3. Menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan. Untuk membuat suatu model, UML memiliki diagram grafis sebagai berikut:
  - a) *Business Use Case model*
  - b) *Aktivitas Diagram*
  - c) *Use Case model*
  - d) *Behavior diagram: Sequence diagram*
  - e) *Implementation diagram: Component diagram, Deployment diagram*
  - f) *Generate Code*

Diagram diagram tersebut diberi nama berdasarkan sudut pandang yang berbeda-beda terhadap sistem dalam proses analisis atau rekayasa. Dibuatnya berbagai jenis diagram diatas karena:

1. Setiap sistem yang kompleks selalu paling baik jika didekati melalui himpunan berbagai sudut pandang yang kecil yang satu sama lain hampir saling bebas (*independent*). Sudut pandang tunggal senantiasa tidak mencukupi untuk melihat isi item yang besar dan kompleks.
2. Diagram yang berbeda-beda tersebut dapat menyatakan tingkatan yang berbeda-beda dalam proses rekayasa.
3. Diagram-diagram tersebut dibuat agar model yang dibuat semakin mendekati realitas.

Diagram-diagram ini ditambah dengan kemampuan dokumentasi merupakan *artifacts* utama UML. *Data-flow diagram* dan tipe diagram lain yang tidak terdapat dalam UML tidak termasuk dalam paradigma *object-oriented*. *Aktivitas diagram* dan *collaboration diagram* yang terdapat dalam UML menggantikan

*data-flow diagram*. *Aktivitas diagram* juga sangat bermanfaat untuk membuat *workflow*.

### 2.2.5.1 Use Case Diagram

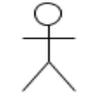
*Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem, yang ditekankan adalah “apa” yang diperbuat sistem dan bukan “bagaimana”. Sebuah *use case* mempresentasikan sebuah interaksi antara *actor* dengan sistem. *Use case* menggambarkan kata kerja seperti *Login* ke sistem, *maintenance user* dan sebagainya (Satzinger, 2011).

Lambang-lambang dalam *use case Diagram*

- 1). *Actor* merupakan sebuah entita yang berinterkasi dengan *use case*. Nama *actor* dituliskan di bawah gambar tersebut.
- 2). *Use case* menggambarkan sebuah fungsi tertentu yang disediakan oleh sistem, sebuah subsistem atau urutan pertukaran pesan antara anggota sistem dan satu atau lebih *actor* melakukan aksi yang dikerjakan oleh sistem.
- 3). Hubungan, menggambarkan hubungan *association*. Garis ini digunakan untuk menghubungkan antara *actor* dengan *use case*.

Komponen - komponen *Use case Diagram* disajikan dalam Tabel 2.1 :

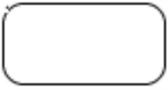
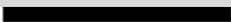
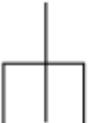
**Tabel 2.1** Notasi *use case diagram* (Satzinger, 2011)

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
3		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
4		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
5		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
6		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

### 2.5.2 Aktivitas Diagram

*Aktivitas* diagram menggambarkan berbagai alir aktifitas dalam sebuah sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi dan bagaimana mereka berakhir (Satzinger, 2011). *Aktivitas* diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Aktivitas* diagram tidak menggambarkan sifat internal dari sebuah sistem dan interaksi antara beberapa sub sistem secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Lambang *Aktivitas* Diagram dijelaskan pada Tabel 2.2 :

**Tabel 2.2** Notasi Aktivitas Diagram (Satzinger, 2011)

NO	GAMBAR	NAMA	KETERANGAN
1		<i>State</i>	Nilai atribut dan nilai link pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		<i>Initial Pseudo State</i>	Bagaimana objek dibentuk atau diawali
3		<i>Final State</i>	Bagaimana objek dibentuk dan dihancurkan
4		<i>Transition</i>	Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya
5		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
6		<i>Fork</i>	Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu
7		<i>Pilihan</i>	Pilihan untuk mengambil keputusan
8		<i>Rake</i>	Menunjukkan adanya dekomposisi

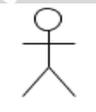
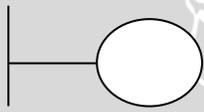
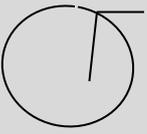
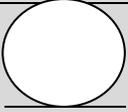
### 2.2.5.2 Sequence Diagram

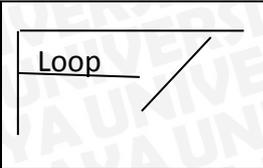
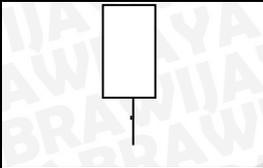
Diagram sekuensial atau *sequence* diagram digunakan untuk menggambarkan aliran fungsionalitas dalam *use case* (Satzinger, 2011). Diagram ini disusun berdasarkan urutan waktu. Kegunaan *sequence* diagram:

- *Sequence* diagram dapat digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *events* untuk menghasilkan *output* tertentu. Diawali dari apa yang *trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan.
- Diagram ini secara khusus berasosiasi dengan *use case* diagram.
- Memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu di dalam *use case*.

Ada pun komponen – komponen yang digunakan dalam *sequence* diagram sebagai berikut:

**Tabel 2.3** Notasi *sequence* Diagram (Satzinger, 2011)

NO	GAMBAR	NAMA	KETERANGAN
1		Actor	Segala sesuatu yang berinteraksi dengan langsung dengan system aplikasi computer.
2		Boundary Class	Boundary terletak diantara system dengan dunia sekelilingnya. Semua <i>form</i> , laporan-laporan, antar muka ke perangkat keras seperti cetaker atau scanner dan antar muka ke system lainnya adalah termasuk kategori.
3		Control Class	Control hubungan dengan fungsionalitas seperti pemanfaatan sumber daya, pemrosesan terdistribusi atau penanganan kesalahan
4		Entity Class	Entity digunaka untuk menangani informasi yang mungkin akan disimpan secara permanen. Entity juga merupakan sebuah table pada struktur basis data
5		Message	Message digambarkan dengan anak panah horizontal atara activation. Message mengidikasikan komunikasi antara objek-objek
6		Self-message	Self-message atau panggilan mandiri mengindikasikan komunikasi kembali ke dalam sebuah objek itu sendiri.

7		Loop	Operator loop adalah fragmen yang dapat mengeksekusi berulang kali dan penjaga menunjukkan dasar iterasi
8		Time line	Tanda waktu

### 2.2.5.3 Class Diagram

*Class* adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan berorientasi objek (Satzinger, 2011). *Class* menggambarkan keadaan (*attribute/property*) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi). *Class* diagram menggambarkan struktur dan deskripsi class, *packed* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi dan lainnya. Simbol *class* diagram disajikan dalam **Gambar 2.8**

Simbol	Keterangan
<i>Class</i>	- <i>Class</i> = nama <i>class</i> .
- <i>attribute</i> : <i>int</i>	- <i>Attribute</i> = Atribut yang di pakai pada <i>class</i> .
* <i>operation</i> ()	- Method / <i>operation</i> = aksi yang dapat dilakukan oleh sebuah <i>class</i>
N	Banyak
1	Satu

**Gambar 2.8** Simbol *class* diagram (Satzinger, 2011)

**Tabel 2.4** Notasi *class* diagram (Satzinger, 2011)

Jenis	Keterangan	Simbol
Class	<b>Class</b> adalah blok-blok pembangunan pada pemrograman berorientasi obyek. Sebuah <i>class</i> digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian	<b><i>Class</i>()</b> - <i>Attribute</i> : <i>int</i> <b>*<i>operation</i>() () : void</b>
<i>Composition</i>	Jika sebuah <i>class</i> tidak bisa berdiri sendiri dan harus	

	merupakan bagian dari <i>class</i> yang lain, maka <i>class</i> tersebut memiliki relasi <i>Composition</i> digambarkan sebagai garis dengan ujung berbentuk jajar genjang berisi.	
<i>Aggregation</i>	<i>Aggregation</i> mengindikasikan keseluruhan bagian <i>relationship</i> dan sebuah <i>aggregation</i> digambarkan sebagai sebuah garis dengan sebuah jajaran genjang yang tidak berisi.	
<i>generalization</i>	Sebuah relasi <i>generalization</i> sepadan dengan sebuah relasi <i>inheritance</i> pada konsep berorientasi obyek. Sebuah <i>generalization</i> dilambangkan dengan sebuah panah dengan kepala panah yang tidak solid yang mengarah ke kelas induknya.	
<i>Implicity/association</i>	<i>Implicity</i> mengindikasikan keterkaitan antara dua relasi. Misalnya relasi A berhubungan dengan relasi B dengan <i>Implicity</i> 1....n ( <i>one to many</i> ) <b>Association</b> merupakan sebuah <i>relationship</i> paling umum antara 2 <i>class</i> dan dilambangkan oleh sebuah garis yang menghubungkan antara 2 <i>class</i> . Garis ini bisa melambangkan tipe-tipe <i>relationship</i> dan juga dapat menampilkan hukum-hukum <i>relationship</i> .	0.....1 *.....n 1.....1 1, 2, 3, .....*



## 2.2.6 Pengujian Sistem

Pengujian dilakukan untuk menguji setiap modul untuk menjamin setiap modul menjalankan fungsinya dengan baik (Fatta, 2007). Ada dua (2) metode untuk melakukan unit testing menurut Fatta (2007), yaitu *blackbox testing* dan *whitebox testing*. Namun di penelitian ini yang digunakan adalah *blackbox testing* dan pengujian kompatibilitas.

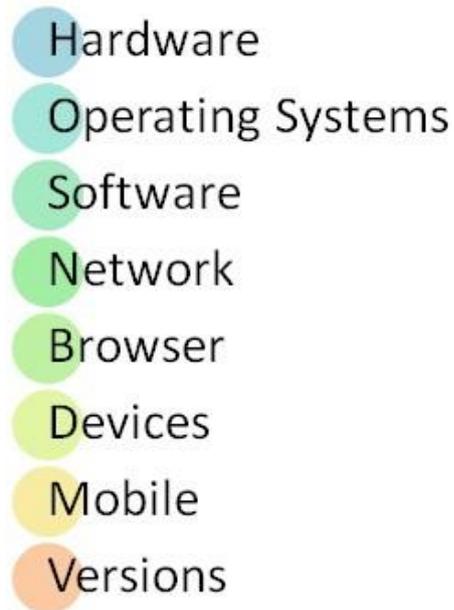
### 2.2.6.1 Blackbox Testing

*Blackbox testing* fokus pada apakah unit program memenuhi kebutuhan (*requirement*) yang disebutkan dalam spesifikasi. Pada *blackbox testing*, cara pengujian hanya dilakukan dengan menjalankan atau mengeksekusi unit atau modul, kemudian diamati apakah hasil dari unit itu sesuai dengan proses bisnis yang diinginkan. Jika ada unit yang tidak sesuai *output*-nya, maka untuk menyelesaikannya diteruskan pada pengujian kedua, yaitu *whitebox testing*. Menurut Yulianto (2012), *blackbox testing* cenderung untuk menemukan hal-hal berikut:

- a. Fungsi yang tidak benar atau tidak ada.
- b. Kesalahan antarmuka (*interface errors*).
- c. Kesalahan pada struktur data dan akses basis data.
- d. Kesalahan performansi (*performance errors*).
- e. Kesalahan inisialisasi dan terminasi.

### 2.2.6.2 Pengujian kompatibilitas

Pengujian *Compatibility* merupakan salah satu tipe pengujian non-fungsional yang bertujuan untuk memastikan bahwa perangkat lunak yang dibangun mampu berjalan dengan baik pada lingkungan yang berbeda (lingkungan pengguna). Tipe pengujian *compatibility* ditunjukkan pada **Gambar 6.1**.



**Gambar 2.9** Tipe Pengujian *Compatibility* (Guru99, 2016)

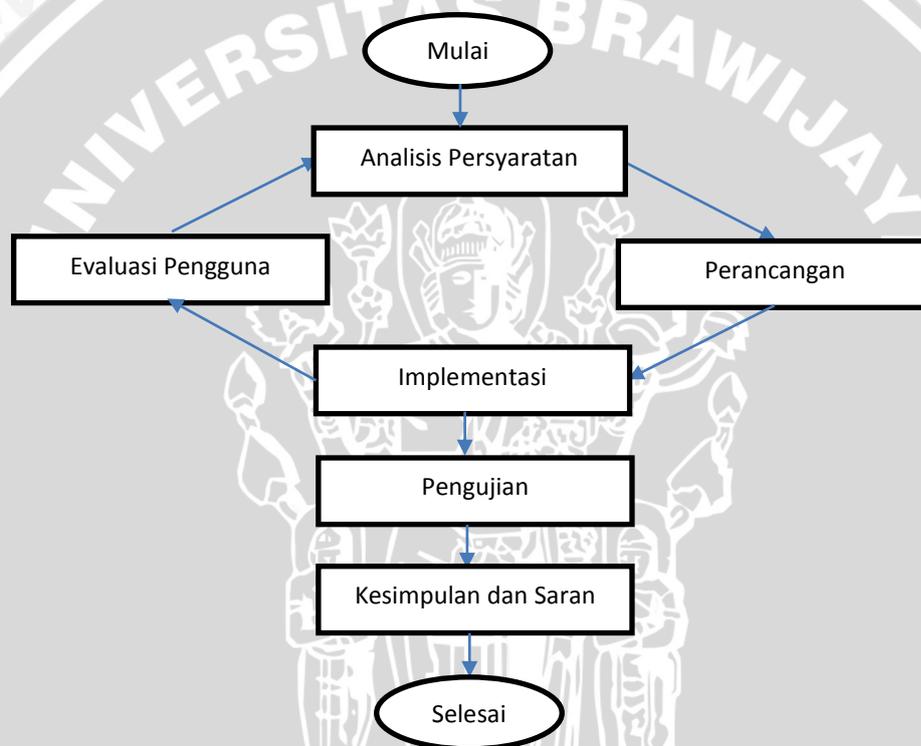
Tipe dari pengujian *compatibility* yang akan digunakan pada penelitian ini adalah tipe pengujian pada sisi *browser*. Pengujian ini dilakukan pada beberapa *browser* berbeda yang berjalan pada perangkat *personal computer* maupun perangkat *mobile*, seperti *Mozilla Firefox*, *Google Chrome*, *Internet Explorer*, *Microsoft Edge*, *Safari*, *browser Android*, dan *browser iOS*. Pengujian *compatibility* pada sisi *browser* dapat dilakukan dengan menggunakan *tools* atau alat bantu berupa perangkat lunak, diantaranya adalah *SortSite*, *Secure Platform*, *Virtual Desktops*, *Selenium*, *iMacros*, *Sahi*, *Ranorex Studio*, dan lain sebagainya.

*Tools* yang akan digunakan pada penelitian ini adalah *Sortsite*. *SortSite* dipilih karena mampu menguji perangkat lunak berbasis *web* secara keseluruhan pada setiap halaman yang dimiliki. Selain itu, *SortSite* mampu menguji perangkat lunak berbasis *web* pada beberapa *checkpoints* diantaranya *Accessibility*, *Broken Links*, *Compatibility*, *Search Engine Optimization*, *Privacy*, *Web Standart*, dan *Usability* (PowerMapper, 2016).

## BAB 3 METODOLOGI

Bagian metode penelitian akan menjelaskan tahapan pengembangan sistem yang digunakan untuk sistem informasi beasiswa dengan menggunakan metode iterasi. Tahapan yang dilakukan mulai dari analisis persyaratan, perancangan, implementasi, evaluasi oleh pengguna, pengujian dan penarikan kesimpulan serta saran.

Alur penelitian merupakan tahapan bagaimana langkah-langkah yang ditempuh untuk menyelesaikan penelitian. Adapun alur metode penelitian yang dilakukan dapat dilihat pada gambar 3.1.



**Gambar 3.1** Diagram Alur Penelitian

### 3.1 Analisis Persyaratan

Pada tahapan ini dilakukan analisis terhadap kondisi yang terjadi saat ini. Tahap ini merupakan tahap terpenting dalam mengembangkan suatu sistem. Pada tahap ditujuan untuk mengetahui persyaratan – persyaratan apa saja yang diinginkan oleh pengguna. Dalam tahap ini akan dilakukan wawancara yang akan dilakukan kepada bagian kemahasiswaan fakultas dan BEM bagian beasiswa. Kemudian dilakukan survei dan analisis terhadap sistem yang sudah ada. Dari situ dapat diperoleh gambaran umum sistem, identifikasi aktor yang terlibat, daftar persyaratan yang nantinya dimodelkan ke dalam diagram *use case*.

### 3.2 Perancangan

Perancangan dilakukan ketika semua persyaratan sistem telah tersedia persyaratan sistem yang sudah diperoleh melalui tahap analisis persyaratan kemudian akan menghasilkan perancangan awal. Lalu dari perancangan awal akan diiterasi mendapatkan perancangan iterasi pertama sampai iterasi kedua. Tahap ini merupakan proses menerjemahkan persyaratan yang didapat sebelumnya menjadi model rancangan perangkat lunak menggunakan pemodelan UML (Unified Modeling Language).

### 3.3 Implementasi

Pada tahapan ini akan dilakukan proses menerjemahkan hasil perancangan yang didapat pada tahap sebelumnya kedalam bahasa pemrograman. Implementasi dilakukan menggunakan bahasa pemrograman PHP dengan menggunakan framework CI (Code Ignitier). Tahap ini akan menghasilkan prototipe versi awal yang kemudian dilakukan proses iterasi hingga menghasilkan sistem yang lebih lengkap.

### 3.4 Evaluasi Pengguna

Pada tahapan ini dilakukan proses evaluasi oleh pengguna. Pengguna akan menilai apakah rancangan yang dibangun sudah sesuai. Jika belum sesuai, rancangan akan direvisi berdasarkan umpan balik dari pengguna. Setelah proses ini diharapkan perangkat lunak nantinya akan sesuai dengan persyaratan yang diharapkan pengguna.

### 3.5 Pengujian

Pada tahapan pengujian sistem, parameter yang dilakukan disesuaikan dengan perancangan sistem. Pengujian menggunakan metode *blackbox* dan pengujian kompatibilitas browser. Pengujian secara *blackbox* menguji apakah masukan yang diberikan dan keluaran yang diberikan sesuai. Metode ini juga digunakan untuk mengecek apakah sistem yang dibuat sudah memenuhi requirement yang dijabarkan di awal. Sedangkan pengujian kompatibilitas browser menguji apakah sistem dapat berjalan diberbagai macam *browser*.

### 3.6 Kesimpulan dan Saran

Kesimpulan didapatkan dari hasil pengujian dan analisis setelah semua tahapan selesai dilakukan. Saran didaptkn untuk memperbaiki kesalahan-kesalahan yang terjadi dan sebagai bahan pertimbangan untuk pengembangan sistem selanjutnya.

## BAB 4 ANALISIS PERSYARATAN SISTEM

Tahap ini akan menjelaskan analisis terhadap sistem yang ada sekarang dan sistem yang akan dibuat perancangannya tentang pelaksanaan yang berkaitan dengan sistem informasi beasiswa universitas brawijaya. Perancangan ini menggunakan pendekatan berorientasi objek dan UML. Pada skripsi ini menggunakan metode iteratif sehingga akan dilakukan iterasi 2 kali pada tahap analisis persyaratan dan perancangan. Pada bab ini akan dituliskan terakhir sedangkan iterasi awal akan dijelaskan pada lampiran bab proses iterasi.

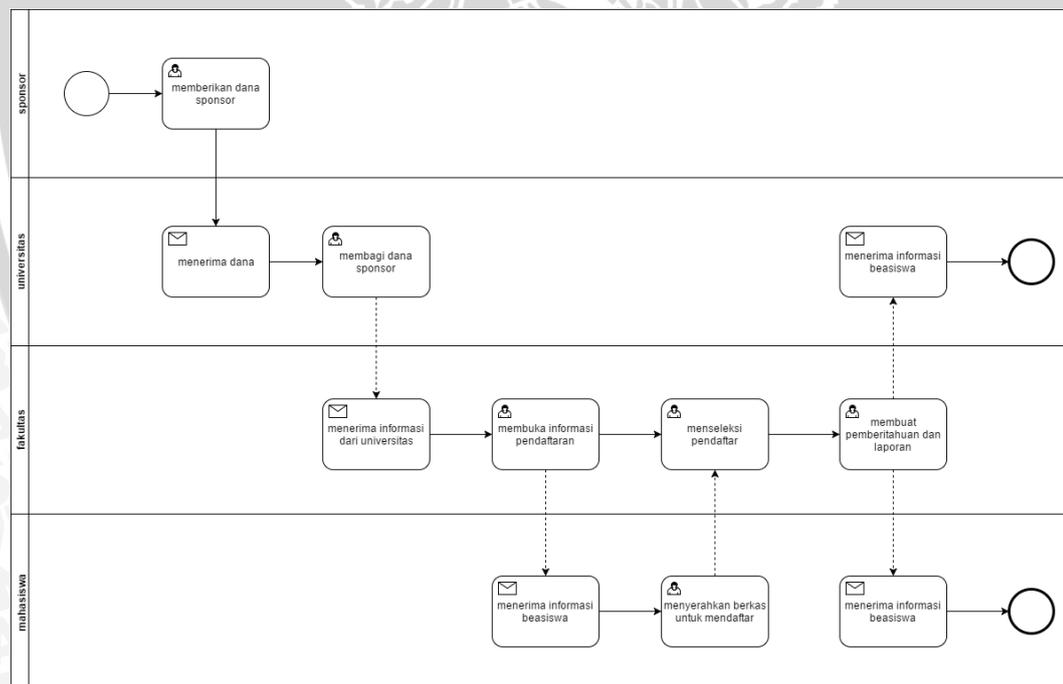
### 4.1 Analisis Persyaratan

Tahapan ini berisi analisis awal seperti pemodelan proses bisnis, problem statement yang melatar belakangi persyaratan sistem informasi beasiswa mulai dari analisis stakeholder dan pengguna, fitur serta persyaratan sistem.

#### 4.1.1 Analisis Proses Bisnis

Analisis proses bisnis dilakukan dengan melakukan pemodelan menggunakan BPMN.

#### 4.1.2 Proses Bisnis Saat Ini



**Gambar 4.1** Proses Bisnis Saat Ini

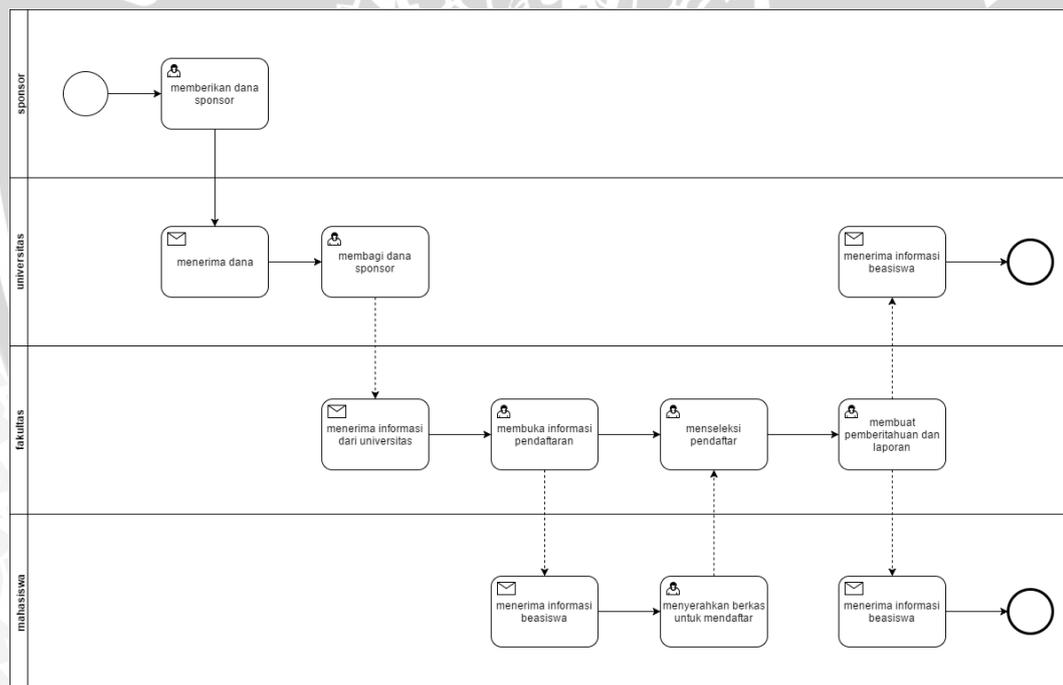
Dari gambar 4.1 dapat dilihat pihak-pihak yang terlibat mulai dari Sponsor, Universitas, Fakultas, dan Mahasiswa. Mulai dari pihak sponsor memberikan dana yang kemudian diterima oleh pihak universitas, pihak universitas kemudian membagi dana tersebut ke setiap fakultas, fakultas kemudian membagikan informasi ke mahasiswa dan menseleksi mahasiswa yang mendaftar sampai

mendapatkan calon yang sesuai. Kelebihan dan kekurangan pada proses bisnis saat ini dapat dilihat pada tabel 4.1 :

**Tabel 4.1** Kelebihan dan Kekurangan Proses Bisnis Saat Ini

Kekurangan Proses Bisnis Saat ini	Kelebihan Proses Bisnis Saat ini
<ol style="list-style-type: none"> <li>1. Adanya Website Beasiswa UB namun hanya mampu menampilkan berita, ada fitur login namun hanya untuk penerima beasiswa PPA-BBP</li> <li>2. Untuk beasiswa yang belum ditangani pendaftarannya masih manual.</li> <li>3. Beasiswa salah target dikarenakan satu mahasiswa mampu menerima lebih dari satu beasiswa.</li> </ol>	<ol style="list-style-type: none"> <li>1. kemudahan mencari informasi beasiswa</li> </ol>

#### 4.1.3 Proses Bisnis Usulan



**Gambar 4.2** Proses Bisnis

Gambar 4.2 menampilkan gambaran proses bisnis yang ada sekarang , disini terlihat tidak ada perubahan , karena penulis tidak mengubah proses bisnis namun ingin memberikan sistem informasi yang cakupannya tidak hanya membuat pendaftaran beasiswa namun sampai proses mahasiswa menerima pemberitahuan.

**Tabel 4.2** Perbandingan sistem informasi saat ini dengan sistem usulan

Beasiswa.ub.ac.id	Sistem Informasi Beasiswa
<p>1. Sistem dapat menampilkan informasi beasiswa</p>	<p>1. Sistem dapat menampilkan informasi beasiswa</p> <p>2. Mahasiswa dapat mendaftar melalui sistem</p> <p>3. Sistem dapat membantu seleksi beasiswa</p> <p>4. Sistem dapat mencetak laporan penerima beasiswa</p> <p>5. Sistem dapat mengecek apakah mahasiswa sedang menerima beasiswa atau tidak</p>

#### 4.2 Pemetaan Proses Bisnis dengan Persyaratan Fungsional

Tabel 4.3 merupakan pemetaan persyaratan fungsional dari proses bisnis diatas

**Tabel 4.3** Aktivitas pada BPMN

Kode Aktivitas	Nama	Keterangan
A01	Memberikan dana sponsor	Pihak sponsor memberitahu tentang adanya pemberian dana sponsor atau beasiswa
A02	Menerima informasi	Informasi diterima oleh pihak universitas
A03	Membagi dana sponsor	Pihak universitas membagi dana tersebut menjadi kuota pada tiap tiap fakultas
A04	Menerima informasi dari universitas	Pihak fakultas menerima pemberitahuan dari universitas
A06	Membuka informasi pendaftaran	Pihak fakultas membuat pengumuman mengenai beasiswa
A07	Menerima informasi beasiswa	Mahasiswa menerima informasi beasiswa
A08	Menyerahkan berkas untuk mendaftar	Mahasiswa melakukan pendaftaran
A09	Menseleksi pendaftar	Pihak fakultas melakukan seleksi
A10	Membuat pemberitahuan dan laporan	Pihak fakultas membuat laporan dan pengumuman
A11	Menerima informasi	Informasi diterima

#### 4.2.1 Analisis Pemangku Kepentingan dan Pengguna

Menjelaskan mengenai pemangku kepentingan yang karakteristik dan relasi yang sama dengan sistem. Berikut ini pada tabel 4.4 merupakan tipe pemangku kepentingan pada sistem informasi beasiswa :

**Tabel 4.4** Tipe pemangku kepentingan

Tipe Pemangku Kepentingan	Deskripsi Pemangku Kepentingan	Peran Pemangku Kepentingan
Mahasiswa	Mahasiswa universitas universitas brawijaya	Pendaftar beasiswa
Fakultas	Sekelompok orang yang bertanggung jawab dalam menseleksi beasiswa dalam lingkup fakultas	Menseleksi calon penerima beasiswa
Universitas	Sekelompok orang yang bertanggungjawab dalam membagi dana sponsor ketiap fakultas	Mengalokasi dana sponsor
Sponsor	Sekelompok orang yang memberikan dana beasiswa	Memberikan beasiswa

#### 4.2.2 Key Stakeholder and User Needs

Tahap ini menjelaskan persyaratan sistem yang dibutuhkan oleh pemangku kepentingan. Tabel 4.5 memperlihatkan persyaratan stakeholder-stakeholder , kondisi saat ini dan kondisi yang dituju setelah dilakukan evaluasi sampai iterasi terakhir.

**Tabel 4.5** Key Stakeholder Needs iterasi ke-2

Need	Stakeholder	Priority	Current Situation	Proposed Solutions
Kemudahan mencari informasi beasiswa	mahasiswa	MO	Pencarian informasi beasiswa sudah memudahkan	Pencarian beasiswa ditampilkan dalam tabel yang bisa diurutkan berdasarkan nama , periode

Tabel 4.5 Key Stakeholder Needs iterasi ke-2 (lanjutan)

Need	Stakeholder	Priority	Current Situation	Proposed Solutions
Proses pendaftaran lebih mudah	mahasiswa	MO	Manual dengan datang ke lembaga kemahasiswaan	Terdapat fitur untuk mendaftar beasiswa dengan mengunggah berkas
Proses seleksi dipermudah	fakultas	MO	Manual dengan menseleksi berkas atau wawancara	Terdapat fitur <i>sorting</i> pendaftar berdasarkan IPK tertinggi
Mahasiswa hanya bisa menerima 1 beasiswa	mahasiswa	MO	-	Sistem mendeteksi status mahasiswa, jika sudah menerima beasiswa tidak dapat mendaftar lagi
Mahasiswa dapat melihat pengumuman beasiswa	mahasiswa	S	Dengan kertas pengumuman dan pengumuman di website	Terdapat fitur pengumuman hasil seleksi
Pengalokasian dana terkomputerisasi	universitas	MO	Manual menggunakan <i>form</i> kertas	Pihak universitas dapat mengalokasikan dana ketika membuat beasiswa baru
Dapat membuat laporan penerima beasiswa	Universitas dan fakultas	MO	Manual menggunakan <i>form</i> kertas	Sistem menyediakan fitur untuk mencetak laporan

**Tabel 4.5** Key Stakeholder Needs iterasi ke-2 (lanjutan)

Need	Stakeholder	Priority	Current Situation	Proposed Solutions
				penerima beasiswa
Kemudahan dalam melakukan	Universitas dan fakultas	MO	Ditulis secara manual	Sistem menyediakan fitur untuk mengelola data
pengelolaan data <i>user</i> dan fakultas serta jurusan		MO		yang meliputi membuat ,menghapus dan mengubah data

#### 4.2.3 Pemetaan Proses Bisnis Terhadap Fitur Produk

Berikut ini pada tabel 4.6 merupakan pemetaan dari aktivitas pada proses bisnis terhadap fitur produk berdasarkan kebutuhan *user*

**Tabel 4.6** Fitur Produk

Kode Aktivitas	Kode Fungsi	Nama Fungsi	Deskripsi
A03	F01	Membagi dana sponsor	Pihak universitas membagi dana tersebut menjadi kuota pada tiap fakultas
A06	F01	Membuka informasi pendaftaran	Pihak fakultas membuat pengumuman mengenai beasiswa
A07	F02	Menerima informasi beasiswa	Mahasiswa menerima informasi beasiswa
A08	F02	Menyerahkan berkas untuk mendaftar	Mahasiswa melakukan pendaftaran
A09	F03	Menseleksi pendaftar	Pihak fakultas melakukan seleksi
A10	F04 F05	Membuat pemberitahuan dan laporan	Pihak fakultas membuat laporan dan pengumuman

#### 4.2.4 Fitur Produk

Tabel 4.7 menjelaskan kesimpulan kemampuan sistem informasi beasiswa yang dibangun secara umum. Prioritas pada fitur menggunakan aturan MoSCoW.

