



**PENGEMBANGAN SISTEM INFORMASI MANAJEMEN TESIS
(Studi Kasus: Program Studi Magister Administrasi Bisnis
Fakultas Ilmu Administrasi UB)**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Afif Nandya Saputra
135150401111071



**PROGRAM STUDI SISTEM INFORMASI
JURUSAN SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2017**



PENGESAHAN

PENGEMBANGAN SISTEM INFORMASI MANAJEMEN TESIS
(Studi Kasus: Program Studi Magister Administrasi Bisnis Fakultas Ilmu
Administrasi UB)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Afif Nandya Saputra
NIM: 135150401111071

Skripsi ini telah diuji dan dinyatakan lulus pada
10 Januari 2018
Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

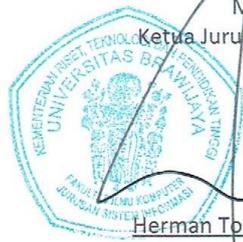
Satrio Agung Wicaksono, S.Kom.,
M.Kom.
NIP: 19860521 201212 1 001

Dosen Pembimbing II

Fajar Pradana, S.ST, M.Eng
NIP: 19871121 201504 1 004

Mengetahui
Ketua Jurusan Sistem Informasi

Herman Tolle, Dr. Eng., S.T., M.T
NIP: 19740823 200012 1 001





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, 8 Januari 2018



Afif Nandya Saputra

NIM: 135150401111071



KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT karena berkat Rahmat dan Karunia-Nya penulis dapat menyelesaikan penyusunan skripsi ini. Shalawat beserta salam semoga senantiasa terlimpah curahkan kepada Nabi Muhammad SAW, kepada keluarganya, para sahabatnya, hingga kepada umatnya hingga akhir zaman, amin. Penulisan skripsi ini diajukan untuk memenuhi salah satu syarat Memperoleh gelar Sarjana pada Program Studi Sistem Informasi Fakultas Ilmu Komputer Universitas Brawijaya. Judul skripsi ini adalah “Pengembangan Sistem Informasi Manajemen Tesis (Studi Kasus: Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB)”.

Dalam penyusunan dan penulisan skripsi ini tidak terlepas dari bantuan, bimbingan serta dukungan dari berbagai pihak. Oleh karena itu dalam kesempatan ini penulis dengan senang hati menyampaikan terima kasih kepada yang terhormat:

1. Bapak, Ibu, dan seluruh keluarga yang dengan tulus selalu memberi doa, dukungan baik berupa materil maupun non materil, nasihat untuk bagaimana sabar, dan tetap berusaha serta motivasi yang tiada henti untuk memberikan semangat kepada penulis dalam pengerjaan skripsi ini hingga selesai.
2. Bapak Satrio Agung Wicaksono, S.Kom, M.Kom sebagai dosen pembimbing I dan Bapak Fajar Pradana, S.ST, M.Eng sebagai dosen pembimbing II yang selalu dengan senang hati dan sabar dalam memberikan arahan, masukan, saran dukungan, nasihat, dan motivasi dalam pengerjaan skripsi ini. Serta penulis banyak belajar hal-hal baru selama proses pengerjaan skripsi kepada beliau.
3. Bapak Aryo Pinandito, ST, M.MT sebagai dosen penasehat akademik yang selalu memberikan nasehat kepada penulis selama menempuh masa studi.
4. Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB yang dengan baik memberikan bantuan berupa data-data yang dibutuhkan untuk menyelesaikan skripsi ini.
5. Seluruh Dosen Fakultas Ilmu Komputer yang telah memberikan ilmu kepada penulis dari awal sampai akhir masa studi.
6. Seluruh Civitas Akademika Fakultas Ilmu Komputer Universitas Brawijaya yang telah banyak memberikan dukungan dan bantuan selama masa studi dan penyelesaian skripsi ini.
7. Teman-teman Sistem Informasi 2013 yang penulis tidak bisa sebutkan namanya satu persatu yang telah membantu penulis secara langsung maupun tidak langsung.
8. Serta semua pihak yang telah terlibat dalam pengerjaan skripsi ini yang tidak bisa disebutkan penulis satu persatu.



Semoga Allah SWT memberikan balasan yang berlipat ganda kepada semuanya. Demi perbaikan selanjutnya, saran dan kritik yang membangun akan penulis terima dengan senang hati. Akhirnya, hanya kepada Allah SWT penulis serahkan segalanya mudah-mudahan dapat bermanfaat khususnya bagi penulis umumnya bagi kita semua.

Malang, 1 Desember 2017

Penulis

afifnandya@gmail.com



ABSTRAK

Magister Administrasi Bisnis merupakan program studi magister pada Fakultas Ilmu Administrasi Universitas Brawijaya Malang. Saat ini setiap tahapan administrasi tesis tersebut menggunakan formulir-formulir fisik yang harus diisi mahasiswa. Formulir fisik tersebut dipindah tangan atau diteruskan ke pihak-pihak yang terlibat dalam alur administrasi tesis guna mendapat persetujuan ataupun tanda tangan. Penggunaan formulir fisik yang dipindah tangankan atau diteruskan guna mendapat persetujuan ini dapat memperlambat alur administrasi tesis. Hal tersebut terjadi karena formulir fisik tersebut diserahkan dan diperiksa secara manual. Berdasarkan masalah tersebut, maka solusi yang dapat diberikan adalah mengembangkan suatu sistem informasi manajemen tesis yang mempunyai fungsi untuk mengatasi masalah karena penggunaan formulir fisik tersebut. Hasil penelitian ini dihasilkan 52 kebutuhan fungsional dan 1 kebutuhan non fungsional. Dari pengujian *Black-box* didapatkan hasil sebesar 100% valid dari 52 fungsi yang diuji. Berdasarkan pengujian *compatibility*, sistem dapat berjalan dengan lancar pada *web browser* komputer dan *web browser* perangkat bergerak kecuali pada *web browser* IE 9 dan *web browser* BlackBerry 7.1.

Kata kunci: Pengembangan, Sistem Informasi Manajemen



ABSTRACT

Master of Business Administration is a program of magister study at Faculty of Administrative Sciences Universitas Brawijaya Malang. Currently every stage of thesis work is using physical forms that must be filled by students. Physical form is changed hands or forwarded to the parties involved in the thesis work flow for approval or signature. The use of physical forms that are transferred or forwarded in order to obtain this approval may slow the flow of thesis work. This happens because the physical form is submitted and checked manually. Based on the problem, the solution can be given is to develop a thesis management information system that has the function to overcome the problem because of the use of the physical form. The results of this study produced 52 functional requirements and 1 non-functional needs. From the Black-box test results obtained 100% valid of 52 functions tested. Based on compatibility testing, the system can run smoothly on computer web browser and mobile device web browser except on IE 9 web browser and BlackBerry 7.1 web browser.

Keywords: *Development, Management information System*



DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT.....	vii
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xiv
DAFTAR GAMBAR.....	xix
DAFTAR LAMPIRAN	xxi
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	3
1.3 Tujuan	3
1.4 Manfaat.....	3
1.5 Batasan masalah	3
1.6 Sistematika pembahasan.....	4
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Tinjauan Pustaka.....	5
2.2 Profil Prodi Magister Administrasi Bisnis FIA UB.....	6
2.2.1 Visi Program Studi Magister Administrasi Bisnis FIA UB.....	6
2.2.2 Misi Prodi Magister Administrasi Bisnis FIA UB.....	6
2.2.3 Tujuan Program Studi Magister Administrasi FIA UB	7
2.3 Sistem Informasi	7
2.4 Sistem Informasi Manajemen.....	9
2.5 <i>Waterfall Model</i>	10
2.5.1 <i>Communication</i>	11
2.5.2 <i>Planning</i>	12
2.5.3 <i>Modelling</i>	21
2.5.4 <i>Construction</i>	24



2.5.5 Deployment	29
BAB 3 METODOLOGI	31
3.1 Studi Pustaka.....	32
3.2 Pengumpulan Data	32
3.3 Pemodelan Proses Bisnis	32
3.4 Analisis Kebutuhan	32
3.5 Perancangan	32
3.6 Implementasi	32
3.7 Pengujian	33
3.8 Kesimpulan dan Saran	33
BAB 4 ANALISIS KEBUTUHAN	34
4.1 Gambaran Umum Sistem.....	34
4.2 Analisis Proses Bisnis	34
4.2.1 Identifikasi Proses Bisnis Saat ini (as-is).....	34
4.2.2 Identifikasi Proses Bisnis Usulan (to-be).....	43
4.3 Pemetaan Proses Bisnis	51
4.4 Identifikasi Aktor.....	54
4.5 Analisis dan Spesifikasi Kebutuhan.....	55
4.5.1 Kebutuhan Fungsional Pengguna.....	59
4.5.2 Kebutuhan Fungsional Pegawai.....	59
4.5.3 Kebutuhan Fungsional Mahasiswa	60
4.5.4 Kebutuhan Fungsional Dosen Pembimbing.....	61
4.5.5 Kebutuhan Fungsional Dosen Penguji	62
4.5.6 Kebutuhan Fungsional Kaprodi	63
4.5.7 Kebutuhan Fungsional Akademik.....	64
4.5.8 Kebutuhan Non Fungsional.....	65
4.6 Pemodelan Use Case Diagram	65
4.7 Use Case Scenario	66
4.7.1 Use Case Scenario Mendaftar Sebagai Mahasiswa.....	66
4.7.2 Use Case Scenario Mendaftar Sebagai Dosen.....	67
4.7.3 Use Case Scenario Login.....	68
4.7.4 Use Case Scenario Logout	68



4.7.5 <i>Use Case Scenario</i> Menyetujui Pengajuan Judul dan Dosen Pembimbing	69
4.7.6 <i>Use Case Scenario</i> Mengisi Formulir Pengajuan Judul dan Dosen Pembimbing	69
4.7.7 <i>Use Case Scenario</i> Mengisi Formulir Persetujuan Ujian Proposal	70
4.7.8 <i>Use Case Scenario</i> Mengisi Formulir Persetujuan Seminar Hasil	71
4.7.9 <i>Use Case Scenario</i> Mengisi Formulir Persetujuan Ujian Tesis	71
4.7.10 <i>Use Case Scenario</i> Mengisi Formulir Pengajuan Ujian Proposal	72
4.7.11 <i>Use Case Scenario</i> Mengisi Formulir Pengajuan Seminar Hasil	73
4.7.12 <i>Use Case Scenario</i> Mengisi Formulir Pengajuan Ujian Tesis.....	73
4.7.13 <i>Use Case Scenario</i> Mengecek Syarat Pengajuan Judul dan Dosen Pembimbing	74
4.7.14 <i>Use Case Scenario</i> Mengisi Nilai dan Saran Perbaikan Ujian Proposal.....	75
4.7.15 <i>Use Case Scenario</i> Mengisi Nilai dan Saran Perbaikan Seminar Hasil.....	75
4.7.16 <i>Use Case Scenario</i> Mengisi Nilai dan Saran Perbaikan Ujian Tesis	76
4.7.17 <i>Use Case Scenario</i> Menolak Pengajuan Judul dan Dosen Pembimbing	77
4.7.18 <i>Use Case Scenario</i> Menyetujui Pengajuan Judul dan Dosen Pembimbing	78
4.7.19 <i>Use Case Scenario</i> Menyetujui Persetujuan Ujian Proposal.....	78
4.7.20 <i>Use Case Scenario</i> Menolak Persetujuan Ujian Proposal	79
4.7.21 <i>Use Case Scenario</i> Menyetujui Persetujuan Seminar Hasil	79
4.7.22 <i>Use Case Scenario</i> Menolak Persetujuan Seminar Hasil.....	80
4.7.23 <i>Use Case Scenario</i> Menyetujui Persetujuan Ujian Tesis	81
4.7.24 <i>Use Case Scenario</i> Menolak Persetujuan Ujian Tesis.....	81
4.7.25 <i>Use Case Scenario</i> Menyetujui Pengajuan Ujian Proposal	82
4.7.26 <i>Use Case Scenario</i> Menyetujui Pengajuan Ujian Seminar Hasil	82
4.7.27 <i>Use Case Scenario</i> Menyetujui Pengajuan Ujian Tesis.....	83
4.7.28 <i>Use Case Scenario</i> Menentukan Dosen Pembimbing	84
4.7.29 <i>Use Case Scenario</i> Menentukan Dosen Penguji.....	84



4.7.30 Use Case Scenario Menyetujui Menjadi Dosen Pembimbing...	85
4.7.31 Use Case Scenario Melihat Data Tesis	86
4.7.32 Use Case Scenario Melihat Data Pengajuan Judul dan Dosen Pembimbing	86
4.7.33 Use Case Scenario Melihat Data Ujian Proposal	87
4.7.34 Use Case Scenario Melihat Data Seminar Hasil	88
4.7.35 Use Case Scenario Melihat Data Ujian Tesis	88
4.7.36 Use Case Scenario Melihat Profil	89
4.7.37 Use Case Scenario Melihat Profil Mahasiswa	89
4.7.38 Use Case Scenario Melihat Profil Dosen	90
4.7.39 Use Case Scenario Mengubah Password	90
BAB 5 PERANCANGAN	92
5.1 Pemodelan Arsitektur	92
5.2 Pemodelan <i>Sequence Diagram</i>	93
5.2.1 Sequence Diagram Mengisi Formulir Pengajuan Judul dan Dosen Pembimbing	93
5.2.2 Sequence Diagram Mengisi Formulir Persetujuan Ujian Proposal	95
5.2.3 Sequence Diagram Menyetujui Persetujuan Ujian Proposal	96
5.2.4 Sequence Diagram Melihat Profil Mahasiswa	98
5.3 Pemodelan <i>Class Diagram</i>	99
5.4 <i>Physical Data Model</i>	102
5.5 Perancangan Komponen.....	102
5.5.1 Perancangan Komponen <i>Front End</i>	103
5.5.2 Perancangan Komponen <i>Back End</i>	105
5.6 Perancangan Antarmuka	108
5.6.1 Halaman <i>Login</i>	108
5.6.2 Halaman Pengajuan Judul dan Dosen Pembimbing	108
5.6.3 Halaman Detail Progress Pengajuan Judul dan Dosen Pembimbing	109
5.6.4 Halaman Profil Mahasiswa.....	110
BAB 6 IMPLEMENTASI	111
6.1 Spesifikasi Sistem	111



6.2 Implementasi Algoritme	111
6.2.1 Implementasi Front End.....	111
6.2.2 Implementasi <i>Back End</i>	113
6.3 Implementasi Antarmuka	118
6.3.1 Halaman <i>Login</i>	118
6.3.2 Halaman Pengajuan Judul dan Dosen Pembimbing	118
6.3.3 Halaman Detail Progres Pengajuan Judul dan Dosen Pembimbing	119
6.3.4 Halaman Profil Mahasiswa.....	120
BAB 7 PENGUJIAN	121
7.1 Pengujian <i>Black-box</i>	121
7.1.1 Pengujian Validasi Aktor Pengguna	121
7.1.2 Pengujian Validasi Aktor Pegawai	124
7.1.3 Pengujian Validasi Aktor Mahasiswa	127
7.1.4 Pengujian Validasi Aktor Dosen Pembimbing	135
7.1.5 Pengujian Validasi Aktor Dosen Penguji	139
7.1.6 Pengujian Validasi Aktor Kaprodi	140
7.1.7 Pengujian Validasi Aktor Akademik	144
7.2 Pengujian <i>Compatibility</i>	147
BAB 8 Penutup	149
8.1 Kesimpulan.....	149
8.2 Saran	149
DAFTAR PUSTAKA.....	150
LAMPIRAN A HASIL WAWANCARA.....	152
LAMPIRAN B FORMULIR PROSEDUR TESIS	154
B.1 Form Pengajuan Judul dan Dosen Pembimbing.....	154
B.2 Form Persetujuan Komisi Pembimbing	155
B.3 Form Ujian Proposal	156
B.4 Form Seminar Hasil.....	163
B.5 Form Ujian Tesis	171
LAMPIRAN C PROSEDUR TESIS	178
C.1 Prosedur Ujian Proposal	178



Repository Universitas Brawijaya

C.2 Prosedur Seminar Hasil.....	178
C.3 Prosedur Ujian Tesis	179
LAMPIRAN D KODE SUMBER.....	180
D.1 Kode Sumber Vue komponen pengajuan_proposal_list.....	180
D.2 Kode Sumber Vue komponen pengajuan_proposal_form	183



DAFTAR TABEL

Tabel 2.1 Jenis sistem informasi	8
Tabel 2.2 Daftar tipe <i>activity</i> pada BPMN.....	14
Tabel 2.3 Daftar tipe <i>sub-process</i> pada BPMN	15
Tabel 2.4 Daftar tipe <i>start event</i> pada BPMN.....	15
Tabel 2.5 Daftar tipe <i>Intermediate event</i> pada BPMN.....	16
Tabel 2.6 Daftar tipe <i>End event</i> pada BPMN	17
Tabel 2.7 Daftar tipe <i>gateway</i> pada BPMN	18
Tabel 2.8 Elemen dari <i>use case diagram</i>	21
Tabel 2.9 Elemen dari <i>sequence diagram</i>	23
Tabel 4.1 Daftar proses bisnis	51
Tabel 4.2 Pemetaan proses bisnis.....	51
Tabel 4.3 Identifikasi aktor.....	54
Tabel 4.4 Analisis dan spesifikasi kebutuhan.....	55
Tabel 4.5 Kebutuhan fungsional pengguna	59
Tabel 4.6 Kebutuhan fungsional pegawai.....	59
Tabel 4.7 Kebutuhan fungsional mahasiswa.....	60
Tabel 4.8 Kebutuhan fungsional dosen pembimbing	61
Tabel 4.9 Kebutuhan fungsional dosen penguji.....	62
Tabel 4.10 Kebutuhan fungsional Kaprodi.....	63
Tabel 4.11 Kebutuhan fungsional akademik.....	64
Tabel 4.12 Kebutuhan non fungsional	65
Tabel 4.13 <i>Use case scenario</i> mendaftar sebagai mahasiswa	67
Tabel 4.14 <i>Use case scenario</i> Mendaftar Sebagai Dosen	67
Tabel 4.15 <i>Use case scenario login</i>	68
Tabel 4.16 <i>Use case scenario logout</i>	68
Tabel 4.17 <i>Use case scenario</i> menyetujui pengajuan judul tesis dan dosen pembimbing	69
Tabel 4.18 <i>Use case scenario</i> mengisi formulir pengajuan judul dan dosen pembimbing	70
Tabel 4.19 <i>Use case scenario</i> mengisi formulir persetujuan ujian proposal	70



Tabel 4.20 <i>Use case scenario</i> mengisi formulir persetujuan seminar hasil.....	71
Tabel 4.21 <i>Use case scenario</i> mengisi formulir persetujuan ujian tesis	72
Tabel 4.22 <i>Use case scenario</i> mengisi formulir pengajuan ujian proposal.....	72
Tabel 4.23 <i>Use case scenario</i> mengisi formulir pengajuan ujian seminar hasil ...	73
Tabel 4.24 <i>Use case scenario</i> mengisi formulir pengajuan ujian tesis.....	73
Tabel 4.25 <i>Use case scenario</i> mengecek syarat pengajuan judul tesis dan dosen pembimbing	74
Tabel 4.26 <i>Use case scenario</i> mengisi nilai dan saran perbaikan ujian proposal .	75
Tabel 4.27 <i>Use case scenario</i> mengisi nilai dan saran perbaikan seminar hasil...	76
Tabel 4.28 <i>Use case scenario</i> mengisi nilai dan saran perbaikan ujian tesis.....	76
Tabel 4.29 <i>Use case scenario</i> menolak pengajuan judul tesis dan dosen pembimbing	77
Tabel 4.30 <i>Use case scenario</i> menyetujui pengajuan judul tesis dan dosen pembimbing	78
Tabel 4.31 <i>Use case scenario</i> menyetujui persetujuan ujian proposal	78
Tabel 4.32 <i>Use case scenario</i> menolak persetujuan ujian proposal.....	79
Tabel 4.33 <i>Use case scenario</i> menyetujui persetujuan ujian seminar hasil	80
Tabel 4.34 <i>Use case scenario</i> menolak persetujuan seminar hasil	80
Tabel 4.35 <i>Use case scenario</i> menyetujui pengajuan persetujuan ujian tesis	81
Tabel 4.36 <i>Use case scenario</i> menolak persetujuan ujian tesis.....	81
Tabel 4.37 <i>Use case scenario</i> menyetujui pengajuan ujian proposal.....	82
Tabel 4.38 <i>Use case scenario</i> menyetujui pengajuan ujian seminar hasil.....	83
Tabel 4.39 <i>Use case scenario</i> menyetujui pengajuan ujian tesis.....	83
Tabel 4.40 <i>Use case scenario</i> menentukan dosen pembimbing	84
Tabel 4.41 <i>Use case scenario</i> menentukan dosen penguji	85
Tabel 4.42 <i>Use case scenario</i> menyetujui menjadi dosen pembimbing.....	85
Tabel 4.43 <i>Use case scenario</i> melihat data tesis.....	86
Tabel 4.44 <i>Use case scenario</i> melihat data pengajuan judul dan dosen pembimbing	87
Tabel 4.45 <i>Use case scenario</i> melihat data ujian proposal.....	87
Tabel 4.46 <i>Use case scenario</i> melihat data seminar hasil.....	88
Tabel 4.47 <i>Use case scenario</i> melihat data ujian tesis.....	88
Tabel 4.48 <i>Use case scenario</i> melihat profil	89



Tabel 4.49 Use case scenario melihat profil mahasiswa.....	89
Tabel 4.50 Use case scenario melihat profil dosen.....	90
Tabel 4.51 Use case scenario mengubah password.....	90
Tabel 5.1 Perancangan komponen main.js.....	103
Tabel 5.2 Perancangan komponen route.js	103
Tabel 5.3 Perancangan komponen pengajuan_proposal_index	104
Tabel 5.4 Perancangan komponen pengajuan_proposal_list.....	104
Tabel 5.5 Perancangan komponen pengajuan_proposal_form	104
Tabel 5.6 Perancangan komponen fungsi index().....	105
Tabel 5.7 Perancangan komponen fungsi insert_progres().....	106
Tabel 5.8 Perancangan komponen fungsi insert_pengajuan_proposal().....	106
Tabel 5.9 Perancangan komponen fungsi insert_proposal_has_dosen()	107
Tabel 5.10 Perancangan komponen fungsi insert_ubah_status_progres().....	107
Tabel 6.1 Implementasi algoritme main.js.....	111
Tabel 6.2 Implementasi algoritme route.js.....	112
Tabel 6.3 Implementasi algoritme pengajuan_proposal_index	113
Tabel 6.4 Implementasi algoritme fungsi index.....	114
Tabel 6.5 Implementasi algoritme fungsi insert_progres.....	115
Tabel 6.6 Implementasi algoritme fungsi insert_pengajuan_proposal.....	116
Tabel 6.7 Implementasi algoritme fungsi insert_proposal_has_dosen.....	116
Tabel 6.8 Implementasi algoritme fungsi insert_ubah_status_progres.....	117
Tabel 7.1 Kasus uji melakukan login	121
Tabel 7.2 Kasus uji melakukan login alternatif 1: email dan password tidak terdaftar	121
Tabel 7.3 Kasus uji mendaftar sebagai mahasiswa.....	122
Tabel 7.4 Kasus uji mendaftar sebagai mahasiswa alternatif 1: email tidak valid atau sudah terdaftar	122
Tabel 7.5 Kasus uji mendaftar sebagai dosen.....	122
Tabel 7.6 Kasus uji mendaftar sebagai dosen alternatif 1: email tidak valid atau sudah terdaftar	123
Tabel 7.7 Kasus uji mengatur ulang password.....	123
Tabel 7.8 Kasus uji melihat profil mahasiswa	124
Tabel 7.9 Kasus uji melihat profil	124



Tabel 7.10 Kasus uji melihat data tesis	124
Tabel 7.11 Kasus uji melihat data pengajuan judul dan dosen pembimbing	125
Tabel 7.12 Kasus uji melihat data ujian proposal	125
Tabel 7.13 Kasus uji melihat data seminar hasil	125
Tabel 7.14 Kasus uji melihat data ujian tesis	126
Tabel 7.15 Kasus uji melakukan logout.....	126
Tabel 7.16 Kasus uji mengubah password	127
Tabel 7.17 Kasus uji mengubah <i>password</i> alternatif 1: isi kolom <i>password</i> baru dan ulangi <i>password</i> baru tidak sama	127
Tabel 7.18 Kasus uji mengubah <i>password</i>	127
Tabel 7.19 Kasus uji mengubah <i>password</i> alternatif 1: isi kolom <i>password</i> baru dan ulangi <i>password</i> baru tidak sama	128
Tabel 7.20 Kasus uji mengisi formulir pengajuan judul dan dosen pembimbing.....	128
Tabel 7.21 Kasus uji mengisi formulir pengajuan judul dan dosen pembimbing alternatif 1: kolom yang wajib diisi tidak diisi.....	129
Tabel 7.22 Kasus uji melihat data pengajuan judul dan dosen pembimbing	130
Tabel 7.23 Mengisi formulir persetujuan ujian proposal.....	130
Tabel 7.24 Kasus uji mengisi formulir pengajuan ujian proposal	131
Tabel 7.25 Kasus uji melihat data ujian proposal	131
Tabel 7.26 Kasus uji mengisi formulir persetujuan seminar hasil	131
Tabel 7.27 Mengisi formulir pengajuan seminar hasil.....	132
Tabel 7.28 Kasus uji melihat data seminar hasil	133
Tabel 7.29 Kasus uji mengisi formulir persetujuan ujian tesis.....	133
Tabel 7.30 Kasus uji mengisi formulir pengajuan ujian tesis	133
Tabel 7.31 Kasus uji melihat data ujian tesis	134
Tabel 7.32 Kasus uji melihat profil	134
Tabel 7.33 Kasus uji melakukan <i>logout</i>	135
Tabel 7.34 Kasus uji menyetujui menjadi dosen pembimbing	135
Tabel 7.35 Kasus uji megeoloal persetujuan ujian proposal.....	136
Tabel 7.36 Kasus uji mengelola persetujuan seminar hasil	136
Tabel 7.37 Kasus uji megelola persetujuan ujian tesis	137
Tabel 7.38 Kasu uji menyetujui pengajuan ujian proposal	137
Tabel 7.39 Kasus uji menyetujui pengajuan seminar hasil	138



Tabel 7.40 Kasus uji menyetujui ujian tesis	138
Tabel 7.41 Kasus uji menyetujui ujian proposal	139
Tabel 7.42 Kasus uji menyetujui ujian tesis	139
Tabel 7.43 Kasus uji mengelola pengajuan judul dan menentukan dosen pembimbing	140
Tabel 7.44 Kasus uji mengelola persetujuan ujian proposal	140
Tabel 7.45 Kasus uji mengelola persetujuan seminar hasil	141
Tabel 7.46 Kasus uji mengelola persetujuan ujian tesis	142
Tabel 7.47 Kasus uji menyetujui pengajuan ujian proposal dan menentukan dosen penguji.....	142
Tabel 7.48 Kasus uji menyetujui pengajuan seminar hasil	143
Tabel 7.49 Kasus uji menyetujui pengajuan ujian tesis dan menentukan dosen penguji.....	143
Tabel 7.50 Kasus uji melihat profil dosen	144
Tabel 7.51 Kasus uji mengelola pengajuan judul dan dosen pembimbing.....	144
Tabel 7.52 Kasus uji menyetujui pengajuan ujian proposal	144
Tabel 7.53 Kasus uji menyetujui pengajuan seminar hasil	145
Tabel 7.54 Kasus uji menyetujui pengajuan ujian tesis	145
Tabel 7.55 Kasus uji mengisi nilai dan saran perbaikan ujian proposal.....	146
Tabel 7.56 Kasus uji mengisi nilai dan saran perbaikan seminar hasil	146
Tabel 7.57 Kasus uji mengisi nilai dan saran perbaikan ujian tesis	147
Tabel 7.58 Keterangan simbol pada <i>software</i> sortsite	147



DAFTAR GAMBAR

Gambar 2.1 Komponen sistem informasi	7
Gambar 2.2 Tahapan pada <i>waterfall model</i>	10
Gambar 2.3 <i>Use case diagram</i> sistem pengunduh musik.....	22
Gambar 2.4 Contoh class diagram	22
Gambar 2.5 Contoh <i>sequence diagram</i> penjualan kendaraan	24
Gambar 2.6 Perbedaan <i>web</i> biasa dengan SPA	26
Gambar 2.7 Membuat vue komponen	27
Gambar 2.8 Memanggil vue komponen	27
Gambar 2.9 Keluaran vue komponen	27
Gambar 2.10 Cara memasang vue <i>router</i>	28
Gambar 2.11 Cara menggunakan vue <i>router</i>	28
Gambar 2.12 Contoh penggunaan vue router	28
Gambar 2.13 Contoh JWT	28
Gambar 4.1 Proses bisnis pengajuan judul dan pembimbing tesis saat ini.....	36
Gambar 4.2 Proses bisnis ujian proposal saat ini.....	38
Gambar 4.3 Proses bisnis ujian seminar hasil saat ini	40
Gambar 4.4 Proses bisnis ujian tesis saat ini	42
Gambar 4.5 Proses bisnis pengajuan judul dan pembimbing tesis usulan.....	44
Gambar 4.6 Proses bisnis ujian proposal usulan	46
Gambar 4.7 Proses bisnis seminar hasil usulan	48
Gambar 4.8 Proses bisnis ujian tesis usulan	50
Gambar 4.9 Pemodelan <i>use case diagram</i> Sistem Informasi Manajemen Tesis..	66
Gambar 5.1 Pemodelan arsitektur.....	92
Gambar 5.2 Sequence diagram mengisi formulir pengajuan judul dan dosen pembimbing	94
Gambar 5.3 Sequence diagram mengisi formulir persetujuan ujian proposal.....	96
Gambar 5.4 Sequence diagram menyetujui persetujuan ujian proposal.....	98
Gambar 5.5 Sequence diagram melihat profil mahasiswa	99
Gambar 5.6 <i>Class diagram controller</i>	100
Gambar 5.7 <i>Class diagram model</i>	101



Gambar 5.8 <i>Class diagram</i> keseluruhan	102
Gambar 5.9 <i>Physical data model</i>	102
Gambar 5.10 Perancangan antarmuka halaman login	108
Gambar 5.11 Perancangan antarmuka halaman pengajuan judul dan dosen pembimbing	109
Gambar 5.12 Perancangan antarmuka halaman detail progress pengajuan judul dan dosen pembimbing	109
Gambar 5.13 Perancangan antarmuka halaman profil mahasiswa.....	110
Gambar 6.1 Implementasi antarmuka halaman <i>login</i>	118
Gambar 6.2 Implementasi antarmuka halaman pengajuan judul dan dosen pembimbing	119
Gambar 6.3 Implementasi antarmuka halaman detail progres pengajuan judul dan dosen pembimbing	119
6.4 Gambar 6.4 Implementasi antarmuka halaman profil mahasiswa.....	Error!
Bookmark not defined.	
Gambar 7.1 Hasil pengujian <i>compatibility</i>	148



DAFTAR LAMPIRAN

LAMPIRAN A HASIL WAWANCARA.....	152
LAMPIRAN B FORMULIR PROSEDUR TESIS	154
B.1 Form Pengajuan Judul dan Dosen Pembimbing.....	154
B.2 Form Persetujuan Komisi Pembimbing	155
B.3 Form Ujian Proposal	156
B.4 Form Seminar Hasil.....	163
B.5 Form Ujian Tesis	171
LAMPIRAN C PROSEDUR TESIS.....	178
C.1 Prosedur Ujian Proposal	178
C.2 Prosedur Seminar Hasil.....	178
C.3 Prosedur Ujian Tesis	179
LAMPIRAN D KODE SUMBER.....	180
D.1 Kode Sumber Vue komponen pengajuan_proposal_list.....	180
D.2 Kode Sumber Vue komponen pengajuan_proposal_form	183



BAB 1 PENDAHULUAN

1.1 Latar belakang

Perkembangan teknologi informasi yang cepat telah berdampak pada berbagai sektor kehidupan. Dengan adanya teknologi informasi suatu pekerjaan dapat diselesaikan dengan lebih efektif dan efisien. Penggunaan teknologi informasi tidak hanya dimanfaatkan oleh individu tetapi juga perusahaan. Perusahaan yang sadar akan manfaat teknologi informasi mulai menginvestasikan modalnya pada bidang tersebut. Laudon & Laudon (2014) menjelaskan bahwa terjadi peningkatan investasi pada bidang teknologi informasi dari 32 persen menjadi 52 persen dari total modal perusahaan dari tahun 1980 ke 2011. Tujuan dari investasi ini adalah untuk membantu perusahaan dalam menjalankan fungsi bisnisnya. Pemanfaatan teknologi informasi selain untuk membantu menjalankan fungsi bisnis, bila digabungkan dengan fungsi organisasi dan proses manajemen dapat membentuk sebuah sistem informasi.

Sistem informasi merupakan kumpulan dari komponen yang saling berhubungan yang mengumpulkan (*input*), mengubah (*process*), menyimpan dan menyebarkan (*output*) data serta informasi untuk suatu tujuan tertentu (Laudon & Laudon, 2014). Sistem informasi memiliki 3 dimensi yaitu: organisasi, manajemen, dan teknologi informasi. Sistem informasi pada dimensi organisasi mencakup proses bisnis, struktur organisasi, dan budaya organisasi. Dimensi manajemen pada sistem informasi menjelaskan bahwa sistem informasi dapat membantu dalam proses pengambilan keputusan dalam organisasi. Dimensi teknologi informasi mencakup kumpulan *software* dan *hardware* yang mendukung berjalannya sistem informasi. Dengan demikian sistem informasi dapat menciptakan nilai pada organisasi serta membantu proses manajemen dan pengambilan keputusan dengan bantuan teknologi informasi. Instansi pendidikan khususnya perguruan tinggi mulai menggunakan sistem informasi guna membantu tugas-tugas operasionalnya dan fungsi manajemennya.

Program Studi Magister Administrasi Bisnis merupakan salah satu program studi magister yang ada pada Fakultas Ilmu Administrasi (FIA) Universitas Brawijaya. Dalam sistem pendidikan Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB administrasi tesis dibimbing oleh dua dosen pembimbing dengan kualifikasi minimal Doktor dan memiliki jabatan akademik lektor. Tahapan dalam administrasi tesis yang terdiri dari: (a) Proposal penelitian, (b) Pelaksanaan penelitian, (c) Penulisan artikel jurnal dan penulisan naskah tesis, (d) Seminar hasil penelitian, (e) Ujian tesis dan (f) Ujian tesis (Fakultas Ilmu Administrasi UB, 2016). Pada wawancara dalam Lampiran A, setiap tahapan alur administrasi tesis tersebut menggunakan formulir-formulir yang harus diisi mahasiswa. Dimulai dari proses pengajuan judul tesis dan dosen pembimbing, mahasiswa mengisi formulir pengajuan judul tesis dan dosen pembimbing dan diserahkan ke ketua program studi melalui sekretaris program. Ketua program studi selanjutnya menetapkan dosen pembimbing dengan topik tesis yang



diajukan oleh mahasiswa. Dosen pembimbing yang sudah ditunjuk untuk membantu mahasiswa dalam pengerjaan tesis serta memonitoring perkembangannya. Selanjutnya mahasiswa bisa mengisi formulir pengajuan ujian proposal skripsi yang diserahkan ke akademik untuk proses selanjutnya. Ketua program studi menerbitkan surat yang berisi daftar tim penguji serta alokasi waktu untuk melakukan ujian. Jika ujian proposal sudah selesai dan memenuhi syarat maka mahasiswa bisa melanjutkan ke proses pelaksanaan penelitian serta penulisan naskah tesis. Setelah penelitian dilakukan maka mahasiswa bisa mendaftar untuk pelaksanaan seminar hasil pada waktu yang sudah disepakati dengan dosen pembimbing. Setelah seminar hasil, mahasiswa bisa menghadap ketua program studi untuk pengajuan ujian tesis.

Pada alur administrasi tesis saat ini, formulir isian merupakan media yang digunakan dalam tiap tahap alur administrasi tesis. Formulir isian tersebut dipindah tangan atau diteruskan ke pihak-pihak yang terlibat dalam alur administrasi tesis guna mendapat persetujuan ataupun tanda tangan. Penggunaan formulir yang dipindah tangankan atau diteruskan guna mendapat persetujuan ini dapat memperlambat alur administrasi tesis. Hal tersebut terjadi karena formulir tersebut diserahkan dan diperiksa secara manual. Penggunaan formulir isian dalam penyelenggaraan administrasi tesis memiliki beberapa kekurangan. Menurut Mills-Senn (2014) Penggunaan formulir-formulir kertas tersebut memiliki beberapa resiko yaitu hilang, rusak, diambil oleh pihak yang tidak berwenang dan salah dalam pengisian. Selain itu penggunaan formulir isian tersebut membutuhkan tempat penyimpanan dan menimbulkan biaya tambahan. Penggunaan formulir isian juga memerlukan waktu tambahan bagi akademik untuk merekap data ke dalam komputer.

Penggunaan formulir isian pada tiap tahap alur administrasi tesis menyebabkan pihak-pihak penyelenggaraan alur administrasi tesis mengalami kesulitan dalam mengkases informasi. Kaprodi ataupun dosen jika ingin mengetahui informasi tentang progres mahasiswa dalam mengerjakan tesis harus menemui akademik dahulu. Sedangkan pada akademik untuk mengetahui informasi tersebut harus membuka file rekap yang berhubungan dengan administrasi tesis secara manual. Saat ini dosen pembimbing tesis tidak mengetahui berapa jumlah mahasiswa yang dibimbingnya serta progres tiap mahasiswa tersebut.

Berdasarkan masalah yang telah dijelaskan sebelumnya, maka dibutuhkan sebuah sistem informasi manajemen tesis pada Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB yang dapat membantu menyelesaikan masalah yang timbul pada alur administrasi tesis. Sistem informasi tersebut diharapkan dapat mengurangi penggunaan formulir isian dalam tiap alur administrasi tesis dan digantikan dengan formulir digital. Pengguna formulir digital diharapkan dapat mengurangi resiko yang timbul dari formulir isian berbasis kertas dan diharapkan lebih mudah untuk pengaksesan informasi didalamnya. Selain itu dengan adanya sistem informasi manajemen tesis diharapkan informasi yang dibutuhkan tiap pihak yang terlibat didalamnya tersedia dengan cepat dan



mudah. Berdasarkan solusi tersebut, maka dilakukan penelitian penelitian dengan judul “Pengembangan Sistem Informasi Manajemen Tesis (Studi Kasus Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB)”.

1.2 Rumusan masalah

Berdasarkan latar belakang yang sudah dijelaskan, dapat dirumuskan permasalahan yang dibahas dalam penelitian ini sebagai berikut:

1. Apa saja kebutuhan fungsional dan non-fungsional pada sistem informasi manajemen tesis berdasarkan data yang telah didapatkan?
2. Apakah kebutuhan fungsional dan non-fungsional yang telah didapatkan di diterapkan semua ?
3. Bagaimana hasil pengujian Black-box dan pengujian *compatibility* pada sistem informasi manajemen tesis?

1.3 Tujuan

Berdasarkan rumusan masalah yang sudah dijelaskan, penelitian ini memiliki tujuan sebagai berikut:

1. Mengetahui kebutuhan fungsional dan no-fungsional pada sistem informasi manajemen tesis berdasarkan data yang telah didapatkan.
2. Mengetahui kebutuhan fungsional dan non-fungsional sistem infromasi manajemen tesis yang diterapkan.
3. Mengetahui hasil pengujian Black-box dan pengujian *compatibility* sistem informasi Manajemen tesis.

1.4 Manfaat

Adapun manfaat yang ingin dicapai dalam skripsi ini adalah sebagai berikut:

1. Bagi penulis dapat mengetahui bagaimana cara menerapkan ilmu-ilmu yang sudah dipelajari selama kuliah terutama yang berkaitan dengan perancangan, implementasi, dan pengujian sistem informasi.
2. Bagi pihak lain khususnya Program Magister Fakultas Ilmu Administrasi UB mendapatkan sistem informasi manajemen tesis yang diharapkan dapat membantu manajemen tesis dengan lebih baik.

1.5 Batasan masalah

Adapun batasan masalah dari penelitian ini adalah sebagai berikut:

1. Analisis difokuskan pada permasalahan kegiatan manajemen tesis yang ada pada Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB.
2. Data yang digunakan merupakan data yang didapat dari Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB.
3. Proses bisnis yang digambarkan hanya sebagai acuan untuk menggambarkan alur atau jalannya aktivitas.



4. Sistem yang dikembangkan berbasis *web*.

1.6 Sistematika pembahasan

Adapun sistematika pembahasan isi dalam penelitian ini adalah sebagai berikut:

BAB 1 PENDAHULUAN

Berisi latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, dan sistematika penulisan.

BAB 2 LANDASAN KEPUSTAKAAN

Pada bab ini dibahas dasar teori, teori pendukung, temuan, dan atau bahan penelitian sebelumnya yang akan digunakan sebagai dasar pada penulisan penelitian ini.

BAB 3 METODOLOGI

Pada bab ini dibahas mengenai urutan langkah-langkah yang dilakukan dalam membangun sistem informasi manajemen tesis.

BAB 4 ANALISIS KEBUTUHAN

Pada bab ini akan dibahas analisis kebutuhan informasi manajemen tesis berdasarkan data-data yang sudah didapatkan.

BAB 5 PERANCANGAN

Pada bab ini akan dibahas perancangan manajemen tesis berdasarkan kebutuhan sistem yang telah didapatkan.

BAB 6 IMPLEMENTASI

Pada bab ini akan membahas implementasi dari sistem informasi manajemen tesis berdasarkan perancangan yang telah dilakukan pada bab sebelumnya.

BAB 7 PENGUJIAN

Pada bab ini akan membahas pengujian terhadap sistem informasi manajemen tesis yang telah dibangun.

BAB 8 PENUTUP

Pada bab ini berisi kesimpulan dari hasil pengujian sistem informasi manajemen tesis yang akan didapatkan dan saran untuk penelitian atau pengembangan kedepannya.



BAB 2 LANDASAN KEPUSTAKAAN

2.1 Tinjauan Pustaka

Andri & Sutrisno (2013) melakukan penelitian dengan judul Rancang Bangun Sistem Informasi Pengajuan Judul Tugas Akhir dan Skripsi Berbasis *Web Service*. Penelitian ini merupakan penelitian implementatif yang membahas masalah pengajuan judul skripsi yang memakan waktu lama karena mahasiswa harus menyerahkan banyak berkas persyaratan untuk mendapat persetujuan dari tiap bagian. Untuk mempermudah mahasiswa dalam mengajukan judul skripsi maka dibuat sistem informasi ini adalah untuk mempermudah mahasiswa di Universitas Bina Darma dalam pengajuan judul tugas akhir. Peneliti dalam mengumpulkan data dengan melakukan observasi, studi pustaka, dan wawancara. Setelah mendapatkan data yang dibutuhkan, dilanjutkan dengan pengembangan sistem dengan metode *web engineering*. Hasil dari penelitian ini adalah sistem informasi pengajuan judul tugas akhir dan skripsi berbasis *web service*. Dalam penelitian ini menggunakan metode SOA (*service oriented architecture*) sehingga peneliti membuat *web service*. Dengan adanya *web service* tersebut dapat mengintegrasikan antara akademik dan bagian perpustakaan.

Penelitian yang dilakukan Kluge, et al. (2007) melakukan uji pengguna untuk mengukur usability pada aplikasi *web* berbasis AJAX (*Asynchronous javaScript and xml*) dengan aplikasi *web* tanpa menggunakan AJAX. Subyek dari penelitian tersebut yaitu penggunaan AJAX pada form yang mendukung *auto complete* dan form komentar yang otomatis dimasukkan ke tampilan tanpa harus memperbarui halaman. Pengukuran yang digunakan pada penelitian tersebut adalah tingkat kepuasan dan efisiensi. Penelitian tersebut melibatkan 123 orang dengan rentang usia 14 hingga 68 tahun dan terdiri dari 42 orang perempuan dan 81 orang laki-laki. Hasil pengukuran menunjukkan terjadi peningkatan kepuasan pengguna dari 3.7 menjadi 4,37 (dari skala 1 – 5) saat menggunakan form yang mendukung *autocomplete*. Waktu yang dibutuhkan untuk menyelesaikan tugas tertentu juga berkurang saat menggunakan form *autocomplete* dari 92.5s menjadi 71.5s. Dari hasil tersebut dapat disimpulkan bahwa penggunaan AJAX membantu peningkatan dalam hal kepuasan pengguna dan efisiensi waktu.

Ayuba, et al. (2013) melakukan penelitian dengan tujuan untuk membandingkan waktu respons halaman antara *full rendering* dan *partial rendering* pada sistem katalog berbasis *web*. Penelitian ini dilakukan dengan membuat sistem katalog berbasis *web* yang memiliki fungsi sama dengan 2 versi yaitu yang menggunakan AJAX (*Asynchronous javaScript and xml*) dan tanpa AJAX. Fungsi yang diuji adalah fungsi pencarian data katalog dan fungsi *bookmark* katalog. Pengujian dilakukan dengan membandingkan waktu respons halaman saat menjalankan kedua fungsi tersebut. Hasil pengujian fungsi yang tidak menggunakan AJAX membutuhkan 417ms untuk memperbarui seluruh halaman. Sedangkan fungsi yang menggunakan AJAX hanya membutuhkan 16ms untuk memperbarui bagian penting halaman.



Hasil pengujian 2 fungsi tersebut menunjukkan bahwa *web* yang menggunakan AJAX memiliki kelebihan dibanding tanpa menggunakan AJAX dalam hal waktu respons.

2.2 Profil Prodi Magister Administrasi Bisnis FIA UB

Fakultas Ilmu Administrasi (FIA) merupakan salah satu fakultas yang ada di lingkungan Universitas Brawijaya, Lembaga Pendidikan Tinggi Negeri di kota Malang. Fakultas Ilmu Administrasi memperoleh status negeri tahun 1963 bersama-sama fakultas lain di lingkungan universitas Brawijaya Malang berdasarkan Ketetapan Menteri Pendidikan No. 1 tanggal 5 Januari 1963. Dalam perkembangannya hingga saat ini Fakultas Ilmu Administrasi telah berganti nama dua kali. Pertama tahun 1962, yang semula bernama Fakultas Administrasi Niaga berubah menjadi Fakultas Ketatanegaraan dan Ketataniagaan untuk menyesuaikan diri dengan Undang-undang Perguruan Tinggi Nomor 22 tahun 1961. Perubahan nama yang kedua kalinya tahun 1982, yaitu Fakultas Ilmu Administrasi berdasarkan Peraturan Pemerintah nomor 27 tahun 1982 tentang Penataan Fakultas pada Universitas atau Institut Negeri dan Keputusan Presiden Nomor 59 tahun 1982 tentang susunan Organisasi Universitas Brawijaya Malang. Fakultas Ilmu Administrasi mengelola 7 program studi, yang terdiri dari dua program studi Diploma 3 (S0), dua program studi sarjana (S1), 2 program studi magister (S2) dan 1 program studi doktor (S3) (Program Studi Magister Administrasi Bisnis, 2013).

2.2.1 Visi Program Studi Magister Administrasi Bisnis FIA UB

Menjadikan Program Studi Magister Administrasi Bisnis sebagai *center of excellent* di bidang pendidikan, penelitian dan pengabdian kepada masyarakat, berperan aktif dalam peningkatan peradaban, ilmu pengetahuan, teknologi dan mampu menyelesaikan berbagai macam masalah kemasyarakatan yang relevan dalam bidang administrasi bisnis.

2.2.2 Misi Prodi Magister Administrasi Bisnis FIA UB

1. Melaksanakan proses pembelajaran yang berkualitas dan akuntabel untuk mencapai kompetensi program studi.
2. Meningkatkan kualitas kegiatan penelitian untuk menghasilkan produk ilmu pengetahuan dan teknologi di bidang ilmu Administrasi Bisnis / Niaga yang bermanfaat bagi masyarakat dunia usaha dan pemerintah.
3. Meningkatkan aplikasi ilmu dan pengetahuan administrasi bisnis melalui kegiatan pengabdian kepada masyarakat baik melalui kegiatan pelatihan, pengabdian maupun seminar yang bermanfaat bagi masyarakat, dunia usaha dan pemerintah.



2.2.3 Tujuan Program Studi Magister Administrasi FIA UB

Tujuan Program Studi Magister Administrasi Bisnis Menghasilkan Magister Ilmu Administrasi Bisnis yang berkualitas sebagai berikut :

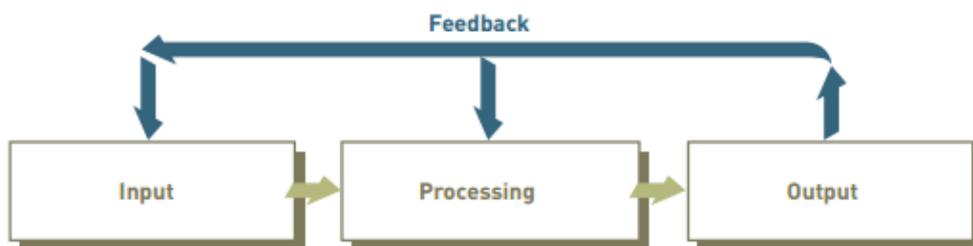
1. Bersifat terbuka, tanggap terhadap perubahan dan kemajuan ilmu dan teknologi maupun masalah yang dihadapi masyarakat.
2. Menguasai teori dan metode penelitian Ilmu Administrasi Bisnis, sehingga analisis hasil penelitian berguna bagi pengembangan ilmu pengetahuan dan kepentingan masyarakat.
3. Mempunyai kemampuan untuk mengembangkan kompetensi profesionalnya dalam suatu organisasi profesi.

2.3 Sistem Informasi

Sistem Informasi merupakan kumpulan komponen yang mengumpulkan (input), memproses (process), menyimpan dan mendistribusikan (output) informasi untuk mendukung pembuatan keputusan dan mengontrol perusahaan. Aktivitas pada sistem informasi menghasilkan informasi yang dapat digunakan oleh perusahaan untuk membuat suatu keputusan, mengontrol proses, menganalisa masalah dan menciptakan produk baru. (Laudon & Laudon, 2014).

Sistem informasi adalah kumpulan dari komponen yang saling berhubungan yang mengumpulkan (*input*), mengubah (*process*), menyimpan dan menyebarkan (*output*) data serta informasi dan menyediakan mekanisme umpan balik (*feedback*) untuk suatu tujuan tertentu. Sistem Informasi dapat membantu perusahaan dalam mencapai tujuannya seperti meningkatkan keuntungan atau meningkatkan layanan pelanggan (Stair & Reynolds, 2012).

Dapat disimpulkan bahwa sistem informasi merupakan kumpulan komponen yang saling berhubungan untuk mengumpulkan, mengubah, menyimpan dan menyebarkan data serta informasi untuk membantu perusahaan dalam mencapai tujuan bisnisnya dan membantu dalam pembuatan keputusan. Hubungan antar komponen digambarkan pada Gambar 2.1.



Gambar 2.1 Komponen sistem informasi

Sumber: Stair & Reynolds (2012)

Pada Gambar 2.1 dijelaskan bahwa terdapat 4 komponen dan hubungan antar komponen dari sistem informasi terdiri dari *input*, *processing*, *output* dan



feedback. Penjelasan tiap komponen tersebut adalah sebagai berikut (Stair & Reynolds, 2012):

1. *Input*

Dalam Sistem Informasi, Input adalah aktivitas mengumpulkan dan mendapatkan data mentah. Contohnya dalam perhitungan gaji karyawan, lama waktu bekerja karyawan harus didapatkan terlebih dahulu sebelum perhitungan gaji.

2. *Processing*

Dalam Sistem Informasi, processing berarti mengubah data menjadi suatu keluaran yang berguna. Processing dapat melibatkan kegiatan perhitungan, perbandingan dan penyimpanan data. Contohnya pada aplikasi penggajian karyawan, jumlah waktu bekerja karyawan dikalikan dengan standar gaji untuk mendapatkan total gajinya.

3. *Output*

Dalam Sistem Informasi, Output merupakan proses yang menghasilkan informasi yang berguna, biasanya berbentuk dokumen atau laporan. Contoh output bisa berupa slip gaji karyawan, laporan bulanan untuk manajer atau informasi tertentu untuk stakeholder.

4. *Feedback*

Dalam Sistem Informasi, Feedback adalah proses perubahan tahap input atau processing berdasarkan informasi yang keluar dari sistem. Contohnya dalam perhitungan gaji karyawan terdapat kesalahan penginputan lama waktu bekerja sehingga perlu dirubah pada proses input lama waktu bekerja.

Sistem informasi menurut Kadir (2014) memiliki beberapa jenis yaitu *E-Commerce*, Sistem Informasi Eksekutif, Sistem Pakar, Sistem Informasi Strategik, Kecerdasan Bisnis, *Customer Relationship Management*, *Supply Chain Management*, *Enterprise Resource Planning* yang dijelaskan pada Tabel 2.1

Tabel 2.1 Jenis sistem informasi

Sistem Informasi	Keterangan
<i>E-Commerce</i>	Sistem untuk melayani penjualan secara elektronik
Sistem Informasi Eksekutif (<i>Executive Information System</i>)	Sistem yang ditujukan secara khusus untuk memberikan informasi penting bagi eksekutif organisasi
Sistem Pakar (<i>Expert System</i>)	Sistem yang berisi pengetahuan pakar yang memungkinkan pemakai akhir seolah-olah berinteraksi dengan pakar



Sistem Informasi Strategik	Sistem Informasi yang ditujukan untuk meraih keunggulan kompetitif, atau memenuhi sasaran-sasaran strategik organisasi
Kecerdasan Bisnis (<i>Bussiness Intelligence</i>)	Segala aplikasi yang difokuskan untuk memperoleh dan menganalisis data (informasi) yang digunakan untuk mengarahkan keputusan-keputusan bisnis strategik
<i>Customer Relationship Management</i> (CRM)	Sistem yang ditujukan untuk menangani proses-proses bisnis yang tertumpu pada pelayanan kepada pelanggan dengan tujuan dapat memberikan layanan yang lebih baik kepada pelanggan, termasuk dalam hal-hal yang disukai pelanggan
<i>Supply Chain Management</i> (SCM)	Sistem yang digunakan untuk pengelolaan hubungan antara proses-proses bisnis kunci dengan vendor, pelanggan, atau mitra bisnis lain
<i>Enterprise Resource Planning</i> (ERP)	Sistem yang mengintegrasikan dan mengoptimasikan proses-proses bisnis di organisasi untuk mendapatkan efisiensi dan ketangkasan dalam menanggapi berbagai situasi di organisasi. ERP menggabungkan kemampuan SCM dan CRM. Konsekuensi dari penggunaan ERP, proses bisnis harus diubah mengikuti proses sistem yang disediakan oleh sistem

Sumber: Kadir (2014)

2.4 Sistem Informasi Manajemen

Sistem Informasi Manajemen adalah kumpulan orang, prosedur, basis data, dan alat yang menyediakan informasi untuk manajer dan pembuat keputusan (Stair & Reynolds, 2012). Sedangkan menurut Kadir (2014) sistem informasi manajemen adalah sistem informasi yang digunakan untuk menyajikan informasi yang digunakan untuk mendukung operasi, manajemen, dan pengambilan keputusan organisasi.

Turban (1995) dalam Kadir (2014) menyebutkan karakteristik sistem informasi manajemen yaitu :

1. Beroperasi pada tugas-tugas yang terstruktur, yakni pada lingkungan yang telah mendefinisikan hal-hal berikut secara tegas dan jelas: prosedur operasi, aturan pengambilan keputusan, dan arus informasi
2. Meningkatkan efisiensi dengan mengurangi biaya



3. Menyediakan laporan dan kemudahan akses yang berguna untuk pengambilan keputusan

Macam-macam laporan yang dihasilkan oleh sistem informasi manajemen menurut Kadir (2014) berupa:

1. Laporan periodis

Laporan Periodis adalah laporan yang dihasilkan dalam selang waktu tertentu seperti harian, mingguna, bulanan

2. Laporan ikhtisar

Laporan ikhtisar merupakan laporan yang memberikan ringkasan terhadap sejumlah data atau informasi

3. Laporan perkecualian

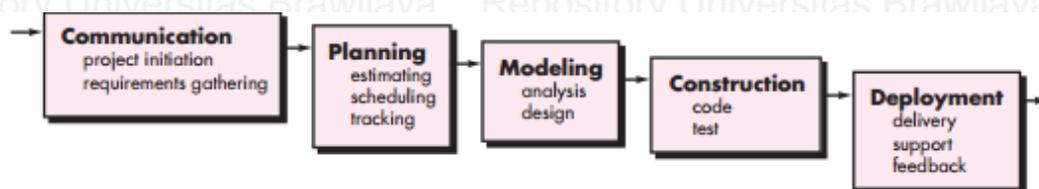
Laporan perkecualian adalah laporan yang hanya akan muncul jika terjadi keadaan tertentu yang sudah ditentukan sebelumnya terpenuhi

4. Laporan perbandingan

Laporan perbandingan menunjukkan dua atau lebih himpunan informasi yang serupa untuk dibandingkan

2.5 Waterfall Model

Menurut Pressman (2010) *waterfall model* adalah model pengembangan perangkat lunak klasik yang sistematis dengan pendekatan berurutan. Pada *waterfall model* tahapan pengembangan perangkat lunak harus berurutan dimana suatu tahapan harus diselesaikan terlebih dahulu untuk melanjutkan ke tahapan berikutnya. Gambar 2.2 menjelaskan tahapan *waterfall model* yaitu : *communication, planning, modeling, construction, dan deployment*.



Gambar 2.2 Tahapan pada *waterfall model*

Sumber: Pressman (2010)

Gambar 2.2 menjelaskan tahapan-tahapan *waterfall model* sebagai berikut (Pressman, 2010) :

1. *Communication*

Merupakan tahapan untuk berkomunikasi dengan *client* demi memahami tujuan dari sistem yang ingin dibuat. Tahap komunikasi ini merupakan tahap Inisialisasi proyek untuk menganalisis masalah yang



dihadapi, mengumpulkan data-data yang dibutuhkan dan mendefinisikan fungsi dari sistem.

2. *Planning*

Setelah data yang diperlukan telah didapatkan dilanjutkan dengan tahap perencanaan. Tahap ini menjelaskan tentang perkiraan tugas teknis yang akan dilakukan, perkiraan resiko yang mungkin terjadi, sumber daya yang diperlukan sistem, penjadwalan pengerjaan sistem serta *tracking* pengerjaan sistem.

3. *Modeling*

Tahap ini berfokus pada penggambaran dari kerja sistem, bagaimana tampilanya dan mendefinisikan arsitektur sistem.

4. *Construction*

Tahapan *Construction* ini merupakan proses pembuatan program berdasarkan pemodelan yang sudah dilakukan pada tahap selanjutnya.

5. *Deployment*

Tahapan *Deployment* merupakan tahapan implementasi program ke customer, pemeliharaan program secara berkala, evaluasi *software*, dan pengembangan *software* berdasarkan umpan balik yang diberikan agar sistem dapat tetap berjalan dan berkembang sesuai dengan fungsinya.

2.5.1 Communication

Merupakan tahapan untuk berkomunikasi dengan *client* demi memahami tujuan dari sistem yang ingin dibuat. Tahap komunikasi ini merupakan tahap Inisialisasi proyek untuk menganalisis masalah yang dihadapi, mengumpulkan data-data yang dibutuhkan dan mendefinisikan fungsi dari sistem. Dalam mengumpulkan data yang dibutuhkan dapat menggunakan metode seperti wawancara, observasi dan studi dokumen.

2.5.1.1 Wawancara

Wawancara dilakukan untuk mendapatkan data lapang yang tidak bisa diakses melalui kegiatan observasi, wawancara juga dimaksudkan untuk memperdalam pengetahuan atau pemahaman tentang obyek yang diteliti, dan hal ini dilakukan untuk menghindari kesalahan dalam mejabarkan masalah yang berkaitan dengan penelitian.

Dalam penelitian ini, wawancara dilakukan untuk mendapatkan informasi detail mengenai alur administrasi tesis yang ada Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB. Dalam penelitian ini, pihak yang diwawancarai adalah akademik.

2.5.1.2 Observasi

Observasi merupakan cara pengumpulan data dengan langsung ke lokasi penelitian. Observasi ini dilakukan untuk memperoleh gambaran secara umum tentang kawasan yang hendak diteliti atau mengamati sumber-sumber peristiwa.



Informasi umum yang diperoleh ini kemudian ditindaklanjuti dengan wawancara untuk mendapatkan informasi detail. Pada penelitian ini, observasi dilakukan dengan mengamati secara langsung alur administrasi tesis yang ada pada Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB.

2.5.1.3 Studi Dokumen

Studi dokumen merupakan cara pengumpulan data dengan mereview dokumen-dokumen yang memiliki hubungan dengan penelitian. Dalam penelitian ini, dokumen yang direview adalah form yang digunakan pada alur administrasi tesis dan diagram alir administrasi tesis pada Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB.

2.5.2 Planning

Tahap ini menjelaskan tentang perkiraan dari tugas teknis yang akan dilakukan, perkiraan resiko yang mungkin terjadi, sumber daya yang diperlukan sistem, penjadwalan pengerjaan sistem serta *tracking* pengerjaan sistem.

2.5.2.1 Proses Bisnis

Proses Bisnis adalah kumpulan pekerjaan yang terorganisir, terkoordinasi dan terfokus untuk menghasilkan produk atau layanan yang bernilai atau untuk tujuan bisnis tertentu (Laudon & Laudon, 2014). Aktivitas pada proses bisnis tersusun dari aliran material, informasi serta pengetahuan dari tiap pihak yang terlibat didalamnya. Menurut Weske (2007) aktivitas pada proses bisnis dapat berupa aktivitas manual, aktivitas interaksi pengguna dan aktivitas sistem.

Aktivitas manual merupakan aktivitas yang tidak didukung oleh sistem informasi. Contoh dari aktivitas manual yaitu mengirim barang kepada perusahaan lain. Aktivitas interaksi pengguna berupa aktivitas yang diselesaikan menggunakan sistem informasi. Contoh aktivitas interaksi pengguna yaitu memasukkan data pegawai ke dalam sistem informasi kepegawaian. Aktivitas sistem merupakan aktivitas yang diselesaikan oleh sistem informasi itu sendiri dan tidak melibatkan interaksi dengan pengguna. Contoh dari aktivitas sistem adalah *e-mail* pemberitahuan biaya langganan jasa penyewaan *hosting* yang dikirim otomatis oleh sistem.

Untuk memahami proses bisnis perusahaan perlu dilakukan kegiatan memodelkan proses bisnis dalam bentuk diagram yang disebut dengan *business process modeling (BPM)*. BPM memiliki fungsi untuk memetakan atau menghubungkan proses bisnis yang ada sehingga proses bisnis yang ada dapat lebih efektif dan efisien (Businessballs, 2017). Ada 2 jenis yang berbeda dari BPM yaitu (Businessballs, 2017):

1. Proses bisnis 'as-is' atau proses bisnis saat ini, adalah gambaran dari proses bisnis yang ada saat ini. Setelah model dibuat, model tersebut dianalisis sehingga proses bisnis yang ada dapat ditingkatkan.



2. Proses bisnis 'to-be' atau proses bisnis yang baru, adalah gambaran proses bisnis yang diusulkan. Proses bisnis usulan nantinya akan dimodelkan dan sebelum diterapkan.

Menurut Przybytek (2011), ada beberapa tahapan yang harus dilakukan untuk menjadikan proses bisnis sebagai dasar pengembangan sistem informasi:

1. *As-Is Business Process Modeling* atau Pemodelan proses bisnis saat ini

Pemodelan proses bisnis yang ada saat ini dapat dilakukan setelah mendapatkan data mengenai proses bisnis yang ada pada organisasi. Untuk mendapatkan data tersebut, dapat dilakukan dengan metode pengumpulan data seperti pengamatan, studi dokumen, serta wawancara. Data proses bisnis saat ini yang didapatkan selanjutnya dimodelkan proses bisnisnya. Proses bisnis saat ini atau proses bisnis 'as-is' yang telah dimodelkan selanjutnya dikonsultasikan dengan pihak stakeholder.

2. *Business Process Improvement* atau Pemodelan proses bisnis usulan

Pada tahap ini pihak stakeholder menganalisa proses bisnis mana yang dirasa kurang efektif dan efisien yang dan dapat dilakukan pengoptimalisasian. Proses bisnis yang diputuskan dapat ditingkatkan dengan sistem informasi selanjutnya akan dimodelkan.

3. *Functional Requirements Elicitation* atau Elisitasi kebutuhan fungsional

Tahap ini menjelaskan bahwa proses bisnis usulan yang telah dimodelkan selanjutnya menjadi dasar pengembangan sistem informasi.

Pada pemodelan proses bisnis dibutuhkan notasi untuk memodelkan proses bisnis yang ada. Beberapa notasi yang biasa digunakan dalam BPM yaitu: EPC (*Event Process Chain*), UML (*Unified Modeling Language*) dan BPMN (*Business Process Model and Notation*) (Przybytek, 2011).

2.5.2.2 Business Process Model and Notation (BPMN)

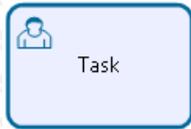
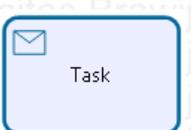
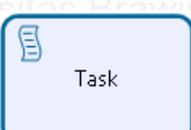
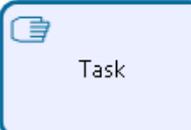
Business process model (BPMN) merupakan suatu standar pada pemodelan proses bisnis, dan juga digunakan sebagai alat desain pada sistem *e-Business* (Rosmala & Falahah, 2007). Tujuan memodelkan proses bisnis menggunakan BPMN adalah notasi yang bisa dimengerti oleh seluruh pihak yang terlibat dari business analyst hingga pengembang yang bertugas membuat sistem (Object Management Group, 2013). Sehingga dapat disimpulkan bahwa BPMN dapat memodelkan proses bisnis yang bisa dipahami oleh semua pihak yang terlibat dan menjembatani antara desain proses bisnis dengan implementasi sistem nantinya.

BPMN memiliki elemen-elemen sebagai berikut: (Bizagi, 2016)

1. *Activity*

Activity merupakan penggambaran dari pekerjaan atau tugas yang dilakukan oleh anggota organisasi. *Activity* dapat berupa *single activity* (aktivitas tunggal) atau *compound activity* (aktivitas gabungan). *Activity* memiliki beberapa tipe yang dijelaskan pada Tabel 2.2.

Tabel 2.2 Daftar tipe *activity* pada BPMN

Nama Elemen	Deskripsi	Notasi
<i>Task</i>	<i>Task</i> adalah aktivitas dasar didalam <i>process flow</i> . <i>Task</i> digunakan apabila suatu proses tidak bisa lebih di detailkan.	
<i>Pengguna Task</i>	<i>Pengguna Task</i> adalah <i>task</i> yang dilakukan oleh pengguna dengan bantuan <i>software</i> .	
<i>Service Task</i>	<i>Service Task</i> adalah <i>task</i> yang menggunakan sebuah layanan yang bisa berupa <i>Web service</i> atau aplikasi yang otomatis.	
<i>Receive Task</i>	<i>Receive Task</i> adalah <i>task</i> yang menunggu pesan datang dari peserta luar (relatif dari proses).	
<i>Send Task</i>	<i>Send Task</i> adalah <i>task</i> yang mengirim ke peserta luar (relatif dari proses).	
<i>Script Task</i>	<i>Script Task</i> adalah <i>task</i> yang dieksekusi oleh <i>Business Process Engine</i> .	
<i>Manual Task</i>	<i>Manual Task</i> adalah <i>task</i> yang dikerjakan tanpa bantuan eksekusi suatu proses bisnis atau aplikasi.	

Sumber: Bizagi (2016)

2. Sub-process

Sub-process adalah *compound activities* (aktivitas gabungan) yang ada pada suatu proses. Aktivitas gabungan berarti aktivitas tersebut dapat



dipecah menjadi tingkatan yang lebih detail. *Sub-process* memiliki beberapa tipe yang dijelaskan pada Tabel 2.3.

Tabel 2.3 Daftar tipe *sub-process* pada BPMN

Nama Elemen	Deskripsi	Notasi
<i>Sub-process</i>	<i>Sub-process</i> adalah aktivitas gabungan yang gambaran detailnya sudah dimodelkan menggunakan <i>activities</i> , <i>gateways</i> , <i>events</i> , dan <i>sequence flows</i> .	
<i>Reusable Sub-process</i>	<i>Reusable Sub-process</i> adalah notasi yang digunakan untuk menandakan proses yang sudah ada digunakan kembali.	
<i>Event Sub-process</i>	<i>Event Sub-process</i> berada di dalam sebuah <i>subprocess</i> . Aktivitas dalam <i>event subprocess</i> akan dijalankan ketika <i>event</i> yang didefinisikan terpenuhi.	