Tabel 4.7 Fitur Produk

Kode	Deskripsi	Nama Fitur	Prioritas
F01	Sistem dapat digunakan untuk mengelola data beasiswa	Mengelola data beasiswa	MO
F02	Sistem dapat digunakan untuk melakukan proses mendaftar	Mengajukan beasiswa	MO
F03	Sistem dapat digunakan untuk melakukan proses seleksi mahasiswa	Menseleksi partisipan	MO
F04	Sistem dapat digunakan untuk menampilkan pengumuman beasiswa	Melihat pengumuman	MO
F05	Sistem dapat digunakan untuk membuat laporan	Membuat laporan	MO
F06	Sistem dapat digunakan untuk mengubah profil	Mengubah profil	S
F07	Sistem dapat digunakan untuk mengelola data <i>user</i>	Mengelola data <i>user</i>	MO
F08	Sistem dapat digunakan untuk mengelola data fakultas dan jurusan	Mengelola data fakultas dan jurusan	MO
F09	Adanya autentikasi pengguna	Login	MO
F10	Sistem mampu diakses di berbagai web browser	Compabilitas	MO

#### 4.2.5 Spesifikasi Persyaratan

Di tahap ini persyaratan dibedakan menjadi dua, yaitu persyaratan fungsional dan persyaratan non fungsional. Pada penentuan spesifikasi persyaratan diawal memiliki persyaratan sebanyak 7 persyaratan fungsional dan 3 persyaratan non-fungsional. Kemudian dilakukan iterasi pertama spesifikasi persyaratan menjadi 9. Pada iterasi kedua hanya mengalami perubahan persyaratan non-fungsional menjadi 1.

##### 4.2.5.1 Persyaratan Fungsional

Berikut ini pada tabel 4.7 merupakan persyaratan fungsional sistem informasi beasiswa :

Tabel 4.7 Persyaratan Fungsional iterasi ke-2

No	Kode Fitur	Kode Dasar Fungsi	Kode Lengkap Fungsi	Deskripsi
1	F01	SRS-F-01	F01-SRS-F-01	Sistem dapat digunakan untuk menambah beasiswa baru
			F01-SRS-F-02	Sistem dapat digunakan untuk mengubah data beasiswa
			F01-SRS-F-03	Sistem dapat digunakan untuk menghapus data beasiswa
2	F02	SRS-F-02	F02-SRS-F-01	Sistem dapat digunakan untuk menampilkan informasi mengenai beasiswa yang tersedia
			F02-SRS-F-02	User mahasiswa dapat mendaftar beasiswa melalui sistem
			F02-SRS-F-03	Sistem dapat digunakan untuk menverifikasi apakah mahasiswa sedang menerima beasiswa atau tidak
3	F03	SRS-F-03	F03-SRS-F-01	Sistem dapat digunakan untuk membantu fakultas dalam menyeleksi mahasiswa yang mendaftar dengan tampilan table dan adanya <i>sorting</i>
4	F04	SRS-F-04	F04-SRS-F-01	Sistem dapat digunakan untuk menampilkan pengumuman hasil seleksi beasiswa
5	F05	SRS-F-05	F05-SRS-F-01	Sistem dapat digunakan untuk mencetak laporan penerima beasiswa
6	F06	SRS-F-06	F06-SRS-F-01	Sistem dapat digunakan untuk mengubah data profil
			F07-SRS-F-05	Sistem terdapat fitur mengubah <i>password</i>
7	F07	SRS-F-07	F07-SRS-F-01	Sistem dapat digunakan untuk menambah <i>user</i> baru
			F07-SRS-F-02	Sistem dapat digunakan untuk mengubah data <i>user</i>
			F07-SRS-F-03	Sistem dapat digunakan untuk menghapus data <i>user</i>

**Tabel 4.7** Persyaratan Fungsional iterasi ke-2(lanjutan)

No	Kode Fitur	Kode Dasar Fungsi	Kode Lengkap Fungsi	Deskripsi
8	F08	SRS-F-08	F08-SRS-F-01	Sistem dapat digunakan untuk menambah jurusan baru
			F08-SRS-F-02	Sistem dapat digunakan untuk mengubah data jurusan
			F08-SRS-F-03	Sistem dapat digunakan untuk menghapus data jurusan
			F08-SRS-F-04	Sistem dapat digunakan untuk menambah fakultas baru
			F08-SRS-F-05	Sistem dapat digunakan untuk mengubah data fakultas
			F08-SRS-F-06	Sistem dapat digunakan untuk menghapus data fakultas
9	F09	SRS-F-09	F09-SRS-F-01	Sistem terdapat fitur untuk dapat masuk kedalam sistem
			F09-SRS-F-02	Sistem terdapat fitur untuk dapat keluar dari sistem

#### 4.2.5.2 Persyaratan Non Fungsional

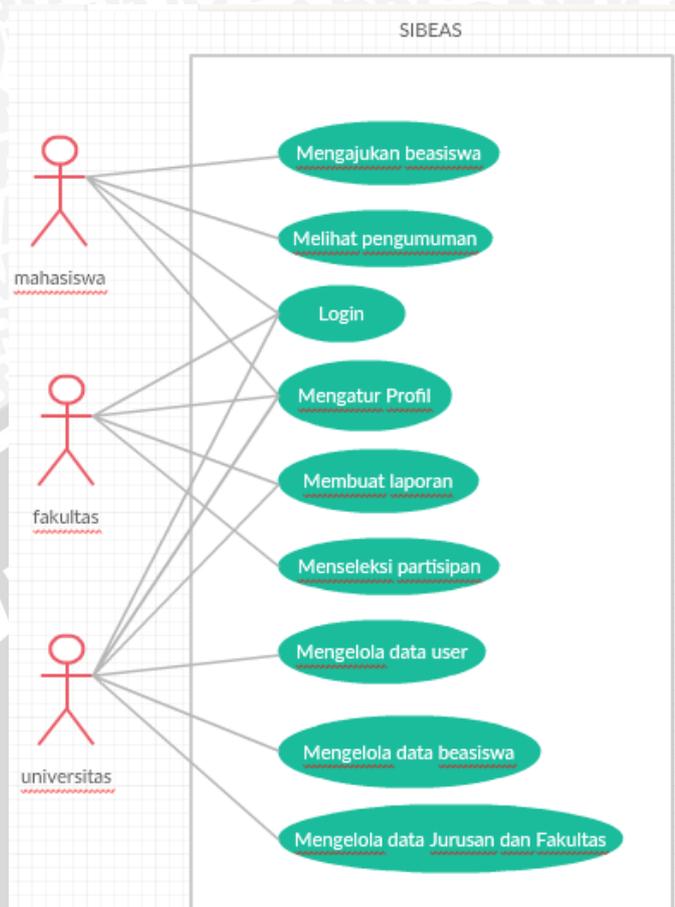
Berikut ini pada tabel 4.8 merupakan persyaratan non fungsional sistem informasi beasiswa:

**Tabel 4.8** Persyaratan Non Fungsional iterasi ke-2

Kode	Deskripsi persyaratan
SRS-NF-01	Sistem dapat dibuka pada Internet Explorer, Edge, Firefox, Safari, Opera, Chrome, Browser iOS, Browser Android.

#### 4.3 Persyaratan Naratif

Persyaratan sistem dijelaskan secara naratif menggunakan *use cases*. Diagram *use case* terdiri dari sekumpulan *use case*, aktor dan hubungannya. Berikut ini pada gambar 4.3 adalah *use case* keseluruhan dari sistem informasi beasiswa :



Gambar 4.3 Use Case Diagram

### 4.3.1 Deskripsi Aktor

Berikut ini pada tabel 4.9 merupakan deskripsi actor dari use case diagram sistem informasi beasiswa :

Tabel 4.9 Deskripsi aktor

Nama Aktor	Deskripsi Aktor
mahasiswa	Orang yang hendak mencari informasi terkait beasiswa dan mendaftar untuk mengajukan permohonan beasiswa
fakultas	Orang yang bertanggung jawab untuk menseleksi calon penerima beasiswa dan membuat laporan mengenai penerima ditingkat fakultas
universitas	Orang yang bertanggung jawab untuk mengelola data beasiswa, user dan fakultas serta jurusan

### 4.3.2 Spesifikasi Use Case

#### 1. Login

##### a. Brief Description

*Use case* ini digunakan untuk menggambarkan bagaimana *user* dapat melakukan login.

**b. Basic Flow of Events**

1. *Use case* dimulai ketika *user* membuka sistem.
2. Sistem menampilkan fungsi yang tersedia untuk *user* yaitu melakukan login.  
**{ Mengisi *username* dan *password* }**
3. Warga memilih untuk melakukan *sign in*.  
**{Melakukan Login}**
4. Sistem menampilkan halaman home *user*.  
**{Use Case Selesai}**
5. *Use Case* melakukan *login* selesai.

**c. Alternative Flows**

1. Terdapat data yang belum diisi  
Pada **{Mengisi *username* dan *password*}** di *basic flow*, apabila *user* tidak mengisi pada seluruh data yang diminta maka sistem akan menampilkan pesan untuk mengisi seluruh data dan melanjutkan *basic flow* pada **{Mengisi *username* dan *password*}**.
2. Tidak jadi melakukan pengelolaan data  
Pada **{Melakukan Login}** di *basic flow*, apabila *username* atau *password* yang dimasukan tidak sesuai, maka sistem akan menampilkan pesan bahwa *username* atau *password* salah dan melanjutkan *basic flow* pada **{Mengisi *username* dan *password*}**.
3. Tidak jadi *login*  
Kapanpun pada alur *basic flow*, *user* dapat tidak jadi melakukan *login* dan **{Use Case Selesai }**.

**d. Subflow**

Tidak terdapat *subflow* pada *use case* ini

**e. Key Scenarios**

1. *User* berhasil melakukan pendaftaran : *Basic flow*
2. Terdapat data yang belum diisi : *Basic flow*, Terdapat data yang belum diisi, *Basic flow*.
3. *Username* atau *password* salah : *Basic flow*, *Username* atau *password* salah.
4. Tidak jadi melakukan login : *Basic flow*, Tidak jadi login.

**f. Pre-Conditions**

Tidak terdapat pre-condition pada *use case* ini.

**g. Post-Conditions**

*User* berhasil melakukan login dan masuk ke dalam sistem informasi beasiswa.

**h. Extension Points**

Tidak ada *extension points* pada *use case* ini

**i. Special Requirements**

Tidak ada *special requirements* pada *use case* ini

## 2. Melihat pengumuman beasiswa

### a. *Brief Description*

*Use case* ini digunakan untuk menggambarkan bagaimana mahasiswa mengetahui pengumuman beasiswa apakah dirinya diterima atau tidak melalui halaman notifikasi yang tersedia dalam sistem

### b. *Basic Flow of Events*

1. *Use case* dimulai ketika mahasiswa masuk kedalam sistem dan mengakses laporan untuk melihat pengumuman.

**{Memilih laporan}**

2. Sistem menampilkan data berupa nama beasiswa, periode pendaftaran, status pendaftaran dan aksi untuk melihat *detail* beasiswa.

**{Use case selesai}**

3. *Use case* mengelola data survey selesai.

### c. *Alternative Flows*

Status belum tersedia

Pada **{Memilih laporan}** di *basic flow*, status kosong dikarenakan mahasiswa belum mendaftar beasiswa.

### d. *Subflow*

Tidak terdapat *subflow* pada *use case* ini

### e. *Key Scenarios*

1. mahasiswa berhasil melihat pengumuman beasiswa : *Basic flow*.
2. status belum tersedia : *Basic flow*, Status belum tersedia.

### f. *Pre-Conditions*

Mahasiswa sudah masuk ke dalam sistem dan telah mendaftar beasiswa.

### g. *Post-Conditions*

Mahasiswa berhasil melihat status pengumuman beasiswa apakah dirinya dinyatakan diterima atau tidak.

### h. *Extension Points*

Tidak ada *extension points* pada *use case* ini .

### i. *Special Requirements*

Tidak ada *special requirements* pada *use case* ini.

## 3. Mengatur Profil

### a. *Brief Description*

*Use case* ini menggambarkan ketika *user* ingin melihat dan mengatur profil atau mengubah *password*.

### b. *Basic Flow of Events*

1. *Use case* dimulai ketika *user* masuk kedalam sistem dan mengakses profil.

2. Sistem menampilkan data berupa informasi pribadi dan fungsi yang tersedia untuk mengelola profil. Fungsi tersebut meliputi mengubah profil dan ubah *password*.

**{Memilih operasi pengolahan data}**

3. Jika *user* memilih operasi untuk mengubah profil, maka sub alur **Mengubah profil** akan dijalankan.
4. Jika *user* memilih operasi untuk ubah *password*, maka sub alur **ubah Password** akan dijalankan.

**{Use Case Selesai}**

5. *Use Case* melakukan pendaftaran selesai

**c. Alternative Flows**

1. Tidak jadi melakukan pengelolaan data  
Pada seluruh waktu di *basic flow*, *user* dapat tidak jadi melakukan pengelolaan data dan **{Use Case Selesai}**

2. *Password* tidak sesuai  
Pada **{Memilih operasi pengolahan data}** di *basic flow*, apabila *password* lama yang dimasukan oleh *user* tidak sesuai dengan yang terdaftar pada sistem, maka sistem akan menampilkan pesan bahwa *password* salah. *user* dapat memasukan *password* kembali atau tidak jadi memperbaharui data dan **{Use Case Selesai}**

**d. Subflow**

**Mengubah profil**

- *User* memilih untuk melakukan mengubah profi
- Sistem menampilkan data dan meminta *user* untuk mengisi data yang akan diperbaharui.  
**{Mengisi Data Pembaharuan}**
- *User* mengisi data pemohon yang hendak diperbaharui.  
**{Melakukan Pembaharuan}**
- *User* melakukan pembaharuan data dengan menyimpan data yang sudah diperbaharui pada sistem

**Ubah password**

- *User* memilih untuk melakukan ubah *password*
- Sistem meminta *user* untuk mengisi *password* lama dan *password* baru  
**{Mengisi Password}**
- *User* mengisi *password* lama dan *password* yang hendak diperbaharui.  
**{Melakukan Pembaharuan}**
- *User* melakukan pembaharuan data dengan menyimpan data yang sudah diperbaharui pada sistem

**e. Key Scenarios**

1. *User* berhasil melakukan pengelolaan data : *Basic flow*
2. Tidak jadi mengelola data : *Basic flow*, Tidak jadi melakukan pengelolaan data.

3. *Password Salah* : *Basic flow*, *Password* tidak sesuai

**f. Pre-Conditions**

User sudah masuk ke dalam sistem.

**g. Post-Conditions**

User berhasil melakukan pembaharuan data terkait dengan identitas pemohon.

**h. Extension Points**

Tidak ada *extension points* pada *use case* ini

**i. Special Requirements**

Tidak ada *special requirements* pada *use case* ini

**4. Membuat laporan**

**a. Brief Description**

*Use case* ini menggambarkan ketika pihak Fakultas atau universitas membuat laporan penerima beasiswa.

**b. Basic Flow of Events**

1. *Use case* dimulai ketika fakultas atau universitas masuk kedalam sistem dan laporan

2. Sistem menampilkan data berupa informasi beasiswa dan fungsi yang tersedia untuk melihat laporan.

**{Memilih detail}**

3. Aktor memilih *detail* salah satu beasiswa  
Sistem akan menampilkan *detail* beasiswa dan mahasiswa yang diterima

**{Memilih cetak}**

4. User memilih cetak

5. Sistem akan menampilkan tampilan untuk cetak

**{Use Case Selesai}**

6. *Use Case* membuat laporan selesai

**c. Alternative Flows**

1. Tidak jadi mencetak laporan

Pada seluruh waktu di *basic flow*, user dapat tidak jadi melakukan mencetak laporan dan **{Use Case Selesai}**

**d. Subflow**

Tidak ada *subflow* pada *use case* ini

**e. Key Scenarios**

1. User berhasil mencetak laporan: *Basic flow*

2. Tidak jadi mencetak laporan : *Basic flow*, Tidak jadi mencetak laporan.

**f. Pre-Conditions**

User sudah masuk ke dalam sistem.

**g. Post-Conditions**

User berhasil melakukan pembaharuan data terkait dengan identitas pemohon.

**h. Extension Points**

Tidak ada *extension points* pada *use case* ini

**i. Special Requirements**

Tidak ada *special requirements* pada *use case* ini

**5. Mengelola Data User**

**a. Brief Description**

*Use case* ini menggambarkan ketika pihak Universitas ingin mengelola data *user*.

**b. Basic Flow of Events**

1. *Use case* dimulai ketika pihak universitas masuk kedalam sistem dan memilih mengelola data *user*.
2. Sistem menampilkan data pengguna yang terdapat pada sistem dan fungsi yang tersedia untuk mengelola data pengguna. Fungsi tersebut meliputi tambah *user* , ubah dan hapus *user* dari sistem.

**{Memilih operasi pengolahan data}**

3. Jika aktor memilih operasi untuk tambah *user*, maka sub alur **Tambah user** akan dijalankan.
4. Jika aktor memilih operasi untuk ubah, maka sub alur **Ubah** akan dijalankan.
5. Jika aktor memilih operasi untuk hapus, maka sub alur **hapus** akan dijalankan.

**{Use case selesai}**

6. *Use case* mengelola data pengguna selesai.

**c. Alternative Flows**

1. Data tidak lengkap  
Pada **{Memilih fungsi pengolahan data}** di *basic flow* dan **{Mengisi data pengguna}** pada *subflow*, apabila aktor tidak mengisi seluruh data maka sistem akan menampilkan pesan bahwa terdapat data yang belum diisi.

2. Data sama  
Pada **{Memilih fungsi pengolahan data}** di *basic flow* dan **{Mengisi data pengguna}** pada *subflow*, apabila *username* yang dimasukan oleh aktor sudah terdapat pada sistem maka sistem akan menampilkan pesan terdapat *username* yang sama. aktor dapat melanjutkan pada **{Memilih fungsi pengolahan data}** atau mengganti *username* **{Use case selesai}**.

3. Tidak jadi melakukan pengelolaan data  
Pada seluruh waktu di *basic flow*, aktor dapat tidak jadi melakukan pengelolaan data dan **{Use Case Selesai}**

**d. Subflow**

**Tambah user**

- Aktor memilih operasi untuk tambah *user*

- Sistem meminta agar aktor mengisi data yang dibutuhkan untuk menambahkan pengguna sistem  
**{Mengisi data pengguna}**
- Aktor mengisi data yang dibutuhkan dan menambahkan pengguna pada sistem
- Sistem menampilkan pesan berhasil menambah

#### Ubah

- Aktor memilih operasi untuk ubah pada *user* terdaftar dalam sistem
- Sistem menampilkan *detail* data *user* yang akan diperbaharui
- Aktor memperbaharui data *user* pada sistem dan menyimpan hasil pembaharuan
- Sistem menampilkan pesan data *user* berhasil diubah

#### Hapus

- Aktor memilih data yang akan dihapus dan sistem menampilkan pesan verifikasi
- Aktor melakukan verifikasi penghapusan dan sistem menghapus data pengguna tersebut.
- Sistem menampilkan pesan berhasil menghapus data

#### e. Key Scenarios

1. Aktor berhasil melakukan pengelolaan data : *Basic flow*
2. Terdapat data yang belum diisi : *Basic flow*, Data tidak lengkap
3. *username* sama : *Basic flow*, data sama
4. Aktor tidak jadi melakukan pengelolaan data : *Basic flow*, Tidak jadi melakukan pengelolaan data

#### f. Pre-Conditions

Aktor telah masuk ke dalam sistem.

#### g. Post-Conditions

Aktor berhasil melakukan pengelolaan data *user*.

#### h. Extension Points

Tidak ada *special requirements* pada *use case* ini

#### i. Special Requirements

Tidak ada *special requirements* pada *use case* ini

### 6. Mengelola Data Beasiswa

#### a. Brief Description

*Use case* ini digunakan untuk menggambarkan bagaimana universitas mengatur data beasiswa meliputi penambahan, pengubahan, dan penghapusan data beasiswa.

#### b. Basic Flow of Events

1. *Use case* dimulai ketika pihak universitas masuk kedalam sistem dan memilih beasiswa.
2. Sistem menampilkan data beasiswa yang terdapat pada sistem dan fungsi yang tersedia untuk mengelola data

beasiswa yang terdapat dalam *detail* beasiswa. Fungsi tersebut meliputi tambah *user* , ubah dan hapus beasiswa dari sistem.

**{Memilih operasi pengolahan data}**

3. Jika aktor memilih operasi untuk tambah beasiswa, maka sub alur **Tambah beasiswa** akan dijalankan.
4. Jika aktor memilih operasi untuk ubah, maka sub alur **Ubah** akan dijalankan.
5. Jika aktor memilih operasi untuk hapus, maka sub alur **hapus** akan dijalankan.

**{Use case selesai}**

6. *Use case* mengelola data pengguna selesai.

**c. Alternative Flows**

7. Data tidak lengkap

Pada **{Memilih fungsi pengolahan data}** di *basic flow* dan **{Mengisi data pengguna}** pada *subflow*, apabila aktor tidak mengisi seluruh data maka sistem akan menampilkan pesan bahwa terdapat data yang belum diisi.

8. Tidak jadi melakukan pengelolaan data

Pada seluruh waktu di *basic flow*, aktor dapat tidak jadi melakukan pengelolaan data dan **{Use Case Selesai}**

**d. Subflow**

**Tambah beasiswa**

- Aktor memilih operasi untuk tambah beasiswa
- Sistem meminta agar aktor mengisi data yang dibutuhkan untuk menambahkan beasiswa
- **{Mengisi data pengguna}**
- Aktor mengisi data yang dibutuhkan dan menambahkan beasiswa pada sistem
- Sistem menampilkan pesan berhasil menambah

**Ubah**

- Aktor memilih operasi untuk ubah pada beasiswa yang ada dalam sistem
- Sistem menampilkan *detail* data beasiswa yang akan diperbaharui
- Aktor memperbaharui data beasiswa pada sistem dan menyimpan hasil pembaharuan
- Sistem menampilkan pesan data beasiswa berhasil diubah

**Hapus**

- Aktor memilih data yang akan dihapus dan sistem menampilkan pesan verifikasi
- Aktor melakukan verifikasi penghapusan dan sistem menghapus data beasiswa tersebut.
- Sistem menampilkan pesan berhasil menghapus beasiswa

**e. Key Scenarios**

9. Aktor berhasil melakukan pengelolaan data : *Basic flow*
10. Terdapat data yang belum diisi : *Basic flow*, Data tidak lengkap
11. Aktor tidak jadi melakukan pengelolaan data : *Basic flow*, Tidak jadi melakukan pengelolaan data

**f. Pre-Conditions**

Aktor telah masuk ke dalam sistem.

**g. Post-Conditions**

Aktor berhasil melakukan pengelolaan data beasiswa.

**h. Extension Points**

Tidak ada *special requirements* pada *use case* ini

**i. Special Requirements**

Tidak ada *special requirements* pada *use case* ini

**7. Mengajukan beasiswa**

**a. Brief Description**

*Use case* ini digunakan untuk menggambarkan ketika mahasiswa akan mendaftar menjadi penerima beasiswa. Mahasiswa melakukan pendaftaran dengan memasukkan data yang dibutuhkan.

**b. Basic Flow of Events**

1. *Use case* dimulai ketika mahasiswa membuka sistem dan memilih beasiswa
2. Sistem menampilkan list data beasiswa yang terdapat pada sistem dan fungsi untuk melihat *detail* beasiswa.  
**{Memilih Detail}**
3. Aktor memilih *detail* salah satu beasiswa
4. Sistem menampilkan *detail* keterangan beasiswa yang dipilih pada sistem dan fungsi untuk mendaftar beasiswa.  
**{Memilih Daftar beasiswa}**
5. Aktor memilih daftar beasiswa
6. Sistem meminta aktor untuk mengisi data yang dibutuhkan dan mengunggah berkas yang diperlukan  
**{Mengisi Data dan mengunggah berkas}**
7. aktor mengisi data dan mengunggah berkas melalui sistem.  
**{Melakukan Penyimpanan}**
8. Aktor melakukan penyimpanan dengan menyimpan data yang sudah diisi pada sistem
9. Data berhasil disimpan dan sistem menambahkan data progres
10. Sistem menampilkan pesan berhasil menyimpan  
**{Use Case Selesai}**

**c. Alternative Flows**

1. Terdapat data yang belum diisi  
Pada **{Mengisi Data dan mengunggah berkas}** di *basic flow*, apabila aktor tidak mengisi pada seluruh data yang

dibutuhkan maka sistem akan menampilkan pesan untuk mengisi seluruh data dan melanjutkan *basic flow* pada **{Mengisi Data dan mengunggah berkas}**.

2. Tidak dapat mendaftar  
Pada **{Memilih Daftar beasiswa}** di *basic flow*, aktor tidak bisa memilih mendaftar dikarenakan aktor sudah mendaftar dan **{Use Case Selesai}**
3. Profil belum lengkap  
Pada **{Memilih Daftar beasiswa}** di *basic flow*, aktor tidak bisa memilih mendaftar dikarenakan aktor belum melengkapi profil dan **{Mengubah Profil}**

**d. Subflow**

**Mengubah Profil**

1. Sistem menampilkan *form* pengisian profil.  
**{Mengisi data}**
2. aktor mengisi data profil.  
**{Melakukan Penyimpanan}**
3. Data berhasil disimpan.
4. Sistem menampilkan pesan berhasil menyimpan  
**{Use Case Selesai}**

**e. Key Scenarios**

1. Aktor berhasil melakukan pendaftaran : *Basic flow*
2. Terdapat data yang belum diisi : *Basic flow*, Terdapat data yang belum diisi, *Basic flow*.
3. Tidak dapat mendaftar : *Basic flow*, Tidak dapat mendaftar

**f. Pre-Conditions**

Aktor sudah masuk ke dalam sistem dan teridentifikasi sebagai mahasiswa.

**g. Post-Conditions**

Aktor berhasil mendaftar beasiswa.

**h. Extension Points**

Tidak ada *extension points* pada *use case* ini

**i. Special Requirements**

Tidak ada *special requirements* pada *use case* ini

**8. Mengelola Data Jurusan Fakultas**

**a. Brief Description**

*Use case* ini digunakan untuk menggambarkan bagaimana universitas mengatur data fakultas dan jurusan meliputi penambahan, pengubahan, dan penghapusan data.

**b. Basic Flow of Events**

12. *Use case* dimulai ketika pihak universitas masuk kedalam sistem dan memilih fakultas dan jurusan.

13. Sistem menampilkan data fakultas dan jurusan yang terdapat pada sistem dan fungsi yang tersedia untuk mengelola data beasiswa yang terdapat dalam *detail* beasiswa. Fungsi tersebut meliputi tambah fakultas, tambah jurusan, ubah dan hapus beasiswa dari sistem.

**{Memilih operasi pengolahan data}**

14. Jika aktor memilih operasi untuk tambah fakultas, maka sub alur **Tambah fakultas** akan dijalankan.

15. Jika aktor memilih operasi untuk tambah jurusan, maka sub alur **Tambah jurusan** akan dijalankan.

16. Jika aktor memilih operasi untuk ubah, maka sub alur **Ubah** akan dijalankan.

17. Jika aktor memilih operasi untuk hapus, maka sub alur **hapus** akan dijalankan.

**{Use case selesai}**

18. *Use case* mengelola data pengguna selesai.

**c. Alternative Flows**

1. Tidak jadi melakukan pengelolaan data  
Pada seluruh waktu di *basic flow*, aktor dapat tidak jadi melakukan pengelolaan data dan **{Use Case Selesai}**

**d. Subflow**

**Tambah fakultas**

- Aktor memilih operasi untuk tambah beasiswa
- Sistem meminta agar aktor mengisi data yang dibutuhkan untuk menambahkan fakultas  
**{Mengisi data}**
- Aktor mengisi data yang dibutuhkan dan menambahkan fakultas pada sistem
- Sistem menampilkan pesan berhasil menambah

**Tambah jurusan**

- Aktor memilih operasi untuk tambah jurusan
- Sistem meminta agar aktor mengisi data yang dibutuhkan untuk menambahkan jurusan  
**{Mengisi data}**
- Aktor mengisi data yang dibutuhkan dan menambahkan jurusan pada sistem
- Sistem menampilkan pesan berhasil menambah

**Ubah**

- Aktor memilih operasi untuk ubah pada fakultas atau jurusan yang ada dalam sistem
- Sistem menampilkan *detail* data yang akan diperbaharui
- Aktor memperbaharui data fakultas atau jurusan pada sistem dan menyimpan hasil pembaharuan
- Sistem menampilkan pesan data beasiswa berhasil diubah

**Hapus**

- Aktor memilih data yang akan dihapus dan sistem menampilkan pesan verifikasi
- Aktor melakukan verifikasi penghapusan dan sistem menghapus data fakultas atau jurusan tersebut.
- Sistem menampilkan pesan berhasil menghapus beasiswa

**e. Key Scenarios**

1. Aktor berhasil melakukan pengelolaan data : *Basic flow*
2. Aktor tidak jadi melakukan pengelolaan data : *Basic flow*,  
Tidak jadi melakukan pengelolaan data

**f. Pre-Conditions**

Aktor telah masuk ke dalam sistem.

**g. Post-Conditions**

Aktor berhasil melakukan pengelolaan data beasiswa.

**h. Extension Points**

Tidak ada *special requirements* pada *use case* ini

**i. Special Requirements**

Tidak ada *special requirements* pada *use case* ini

**9. Menseleksi partisipan**

**a. Brief Description**

*Use case* ini menggambarkan ketika pihak Fakultas melihat mahasiswa yang sudah mengajukan diri untuk mendapatkan beasiswa dan menentukan apakah mahasiswa tersebut lolos atau tidak.

**b. Basic Flow of Events**

1. *Use case* dimulai ketika pihak fakultas masuk kedalam sistem dan memilih beasiswa.
2. Sistem menampilkan data beasiswa yang terdapat pada sistem dan fungsi yang tersedia untuk mengelola data participant yang terdapat dalam *detail* beasiswa. Fungsi tersebut mengatur status participant.

**{Memilih detail}**

3. Aktor memilih *detail* salah satu beasiswa yang ada.
4. Sistem akan menampilkan *detail* keterangan beasiswa dan mahasiswa-mahasiswa yang sudah mendaftar.

**{Memilih salah satu mahasiswa}**

5. Aktor memilih salah satu mahasiswa.
6. Sistem akan menampilkan *detail* data pendaftar dan berkas yang bisa diunduh.

**{Memilih aksi}**

7. Aktor memilih aksi terhadap mahasiswa.  
**{Use case selesai}**
8. *Use case* mengelola data pendaftar selesai.

**c. Alternative Flows**

1. Tidak jadi melakukan pengelolaan data

Pada seluruh waktu di *basic flow*, aktor dapat tidak jadi melakukan pengelolaan data dan **{Use Case Selesai}**

**d. Subflow**

Tidak ada *subflow* pada *use case* ini

**e. Key Scenarios**

1. Aktor berhasil melakukan pengelolaan data pendaftar: *Basic flow*
2. Aktor tidak jadi melakukan pengelolaan data pendaftar : *Basic flow*, Tidak jadi melakukan pengelolaan data

**f. Pre-Conditions**

Aktor telah masuk ke dalam sistem.

**g. Post-Conditions**

Aktor berhasil melakukan penyeleksian data pendaftar.

**h. Extension Points**

Tidak ada *special requirements* pada *use case* ini

**i. Special Requirements**

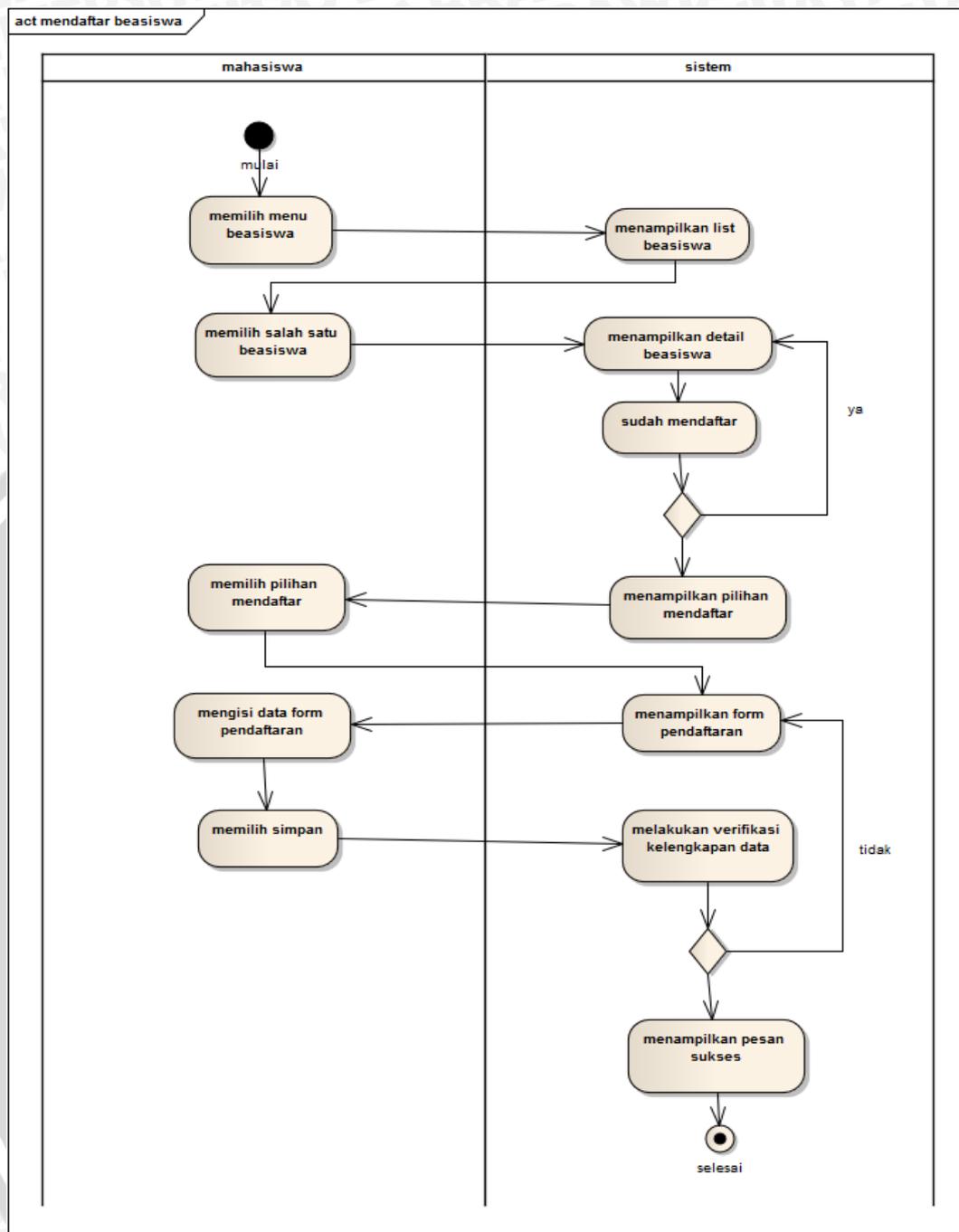
Tidak ada *special requirements* pada *use case* ini

#### 4.4 Perancangan Diagram aktivitas

*Aktivitas diagrams* adalah salah satu cara untuk memodelkan event-event yang terjadi dalam suatu *use case* yang sudah dijelaskan sebelumnya. *Aktivitas diagrams* menggambarkan berbagai aliran aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing aliran berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. Pada tahap perancangan aktivitas tidak semua persyaratan fungsional akan dituliskan hanya menuliskan lima diagram pada kasus ini.