Sumber: Bizagi (2016)

3. *Event*

Event merupakan peristiwa yang terjadi didalam sebuah proses dan mempengaruhi jalanya proses. *Event* biasanya memiliki pemicu atau hasil tertentu. *Event* dibagi menjadi *start event*, *Intermediate event* dan *end event*

a. *Start event*

Notasi yang menandakan bahwa sebuah proses dimulai. *Start event* memiliki beberapa tipe yang dijelaskan pada Tabel 2.4.

Tabel 2.4 Daftar tipe *start event* pada BPMN

Nama Elemen	Deskripsi	Notasi
<i>Start Event</i>	Notasi yang menandakan bahwa sebuah proses dimulai.	 Start Event



<i>Message Start Event</i>	Notasi yang digunakan saat sebuah <i>message</i> sampai dan memicu dimulainya proses.	 Message Start Event
<i>Timer Start Event</i>	Notasi yang digunakan untuk memulai sebuah proses pada waktu tertentu.	 Timer Event
<i>Conditional Start Event</i>	Notasi yang digunakan untuk memulai sebuah proses saat kondisinya terpenuhi.	 Conditional Start Event

Sumber: Bizagi (2016)

b. Intermediate event

Intermediate event merupakan notasi yang menandakan bahwa terjadi sesuatu antara awal dan akhir sebuah proses. *Intermediate event* memiliki beberapa tipe yang dijelaskan pada Tabel 2.5.

Tabel 2.5 Daftar tipe *Intermediate event* pada BPMN

Nama Elemen	Deskripsi	Notasi
<i>Intermediate Event</i>	Notasi yang menandakan bahwa terjadi sesuatu antara awal dan akhir sebuah proses.	 Intermediate Event
<i>Message Event</i>	Notasi yang menandakan <i>event</i> dapat mengirim atau menerima sebuah pesan. <i>Catch Message Event</i> menunggu pesan untuk diterima sebelum proses dilanjutkan. <i>Throw Message Event</i> mengirim pesan ke <i>participant</i> lain.	 Throw Message
<i>Timer Event</i>	Notasi yang menandakan penundaan didalam proses.	 Timer Event



<i>Escalation Event</i>	Notasi yang menandakan sebuah peningkatan didalam proses.	 Escalation Event
<i>Compensation Event</i>	Notasi yang menandakan adanya sebuah kompensasi didalam proses.	 Compensation Event
<i>Conditional Event</i>	Notasi yang menandakan sebuah event dijalankan apabila kondisi terpenuhi.	 Conditional Event

Sumber: Bizagi (2016)

c. *End event*

End event merupakan notasi yang menandakan bahwa event sudah selesai atau berakhir. *End event* memiliki beberapa tipe yang dijelaskan pada Tabel 2.6

Tabel 2.6 Daftar tipe *End event* pada BPMN

Nama Elemen	Deskripsi	Notasi
<i>End Event</i>	Notasi yang menandakan akhir dari sebuah proses.	 End Event
<i>Message End</i>	Notasi yang menandakan pesan yang dikirimkan saat <i>flow</i> sudah berakhir.	 Message End
<i>Escalation End</i>	Notasi yang menandakan sebuah <i>escalation</i> dibutuhkan saat <i>flow</i> sudah berakhir.	 Escalation End
<i>Error End</i>	Notasi yang menandakan sebuah <i>error</i> tertentu harus dihasilkan. Semua <i>threads</i> aktif yang ada pada proses akan dihentikan.	 Error End



<i>Cancel End</i>	Notasi yang digunakan pada <i>Transaction sub-process</i> . Notasi ini menandakan bahwa <i>Transaction</i> tersebut harus digagalkan dan <i>alternate flow</i> bisa mulai berjalan.	 Cancel End
<i>Multiple End</i>	Notasi yang menandakan bahwa terdapat banyak konsekuensi dari berakhirnya <i>flow</i> .	 Multiple End
<i>Terminate End</i>	Notasi yang mengakhiri sebuah proses dan semua aktivitas didalamnya seketika.	 Terminate End

Sumber: Bizagi (2016)

4. Gateway

Gateway digunakan untuk mengatur *divergence* (titik percabangan) dan *convergence* (titik pertemuan) dari aliran proses. *Gateway* memiliki beberapa tipe yang dijelaskan pada Tabel 2.7.

Tabel 2.7 Daftar tipe gateway pada BPMN

Nama Elemen	Deskripsi	Notasi
<i>Exclusive Gateway</i>	Sebagai titik percabangan: Digunakan untuk membuat jalur alternatif didalam proses, tetapi hanya satu yang terpilih Sebagai titik pertemuan: Digunakan untuk menggabungkan jalur alternatif.	 Exclusive Gateway Exclusive Gateway
<i>Event Based Gateway</i>	Notasi yang digunakan sebagai titik percabangan proses yang jalur alternatif didalamnya dijalankan tergantung pada <i>event</i> yang terjadi. Saat <i>event</i> pertama dipicu maka jalur alternatifnya akan dijalankan dan jalur alternatif yang lain menjadi tidak valid.	 Event Based Gateway

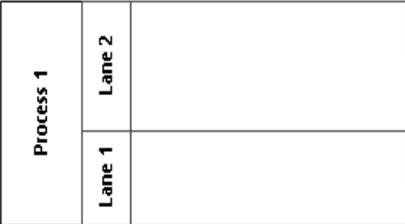


<p><i>Exclusive Event Based Gateway</i></p>	<p>Notasi yang merupakan variasi dari <i>Event based gateway</i> yang hanya digunakan untuk instansiasi proses. Salah satu event yang ada pada konfigurasi gateway harus dipicu terlebih dahulu untuk menginstansiasi proses.</p>	 <p>Exclusive Event Based Gateway</p>
<p><i>Paralell Event Based Gateway</i></p>	<p>Notasi yang mirip dengan <i>Exclusive Event Based Gateway</i> namun semua event yang ada pada konfigurasi gateway harus dipicu semuanya agar bisa menginstansi proses.</p>	 <p>Paralell Event Based Gateway</p>
<p><i>Paralell Gateway</i></p>	<p>Sebagai titik percabangan: Digunakan untuk membuat jalur alternatif didalam proses tanpa mengecek tertentu.</p> <p>Sebagai titik pertemuan: Digunakan untuk menggabungkan jalur alternatif. <i>Gateway</i> ini menunggu semua <i>flow</i> yang terhubung masuk dahulu sebelum melanjutkan ke proses selanjutnya.</p>	 <p>Paralell Gateway</p>
<p><i>Inclusive Gateway</i></p>	<p>Sebagai titik percabangan: menggambarkan titik pecabangan dimana jalur alternatifnya tergantung kondisi <i>expressions</i>.</p> <p>Sebagai titik pertemuan: Digunakan untuk menggabungkan kombinasi dari jalur alternatif dan jalur paralel.</p>	 <p>Inclusive Gateway</p>

Sumber: Bizagi (2016)

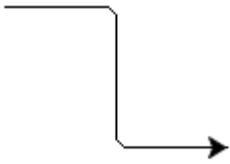


5. Swimlane

Nama Elemen	Deskripsi	Notasi
<i>Pool</i>	Sebuah <i>pool</i> merupakan wadah dari satu proses yang didalamnya terdapat aliran aktivitas.	
<i>Lane</i>	<i>Lane</i> merupakan bagian-bagian yang terdapat di dalam pool. Sebuah lane digunakan untuk membedakan elemen bisnis seperti posisi, departemen dan fungsi internal	

Sumber: Bizagi (2016)

6. Connector

Nama Elemen	Deskripsi	Notasi
<i>Sequence Flow</i>	Sequence flow digunakan untuk menunjukkan urutan activity yang akan dilakukan didalam proses.	
<i>Association</i>	<i>Association</i> digunakan untuk menghubungkan informasi dan artefak dengan <i>flow object</i> .	
<i>Message Flow</i>	<i>Message flow</i> digunakan untuk menunjukkan aliran informasi antara 2 entitas.	

Sumber: Bizagi (2016)



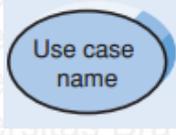
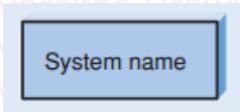
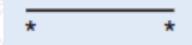
2.5.3 Modelling

Tahap ini berfokus pada penggambaran dari kerja sistem, bagaimana tampilannya dan mendefinisikan arsitektur sistem.

2.5.3.1 Use Case Diagram

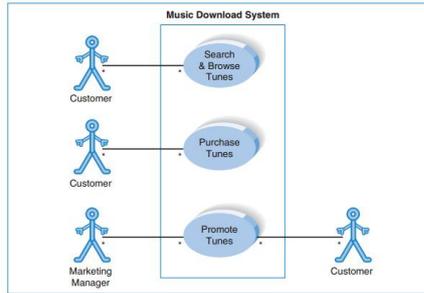
Use case diagram merupakan diagram yang menggambarkan interaksi antara pengguna dengan sistem. *Use case diagram* dibuat pada awal pengembangan karena dapat dengan mudah menjelaskan bagaimana interaksi antara aktor dan sistem nantinya (Dennis, et al., 2012). Pada Tabel 2.8 dijelaskan elemen dari *use case diagram*.

Tabel 2.8 Elemen dari *use case diagram*

Nama dan Deskripsi	Simbol
<p>Aktor</p> <ul style="list-style-type: none"> • Adalah orang yang berinteraksi dengan sistem • Memiliki label nama aktor • Berada diluar <i>system boundary</i> 	
<p>Use case</p> <ul style="list-style-type: none"> • Menggambarkan fungsi sistem • Bisa menjadi perpanjangan usecase lain • Berada di dalam <i>system boundary</i> • Menggunakan kalimat kerja yang deskriptif 	
<p>System boundary</p> <ul style="list-style-type: none"> • Nama sistem ditulis diatas • Menggambarkan cakupan sistem 	
<p>Accosiation relationship</p> <ul style="list-style-type: none"> • Menghubungkan usecase dengan aktor 	

Sumber: Dennis, et al. (2012)

Pada Gambar 2.3 merupakan contoh dari *use case diagram* dari sistem *download music*.

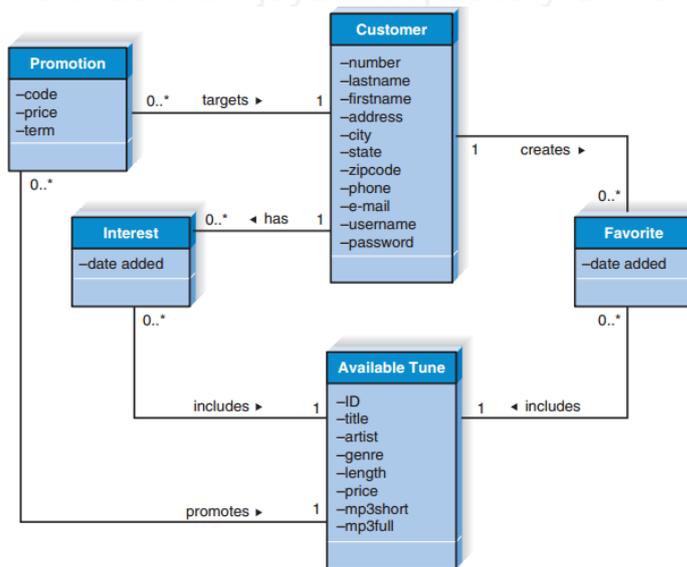


Gambar 2.3 Use case diagram sistem pengunduh musik

Sumber: Dennis, et al. (2012)

2.5.3.2 Class Diagram

Class diagram merupakan diagram yang menampilkan kelas dan hubungan antar kelas yang tetap konstan dari sistem. Class diagram mirip dengan entity relationship diagram (ERD) namun memiliki kelas yang didalamnya terdapat atribut, fungsi dan kondisi (Dennis, et al., 2012). Pada Gambar 2.2 digambarkan contoh dari class diagram.



Gambar 2.4 Contoh class diagram

Sumber: Dennis,et al. (2012)

2.5.3.3 Sequence Diagram

Sequence Diagram digunakan untuk menunjukkan komunikasi antara objek pada saat suatu fungsi dijalankan. Sequence diagram menampilkan pemanggilan metode menggunakan anak panah horizontal dari pemanggil metode ke tujuan pemanggil, dilabelkan dengan nama metode dan bisa ditambahkan nama parameter, tipe data, dan tipe kembalian jika ada (Dennis, et al., 2012). Pada Tabel 2.9 elemen dari sequence diagram.



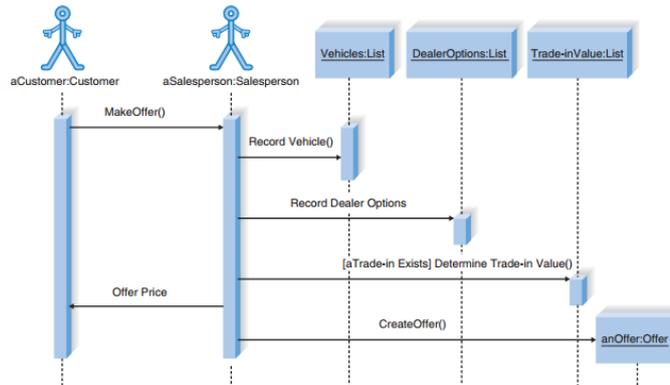
Tabel 2.9 Elemen dari *sequence diagram*

Nama dan Deskripsi	Simbol
<p>Aktor</p> <ul style="list-style-type: none"> • Adalah orang yang berinteraksi dengan sistem • Berada di bagian atas diagram • Berperan mengirim atau menerima pesan 	
<p>Object</p> <ul style="list-style-type: none"> • Berperan mengirim atau menerima pesan • Berada di bagian atas diagram 	
<p>Lifeline</p> <ul style="list-style-type: none"> • Menunjukkan waktu hidup dari proses 	
<p>Focus Control</p> <ul style="list-style-type: none"> • Berbentuk persegi panjang yang berada diatas lifeline • Menunjukkan suatu objek mengirim atau menerima pesan 	
<p>Message</p> <p>a) Membawa informasi dari objek ke objek lain</p>	
<p>Object Destruction</p> <p>b) Menandakan bahwa suatu objek sudah tidak digunakan lagi</p>	

Sumber: Dennis, et al. (2012)



Pada Gambar 2.5 digambarkan contoh *sequence diagram* dari proses penjualan kendaraan



Gambar 2.5 Contoh *sequence diagram* penjualan kendaraan

Sumber: Dennis, et al. (2012).

2.5.3.4 Pemodelan Arsitektur

Desain arsitektur merupakan konsep yang berfokus memahami bagaimana sistem seharusnya tersusun dan desain dari keseluruhan arsitektur dari sistem (Pressman, 2010). Desain arsitektur menghasilkan sebuah model arsitektur yang menjelaskan bagaimana sistem tersusun sebagai komponen yang saling berhubungan.

2.5.3.5 Physical Data Model

Data model merupakan suatu cara merepresentasikan data yang digunakan dan yang diciptakan di dalam sebuah sistem (Dennis, et al., 2012). Cara penggambaran data model bersifat bebas sehingga akan memudahkan mencocokkan model tersebut dengan kebutuhan bisnis yang sesungguhnya. Sedangkan Physical data model merupakan cara penggambaran bagaimana data akan tersimpan secara teknis (Dennis, et al., 2012). Jadi dapat disimpulkan bahwa physical data model merupakan bentuk teknis dari data model yang sudah siap diimplementasikan.

Ada beberapa tahapan mengubah data model yang bersifat konseptual ke *physical data model* yang siap diimplementasikan (Dennis, et al., 2012):

1. Mengubah entitas menjadi tabel
2. Mengubah atribut menjadi kolom
3. Menambahkan *primary key*
4. Menambahkan komponen berbasis sistem

2.5.4 Construction

Tahapan *Construction* ini merupakan proses pembuatan program berdasarkan pemodelan yang sudah dilakukan pada tahap sebelumnya.



2.5.4.1 Service Oriented Architecture (SOA)

Service Oriented Architecture (SOA) adalah pola arsitektur yang mendukung konsep orientasi layanan. *SOA* merupakan pendekatan bisnis untuk membangun sistem IT yang mampu untuk meningkatkan asset bisnis yang sudah ada, membuat bisnis yang baru, memudahkan dalam investasi perubahan yang diperlukan untuk mendukung bisnis (Opengroup, 2016).

Dengan *SOA*, bisnis dan IT dapat saling bertemu dan bekerja sama untuk membangun teknologi yang bisa membangun perusahaan. *SOA* membantu perusahaan untuk mengembangkan *software* yang dibutuhkan untuk mengikuti trend dan peluang atau ide-ide baru (Judith Hurwitz, 2009).

2.5.4.2 Web Service

Web Service merupakan sistem *software* yang didesain untuk membantu komunikasi antar mesin-mesin dalam suatu jaringan (W3C, 2004a). *Web service* juga bisa diartikan sebagai sistem *software* yang diidentifikasi melalui *Uniform Resource Identifiers (URI)* dimana menggunakan format XML (*extensible markup language*). *Web service* harus bisa digunakan oleh sistem *software* yang lain menggunakan pesan berbasis XML (*extensible markup language*) melalui *internet protocol* (W3c, 2004b).

2.5.4.3 Asynchronous JavaScript and XML (AJAX)

AJAX merupakan suatu teknologi web yang memungkinkan suatu aplikasi web dapat berinteraksi dengan pengguna dan bergerak di background untuk mengambil data dari *web server* secara bersamaan (Ford, 2009). Data yang didapat dari *web server* ditampilkan pada *web browser* tanpa harus memperbarui halaman. Hal ini membuat aplikasi *web* menjadi dinamis saat menggunakan Ajax.

2.5.4.4 Javascript Object Notation (JSON)

Menurut Deitel (2012) JSON (JavaScript Object Notation) adalah suatu format pertukaran data komputer yang berbasis teks, dapat terbaca oleh manusia, digunakan untuk mempresentasikan struktur data sederhana, dan tidak bergantung dengan bahasa apapun.

Format JSON sering digunakan untuk mentransmisikan data terstruktur melalui koneksi jaringan. Secara umum, JSON digunakan untuk mentransmisikan data antara server dan aplikasi web. Format JSON sering digunakan untuk pertukaran data terstruktur melalui koneksi jaringan, terutama untuk pengiriman data antara server dan aplikasi web melayani sebagai alternatif ke XML (*extensible markup language*).

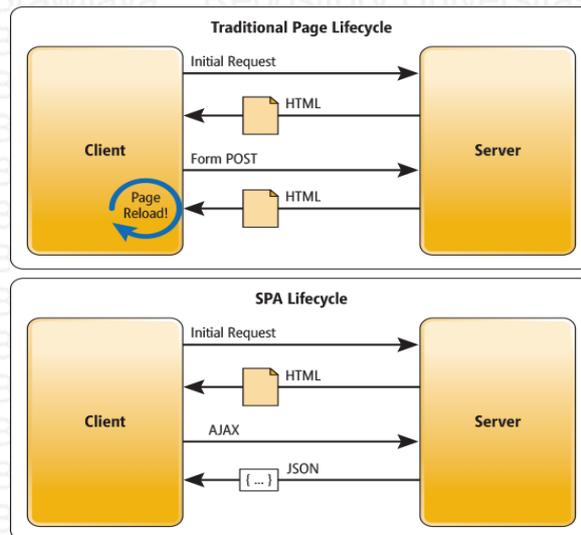
2.5.4.5 Single Page Application (SPA)

Single page application (SPA) merupakan aplikasi *web* atau *website* yang memiliki satu halaman *web* dengan tujuan menyediakan *user experience* yang lebih responsif dan interface yang interaktif. Tujuan utama dari SPA adalah



kemampuan untuk memperbaiki sebagian dari *interface* tanpa mengirim atau menerima *request* untuk *reload* seluruh halaman (Monteiro, 2014).

SPA merupakan aplikasi *web* yang meload semua *resource* yang dibutuhkan pada *request* pertama dan selanjutnya komponen halaman akan digantikan dengan komponen yang lain tergantung interaksi pengguna. SPA terdiri dari individual komponen yang bisa diganti atau diperbarui secara terpisah tanpa harus memperbaiki seluruh halaman pada tiap aksi pengguna (Jadhav, 2015). Dibawah ini adalah gambar perbedaan *web* biasa dengan SPA. Pada Gambar 2.5 digambarkan perbedaan *lifecycle* dari *web* biasa dengan *single page application*.



Gambar 2.6 Perbedaan *web* biasa dengan SPA

Sumber: Wasson (2013)

SPA memiliki beberapa kelebihan dibandingkan dengan *multipage application* yaitu (Khadka, 2016):

1. Mengurangi kode yang bergantung pada struktur halaman

Saat ini *javascript* digunakan untuk memanipulasi struktur ataupun konten dari halaman secara dinamis. Untuk bisa melakukan hal tersebut *javascript* membutuhkan id atau *class* dari elemen. Keterikatan ini membuat *javascript* akan gagal dijalankan apabila *id* atau *class* dari elemen tersebut berubah. Sehingga *id* atau *class* dari elemen penting untuk tidak sembarangan dirubah agar *javascript* bisa berjalan dengan baik.

2. Sisi *server* yang independen

SPA merupakan aplikasi yang berjalan pada sisi *client*. SPA berkomunikasi ke *web server* dengan format xml atau json. Dengan begitu, bagaimanapun kode pada sisi *server* tidak menjadi masalah untuk SPA asalkan tetap mengirimkan data dengan format xml atau json.



3. Transisi yang lebih halus

Pada *web* tradisional saat terjadi perpindahan halaman akan menampilkan halaman putih kosong untuk beberapa waktu. Halaman putih tersebut muncul karena *website* membutuhkan waktu untuk menampilkan konten *website* yang diambil dari *web server*. Sedangkan pada SPA, hal ini tidak terjadi sehingga menciptakan *pengguna experience* yang lebih baik.

2.5.4.6 Vue.JS

Vue.js merupakan kerangka kerja yang progresif untuk membangun antarmuka pengguna. Vue.js sendiri hanya fokus pada lapisan antarmuka saja sehingga akan mudah untuk diimplementasikan pada proyek yang sudah ada atau diintegrasikan dengan library lain. Vue.js juga dapat digunakan untuk mengembangkan single page application dengan bantuan *vue router* dan *vue component*.

1. Vue component

Vue component merupakan fitur yang paling berguna dari vue. Dengan memanfaatkan *Vue component*, dapat mengekspansilasi kode program sehingga dapat digunakan ulang ulang. Pada Gambar 2.7 merupakan contoh dari membuat vue komponen.

```
Vue.component('my-component', {
  template: '<div>A custom component!</div>'
});
new Vue({
  el: '#example'
});
```

Gambar 2.7 Membuat vue komponen

Pada Gambar 2.8 merupakan cara dari memanggil vue komponen yang sudah dibuat sebelumnya.

```
<div id="example">
  <my-component></my-component>
</div>
```

Gambar 2.8 Memanggil vue komponen

Keluaran dari komponen tersebut dapat dilihat pada Gambar 2.9

A custom component!

Gambar 2.9 Keluaran vue komponen

2. Vue router

Vue router merupakan plugin dari vue untuk membuat routing dari sisi pengguna. Hal ini dapat dimanfaatkan untuk membuat SPA. Pada Gambar 2.10 merupakan perintah yang dapat diketikkan pada *terminal* untuk memasang *vue router*.



```
npm install vue-router
```

Gambar 2.10 Cara memasang vue router

Setelah vue router terpasang, selanjutnya vue router harus di import dan dipanggil pada file javascript utama dimana terdapat instant dari vue seperti pada Gambar 2.11.

```
import Vue from 'vue'
import VueRouter from 'vue-router'

Vue.use(VueRouter)
```

Gambar 2.11 Cara menggunakan vue router

Pada Gambar 2.12 merupakan contoh membuat rute dari vue router. Pada Gambar 2.12 saat pengguna mengakses rute pengajuan-judul maka vue komponen dengan nama pengajuan_proposal_index.vue akan dipanggil.

```
import [Pengajuan_Proposal_Index] from
'./components/Pengajuan_Proposal_Index.vue';

const routes = [
  {
    path: 'pengajuan-judul',
    component: Pengajuan-Proposal,
  },
]

export default routes;
```

Gambar 2.12 Contoh penggunaan vue router

2.5.4.7 JSON Web Token (JWT)

JWT merupakan sebuah token dengan struktur data JSON yang bersifat *compact* dan *self-contained* yang digunakan untuk pengiriman data secara aman. JWT bersifat *compact* yang berarti JWT memiliki ukuran yang kecil sehingga dapat dikirim melalui *Uniform Resource Locator* (URL), *Hypertext Transfer Protocol* (HTTP) *header* atau di dalam parameter POST dengan cepat. *Self-contained* berarti bahwa JWT dapat menyimpan segala informasi yang diperlukan mengenai *user* (Auth0, 2016). Pada Gambar 2.13 merupakan contoh dari JWT.