##### 4.4.1 Mendaftar beasiswa

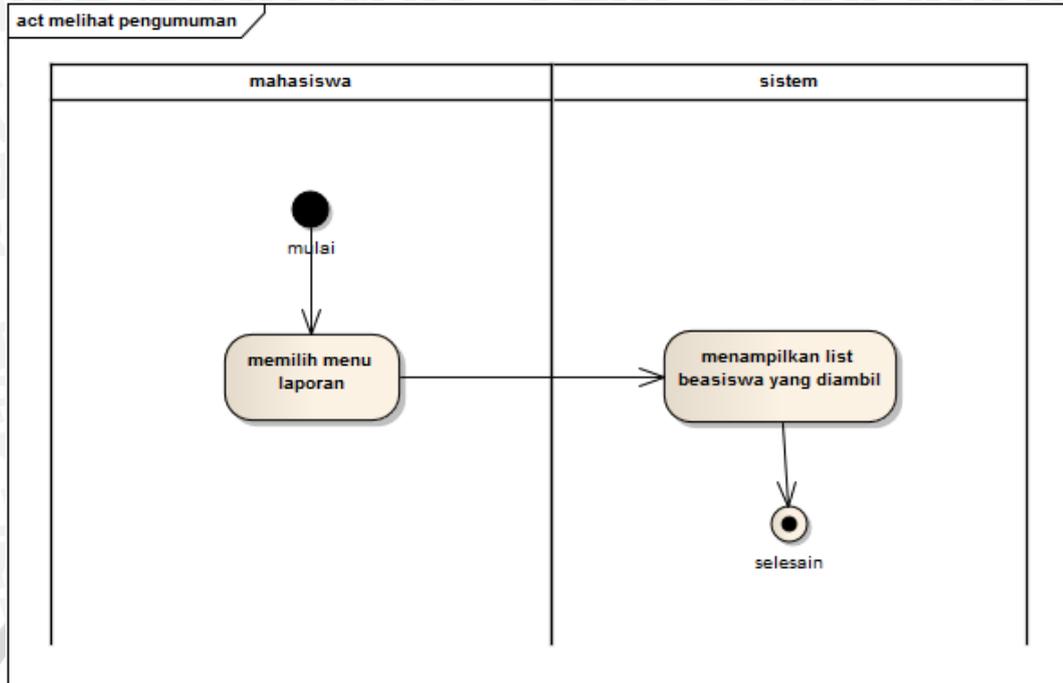
Gambar 4.4 menerangkan aktivitas yang dimulai ketika mahasiswa memilih untuk melakukan pendaftaran beasiswa kemudian mahasiswa melengkapi *form* data pendaftaran dan aktivitas akhir adalah sistem menampilkan pesan berhasil atau mahasiswa melengkapi data ulang.



Gambar 4.4 Aktivitas mendaftar beasiswa

#### 4.4.2 Melihat pengumuman

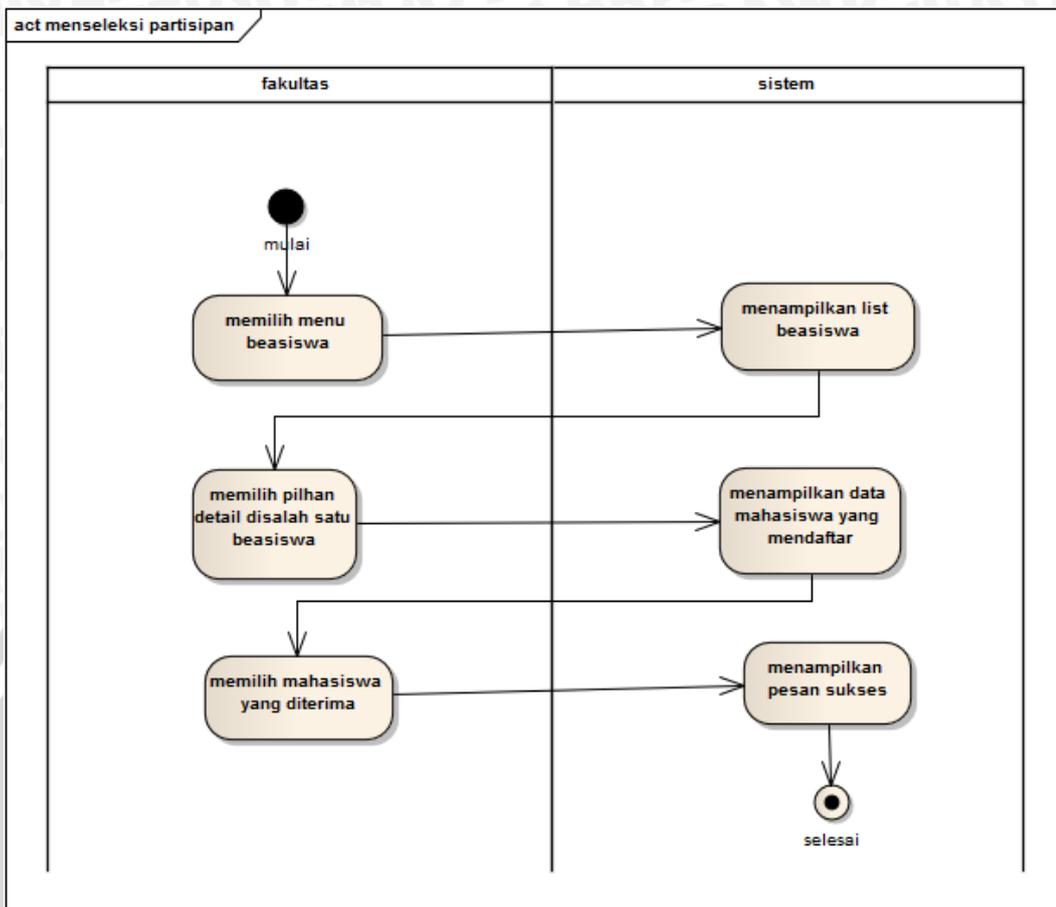
Gambar 4.5 menerangkan aktivitas yang dimulai ketika mahasiswa yang sudah login memilih menu laporan yang kemudian menampilkan hasil pengumuman beasiswa yang sudah atau pernah didaftar.



**Gambar 4.5** Aktivitas Melihat pengumuman

#### 4.4.3 Menseleksi Partisipan

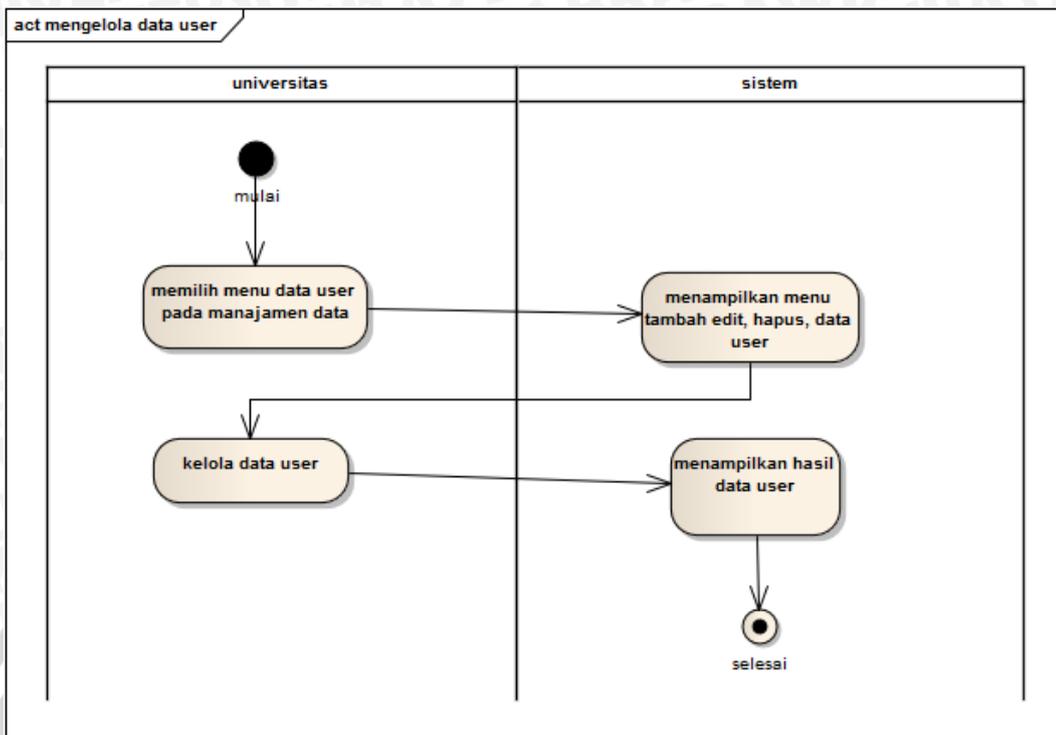
Gambar 4.6 menerangkan aktivitas yang dimulai ketika fakultas yang sudah login memilih menu beasiswa yang kemudian menampilkan list beasiswa yang ada kemudian pengguna memilih *detail* beasiswa yang akan menampilkan mahasiswa yang sudah mendaftar disitu *user* dapat memilih aksi menerima atau menolak mahasiswa.



**Gambar 4.6** Aktivitas Menseleksi Partisipan

#### 4.4.4 Mengelola data *user*

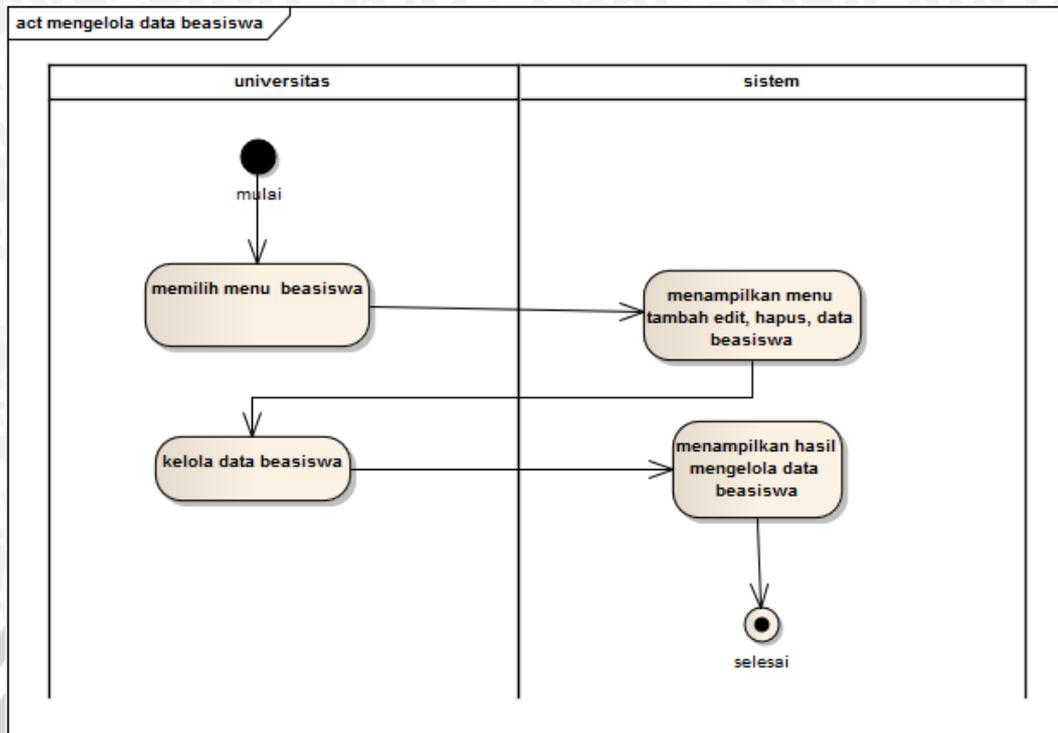
Gambar 4.7 menerangkan aktivitas yang dimulai ketika universitas yang sudah login memilih menu data *user* yang kemudian menampilkan list *user* yang ada, disitu *user* dapat menambah, mengubah atau menghapus *user*.



Gambar 4.7 Aktivitas Mengelola data user

#### 4.4.5 Mengelola data beasiswa

Gambar 4.8 menerangkan aktivitas yang dimulai ketika universitas yang sudah login memilih menu data user yang kemudian menampilkan list user yang ada, disitu user dapat menambah, mengubah atau menghapus user.

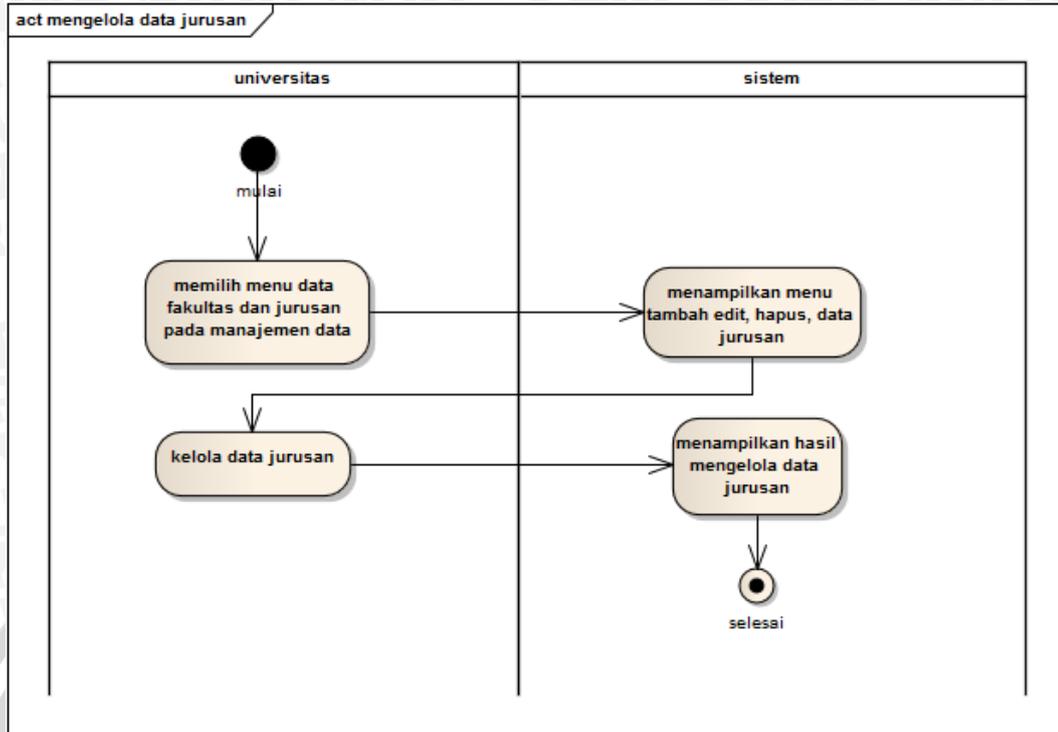


**Gambar 4.8** Aktivitas Mengelola data beasiswa

#### 4.4.6 Mengelola data jurusan dan fakultas

Gambar 4.9 menerangkan aktivitas yang dimulai ketika universitas yang sudah login memilih menu data user yang kemudian menampilkan list user yang ada, disitu user dapat menambah, mengubah atau menghapus user.

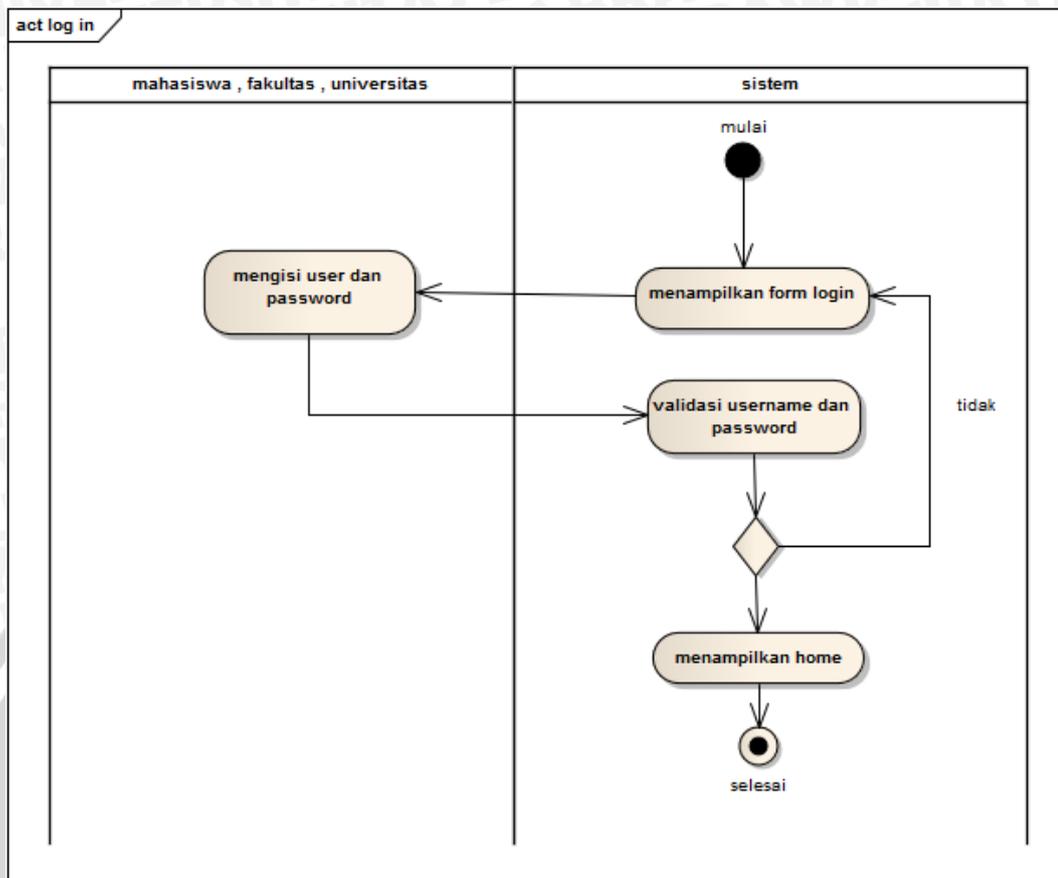




**Gambar 4.9** Aktivitas Mengelola data jurusan

#### 4.4.7 Login

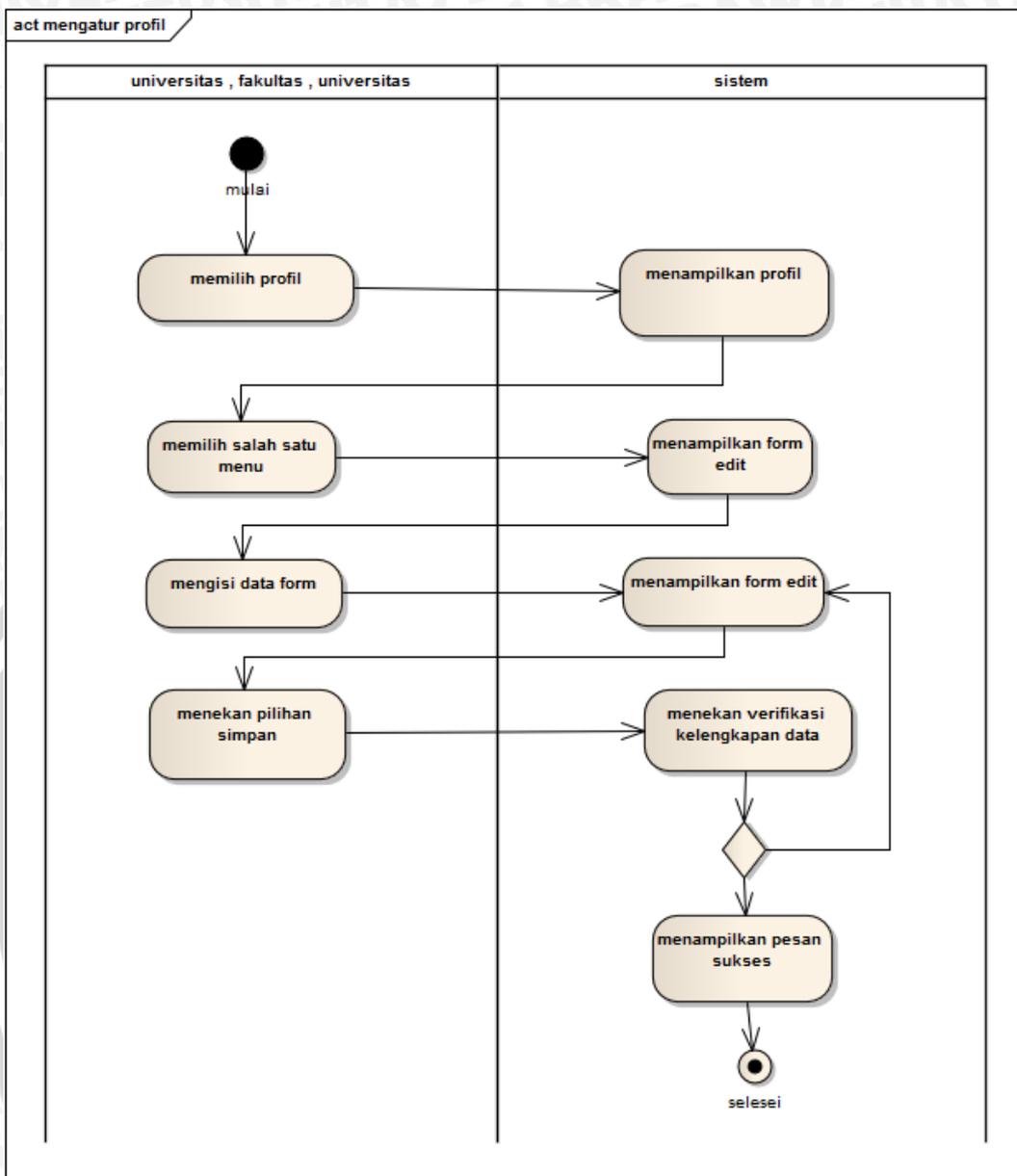
Gambar 4.10 menerangkan aktivitas yang dimulai ketika universitas yang sudah login memilih menu data user yang kemudian menampilkan list user yang ada, disitu user dapat menambah, mengubah atau menghapus user.



Gambar 4.10 Login

#### 4.4.8 Mengatur profil

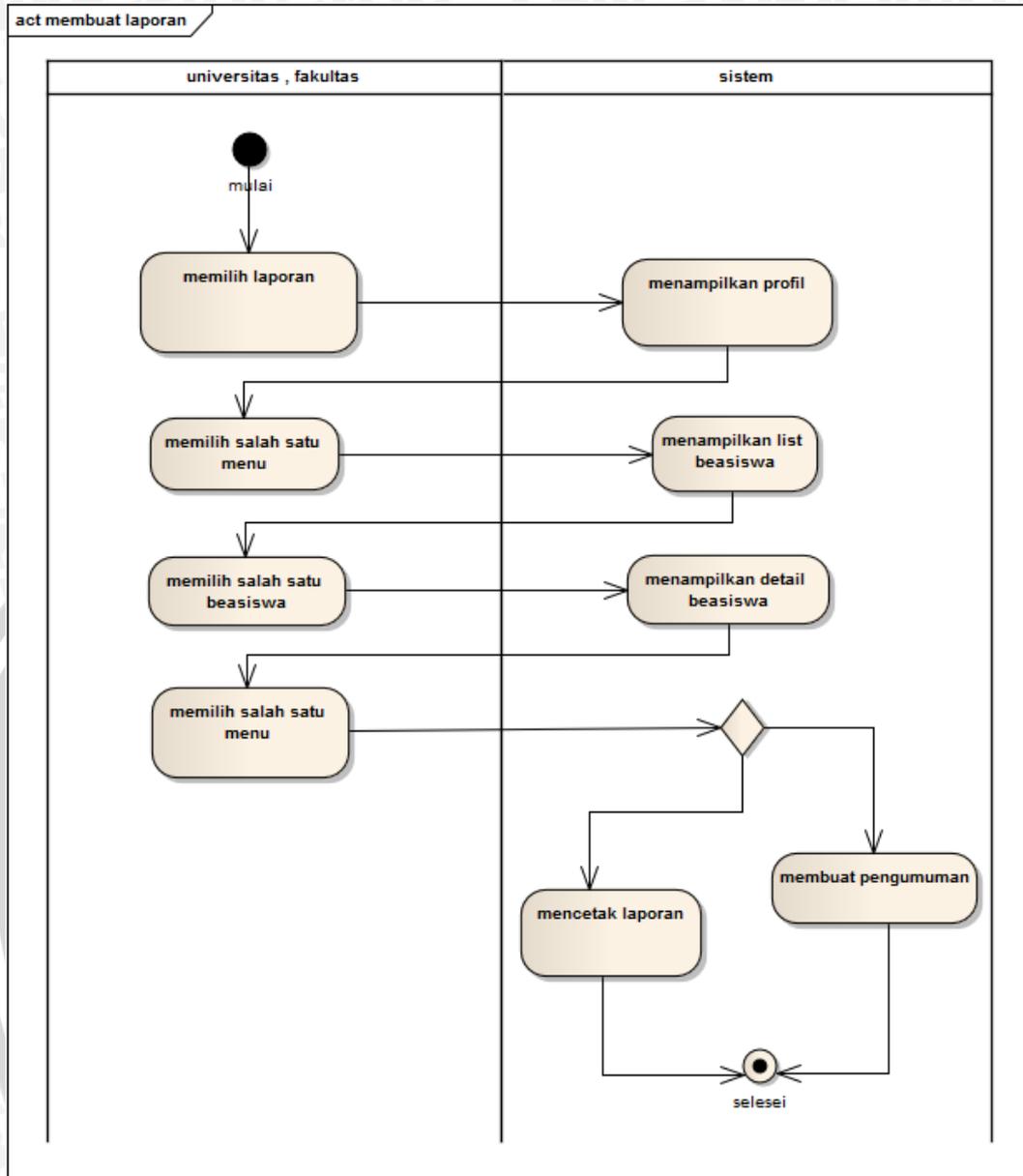
Gambar 4.11 menerangkan aktivitas yang dimulai ketika universitas yang sudah login memilih menu data *user* yang kemudian menampilkan list *user* yang ada, disitu *user* dapat menambah, mengubah atau menghapus *user*.



Gambar 4.11 Aktivitas Mengatur profil

#### 4.4.9 Membuat laporan

Gambar 4.12 menerangkan aktivitas yang dimulai ketika universitas yang sudah login memilih menu data user yang kemudian menampilkan list user yang ada, disitu user dapat menambah, mengubah atau menghapus user.



Gambar 4.12 Aktivitas Membuat laporan

#### 4.5 Pemodelan kelas

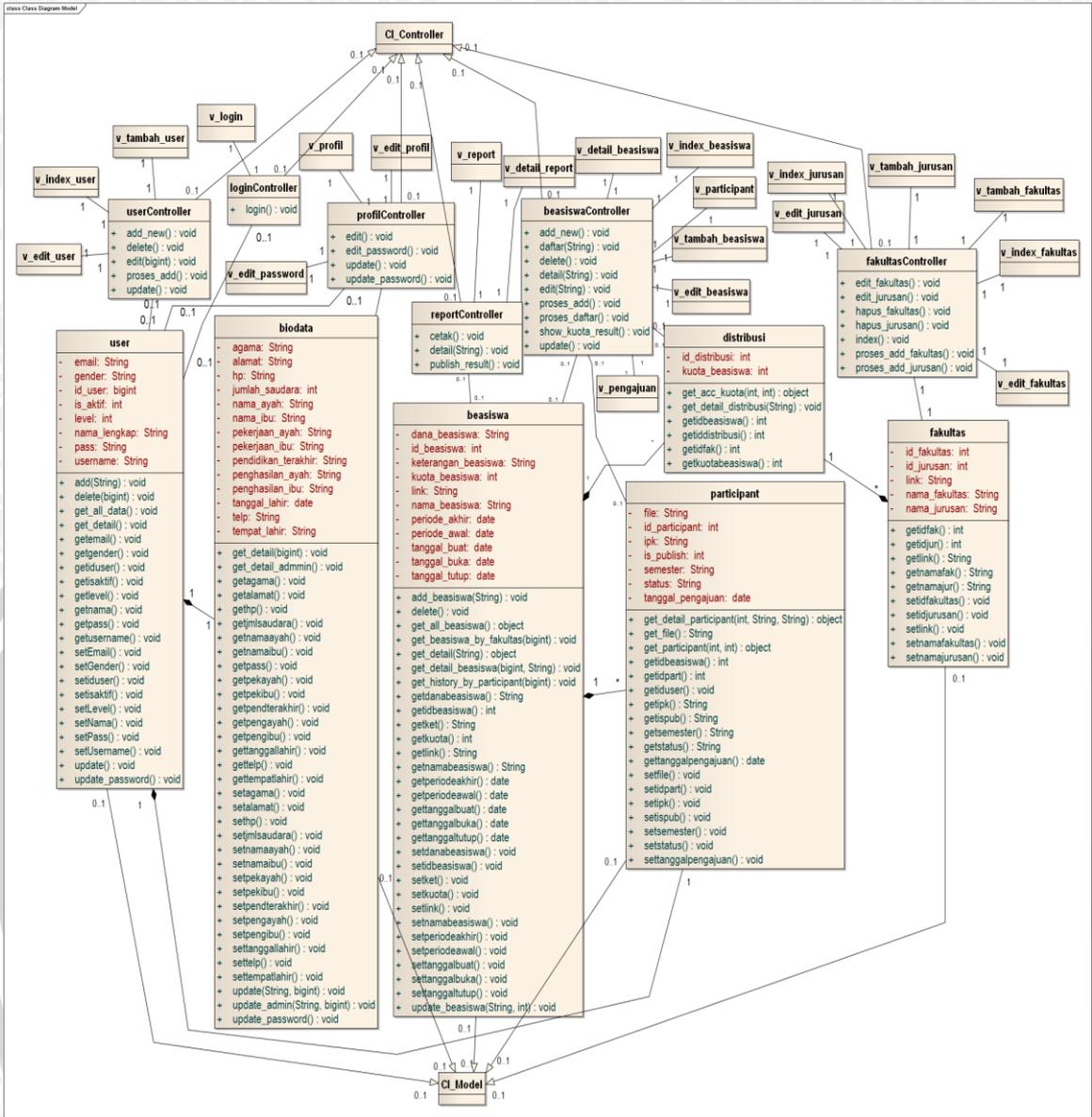
Pada tahap ini didapatkan kelas-kelas dengan menganalisis secara *detail* terhadap *use case* yang dimodelkan. Perancangan kelas ini dilakukan sebanyak 2 kali iterasi sesuai dengan perubahan pada persyaratan sistem. Penambahan kelas dapat dilihat pada tabel 4.10 sedangkan keseluruhan pemodelan kelas pada gambar 4.13

**Tabel 4.10** Penambahan kelas pada iterasi ke 1

Nama Kelas	Nama Method
reportController	+ publish_result()

**Tabel 4.11** Penambahan kelas pada iterasi ke 2

Nama Kelas	Nama Method
beasiswa	-link: String +setlink() +getlink()
fakultas	-link: String +setlink() +getlink()
participant	+get_detail_participant(int,String,String )
beasiswa	+get_detail_beasiswa(bigint,String)



Gambar 4.13 Kelas Diagram

4.5.1.1 Penjelasan dan Tipe Kelas

Tabel 4.12 Kelas userController

Nama Kelas	UserController
Deskripsi	Kelas yang menjembatani antara entity dan boundary yang berkaitan dengan akun user terkait data user pada sistem
Tipe Kelas	Control

Tabel 4.13 Kelas loginController

Nama Kelas	loginController
Deskripsi	Kelas yang menjembatani antara entity dan boundary yang berkaitan dengan login masuk ke sistem

Tipe Kelas	Control
------------	---------

**Tabel 4.14** Kelas profilController

Nama Kelas	profilController
Deskripsi	Kelas yang menjembatani antara entity dan boundary yang berkaitan dengan biodata <i>user</i> dan penggantian <i>password user</i>
Tipe Kelas	Control

**Tabel 4.15** Kelas reportController

Nama Kelas	reportController
Deskripsi	Kelas yang menjembatani antara entity dan boundary yang berkaitan dengan pembuatan laporan pengumuman
Tipe Kelas	Control

**Tabel 4.16** Kelas beasiswaController

Nama Kelas	beasiswaController
Deskripsi	Kelas yang menjembatani antara entity dan boundary yang berkaitan dengan beasiswa mulai dari mengatur data beasiswa sampai pendaftaran beasiswa
Tipe Kelas	Control

**Tabel 4.17** Kelas fakultasController

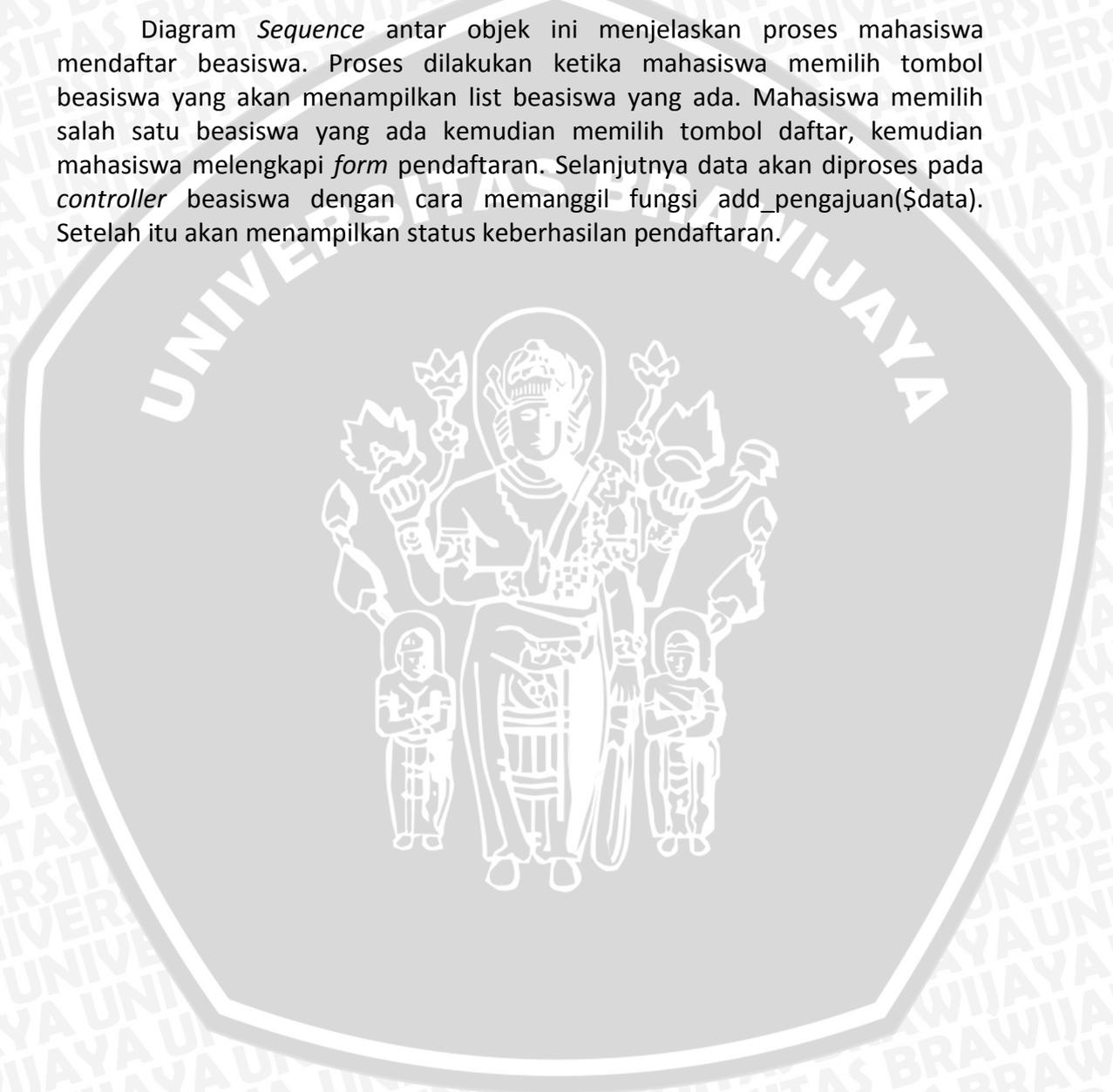
Nama Kelas	fakultasController
Deskripsi	Kelas yang menjembatani antara entity dan boundary yang berkaitan dengan data fakultas dan jurusan
Tipe Kelas	Control

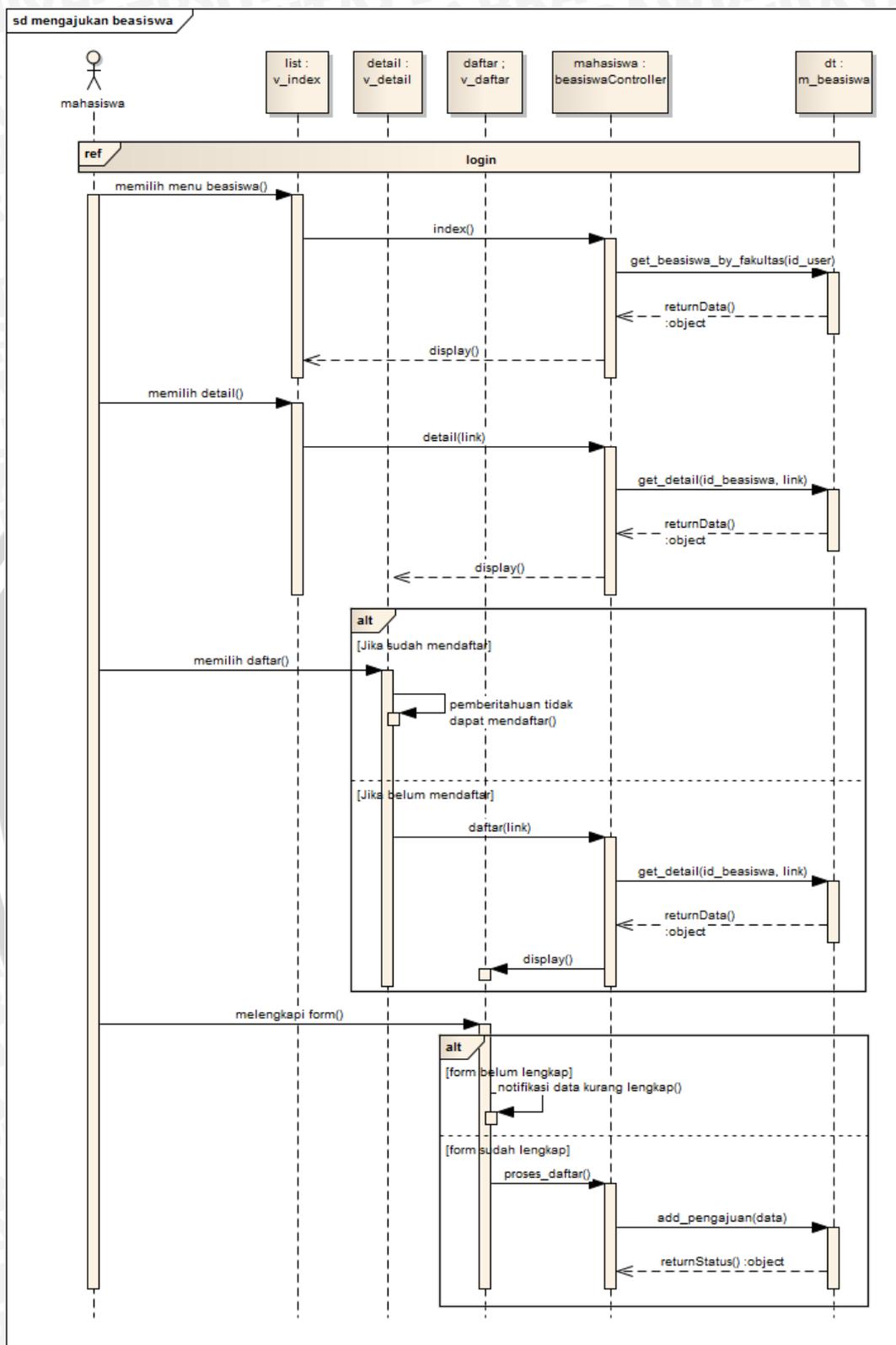
## 4.6 Perancangan Interaksi Antar Objek

*Sequence* diagram menampilkan interaksi antar objek pada setiap usecase berdasarkan urutan waktu.

### 4.6.1 Diagram *Sequence* Mengajukan Beasiswa

Diagram *Sequence* antar objek ini menjelaskan proses mahasiswa mendaftar beasiswa. Proses dilakukan ketika mahasiswa memilih tombol beasiswa yang akan menampilkan list beasiswa yang ada. Mahasiswa memilih salah satu beasiswa yang ada kemudian memilih tombol daftar, kemudian mahasiswa melengkapi *form* pendaftaran. Selanjutnya data akan diproses pada *controller* beasiswa dengan cara memanggil fungsi `add_pengajuan($data)`. Setelah itu akan menampilkan status keberhasilan pendaftaran.



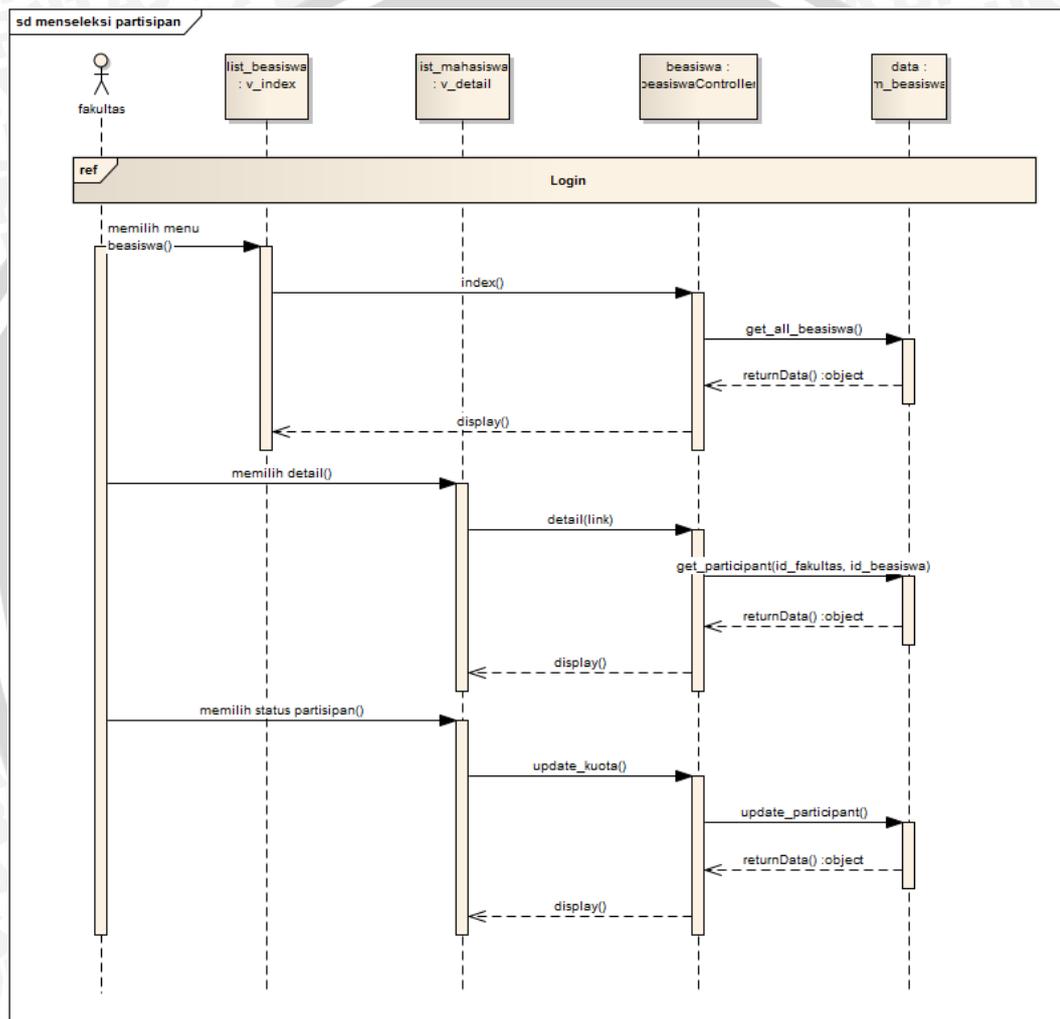


Gambar 4.14 Sequence mengajukan beasiswa



#### 4.6.2 Diagram Sequence Menyeleksi Partisipan

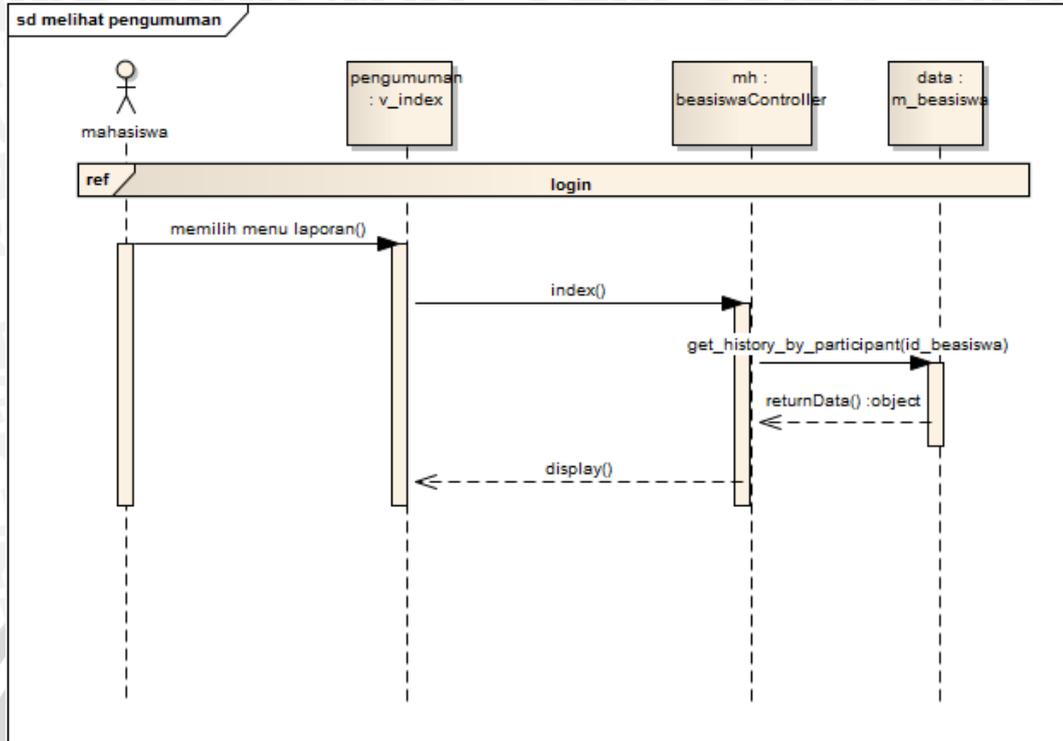
Diagram *Sequence* ini menjelaskan proses fakultas dalam menyeleksi partisipan. Proses dilakukan ketika aktor memilih tombol beasiswa yang akan menampilkan list beasiswa yang ada. Aktor memilih salah satu beasiswa yang ada kemudian akan menampilkan list mahasiswa yang sudah mendaftar, kemudian aktor memilih mahasiswa yang diterima. Mahasiswa yang diterima akan mengurangi kuota yang ada .



Gambar 4.15 Sequence Menyeleksi partisipan

#### 4.6.3 Diagram Sequence Melihat Pengumuman

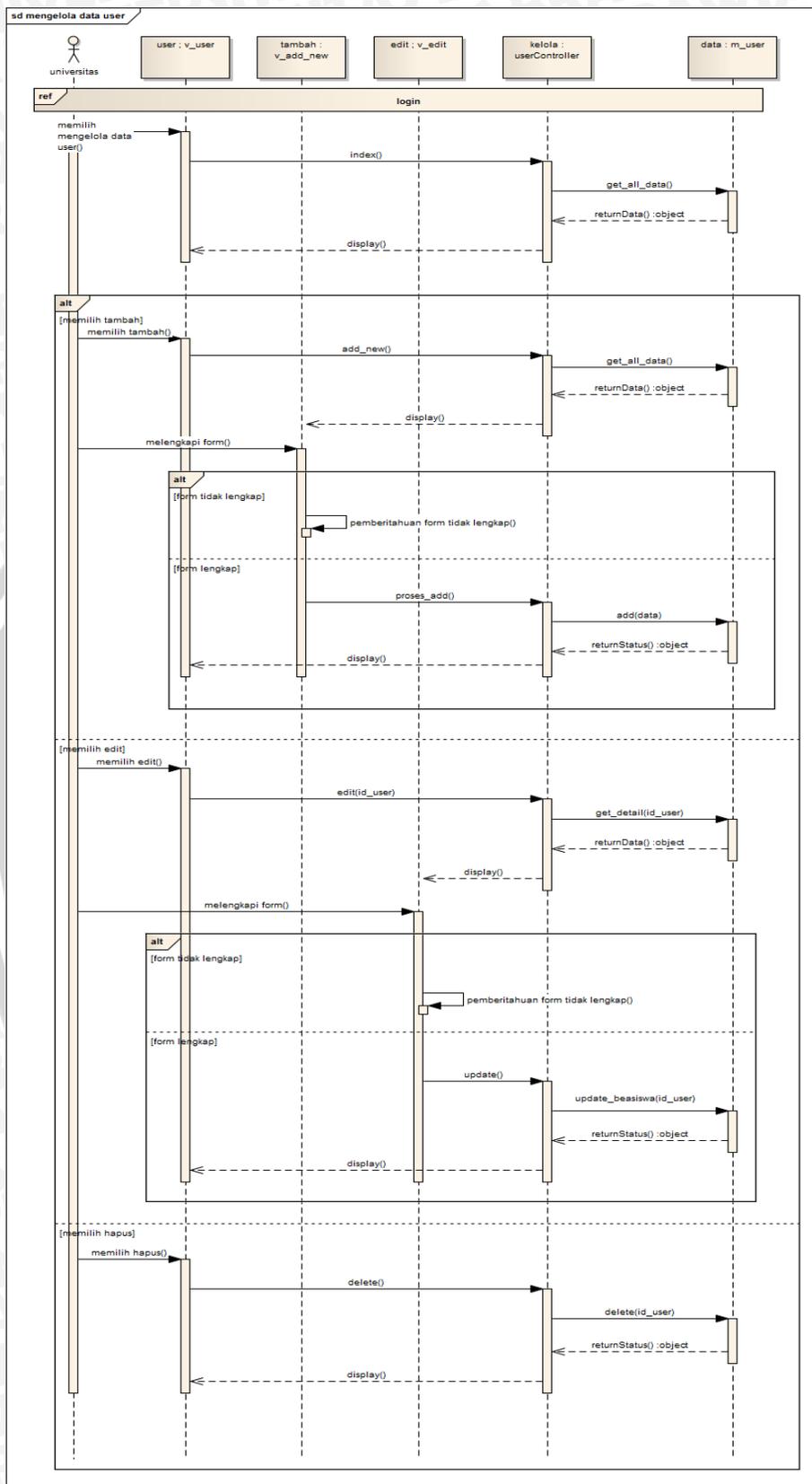
Diagram *Sequence* ini menjelaskan proses mahasiswa dalam melihat pengumuman. Proses dilakukan ketika aktor memilih tombol laporan yang akan menampilkan history pendaftaran. proses ini pada *controller* beasiswa dengan cara memanggil fungsi `get_history_by_participant($id_basiswa)` yang kemudian menampilkan data history pendaftaran beserta hasilnya.



Gambar 4.16 Sequence Melihat pengumuman

#### 4.6.4 Diagram Sequence Mengelola User

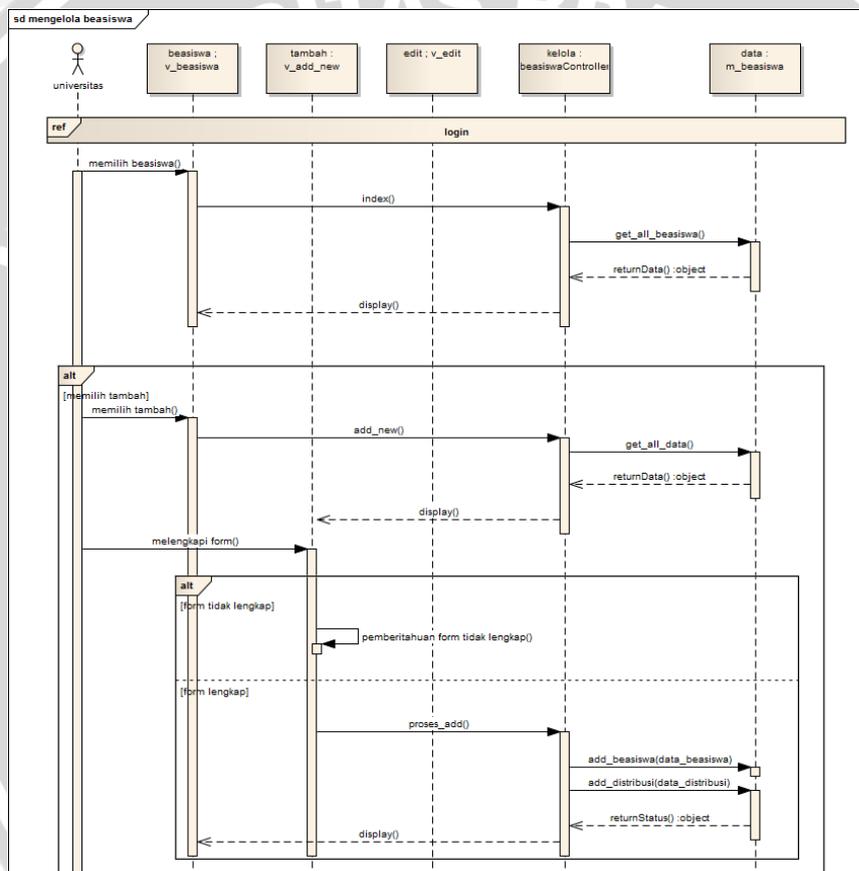
Diagram Sequence ini menjelaskan proses universitas dalam mengelola data user. Proses dilakukan ketika aktor memilih tombol user yang akan menampilkan pilihan tambah, ubah dan hapus. Aktor memilih salah satu pilihan yang ada, kemudian akan diterima oleh controller user. Disitu akan menampilkan halaman view sesuai dengan yang aktor pilih. Ketika memilih tambah akan menampilkan v\_add\_new yang kemudian aktor melengkapi form setelah itu controller memanggil fungsi add(\$data), ketika memilih ubah akan menampilkan v\_ubah kemudian aktor melengkapi form kemudian controller akan memanggil fungsi update(\$id\_user) dan ketika memilih hapus controller akan memanggil fungsi delete(\$id\_user).



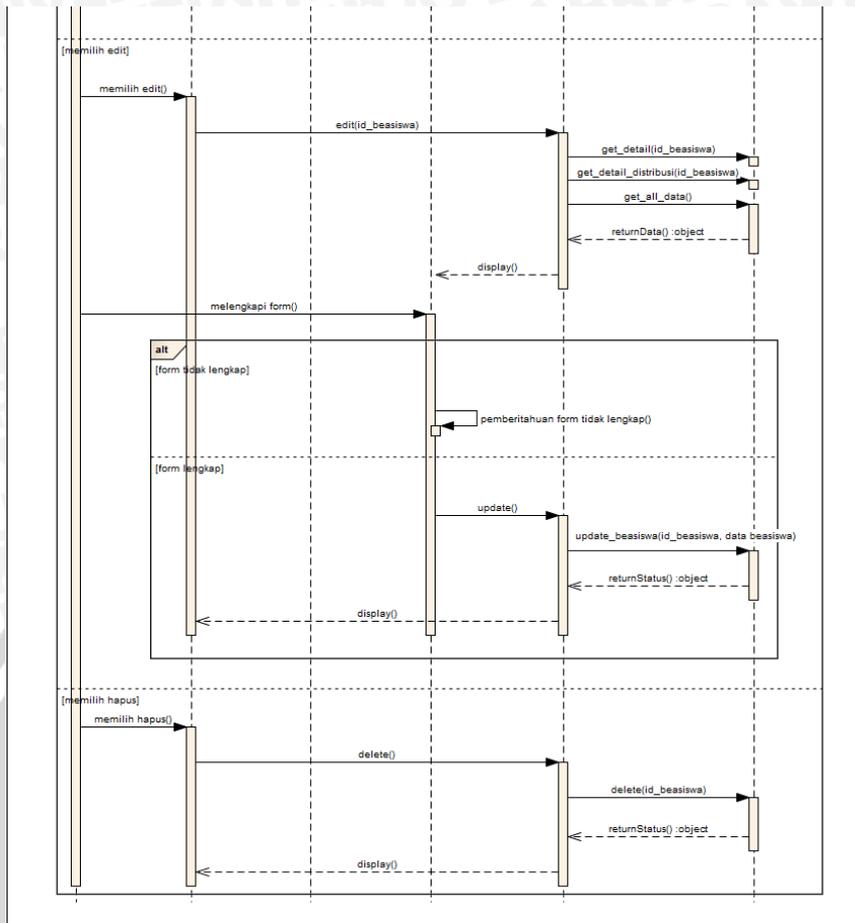
Gambar 4.17 Sequence mengelola user

#### 4.6.5 Diagram Sequence Mengelola Beasiswa

Diagram *Sequence* ini menjelaskan proses universitas dalam mengelola data beasiswa. Proses dilakukan ketika aktor memilih tombol beasiswa yang akan menampilkan pilihan tambah, ubah dan hapus. Aktor memilih salah pilihan yang ada, kemudian akan diterima oleh *controller* beasiswa. Disitu akan menampilkan halaman view sesuai dengan yang aktor pilih. Ketika memilih tambah akan menampilkan *v\_add\_new* yang kemudian aktor melengkapi *form* setelah itu *controller* memanggil fungsi `add_basiswa($data_basiswa)` dan `add_distribusi($data_distribusi)`, ketika memilih ubah akan menampilkan *v\_ubah* kemudian aktor melengkapi *form* kemudian *controller* akan memanggil fungsi `update_basiswa($id_basiswa)` dan ketika memilih hapus *controller* akan memanggil fungsi `delete($id_basiswa)`.



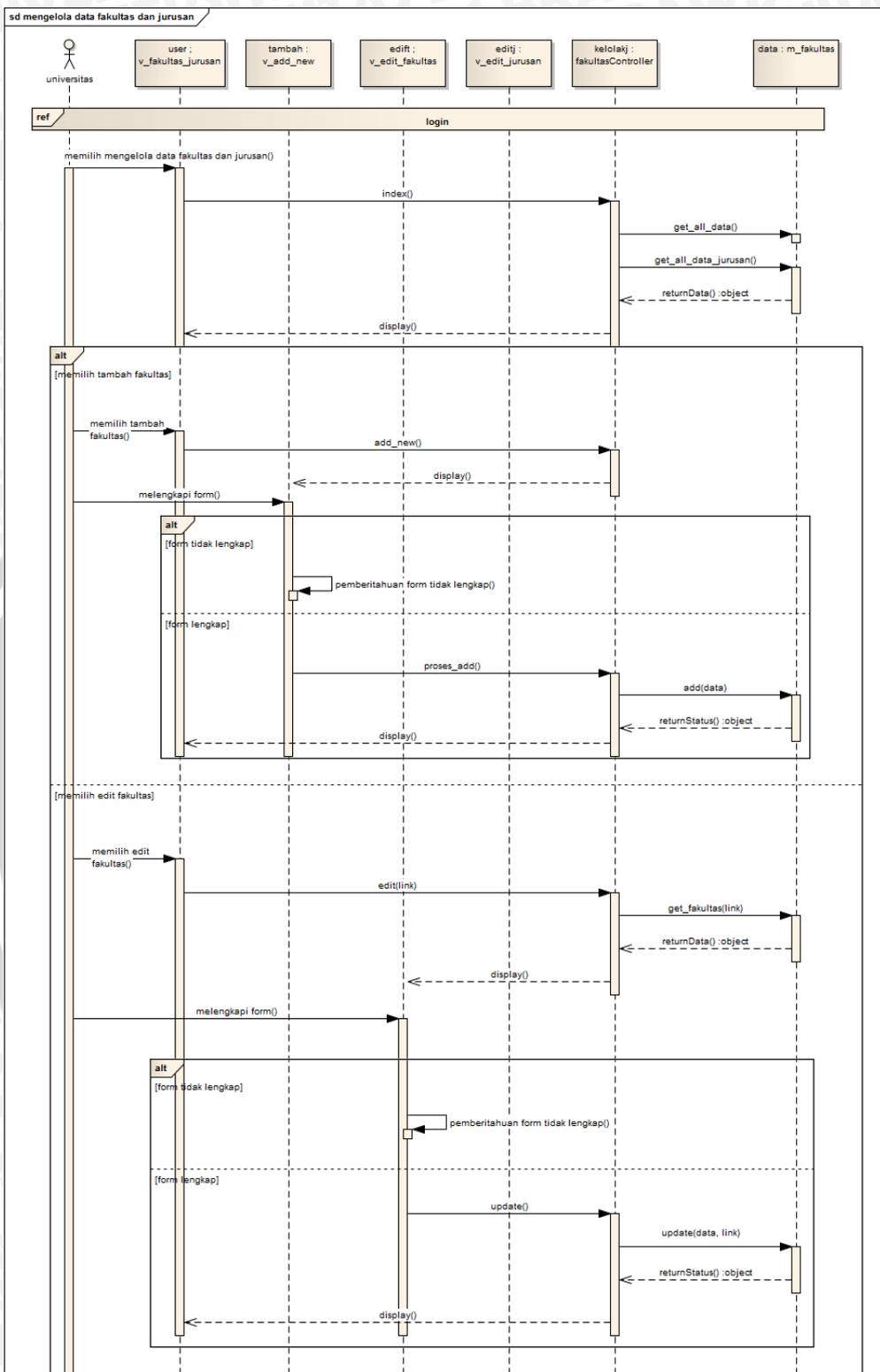
Gambar 4.18 Sequence mengelola beasiswa



Gambar 4.18 Sequence mengelola beasiswa(lanjutan)

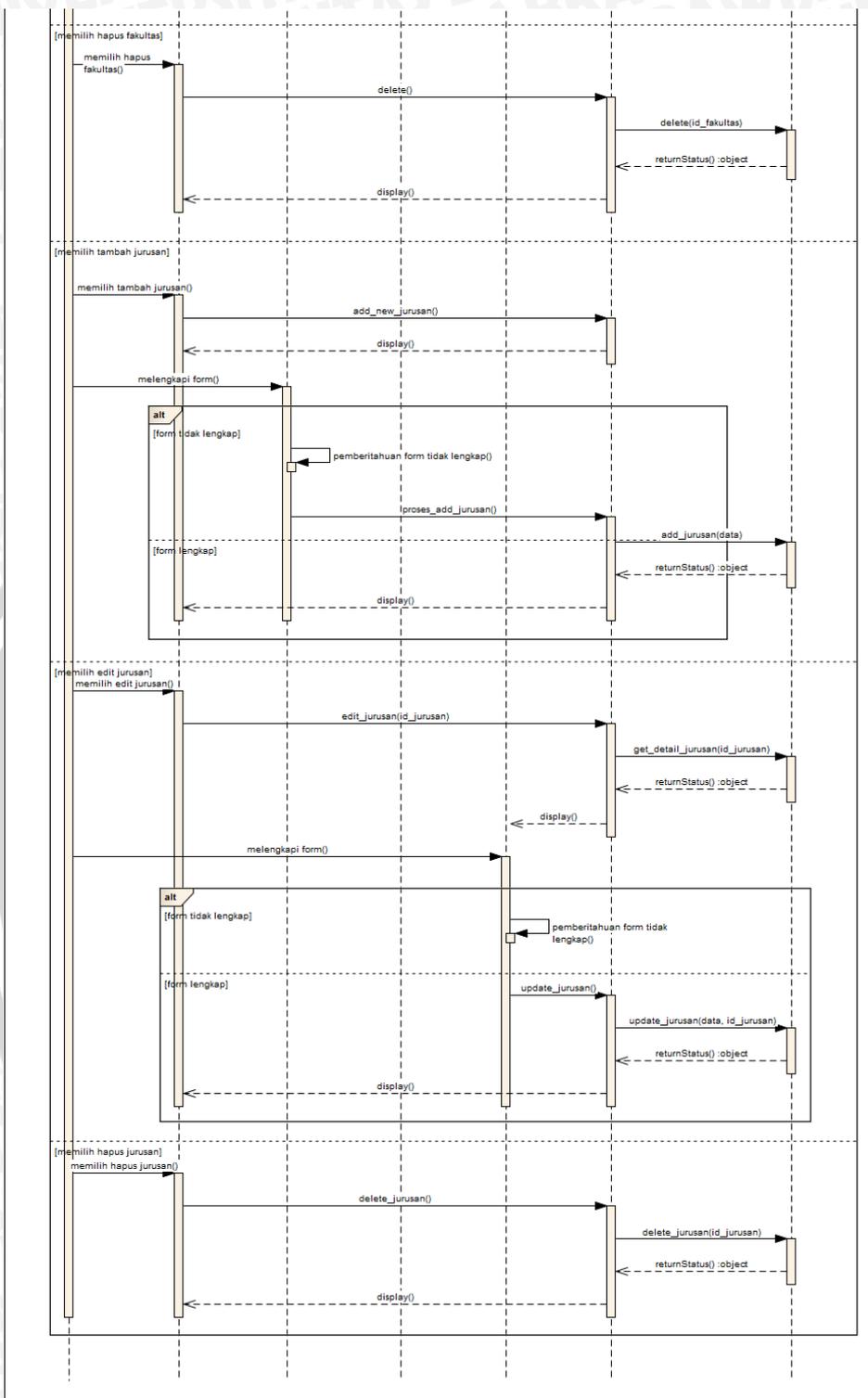
#### 4.6.6 Diagram Sequence Mengelola data fakultas dan jurusan

Diagram *Sequence* ini menjelaskan proses universitas dalam mengelola data fakultas dan jurusan. Proses dilakukan ketika aktor memilih tombol manajemen data fakultas dan jurusan yang akan menampilkan pilihan tambah, ubah dan hapus. Aktor memilih salah pilihan yang ada, kemudian akan diterima oleh *controller* beasiswa. Disitu akan menampilkan halaman view sesuai dengan yang aktor pilih. Ketika memilih tambah akan menampilkan *v\_add\_new* yang kemudian aktor melengkapi *form* setelah itu *controller* memanggil fungsi *add\_new(\$data)*, ketika memilih ubah akan menampilkan *v\_ubah* kemudian aktor melengkapi *form* kemudian *controller* akan memanggil fungsi *update(\$data)* dan ketika memilih hapus *controller* akan memanggil fungsi *delete(\$id\_fakultas) / delete(\$id\_jurusan)*.