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRGR9IiwiaWF0IjoiYXNjaWV3LnRydWV9.4pcPyMD09o1PSyXnrXCjTwXyr4BsezdI1AVTmud2fU4
```

Gambar 2.13 Contoh JWT



Struktur dari JWT terdiri dari 3 bagian yang dipisahkan dengan tanda titik yaitu (Auth0, 2016) :

1. *Header*

Header merupakan bagian pertama dari token yang terdiri dari 2 bagian, tipe dari *token* yaitu JWT dan algoritme *hash* yang digunakan seperti HMAC SHA256 atau RSA.

2. *Payload*

Payload merupakan bagian kedua dari *token* yang menyimpan informasi tertentu seperti informasi dasar dari pengguna.

3. *Signature*

Signature merupakan bagian terakhir dari *token* yang terdiri dari *header* yang sudah dikodekan, *payload* yang sudah dikodekan dan sebuah kata kunci rahasia.

Konsep dari cara kerja dari JWT adalah sebagai berikut (Auth0, 2016) :

1. Pengguna melakukan *login* ke dalam sistem.
2. Jika data login berhasil, JWT akan dikirimkan kembali ke pengguna dan harus disimpan pada *web browser* pengguna pada *cookies* atau *localStorage*.
3. Saat pengguna ingin mengakses URL tertentu atau sumber daya tertentu, JWT akan dikirim dan dicek oleh server. Apabila JWT valid, maka pengguna dapat melanjutkan perintahnya.

2.5.5 Deployment

Pada tahapan dilakukan pengujian program yang telah dikembangkan

2.5.5.1 Pengujian Perangkat Lunak

Pengujian perangkat lunak digunakan untuk menguji apakah program berfungsi dengan semestinya dan untuk mengetahui kesalahan-kesalahan program sebelum program tersebut siap digunakan (Sommerville, 2011).

Sommerville (2011) mendetailkan bahwa tujuan dari pengujian perangkat lunak sebagai berikut :

1. Untuk menguji coba dihadapan pihak pengembang dan pihak *customer* apakah program tersebut memenuhi kebutuhan dari sisi pengembang dan sisi *customer*.
2. Untuk mencari situasi dimana program berjalan tidak berjalan dengan benar, tidak sesuai dengan keinginan ataupun tidak memenuhi spesifikasi yang telah ditentukan. Kegagalan program bisa terjadi dalam bentuk program *crash*, interaksi yang tidak diinginkan dengan sistem atau pengguna, kesalahan perhitungan dan hilangnya data.

Sehingga dapat disimpulkan bahwa pengujian perangkat lunak penting dilakukan agar program yang dihasilkan memenuhi kebutuhan serta memastikan



program berjalan dengan semestinya. Pada penelitian ini pengujian yang dilakukan adalah *black-box testing* dan *user acceptance testing*.

1. Black-box Testing

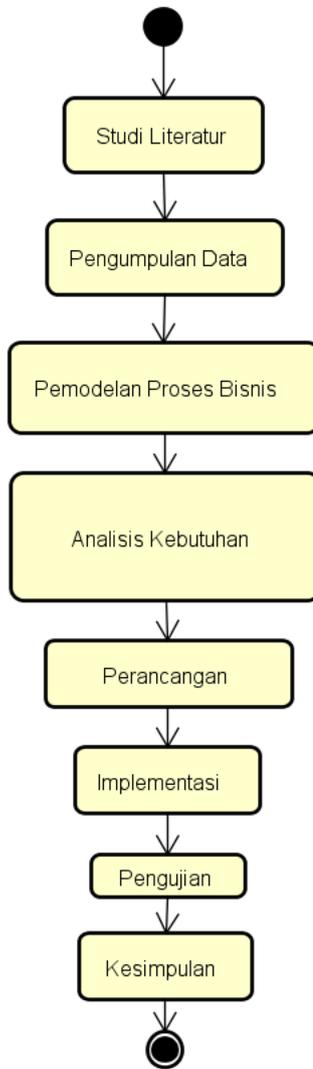
Black-box testing atau juga disebut dengan *behavioral testing*, merupakan pengujian yang fokus pada kebutuhan fungsional perangkat lunak. *Black-box testing* memungkinkan untuk memasukkan kondisi tertentu untuk menguji kebutuhan fungsional perangkat lunak dapat terpenuhi atau tidak (Pressman, 2010).

2. Compatibility Testing

Software compatibility testing merupakan pengujian yang ditujukan untuk memastikan bahwa komponen dari sistem yang dikembangkan bekerja pada lintas lingkungan (Yoon, et al., 2008). Pada penelitian pengujian compatibility yang dilakukan pada *web browser* guna mengetahui apakah sistem yang dikembangkan dapat bekerja pada berbagai macam *web browser*.

BAB 3 METODOLOGI

Penelitian ini merupakan penelitian yang bertipe implementatif, dimana penelitian ini menghasilkan *software*. Pendekatan implementatif yang digunakan dalam penelitian ini adalah pendekatan pengembangan. Pendekatan pengembangan merupakan pendekatan penelitian implementatif untuk membuat *software* jadi melalui proses yang utuh mulai dari tahap analisis perancangan hingga pengujian. Pada bab ini peneliti menjelaskan langkah-langkah yang akan dilakukan pada penelitian. Langkah-langkah yang dilakukan dalam penelitian ini adalah studi pustaka, pengumpulan data, pemodelan proses bisnis, analisis kebutuhan, perancangan, implementasi, pengujian, evaluasi sistem, dan pengambilan kesimpulan. Gambar 3.1 menjelaskan alur penelitian yang akan digunakan dalam penelitian ini



Gambar 3.1 Alur penelitian



3.1 Studi Pustaka

Studi pustaka merupakan tahap mencari, memilih serta mempelajari dan menyusun teori dasar dan referensi yang mendukung penelitian yang akan dilakukan. Tahap yang dilakukan dalam studi pustaka ini diantaranya melakukan pengumpulan informasi tentang cara pengembangan sistem informasi berdasarkan proses bisnis.

3.2 Pengumpulan Data

Tahap pengumpulan data pada penelitian ini memiliki tujuan mencari dan mengumpulkan data yang diperlukan. Pada tahap ini digunakan untuk mendapatkan data mengenai proses bisnis manajemen tesis. Tahap pengumpulan data ini dilakukan dengan cara wawancara dan studi dokumen pada Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB. Wawancara dilakukan langsung dengan akademik pada Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB.

3.3 Pemodelan Proses Bisnis

Pemodelan proses bisnis adalah tahap memodelkan proses bisnis manajemen tesis saat ini dan memodelkan proses bisnis usulan pada Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB. Pemodelan proses bisnis dilakukan berdasarkan data hasil yang di dapat dari proses wawancara dan studi dokumen sebelumnya.

3.4 Analisis Kebutuhan

Analisis kebutuhan merupakan tahap memahami dan menyusun persyaratan yang dibutuhkan sistem. Analisis kebutuhan dilakukan berdasarkan pemodelan proses bisnis usulan yang telah dibuat. Setelah tahap ini selesai maka dilanjutkan dengan perancangan.

3.5 Perancangan

Perancangan merupakan tahap memodelkan hasil analisis kebutuhan ke dalam diagram. Hasil dari tahap ini digunakan sebagai dasar sistem yang akan di buat. Pada perancangan ini dilakukan perancangan arsitektur, pemodelan sequence diagram, pemodelan class diagram, physical data model, perancangan komponen dan perancangan antarmuka.

3.6 Implementasi

Implementasi adalah tahap merubah hasil perancangan ke dalam kode program. Implementasi dibagi menjadi 2 bagian yaitu : membuat *web service* dan pembuatan sistem informasi manajemen tesis. *Web service* yang akan dibuat menggunakan bahasa pemrograman php oop dengan framework codeigniter. Untuk pembuatan sistem informasi manajemen tesis dibuat dengan menerapkan



konsep *single page application*. Untuk menerapkan konsep *single page application* pada sistem informasi manajemen tesis digunakan *framework* Vue.js.

3.7 Pengujian

Pengujian dilakukan untuk mengetahui apakah sistem informasi yang dibangun memiliki kesalahan atau error. Pengujian yang digunakan adalah pengujian Black-Box dan pengujian *compatibility*. Pengujian Black-box dimaksudkan untuk mengecek fungsionalitas sistem dan menghasilkan output yang sesuai harapan. Pengujian *compatibility* digunakan untuk mengetahui apakah sistem yang telah dikembangkan dapat digunakan dengan baik dalam *web browser* yang berbeda-beda.

3.8 Kesimpulan dan Saran

Pembuatan kesimpulan merupakan tahap akhir dalam penelitian ini. Pada bagian ini terdapat jawaban dari masalah yang telah dirumuskan serta menjelaskan saran untuk penelitian berikutnya.



BAB 4 ANALISIS KEBUTUHAN

Pada bab ini akan dibahas mengenai analisis kebutuhan dari sistem yang akan dikembangkan. Proses yang ada pada bab ini yaitu analisis proses bisnis, untuk memahami proses bisnis yang ada saat ini dan proses bisnis usulan yang digunakan sebagai dasar pengembangan sistem, identifikasi aktor untuk merumuskan pihak-pihak yang nantinya terlibat dalam sistem dan perumusan kebutuhan fungsional dan non fungsional.

4.1 Gambaran Umum Sistem

Sistem Informasi Manajemen Tesis ini merupakan sistem informasi berbasis web yang dibuat untuk membantu proses manajemen tesis mulai pengajuan judul dan dosen pembimbing tesis hingga ujian tesis yang selama ini masih manual. Sistem ini menyediakan formulir digital yang dapat membantu tiap pihak yang terlibat untuk melakukan proses manajemen tesis sehingga proses manajemen tesis dapat dilakukan tanpa harus menemui pihak tertentu secara langsung. Sistem ini juga dapat mempermudah pihak yang terlibat untuk mengakses informasi tertentu mengenai manajemen tesis seperti melihat jumlah bimbingan tiap dosen, melihat judul tesis dari mahasiswa yang ada dan melihat perkembangan proses pengerjaan tesis mahasiswa. Sistem ini juga menyediakan fungsi manajemen yang dapat membantu pengambilan keputusan seperti pada saat memilih dosen pembimbing bagi mahasiswa, kaprodi dapat memilih dosen pembimbing berdasarkan data jumlah bimbingan tiap dosen saat ini.

4.2 Analisis Proses Bisnis

Berdasarkan hasil wawancara dan studi dokumen, dapat dihasilkan beberapa gambaran mengenai proses bisnis alur penyelenggaraan tesis yang ada pada Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB.

4.2.1 Identifikasi Proses Bisnis Saat ini (as-is)

Berdasarkan hasil studi dokumen dan wawancara dengan petugas akademik Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB dapat digambarkan proses bisnis saat ini.

4.2.1.1 Proses Bisnis Pengajuan Judul dan Pembimbing Tesis

Pengajuan judul tesis dan dosen pembimbing merupakan proses yang dilakukan mahasiswa untuk mengajukan judul penelitian tesisnya dan menentukan dosen pembimbingnya. Proses pengajuan judul tesis dan dosen pembimbing yang ada pada Program Studi Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB saat ini dapat dilihat pada Gambar 4.1. Proses pengajuan dimulai dengan mahasiswa mengambil formulir pengajuan judul tesis dan usulan dosen pembimbing di akademik. Pada formulir tersebut, mahasiswa mengisi judul dan keminatan penelitian tesis, linkage penelitian, 5 nama dosen sebagai usulan menjadi dosen pembimbing. Setelah mengisi formulir tersebut, mahasiswa



menyerahkannya ke ketua program studi melalui sekretaris program. Sekretaris program selanjutnya menyerahkan formulir tersebut ke ketua program studi. Ketua program studi selanjutnya akan memeriksa dan menyetujui formulir pengajuan judul tesis serta menentukan 2 dosen pembimbing bagi mahasiswa.

Akademik yang telah menerima formulir dari ketua program studi akan merekap judul tesis dan dosen pembimbing sesuai dengan data yang ada pada formulir tersebut. Setelah merekap judul tesis dan dosen pembimbing, akademik mengumumkan judul dan dosen pembimbing ke pada mahasiswa. Mahasiswa selanjutnya dapat mengambil dan mengisi formulir persetujuan dosen pembimbing di akademik. Formulir persetujuan tersebut selanjutnya harus ditandatangani oleh dosen yang telah ditunjuk menjadi dosen pembimbing. Setelah mendapat tanda tangan dosen pembimbing, mahasiswa menemui ketua program studi untuk meminta tanda tangan pada formulir tersebut. Selanjutnya mahasiswa dapat memperbanyak lembar persetujuan tersebut dan diserahkan ke akademik.



4.2.1.2 Proses Bisnis Ujian Proposal

Proses bisnis ujian proposal merupakan serangkaian tahapan yang harus dilakukan mahasiswa untuk dapat melakukan ujian proposal. Pada Gambar 4.2 menjelaskan tahapan ujian proposal tesis yang ada pada Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB. Untuk dapat melakukan ujian proposal, mahasiswa membuat lembar persetujuan ujian proposal terlebih dahulu. Lembar persetujuan tersebut selanjutnya harus ditandatangani oleh dosen pembimbing dan ketua program studi sebagai bukti persetujuan bahwa mahasiswa tersebut dapat mengajukan ujian proposal. Setelah lembar persetujuan mendapat tanda tangan dosen pembimbing dan ketua program studi, mahasiswa dapat mendaftar ujian proposal ke akademik.

Pendaftaran ujian proposal perlu mencantumkan waktu pelaksanaan ujian yang telah disepakati oleh 2 dosen pembimbing. Mahasiswa juga perlu membawa fotocopy sertifikat tes potensi akademik (TPA) dan *Test Of English as Foreign Language* (TOEFL), bukti pembayaran sumbangan pembinaan pendidikan (SPP) semester terakhir, asli lembar persetujuan dan asli lembar mengikuti seminar saat melakukan pendaftaran ujian proposal. Akademik selanjutnya mengecek waktu pelaksanaan ujian proposal yang diajukan mahasiswa (sekurang-kurangnya 1 minggu sebelum pelaksanaan ujian sejak pendaftaran ujian proposal) dan mengecek berkas prasyarat pengajuan ujian proposal. Bila syarat tersebut terpenuhi, akademik selanjutnya merekap data pengajuan ujian proposal dan membuat formulir pengajuan ujian proposal untuk diserahkan ke mahasiswa (Formulir UP1-UP5).

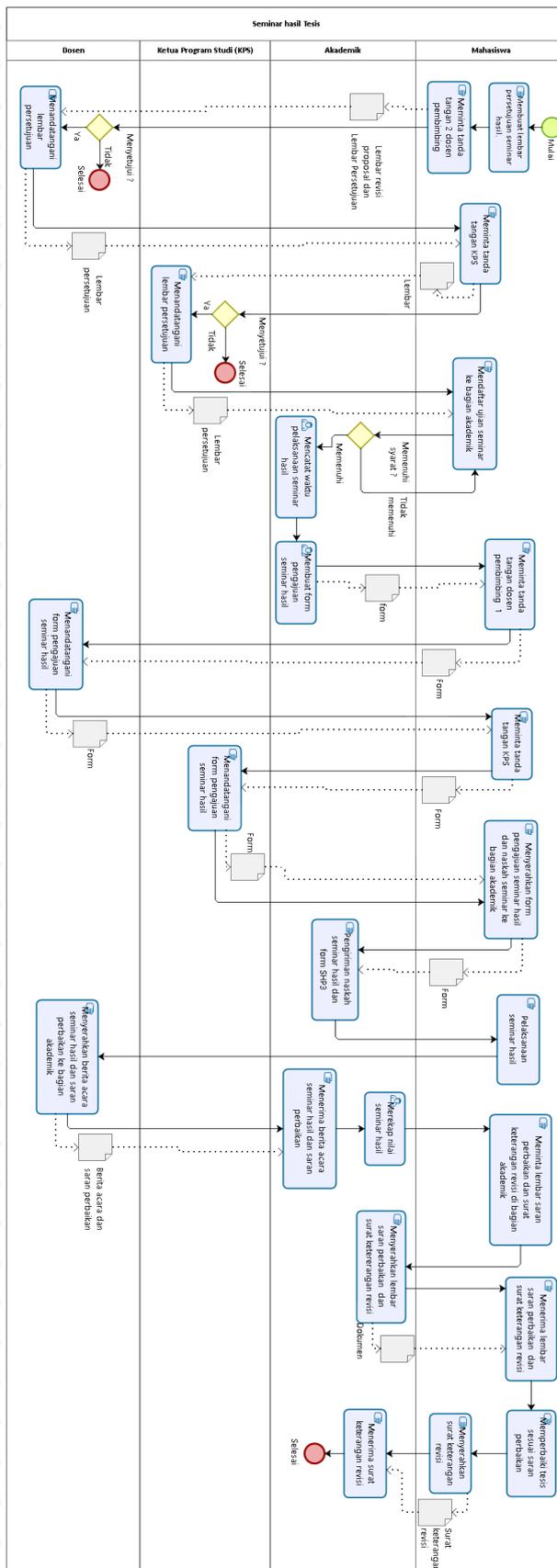
Mahasiswa selanjutnya meminta tanda tangan ketua dosen pembimbing pada formulir UP1 sebagai tanda persetujuan dilakukan ujian proposal. Selanjutnya mahasiswa menemui kaprodi untuk meminta tanda tangan pada formulir UP2 sebagai tanda persetujuan dilakukan ujian proposal dan juga menyerahkan formulir UP3 untuk penentuan dosen penguji ujian proposal. Mahasiswa selanjutnya menemui dosen pembimbing dan dosen penguji untuk menandatangani formulir UP3. Bila formulir UP3 sudah mendapat tanda tangan dosen pembimbing dan dosen penguji, mahasiswa kembali menemui kaprodi untuk meminta tanda tangan pada formulir UP3, UP4 dan UP5. Selanjutnya mahasiswa menuju ke akademik dengan menyerahkan Formulir UP1 sampai UP5 yang sudah ditandatangani dan naskah tesis. Akademik selanjutnya mengirim naskah tesis mahasiswa sebelum pelaksanaan ujian proposal. Pada saat ujian proposal sudah selesai, selanjutnya dosen pembimbing dan dosen penguji menyerahkan berita acara ujian proposal ke akademik untuk direkap. Mahasiswa dapat mengambil lembar saran perbaikan dan surat keterangan revisi dari akademik untuk selanjutnya dilakukan perbaikan tesis.



4.2.1.3 Proses Bisnis Seminar Hasil

Proses pengajuan ujian seminar hasil pembimbing merupakan proses yang dilakukan mahasiswa untuk dapat melaksanakan ujian seminar hasil. Pada Gambar 4.3 menjelaskan tahapan seminar hasil yang ada pada Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB. Untuk bisa mengajukan seminar hasil, mahasiswa harus terlebih dahulu membuat jurnal ilmiah. Apabila sudah membuat jurnal ilmiah, mahasiswa dapat membuat lembar persetujuan pengajuan seminar hasil. Lembar persetujuan tersebut selanjutnya harus ditandatangani oleh ketua program studi dan 2 dosen pembimbing. Saat meminta tanda tangan, mahasiswa juga harus menunjukkan lembar revisi proposal yang sudah ditanda tangani oleh dosen pembimbing dan dosen penguji. Bila lembar persetujuan sudah mendapat tanda tangan selanjutnya mahasiswa bisa mendaftar seminar hasil di akademik. Pendaftaran seminar hasil perlu mencantumkan hari, tanggal, dan jam pelaksanaan sesuai kesepakatan dengan dosen pembimbing. Pendaftaran seminar hasil juga harus dilakukan 1 minggu sebelum hari pelaksanaan ujian. Selanjutnya akademik akan memberikan formulir pengajuan seminar hasil (Formulir SHP1 – SHP4) kepada mahasiswa.

Formulir SHP1 – SHP4 tersebut harus ditandatangani oleh dosen pembimbing dan ketua program studi sebagai bukti persetujuan dilakukan seminar hasil pada waktu yang telah disepakati. Mahasiswa selanjutnya menyerahkan formulir tersebut ke akademik 4 hari sebelum waktu pelaksanaan seminar hasil. Akademik yang sudah menerima formulir tersebut selanjutnya dapat mengirimkan naskah seminar hasil dan formulir SHP3 melalui bagian ekspedisi. Setelah pelaksanaan seminar hasil, dosen pembimbing menyerahkan berita acara seminar hasil lengkap dengan nilai dan saran perbaikan ke akademik untuk direkap. Lembar saran perbaikan tersebut selanjutnya dapat diberikan kepada mahasiswa lengkap dengan surat keterangan revisi seminar hasil di akademik. Setelah menerima formulir tersebut, mahasiswa diberikan waktu 1 bulan untuk menyelesaikan perbaikan naskah seminar hasil. Apabila perbaikan sudah selesai, mahasiswa dapat menemui dosen pembimbing dan ketua program studi untuk meminta tanda tangan pada surat keterangan revisi seminar hasil. Surat keterangan revisi seminar hasil tersebut selanjutnya dapat diserahkan ke akademik untuk direkap.



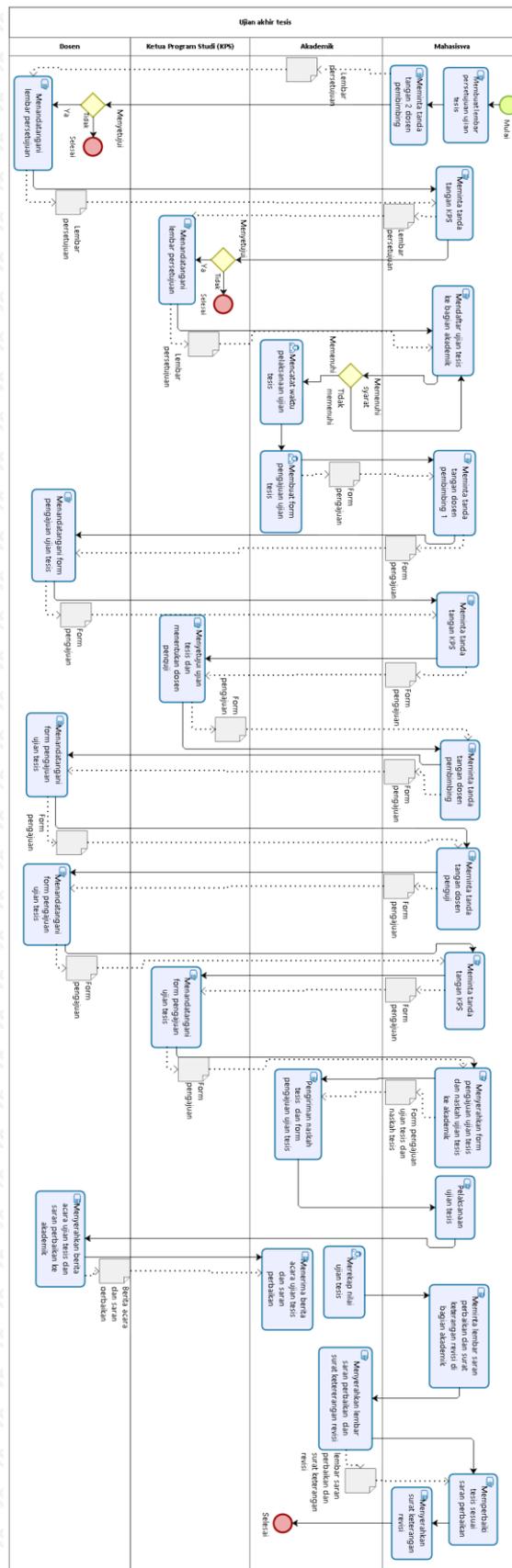
Gambar 4.3 Proses bisnis ujian seminar hasil saat ini



4.2.1.4 Proses Bisnis Ujian Tesis

Proses pengajuan ujian tesis merupakan proses yang dilakukan mahasiswa untuk dapat melaksanakan ujian tesis. Pada Gambar 4.4 menjelaskan tahapan ujian tesis yang ada pada Magister Administrasi Bisnis Fakultas Ilmu Administrasi UB. Untuk bisa mengajukan ujian tesis, mahasiswa sebelumnya harus membuat formulir persetujuan ujian tesis dan ditandatangani oleh dosen pembimbing dan ketua program studi. Bila dosen pembimbing dan ketua program studi sudah menyetujui, mahasiswa dapat mendaftar ujian tesis ke akademik. Pendaftaran ujian tesis dilakukan mahasiswa dengan mencantumkan tanggal hari, tanggal, dan jam pelaksanaan ujian sesuai kesepakatan dengan dosen pembimbing. Akademik selanjutnya merekap data pengajuan ujian tersebut dan membuat formulir pengajuan ujian tesis (UP-1 sampai UP-5).

Formulir UP-1 sampai UP-5 tersebut harus ditandatangani oleh dosen pembimbing, dosen penguji dan ketua program studi sebagai bukti persetujuan dilakukan ujian tesis pada waktu yang telah disepakati. Mahasiswa selanjutnya menyerahkan formulir tersebut ke akademik 4 hari sebelum waktu pelaksanaan ujian tesis. Akademik yang sudah menerima formulir tersebut selanjutnya dapat mengirimkan naskah seminar hasil dan formulir SHP3 melalui bagian ekspedisi. Setelah pelaksanaan ujian tesis, dosen pembimbing menyerahkan berita acara ujian tesis lengkap dengan nilai dan saran perbaikan ke akademik untuk direkap. Lembar saran perbaikan tersebut selanjutnya dapat diberikan kepada mahasiswa lengkap dengan surat keterangan revisi ujian tesis di akademik. Setelah menerima formulir tersebut, mahasiswa diberikan waktu 1 bulan untuk menyelesaikan perbaikan naskah tesis sesuai saran dari dosen pembimbing dan dosen penguji. Apabila perbaikan sudah selesai, mahasiswa dapat menemui dosen pembimbing, dosen penguji dan ketua program studi untuk meminta tanda tangan pada surat keterangan revisi ujian tesis. Surat keterangan revisi ujian tesis tersebut selanjutnya dapat diserahkan ke akademik untuk direkap.



Gambar 4.4 Proses bisnis ujian tesis saat ini



4.2.2 Identifikasi Proses Bisnis Usulan (to-be)

Berikut merupakan proses bisnis usulan (to-be) yang ditawarkan dengan adanya sistem informasi manajemen tesis.

4.2.2.1 Proses Bisnis Pengajuan Judul dan Pembimbing Tesis

Proses bisnis usulan pengajuan judul tesis dan dosen pembimbing dapat dilihat pada Gambar 4.5. Dengan adanya sistem informasi manajemen tesis, mahasiswa tidak perlu mengambil formulir pengajuan judul tesis dan dosen pembimbing pada akademik. Mahasiswa dapat mengisi formulir peganjuan tersebut dalam sistem. Saat mengisi formulir pengajuan tersebut, mahasiswa dapat melihat daftar dosen yang tersedia beserta bidang minat dan jumlah bimbinganya saat ini. Dengan adanya informasi mengenai dosen yang akan dipilih, diharapkan dapat membantu mahasiswa untuk memilih calon dosen pembimbing yang sesuai. Setelah formulir pengajuan tersebut dikirim, akademik akan mendapat *email* pemberitahuan jika terdapat pengajuan judul dan pembimbing tesis baru.

Selanjutnya akademik mengecek apakah mahasiswa yang mengajukan judul dan dosen pembimbing tersebut telah memenuhi syarat atau belum. Jika mahasiswa tersebut memenuhi syarat, sistem akan mengirim *email* notifikasi pada ketua program studi untuk perlu persetujuan selanjutnya. Ketua program studi akan mengecek judul, bidang minat serta pilihan dosen pembimbing yang diajukan mahasiswa. Ketua program studi dapat melihat data dosen beserta jumlah bimbinganya sebagai pertimbangan dalam memilih dosen pembimbing untuk mahasiswa. Bila ketua program studi menyetujui, sistem akan mengirimkan *email* kepada dosen yang telah ditunjuk menjadi dosen pembimbing. Dosen yang telah ditunjuk selanjutnya dapat menyetujui menjadi dosen pembimbing. Setelah mendapat persetujuan dosen pembimbing, pengajuan tersebut akan dicek kembali oleh ketua program studi. Bila sudah mendapat persetujuan ketua program studi, mahasiswa akan mendapat *email* notifikasi bahwa pengajuan judul dan dosen pembimbingnya sudah selesai.

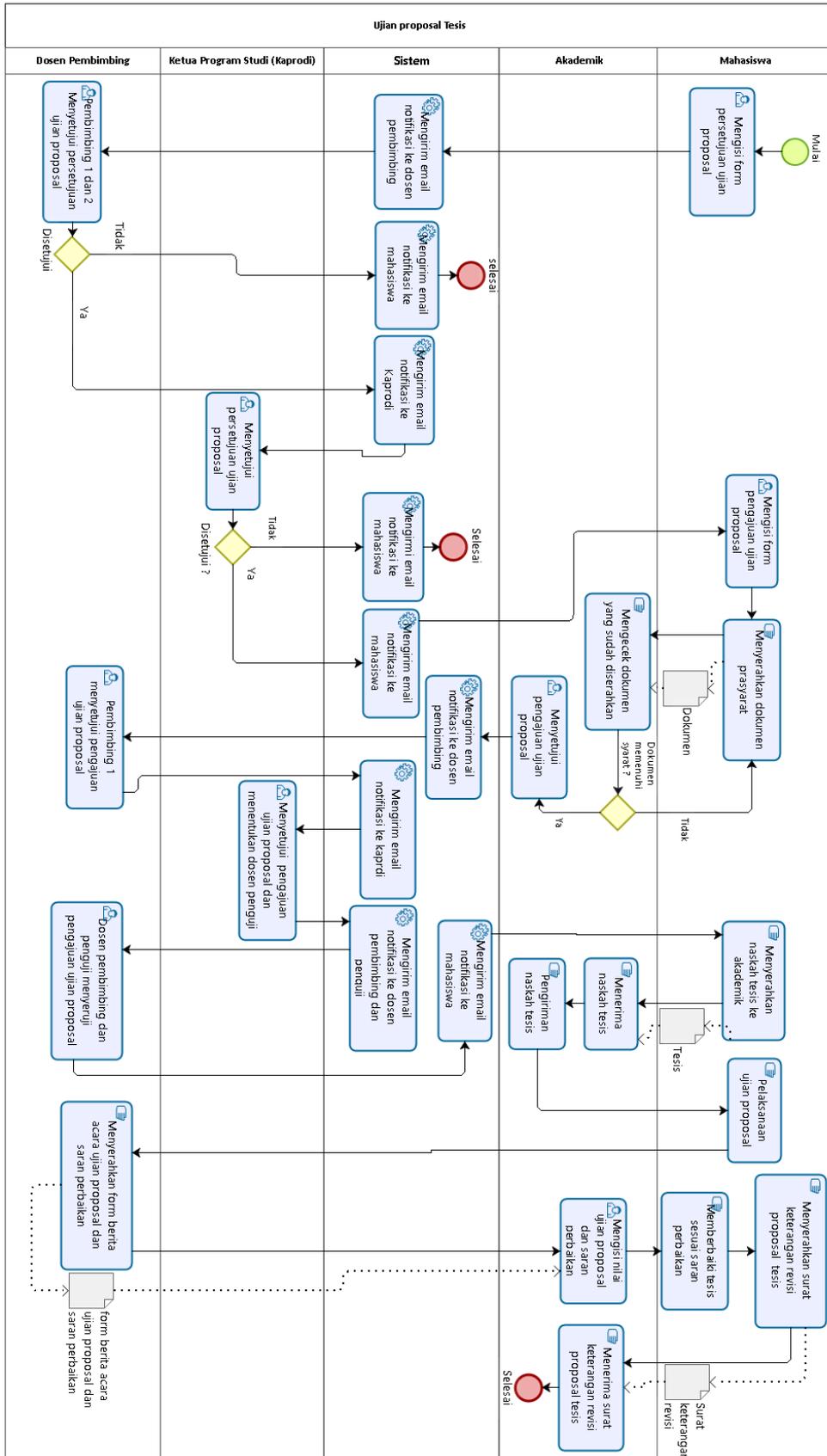


4.2.2.2 Proses Bisnis Ujian Proposal

Proses bisnis usulan ujian proposal dapat dilihat pada Gambar 4.6. Dengan adanya sistem informasi manajemen tesis, lembar atau formulir yang sebelumnya digunakan untuk meminta persetujuan atau mengajukan ujian tesis akan digantikan dengan formulir digital. Dengan adanya formulir digital ini, pihak yang terlibat dalam proses permintaan persetujuan atau pengajuan ujian tesis tidak perlu bertemu secara langsung. Formulir digital ini juga mempermudah pihak yang terlibat untuk melihat kembali data persetujuan atau pengajuan ujian proposal.

Proses permintaan persetujuan ujian proposal dimulai dengan mahasiswa mengisi formulir persetujuan ujian proposal yang ada pada sistem. Formulir persetujuan yang telah diisi mahasiswa akan disimpan dan menunggu persetujuan dari dosen pembimbing. Dosen pembimbing dapat mengecek dan menyetujui permintaan persetujuan ujian yang telah diajukan mahasiswa. Bila sudah mendapat persetujuan dari dosen pembimbing, selanjutnya sistem akan memberi pemberitahuan ke ketua program studi untuk menyetujui persetujuan ujian proposal. Setelah mendapat persetujuan dari ketua program studi, sistem akan mengubah status persetujuan tersebut dan mahasiswa bisa melanjutkan dengan mengisi formulir pengajuan ujian proposal yang ada pada sistem. Pada saat mengisi formulir pengajuan ujian proposal, mahasiswa diminta mengisi waktu dilakukan ujian proposal sesuai kesepakatan dengan dosen pembimbing. Mahasiswa juga dapat mengecek waktu yang tersedia untuk dilakukan ujian proposal tesis. Setelah mensubmit, sistem akan mengecek apakah waktu ujian proposal yang diajukan mahasiswa. Pengajuan ujian proposal hanya akan disimpan bila waktu yang diajukan tersedia dan tidak kurang dari 1 minggu dari waktu pengisian formulir pengajuan. Bila memenuhi syarat, mahasiswa selanjutnya menyerahkan fotocopy sertifikat tes potensi akademik (TPA), fotocopy *Test Of English as Foreign Language* (TOEFL), bukti pembayaran sumbangan pembinaan pendidikan (SPP) semester terakhir, dan lembar mengikuti seminar. Akademik akan mengecek berkas tersebut dan menyetujui pengajuan bila memenuhi syarat. Setelah mendapat persetujuan dari akademik, selanjutnya pengajuan perlu persetujuan sistem akan memberikan pemberitahuan pada dosen pembimbing, dosen penguji dan ketua program studi untuk menyetujui pengajuan ujian proposal tesis.

Setelah mendapat pemberitahuan bahwa pengajuan ujian proposal telah selesai, mahasiswa dapat menyerahkan naskah proposal tesis pada akademik. Setelah pelaksanaan ujian tesis, akademik akan merekap data nilai ujian tesis beserta saran perbaikan tesis berdasarkan formulir yang diserahkan oleh dosen pembimbing. Selanjutnya mahasiswa dapat melihat nilai dan saran perbaikan ujian proposal pada sistem dan mulai melakukan perbaikan tesis. Mahasiswa yang telah melakukan perbaikan tesisnya dapat menyerahkan surat keterangan revisi yang telah ditandatangani oleh dosen pembimbing dan dosen penguji ke akademik. Akademik yang sudah mendapat surat keterangan revisi dapat memberikan persetujuan bahwa proses ujian proposal telah selesai.



Gambar 4.6 Proses bisnis ujian proposal usulan



4.2.2.3 Proses Bisnis Seminar Hasil

Proses bisnis seminar hasil dapat dilihat pada Gambar 4.7. Dengan adanya sistem informasi manajemen tesis, lembar atau formulir yang sebelumnya digunakan untuk meminta persetujuan atau mengajukan seminar hasil akan digantikan dengan formulir digital. Dengan adanya formulir digital ini, pihak yang terlibat dalam proses permintaan persetujuan atau pengajuan seminar tidak perlu bertemu secara langsung. Formulir digital ini juga mempermudah pihak yang terlibat untuk melihat kembali data persetujuan atau pengajuan seminar hasil.

Proses permintaan persetujuan seminar hasil dimulai dengan mahasiswa mengisi formulir persetujuan seminar hasil yang ada pada sistem. Formulir persetujuan yang telah diisi mahasiswa akan disimpan dan menunggu persetujuan dari dosen pembimbing. Dosen pembimbing dapat mengecek dan menyetujui permintaan persetujuan ujian yang telah diajukan mahasiswa. Bila sudah mendapat persetujuan dari dosen pembimbing, selanjutnya sistem akan memberi pemberitahuan ke ketua program studi untuk menyetujui persetujuan ujian proposal. Setelah mendapat persetujuan dari ketua program studi, sistem akan mengubah status persetujuan tersebut dan mahasiswa bisa melanjutkan dengan mengisi formulir pengajuan seminar hasil yang ada pada sistem. Pada saat mengisi formulir pengajuan seminar hasil, mahasiswa diminta mengisi waktu dilakukan seminar hasil sesuai kesepakatan dengan dosen pembimbing. Mahasiswa juga dapat mengecek waktu yang tersedia untuk dilakukan seminar hasil. Setelah mensubmit, sistem akan mengecek apakah waktu seminar hasil yang diajukan mahasiswa. Pengajuan seminar hasil hanya akan disimpan bila waktu yang diajukan tersedia dan tidak kurang dari 1 minggu dari waktu pengisian formulir pengajuan. Bila memenuhi syarat, sistem akan menyimpan data pengajuan dan sistem akan memberikan pemberitahuan pada dosen pembimbing dan ketua program studi untuk menyetujui pengajuan seminar hasil.

Setelah mendapat pemberitahuan bahwa pengajuan seminar hasil telah selesai, mahasiswa dapat menyerahkan naskah tesis pada akademik. Setelah pelaksanaan seminar hasil, akademik akan merekap data nilai seminar hasil beserta saran perbaikan tesis berdasarkan formulir yang diserahkan oleh dosen pembimbing. Selanjutnya mahasiswa dapat melihat nilai dan saran perbaikan seminar hasil pada sistem dan mulai melakukan perbaikan tesis. Mahasiswa yang telah melakukan perbaikan tesisnya dapat menyerahkan surat keterangan revisi yang telah ditandatangani oleh dosen pembimbing ke akademik. Akademik yang sudah mendapat surat keternagan revisi dapat memberikan persetujuan bahwa proses seminar hasil telah selesai.

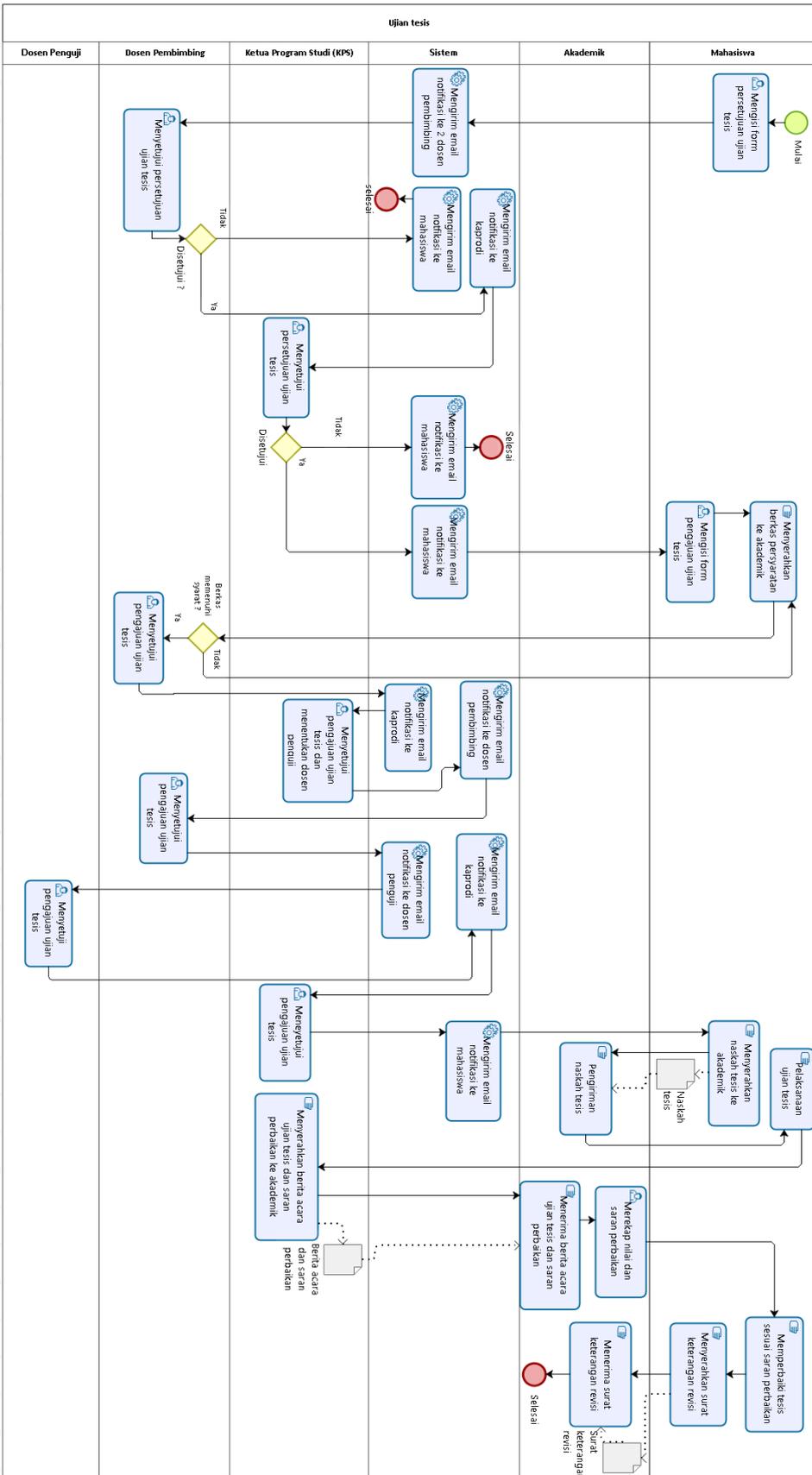


4.2.2.4 Proses Bisnis Ujian Tesis

Proses bisnis ujian tesis dapat dilihat pada Gambar 4.7. Dengan adanya sistem informasi manajemen tesis, lembar atau formulir yang sebelumnya digunakan untuk meminta persetujuan atau mengajukan ujian tesis akan digantikan dengan formulir digital. Dengan adanya formulir digital ini, pihak yang terlibat dalam proses permintaan persetujuan atau pengajuan ujian tesis tidak perlu bertemu secara langsung. Formulir digital ini juga mempermudah pihak yang terlibat untuk melihat kembali data persetujuan atau pengajuan ujian tesis.

Proses permintaan persetujuan ujian tesis dimulai dengan mahasiswa mengisi formulir persetujuan ujian tesis yang ada pada sistem. Formulir persetujuan yang telah diisi mahasiswa akan disimpan dan menunggu persetujuan dari dosen pembimbing. Dosen pembimbing dapat mengecek dan menyetujui permintaan persetujuan ujian yang telah diajukan mahasiswa. Bila sudah mendapat persetujuan dari dosen pembimbing, selanjutnya sistem akan memberi pemberitahuan ke ketua program studi untuk menyetujui persetujuan ujian proposal. Setelah mendapat persetujuan dari ketua program studi, sistem akan mengubah status persetujuan tersebut dan mahasiswa bisa melanjutkan dengan mengisi formulir pengajuan ujian tesis yang ada pada sistem. Pada saat mengisi formulir pengajuan ujian tesis, mahasiswa diminta mengisi waktu dilakukan ujian tesis sesuai kesepakatan dengan dosen pembimbing. Mahasiswa juga dapat mengecek waktu yang tersedia untuk dilakukan ujian tesis. Setelah mensubmit, sistem akan mengecek apakah waktu ujian tesis yang diajukan mahasiswa. Pengajuan ujian tesis hanya akan disimpan bila waktu yang diajukan tersedia dan tidak kurang dari 1 minggu dari waktu pengisian formulir pengajuan. Bila memenuhi syarat, sistem akan menyimpan data pengajuan dan sistem akan memberikan pemberitahuan pada dosen pembimbing, dosen penguji dan ketua program studi untuk menyetujui pengajuan ujian tesis.

Setelah mendapat pemberitahuan bahwa pengajuan ujian tesis telah selesai, mahasiswa dapat menyerahkan naskah tesis pada akademik. Setelah pelaksanaan ujian tesis, akademik akan merekap data nilai ujian tesis beserta saran perbaikan tesis berdasarkan formulir yang diserahkan oleh dosen pembimbing. Selanjutnya mahasiswa dapat melihat nilai dan saran perbaikan ujian tesis pada sistem dan mulai melakukan perbaikan tesis. Mahasiswa yang telah melakukan perbaikan tesisnya dapat menyerahkan surat keterangan revisi yang telah ditandatangani oleh dosen pembimbing dan dosen penguji ke akademik. Akademik yang sudah mendapat surat keterangan revisi dapat memberikan persetujuan bahwa proses ujian tesis telah selesai.



Gambar 4.8 Proses bisnis ujian tesis usulan



4.3 Pemetaan Proses Bisnis

Proses bisnis usulan tersebut selanjutnya dilakukan pengkodean seperti pada Tabel 4.1

Tabel 4.1 Daftar proses bisnis

Kode Proses Bisnis	Nama Proses Bisnis
PB01	Proses bisnis pengajuan judul dan dosen pembimbing tesis
PB02	Proses bisnis ujian proposal
PB03	Proses bisnis ujian seminar hasil
PB04	Proses bisnis ujian tesis

Berdasarkan proses bisnis usulan tersebut maka dapat dipetakan aktivitas proses bisnis. Pemetaan proses bisnis ini memiliki 4 atribut yaitu: kode aktivitas, kode proses bisnis, nama dan keterangan. Hasil pemetaan proses bisnis dapat dilihat pada Tabel 4.2

Tabel 4.2 Pemetaan proses bisnis

Kode Aktivitas	Kode Proses Bisnis	Nama	Keterangan
A01	PB01	Mengisi formulir pengajuan judul tesis dan dosen pembimbing	Mahasiswa mengisi formulir pengajuan judul tesis dan dosen pembimbing melalui sistem.
A02	PB01	Menyetujui pengajuan judul tesis dan dosen pembimbing	Akademik dan ketua program studi dapat menyetujui pengajuan judul tesis dan dosen pembimbing bila memenuhi syarat.
A03	PB01	Menyetujui dan Menentukan dosen pembimbing	Ketua program studi menentukan dosen pembimbing bagi mahasiswa yang mengajukan dosen pembimbing.
A04	PB01	Menyetujui menjadi dosen pembimbing	Dosen menyetujui menjadi dosen pembimbing setelah ditunjuk dosen pembimbing .
A05	PB02	Mengisi formulir	Mahasiswa dapat mengisi formulir persetujuan untuk melakukan ujian proposal.



		persetujuan ujian proposal	
A06	PB02	Menyetujui persetujuan ujian proposal	Dosen pembimbing dan ketua program studi dapat menyetujui formulir persetujuan ujian proposal bila mahasiswa memenuhi syarat atau menolak jika tidak memenuhi syarat.
A07	PB02	Mengisi formulir pengajuan ujian proposal	Mahasiswa dapat mengisi formulir pengajuan ujian proposal setelah mendapat persetujuan dari dosen pembimbing dan ketua program studi.
A08	PB02	Meyetujui pengajuan ujian proposal	Akademik, dosen pembimbing, dosen penguji dan ketua program studi dapat menyetujui pengajuan ujian proposal mahasiswa. Ketua program studi juga dapat memilih dosen penguji saat menyetujui pengajuan ujian proposal
A09	PB02	Mengisi nilai ujian proposal dan saran perbaikan	Akademik dapat mengisi nilai ujian proposal dan saran perbaikan berdasarkan formulir penilaian dari dosen pembimbing dan dosen penguji
A10	PB03	Mengisi formulir persetujuan ujian seminar hasil	Mahasiswa dapat mengisi formulir persetujuan untuk mengadakan seminar hasil.
A11	PB03	Menyetujui persetujuan ujian seminar hasil	Dosen pembimbing dan ketua program studi dapat menyetujui formulir persetujuan ujian seminar hasil bila mahasiswa memenuhi



			syarat atau menolak jika tidak memenuhi syarat.
A12	PB03	Mengisi formulir pengajuan ujian seminar hasil	Mahasiswa dapat mengisi formulir pengajuan ujian seminar hasil setelah mendapat persetujuan dari dosen pembimbing dan ketua program studi.
A13	PB03	Menyetujui pengajuan seminar hasil	Dosen pembimbing dan ketua program studi dapat menyetujui formulir pengajuan ujian seminar hasil mahasiswa.
A14	PB03	Mengisi nilai seminar hasil dan saran perbaikan	Akademik dapat mengisi nilai seminar hasil dan saran perbaikan berdasarkan formulir penilaian dari dosen pembimbing.
A15	PB04	Mengisi formulir persetujuan ujian tesis	Mahasiswa dapat mengisi formulir persetujuan untuk melakukan ujian tesis.
A16	PB04	Menyetujui persetujuan ujian tesis	Dosen pembimbing dan ketua program studi dapat menyetujui formulir persetujuan ujian tesis bila mahasiswa memenuhi syarat atau menolak jika tidak memenuhi syarat.
A17	PB04	Mengisi formulir pengajuan ujian tesis	Mahasiswa dapat mengisi formulir pengajuan ujian tesis setelah mendapat persetujuan dari dosen pembimbing dan ketua program studi.
A18	PB04	Menyetujui pengajuan ujian tesis	Dosen pembimbing, ketua program studi dapat dan dosen penguji menyetujui pengajuan ujian tesis mahasiswa. Ketua program studi dapat memilih dosen penguji.
A19	PB04	Mengisi nilai ujian tesis dan saran perbaikan	Akademik dapat mengisi nilai ujian tesis dan saran perbaikan berdasarkan formulir penilaian dari dosen pembimbing dan dosen penguji.



4.4 Identifikasi Aktor

Berdasarkan pemetaan proses bisnis yang telah dilakukan maka didapatkan beberapa aktor. Aktor adalah seseorang atau sistem yang terlibat dengan sistem secara langsung. Pada Tabel 4.3 menjelaskan aktor dan deskripsinya

Tabel 4.3 Identifikasi aktor

Aktor	Deskripsi
Pengguna	Dalam sistem ini, pengguna dapat login untuk menggunakan sistem atau mendaftar sebagai mahasiswa
Mahasiswa	Dalam sistem ini, mahasiswa dapat mengajukan judul dan dosen pembimbing, mengajukan ujian proposal, mengajukan seminar hasil dan mengajukan ujian tesis.
Pegawai	Dalam sistem ini, pegawai yang dapat berarti pihak akademik, dosen pembimbing, dosen penguji dan ketua program studi dapat melihat data profil dan proses pengerjaan tesis mahasiswa mulai dari ujian proposal hingga ujian tesis.
Akademik	Dalam sistem ini, akademik dapat mengecek kelengkapan mahasiswa yang mengajukan judul dan dosen pembimbing serta yang mengajukan ujian proposal, dan dapat mengisi nilai ujian proposal, seminar hasil dan ujian tesis berdasarkan formulir penilaian dari dosen pembimbing dan penguji.
Dosen Pembimbing	Dalam sistem ini, dosen pembimbing ikut serta berperan dalam proses persetujuan dan proses pengajuan ujian proposal, seminar hasil dan ujian tesis.
Dosen Penguji	Dalam sistem ini, dosen penguji ikut berperan dalam proses pengajuan ujian proposal dan ujian tesis yang diajukan oleh mahasiswa
Ketua Program Studi	Dalam sistem ini, Ketua program studi sebagai pihak yang juga berperan dalam tiap tahap pengerjaan tesis mahasiswa. Ketua program studi dapat menentukan dosen pembimbing bagi mahasiswa serta dosen penguji pada ujian proposal dan ujian tesis.



4.5 Analisis dan Spesifikasi Kebutuhan

Pada Tabel 4.4 menjelaskan analisis dan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan

Tabel 4.4 Analisis dan spesifikasi kebutuhan

Kode Aktivitas	Kode Kebutuhan	Deskripsi
-	SIMT-1-100	P/L harus menyediakan halaman untuk <i>login</i> dan <i>logout</i> . Spesifikasi Kebutuhan: <ul style="list-style-type: none"> • P/L harus menyediakan formulir untuk <i>login</i> kedalam sistem (SIMT-1-101). • P/L harus menyediakan fungsi untuk <i>logout</i> dari sistem (SIMT-1-102).
-	SIMT-1-200	P/L harus menyediakan halaman pendaftaran Spesifikasi Kebutuhan: <ul style="list-style-type: none"> • P/L harus menyediakan formulir pendaftaran untuk mendaftar sebagai mahasiswa (SIMT-2-201) • P/L harus menyediakan formulir pendaftaran untuk mendaftar sebagai dosen (SIMT-2-202)
-	SIMT-1-300	P/L harus menyediakan halaman untuk mengubah password Spesifikasi Kebutuhan: <ul style="list-style-type: none"> • P/L harus menyediakan halaman untuk mengatur ulang password (SIMT-1-301) • P/L harus menyediakan halaman untuk mengubah password (SIMT-1-302)
A01, A02, A03, A04,	SIMT-1-400	P/L harus mampu menyediakan halaman untuk manajemen pengajuan judul dan dosen pembimbing tesis Spesifikasi Kebutuhan: <ul style="list-style-type: none"> • P/L harus mampu menyediakan halaman untuk mahasiswa dapat mengajukan judul tesis dan memilih dosen sebagai calon dosen pembimbing tesis (SIMT-1-401)



		<ul style="list-style-type: none"> • P/L harus mampu menyediakan halaman untuk akademik dapat menolak atau menyetujui pengajuan judul tesis dan dosen pembimbing tesis (SIMT-1-402) • P/L harus mampu menyediakan halaman untuk kaprodi dapat memilih dosen sebagai dosen pembimbing tesis dan menyetujui atau menolak pengajuan judul tesis (SIMT-1-403) • P/L harus mampu menyediakan halaman untuk dosen dapat menyetujui menjadi dosen pembimbing (SIMT-1-404). • P/L harus mampu menyediakan halaman untuk kaprodi dapat memberikan persetujuan akhir pengajuan judul dan dosen pembimbing tesis (SIMT-1-405). • P/L harus mampu menyediakan halaman untuk melihat data tesis dan pengajuan judul dan dosen pembimbing tesis (SIMT-1-406).
A05, A06, A07, A08, A09	SIMT-1-500	<p>P/L harus mampu menyediakan halaman untuk manajemen pengajuan ujian proposal tesis</p> <p>Spesifikasi Kebutuhan:</p> <ul style="list-style-type: none"> • P/L harus mampu menyediakan halaman untuk mahasiswa mengisi formulir persetujuan pengajuan ujian proposal tesis (SIMT-1-501). • P/L harus mampu menyediakan halaman untuk kaprodi dan dosen dapat menolak atau menyetujui persetujuan ujian proposal tesis (SIMT-1-502). • P/L harus mampu menyediakan halaman untuk mahasiswa mengisi formulir pengajuan ujian proposal tesis (SIMT-1-503). • P/L harus mampu menyediakan halaman untuk kaprodi, akademik, dosen pembimbing dan dosen penguji untuk menyetujui pengajuan ujian proposal tesis (SIMT-1-504).



		<ul style="list-style-type: none"> • P/L harus mampu menyediakan halaman untuk mahasiswa, dosen pembimbing, dosen penguji, kaprodi dan akademik untuk dapat melihat data ujian proposal (SIMT-1-505). • P/L harus mampu menyediakan halaman untuk akademik dapat mengisi nilai dan saran perbaikan ujian proposal (SIMT-1-506).
A10, A11, A12, A13, A14	SIMT-1-600	<p>P/L harus mampu menyediakan halaman untuk manajemen pengajuan seminar hasil</p> <p>Spesifikasi Kebutuhan:</p> <ul style="list-style-type: none"> • P/L harus mampu menyediakan halaman untuk mahasiswa dapat mengisi formulir persetujuan pengajuan seminar hasil (SIMT-1-601). • P/L harus mampu menyediakan halaman untuk dosen pembimbing dan kaprodi untuk menolak atau menyetujui formulir persetujuan pengajuan seminar hasil (SIMT-1-602). • P/L harus mampu menyediakan halaman untuk mahasiswa mengisi formulir pengajuan seminar hasil (SIMT-1-603). • P/L harus mampu menyediakan halaman untuk dosen pembimbing, kaprodi dan kaprodi untuk menyetujui pengajuan seminar hasil (SIMT-1-604). • P/L harus mampu menyediakan halaman untuk mahasiswa, dosen pembimbing, kaprodi dan akademik untuk dapat melihat data pelaksanaan seminar hasil (SIMT-1-605). • P/L harus mampu menyediakan halaman untuk akademik dapat mengisi nilai dan saran perbaikan seminar hasil (SIMT-1-606).
A15, A16, A17, A18, A19,	SIMT-1-700	<p>P/L harus mampu menyediakan halaman untuk manajemen pengajuan ujian tesis</p> <p>Spesifikasi Kebutuhan:</p> <ul style="list-style-type: none"> • P/L harus mampu menyediakan halaman untuk mahasiswa dapat mengisi formulir persetujuan pengajuan ujian tesis (SIMT-1-701).



		<ul style="list-style-type: none"> • P/L harus mampu menyediakan halaman untuk dosen pembimbing dan kaprodi untuk menolak atau menyetujui formulir persetujuan pengajuan ujian tesis (SIMT-1-702). • P/L harus mampu menyediakan halaman untuk mahasiswa dapat mengisi formulir pengajuan ujian tesis (SIMT-1-703). • P/L harus mampu menyediakan halaman untuk dosen pembimbing, dosen penguji, akademik dan kaprodi untuk menyetujui pengajuan ujian tesis (SIMT-1-704). • P/L harus mampu menyediakan halaman untuk mahasiswa, dosen pembimbing, kaprodi dan akademik untuk dapat melihat data pelaksanaan ujian tesis (SIMT-1-705). • P/L harus mampu menyediakan halaman untuk akademik dapat mengisi nilai dan saran perbaikan ujian tesis (SIMT-1-706).
-	SIMT-1-800	P/L harus mampu menyediakan halaman profil.
		<p>Spesifikasi Kebutuhan</p> <ul style="list-style-type: none"> • P/L harus menyediakan fungsi bagi mahasiswa, dosen pembimbing, kaprodi dan akademik dapat melihat profilnya (SIMT-1-801) • P/L harus menyediakan fungsi bagi dosen pembimbing, dosen penguji, kaprodi dan akademik untuk melihat profil mahasiswa (SIMT-1-802) • P/L harus menyediakan fungsi bagi kaprodi untuk melihat profil dosen (SIMT-1-803)



4.5.1 Kebutuhan Fungsional Pengguna

Pada Tabel 4.5 menjelaskan kebutuhan fungsional sistem yang dapat dilakukan pengguna

Tabel 4.5 Kebutuhan fungsional pengguna

Kode Kebutuhan	Kode Fungsi	Nama Fungsi	Deskripsi
SIMT-1-101	SIMT-2-001	<i>Login</i>	Pengguna dapat melakukan <i>login</i> untuk mendapatkan hak akses sebagai mahasiswa atau pegawai.
SIMT-2-201	SIMT-2-002	Mendaftar sebagai mahasiswa	Pengguna dapat melakukan pendaftaran untuk mendapatkan hak akses sebagai mahasiswa.
SIMT-2-202	SIMT-2-003	Mendaftar sebagai dosen	Pengguna dapat melakukan pendaftaran untuk mendapatkan hak akses sebagai dosen.
SIMT-1-301	SIMT-2-004	Atur ulang <i>password</i>	Pengguna dapat melakukan pengaturan ulang <i>password</i> nya.

4.5.2 Kebutuhan Fungsional Pegawai

Pada Tabel 4.6 menjelaskan kebutuhan fungsional sistem yang dapat dilakukan pegawai

Tabel 4.6 Kebutuhan fungsional pegawai

Kode Kebutuhan	Kode Fungsi	Nama Fungsi	Deskripsi
SIMT-1-802	SIMT-2-005	Melihat profil mahasiswa	Pegawai dapat melihat profil mahasiswa.
SIMT-1-801	SIMT-2-006	Melihat profil	Pegawai dapat melihat profilnya sendiri.
SIMT-1-406	SIMT-2-007	Melihat data tesis	Pegawai dapat melihat data tesis dari mahasiswa.
SIMT-1-406	SIMT-2-008	Melihat data pengajuan judul dan dosen	Pegawai dapat melihat data pengajuan judul dan dosen



		pembimbing tesis	pembimbing tesis dari mahasiswa.
SIMT-1-505	SIMT-2-009	Melihat data ujian proposal	Pegawai dapat melihat data ujian proposal dari mahasiswa.
SIMT-1-605	SIMT-2-010	Melihat data seminar hasil	Pegawai dapat melihat data seminar hasil dari mahasiswa.
SIMT-1-705	SIMT-2-011	Melihat data ujian tesis	Pegawai dapat melihat data ujian tesis dari mahasiswa.
SIMT-1-102	SIMT-2-012	<i>Logout</i>	Pegawai dapat <i>logout</i> dari sistem.
SIMT-1-302	SIMT-2-013	Ubah <i>password</i>	Pegawai dapat mengubah <i>password</i> nya.

4.5.3 Kebutuhan Fungsional Mahasiswa

Pada Tabel 4.7 menjelaskan kebutuhan fungsional sistem yang dapat dilakukan mahasiswa.

Tabel 4.7 Kebutuhan fungsional mahasiswa

Kode Kebutuhan	Kode Fungsi	Nama Fungsi	Deskripsi
SIMT-1-302	SIMT-2-014	Ubah <i>password</i>	Mahasiswa dapat mengubah <i>password</i> nya.
SIMT-1-401	SIMT-2-015	Mengisi formulir pengajuan judul dan dosen pembimbing	Mahasiswa dapat mengisi formulir pengajuan judul tesis dan memilih dosen sebagai calon dosen pembimbing.
SIMT-1-406	SIMT-2-016	Melihat data pengajuan judul dan dosen pembimbing	Mahasiswa dapat melihat data tesis dan pengajuan judul sudah diajukan.
SIMT-1-501	SIMT-2-017	Mengisi formulir persetujuan ujian proposal	Mahasiswa dapat mengisi formulir permintaan persetujuan dilakukan pengajuan ujian proposal.
SIMT-1-503	SIMT-2-018	Mengisi formulir pengajuan ujian proposal	Mahasiswa dapat mengisi formulir pengajuan ujian proposal berdasarkan waktu



			yang telah disepakati dengan dosen pembimbing.
SIMT-1-505	SIMT-2-019	Melihat data ujian proposal	Mahasiswa dapat melihat data mengenai ujian proposal.
SIMT-1-601	SIMT-2-020	Mengisi formulir persetujuan seminar hasil	Mahasiswa mengisi formulir permintaan persetujuan dilakukan pengajuan seminar hasil.
SIMT-1-603	SIMT-02-021	Mengisi formulir pengajuan seminar hasil	Mahasiswa mengisi formulir pengajuan seminar hasil waktu yang telah disepakati dengan dosen pembimbing.
SIMT-1-605	SIMT-02-022	Melihat data seminar hasil	Mahasiswa dapat melihat data mengenai seminar hasil.
SIMT-1-701	SIMT-02-023	Mengisi formulir persetujuan ujian tesis	Mahasiswa mengisi formulir permintaan persetujuan dilakukan pengajuan ujian tesis.
SIMT-1-703	SIMT-02-024	Mengisi formulir pengajuan ujian tesis	Mahasiswa mengisi formulir pengajuan ujian tesis berdasarkan waktu yang telah disepakati dengan dosen pembimbing.
SIMT-1-705	SIMT-02-025	Melihat data ujian tesis	Mahasiswa dapat melihat data ujian tesis.
SIMT-1-901	SIMT-02-026	Melihat profil	Mahasiswa dapat melihat profilnya.
SIMT-1-102	SIMT-02-027	<i>Logout</i>	Mahasiswa dapat <i>logout</i> dari sistem

4.5.4 Kebutuhan Fungsional Dosen Pembimbing

Pada Tabel 4.8 menjelaskan kebutuhan fungsional sistem yang dapat dilakukan dosen pembimbing.

Tabel 4.8 Kebutuhan fungsional dosen pembimbing

Kode Kebutuhan	Kode Fungsi	Nama Fungsi	Deskripsi
SIMT-1-404	SIMT-02-028	Menyetujui menjadi dosen pembimbing	Dosen pembimbing dapat menyetujui menjadi dosen



			pembimbing apabila telah ditunjuk oleh kaprodi.
SIMT-1-502	SIMT-02-029	Mengelola persetujuan ujian proposal	Dosen pembimbing dapat menyetujui atau menolak permintaan persetujuan ujian proposal.
SIMT-1-602	SIMT-02-030	Mengelola persetujuan seminar hasil	Dosen pembimbing dapat menyetujui atau menolak permintaan persetujuan seminar hasil.
SIMT-1-702	SIMT-02-031	Mengelola persetujuan ujian tesis	Dosen pembimbing dapat menyetujui atau menolak permintaan persetujuan ujian tesis.
SIMT-1-504	SIMT-02-032	Menyetujui pengajuan ujian proposal	Dosen pembimbing dapat menyetujui pengajuan ujian proposal.
SIMT-1-604	SIMT-02-033	Menyetujui pengajuan seminar hasil	Dosen pembimbing dapat menyetujui pengajuan seminar hasil.
SIMT-1-704	SIMT-02-034	Menyetujui pengajuan ujian tesis	Dosen pembimbing dapat menyetujui pengajuan ujian tesis.

4.5.5 Kebutuhan Fungsional Dosen Penguji

Pada Tabel 4.9 menjelaskan kebutuhan fungsional sistem yang dapat dilakukan dosen penguji.

Tabel 4.9 Kebutuhan fungsional dosen penguji

Kode Kebutuhan	Kode Fungsi	Nama Fungsi	Deskripsi
SIMT-1-504	SIMT-02-035	Menyetujui pengajuan ujian proposal	Dosen penguji dapat menyetujui pengajuan ujian proposal.
SIMT-1-704	SIMT-02-036	Menyetujui pengajuan ujian tesis	Dosen penguji dapat menyetujui pengajuan ujian tesis.



4.5.6 Kebutuhan Fungsional Kaprodi

Pada Tabel 4.10 menjelaskan kebutuhan fungsional sistem yang dapat dilakukan Kaprodi.

Tabel 4.10 Kebutuhan fungsional Kaprodi

Kode Kebutuhan	Kode Fungsi	Nama Fungsi	Deskripsi
SIMT-1-403	SIMT-02-037	Mengelola pengajuan judul dan menentukan dosen pembimbing	Kaprodi dapat menyetujui atau menolak pengajuan judul tesis mahasiswa dan menentukan dosen pembimbing.
SIMT-1-502	SIMT-02-038	Mengelola persetujuan ujian proposal	Kaprodi dapat menyetujui atau menolak persetujuan ujian proposal.
SIMT-1-602	SIMT-02-039	Mengelola persetujuan seminar hasil	Kaprodi dapat menyetujui atau menolak persetujuan seminar hasil.
SIMT-1-702	SIMT-02-040	Mengelola persetujuan ujian tesis	Kaprodi dapat menyetujui atau menolak persetujuan ujian tesis.
SIMT-1-504	SIMT-02-041	Menyetujui pengajuan ujian proposal dan menentukan dosen penguji	Kaprodi dapat menyetujui pengajuan ujian proposal dan menentukan dosen penguji.
SIMT-1-604	SIMT-02-042	Menyetujui pengajuan seminar hasil	Kaprodi dapat menyetujui pengajuan seminar hasil.
SIMT-1-704	SIMT-02-043	Menyetujui pengajuan ujian tesis dan menentukan dosen penguji	Kaprodi dapat menyetujui pengajuan ujian tesis dan menentukan dosen penguji.
SIMT-1-803	SIMT-02-044	Melihat profil dosen	Kaprodi dapat melihat profil dosen pembimbing dan dosen penguji.



4.5.7 Kebutuhan Fungsional Akademik

Pada Tabel 4.11 menjelaskan kebutuhan fungsional sistem yang dapat dilakukan Akademik.

Tabel 4.11 Kebutuhan fungsional akademik

Kode Kebutuhan	Kode Fungsi	Nama Fungsi	Deskripsi
SIMT-1-402	SIMT-02-045	Mengelola pengajuan judul dan dosen pembimbing	Akademik dapat menyetujui atau menolak pengajuan judul dan dosen pembimbing.
SIMT-1-504	SIMT-02-046	Menyetujui pengajuan ujian proposal	Akademik dapat menyetujui pengajuan ujian proposal.
SIMT-1-604	SIMT-02-047	Menyetujui pengajuan ujian seminar hasil	Akademik dapat menyetujui pengajuan ujian seminar hasil
SIMT-1-704	SIMT-02-048	Menyetujui pengajuan ujian tesis	Akademik dapat menyetujui pengajuan ujian tesis.
SIMT-1-506	SIMT-02-049	Mengisi nilai dan saran perbaikan ujian proposal	Akademik dapat mengisi nilai dan saran perbaikan ujian proposal.
SIMT-1-606	SIMT-02-050	Mengisi nilai dan saran perbaikan seminar hasil	Akademik dapat mengisi nilai dan saran perbaikan seminar hasil.
SIMT-1-706	SIMT-02-051	Mengisi nilai dan saran perbaikan ujian tesis	Akademik dapat mengisi nilai dan saran perbaikan ujian tesis.



4.5.8 Kebutuhan Non Fungsional

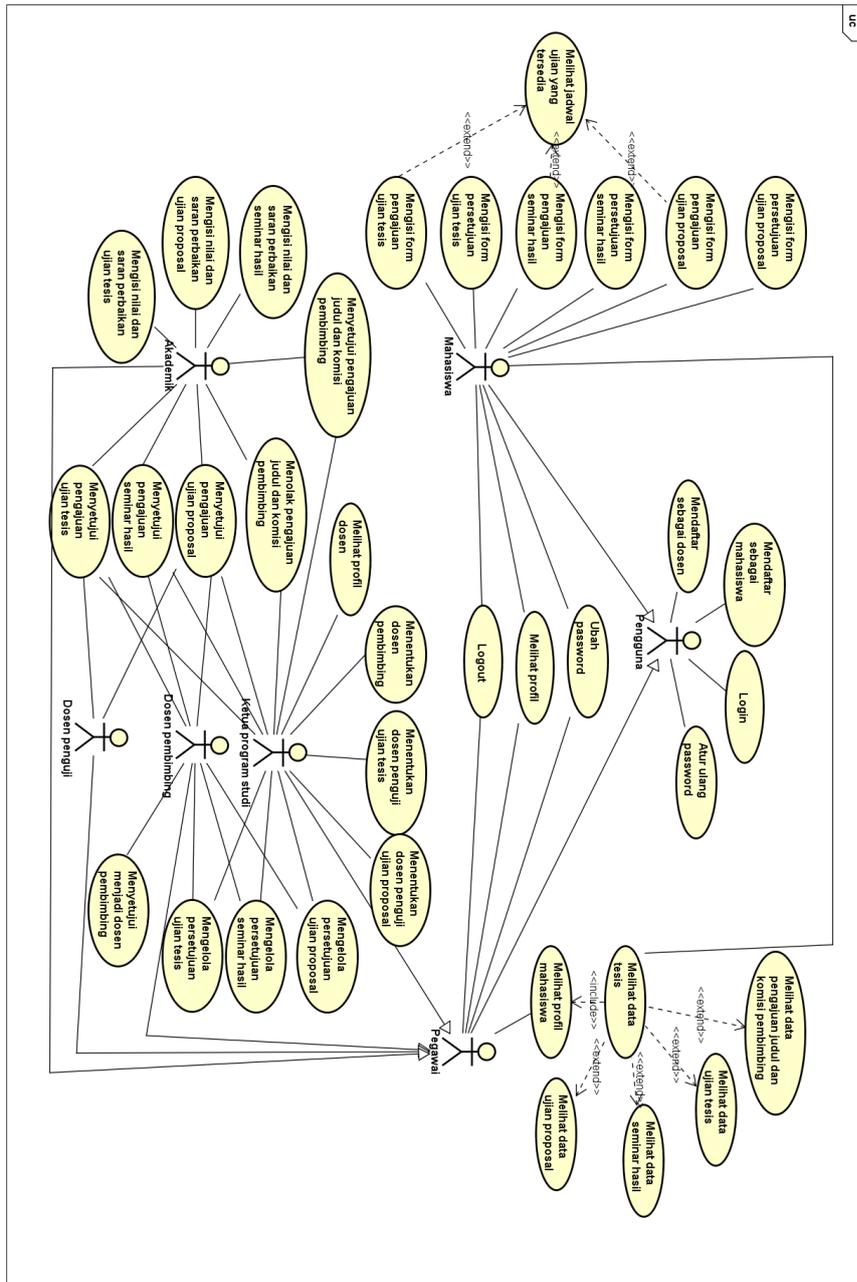
Kebutuhan non fungsional dari sistem yang akan dikembangkan dapat dilihat pada Tabel 4.12.

Tabel 4.12 Kebutuhan non fungsional

Kode Fungsi	Nama Fitur	Deskripsi
SIMT-02-052	Kompatibilitas	Sistem dapat dibuka pada berbagai macam <i>web browser</i> .

4.6 Pemodelan Use Case Diagram

Pemodelan *use case diagram* menggambarkan aktor-aktor yang terlibat dalam sistem beserta kegiatannya di dalam sistem tersebut. Pemodelan *use case diagram* dibuat berdasarkan kebutuhan fungsional sistem yang telah dijelaskan sebelumnya. Pemodelan *use case diagram* dari Sistem Informasi Manajemen Tesis dapat dilihat pada Gambar 4.9.



Gambar 4.9 Pemodelan use case diagram Sistem Informasi Manajemen Tesis

4.7 Use Case Scenario

Use Case Scenario merupakan penjabaran dari tiap use case yang dikerjakan oleh aktor-aktor di dalam sistem.

4.7.1 Use Case Scenario Mendaftar Sebagai Mahasiswa

Use case scenario mendaftar sebagai mahasiswa merupakan penjelasan alur dari proses mendaftar sebagai mahasiswa. Pada Tabel 4.13 merupakan use case scenario mendaftar sebagai mahasiswa.

Tabel 4.13 *Use case scenario* mendaftar sebagai mahasiswa

Alur Proses untuk mendaftar sebagai mahasiswa	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor pengguna dapat mendaftar sebagai mahasiswa.
Aktor	Pengguna
Pre-Condition	Aktor telah membuka halaman pendaftaran
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor mengisi <i>email</i>, <i>password</i>, nama, NIM dan nomor telepon pada form pendaftaran 2. Sistem memproses data dari formulir pendaftaran 3. Sistem menampilkan pesan pemberitahuan bahwa pendaftaran berhasil
Alternative Flows	3.a Sistem menampilkan pesan pemberitahuan bahwa NIM/email sudah terdaftar 3.a.1 Kembali ke langkah 1
Post-Conditions	Aktor berhasil mendaftar menjadi mahasiswa

4.7.2 *Use Case Scenario* Mendaftar Sebagai Dosen

Use case scenario mendaftar sebagai dosen merupakan penjelasan alur dari proses mendaftar sebagai dosen. Pada Tabel 4.14 merupakan *use case scenario* mendaftar sebagai dosen.

Tabel 4.14 *Use case scenario* Mendaftar Sebagai Dosen

Alur Proses untuk mendaftar sebagai dosen	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor pengguna dapat mendaftar sebagai dosen.
Aktor	Pengguna
Pre-Condition	Aktor telah membuka halaman pendaftaran
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor mengisi NIP, nama, email, password dan memilih bidang minat pada formulir pendaftaran 2. Sistem memproses data dari formulir pendaftaran 3. Sistem menampilkan pesan pemberitahuan bahwa pendaftaran berhasil
Alternative Flows	3.a Sistem menampilkan pesan pemberitahuan bahwa NIP/email sudah terdaftar 3.a.1 Kembali ke langkah 1
Post-Conditions	Aktor berhasil mendaftar menjadi dosen



4.7.3 Use Case Scenario Login

Use case scenario mendaftar sebagai dosen merupakan penjelasan alur dari proses mendaftar sebagai dosen. Pada Tabel 4.15 merupakan *use case scenario* mendaftar sebagai dosen.

Tabel 4.15 Use case scenario login

Alur Proses untuk login	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor pengguna dapat <i>login</i> ke dalam sistem. Setelah pengguna berhasil login, pengguna dapat menjadi aktor mahasiswa, akademik, kaprodi, dosen pembimbing ataupun dosen penguji tergantung data yang dimasukkan pada formulir <i>login</i> .
Aktor	Pengguna
Pre-Condition	Aktor telah membuka halaman <i>login</i> .
Basic Flow of Events	<ol style="list-style-type: none"> 1. Pengguna mengisi <i>email</i> dan password pada formulir <i>login</i>. 2. Sistem melakukan pengecekan <i>email</i> dan <i>password</i>. 3. Sistem membuat <i>token login</i>. 4. Sistem menampilkan halaman <i>home</i>. 5. Aktor berhasil masuk ke dalam sistem.
Alternative Flows	<p>2.a Sistem menampilkan pesan bahwa <i>email</i> dan <i>password</i> tidak terdaftar.</p> <p>2.a.1 Kembali ke langkah 1.</p>
Post-Conditions	Aktor berhasil <i>login</i> kedalam sistem

4.7.4 Use Case Scenario Logout

Use case scenario mendaftar sebagai dosen merupakan penjelasan alur dari proses mendaftar sebagai dosen. Pada Tabel 4.16 merupakan *use case scenario* mendaftar sebagai dosen.

Tabel 4.16 Use case scenario logout

Alur Proses untuk Logout	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor pengguna dapat <i>logout</i> dari sistem
Aktor	Pegawai dan Mahasiswa
Pre-Condition	Aktor telah telah <i>login</i> ke dalam sistem.



Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor menekan tombol <i>logout</i> 2. Sistem menghapus token login 3. Sistem menampilkan halaman login 4. Aktor berhasil <i>logout</i> dari sistem
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil <i>logout</i> dari sistem

4.7.5 Use Case Scenario Menyetujui Pengajuan Judul dan Dosen Pembimbing

Use case scenario menyetujui pengajuan judul dan dosen pembimbing merupakan penjelasan alur dari proses menyetujui judul dan dosen pembimbing. Pada Tabel 4.17 merupakan *use case scenario* menyetujui pengajuan judul dan dosen pembimbing.

Tabel 4.17 Use case scenario menyetujui pengajuan judul tesis dan dosen pembimbing

Alur proses untuk menyetujui pengajuan judul tesis dan dosen pembimbing	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor menyetujui pengajuan judul tesis dan dosen pembimbing
Aktor	Kaprodi dan Akademik
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu pengajuan judul. 2. Sistem menampilkan pengajuan judul dan dosen pembimbing. 3. Aktor mengisi formulir pengajuan judul tesis dan dosen pembimbing 4. Aktor menekan tombol submit 5. Sistem menyimpan data formulir pengajuan judul tesis dan dosen pembimbing 6. Sistem menampilkan pesan berhasil
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil menyetujui pengajuan judul tesis dan dosen pembimbing

4.7.6 Use Case Scenario Mengisi Formulir Pengajuan Judul dan Dosen Pembimbing

Use case scenario mengisi formulir pengajuan judul dan dosen pembimbing merupakan penjelasan alur dari proses mahasiswa mengajukan judul dan dosen



pembimbing. Pada Tabel 4.18 merupakan *use case scenario* formulir pengajuan judul dan dosen pembimbing.

Tabel 4.18 Use case scenario mengisi formulir pengajuan judul dan dosen pembimbing

Alur proses untuk mengisi formulir pengajuan judul dan dosen pembimbing	
Deskripsi	Use case ini menjelaskan bagaimana mahasiswa mengajukan judul dan dosen pembimbing
Aktor	Mahasiswa
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu pengajuan judul 2. Sistem menampilkan pengajuan judul dan dosen pembimbing 3. Aktor menekan tombol ajukan judul dan mengisi formulir pengajuan judul tesis dan dosen pembimbing 4. Aktor menekan tombol submit 5. Sistem menyimpan data formulir pengajuan judul dan dosen pembimbing 6. Sistem menampilkan halaman pengajuan judul
Alternative Flows	<ol style="list-style-type: none"> 3.a.1 Aktor tidak mengisi kolom yang wajib diisi 3.a.2 Sistem menampilkan pesan kesalahan
Post-Conditions	Aktor berhasil mengajukan pengajuan judul dan dosen pembimbing

4.7.7 Use Case Scenario Mengisi Formulir Persetujuan Ujian Proposal

Use case scenario mengisi formulir persetujuan ujian proposal merupakan penjelasan alur dari proses mahasiswa mengajukan persetujuan ujian proposal. Pada Tabel 4.19 merupakan *use case scenario* mengisi formulir persetujuan ujian proposal.

Tabel 4.19 Use case scenario mengisi formulir persetujuan ujian proposal

Alur proses untuk mengisi formulir persetujuan ujian proposal	
Deskripsi	Use case ini menjelaskan bagaimana mahasiswa mengisi formulir persetujuan ujian proposal
Aktor	Mahasiswa
Pre-Condition	Aktor telah login ke dalam sistem



Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu ujian proposal 2. Sistem menampilkan halaman ujian proposal 3. Aktor menekan judul tesis dan mengisi formulir persetujuan ujian proposal 4. Aktor menekan tombol submit 5. Sistem menampilkan pesan berhasil dan menyimpan data persetujuan ujian proposal 6. Sistem menampilkan halaman ujian proposal
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil mengajukan persetujuan ujian proposal