Gambar 4.19 Sequence mengelola data jurusan dan fakultas bagian 1



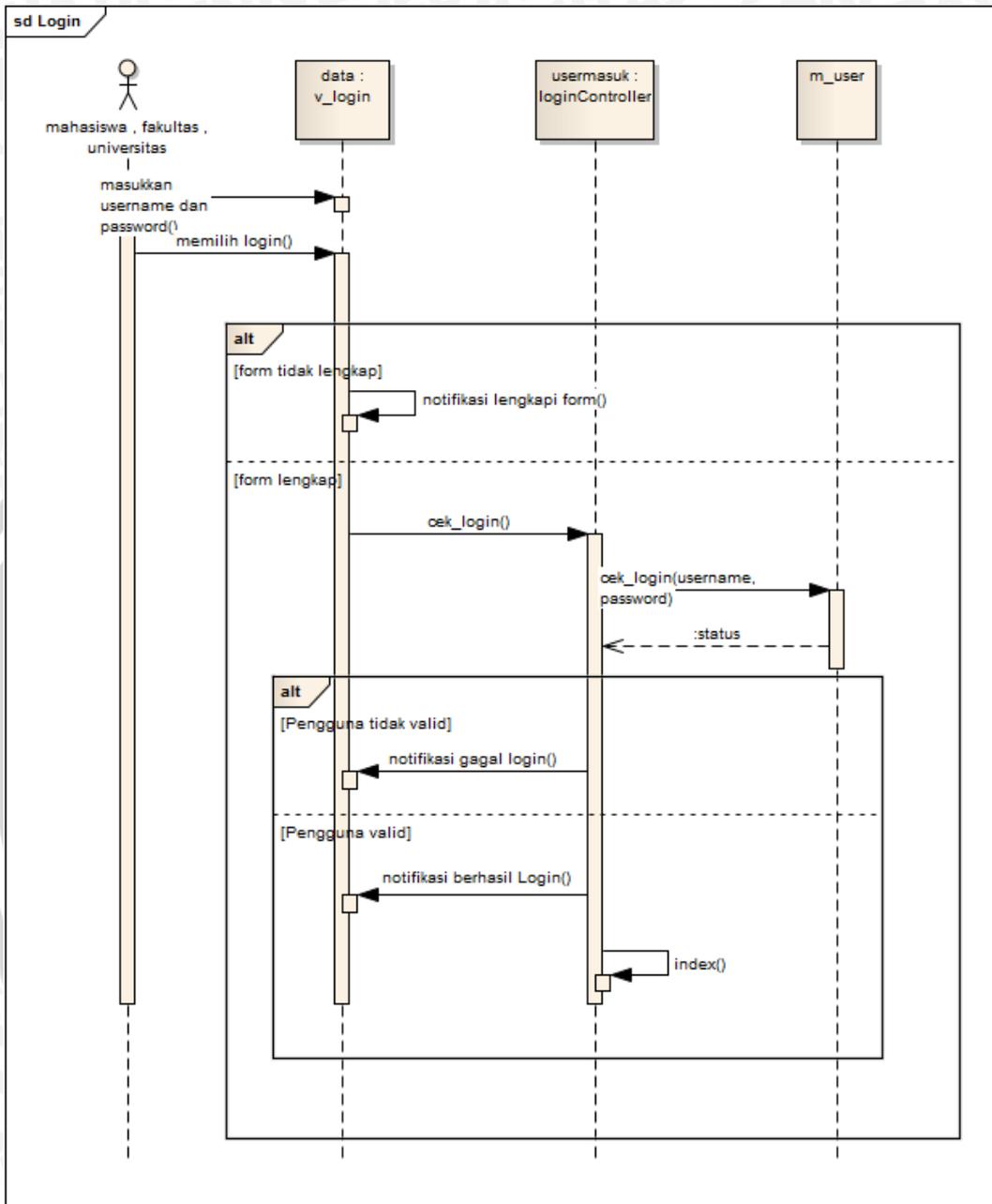


Gambar 4.20 Sequence mengelola data jurusan dan fakultas bagian 2

#### 4.6.7 Diagram Sequence Login

Diagram Sequence ini menjelaskan proses aktor dalam login kedalam sistem. Proses dilakukan ketika actor memasukkan *username* dan *password* dan memilih tombol login. proses ini pada *controller* login dengan cara memanggil

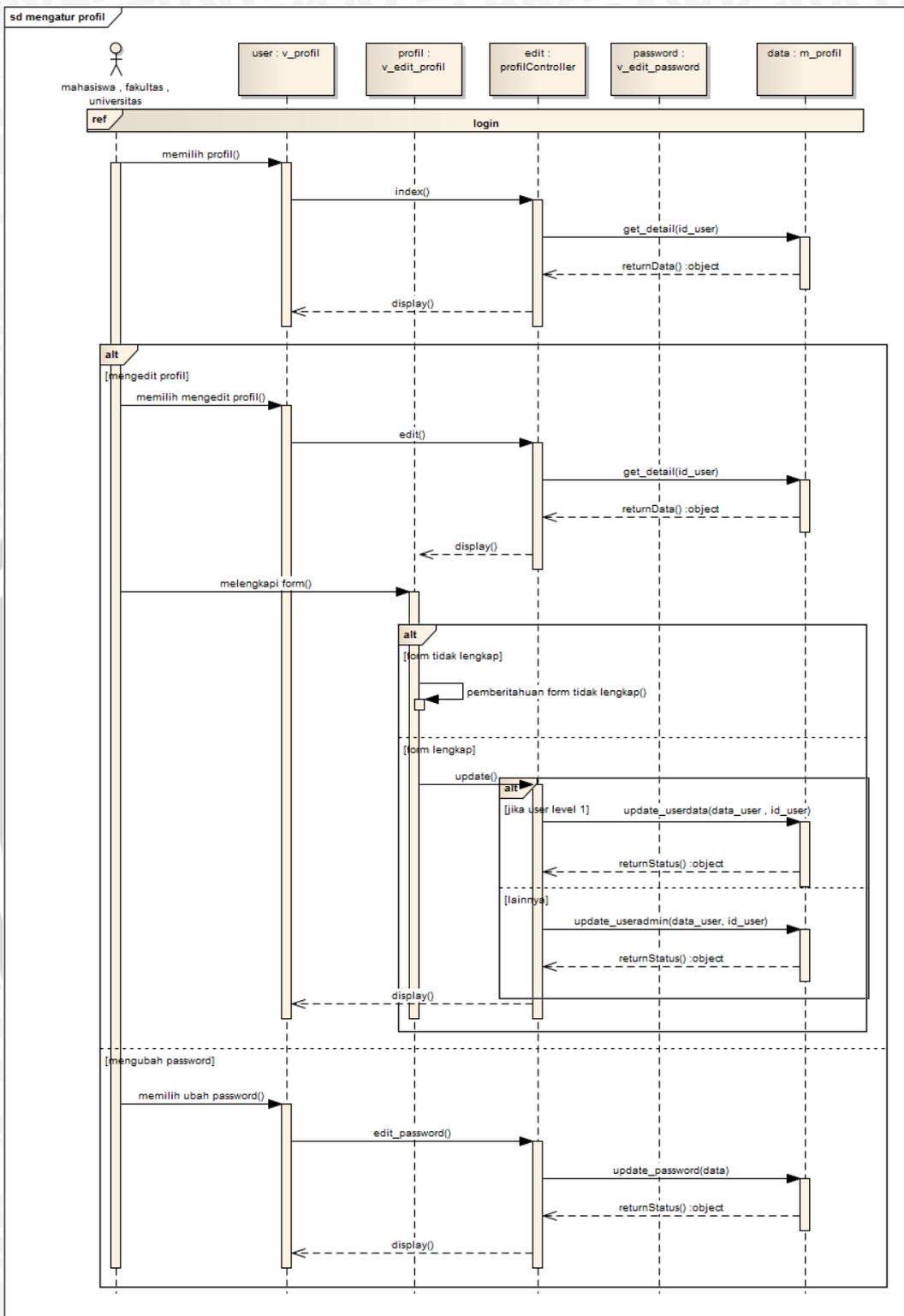
fungsi `cek_login($username,$password)` yang kemudian dialihkan ke halaman home.



Gambar 4.20 Sequence login

#### 4.6.8 Diagram Sequence Mengatur Profil

Diagram *Sequence* ini menjelaskan proses aktor dalam Mengatur profil. Proses dilakukan ketika aktor memilih tombol profil yang akan menampilkan profil. proses ini pada *controller* profil dengan cara memanggil fungsi `get_detail($id_user)` kemudian actor dapat mengubah profil atau mengganti *password* yang akan membuat *controller* menampilkan *form* dan memanggil fungsi `update_userdata($data_user,$id_user)` atau `update_password($data)`

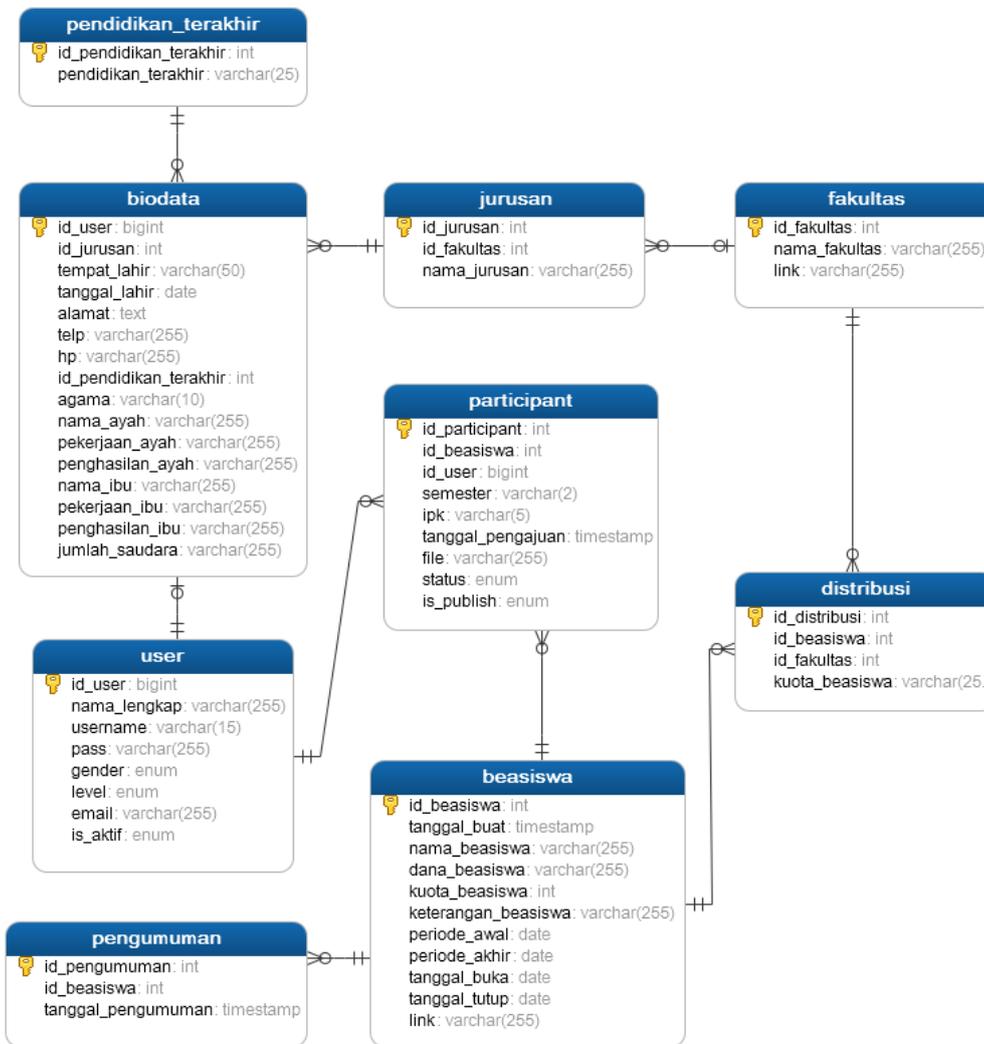


Gambar 4.21 Sequence mengatur profil



## 4.7 Data model

Data model pada aplikasi yang akan dibangun dapat dilihat pada Gambar 4.22.



Gambar 4.22 Data model

### 4.7.1.1 Tabel beasiswa

Nama tabel : beasiswa  
Jumlah *field* : 11  
Fungsi : Untuk menyimpan data beasiswa

**Tabel 4.18** Penjelasan tabel beasiswa

Nama field	Tipe	lebar	Keterangan
id_beasiswa	int	11	Id beasiswa
tanggal_buat	timestamp	-	Tanggal pembuatan
nama_beasiswa	varchar	255	Nama beasiswa
dana_beasiswa	varchar	255	Dana beasiswa
kuota_beasiswa	Int	11	Kuota beasiswa
keterangan_beasiswa	varchar	255	Keterangan beasiswa
periode_awal	Date	-	Periode awal beasiswa
periode_akhir	date	-	Periode akhir beasiswa
tanggal_buka	date	-	Tanggal buka pendaftaran
tanggal_tutup	date	-	Tanggal tutup pendaftaran
link	varchar	255	Singkatan beasiswa

**4.7.1.2 Tabel biodata**

Nama tabel : biodata  
 Jumlah *field* : 16  
 Fungsi : Untuk menyimpan data biodata *user*

**Tabel 4.19** Penjelasan tabel biodata

Nama field	Tipe	lebar	Keterangan
id_user	bigint	20	Id user
id_jurusan	int	11	Id jurusan
tempat_lahir	varchar	50	Tempat lahir
tanggal_lahir	Date	-	Tanggal lahir



**Tabel 4.19** Penjelasan tabel biodata(lanjutan)

Nama field	Tipe	lebar	Keterangan
Alamat	text	11	Alamat
telp	varchar	255	No telp
hp	varchar	255	No handphone
id_pendidikan_terakhir	int	11	Id Pendidikan terakhir
agama	varchar	10	agama
nama_ayah	varchar	255	Nama ayah
pekerjaan_ayah	varchar	255	Pekerjaan ayah
penghasilan_ayah	varchar	255	Penghasilan ayah
nama_ibu	varchar	255	Nama ibu
pekerjaan_ibu	varchar	255	Pekerjaan ibu
penghasilan_ibu	varchar varchar	255	Penghasilan ibu
jumlah_saudara		255	Jumlah saudara

**4.7.1.3 Tabel participant**

Nama tabel : participant  
 Jumlah field : 9  
 Fungsi : Untuk menyimpan data partisipan beasiswa

**Tabel 4.20** Penjelasan tabel partisipan

Nama field	Tipe	lebar	Keterangan
id_participant	Int	11	Id partisipan
id_basiswa	Int	11	Id beasiswa
id_user	Bigint	20	Id user
Semester	Varchar	2	semester



**Tabel 4.20** Penjelasan tabel partisipan

Nama field	Tipe	lebar	Keterangan
ipk	Varchar	5	ipk
tanggal_pengajuan	Timestamp	-	Tanggal mendaftar
file	varchar	255	file
status	enum	-	Status partisipan
is_publish	enum	-	Status pengumuman

**4.7.1.4** Tabel user

Nama tabel : *user*  
 Jumlah field : 18  
 Fungsi : Untuk menyimpan data *user*

**Tabel 4.21** Penjelasan tabel *user*

Nama field	Tipe	lebar	Keterangan
<i>id_user</i>	bigint	20	Id <i>user</i>
<i>nama_lengkap</i>	varchar	255	Nama lengkap
<i>username</i>	varchar	15	<i>Username</i>
<i>pass</i>	varchar	255	<i>Password</i>
<i>gender</i>	enum	-	Kelamin
<i>Level</i>	enum	-	Level <i>user</i>
<i>email</i>	varchar	255	<i>email</i>
<i>is_aktif</i>	enum	-	Status <i>user</i>

**4.7.1.5** Tabel pengumuman

Nama tabel : pengumuman  
 Jumlah field : 3  
 Fungsi : Untuk menyimpan data pengumuman



**Tabel 4.22** Penjelasan tabel pengumuman

Nama field	Tipe	lebar	Keterangan
id_pengumuman	Int	11	Id pengumuman
id_basiswa	Int	11	Id beasiswa
tanggal_pengumuman	timestamp	-	Tanggal pengumuman

**4.7.1.6 Tabel distribusi**

Nama tabel : distribusi  
 Jumlah field : 4  
 Fungsi : Untuk menyimpan data distribusi

**Tabel 4.23** Penjelasan tabel distribusi

Nama field	Tipe	lebar	Keterangan
id_distribusi	Int	11	Id distribusi
id_basiswa	Int	11	Id beasiswa
id_fakultas	Int	11	Id fakultas
kuota_basiswa	varchar	255	Kuota beasiswa

**4.7.1.7 Tabel fakultas**

Nama tabel : fakultas  
 Jumlah field : 3  
 Fungsi : Untuk menyimpan data distribusi

**Tabel 4.24** Penjelasan tabel fakultas

Nama field	Tipe	lebar	Keterangan
id_fakultas	int	11	Id fakultas
nama_fakultas	varchar	255	Nama fakultas
link	varchar	255	Singkatan fakultas

**4.7.1.8 Tabel jurusan**

Nama tabel : jurusan  
 Jumlah field : 3  
 Fungsi : Untuk menyimpan data distribusi



**Tabel 4.25** Penjelasan tabel jurusan

Nama field	Tipe	lebar	Keterangan
id_jurusan	int	11	Id jurusan
id_fakultas	Int	11	Id fakultas
nama_jurusan	varchar	255	Nama jurusan

#### 4.7.1.9 Tabel pendidikan\_terakhir

Nama tabel : pendidikan\_terakhir  
 Jumlah field : 2  
 Fungsi : Untuk menyimpan data distribusi

**Tabel 4.26** Penjelasan tabel pendidikan\_terakhir

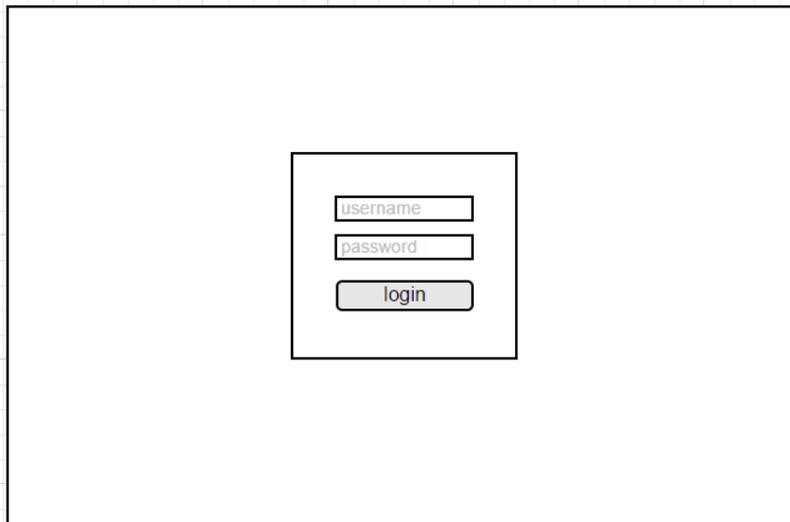
Nama field	Tipe	lebar	Keterangan
id_pendidikan_terakhir	Int	11	Id pendidikan terakhir
pendidikan_terakhir	Varchar	25	Pendidikan terakhir

### 4.8 Perancangan Antarmuka

Pada bagian ini akan dijelaskan tentang perancangan prototipe perangkat lunak yang akan dibangun.

#### 4.8.1 Login

login merupakan halaman awal dari sistem yang berisi *form* login. Berikut ini merupakan rancangan halaman login pada Gambar 4.23.

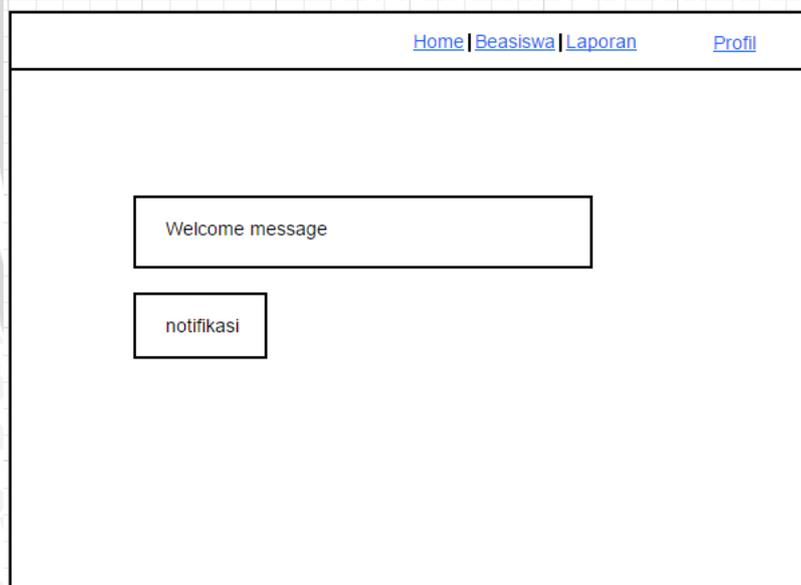


A wireframe of a login form. It consists of a central box containing three elements: a text input field labeled 'username', a text input field labeled 'password', and a button labeled 'login'.

**Gambar 4.23** Perancangan antarmuka login

#### 4.8.2 Home Mahasiswa

Home mahasiswa merupakan halaman utama dari sistem yang berisi menu. Berikut ini merupakan rancangan halaman home pada Gambar 4.24.



A wireframe of a student home page. At the top, there is a navigation bar with links: [Home](#), [Beasiswa](#), [Laporan](#), and [Profil](#). Below the navigation bar, there are two content boxes: a larger one labeled 'Welcome message' and a smaller one labeled 'notifikasi'.

**Gambar 4.24** Perancangan antarmuka home mahasiswa

#### 4.8.3 Menu Beasiswa

Halaman menu beasiswa merupakan halaman menu beasiswa dari sistem yang berisi list list beasiswa yang ada. Berikut ini merupakan rancangannya pada Gambar 4.25.

[Home](#) | [Basiswa](#) | [Laporan](#) | [Profil](#)

List beasiswa

▼ No	▼ Nama Beasiswa	▼ Periode Beasiswa	▼ Periode Pendaftaran	▼ kuota	▼ aksi
1	beasiswa astra	1 jan 2016 sd 30 juli 2016	1 des 2015 sd 30 des 2015	30	<a href="#">detail</a>
2	beasiswa rokok	11 feb 2016 sd 5 ags 2016	1 des 2015 sd 30 des 2015	15	<a href="#">detail</a>

**Gambar 4.25** Perancangan antarmuka menu beasiswa

#### 4.8.4 halaman *Detail* Beasiswa Mahasiswa

Halaman *detail* beasiswa merupakan halaman *detail* beasiswa dari sistem yang berisi data beasiswa yang dipilih. Berikut ini merupakan rancangannya pada Gambar 4.26.

[Home](#) | [Basiswa](#) | [Laporan](#) | [Profil](#)

Detail beasiswa

Nama Beasiswa	beasiswa astra
Periode Beasiswa	1 jan 2016 sd 30 juli 2016
Periode Pendaftaran	1 des 2015 sd 30 des 2016
Kuota	30
Dana Beasiswa	1.000.000 per bulan
Keterangan	-

**Gambar 4.26** Perancangan antarmuka *detail* beasiswa mahasiswa

#### 4.8.5 Halaman Mendaftar Beasiswa

Halaman mendaftar beasiswa merupakan halaman mengajukan beasiswa dari sistem yang berisi *form* pendaftaran beasiswa yang dipilih. Berikut ini merupakan rancangannya pada Gambar 4.27

Home | [Beasiswa](#) | [Laporan](#) | [Profil](#)

Daftar beasiswa

[Back](#)

Nim	123
Nama mahasiswa	andre
Nama beasiswa	beasiswa astra
ipk	<input type="text"/>
semester	<input type="text"/>
file	<input type="button" value="upload"/>

**Gambar 4.27** Perancangan antarmuka mendaftar beasiswa

#### 4.8.6 Halaman Profil

Halaman profil merupakan halaman *detail* beasiswa dari sistem yang berisi data profil. Berikut ini merupakan rancangannya pada Gambar 4.28.

Home | [Beasiswa](#) | [Laporan](#) | [Profil](#)

Profil

Username	123	Nama Ayah	Anton
Nama lengkap	Andre	Pekerjaan Ayah	BUMN
Tempat, tanggal lahir	Surabaya, 30 april 94	Penghasilan Ayah	7.000.000
Jenis kelamin	Laki laki	Nama Ibu	Ratna
Telp	08120313	Pekerjaan Ibu	Ibu rumah tangga
HP	123123213	Penghasilan Ibu	-
Email	andre@mail.com	Jumlah saudara	2
Agama	Budha		
Afiliasi	FILKOM, SISTEM INFORMASI		
Pendidikan terakhir	SMA		

**Gambar 4.28** Perancangan antarmuka halaman profil

#### 4.8.7 Halaman Ubah *Password*

Halaman ubah *password* merupakan halaman ubah *password* dari sistem yang berisi *form* penggantian *password*. Berikut ini merupakan rancangannya pada Gambar 4.29.

**Gambar 4.29** Perancangan antarmuka halaman ubah *password*

#### 4.8.8 Halaman Seleksi Partisipan

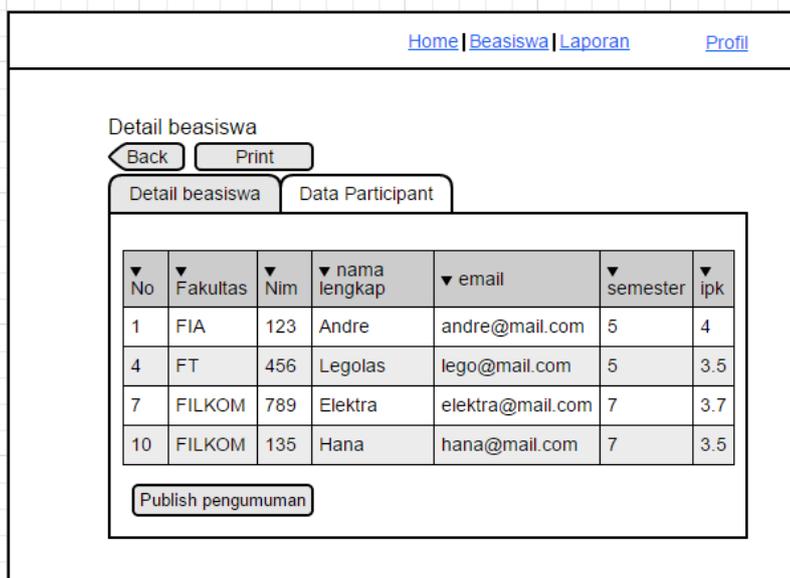
Halaman seleksi partisipan merupakan halaman yang berisi mahasiswa mahasiswi yang mendaftar untuk diseleksi. Berikut ini merupakan rancangannya pada Gambar 4.30.

▼ No	▼ Nim	▼ nama lengkap	▼ tanggal pengajuan	▼ aksi
1	123	Andre	11 maret	Diterima ▼
4	456	Legolas	11 maret	Diterima ▼
7	789	Elektra	11 maret	Diterima ▼
10	135	Hana	11 maret	Ditolak ▼

**Gambar 4.30** Perancangan antarmuka halaman seleksi partisipan

#### 4.8.9 Halaman Membuat Laporan

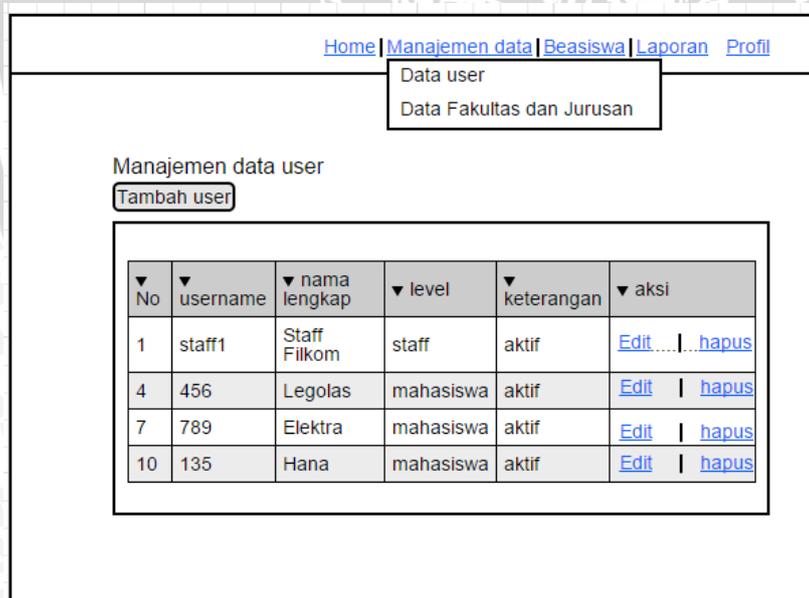
Halaman membuat laporan merupakan yang berisi list penerima beasiswa. Berikut ini merupakan rancangannya pada Gambar 4.31



Gambar 4.31 Perancangan antarmuka membuat laporan

#### 4.8.10 Halaman Manajemen Data User

Halaman manajemen data *user* merupakan yang berisi list *user* yang ada. Halaman ini mirip dengan manajemen data fakultas , jurusan dan beasiswa. Berikut ini merupakan rancangannya pada Gambar 4.32



Gambar 4.32 Perancangan antarmuka manajemen data *user*

#### 4.8.11 Halaman Menambah Beasiswa

Halaman menambah beasiswa merupakan yang berisi *form* penambahan data. Halaman ini mirip dengan menambah data *user*. Berikut ini merupakan rancangannya pada Gambar 4.33

[Home](#) | [Manajemen data](#) | [Beasiswa](#) | [Laporan](#) | [Profil](#)

Tambah data beasiswa

nama beasiswa	periode beasiswa
<input type="text"/>	4/22/2012 <input type="button" value="Calendar"/>
alokasi dana	periode pendaftaran
<input type="text"/>	4/22/2012 <input type="button" value="Calendar"/>
Kuota	
<input type="text"/>	
Pembagian kuota	
No Nama fakultas kuota	
1 FILKOM	<input type="text"/>
2 FIA	<input type="text"/>
Keterangan	
<input type="text"/>	
<input type="button" value="Cancel"/> <input type="button" value="Simpan"/>	

**Gambar 4.33** Perancangan antarmuka menambah beasiswa



## BAB 5 IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan dilakukan proses implementasi ,pengujian dan analisis hasil pengujian. Tujuan tahapan ini adalah untuk menemukan kesalahan yang bisa saja terjadi dalam sistem. Tujuannya untuk memastikan bahwa sistem yang dibangun telah sesuai dengan apa yang direncanakan

### 5.1 Spesifikasi Sistem

Sistem informasi beasiswa merupakan sistem informasi yang berbasis *website*, sehingga memerlukan koneksi internet untuk mengakses. sistem informasi beasiswa akan diakses untuk mahasiswa yang ingin mendaftar beasiswa. sistem informasi beasiswa juga akan diakses oleh staff tingkat fakultas dan universitas. Berikut ini merupakan spesifikasi perangkat lunak yang dibutuhkan untuk mengakses sistem informasi beasiswa adalah sebagai berikut :

1. Microsoft Windows XP, Vista, 7, 8, 8.1, 10, Linux.
2. Web Browser yang mendukung *javascript*, seperti Mozilla Firefox, Google Chrome, Safari, Microsoft Edge.