4.7.8 Use Case Scenario Mengisi Formulir Persetujuan Seminar Hasil

Use case scenario mengisi formulir persetujuan seminar hasil merupakan penjelasan alur dari proses mahasiswa mengajukan persetujuan seminar hasil. Pada Tabel 4.20 merupakan *use case scenario* mengisi formulir persetujuan seminar hasil.

Tabel 4.20 Use case scenario mengisi formulir persetujuan seminar hasil

Alur Proses untuk persetujuan ujian seminar hasil	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana mahasiswa mengisi formulir persetujuan ujian seminar hasil
Aktor	Mahasiswa
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu ujian seminar hasil 2. Sistem menampilkan ujian seminar hasil 3. Aktor menekan judul tesis dan mengisi formulir persetujuan ujian seminar hasil 4. Aktor menekan tombol submit 5. Sistem menampilkan pesan berhasil dan menyimpan data persetujuan ujian seminar hasil 6. Sistem menampilkan seminar hasil
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil mengajukan persetujuan ujian seminar hasil

4.7.9 Use Case Scenario Mengisi Formulir Persetujuan Ujian Tesis

Use case scenario mengisi formulir persetujuan ujian tesis merupakan penjelasan alur dari proses mahasiswa mengajukan persetujuan ujian tesis. Pada Tabel 4.21 merupakan *use case scenario* mengisi formulir persetujuan ujian tesis.

Tabel 4.21 *Use case scenario* mengisi formulir persetujuan ujian tesis

Alur Proses untuk persetujuan ujian tesis	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana mahasiswa mengisi formulir persetujuan ujian tesis
Aktor	Mahasiswa
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu ujian tesis 2. Sistem menampilkan halaman ujian tesis 3. Aktor menekan judul tesis dan mengisi formulir persetujuan ujian tesis 4. Aktor menekan tombol submit 5. Sistem menampilkan pesan berhasil dan menyimpan data persetujuan ujian tesis 6. Sistem menampilkan halaman ujian tesis
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil mengajukan persetujuan ujian tesis

4.7.10 *Use Case Scenario* Mengisi Formulir Pengajuan Ujian Proposal

Use case scenario mengisi formulir pengajuan ujian proposal merupakan penjelasan alur dari proses mahasiswa mengajukan ujian proposal. Pada Tabel 4.22 merupakan *use case scenario* mengisi formulir pengajuan ujian proposal.

Tabel 4.22 *Use case scenario* mengisi formulir pengajuan ujian proposal

Alur proses untuk mengisi formulir pengajuan ujian proposal	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana mahasiswa mengisi formulir pengajuan ujian proposal
Aktor	Mahasiswa
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu ujian proposal 2. Sistem menampilkan ujian proposal 3. Aktor menekan tombol ujian proposal pada judul tesis 4. Aktor mengisi formulir pengajuan ujian proposal 5. Aktor menekan tombol submit



	6. Sistem menampilkan pesan berhasil dan menyimpan data formulir pengajuan ujian proposal 7. Sistem menampilkan halaman ujian proposal
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil mengajukan ujian proposal

4.7.11 Use Case Scenario Mengisi Formulir Pengajuan Seminar Hasil

Use case scenario mengisi formulir pengajuan seminar hasil merupakan penjelasan alur dari proses mahasiswa mengajukan seminar hasil. Pada Tabel 4.23 merupakan *use case scenario* mengisi formulir pengajuan seminar hasil.

Tabel 4.23 Use case scenario mengisi formulir pengajuan ujian seminar hasil

Alur Proses untuk mengisi formulir pengajuan ujian seminar hasil	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana mahasiswa mengisi formulir pengajuan ujian seminar hasil
Aktor	Mahasiswa
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	1. Aktor memilih menu ujian seminar hasil 2. Sistem menampilkan halaman seminar hasil 3. Aktor menekan tombol seminar hasil pada judul tesis 4. Aktor mengisi formulir pengajuan seminar hasil 5. Aktor menekan tombol submit 6. Sistem menampilkan pesan berhasil dan menyimpan data formulir pengajuan ujian seminar hasil 7. Sistem menampilkan halaman seminar hasil
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil mengajukan ujian seminar hasil

4.7.12 Use Case Scenario Mengisi Formulir Pengajuan Ujian Tesis

Use case scenario mengisi formulir pengajuan ujian tesis merupakan penjelasan alur dari proses mahasiswa mengajukan ujian tesis. Pada Tabel 4.24 merupakan *use case scenario* mengisi formulir pengajuan ujian tesis.

Tabel 4.24 Use case scenario mengisi formulir pengajuan ujian tesis

Alur Proses untuk mengisi formulir pengajuan ujian tesis



Deskripsi	<i>Use case</i> ini menjelaskan bagaimana mahasiswa mengisi formulir pengajuan ujian tesis
Aktor	Mahasiswa
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu ujian tesis 2. Sistem menampilkan halaman ujian tesis 3. Aktor menekan tombol ujian tesis pada judul tesis 4. Aktor mengisi formulir pengajuan ujian tesis 5. Aktor menekan tombol submit 6. Sistem menyimpan data formulir pengajuan ujian tesis 7. Sistem menampilkan halaman ujian tesis
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil mengajukan ujian tesis

4.7.13 *Use Case Scenario* Mengecek Syarat Pengajuan Judul dan Dosen Pembimbing

Use case scenario mengecek syarat pengajuan judul dan dosen pembimbing merupakan penjelasan alur dari proses akademik mengecek syarat pengajuan judul dan dosen pembimbing. Pada Tabel 4.25 merupakan *use case scenario* mengecek syarat pengajuan judul dan dosen pembimbing.

Tabel 4.25 *Use case scenario* mengecek syarat pengajuan judul tesis dan dosen pembimbing

Alur Proses untuk mengecek syarat pengajuan judul tesis dan dosen pembimbing	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana akademik mengecek syarat pengajuan judul tesis dan dosen pembimbing
Aktor	Akademik
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu pengajuan judul 2. Sistem menampilkan halaman pengajuan judul dan dosen pembimbing 3. Aktor mengisi formulir persyaratan pengajuan judul tesis dan dosen pembimbing 4. Aktor menekan tombol submit 5. Sistem menyimpan data persetujuan dari formulir persyaratan pengajuan judul tesis dan dosen pembimbing



	6. Sistem menampilkan pengajuan judul dan dosen pembimbing
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil mengecek syarat pengajuan judul tesis dan dosen pembimbing

4.7.14 Use Case Scenario Mengisi Nilai dan Saran Perbaikan Ujian Proposal

Use case scenario mengisi nilai dan saran perbaikan ujian proposal merupakan penjelasan alur dari proses akademik mengisi nilai dan saran perbaikan ujian proposal. Pada Tabel 4.26 merupakan *use case scenario* mengisi nilai dan saran perbaikan ujian proposal.

Tabel 4.26 Use case scenario mengisi nilai dan saran perbaikan ujian proposal

Alur Proses untuk mengisi nilai dan saran perbaikan ujian proposal	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor mengisi nilai dan saran perbaikan ujian proposal
Aktor	Akademik
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih ujian proposal 2. Aktor menekan tombol isi nilai pada judul tesis 3. Sistem menampilkan formulir nilai dan saran perbaikan ujian proposal 4. Aktor mengisi formulir nilai dan saran perbaikan ujian proposal 5. Aktor menekan tombol submit 6. Sistem menampilkan pesan berhasil dan menyimpan data formulir nilai dan saran ujian proposal
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil mengisi data nilai dan saran perbaikan ujian proposal mahasiswa

4.7.15 Use Case Scenario Mengisi Nilai dan Saran Perbaikan Seminar Hasil

Use case scenario mengisi nilai dan saran perbaikan seminar hasil merupakan penjelasan alur dari proses akademik mengisi nilai dan saran perbaikan seminar



hasil. Pada Tabel 4.27 merupakan *use case scenario* mengisi nilai dan saran perbaikan seminar hasil.

Tabel 4.27 Use case scenario mengisi nilai dan saran perbaikan seminar hasil

Alur Proses untuk mengisi nilai dan saran perbaikan seminar hasil	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor mengisi nilai dan saran perbaikan seminar hasil
Aktor	Akademik
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih seminar hasil 2. Aktor menekan tombol isi nilai pada judul tesis 3. Sistem menampilkan formulir nilai dan saran perbaikan seminar hasil 4. Aktor mengisi formulir nilai dan saran perbaikan seminar hasil 5. Aktor menekan tombol submit 6. Sistem menampilkan pesan berhasil dan menyimpan data formulir nilai dan saran seminar hasil
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil mengisi data nilai dan saran perbaikan seminar hasil mahasiswa

4.7.16 Use Case Scenario Mengisi Nilai dan Saran Perbaikan Ujian Tesis

Use case scenario mengisi nilai dan saran perbaikan ujian tesis merupakan penjelasan alur dari proses akademik mengisi nilai dan saran perbaikan ujian tesis. Pada Tabel 4.28 merupakan *use case scenario* mengisi nilai dan saran perbaikan ujian tesis.

Tabel 4.28 Use case scenario mengisi nilai dan saran perbaikan ujian tesis

Alur Proses untuk mengisi nilai dan saran perbaikan ujian tesis	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor mengisi nilai dan saran perbaikan ujian tesis
Aktor	Akademik
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu ujian tesis 2. Aktor menekan tombol isi nilai pada judul tesis 3. Sistem menampilkan formulir nilai dan saran perbaikan ujian tesis



	<ol style="list-style-type: none"> 4. Aktor mengisi formulir nilai dan saran perbaikan ujian tesis 5. Aktor menekan tombol submit 6. Sistem menampilkan pesan berhasil dan menyimpan data formulir nilai dan saran ujian tesis
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil mengisi data nilai dan saran perbaikan ujian tesis mahasiswa

4.7.17 Use Case Scenario Menolak Pengajuan Judul dan Dosen Pembimbing

Use case scenario menolak pengajuan judul dan dosen pembimbing merupakan penjelasan alur dari proses akademik atau kaprodi untuk menolak pengajuan judul dan dosen pembimbing bila tidak memenuhi syarat. Pada Tabel 4.29 merupakan *use case scenario* menolak pengajuan judul dan dosen pembimbing.

Tabel 4.29 Use case scenario menolak pengajuan judul tesis dan dosen pembimbing

Alur proses untuk menolak pengajuan judul tesis dan dosen pembimbing	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor menolak pengajuan judul tesis dan dosen pembimbing
Aktor	Ketua Program Studi dan Akademik
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu pengajuan judul 2. Sistem menampilkan halaman pengajuan judul dan dosen pembimbing 3. Aktor menolak persetujuan pengajuan judul tesis dan dosen pembimbing 4. Aktor menekan tombol submit 5. Sistem menampilkan pesan berhasil dan menyimpan data penolakan pengajuan judul tesis dan dosen pembimbing 6. Sistem menampilkan halaman pengajuan judul
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil menolak pengajuan judul tesis dan dosen pembimbing



4.7.18 Use Case Scenario Menyetujui Pengajuan Judul dan Dosen Pembimbing

Use case scenario menyetujui pengajuan judul dan dosen pembimbing merupakan penjelasan alur dari proses akademik atau kaprodi untuk menyetujui pengajuan judul dan dosen pembimbing bila memenuhi syarat. Pada Tabel 4.30 merupakan *use case scenario* menyetujui pengajuan judul dan dosen pembimbing.

Tabel 4.30 Use case scenario menyetujui pengajuan judul tesis dan dosen pembimbing

Alur proses untuk menyetujui pengajuan judul tesis dan dosen pembimbing	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor menyetujui pengajuan judul tesis dan dosen pembimbing
Aktor	Ketua Program Studi dan Akademik
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu pengajuan judul 2. Sistem menampilkan halaman pengajuan judul dan dosen pembimbing 3. Aktor mengisi formulir persetujuan pengajuan judul tesis dan dosen pembimbing 4. Aktor menekan tombol submit 5. Sistem menyimpan data formulir pengajuan judul tesis dan dosen pembimbing 6. Sistem menampilkan halaman pengajuan judul
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil menyetujui pengajuan judul tesis dan dosen pembimbing

4.7.19 Use Case Scenario Menyetujui Persetujuan Ujian Proposal

Use case scenario menyetujui persetujuan ujian proposal merupakan penjelasan alur dari proses dosen pembimbing dan kaprodi untuk menyetujui persetujuan ujian proposal. Pada Tabel 4.31 merupakan *use case scenario* menyetujui persetujuan ujian proposal.

Tabel 4.31 Use case scenario menyetujui persetujuan ujian proposal

Alur Proses untuk menyetujui persetujuan ujian proposal	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor dapat menyetujui persetujuan ujian proposal
Aktor	Dosen pembimbing dan Ketua Program Studi
Pre-Condition	Aktor telah login ke dalam sistem



Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu ujian proposal 2. Sistem menampilkan halaman ujian proposal 3. Aktor menekan tombol setuju pada ujian proposal yang ingin disetujui 4. Aktor menekan tombol ok untuk konfirmasi persetujuan 5. Sistem menyimpan data persetujuan ujian proposal 6. Sistem menampilkan pesan berhasil
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil menyetujui persetujuan ujian proposal

4.7.20 Use Case Scenario Menolak Persetujuan Ujian Proposal

Use case scenario menolak persetujuan ujian proposal merupakan penjelasan alur dari proses dosen pembimbing dan kaprodi untuk menolak persetujuan ujian proposal. Pada Tabel 4.32 merupakan *use case scenario* menolak persetujuan ujian proposal.

Tabel 4.32 Use case scenario menolak persetujuan ujian proposal

Alur Proses untuk menolak persetujuan ujian proposal	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor dapat menolak persetujuan ujian proposal
Aktor	Dosen pembimbing dan Ketua Program Studi
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu ujian proposal 2. Sistem menampilkan halaman ujian proposal 3. Aktor menolak persetujuan ujian proposal 4. Aktor menekan tombol submit 5. Sistem menyimpan data persetujuan ujian proposal 6. Sistem menampilkan pesan berhasil
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil menolak persetujuan ujian proposal

4.7.21 Use Case Scenario Menyetujui Persetujuan Seminar Hasil

Use case scenario menyetujui persetujuan seminar hasil merupakan penjelasan alur dari proses dosen pembimbing dan kaprodi untuk menyetujui persetujuan seminar hasil. Pada Tabel 4.33 merupakan *use case scenario* menyetujui persetujuan seminar hasil.



Tabel 4.33 Use case scenario menyetujui persetujuan ujian seminar hasil

Alur Proses untuk menyetujui persetujuan ujian seminar hasil	
Deskripsi	Use case ini menjelaskan bagaimana aktor dapat menyetujui persetujuan ujian seminar hasil
Aktor	Dosen pembimbing dan Ketua Program Studi
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu ujian seminar hasil 2. Sistem menampilkan halaman ujian seminar hasil 3. Aktor menyetujui persetujuan ujian seminar hasil 4. Aktor menekan tombol submit 5. Sistem menyimpan data persetujuan ujian seminar hasil 6. Sistem menampilkan pesan berhasil
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil menyetujui persetujuan ujian seminar hasil

4.7.22 Use Case Scenario Menolak Persetujuan Seminar Hasil

Use case scenario menolak persetujuan seminar hasil merupakan penjelasan alur dari proses dosen pembimbing dan kaprodi untuk menolak persetujuan seminar hasil. Pada Tabel 4.34 merupakan use case scenario menolak persetujuan seminar hasil.

Tabel 4.34 Use case scenario menolak persetujuan seminar hasil

Alur Proses untuk menolak persetujuan ujian seminar hasil	
Deskripsi	Use case ini menjelaskan bagaimana aktor dapat menolak persetujuan ujian seminar hasil
Aktor	Dosen pembimbing dan Ketua Program Studi
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu ujian seminar hasil 2. Sistem menampilkan halaman seminar hasil 3. Aktor menolak persetujuan seminar hasil 4. Aktor menekan tombol submit 5. Sistem menyimpan data penolakan seminar hasil 6. Sistem menampilkan pesan berhasil
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil menolak ujian seminar hasil



4.7.23 Use Case Scenario Menyetujui Persetujuan Ujian Tesis

Use case scenario menyetujui persetujuan ujian tesis merupakan penjelasan alur dari proses dosen pembimbing dan kaprodi untuk menyetujui persetujuan ujian tesis. Pada Tabel 4.35 merupakan *use case scenario* menyetujui persetujuan ujian tesis.

Tabel 4.35 Use case scenario menyetujui pengajuan persetujuan ujian tesis

Alur Proses untuk menyetujui persetujuan ujian tesis	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor dapat menyetujui persetujuan ujian tesis
Aktor	Dosen pembimbing dan Ketua Program Studi
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu ujian tesis 2. Sistem menampilkan halaman ujian tesis 3. Aktor menyetujui persetujuan ujian tesis 4. Aktor menekan tombol submit 5. Sistem menyimpan data persetujuan ujian tesis 6. Sistem menampilkan pesan berhasil
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil menyetujui persetujuan pengajuan ujian tesis

4.7.24 Use Case Scenario Menolak Persetujuan Ujian Tesis

Use case scenario menyetujui persetujuan seminar hasil merupakan penjelasan alur dari proses dosen pembimbing dan kaprodi untuk menolak persetujuan ujian tesis. Pada Tabel 4.36 merupakan *use case scenario* menolak persetujuan ujian tesis.

Tabel 4.36 Use case scenario menolak persetujuan ujian tesis

Alur Proses untuk menyetujui persetujuan ujian tesis	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor dapat menolak persetujuan ujian tesis
Aktor	Dosen pembimbing dan Ketua Program Studi
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu ujian tesis 2. Sistem menampilkan halaman ujian tesis 3. Aktor menolak persetujuan ujian tesis 4. Aktor menekan tombol submit 5. Sistem menyimpan data penolakan ujian tesis



	6. Sistem menampilkan pesan berhasil
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil menolak persetujuan ujian tesis

4.7.25 Use Case Scenario Menyetujui Pengajuan Ujian Proposal

Use case scenario menyetujui pengajuan ujian proposal merupakan penjelasan alur dari proses dosen pembimbing, dosen penguji dan kaprodi untuk menyetujui pengajuan ujian proposal. Pada Tabel 4.37 merupakan *use case scenario* menyetujui pengajuan ujian proposal.

Tabel 4.37 Use case scenario menyetujui pengajuan ujian proposal

Alur Proses untuk menyetujui pengajuan ujian proposal	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor menyetujui peganjuan ujian proposal
Aktor	Dosen pembimbing, dosen penguji dan ketua program studi
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu ujian proposal 2. Sistem menampilkan halaman pengajuan ujian proposal 3. Aktor tombol setuju pada pengajuan ujian proposal mahasiswa 4. Aktor menekan tombol submit 5. Sistem menyimpan data pengajuan ujian proposal 6. Sistem menampilkan halaman pengajuan ujian proposal
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil menyetujui pengajuan ujian proposal

4.7.26 Use Case Scenario Menyetujui Pengajuan Ujian Seminar Hasil

Use case scenario menyetujui pengajuan seminar hasil merupakan penjelasan alur dari proses dosen pembimbing, dosen penguji dan kaprodi untuk menyetujui pengajuan seminar hasil. Pada Tabel 4.38 merupakan *use case scenario* menyetujui pengajuan seminar hasil.



Tabel 4.38 Use case scenario menyetujui pengajuan ujian seminar hasil

Alur Proses untuk menyetujui pengajuan ujian seminar hasil	
Deskripsi	Use case ini menjelaskan bagaimana aktor menyetujui peganjuan ujian seminar hasil
Aktor	Dosen pembimbing, dosen penguji dan ketua program studi
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu ujian seminar hasil 2. Sistem menampilkan halaman ujian seminar hasil 3. Aktor menekan tombol setuju pada pengajuan ujian seminar hasil mahasiswa 4. Aktor menekan tombol submit 5. Sistem menyimpan data persetujuan ujian seminar hasil 6. Sistem menampilkan halaman pengajuan ujian seminar hasil
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil menyetujui pengajuan ujian seminar hasil

4.7.27 Use Case Scenario Menyetujui Pengajuan Ujian Tesis

Use case scenario menyetujui pengajuan ujian tesis merupakan penjelasan alur dari proses dosen pembimbing, dosen penguji dan kaprodi untuk menyetujui pengajuan ujian tesis. Pada Tabel 4.39 merupakan use case scenario menyetujui pengajuan ujian tesis.

Tabel 4.39 Use case scenario menyetujui pengajuan ujian tesis

Alur Proses untuk menyetujui pengajuan ujian tesis	
Deskripsi	Use case ini menjelaskan bagaimana aktor menyetujui peganjuan ujian tesis
Aktor	Dosen pembimbing, dosen penguji dan ketua program studi
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu pengajuan ujian tesis 2. Sistem menampilkan halaman pengajuan ujian tesis 3. Aktor menekan tombol setuju pada pengajuan ujian tesis mahasiswa 4. Aktor menekan tombol submit



	<ol style="list-style-type: none"> 5. Sistem menyimpan data pengajuan ujian seminar tesis 6. Sistem menampilkan halaman pengajuan ujian tesis
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil menyetujui pengajuan ujian tesis

4.7.28 Use Case Scenario Menentukan Dosen Pembimbing

Use case scenario menentukan dosen pembimbing merupakan penjelasan alur dari proses kaprodi untuk menentukan dosen pembimbing mahasiswa. Pada Tabel 4.40 merupakan *use case scenario* menentukan dosen pembimbing.

Tabel 4.40 Use case scenario menentukan dosen pembimbing

Alur Proses untuk menentukan dosen pembimbing	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor menentukan dosen pembimbing
Aktor	Ketua Program Studi
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu pengajuan judul 2. Sistem menampilkan halaman pengajuan judul dan dosen pembimbing 3. Aktor mengisi formulir pilih dosen pembimbing pada pengajuan judul dan dosen pembimbing mahasiswa 4. Aktor menekan tombol submit 5. Sistem menyimpan data formulir pilih dosen pembimbing 6. Sistem menampilkan halaman pengajuan judul dan dosen pembimbing
Alternative Flows	Tidak ada
Subflow	Tidak ada
Post-Conditions	Aktor berhasil menentukan dosen pembimbing

4.7.29 Use Case Scenario Menentukan Dosen Penguji

Use case scenario menentukan dosen penguji merupakan penjelasan alur dari proses kaprodi untuk menentukan dosen penguji ujian proposal atau ujian tesis bagi mahasiswa. Pada Tabel 4.41 merupakan *use case scenario* menentukan dosen penguji.



Tabel 4.41 Use case scenario menentukan dosen penguji

Alur Proses untuk menentukan dosen penguji	
Deskripsi	Use case ini menjelaskan bagaimana aktor menentukan dosen penguji ujian proposal atau ujian tesis
Aktor	Kaprodi
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu ujian proposal atau ujian tesis 2. Sistem menampilkan halaman ujian proposal atau ujian tesis 3. Aktor mengisi formulir pilih dosen penguji pada pengajuan ujian proposal atau ujian tesis mahasiswa 4. Aktor menekan tombol submit 5. Sistem menyimpan data formulir pilih dosen penguji 6. Sistem menampilkan halaman ujian proposal atau ujian tesis
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil menentukan dosen penguji ujian proposal dan ujian tesis

4.7.30 Use Case Scenario Menyetujui Menjadi Dosen Pembimbing

Use case scenario menyetujui menjadi dosen pembimbing merupakan penjelasan alur dari proses dosen pembimbing untuk menyetujui menjadi dosen pembimbing. Pada Tabel 4.42 merupakan use case scenario menentukan dosen penguji.

Tabel 4.42 Use case scenario menyetujui menjadi dosen pembimbing

Alur Proses untuk menyetujui menjadi dosen pembimbing	
Deskripsi	Use case ini menjelaskan bagaimana aktor menyetujui menjadi dosen pembimbing
Aktor	Dosen pembimbing
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu pengajuan judul 2. Sistem menampilkan halaman pengajuan judul dan dosen pembimbing



	<ol style="list-style-type: none"> 3. Aktor menekan tombol setuju pada pengajuan judul dan dosen pembimbing mahasiswa 4. Aktor menekan tombol submit 5. Sistem menyimpan data formulir persetujuan menjadi dosen pembimbing 6. Sistem menampilkan halaman pengajuan judul dan dosen pembimbing
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil menyetujui menjadi dosen pembimbing

4.7.31 Use Case Scenario Melihat Data Tesis

Use case scenario melihat data tesis penjelasan alur dari pegawai dan mahasiswa melihat data tesis milik mahasiswa. Pada Tabel 4.43 merupakan *use case scenario* melihat data tesis.

Tabel 4.43 Use case scenario melihat data tesis

Alur Proses untuk melihat data tesis	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihat data tesis
Aktor	Pegawai dan Mahasiswa
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu mahasiswa 2. Aktor menekan nama mahasiswa 3. Sistem menampilkan halaman profil mahasiswa 4. Aktor menekan judul tesis 5. Sistem menampilkan data tesis
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil melihat data tesis mahasiswa

4.7.32 Use Case Scenario Melihat Data Pengajuan Judul dan Dosen Pembimbing

Use case scenario melihat data pengajuan judul dan dosen pembimbing merupakan penjelasan alur dari pegawai dan mahasiswa melihat data pengajuan judul milik mahasiswa. Pada Tabel 4.44 merupakan *use case scenario* melihat data pengajuan judul dan dosen pembimbing.



Tabel 4.44 Use case scenario melihat data pengajuan judul dan dosen pembimbing

Alur Proses untuk melihat data pengajuan judul dan dosen pembimbing	
Deskripsi	Use case ini menjelaskan bagaimana aktor melihat data pengajuan judul dan dosen pembimbing
Aktor	Pegawai dan Mahasiswa
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu mahasiswa 2. Aktor menekan nama mahasiswa 3. Sistem menampilkan halaman profil mahasiswa 4. Aktor menekan judul pengajuan dan dosen pembimbing 5. Sistem menampilkan data pengajuan judul dan dosen pembimbing
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil melihat data pengajuan judul dan dosen pembimbing

4.7.33 Use Case Scenario Melihat Data Ujian Proposal

Use case scenario melihat data ujian proposal merupakan penjelasan alur dari pegawai dan mahasiswa melihat data ujian proposal milik mahasiswa. Pada Tabel 4.45 merupakan use case scenario melihat data pengajuan ujian proposal.

Tabel 4.45 Use case scenario melihat data ujian proposal

Alur Proses untuk melihat data ujian proposal	
Deskripsi	Use case ini menjelaskan bagaimana aktor melihat data ujian proposal
Aktor	Pegawai dan Mahasiswa
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu mahasiswa 2. Aktor menekan nama mahasiswa 3. Sistem menampilkan halaman profil mahasiswa 4. Aktor menekan judul tesis 5. Sistem menampilkan data tesis 6. Aktor memilih menu ujian proposal 7. Sistem menampilkan data ujian proposal
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil melihat data ujian proposal mahasiswa



4.7.34 Use Case Scenario Melihat Data Seminar Hasil

Use case scenario melihat data seminar hasil merupakan penjelasan alur dari pegawai dan mahasiswa melihat data seminar hasil milik mahasiswa. Pada Tabel 4.45 merupakan *use case scenario* melihat data pengajuan seminar hasil.

Tabel 4.46 Use case scenario melihat data seminar hasil

Alur Proses untuk melihat data seminar hasil	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihat data seminar hasil
Aktor	Pegawai dan Mahasiswa
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu mahasiswa 2. Aktor menekan nama mahasiswa 3. Sistem menampilkan halaman profil mahasiswa 4. Aktor menekan judul tesis 5. Sistem menampilkan data tesis 6. Aktor memilih menu seminar hasil 7. Sistem menampilkan data seminar hasil
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil melihat data seminar hasil mahasiswa

4.7.35 Use Case Scenario Melihat Data Ujian Tesis

Use case scenario melihat data ujian tesis merupakan penjelasan alur dari pegawai dan mahasiswa melihat data ujian tesis milik mahasiswa. Pada Tabel 4.47 merupakan *use case scenario* melihat data ujian tesis.

Tabel 4.47 Use case scenario melihat data ujian tesis

Alur Proses untuk melihat data ujian tesis	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihat data ujian tesis
Aktor	Pegawai dan Mahasiswa
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu mahasiswa 2. Aktor menekan nama mahasiswa 3. Sistem menampilkan halaman profil mahasiswa 4. Aktor menekan judul tesis 5. Sistem menampilkan data tesis