Sedangkan syarat minimum perangkat keras yang dibutuhkan dalam menjalankan sistem informasi beasiswa adalah sebagai berikut :

1. *Processor* Intel Dual-Core 2.0GHz atau AMD Athlon X2 2.0GHz atau di atasnya.
2. RAM 1Gb atau lebih besar.
3. *VGA Onboard*.
4. *Wifi Onboard*.

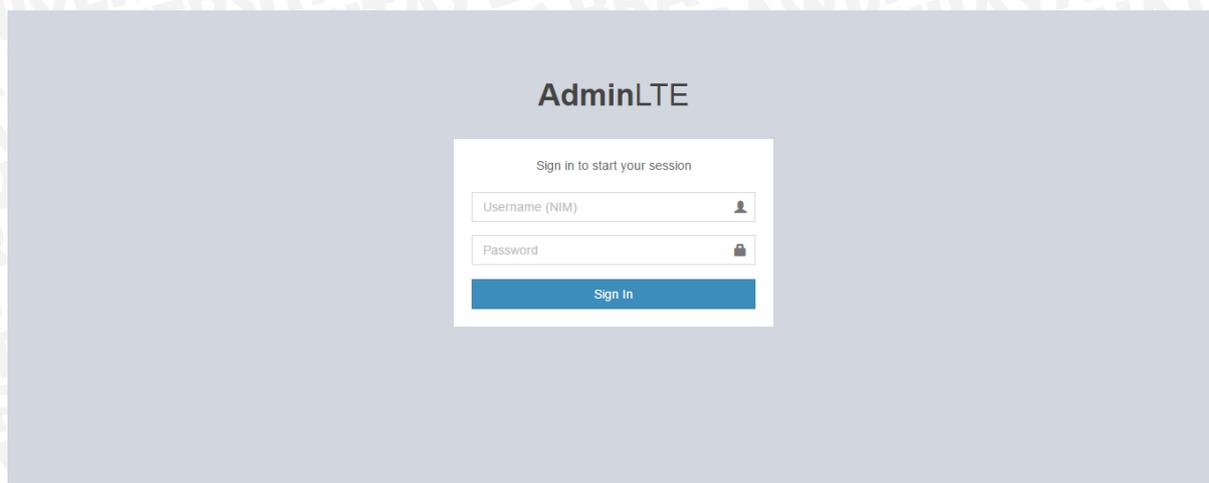
### 5.2 Implementasi

Berikut ini merupakan hasil implementasi yang sudah dibuat.

#### 5.2.1 Halaman Login

Halaman login pada gambar 5.1 merupakan halaman yang digunakan untuk masuk kedalam sistem.

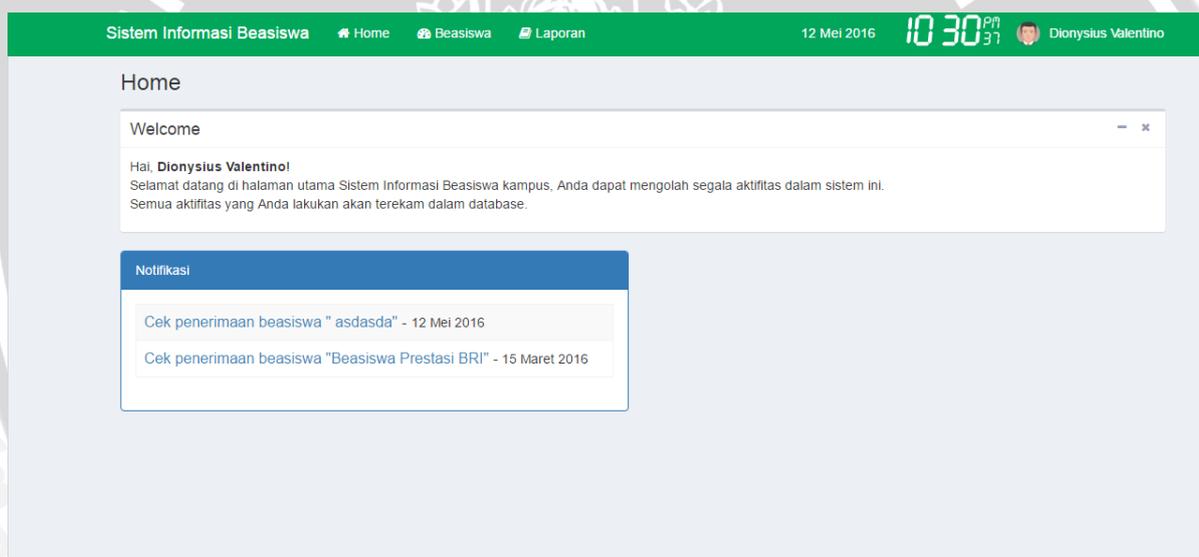




**Gambar 5.1** Antarmuka login

## 5.2.2 Halaman Home Mahasiswa

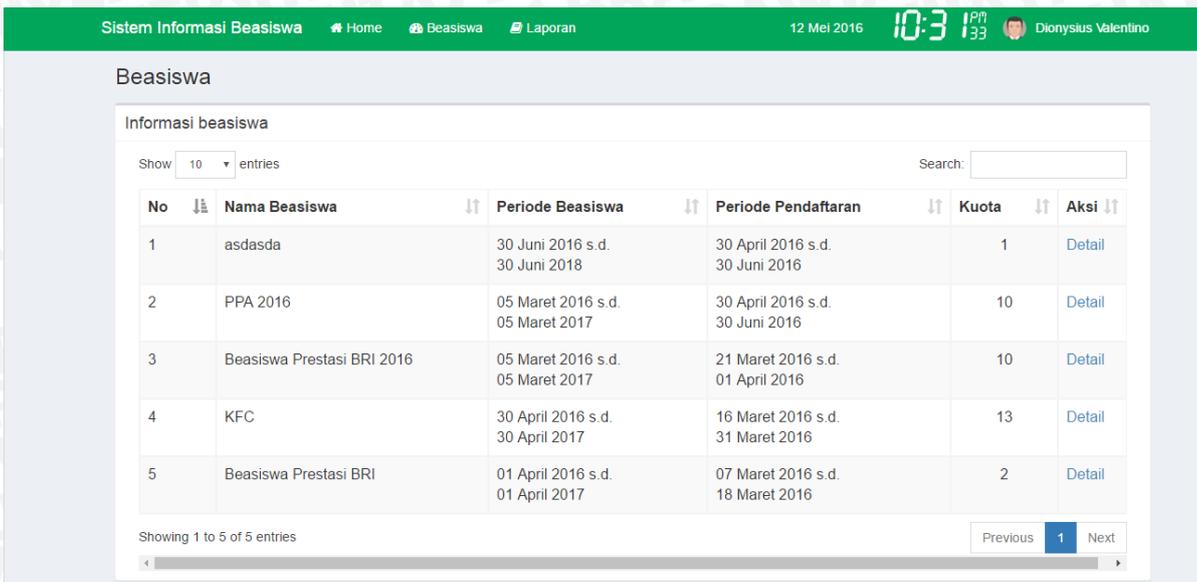
Halaman home pada gambar 5.2 merupakan halaman awal ketika berhasil masuk kedalam sistem.



**Gambar 5.2** Antarmuka halaman home mahasiswa

## 5.2.3 Halaman Beasiswa Mahasiswa

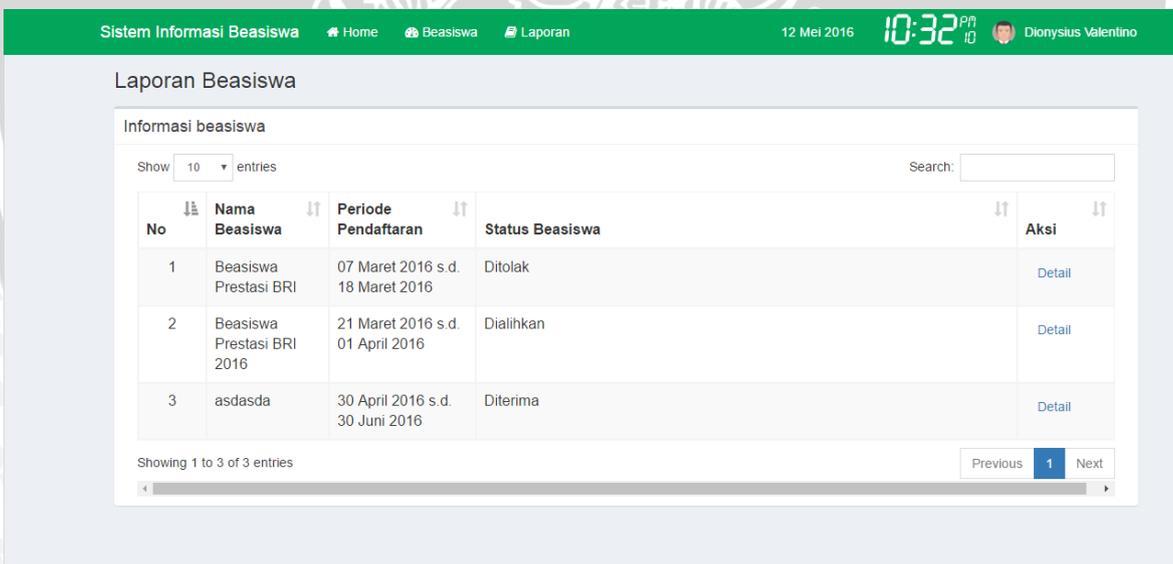
Halaman beasiswa mahasiswa pada gambar 5.3 merupakan halaman yang berisi list beasiswa yang ada.



**Gambar 5.3** Antarmuka halaman beasiswa mahasiswa

### 5.2.4 Halaman Laporan Mahasiswa

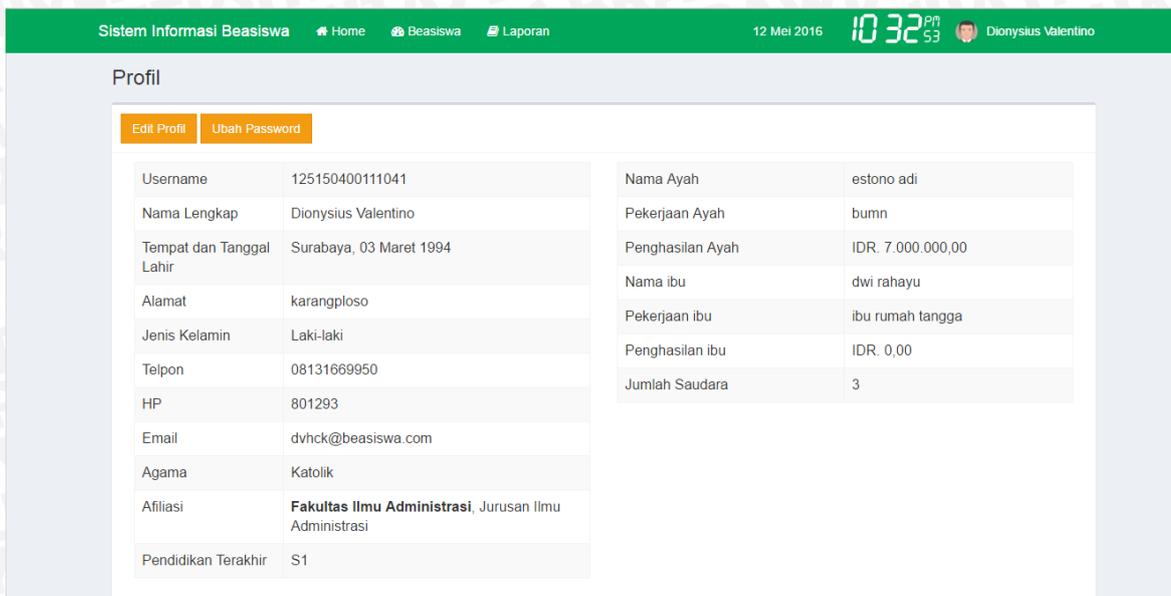
Halaman laporan mahasiswa pada gambar 5.4 merupakan halaman yang berisi status dari beasiswa yang telah didaftar.



**Gambar 5.4** Antarmuka halaman laporan mahasiswa

### 5.2.5 Halaman Profil Mahasiswa

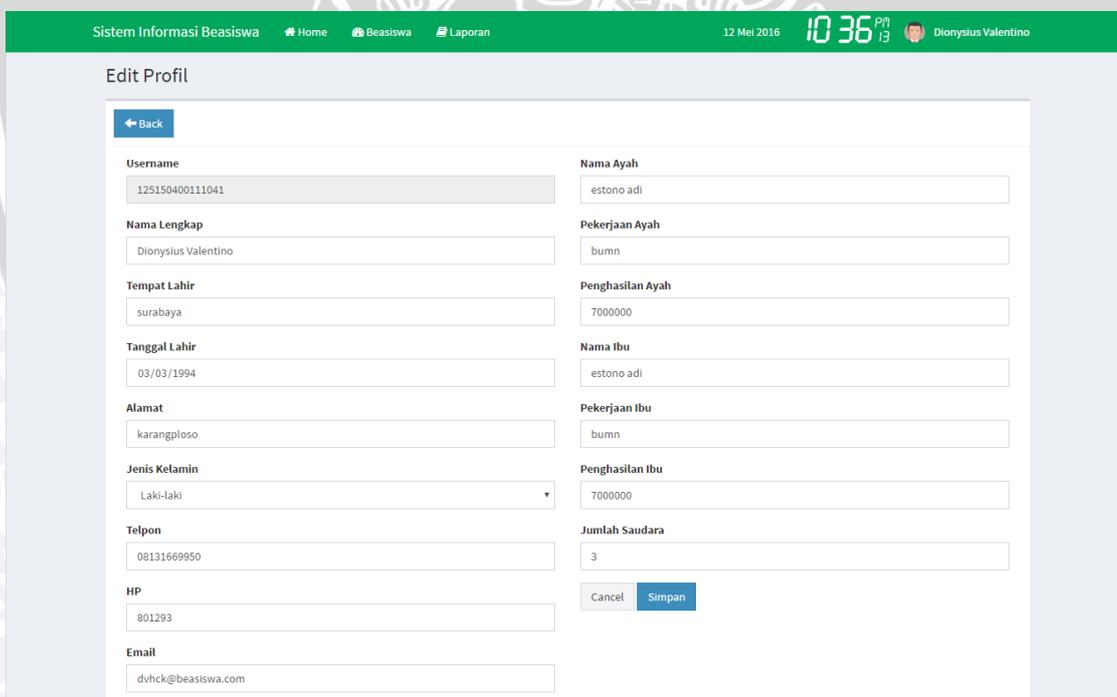
Halaman profil mahasiswa pada gambar 5.5 merupakan halaman yang berisi profil dari user mahasiswa.



**Gambar 5.5** Antarmuka halaman profil mahasiswa

### 5.2.6 Halaman Ubah Profil Mahasiswa

Halaman ubah profil pada gambar 5.6 merupakan halaman untuk mengubah data profil yang berisi *form* ubah profil.



**Gambar 5.6** Antarmuka halaman ubah profil mahasiswa

### 5.2.7 Halaman Ubah Password

Halaman ubah *password* pada gambar 5.7 merupakan halaman untuk mengubah *password user* yang berisi *form* ubah *password*.

**Gambar 5.7** Antarmuka halaman ubah *password*

### 5.2.8 Halaman *Detail* Beasiswa Mahasiswa

Halaman *detail* beasiswa pada gambar 5.8 mahasiswa merupakan halaman yang berisi *detail* beasiswa yang dipilih.

Detail Beasiswa	
<a href="#">Back</a>	
<a href="#">Participants</a>	
Nama Beasiswa	Beasiswa Prestasi BRI
Periode Beasiswa	01 April 2016 s.d. 01 April 2017 (12 Bulan)
Periode Pendaftaran	07 Maret 2016 s.d. 18 Maret 2016 (11 Hari)
Dana Beasiswa	IDR. 1.000.000,00 * 12 Bulan
Kuota Beasiswa	2
Keterangan	<ul style="list-style-type: none"> <li>Semester &gt; 3</li> <li>IPK &gt; 3.00</li> </ul>

**Gambar 5.8** Antarmuka halaman *detail* beasiswa mahasiswa

### 5.2.9 Halaman Mengajukan Beasiswa

Halaman mengajukan beasiswa pada gambar 5.9 merupakan halaman untuk mengajukan beasiswa yang berisi *form* daftar pengajuan beasiswa.

Sistem Informasi Beasiswa Home Beasiswa Laporan 12 Mei 2016 10:39 AM Dionysius Valentino

### Daftar Pengajuan Beasiswa

[← Back](#)

**NIM \***  
125150400111041

**Nama Mahasiswa \***  
Dionysius Valentino

**Nama Beasiswa \***  
PPA 2016

**IPK \***

**Semester \***

**File Pendukung \***  
 Tidak ada file yang dipilih  
 Format file di convert dalam 1 zip/winrar

**\*) Isikan secara lengkap**

Copyright © 2016 Sistem Informasi beasiswa. All rights reserved.

Version 2.0

**Gambar 5.9** Antarmuka halaman mengajukan beasiswa

### 5.2.10 Halaman Home Fakultas

Halaman home pada gambar 5.10 merupakan halaman awal ketika berhasil masuk kedalam sistem.

Sistem Informasi Beasiswa Home Beasiswa Laporan 12 Mei 2016 10:41 AM Staff FIA

### Home

Welcome

Hai, **Staff FIA!**  
 Selamat datang di halaman utama Sistem Informasi Beasiswa kampus, Anda dapat mengolah segala aktifitas dalam sistem ini.  
 Semua aktifitas yang Anda lakukan akan terekam dalam database.

**Gambar 5.10** Antarmuka halaman home fakultas

### 5.2.11 Halaman Beasiswa Fakultas

Halaman beasiswa fakultas pada gambar 5.11 merupakan halaman yang berisi list beasiswa yang ada.

Sistem Informasi Beasiswa Home Beasiswa Laporan 12 Mei 2016 10:41 PM Staff FIA

Beasiswa

Informasi Beasiswa

Show 10 entries Search:

No	Nama Beasiswa	Periode Beasiswa	Periode Pendaftaran	Aksi
1	asdasda	30 Juni 2016 s.d. 30 Juni 2018	30 April 2016 s.d. 30 Juni 2016	<a href="#">Detail</a>
2	PPA 2016	05 Maret 2016 s.d. 05 Maret 2017	30 April 2016 s.d. 30 Juni 2016	<a href="#">Detail</a>
3	Beasiswa Prestasi BRI 2016	05 Maret 2016 s.d. 05 Maret 2017	21 Maret 2016 s.d. 01 April 2016	<a href="#">Detail</a>
4	KFC	30 April 2016 s.d. 30 April 2017	16 Maret 2016 s.d. 31 Maret 2016	<a href="#">Detail</a>
5	Beasiswa Prestasi BRI	01 April 2016 s.d. 01 April 2017	07 Maret 2016 s.d. 18 Maret 2016	<a href="#">Detail</a>

Showing 1 to 5 of 5 entries Previous 1 Next

**Gambar 5.11** Antarmuka halaman beasiswa fakultas

### 5.2.12 Halaman Laporan Fakultas dan Universitas

Halaman laporan fakultas dan universitas pada Gambar 5.12 merupakan halaman yang berisi list beasiswa untuk dibuat laporannya.

Sistem Informasi Beasiswa Home Beasiswa Laporan 12 Mei 2016 10:43 PM Staff FIA

Laporan

Laporan Beasiswa

Show 10 entries Search:

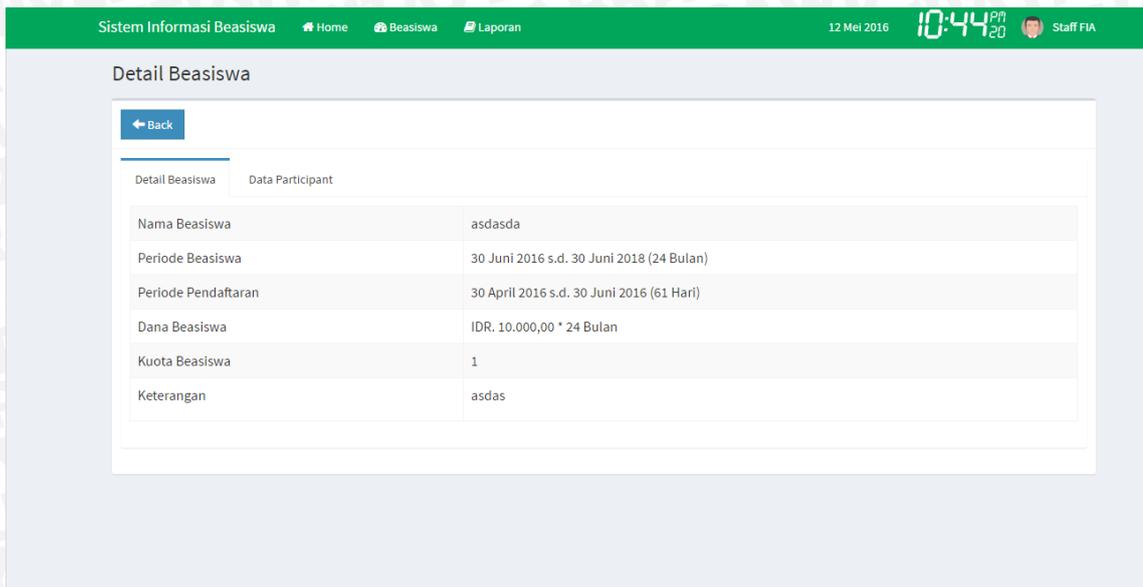
No	Nama Beasiswa	Periode Beasiswa	Periode Pendaftaran	Kuota	Aksi
1	asdasda	30 Juni 2016 s.d. 30 Juni 2018	30 April 2016 s.d. 30 Juni 2016	1	<a href="#">Detail</a>
2	PPA 2016	05 Maret 2016 s.d. 05 Maret 2017	30 April 2016 s.d. 30 Juni 2016	10	<a href="#">Detail</a>
3	Beasiswa Prestasi BRI 2016	05 Maret 2016 s.d. 05 Maret 2017	21 Maret 2016 s.d. 01 April 2016	10	<a href="#">Detail</a>
4	KFC	30 April 2016 s.d. 30 April 2017	16 Maret 2016 s.d. 31 Maret 2016	13	<a href="#">Detail</a>
5	Beasiswa Prestasi BRI	01 April 2016 s.d. 01 April 2017	07 Maret 2016 s.d. 18 Maret 2016	2	<a href="#">Detail</a>

Showing 1 to 5 of 5 entries Previous 1 Next

**Gambar 5.12** Antarmuka halaman laporan fakultas dan universitas

### 5.2.13 Halaman *Detail* Beasiswa Fakultas

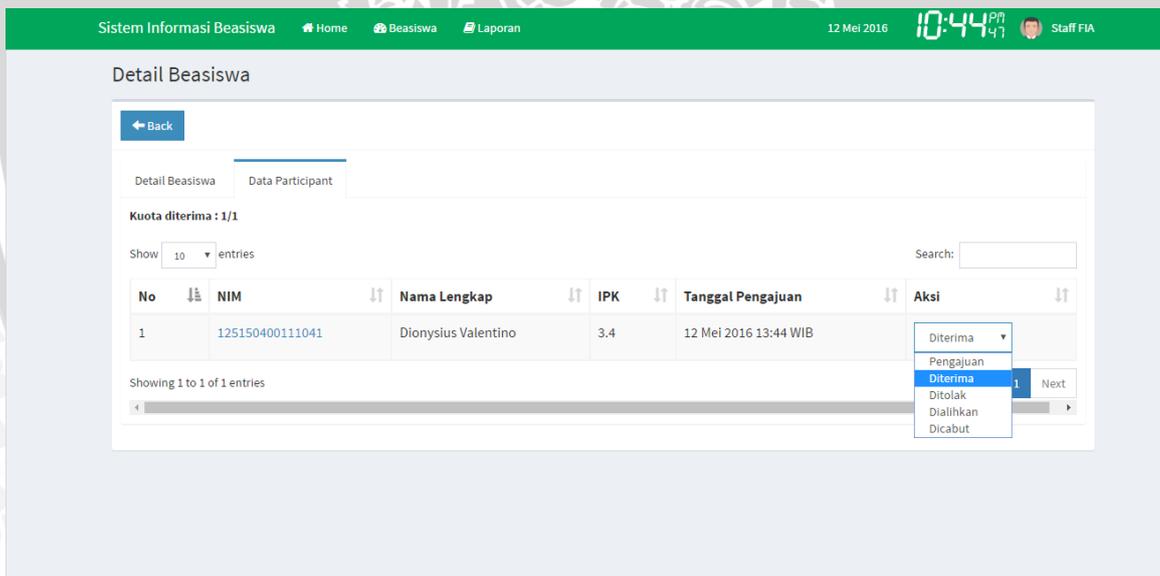
Halaman *detail* beasiswa fakultas pada gambar 5.13 merupakan halaman berisi *detail* beasiswa yang dipilih.



**Gambar 5.13** Antarmuka halaman *detail* beasiswa fakultas

### 5.2.14 Halaman Seleksi Partisipan

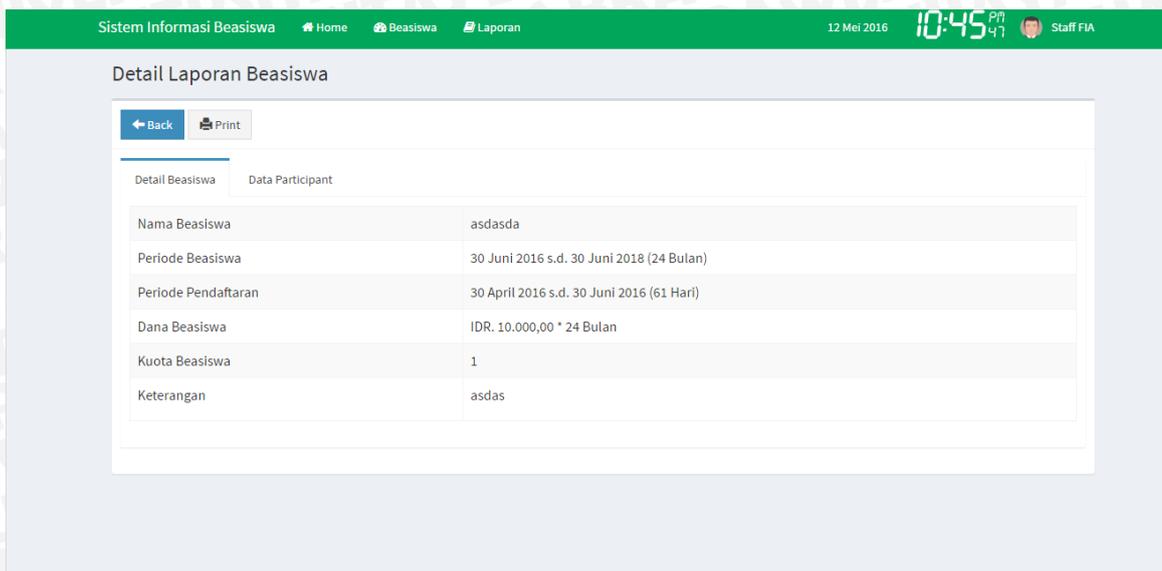
Halaman Seleksi partisipan pada Gambar 5.14 merupakan halaman dari *detail* beasiswa yang berisi data partisipan beasiswa.



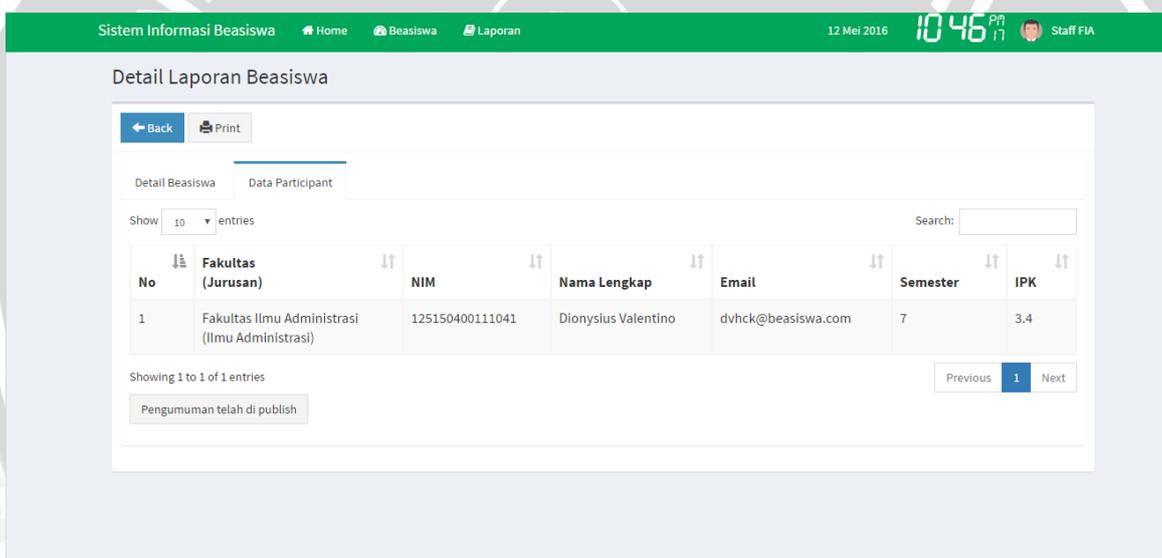
**Gambar 5.14** Antarmuka halaman seleksi partisipan

### 5.2.15 Halaman *Detail* Beasiswa Laporan Fakultas

Halaman *detail* beasiswa laporan fakultas pada gambar 5.15 dan 5.16 merupakan halaman berisi *detail* beasiswa yang dipilih dan tombol cetak untuk mencetak laporan.



**Gambar 5.15** Antarmuka halaman *detail* beasiswa laporan fakultas bagian 1



**Gambar 5.16** Antarmuka halaman *detail* beasiswa laporan fakultas bagian 2

### 5.2.16 Halaman Profil Fakultas

Halaman profil fakultas pada gambar 5.17 merupakan halaman yang berisi profil dari *user*.

Sistem Informasi Beasiswa Home Beasiswa Laporan 12 Mei 2016 10:47 PM Staff FIA

Profil

Edit Profil Ubah Password

Username	staff2
Nama Lengkap	Staff FIA
Tempat dan Tanggal Lahir	
Alamat	
Jenis Kelamin	Laki-laki
Telpon	
HP	
Email	staff2@beasiswa.com
Agama	
Afiliasi	Fakultas Ilmu Administrasi, Jurusan Ilmu Administrasi
Pendidikan Terakhir	Tidak Pernah Sekolah

**Gambar 5.17** Antarmuka halaman profil fakultas

### 5.2.17 Halaman Home Universitas

Halaman home pada gambar 5.18 merupakan halaman awal ketika berhasil masuk kedalam sistem.

Sistem Informasi Beasiswa Home Manajemen Data Beasiswa Laporan 12 Mei 2016 10:48 PM Administrator

Home

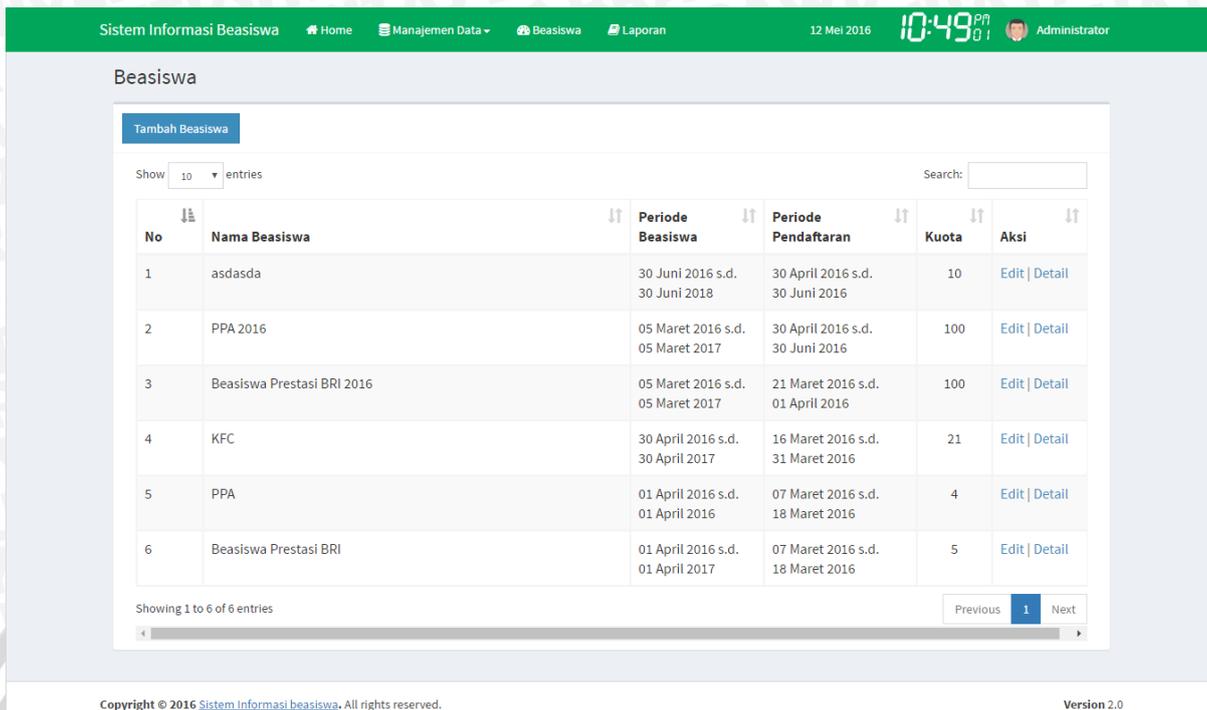
Welcome

Hai, **Administrator!**  
Selamat datang di halaman utama Sistem Informasi Beasiswa kampus, Anda dapat mengolah segala aktifitas dalam sistem ini. Semua aktifitas yang Anda lakukan akan terekam dalam database.