	6. Aktor memilih menu ujian tesis 7. Sistem menampilkan data ujian tesis
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil melihat data ujian tesis mahasiswa

4.7.36 Use Case Scenario Melihat Profil

Use case scenario melihat profil merupakan penjelasan alur dari pegawai dan mahasiswa melihat profilnya sendiri. Pada Tabel 4.48 merupakan *use case scenario* melihat profil.

Tabel 4.48 Use case scenario melihat profil

Alur Proses untuk melihat profil	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihat profilnya sendiri
Aktor	Pegawai dan Mahasiswa
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	1. Aktor memilih menu profil 2. Sistem menampilkan halaman profil
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil melihat halaman profil

4.7.37 Use Case Scenario Melihat Profil Mahasiswa

Use case scenario melihat profil mahasiswa merupakan penjelasan alur dari pegawai melihat profilnya mahasiswa. Pada Tabel 4.49 merupakan *use case scenario* melihat profil mahasiswa.

Tabel 4.49 Use case scenario melihat profil mahasiswa

Alur Proses untuk melihat profil mahasiswa	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihat profil mahasiswa
Aktor	Pegawai
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	1. Aktor memilih menu mahasiswa 2. Aktor menekan nama mahasiswa 3. Sistem menampilkan halaman profil mahasiswa
Alternative Flows	Tidak ada



Post-Conditions	Aktor berhasil melihat halaman profil mahasiswa
------------------------	---

4.7.38 Use Case Scenario Melihat Profil Dosen

Use case scenario melihat profil dosen merupakan penjelasan alur dari kaprodi untuk melihat profil dosen. Pada Tabel 4.50 merupakan *use case scenario* melihat profil dosen.

Tabel 4.50 Use case scenario melihat profil dosen

Alur Proses untuk melihat profil dosen	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihat profil dosen
Aktor	Kaprodi
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih menu dosen 2. Aktor menekan nama dosen 3. Sistem menampilkan halaman profil dosen
Alternative Flows	Tidak ada
Post-Conditions	Aktor berhasil melihat halaman profil dosen

4.7.39 Use Case Scenario Mengubah Password

Use case scenario mengubah *password* penjelasan alur dari pegawai dan mahasiswa untuk mengubah *password*. Pada Tabel 4.51 merupakan *use case scenario* mengubah *password*.

Tabel 4.51 Use case scenario mengubah password

Alur Proses untuk mengubah password	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor dapat mengubah <i>password</i> nya
Aktor	Pegawai dan mahasiswa
Pre-Condition	Aktor telah login ke dalam sistem
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor menekan tombol ganti <i>password</i> pada <i>dropdown</i> nama pengguna. 2. Sistem menampilkan formulir ganti <i>password</i>. 3. Pengguna mengisi kolom <i>password</i> baru dan kolom ulangi <i>password</i> baru. 4. Pengguna menekan tombol <i>submit</i>. 5. Sistem menampilkan pesan <i>password</i> berhasil diubah.

**Alternative Flows**

3.a.1 Pengguna mengisi kolom password baru dan kolom ulang password baru dengan isi yang berbeda.

3.a.2 Sistem menampilkan pesan kesalahan bahwa kolom password baru dan ulang password baru tidak sama.

Post-Conditions

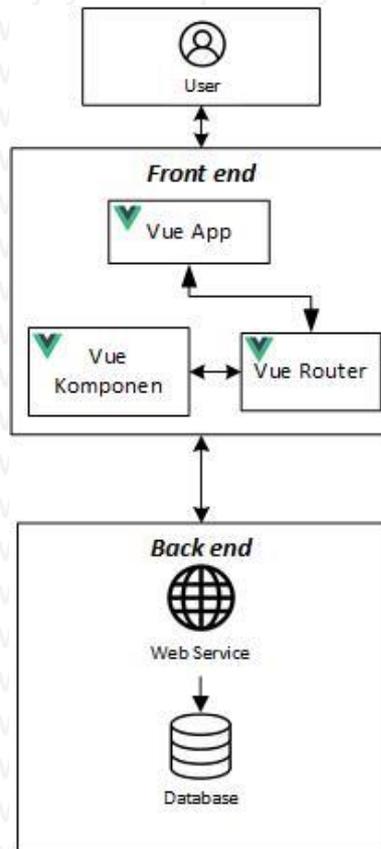
Aktor berhasil melihat halaman profil mahasiswa



BAB 5 PERANCANGAN

Pada bab ini akan dibahas mengenai perancangan dari Sistem Informasi Manajemen Tesis yang akan dikembangkan. Perancangan dibuat berdasarkan kebutuhan yang telah didapatkan pada bab analisis kebutuhan.

5.1 Pemodelan Arsitektur



Gambar 5.1 Pemodelan arsitektur

Pemodelan arsitektur merupakan penggambaran struktur dari perangkat lunak. Bagian *front end* merupakan bagian sistem yang berhubungan langsung dengan pengguna. Sedangkan pada bagian *back end* merupakan bagian sistem yang menangani logika sistem dan basis data. Gambar 5.1 merupakan gambar dari pemodelan arsitektur.

Bagian *front end* merupakan bagian yang berinteraksi langsung dengan pengguna. *Front end* ini dibangun menggunakan konsep *single page application* (SPA). Untuk menerapkan konsep SPA, maka sistem dibangun menggunakan *framework* Vue.js. SPA ini terdiri dari beberapa bagian yaitu *vue app*, *vue router* dan *vue component*. *Vue app* merupakan bagian utama yang berperan menyimpan *instance* dari *vue.js* dan menghubungkan dengan *vue router*. *Vue router* merupakan bagian yang mengatur komponen mana yang akan ditampilkan kepada pengguna. *Vue component* merupakan bagian halaman yang



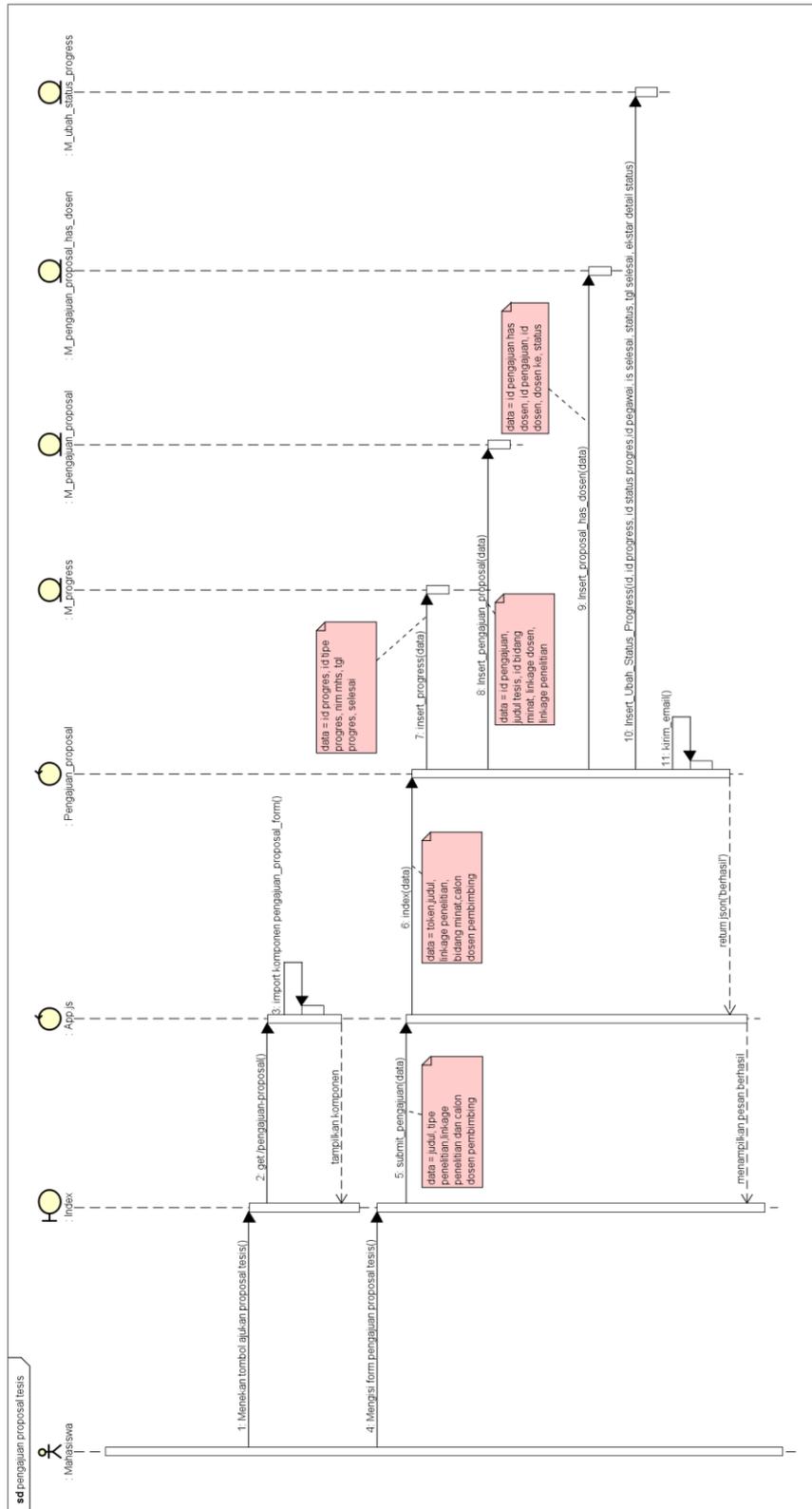
merepresentasikan tiap halaman yang ada pada sistem. Untuk mendapatkan data yang akan ditampilkan ke pengguna, bagian *front end* akan mengirimkan *request* ke *web service* pada bagian *back end* dengan *ajax request*.

Bagian *back end* merupakan bagian yang bertugas mengolah data dan menghubungkan sistem dengan basis data. Back end ini dibangun menggunakan konsep *service oriented architecture* (SOA). Untuk menerapkan konsep SOA, maka dibuat *web service* yang bertugas menanggapi permintaan dari bagian *front end* dan memberikan respon untuk diterima kembali oleh bagian *frontend*. *Web service* pada sistem yang dikembangkan memanfaatkan *framework codeignier*. *Web service* yang mendapatkan *request* dari bagian *front end* akan memproses *request* tersebut dan nantinya akan dikembalikan dalam format json.

5.2 Pemodelan *Sequence Diagram*

5.2.1 *Sequence Diagram* Mengisi Formulir Pengajuan Judul dan Dosen Pembimbing

Proses pengajuan judul dan dosen pembimbing dimulai dengan mahasiswa menekan tombol ajukan judul. Setelah menekan tombol tersebut, *controller* App.js akan menampilkan komponen *pengajuan_proposal_form* yang memapilkan form pengajuan proposal ke mahasiswa. Selanjutnya mahasiswa mengisi data judul dan tipe penelitian proposal serta calon dosen pembimbing pada formulir tersebut. Setelah mahasiswa menekan tombol submit, akan memanggil fungsi *submit_pengajuan* pada komponen *pengajuan_proposal_form*. Fungsi *submit_pengajuan* tersebut selanjutnya mengirim data pengajuan proposal ke fungsi *index* pada *controller* Pengajuan_proposal. Pada *controller* Pengajuan_proposal akan memanggil fungsi *insert_progres* pada model M_progres untuk melakukan *insert* progres baru, fungsi *insert_pengajuan_proposal* pada model M_pengajuan_proposal untuk *insert* pengajuan proposal baru, fungsi *insert_pengajuan_proposal_has_dosen* pada model M_pengajuan_proposal_has_dosen untuk *insert* data pegawai yang menjadi calon pembimbing pada pengajuan proposal tersebut dan memanggil fungsi *insert_ubah_status_progres* pada model M_ubah_status_progres untuk *insert* status pengajuan proposal selanjutnya. Selanjutnya *controller* Pengajuan_proposal akan mengirim data json yang berisi pesan berhasil dan view *index* menampilkan pesan pengajuan proposal berhasil. Pada Gambar 5.2 merupakan *sequence diagram* mengisi formulir pengajuan judul dan dosen pembimbing.

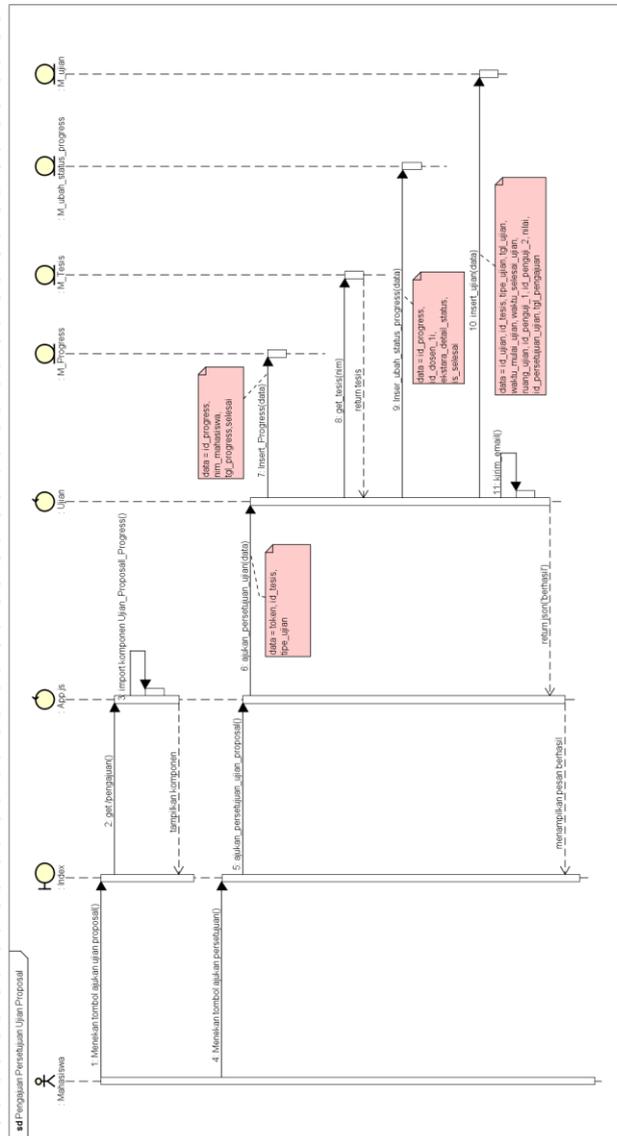


Gambar 5.2 Sequence diagram mengisi formulir pengajuan judul dan dosen pembimbing



5.2.2 Sequence Diagram Mengisi Formulir Persetujuan Ujian Proposal

Proses menyetujui mengisi formulir persetujuan ujian proposal dimulai dengan mahasiswa menekan menu tombol ajukan ujian proposal. Setelah menekan tombol tersebut, *controller* App.js akan menampilkan komponen ujian_proposal_progress. Selanjutnya komponen ujian_proposal_progress akan menampilkan formulir persetujuan ujian proposal yang bisa diisi oleh mahasiswa. Mahasiswa selanjutnya dapat menekan tombol ajukan persetujuan untuk mengajukan persetujuan ujian proposal. Setelah itu, komponen ujian_proposal_form akan memanggil fungsi ajukan_persetujuan_ujian_proposal yang didalamnya memanggil fungsi ajukan_persetujuan_ujian pada controller ujian beserta data token mahasiswa, id_tesis dan tipe_ujian. Fungsi ajukan_persetujuan_ujian tersebut selanjutnya memanggil fungsi insert_progres pada model M_progres dengan parameter id_progres, nim_mahasiswa dan tgl_progres yang berfungsi untuk menambahkan data progres mahasiswa. Selain itu fungsi ajukan_persetujuan_ujian juga memanggil fungsi get_tesis dari model M_tesis dengan parameter nim_mahasiswa untuk mendapatkan data tesis mahasiswa. Setelah mendapat data tesis mahasiswa, akan dipanggil fungsi insert_ubah_status_progres pada model M_ubah_status_progres dengan parameter id_progres, id_pegawai dan is_selesai. Fungsi insert_ubah_status_progres tersebut digunakan untuk meneruskan status progres dari persetujuan proposal. Fungsi lain yang dipanggil adalah fungsi insert_ujian pada model M_ujian dengan parameter id_tesis, tipe_ujian, tgl_ujian, waktu_mulai_ujian, waktu_selesai_ujian, ruang_ujian, id_penguji_1, id_penguji_2 dan nilai. Fungsi insert_ujian tersebut digunakan untuk melakukan penambahan data ujian yang diajukan oleh mahasiswa. Pada Gambar 5.3 merupakan *sequence diagram* mengisi formulir persetujuan ujian proposal.



Gambar 5.3 Sequence diagram mengisi formulir persetujuan ujian proposal

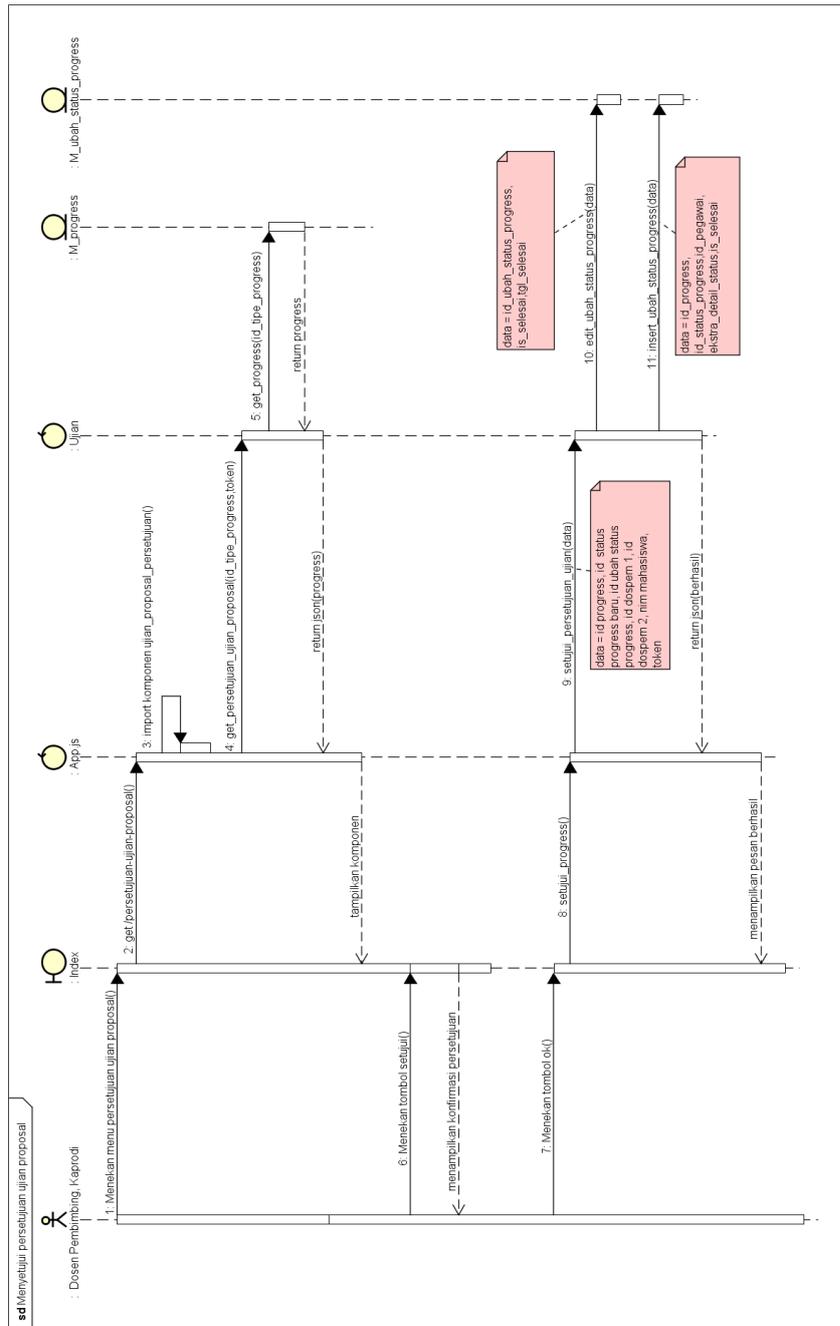
5.2.3 Sequence Diagram Menyetujui Persetujuan Ujian Proposal

Proses menyetujui persetujuan ujian proposal dimulai dengan aktor (dosen pembimbing dan kaprodi) menekan menu ujian proposal. Setelah menekan tombol tersebut, *controller* App.js akan menampilkan komponen ujian_proposal_persetujuan. Selanjutnya komponen ujian_proposal_persetujuan memanggil fungsi *get_persetujuan_ujian* proposal yang mengirim data token dan *id_tipe_progres* ke fungsi *get_progres_ujian* pada *controller* ujian. Fungsi *get_progres_ujian* akan memanggil fungsi *get_progres* pada model *M_progres* dengan parameter *id_tipe_progres* untuk mendapatkan data progres ujian proposal. Setelah data progres ujian proposal didapatkan, data tersebut dikirimkan ke komponen ujian_proposal_persetujuan dalam bentuk



json.Selanjutnya komponen ujian_proposal_persetujuan akan ditampilkan ke *view index* oleh *controller* App.js.

Aktor selanjutnya menekan tombol setuju pada ujian proposal yang ingin disetujui. *View index* selanjutnya akan menampilkan pesan konfirmasi pada aktor apakah benar ingin menyetujui ujian proposal tersebut. Bila aktor menekan tombol ok, maka akan memanggil fungsi *setujui_progres* pada komponen *ujian_proposal_persetujuan* yang mengirim data persetujuan ujian proposal ke fungsi *setujui_persetujuan_ujian* pada *controller* ujian.Fungsi tersebut akan memanggil 2 fungsi pada model *M_ubah_status_progres* yaitu, *edit_ubah_status_progres* dan *insert_ubah_status_progres*. Fungsi *edit_ubah_status_progres* akan memperbarui data status progres berdasarkan *id_ubah_status_progres*. Sedangkan fungsi *insert_ubah_status_progres* untuk meneruskan data persetujuan tersebut ke tahap berikutnya dengan data *id_progres*, *id_status_progres*, *id_pegawai*, *ekstra_detail_status* dan *is_selesai*. Setelah dua fungsi tersebut dijalankan, *controller* ujian akan mengirim data json yang berisi pesan berhasil ke *controller* komponen *ujian_proposal_persetujuan* untuk menampilkan pesan berhasil menyetujui persetujuan ujian proposal. Pada Gambar 5.4 merupakan *sequence diagram* menyetujui persetujuan ujian proposal.



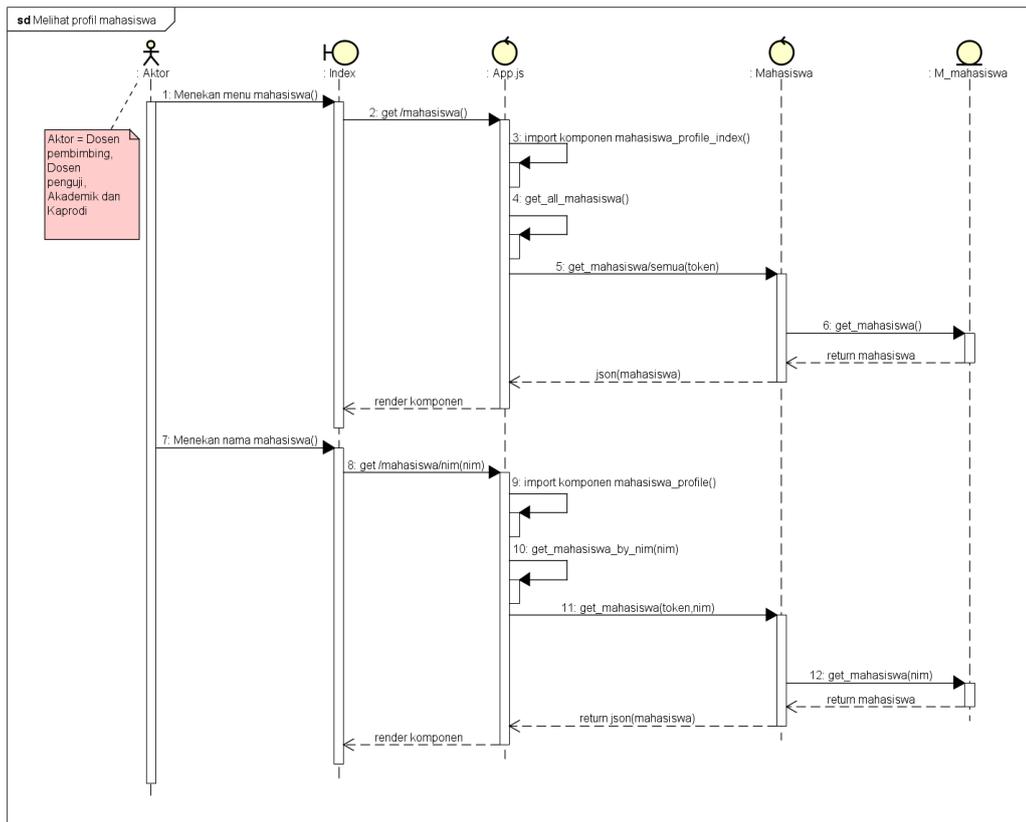
Gambar 5.4 Sequence diagram menyetujui persetujuan ujian proposal

5.2.4 Sequence Diagram Melihat Profil Mahasiswa

Proses melihat profil mahasiswa dimulai dengan aktor (dosen pembimbing, dosen penguji, akademik dan kaprodi) menekan menu mahasiswa. Setelah menekan menu mahasiswa, controller App.js akan mengimpor controller komponen_mahasiswa_index. Controler App.js akan menunggu semua proses yang ada pada controller komponen mahasiswa_profil_index selesai dijalankan sebelum menampilkan ke view index. Pada controller komponen mahasiswa_index akan memanggil fungsi get_all_mahasiswa untuk mendapatkan



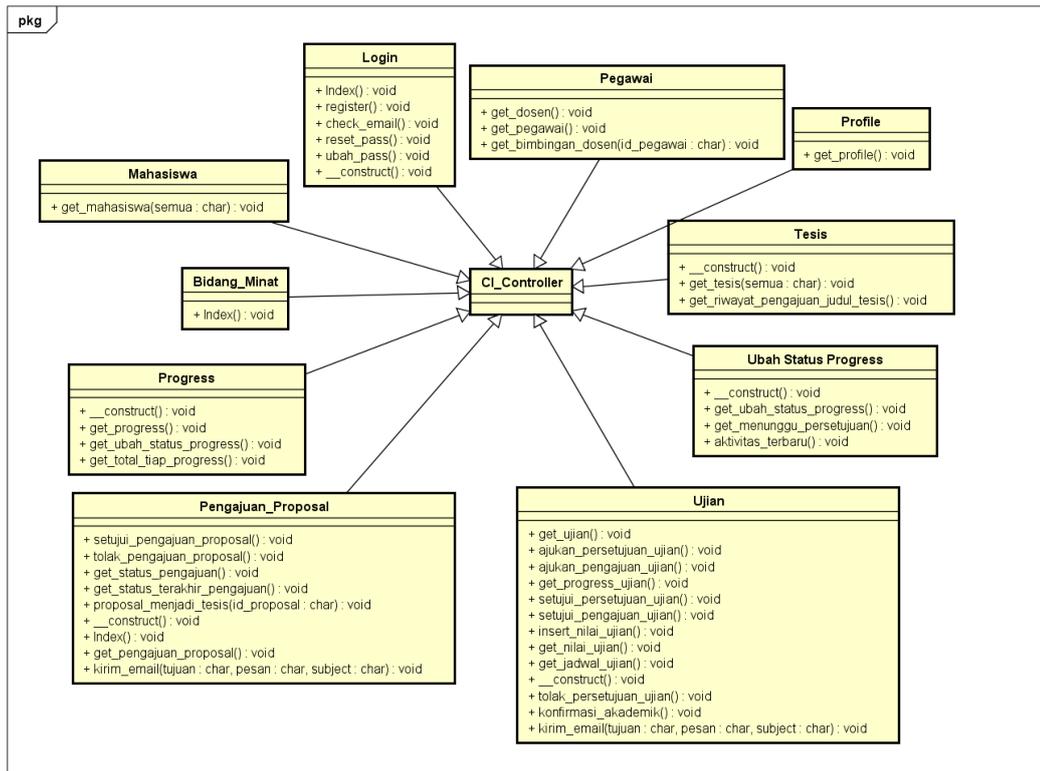
daftar mahasiswa yang ada. Fungsi tersebut selanjutnya memanggil fungsi `get_mahasiswa` pada controller mahasiswa yang memanggil fungsi `get_mahasiswa` pada model `M_mahasiswa` untuk mendapatkan data mahasiswa yang ada. Setelah data mahasiswa berhasil didapatkan dan controller komponen `_mahasiswa_index` sudah ditampilkan pada view `index`, aktor dapat menekan nama mahasiswa untuk melihat profilnya. Pada Gambar 5.5 merupakan sequence diagram melihat profil mahasiswa.



Gambar 5.5 Sequence diagram melihat profil mahasiswa

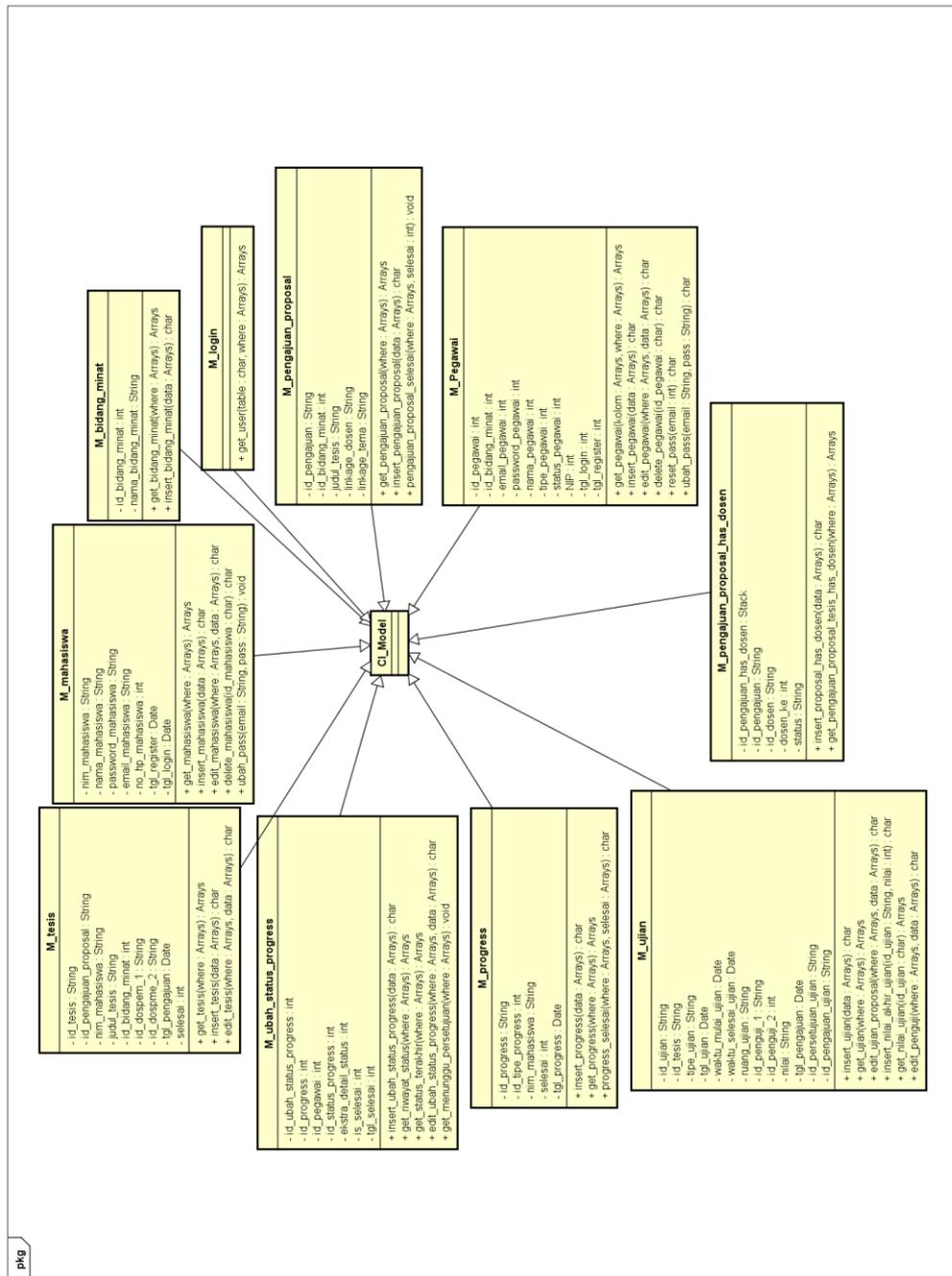
5.3 Pemodelan Class Diagram

Pemodelan *class diagram* dalam penelitian ini menggambarkan *controller* dan *model* berdasarkan implementasi *framework* codeigniter. Pada Gambar 5.6 menunjukkan *class diagram controller* dari Sistem Informasi Manajemen Tesis.



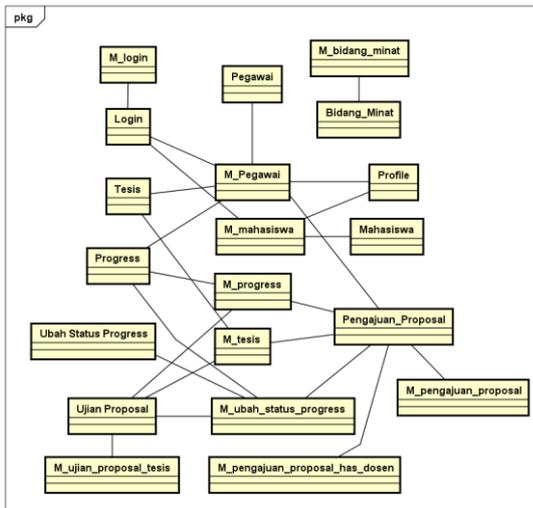
Gambar 5.6 Class diagram controller

Pada Gambar 5.7 merupakan class diagram model dari Sistem Informasi Manajemen Tesis.



Gambar 5.7 Class diagram model

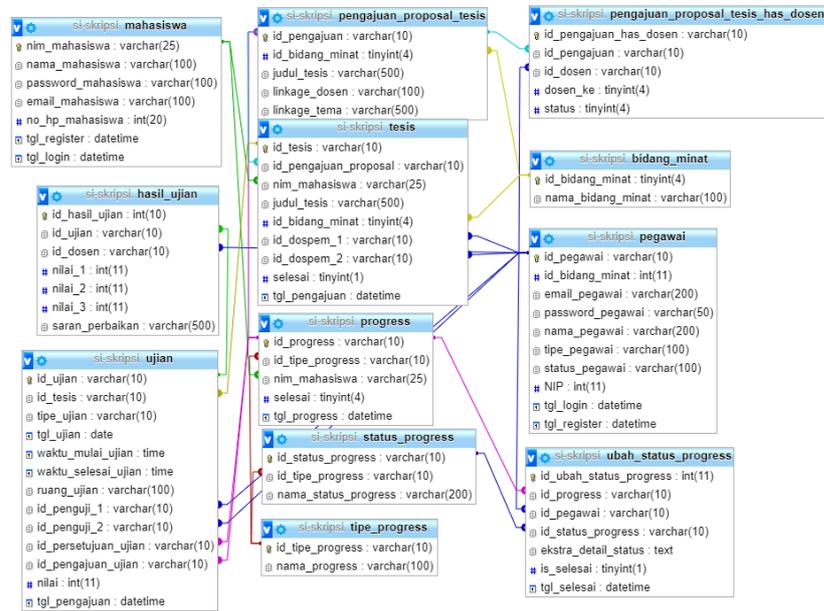
Gambar 5.8 merupakan class diagram keseluruhan yang menunjukkan hubungan antara model dan controller pada sistem informasi manajemen tesis. Sebuah controller dapat memiliki hubungan dengan lebih dari satu model



Gambar 5.8 Class diagram keseluruhan

5.4 Physical Data Model

Physical data model digunakan untuk merepresentasikan rancangan basis data. Physical data model mendefinisikan semua struktur tabel, termasuk nama kolom, tipe data kolom, primary key, foreign key, dan relasi antar tabel. Gambar 5.9 merupakan rancangan physical data model dari sistem informasi manajemen tesis.



Gambar 5.9 Physical data model

5.5 Perancangan Komponen

Perancangan komponen menjelaskan struktur dari data atau algoritme yang digunakan dalam suatu komponen perangkat lunak. Di dalam perancangan komponen ini hanya membahas tentang algoritme pengajuan judul dan dosen



pembimbing. Perancangan komponen ini dibagi menjadi 2 yaitu perancangan komponen *front end* dan perancangan komponen *back end*. Perancangan komponen *front end* menjelaskan algoritme pengajuan judul dan dosen pembimbing yang ada pada sistem informasi manajemen tesis. Perancangan komponen *back end* menjelaskan algoritme pengajuan judul dan dosen pembimbing yang ada pada *web service* sistem informasi manajemen tesis.

5.5.1 Perancangan Komponen *Front End*

Dalam perancangan komponen *front end* ini menjelaskan tentang algoritme pengajuan judul dan dosen pembimbing pada sistem informasi manajemen tesis yang akan dikembangkan. Algoritme pengajuan judul dan dosen pembimbing ini melibatkan *main.js*, *route.js*, *vue* komponen *pengajuan_proposal_index*, *vue* komponen *pengajuan_proposal_list* dan *vue* komponen *pengajuan_proposal_form*.

5.5.1.1 *Main.js*

Perancangan komponen *main.js* merupakan *file* utama dari SPA yang akan dibangun. Pada *file* tersebut terdapat algoritme untuk melakukan instansiasi *vue*, *vue router* dan komponen *route*. Pada Tabel 5.1 merupakan algoritme dari *file main.js*.

Tabel 5.1 Perancangan komponen *main.js*

1	Mulai
2	Mengimpor <i>vue</i> , <i>vue router</i> , komponen <i>app</i> dan komponen <i>route</i>
3	Instansiasi <i>vue router</i> dengan properti komponen <i>route</i>
4	Memanggil fungsi <i>use()</i> untuk mendaftarkan <i>vue router</i>
5	Instansiasi <i>vue</i> dengan properti <i>el</i> , <i>router</i> dan <i>render</i>
6	Selesai

5.5.1.2 *Route.js*

Perancangan komponen *route.js* merupakan *file* yang menyimpan rute dari SPA yang akan dibangun. Pada *file* tersebut terdapat algoritme untuk menyimpan rute pengajuan proposal. Pada Tabel 5.2 merupakan algoritme dari *file route.js*.

Tabel 5.2 Perancangan komponen *route.js*

1	Mulai
2	Mengimpor komponen <i>pengajuan_proposal_index</i> , <i>pengajuan_proposal_form</i> dan <i>pengajuan_proposal_lis</i>
3	Membuat variabel <i>route</i> dengan tipe <i>array</i> yang memiliki atribut <i>path</i> , <i>component</i> dan <i>children</i>
4	Mengisi nilai atribut <i>path</i> pada <i>array route</i> dengan nilai <i>pengajuan-judul</i>
5	Mengisi nilai atribut <i>component</i> pada <i>array route</i> dengan komponen <i>pengajuan_proposal_index</i>
6	Mengisi nilai atribut <i>children</i> pada <i>array route</i> dengan <i>array route</i> masing-masing halaman pengajuan judul



7	Mengekspor variabel route
8	Selesai

5.5.1.3 Vue komponen pengajuan_proposal_index

Perancangan komponen pengajuan_proposal_index merupakan *file* yang menyimpan tampilan saat pengguna mengakses /pengajuan-judul. Pada Tabel 5.3 merupakan algoritme dari komponen pengajuan_proposal_index.

Tabel 5.3 Perancangan komponen pengajuan_proposal_index

1	Mulai
2	Membuat tag template yang didalamnya terdapat tag router-view
3	Membuat variabel route dengan tipe array yang memiliki atribut path, component dan children
4	Membuat tag script yang didalamnya mengekspor variabel name dengan nilai PengajuanProposalIndex
5	Selesai

5.5.1.4 Vue komponen pengajuan_proposal_list

Perancangan komponen pengajuan_proposal_list merupakan *file* yang menyimpan tampilan daftar pengajuan proposal saat pengguna mengakses /pengajuan-judul. Pada Tabel 5.4 merupakan algoritme dari komponen pengajuan_proposal_list.

Tabel 5.4 Perancangan komponen pengajuan_proposal_list

1	Mulai
2	Membuat tag template yang didalamnya terdapat tag-tag untuk membuat tampilan daftar pengajuan proposal
3	Membuat tag script
4	Mengimpor komponen PengajuanProposalCard dan ModalPilihDosen
5	Memanggil fungsi ekspor yang berisi variabel pendukung komponen pengajuan_proposal_list
6	Selesai

5.5.1.5 Vue komponen pengajuan_proposal_form

Perancangan komponen pengajuan_proposal_form merupakan *file* yang menyimpan tampilan form pengajuan judul dan dosen pembimbing saat pengguna mengakses /pengajuan-proposal. Pada Tabel 5.5 merupakan algoritme dari komponen pengajuan_proposal_form.

Tabel 5.5 Pernacangan komponen pengajuan_proposal_form

1	Mulai
2	Membuat tag template yang didalamnya terdapat tag-tag untuk membuat tampilan form pengajuan proposal
3	Membuat tag script
4	Mengimpor komponen ModalPilihDosen



5	Memanggil fungsi ekspor yang berisi variabel pendukung komponen pengajuan_proposal_form
6	Selesai

5.5.2 Perancangan Komponen *Back End*

Dalam perancangan komponen *back end* ini menjelaskan tentang algoritme pengajuan judul dan dosen pembimbing pada web service yang akan dikembangkan. Algoritme pengajuan judul dan dosen pembimbing ini melibatkan *controller* pengajuan_proposal, *model* m_pengajuan_proposal untuk menyimpan data pengajuan proposal dan *model* m_pengajuan_proposal_has_dosen untuk menyimpan data dosen yang dipilih menjadi calon pembimbing dan pembimbing dari pengajuan proposal.

5.5.2.1 Controller *Pengajuan_Proposal (Fungsi Index)*

Perancangan komponen fungsi *index()* merupakan algoritme dari sistem untuk menyimpan data pengajuan proposal, data progres serta data calon dosen pembimbing. Fungsi *index()* termasuk di dalam *controller* pengajuan_proposal. Tabel 5.6 merupakan algoritme dari fungsi *index()*.

Tabel 5.6 Perancangan komponen fungsi *index()*

1	Mulai
2	Memanggil model M_pengajuan_proposal, M_pengajuan_proposal_has_dosen, M_ubah_status_progres, dan M_progres
3	Deklarasi variabel token yang menyimpan masukan token dari pengguna
4	Deklarasi variabel data yang menyimpan data pengajuan judul dan dosen pembimbing
5	Deklarasi variabel cek_login untuk menyimpan hasil dari pengecekan token
6	Jika variabel cek_login bernilai tidak sama dengan gagal
7	Menyimpan data progres ke dalam array progres
8	Memanggil fungsi dari model M_progres untuk menyimpan data progres
9	Menyimpan data pengajuan ke dalam array pengajuan
10	Memanggil fungsi dari model M_pengajuan_proposal untuk menyimpan data pengajuan
11	Perulangan sebanyak 5 kali yang memanggil fungsi dari model M_pengajuan_proposal_has_dosen untuk menyimpan data calon dosen pembimbing
12	Menyimpan data ubah status progres ke dalam array ubah
13	Memanggil fungsi dari model M_ubah_status_progres untuk menyimpan data ubah status progres
14	Mengirim email notifikasi ke akademik
14	Menyimpan data status pada array status
15	Mengembalikan variabel status dalam format json
16	Selesai



5.5.2.2 Model *M_Progres (Fungsi insert_progres)*

Perancangan komponen fungsi `insert_progres` merupakan algoritme dari sistem untuk menyimpan data progres ke dalam basis data. Fungsi ini berfungsi untuk menyimpan data progres perkembangan pengerjaan tesis mahasiswa. Tabel 5.7 merupakan algoritme dari fungsi `insert_progres()`.

Tabel 5.7 Perancangan komponen fungsi `insert_progres()`

1	Mulai
2	Membuat objek dari kelas <code>M_pengajuan_progres</code>
3	Menyimpan data id progres pada properti objek id progres
4	Menyimpan data id tipe progres pada properti objek id tipe progres
5	Menyimpan data nim mahasiswa pada properti objek nim mahasiswa
6	Menyimpan data objek progres ke basis data
7	Jika berhasil
8	Mengembalikan nilai berhasil`
9	Jika gagal
10	Mengembalikan nilai gagal
11	Selesai

5.5.2.3 Model *M_Pengajuan_Proposal (Fungsi insert_pengajuan_proposal)*

Perancangan komponen fungsi `insert_pengajuan_proposal()` merupakan algoritme dari sistem untuk menyimpan data pengajuan proposal ke dalam basis data. Fungsi `insert_pengajuan_proposal ()` termasuk di dalam *model* `m_pengajuan_proposal`. Tabel 5.8 merupakan algoritme dari fungsi `insert_pengajuan_proposal()`.

Tabel 5.8 Perancangan komponen fungsi `insert_pengajuan_proposal()`

1	Mulai
2	Membuat objek dari <code>M_pengajuan_proposal</code>
3	Menyimpan nilai id pengajuan pada properti objek id_pengajuan
4	Menyimpan nilai judul tesis pada properti objek judul_tesis
5	Menyimpan nilai id bidang minat pada properti objek id_bidang_minat
6	Menyimpan nilai selesai pada properti objek selesai
7	Menyimpan data objek pengajuan judul ke basis data
8	Memanggil fungsi dari model <code>M_progres</code> untuk menyimpan data progres
9	Jika berhasil
10	Mengembalikan nilai berhasil`
11	Jika gagal
12	Mengembalikan nilai gagal
13	Selesai



5.5.2.4 Model *M_pengajuan_proposal_has_dosen* (Fungsi *insert_proposal_has_dosen()*)

Perancangan komponen fungsi *insert_proposal_has_dosen()* merupakan algoritme dari sistem untuk menyimpan data calon dosen pembimbing dari pengajuan proposal ke dalam basis data. Fungsi *insert_proposal_has_dosen()* termasuk di dalam *model m_pengajuan_proposal_has_dosen*. Tabel 5.9 merupakan algoritme dari fungsi *insert_proposal_has_dosen ()*.

Tabel 5.9 Perancangan komponen fungsi *insert_proposal_has_dosen()*

1	Mulai
2	Membuat objek dari <i>M_pengajuan_proposal_has_dosen</i>
3	Menyimpan nilai <i>id_pengajuan_has_dosen</i> pada properti objek <i>id_pengajuan_has_dosen</i>
4	Menyimpan nilai <i>id_pengajuan</i> pada properti objek <i>id_pengajuan</i>
5	Menyimpan nilai <i>id_dosen</i> pada properti objek <i>id_dosen</i>
6	Menyimpan nilai <i>dosen_ke</i> pada properti objek <i>dosen_ke</i>
7	Menyimpan nilai <i>status</i> ke properti objek <i>status</i>
8	Menyimpan data objek basis data
9	Jika berhasil menyimpan data
10	Mengembalikan nilai berhasil`
11	Jika gagal
12	Mengembalikan nilai gagal
13	Selesai

5.5.2.5 Model *M_ubah_status_progres* (Fungsi *insert_ubah_status_progres()*)

Perancangan komponen fungsi *insert_ubah_status_progres ()* merupakan algoritme dari sistem untuk menyimpan perubahan status porgres ke dalam basis data. Fungsi *insert_ubah_status_progres()* termasuk di dalam *model m_ubah_status_progres*. Tabel 5.10 merupakan algoritme dari fungsi *insert_ubah_status_progres()*.

Tabel 5.10 Perancangan komponen fungsi *insert_ubah_status_progres()*

1	Mulai
2	Jika indeks <i>id_pegawai</i> dari array parameter sama dengan <i>id_pegawai_akademik</i> maka nilia variabel <i>id_pegawai</i> sama dengan <i>id_pegawai_pengguna_akademik</i>
3	Jika indeks <i>id_pegawai</i> dari array parameter sama dengan <i>id_pegawai_kaprodi</i> maka nilia variabel <i>id_pegawai</i> sama dengan <i>id_pegawai_pengguna_kaprodi</i>
4	Jika yang lain maka nilai variabel <i>id_pegawai</i> sama dengan nilai indeks <i>id_pegawai</i>
5	Membuat objek dari kelas <i>M_ubah_status_progres</i>
6	Menyimpan nilai <i>id_ubah_status</i> pada properti objek
7	Menyimpan nilai <i>id_progres</i> pada properti objek <i>id_progres</i>
8	Menyimpan nilai <i>id_pegawai</i> pada properti objek <i>id_pegawai</i>



9	Menyimpan nilai id status progres pada properti objek id status progres
10	Menyimpan data objek ke basis data
11	Jika berhasil menyimpan data
12	Mengembalikan nilai berhasil
13	Jika gagal
14	Mengembalikan nilai gagal
15	Selesai

5.6 Perancangan Antarmuka

Pada bagian ini akan menjelaskan perancangan antarmuka dari sistem informasi manajemen tesis.

5.6.1 Halaman Login

Halaman *login* merupakan halaman untuk pengguna masuk ke dalam sistem, pengguna harus memasukkan *email* dan *password* untuk masuk ke dalam sistem. Berikut ini merupakan rancangan dari antarmuka *login* pada Gambar 5.10

Gambar 5.10 Perancangan antarmuka halaman login

5.6.2 Halaman Pengajuan Judul dan Dosen Pembimbing

Halaman pengajuan judul dan dosen pembimbing merupakan halaman untuk mahasiswa mengajukan judul dan dosen pembimbing baru. Pada Gambar 5.11 merupakan hasil perancangan halaman pengajuan judul dan dosen pembimbing



Home > Pengajuan Judul > ID tesis

Pengajuan Judul dan Komisi Pembimbing

Form Pengajuan Judul dan Komisi Pembimbing

Judul Tesis

Tipe Penelitian

Calon Dosen Pembimbing 1

Calon Dosen Pembimbing 2

Calon Dosen Pembimbing

Calon Dosen Pembimbing

Calon Dosen Pembimbing

Gambar 5.11 Perancangan antarmuka halaman pengajuan judul dan dosen pembimbing

5.6.3 Halaman Detail Progress Pengajuan Judul dan Dosen Pembimbing

Halaman detail progress pengajuan judul dan dosen pembimbing merupakan halaman yang menampilkan tahapan progres dari pengajuan yang sudah diajukan mahasiswa. Pada Gambar 5.12 merupakan rancangan antarmuka detail progres pengajuan judul dan dosen pembimbing.

Home > Pengajuan Judul > ID tesis

Pengajuan Judul dan Komisi Pembimbing

Tahap 1 > Tahap 2 > Tahap 3 > Tahap 4 > Tahap 5 >

Judul Status Progress

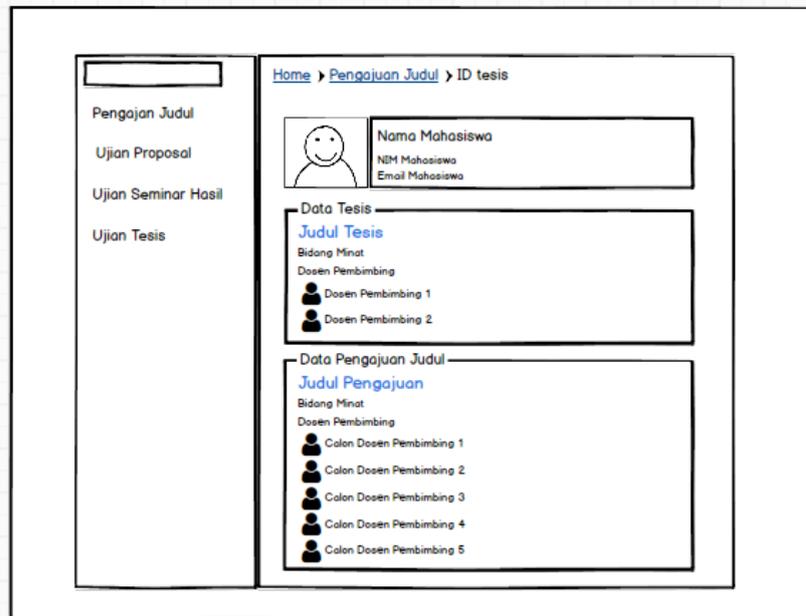
Status

Gambar 5.12 Perancangan antarmuka halaman detail progress pengajuan judul dan dosen pembimbing



5.6.4 Halaman Profil Mahasiswa

Halaman profil mahasiswa merupakan halaman yang terdapat informasi data diri mahasiswa beserta data tesis dan data pengajuan judulnya. Pada Gambar 5.13 merupakan rancangan antarmuka halaman profil mahasiswa.



Gambar 5.13 Perancangan antarmuka halaman profil mahasiswa



BAB 6 IMPLEMENTASI

Bab ini membahas implementasi dari perancangan perangkat lunak yang telah dibuat sebelumnya. Bab ini berisikan spesifikasi sistem, implementasi algoritme, dan implementasi antarmuka.

6.1 Spesifikasi Sistem

Sistem informasi manajemen tesis merupakan sebuah sistem informasi berbasis web, sehingga memerlukan *web browser* dan koneksi internet untuk dapat diakses. Berikut ini merupakan spesifikasi perangkat lunak yang dibutuhkan untuk mengakses sistem informasi manajemen tesis:

1. Sistem operasi windows 7, windows 8, windows 8.1, windows 10, macOS dan linux
2. Web browser yang mendukung HTML versi 5 dan javascript yang aktif seperti Mozilla firefox versi 57.0.1, Google Chrome versi 63.0.3239 dan Microsoft Edge versi 41.16299.15

6.2 Implementasi Algoritme

Implementasi algoritme ini menjelaskan hasil implementasi dari perancangan komponen yang sudah dilakukan pada bab perancangan sebelumnya. Implementasi algoritme ini dibagi menjadi 2 bagian yaitu implementasi *front end* dan implementasi *back end*. Implementasi *front end* menjelaskan hasil implementasi algoritme pengajuan judul dan dosen pembimbing ke kode program pada sistem informasi manajemen tesis. Implementasi *back end* menjelaskan hasil implementasi algoritme pengajuan judul dan dosen pembimbing ke kode program pada *web service* sistem informasi manajemen tesis.

6.2.1 Implementasi Front End

6.2.1.1 Main.js

Pada Tabel 6.1 merupakan hasil implementasi perancangan algoritme dari *file* main.js yang sudah dilakukan pada bab perancangan sebelumnya.

Tabel 6.1 Implementasi algoritme main.js