**Gambar 5.18** Antarmuka halaman home universitas

### 5.2.18 Halaman Beasiswa Universitas

Halaman beasiswa universitas pada gambar 5.19 merupakan halaman berisi list beasiswa yang ada.



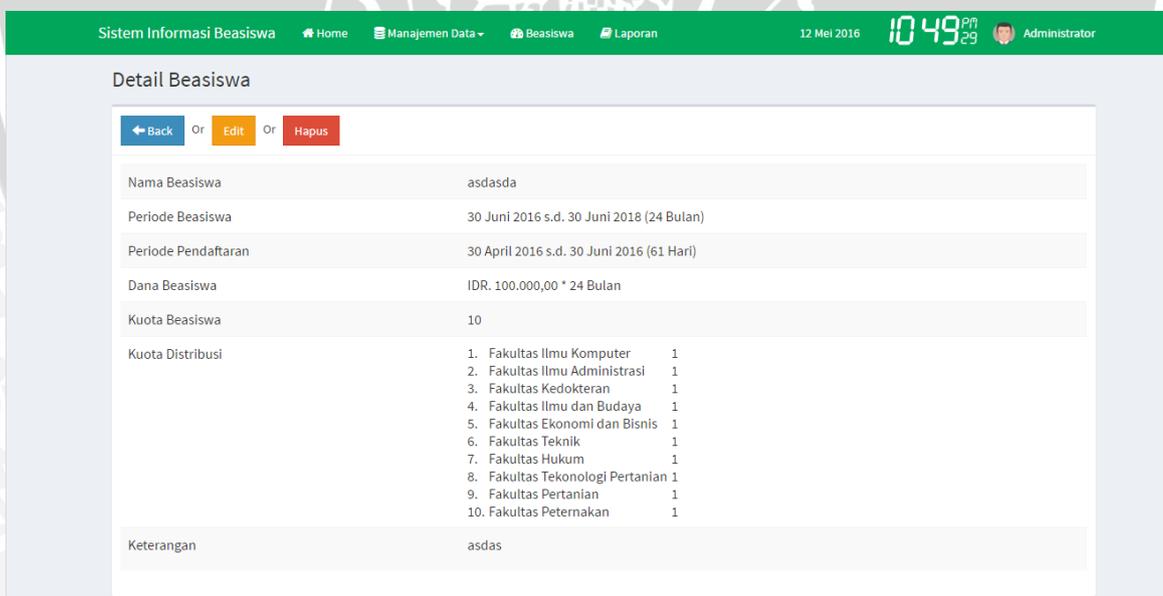
Copyright © 2016 Sistem Informasi beasiswa. All rights reserved.

Version 2.0

**Gambar 5.19** Antarmuka halaman beasiswa universitas

### 5.2.19 Halaman *Detail* Beasiswa Universitas

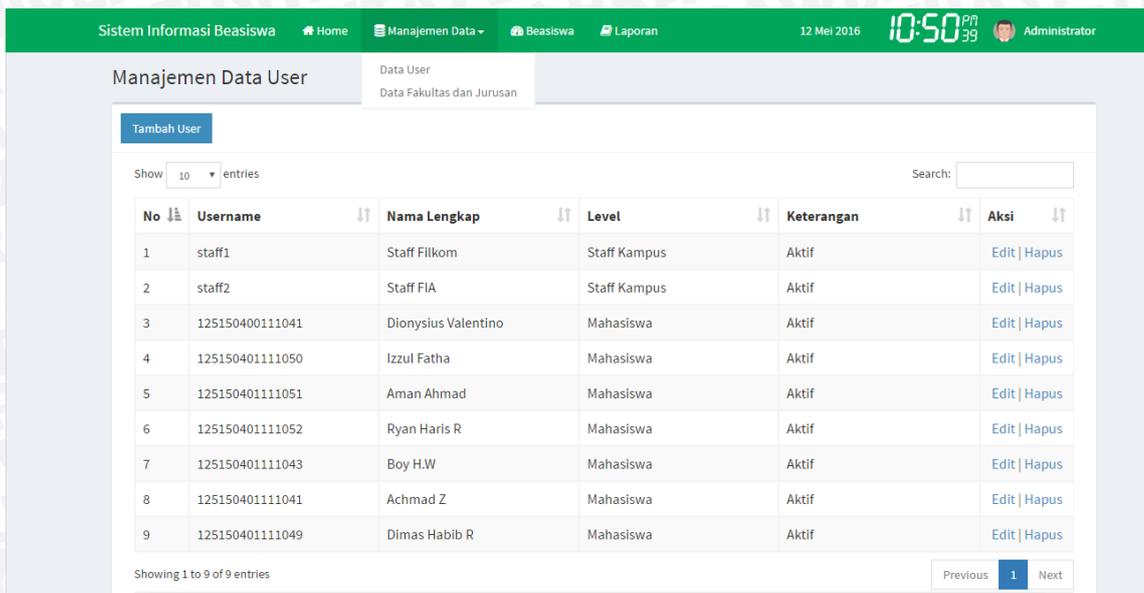
Halaman *detail* beasiswa universitas pada gambar 5.20 merupakan halaman berisi *detail* beasiswa yang dipilih.



**Gambar 5.20** Antarmuka halaman *detail* beasiswa universitas

### 5.2.20 Halaman Manajemen Data *User*

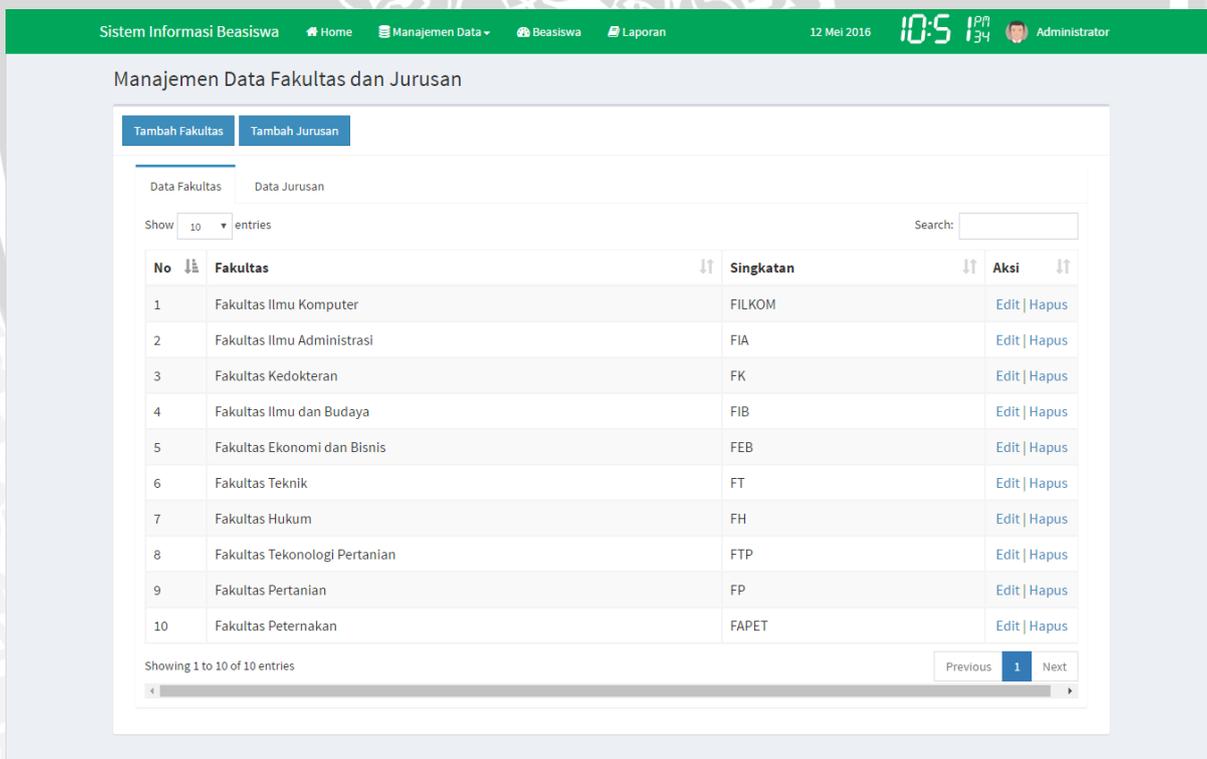
Halaman manajemen data *user* pada gambar 5.21 merupakan halaman berisi list *user* yang ada.



Gambar 5.21 Antarmuka halaman manajemen data user

### 5.2.21 Halaman Manajemen Data Fakultas dan Jurusan

Halaman manajemen data fakultas dan jurusan pada gambar 5.22 dan gambar 5.23 merupakan halaman berisi list data fakultas dan jurusan yang ada.



Gambar 5.22 Antarmuka halaman manajemen data fakultas dan jurusan bagian 1



Sistem Informasi Beasiswa Home Manajemen Data Beasiswa Laporan 12 Mei 2016 10:51:49 Administrator

### Manajemen Data Fakultas dan Jurusan

Tambah Fakultas Tambah Jurusan

Data Fakultas Data Jurusan

Show 10 entries Search:

No	Fakultas	Jurusan	Aksi
1	Fakultas Ilmu Komputer	Sistem Informasi	Edit   Hapus
2	Fakultas Ilmu Komputer	Teknik Informatika	Edit   Hapus
3	Fakultas Ilmu Administrasi	Ilmu Administrasi	Edit   Hapus
4	Fakultas Ilmu Administrasi	Sistem Informasi Manajemen	Edit   Hapus
5	Fakultas Kedokteran	Kedokteran 1	Edit   Hapus
6	Fakultas Ilmu dan Budaya	Sastra Bahasa Inggris	Edit   Hapus

Showing 1 to 6 of 6 entries Previous 1 Next

**Gambar 5.23** Antarmuka halaman manajemen data fakultas dan jurusan bagian 2

#### 5.2.22 Halaman Tambah Beasiswa

Halaman tambah beasiswa pada gambar 5.24 merupakan halaman berisi *form* untuk menambah data beasiswa.

Sistem Informasi Beasiswa Home Manajemen Data Beasiswa Laporan 12 Mei 2016 10:53 PM Administrator

### Tambah Data Beasiswa

[← Back](#)

Nama Beasiswa

Periode Beasiswa

Atokasi Dana (per-mahasiswa per-bulan)

Periode Pendaftaran

Kuota Mahasiswa

Pembagian Kuota

No	Nama Fakultas	Kuota	No	Nama Fakultas	Kuota
1	Fakultas Ilmu Komputer	<input type="text"/>	6	Fakultas Teknik	<input type="text"/>
2	Fakultas Ilmu Administrasi	<input type="text"/>	7	Fakultas Hukum	<input type="text"/>
3	Fakultas Kedokteran	<input type="text"/>	8	Fakultas Teknologi Pertanian	<input type="text"/>
4	Fakultas Ilmu dan Budaya	<input type="text"/>	9	Fakultas Pertanian	<input type="text"/>
5	Fakultas Ekonomi dan Bisnis	<input type="text"/>	10	Fakultas Peternakan	<input type="text"/>

Keterangan

[Cancel](#) [Simpan](#)

Gambar 5.24 Antarmuka halaman tambah beasiswa

### 5.2.23 Halaman Ubah Beasiswa

Halaman ubah beasiswa pada gambar 5.25 merupakan halaman berisi *form* untuk mengubah data beasiswa.

Sistem Informasi Beasiswa Home Manajemen Data Beasiswa Laporan 12 Mei 2016 10:54 Administrator

### Tambah Data Beasiswa

← Back

**Nama Beasiswa**

**Periode Beasiswa**

**Alokasi Dana (per-mahasiswa per-bulan)**

**Periode Pendaftaran**

**Kuota Mahasiswa**

**Pembagian Kuota**

No	Nama Fakultas	Kuota	No	Nama Fakultas	Kuota
1	Fakultas Ilmu Komputer	<input type="text" value="1"/>	7	Fakultas Hukum	<input type="text" value="1"/>
2	Fakultas Ilmu Administrasi	<input type="text" value="1"/>	8	Fakultas Teknologi Pertanian	<input type="text" value="1"/>
3	Fakultas Kedokteran	<input type="text" value="1"/>	9	Fakultas Pertanian	<input type="text" value="1"/>
4	Fakultas Ilmu dan Budaya	<input type="text" value="1"/>	10	Fakultas Peternakan	<input type="text" value="1"/>
5	Fakultas Ekonomi dan Bisnis	<input type="text" value="1"/>			
6	Fakultas Teknik	<input type="text" value="1"/>			

**Keterangan**

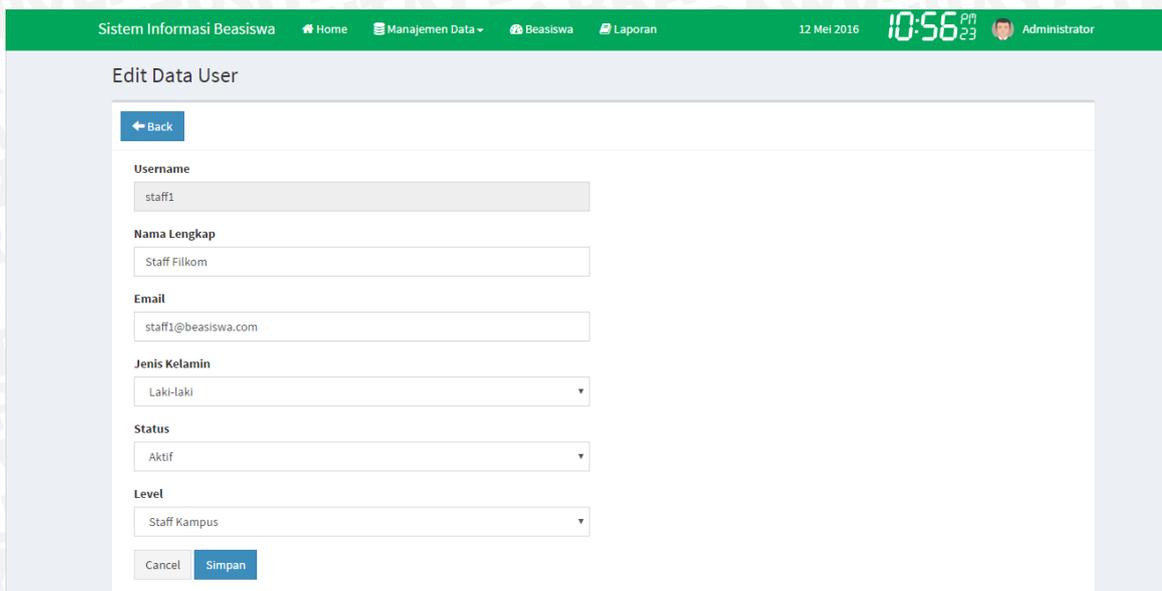
Cancel Simpan

**Gambar 5.25** Antarmuka halaman ubah beasiswa

### 5.2.24 Halaman Ubah User

Halaman ubah user pada gambar 5.26 merupakan halaman berisi form untuk mengubah data user.

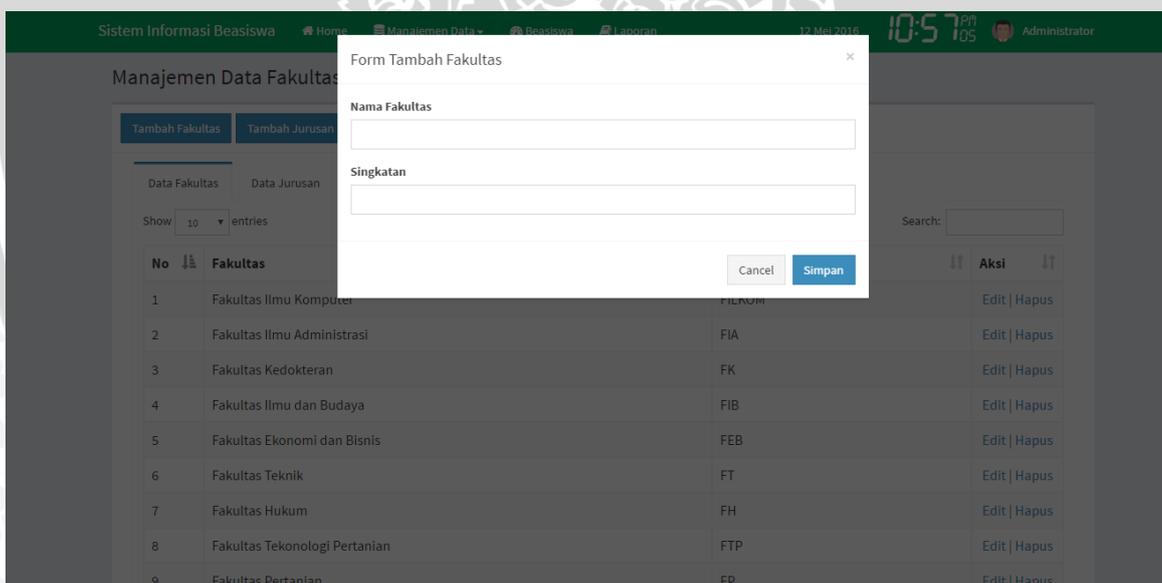




**Gambar 5.26** Antarmuka halaman ubah user

### 5.2.25 Halaman Tambah Fakultas

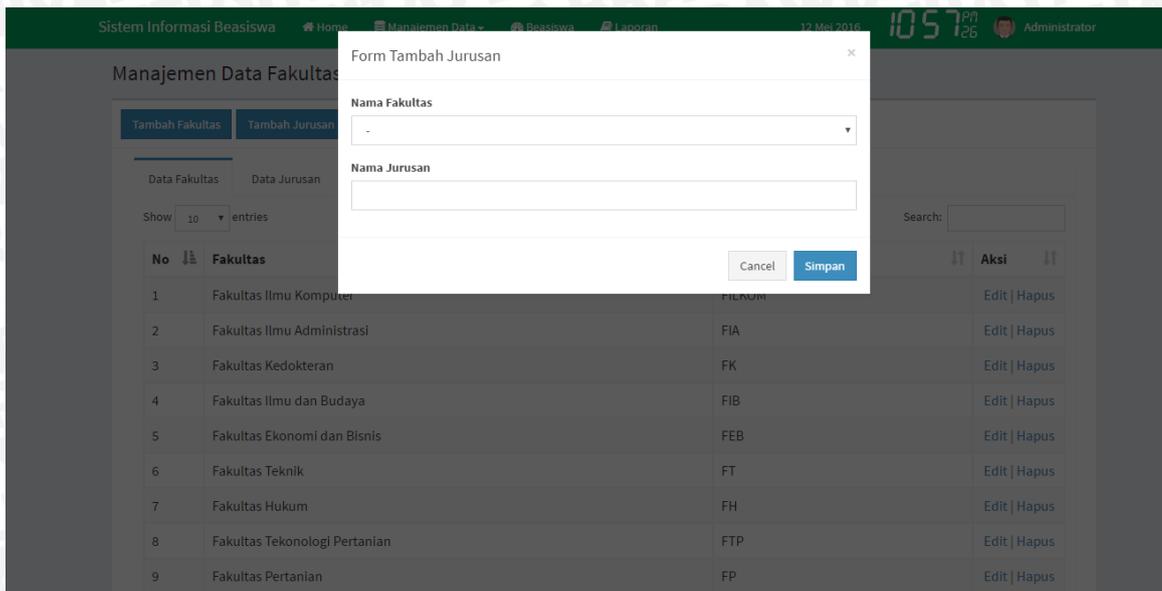
Halaman tambah fakultas pada gambar 5.27 merupakan halaman berisi form untuk menambah data fakultas.



**Gambar 5.27** Antarmuka halaman tambah fakultas

### 5.2.26 Halaman Tambah Jurusan

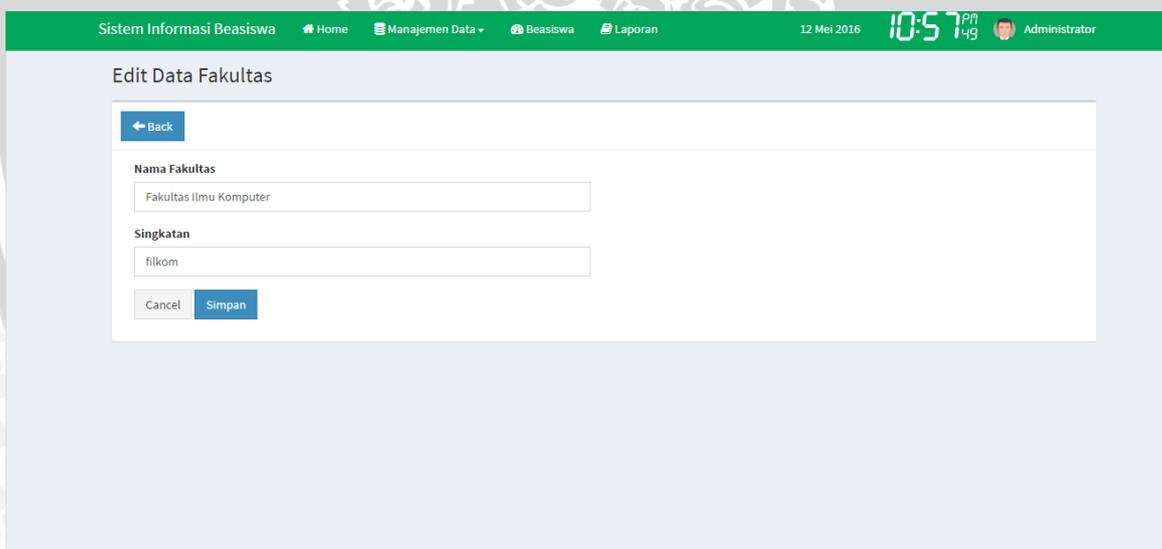
Halaman tambah jurusan pada gambar 5.28 merupakan halaman berisi form untuk menambah data jurusan.



**Gambar 5.28** Antarmuka halaman tambah jurusan

### 5.2.27 Halaman Ubah Fakultas

Halaman ubah fakultas pada gambar 5.29 merupakan halaman berisi *form* untuk mengubah data fakultas.



**Gambar 5.29** Antarmuka halaman ubah fakultas

### 5.2.28 Halaman Ubah Jurusan

Halaman ubah jurusan pada gambar 5.30 merupakan halaman berisi *form* untuk mengubah data jurusan.

**Gambar 5.30** Antarmuka halaman ubah jurusan

### 5.2.29 Halaman Profil Universitas

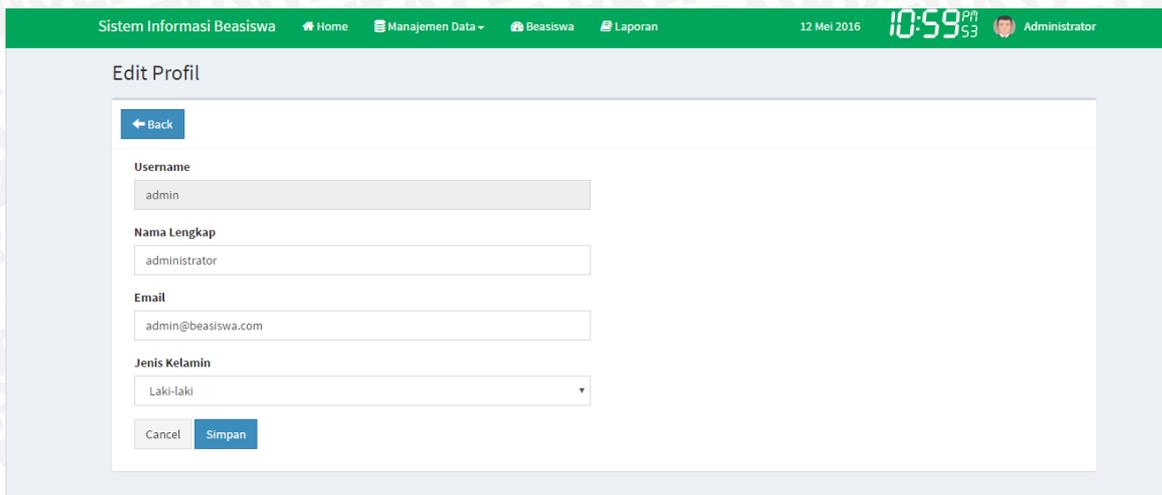
Halaman profil universitas pada gambar 5.31 merupakan halaman yang berisi profil dari *user*.

Username	admin
Nama Lengkap	administrator
Jenis Kelamin	Laki-laki
Email	admin@beasiswa.com

**Gambar 5.31** Antarmuka halaman profil universitas

### 5.2.30 Halaman Ubah Profil Universitas

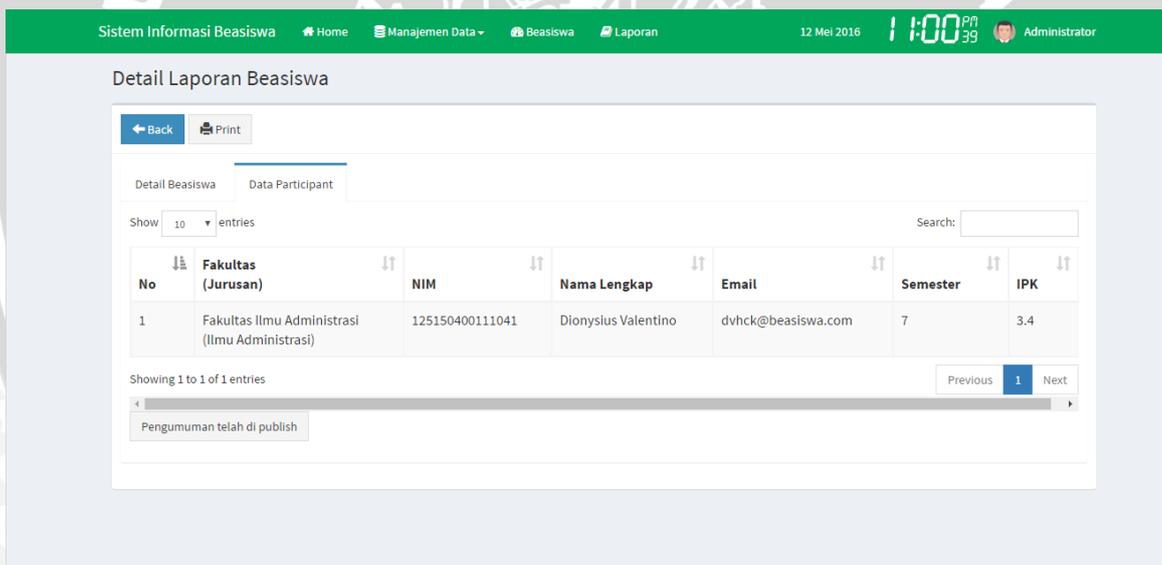
Halaman ubah profil pada gambar 5.32 merupakan halaman berisi *form* untuk mengubah data profil.



**Gambar 5.32** Antarmuka halaman ubah profil universitas

### 5.2.31 Halaman *Detail* Laporan Universitas

Halaman *detail* laporan pada gambar 5.33 merupakan halaman berisi list penerima beasiswa yang sudah diseleksi dan pilihan untuk mempublish pengumuman.



**Gambar 5.33** Antarmuka halaman *detail* laporan universitas

### 5.3 Rencana Pengujian

Pada tabel 5.1 ini berisi dengan apa yang akan diuji.

**Tabel 5.1** Rencana pengujian

Kelas Uji	SRS ID	Butir Uji	Jenis Pengujian
Persyaratan fungsional	F01-SRS-F-01	Pengujian Tambah beasiswa	<i>Black Box</i>

Tabel 5.1 Rencana pengujian (lanjutan)

Kelas Uji	SRS ID	Butir Uji	Jenis Pengujian
	F01-SRS-F-02	Pengujian Ubah beasiswa	<i>Black Box</i>
	F01-SRS-F-03	Pengujian Hapus beasiswa	<i>Black Box</i>
	F02- SRS-F-01	Pengujian Melihat beasiswa	<i>Black Box</i>
	F02- SRS-F-02	Pengujian Mengajukan beasiswa	<i>Black Box</i>
	F02- SRS-F-03	Pengujian Verifikasi status beasiswa	<i>Black Box</i>
	F03-SRS-F-01	Pengujian Menyeleksi partisipan	<i>Black Box</i>
	F04-SRS-F-01	Pengujian Melihat pengumuman	<i>Black Box</i>
	F05-SRS-F-01	Pengujian Membuat laporan	<i>Black Box</i>
	F06-SRS-F-01	Pengujian Mengubah profil	<i>Black Box</i>
	F06-SRS-F-02	Pengujian Mengubah <i>password</i>	<i>Black Box</i>
	F07-SRS-F-01	Pengujian Tambah <i>user</i>	<i>Black Box</i>
	F07-SRS-F-02	Pengujian Ubah <i>user</i>	<i>Black Box</i>
	F07-SRS-F-03	Pengujian Hapus <i>user</i>	<i>Black Box</i>
	F08-SRS-F-01	Pengujian Tambah jurusan	<i>Black Box</i>
	F08-SRS-F-02	Pengujian Ubah jurusan	<i>Black Box</i>
	F08-SRS-F-03	Pengujian Hapus jurusan	<i>Black Box</i>
	F08-SRS-F-04	Pengujian Tambah fakultas	<i>Black Box</i>
	F08-SRS-F-05	Pengujian Ubah fakultas	<i>Black Box</i>
	F08-SRS-F-06	Pengujian Hapus fakultas	<i>Black Box</i>
	F09-SRS-F-01	Pengujian Login	<i>Black Box</i>
	F09-SRS-F-02	Pengujian Logout	<i>Black Box</i>
Persyaratan non fungsional	SRS-NF-01	Perangkat lunak dapat dibuka pada berbagai macam <i>web browser</i> .	

## 5.4 Pengujian Black-Box

Pengujian *Black-box* atau pengujian kotak hitam adalah pengujian yang berfokus kepada persyaratan fungsional perangkat lunak. Hasil pengujian dapat dilihat pada tabel 5.2.

**Tabel 5.2** Hasil pengujian black-box

No	Test Name	SRS ID	Test Case	Expected Result	Result	Status
1	Kelola data beasiswa	F01-SRS-F-01	Pengujian Tambah beasiswa dengan data lengkap	Sistem akan menampilkan notifikasi <i>"Transaksi berhasil data beasiswa telah ditambah dan distribusikan"</i>	Sistem menampilkan notifikasi <i>"Transaksi berhasil data beasiswa telah ditambah dan distribusikan"</i>	valid
			Pengujian dengan salah satu kolom kosong	Sistem akan menampilkan notifikasi <i>"&lt;nama kolom&gt; tidak boleh kosong"</i>	Sistem menampilkan notifikasi <i>"&lt;nama kolom&gt; tidak boleh kosong"</i>	valid
			Pengujian dengan mengisi kolom alokasi dana atau kuota dengan huruf atau karakter	Sistem akan menampilkan notifikasi <i>"hanya boleh angka"</i>	Sistem menampilkan notifikasi <i>"hanya boleh angka"</i>	valid
			Pengujian dengan mengisi jumlah kolom pembagian kuota	Sistem akan menampilkan notifikasi <i>"Total kuota fakultas harus sama"</i>	Sistem menampilkan notifikasi <i>"Total kuota fakultas harus sama"</i>	valid

Tabel 5.2 Hasil pengujian black-box(lanjutan)

No	Test Name	SRS ID	Test Case	Expected Result	Result	Status
			tidak sama dengan kuota beasiswa	dengan Kuota Beasiswa"	dengan Kuota Beasiswa"	
		F01-SRS-F-02	Pengujian Ubah beasiswa	Sistem akan menampilkan notifikasi "Transaksi berhasil data beasiswa telah diupdate"	Sistem menampilkan notifikasi "Transaksi berhasil data beasiswa telah diupdate"	Valid
			Pengujian dengan salah satu kolom kosong	Sistem akan menampilkan notifikasi "<nama kolom> tidak boleh kosong"	Sistem menampilkan notifikasi "<nama kolom> tidak boleh kosong"	valid
			Pengujian dengan mengisi kolom alokasi dana atau kuota dengan huruf atau karakter	Sistem akan menampilkan notifikasi "hanya boleh angka"	Sistem menampilkan notifikasi "hanya boleh angka"	valid
			Pengujian dengan mengisi jumlah kolom pembagian kuota tidak sama dengan kuota	Sistem akan menampilkan notifikasi "Total kuota fakultas harus sama dengan Kuota Beasiswa"	Sistem menampilkan notifikasi "Total kuota fakultas harus sama dengan Kuota Beasiswa"	valid

Tabel 5.2 Hasil pengujian black-box(lanjutan)

No	Test Name	SRS ID	Test Case	Expected Result	Result	Status
			beasiswa			
		F01-SRS-F-03	Pengujian Hapus beasiswa	Sistem akan menampilkan notifikasi <i>"Transaksi berhasil Data beasiswa telah dihapus"</i>	Sistem menampilkan notifikasi <i>"Transaksi berhasil Data beasiswa telah dihapus"</i>	valid
2	Pengujian beasiswa	F02-SRS-F-01	Pengujian Melihat beasiswa	Sistem akan menampilkan data beasiswa beasiswa dalam bentuk tabel	Sistem menampilkan data beasiswa beasiswa dalam bentuk tabel	valid
		F02-SRS-F-02	Pengujian Mengajukan beasiswa dengan data lengkap	Sistem akan menampilkan notifikasi <i>"Transaksi berhasil pengajuan dikirim"</i>	Sistem menampilkan notifikasi <i>"Transaksi berhasil pengajuan dikirim"</i>	Valid
			Pengujian dengan salah satu kolom kosong	Sistem akan menampilkan notifikasi <i>"&lt;nama kolom&gt; tidak boleh kosong"</i>	Sistem menampilkan notifikasi <i>"&lt;nama kolom&gt; tidak boleh kosong"</i>	valid
			Pengujian dengan mengisi kolom ipk huruf atau karakter	Sistem akan menampilkan notifikasi <i>"IPK 0 - 4"</i>	Sistem menampilkan notifikasi <i>"IPK 0 - 4"</i>	valid

Tabel 5.2 Hasil pengujian black-box(lanjutan)

No	Test Name	SRS ID	Test Case	Expected Result	Result	Status
			Pengujian dengan tidak mengunggah file	Sistem akan menampilkan notifikasi "File tidak boleh kosong"	Sistem menampilkan notifikasi "File tidak boleh kosong"	valid
		F02-SRS-F-03	Pengujian Verifikasi status beasiswa	Sistem akan menampilkan notifikasi "Transaksi gagal anda telah mendapatkan beasiswa lain"	Sistem menampilkan notifikasi "Transaksi gagal anda telah mendapatkan beasiswa lain"	Valid
3	Seleksi mahasiswa	F03-SRS-F-01	Menyeleksi partisipan	Sistem akan menampilkan hasil mahasiswa yang diterima pada user universitas	Sistem menampilkan hasil mahasiswa yang diterima pada user universitas	Valid
4	Melihat pengumuman	F04-SRS-F-01	Pengujian Melihat pengumuman	Sistem akan menampilkan laporan hasil beasiswa dalam bentuk tabel	Sistem menampilkan laporan hasil beasiswa dalam bentuk tabel	Valid
5	Membuat laporan	F05-SRS-F-01	Pengujian Membuat laporan	Sistem akan menampilkan halaman cetak	Sistem akan menampilkan halaman cetak	Valid
6	Mengubah profil	F06-SRS-F-01	Pengujian Mengubah profil	Sistem akan menampilkan notifikasi "Transaksi berhasil profil sudah"	Sistem menampilkan notifikasi "Transaksi berhasil profil sudah"	Valid

Tabel 5.2 Hasil pengujian black-box(lanjutan)

No	Test Name	SRS ID	Test Case	Expected Result	Result	Status
				<i>diupdate"</i>	<i>diupdate"</i>	
			Pengujian dengan salah satu kolom kosong	Sistem akan menampilkan notifikasi " <i>&lt;nama kolom&gt; tidak boleh kosong"</i>	Sistem menampilkan notifikasi " <i>&lt;nama kolom&gt; tidak boleh kosong"</i>	valid
		F06-SRS-F-02	Pengujian dengan <i>password</i> lama tidak sesuai.	Sistem akan menampilkan notifikasi " <i>Anda salah memasukan password lama anda "</i>	Sistem menampilkan notifikasi " <i>Anda salah memasukan password lama anda "</i>	Valid
			Pengujian dengan <i>password</i> lama benar.	Sistem akan menampilkan notifikasi " <i>Password berhasil diupdate"</i>	Sistem akan menampilkan notifikasi " <i>Password berhasil diupdate"</i>	valid
			Pengujian dengan salah satu kolom kosong	Sistem akan menampilkan notifikasi " <i>&lt;nama kolom&gt; tidak boleh kosong"</i>	Sistem menampilkan notifikasi " <i>&lt;nama kolom&gt; tidak boleh kosong"</i>	valid
7	Kelola data user	F07-SRS-F-01	Pengujian Tambah user	Sistem akan menampilkan notifikasi " <i>Transaksi berhasil data user sudah ditambahkan "</i>	Sistem menampilkan notifikasi " <i>Transaksi berhasil data user sudah ditambahkan "</i>	valid
			Pengujian dengan	Sistem akan menampilkan	Sistem menampilkan	Valid

Tabel 5.2 Hasil pengujian black-box(lanjutan)

No	Test Name	SRS ID	Test Case	Expected Result	Result	Status
			salah satu kolom kosong	notifikasi " <i>&lt;nama kolom&gt; tidak boleh kosong</i> "	notifikasi " <i>&lt;nama kolom&gt; tidak boleh kosong</i> "	
			Pengujian dengan mengisi kolom email tidak valid	Sistem akan menampilkan notifikasi " <i>Email tidak valid</i> "	Sistem menampilkan notifikasi " <i>Email tidak valid</i> "	Valid
		F07-SRS-F-02	Pengujian Ubah user	Sistem akan menampilkan notifikasi " <i>Transaksi berhasil data user sudah diupdate</i> "	Sistem menampilkan notifikasi " <i>Transaksi berhasil data user sudah diupdate</i> "	Valid
			Pengujian dengan salah satu kolom kosong	Sistem akan menampilkan notifikasi " <i>&lt;nama kolom&gt; tidak boleh kosong</i> "	Sistem menampilkan notifikasi " <i>&lt;nama kolom&gt; tidak boleh kosong</i> "	valid
			Pengujian dengan mengisi kolom email tidak valid	Sistem akan menampilkan notifikasi " <i>Email tidak valid</i> "	Sistem akan menampilkan notifikasi " <i>Email tidak valid</i> "	valid
		F07-SRS-F-03	Pengujian Hapus user	Sistem akan menampilkan notifikasi " <i>Transaksi berhasil data user telah dihapus</i> "	Sistem menampilkan notifikasi " <i>Transaksi berhasil data user telah dihapus</i> "	Valid

**Tabel 5.2** Hasil pengujian black-box(lanjutan)

No	Test Name	SRS ID	Test Case	Expected Result	Result	Status
8	Kelola data fakultas dan jurusan	F08-SRS-F-01	Pengujian Tambah jurusan	Sistem akan menampilkan notifikasi <i>"Transaksi berhasil Data jurusan sudah ditambahkan"</i>	Sistem menampilkan notifikasi <i>"Transaksi berhasil Data jurusan sudah ditambahkan"</i>	valid
			Pengujian dengan salah satu kolom kosong	Sistem akan menampilkan notifikasi <i>"&lt;nama kolom&gt; tidak boleh kosong"</i>	Sistem menampilkan notifikasi <i>"&lt;nama kolom&gt; tidak boleh kosong"</i>	valid
		F08-SRS-F-02	Pengujian Ubah jurusan	Sistem akan menampilkan notifikasi <i>"Transaksi berhasil Data jurusan sudah diupdate"</i>	Sistem menampilkan notifikasi <i>"Transaksi berhasil Data jurusan sudah diupdate"</i>	Valid
			Pengujian dengan salah satu kolom kosong	Sistem akan menampilkan notifikasi <i>"&lt;nama kolom&gt; tidak boleh kosong"</i>	Sistem menampilkan notifikasi <i>"&lt;nama kolom&gt; tidak boleh kosong"</i>	valid
		F08-SRS-F-03	Pengujian Hapus jurusan	Sistem akan menampilkan notifikasi <i>"Transaksi berhasil Data jurusan telah dihapus"</i>	Sistem menampilkan notifikasi <i>"Transaksi berhasil Data jurusan telah dihapus"</i>	Valid

**Tabel 5.2** Hasil pengujian black-box(lanjutan)

No	Test Name	SRS ID	Test Case	Expected Result	Result	Status
		F08-SRS-F-04	Pengujian Tambah fakultas	Sistem akan menampilkan notifikasi <i>"Transaksi berhasil Data fakultas sudah ditambahkan"</i>	Sistem menampilkan notifikasi <i>"Transaksi berhasil Data fakultas sudah ditambahkan"</i>	Valid
			Pengujian dengan salah satu kolom kosong	Sistem akan menampilkan notifikasi <i>"&lt;nama kolom&gt; tidak boleh kosong"</i>	Sistem menampilkan notifikasi <i>"&lt;nama kolom&gt; tidak boleh kosong"</i>	valid
		F08-SRS-F-05	Pengujian Ubah fakultas	Sistem akan menampilkan notifikasi <i>"Transaksi berhasil Data fakultas sudah diupdate"</i>	Sistem menampilkan notifikasi <i>"Transaksi berhasil Data fakultas sudah diupdate"</i>	Valid
			Pengujian dengan salah satu kolom kosong	Sistem akan menampilkan notifikasi <i>"&lt;nama kolom&gt; tidak boleh kosong"</i>	Sistem menampilkan notifikasi <i>"&lt;nama kolom&gt; tidak boleh kosong"</i>	valid
		F08-SRS-F-06	Pengujian Hapus fakultas	Sistem akan menampilkan notifikasi <i>"Transaksi berhasil Data fakultas telah"</i>	Sistem menampilkan notifikasi <i>"Transaksi berhasil Data fakultas telah dihapus"</i>	Valid

**Tabel 5.2** Hasil pengujian black-box(lanjutan)

No	Test Name	SRS ID	Test Case	Expected Result	Result	Status
				<i>dihapus</i>		
9	Pengujian Login	F09-SRS-F-01	Pengujian dengan <i>username</i> atau <i>password</i> kosong.	Sistem akan menampilkan notifikasi " <i>Harap isi bidang ini</i> "	Sistem menampilkan notifikasi " <i>Harap isi bidang ini</i> "	valid
			Pengujian dengan <i>username</i> atau <i>password</i> salah.	Sistem akan menampilkan notifikasi " <i>Username tidak ditemukan atau password salah</i> "	Sistem menampilkan notifikasi " <i>Username tidak ditemukan atau password salah</i> "	Valid
			Pengujian dengan <i>username</i> atau <i>password</i> benar.	Sistem akan menampilkan home	Sistem menampilkan gome	valid
10	Pengujian Logout	F08-SRS-F-02	Pengujian dengan melakukan klik pada tombol <i>logout</i> .	Sistem akan mengarahkan ke halaman <i>login</i> .	Sistem mengarahkan ke halaman <i>login</i> .	Valid

### 5.5 Pengujian Compatibility

Tahap ini adalah proses analisis pengujian validasi memastikan bahwa perangkat lunak yang dibangun mampu berjalan dengan baik pada lingkungan yang berbeda (lingkungan pengguna). Pengujian dilakukan pada beberapa browser berbeda yaitu *Mozilla Firefox*, *Google Chrome*, *Internet Explorer*, *Microsoft Edge*, *Safari*, browser *Android*, dan browser *iOS*.

*Tools* yang akan digunakan pada penelitian ini adalah *Sortsiteversi Trial 5.21.754.0*. *SortSite* dipilih karena mampu menguji perangkat lunak berbasis *web*



secara keseluruhan pada setiap halaman yang dimiliki. *Sortsite* menguji *compability* berdasarkan 3 hal utama yaitu :

1. *Critical Issues* mendasarkan pada masalah fungsional atau hilangnya konten
2. *Major Issues* mendasarkan pada *layout* utama seperti bentuk table , posisi menu , ukuran gambar dan lainnya.
3. *Minor Issues* mendasarkan pada *layout* kecil seperti perbedaan jarak tulisan , pewarnaan tombol dan lainnya

Berikut hasil pengujiannya pada gambar 5.34

Browser	Internet Explorer					Edge	Firefox		Safari		Opera	Chrome	iOS			Android *	
	7.0	8.0	9.0	10.0	11.0	13	46	≤ 8.0	9.0	36	49	≤ 6.0	8.0	9.0	≤ 3.0	4.0	
Critical Issues	●	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Major Issues	●	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Minor Issues	●	●	●	●	●	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

\* Most Android devices from 4.1 onwards use Chrome as the default browser, older versions use the original Android browser

**Gambar 5.34** Hasil pengujian compabilitas menggunakan *sortsite*

## 5.6 Analisis Pengujian

Berdasarkan pengujian yang telah dilakukan didapatkan hasil analisis sebagai berikut :

1. Pengujian fungsionalitas dilakukan pada 45 *test case*. Pengujian ini menghasilkan nilai 100% valid sesuai dengan keluaran yang diharapkan pada sistem.
2. Hasil pengujian *compatibility* menggunakan *SortSite* menunjukkan perangkat lunak dapat berjalan dengan baik pada semua target *browser*, kecuali Internet Explorer versi 7.0 , 8.0 , 9.0 ,10.0 . Hasil pengujian *compatibility* dirangkum dalam Tabel 5.3.

**Tabel 5.3** Kalkulasi hasil pengujian kompabilitas

Browser version	Hasil	Masalah	Keterangan
Internet Explorer 7.0	Tidak dapat berjalan	-CSS <i>atribute</i>	Versi IE7 dan sebelumnya belum mendukung <i>attribute</i> pada CSS

**Tabel 5.3** Kalkulasi hasil pengujian kompatibilitas

Browser version	Hasil	Masalah	Keterangan
Internet Explorer 8.0	Terdapat <i>minor issues</i>	-CSS <i>property border-radius</i> -The <i>box-shadow CSS</i> -The <i>opacity CSS3</i>	Beberapa CSS <i>property</i> belum didukung.
Internet Explorer 9.0	Terdapat <i>minor issues</i>	-REQUIRED <i>attribute</i>	Versi IE9 belum mendukung
Internet Explorer 10.0	Terdapat <i>minor issues</i>	-DXFilters	Versi IE10 belum mendukung
Internet Explorer 11.0	Terdapat <i>minor issues</i>	-DXFilters	Versi IE11 belum mendukung
Edge 13	Dapat berjalan	-	-
Firefox 46	Dapat berjalan	-	-
Safari ≤ 8.0	Dapat berjalan	-	-
Safari 9.0	Dapat berjalan	-	-
Opera 36	Dapat berjalan	-	-
Chrome 49	Dapat berjalan	-	-
Browser iOS ≤ 6.0	Dapat berjalan	-	-
Browser iOS 8.0	Dapat berjalan	-	-
Browser iOS 9.0	Dapat berjalan	-	-
Browser Android ≤ 3.0	Dapat berjalan	-	-
Browser Android 4.0	Dapat berjalan	-	-

Dari hasil pengujian pada **Tabel 5.3**, dapat disimpulkan bahwa sistem ini telah *compatible* dengan 8 *browser* yang diuji sehingga telah memenuhi persyaratan non-fungsional yang telah ditetapkan.

## BAB 6 PENUTUP

Bagian ini memuat kesimpulan dan saran terhadap skripsi. Kesimpulan dan saran disajikan secara terpisah, dengan penjelasan sebagai berikut:

### 6.1 Kesimpulan

Kesimpulan yang diperoleh dari penelitian implementasi sistem informasi beasiswa adalah sebagai berikut:

1. Pada analisis persyaratan setelah dilakukan analisis didapatkan 9 persyaratan fungsional yaitu mengajukan beasiswa, melihat pengumuman, mengatur profil, menseleksi partisipan, membuat laporan, mengelola data *user*, mengelola data beasiswa, mengelola data jurusan dan fakultas serta login. Sedangkan untuk non fungsional terdapat persyaratan kompatibilitas browser perangkat lunak.
2. Perancangan dari sistem informasi beasiswa terdiri dari kelas diagram, model interaksi yang berisi sequence diagram, model data, rancangan antarmuka pengguna.
3. Berdasarkan hasil pengujian fungsional yang telah dilakukan, diperoleh hasil 100% valid pada 45 *test case* yang diuji, dan berdasarkan hasil pengujian komabilitas browser, sistem bisa dijalankan secara kompatibel pada delapan *browser*.

### 6.2 Saran

1. Penelitian selanjutnya dapat dikembangkan dengan menambahkan persyaratan non fungsional dan pengembangan lainnya.
2. Penelitian berikutnya dapat memperluas cakupan beasiswa tak hanya mencakup universitas tapi pihak sponsor juga.

## DAFTAR PUSTAKA

- Agus, Lahinta. (2009). *Konsep Rancangan Sistem Pendukung Keputusan Penentuan Kandidat Penerima Beasiswa (Studi Kasus pada TPSDM Propinsi Gorontalo)*. Tersedia di <http://wances.net46.net/files/jurnal/Agus%20Lahinta.pdf> [Diakses tanggal 4 november 2015].
- Alireza, M., Majid, Y.M., dan Rosnah. 2010. *Simple Additive Weighting Approach to Personnel Selection Problem*. Universitas Putra Malaysia.
- Ananta, N, Adi. 2014. *Sistem Pendukung Keputusan Rekomendasi Kelayakan Penerima Beasiswa Prestasi Menggunakan Metode Analytical Hierarchy Process (AHP)*. Universitas Dian Nuswantoro Semarang.
- Ardiyanto, B. M. Somantri, dan K. I. Satoto. 2011. *Perancangan Sistem Informasi Beasiswa Universitas Diponegoro Berbasis Web*. Universitas Diponegoro. Semarang.
- DSDM Consortium, 2014. *The DSDM Agile Project Framework*. Henwood: DSDM Consortium.
- Elizabeth, T. dan S. Darmawan. 2015. *Sistem Informasi Pemakaian Sparepart Mesin Packing pada PT. XYZ*. Jatisi 1(2), Maret 2015, ISSN: 2407-4322, Hal: 164-165.
- Fatta, H. A., 2007. *Analisis dan Perancangan Sistem Informasi untuk Keunggulan Bersaing Perusahaan dan Organisasi Modern*. Penerbit ANDI. Yogyakarta.
- Gaol, C. J. L. 2008. *Sistem Informasi Manajemen Pemahaman dan Aplikasi*. Grasindo.
- Guru99, 2016. [online] Tersedia di : <http://www.guru99.com/compatibilitytesting.html> [Diakses 10 Mei 2016].
- Ismail, M. 2004. *Konsep Sistem Informasi Manajemen*. Universitas Sumatera Utara. Medan.
- Jogiyanto. 2005. *Analisis dan Desain Sistem Informasi*. Yogyakarta: Penerbit Andi.
- McLeod, R. Jr, dan G. P.Schell. 2007. *Management Information System, 10<sup>th</sup> ed.* Penerjemah Yulianto, A. A. dan A. R. Fitriani. 2008. *Sistem Informasi Manajemen, Edisi 10*. Penerbit Salemba Empat. Jakarta.
- Neni dan N. Uluwiya. 2011. *Sistem Informasi Administrasi Manufaktur pada CV. Adinda Kencana Palembang*. STMIK. Palembang.
- O'brien, J.A., & Marakas, G.M., 2014. *Management Information System*. Diterjemahkan dari Bahasa Inggris oleh Liza Nurbani Puspitasari dan Hirson Kurnia. 2014. Jakarta : Salemba Empat.

Putra, A dan Hardiyanti, D, Yunika. 2011. *Penentuan Penerima Beasiswa Dengan Menggunakan Fuzzy Multiple Atribute Decission Making*. Jurnal Sistem Informasi (JSI), VOL. 3, NO. 1, April 2011.

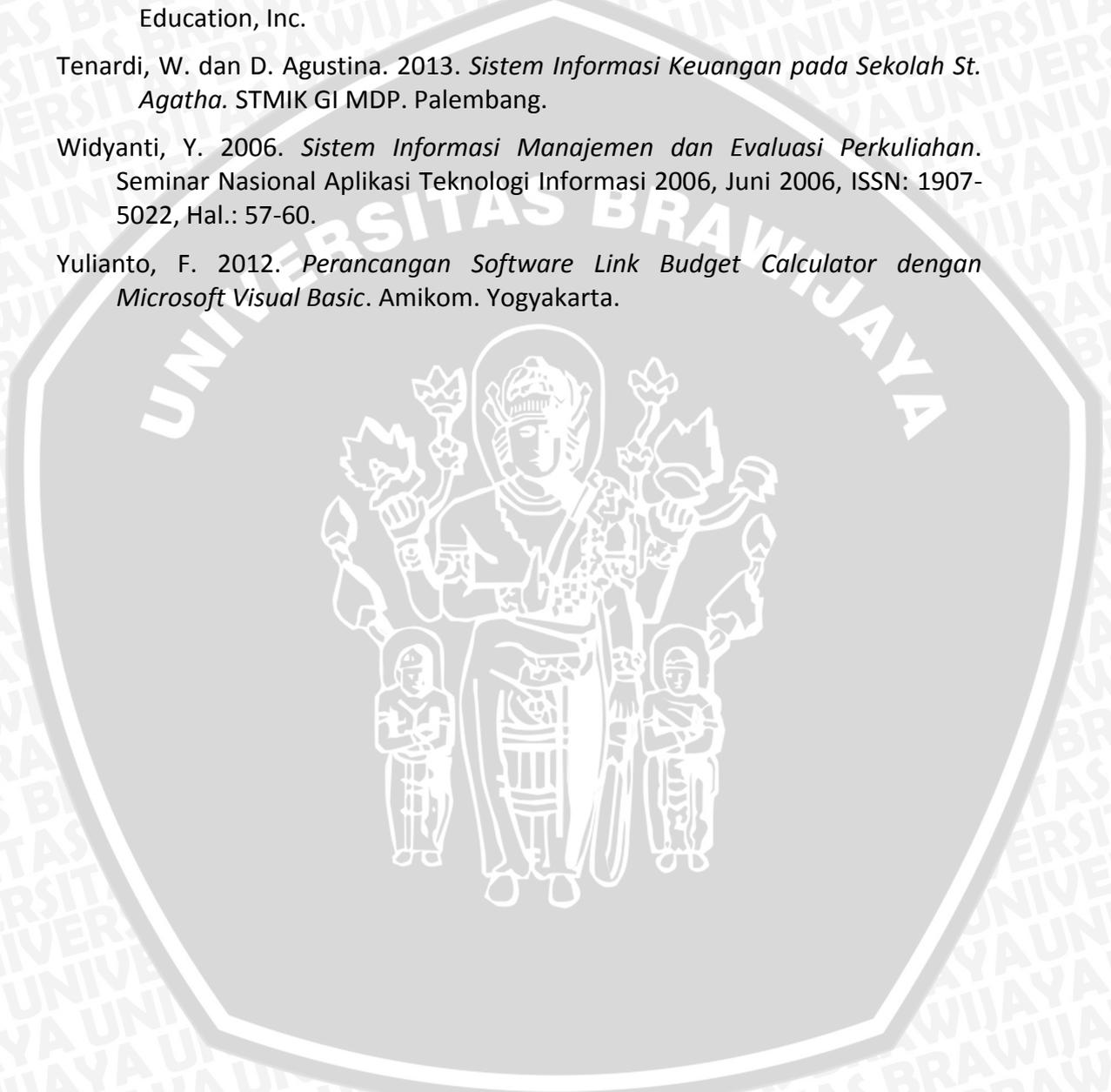
Satzinger, Jackson, Burd. 2010. *System Analisis and Design with the Unified Process*. USA: Course Technology, Cengage Learning.

Sommerville, Ian. 2010. *Software Engineering 9<sup>th</sup> Edition*. USA : Pearson Education, Inc.

Tenardi, W. dan D. Agustina. 2013. *Sistem Informasi Keuangan pada Sekolah St. Agatha*. STMIK GI MDP. Palembang.

Widyanti, Y. 2006. *Sistem Informasi Manajemen dan Evaluasi Perkuliahan*. Seminar Nasional Aplikasi Teknologi Informasi 2006, Juni 2006, ISSN: 1907-5022, Hal.: 57-60.

Yulianto, F. 2012. *Perancangan Software Link Budget Calculator dengan Microsoft Visual Basic*. Amikom. Yogyakarta.



## LAMPIRAN PROSES ITERASI

Proses pembangunan sistem informasi beasiswa dengan menggunakan metode iterasi pada analisis kebutuhan dan perancangan dilakukan lebih dari satu kali iterasi. Iterasi yang akan dilakukan pada pembuatan sistem ini sebanyak 2 kali iterasi, yaitu mulai dari iterasi ke-1 sampai iterasi ke-2

### A.1 Proses Iterasi ke-1

Berikut merupakan kondisi awal pada iterasi ke-1

**Tabel A.1** Key Stakeholder Needs iterasi ke-1

Need	Stakeholder	Priority	Current Situation	Proposed Solutions
Kemudahan mencari informasi beasiswa	mahasiswa	M	Pencarian informasi beasiswa sudah memudahkan	Pencarian beasiswa ditampilkan dalam tabel yang bisa diurutkan berdasarkan nama , periode
Proses pendaftaran lebih mudah	mahasiswa	M	Manual dengan datang ke lembaga kemahasiswaan	Terdapat fitur untuk mendaftar beasiswa dengan mengunggah berkas
Proses seleksi dipermudah	fakultas	M	Manual dengan menseleksi berkas atau wawancara	Terdapat fitur <i>sorting</i> pendaftar berdasarkan IPK tertinggi
Mahasiswa hanya bisa menerima 1 beasiswa	mahasiswa	M	-	Sistem mendeteksi status mahasiswa, jika sudah menerima beasiswa tidak dapat

Pengalokasian dana terkomputerisasi	universitas	M	Manual menggunakan form kertas	mendaftar lagi Pihak universitas dapat mengalokasikan dana ketika membuat beasiswa baru
Dapat membuat laporan penerima beasiswa	Universitas dan fakultas	M	Manual menggunakan form kertas	Sistem menyediakan fitur untuk mencetak laporan penerima beasiswa
Kemudahan dalam melakukan	Universitas dan fakultas	M	Ditulis secara manual	Sistem menyediakan fitur untuk mengelola data
pengelolaan data user dan fakultas serta jurusan		-		yang meliputi membuat ,menghapus dan mengubah data

Persyaratan awal fungsional dan non fungsional sebelum dilakukan evaluasi ada pada table A.2 dan A.3

**Tabel A.2** Tabel persyaratan fungsional iterasi ke-1

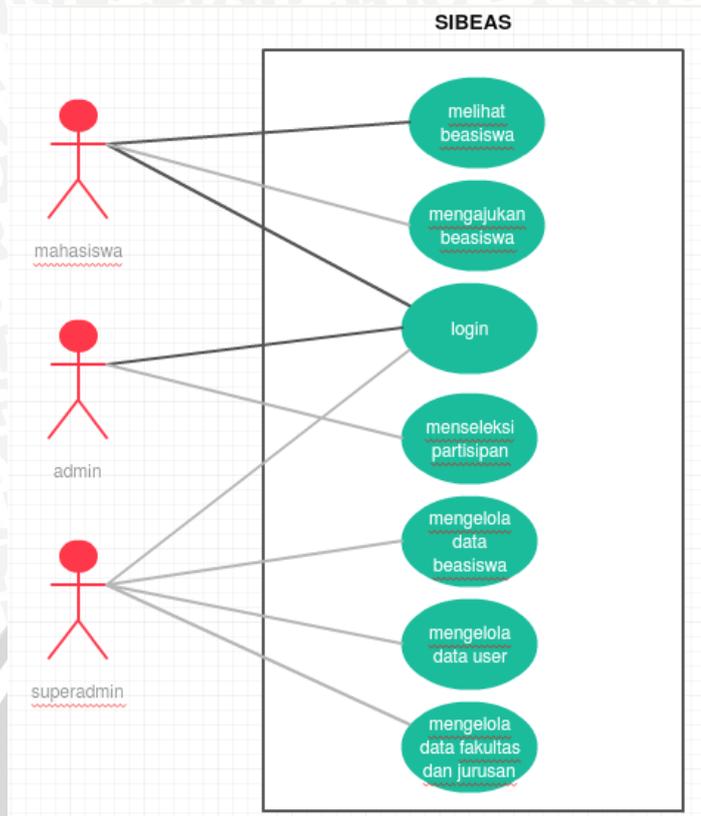
No	Kode Fungsi	Persyaratan Fungsional	Use Case
1	SRS-F-01	User mahasiswa dapat mendaftar beasiswa melalui sistem	Mengajukan Beasiswa
2	SRS-F-02	Sistem menampilkan list beasiswa yang tersedia	Melihat beasiswa
3	SRS-F-03	Sistem dapat membantu fakultas dalam menyeleksi	Menseleksi Partisipan

4	SRS-F-04	Sistem dapat menampilkan informasi data <i>user</i> dan menu untuk menambah, mengubah atau menghapus data <i>user</i>	Mengelola data <i>user</i>
5	SRS-F-05	Sistem dapat menampilkan informasi data beasiswa dan menu untuk menambah, mengubah atau menghapus data beasiswa	Mengelola data beasiswa
6	SRS-F-06	Sistem dapat menampilkan informasi data beasiswa dan menu untuk menambah, mengubah atau menghapus data jurusan dan fakultas	Mengelola data jurusan dan fakultas
7	SRS-F-07	Sistem menyediakan fungsi untuk dapat masuk ke dalam sistem	Login

**Tabel A.3** Persyaratan Non Fungsional iterasi ke-1

Kode	Deskripsi persyaratan
SRS-NF-01	Sistem dapat dibuka pada berbagai macam <i>web browser</i> .
SRS-NF-02	Sistem memiliki respon time kurang dari 10 detik
SRS-NF-03	Sistem dapat diakses 24 jam

Model *use case* perancangan awal akan digambarkan pada gambar B.1 *use case* ini dirancang berdasarkan persyaratan fungsional.



**Gambar A.1** Use case

Setelah melakukan evaluasi kemudian pada iterasi ke-1 terjadi perubahan sebagai berikut :

1. Perubahan aktor superadmin menjadi universitas.
2. Perubahan aktor admin menjadi fakultas.
3. Penambahan persyaratan fungsional dari 7 menjadi 9.
4. Penambahan method pada diagram kelas.

Hasil evaluasi pada tahap awal mengakibatkan perubahan aktor dan persyaratan fungsional dan non fungsional. Dari kebutuhan perangkat lunak akan diperjelas kedalam spesifikasi kebutuhan perangkat lunak. Berikut ini kondisi iterasi ke-1.

**Tabel A.4** Penambahan Key Stakeholder dan user needs iterasi ke-1

Need	Stakeholder	Priority	Current Situation	Proposed Solutions
Mahasiswa dapat melihat pengumuman beasiswa	mahasiswa	M	Dengan kertas pengumuman dan pengumuman di website	Terdapat fitur pengumuman hasil seleksi

**Tabel A.5** Persyaratan fungsional iterasi ke-1

No	Kode Fungsi	Persyaratan Fungsional	Use Case
1	SRS-F-03	User mahasiswa dapat mendaftar beasiswa melalui sistem	Mengajukan Beasiswa
2	SRS-F-05	User melihat informasi status penerimaan	Melihat pengumuman
3	SRS-F-07	Sistem terdapat fitur untuk menampilkan atau mengubah informasi <i>user</i> dan <i>password</i>	Mengatur profil
4	SRS-F-04	Sistem dapat membantu fakultas dalam menyeleksi mahasiswa yang mendaftar dengan tampilan table dan adanya <i>sorting</i>	Menseleksi Partisipan
5	SRS-F-06	Sistem dapat mencetak laporan penerima beasiswa	Membuat Laporan
6	SRS-F-07	Sistem dapat menampilkan informasi data <i>user</i> dan menu untuk menambah, mengubah atau	Mengelola data <i>user</i>

**Tabel A.5** Persyaratan fungsional iterasi ke-1(lanjutan)

No	Kode Fungsional	Persyaratan Fungsional	Use Case
		menghapus data <i>user</i>	
7	SRS-F-01	Sistem dapat menampilkan informasi data beasiswa dan menu untuk menambah , mengubah atau menghapus data beasiswa	Mengelola data beasiswa
8	SRS-F-02	Sistem dapat menampilkan informasi data beasiswa dan menu untuk menambah , mengubah atau menghapus data jurusan dan fakultas	Mengelola data jurusan dan fakultas
9	SRS-F-08	Sistem menyediakan fungsi untuk dapat masuk ke dalam sistem	Login

**Tabel A.6** Penambahan kelas pada iterasi ke 1

Nama Kelas	Nama Method
<i>UserController</i>	+ proses_add() + update()
<i>beasiswaController</i>	+ proses_add() + update() + proses_daftar()
<i>fakultasController</i>	+ proses_add_fakultas() + proses_add_jurusan()

**A.2 Proses Iterasi ke-2**



Iterasi ke-2 dilakukan karena terjadi perubahan pada kebutuhan perangkat lunak setelah dilakukan evaluasi. Hasil dari evaluasi tersebut adalah sebagai berikut.

1. Pengurangan persyaratan non fungsional dari 3 menjadi 1.
2. Penambahan method pada diagram kelas.

Adapun perubahan pada kebutuhan perangkat lunak dan perancangan dapat dilihat pada table

**Tabel A.7** Persyaratan Non Fungsional iterasi ke-2

Kode	Deskripsi persyaratan
SRS-NF-01	Sistem dapat dibuka pada berbagai macam <i>web browser</i> .

**Tabel A.8** Penambahan kelas pada iterasi ke-2

Nama Kelas	Nama Method
reportController	+ publish_result()
beasiswa	-link: String +setlink() +getlink()
fakultas	-link: String +setlink() +getlink()
participant	+get_detail_participant(int,String,String) )
beasiswa	+get_detail_beasiswa(bigint,String)