```

1 import Vue from 'vue';
2 import VueRouter from 'vue-router';
3 import App from './App.vue';
4 import routes from './routes';
5 const router = new VueRouter({
6   routes: routes,
7 });
8 Vue.use(VueRouter);
9 const vm = new Vue({

```



```

10   el: '#apw',
11   router: router,
12   render: h => h(App)
13   });

```

6.2.1.2 Route.js

Pada Tabel 6.2 merupakan hasil implementasi perancangan algoritme dari *file* route.js yang sudah dilakukan pada bab perancangan sebelumnya.

Tabel 6.2 Implementasi algoritme route.js

```

1   import Pengajuan_Proposal_Index from
   './components/Pengajuan_Proposal_Index.vue';
2   import Pengajuan_Proposal_Form from
   './components/Pengajuan_Proposal_Form.vue';
3   import Pengajuan_Proposal_Progress from
   './components/Pengajuan_Proposal_Progress.vue';
4   import Pengajuan_Proposal_List from
   './components/Pengajuan_Proposal_List.vue';
5   const routes = [
6     {
7       path: 'pengajuan-judul',
8       component: Pengajuan_Proposal_Index,
9       children: [
10        { path: '', component: Pengajuan_Proposal_List, meta: {
11          requiresAuth: true } },
12        { path: 'Pengajuan-Proposal', component:
13          Pengajuan_Proposal_Form, meta: { requiresAuth: true } },
14        { path: ':id_proposal', component:
15          Pengajuan_Proposal_Progress, props: true, meta: { requiresAuth:
16            true } },
17      ]
18    }
19  ]

```

6.2.1.3 Vue komponen pengajuan_proposal_index

Pada Tabel 6.3 merupakan hasil implementasi perancangan algoritme dari komponen pengajuan_proposal_index yang sudah dilakukan pada bab perancangan sebelumnya.

**Tabel 6.3 Implementasi algoritme pengajuan_proposal_index**

1	<template lang="html">
2	<transition name="fade" mode="out-in">
3	<router-view key="content"></router-view>
4	</transition>
5	</template>
6	
7	<script>
8	export default {
9	name: "PengajuanProposalIndex",
10	}
11	</script>
12	
13	<style lang="css" scoped>
14	.fade-enter-active, .fade-leave-active {
15	transition: opacity .2s
16	}
17	.fade-enter, .fade-leave-to{
18	opacity: 0
19	}
20	</style>

6.2.1.4 Vue komponen pengajuan_proposal_list

Hasil implementasi perancangan algoritme dari komponen pengajuan_proposal_list yang sudah dilakukan pada bab perancangan sebelumnya dapat dilihat pada Lampiran D.1.

6.2.1.5 Vue komponen pengajuan_proposal_form

Hasil implementasi perancangan algoritme dari komponen pengajuan_proposal_form yang sudah dilakukan pada bab perancangan sebelumnya dapat dilihat pada Lampiran D.2.

6.2.2 Implementasi Back End

6.2.2.1 Controller Pengajuan_Proposal (Fungsi Index)

Fungsi Index merupakan fungsi yang digunakan untuk proses menyimpan data pengajuan proposal judul dan dosen pembimbing. Pada fungsi ini memanggil beberapa model yaitu model M_pengajuan_proposal yang berfungsi untuk menyimpan data pengajuan judul mahasiswa, M_pengajuan_proposal_has_dosen yang berfungsi untuk menyimpan data calon dosen pembimbing yang telah diajukan, M_ubah_status_progress yang berfungsi untuk menyimpan data perkembangan progres pengajuan judul dan M_progress



untuk menyimpan data progres mahasiswa. Fungsi index pada *controller* Pengajuan_Proposal ini dijelaskan pada Tabel 6.4.

Tabel 6.4 Implementasi algoritme fungsi index

```

1  $this->load->model('M_pengajuan_proposal');
2  $this->load->model('M_pengajuan_proposal_has_dosen');
3  $this->load->model('M_ubah_status_progress');
4  $this->load->model('M_progress');
5  $token = $this->input->post('token');
6  $data = $this->input->post('data');
7  $cek_login = $this->token->cek($token);
8  if($cek_login != 'gagal'){
9      $progress['id_progress'] = 'P'.substr(md5(uniqid()),0,8);
10     $progress['id_tipe_progress'] = 'PRG01';
11     $progress['nim_mahasiswa'] = $cek_login['nim_mahasiswa'];
12     $progress['tgl_progress'] = date("Y-m-d H:i:s");
13     $progress['selesai'] = 0;
14     $this->M_progress->insert_progress($progress);
15     $pengajuan['id_pengajuan'] = $progress['id_progress'];
16     $pengajuan['judul_tesis'] = $data['judul_tesis'];
17     $pengajuan['id_bidang_minat'] = $data['tipe_penelitian'];
18     $this->M_pengajuan_proposal->insert_pengajuan_proposal($pengajuan);
19     for ($i=1; $i <= 5; $i++) {
20         $id_dosen = 'dosen_'. $i;
21         if($data[$id_dosen] != ''){
22             $this->M_pengajuan_proposal_has_dosen->insert_proposal_has_dosen(array(
23                 'id_pengajuan_has_dosen' => 'PD'.substr(md5(uniqid()),0,8),
24                 'id_pengajuan' => $progress['id_progress'],
25                 'id_dosen' => $data[$id_dosen]['id_pegawai'],
26                 'dosen_ke' => $i,
27                 'status' => 0
28             ));
29         }
30     }
31     $ubah['id_progress'] = $progress['id_progress'];
32     $ubah['id_status_progress'] = 'SP01';
33     $ubah['id_pegawai'] = 'Akademik';
34     $ubah['ekstra_detail_status'] = '';
35     $ubah['is_selesai'] = 0;
36     $this->M_ubah_status_progress->insert_ubah_status_progress($ubah);
37     $email['subject'] = "Pengajuan Judul dan Komisi Pembimbing";
38     $email['pesan']['kepada'] = "Akademik Program Studi Magister
        Administrasi Bisnis ";

```



```

39     $email['pesan']['p1'] = "Terdapat Pengajuan Judul dan Komisi
    Pembimbing Yang Diajukan Oleh Mahasiswa ";
40     $email['pesan']['p2'] = ucwords($cek_login['nama_mahasiswa']);
41     $email['pesan']['p3'] = $cek_login['nim_mahasiswa'];
42     $email['pesan']['p4'] = $data['judul_tesis'];
43     $email['pesan']['p5'] = '';
44     $email['pesan']['p6'] = '';
    $this->
45     > kirim_email('afifnandya@gmail.com', $email['pesan'], $email['su
    bject']);
46     $status['status'] = 'Berhasil';
47     $status['data'] = 'Data berhasil Dimasukkan';
48     echo json_encode($status);
49 }
50 else{
51     $status['status'] = 'gagal';
52     $status['data'] = 'Tidak terdaftar';
53     echo json_encode($status);
54 }

```

6.2.2.2 Model *M_Progres* (Fungsi *insert_progres*)

Fungsi *insert_progres* merupakan fungsi yang digunakan untuk proses menyimpan data progres mahasiswa. Pada fungsi ini dibuat sebuah objek dari kelas *M_progress*. Objek kelas tersebut disimpan pada variabel tertentu. Properti dari objek tersebut akan menyimpan data dari *array* variabel lain. Selanjutnya objek tersebut akan disimpan pada basis data dengan menjalankan fungsi *insert* milik codeigniter. Fungsi *insert_progres* pada model *M_progress* ini dijelaskan pada Tabel 6.5.

Tabel 6.5 Implementas algoritme fungsi *insert_progres*

```

1     $progress = new M_progress;
2     $progress->id_progress = $data['id_progress'];
3     $progress->id_tipe_progress = $data['id_tipe_progress'];
4     $progress->nim_mahasiswa = $data['nim_mahasiswa'];
5     $progress->selesai = $data['selesai'];
6     $progress->tgl_progress = $data['tgl_progress'];
7     if($this->db->insert('progress', $progress)){
8         return 'berhasil';
9     }
10    else{
11        return 'gagal';
12    }

```



6.2.2.3 Model *M_Pengajuan_Proposal* (Fungsi *insert_pengajuan_proposal*)

Fungsi *insert_pengajuan_proposal* merupakan fungsi yang digunakan untuk proses menyimpan data pengajuan proposal mahasiswa. Pada fungsi ini dibuat sebuah objek dari kelas *M_pengajuan_proposal*. Objek kelas tersebut disimpan pada variabel tertentu. Properti dari objek tersebut akan menyimpan data dari *array* variabel lain. Selanjutnya objek tersebut akan disimpan pada basis data dengan menjalankan fungsi *insert* milik *codeigniter*. Fungsi *insert_pengajuan_proposal* pada model *M_pengajuan_proposal* ini dijelaskan pada Tabel 6.6.

Tabel 6.6 Implementasi algoritme fungsi *insert_pengajuan_proposal*

1	<code>\$pengajuan_proposal = new M_pengajuan_proposal;</code>
2	<code>\$pengajuan_proposal->id_pengajuan = \$data['id_pengajuan'];</code>
3	<code>\$pengajuan_proposal->judul_tesis = \$data['judul_tesis'];</code>
4	<code>\$pengajuan_proposal->id_bidang_minat = \$data['id_bidang_minat'];</code>
5	<code>if(\$this->db >insert('pengajuan_proposal_tesis',\$pengajuan_proposal)){</code>
6	<code> return 'berhasil';</code>
7	<code>}</code>
8	<code>else{</code>
9	<code> return 'gagal';</code>
10	<code>}</code>

6.2.2.4 Model *M_Pengajuan_Proposal_Has_Dosen* (Fungsi *insert_proposal_has_dosen()*)

Fungsi *insert_proposal_has_dosen* merupakan fungsi yang digunakan untuk proses menyimpan data calon dosen pembimbing yang dipilih oleh mahasiswa. Pada fungsi ini dibuat sebuah objek dari kelas *M_pengajuan_proposal_has_dosen*. Objek kelas tersebut disimpan pada variabel tertentu. Properti dari objek tersebut akan menyimpan data dari *array* variabel lain. Selanjutnya objek tersebut akan disimpan pada basis data dengan menjalankan fungsi *insert* milik *codeigniter*. Fungsi *insert_proposal_has_dosen* pada model *M_pengajuan_proposal_has_dosen* ini dijelaskan pada Tabel 6.7.

Tabel 6.7 Implementasi algoritme fungsi *insert_proposal_has_dosen*

1	<code>\$pengajuan_has_dosen = new M_pengajuan_proposal_has_dosen;</code>
2	<code>\$pengajuan_has_dosen->id_pengajuan_has_dosen = \$data['id_pengajuan_has_dosen'];</code>
3	<code>\$pengajuan_has_dosen->id_pengajuan = \$data['id_pengajuan'];</code>
4	<code>\$pengajuan_has_dosen->id_dosen = \$data['id_dosen'];</code>
5	<code>\$pengajuan_has_dosen->dosen_ke = \$data['dosen_ke'];</code>
6	<code>\$pengajuan_has_dosen->status = \$data['status'];</code>
7	<code>if(\$this->db->insert('pengajuan_proposal_tesis_has_dosen',\$pengajuan_has_dosen)</code>
8	<code>{</code>
	<code> return 'berhasil';</code>
	<code>}</code>



```

9     }
10    else{
11        return 'gagal';
12    }

```

6.2.2.5 Model *M_ubah_status_progres* (Fungsi *insert_ubah_status_progres()*)

Fungsi *insert_ubah_status_progres* merupakan fungsi yang digunakan untuk proses menyimpan perkembangan progres tesis mahasiswa. Pada fungsi ini dibuat sebuah objek dari kelas *M_ubah_status_progres*. Objek kelas tersebut disimpan pada variabel tertentu. Properti dari objek tersebut akan menyimpan data dari *array* variabel lain. Selanjutnya objek tersebut akan disimpan pada basis data dengan menjalankan fungsi *insert* milik codeigniter. Fungsi *insert_progres* pada model *M_ubah_status_progres* ini dijelaskan pada Tabel 6.8.

Tabel 6.8 Implementasi algoritme fungsi *insert_ubah_status_progres*

```

1     if($data['id_pegawai'] == 'Akademik'){
2         $this->load->model('M_pegawai');
3         $get_id_akademik = $this->M_pegawai-
4         >get_pegawai(array('id_pegawai'),array('tipe_pegawai' =>
5         'Akademik'));
6         $id_pegawai = $get_id_akademik[0]['id_pegawai'];
7     }
8     else if($data['id_pegawai'] == 'Kaprodi'){
9         $this->load->model('M_pegawai');
10        $get_id_kaprodi = $this->M_pegawai-
11        >get_pegawai(array('id_pegawai'),array('tipe_pegawai' =>
12        'Kaprodi'));
13        $id_pegawai = $get_id_kaprodi[0]['id_pegawai'];
14    }
15    else{
16        $id_pegawai = $data['id_pegawai'];
17    }
18    $ubah_status_progress = new M_ubah_status_progress;
19    $ubah_status_progress->id_ubah_status_progress =
20    $data['id_ubah_status_progress'];
21    $ubah_status_progress->id_progress = $data['id_progress'];
22    $ubah_status_progress->id_pegawai = $id_pegawai;
23    $ubah_status_progress->id_status_progress = $data
24    ['id_status_progress'];
25    $ubah_status_progress->ekstra_detail_status =
26    $data['ekstra_detail_status'];
27    $ubah_status_progress->is_selesai = $data['is_selesai'];
28    $ubah_status_progress->tgl_selesai = $data['tgl_selesai'];
29    if($this->db-
30    >insert('ubah_status_progress',$ubah_status_progress)){
31        return 'berhasil';
32    }
33    else{

```



```
26     return 'gagal';
27 }
```

6.3 Implementasi Antarmuka

Implementasi antarmuka merupakan penjelasan dari antarmuka sistem informasi manajemen tesis yang telah dikembangkan.

6.3.1 Halaman Login

Halaman *login* merupakan halaman awal dari sistem informasi manajemen tesis saat diakses oleh pengguna. Halaman ini berfungsi untuk pengguna *login* ke dalam sistem dengan memasukkan *email* dan *password*. Implementasi antarmuka halaman login ini dibuat berdasarkan perancangan antarmuka yang telah dilakukan pada bab perancangan. Hasil implementasi antarmuka halaman *login* dapat dilihat pada Gambar 6.1.

Gambar 6.1 Implementasi antarmuka halaman *login*

6.3.2 Halaman Pengajuan Judul dan Dosen Pembimbing

Halaman pengajuan judul dan dosen pembimbing merupakan halaman yang menampilkan form pengajuan judul dan dosen pembimbing. Pada halaman ini mahasiswa dapat mengajukan judul dan dosen pembimbing dengan memasukkan judul tesis, tipe penelitian dan memilih calon dosen pembimbing. Implementasi antarmuka halaman pengajuan judul dan dosen pembimbing ini dibuat berdasarkan perancangan antarmuka yang telah dilakukan pada bab perancangan. Hasil implementasi antarmuka halaman pengajuan judul dan dosen pembimbing dapat dilihat pada Gambar 6.2.



Gambar 6.2 Implementasi antarmuka halaman pengajuan judul dan dosen pembimbing

6.3.3 Halaman Detail Progres Pengajuan Judul dan Dosen Pembimbing

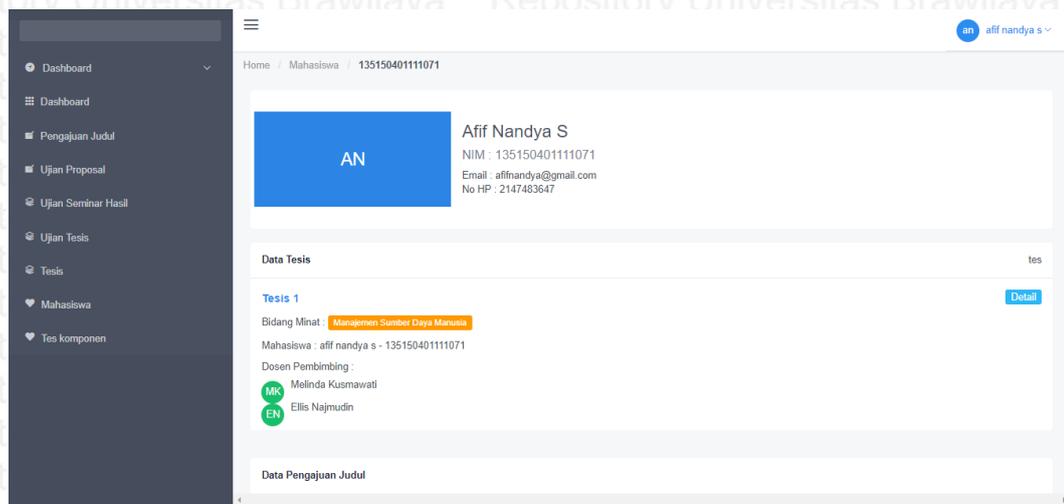
Halaman detail progres pengajuan judul dan dosen pembimbing merupakan halaman yang menampilkan tahapan-tahapan pengajuan judul dan dosen pembimbing. Pada halaman ini menampilkan sejauh mana progres pengajuan yang sudah berjalan. Pada halaman ini juga ditampilkan persetujuan pengajuan dari dosen, akademik dan ketua program studi. Implementasi antarmuka halaman pengajuan judul dan dosen pembimbing ini dibuat berdasarkan perancangan antarmuka yang telah dilakukan pada bab perancangan. Hasil implementasi antarmuka halaman detail progres pengajuan judul dan dosen pembimbing dapat dilihat pada Gambar 6.3.

Gambar 6.3 Implementasi antarmuka halaman detail progres pengajuan judul dan dosen pembimbing



6.3.4 Halaman Profil Mahasiswa

Halaman profil mahasiswa merupakan halaman yang menampilkan data profil mahasiswa. Halaman ini juga menampilkan data tesis dari mahasiswa tersebut dan data pengajuan judul yang telah diajukan. Data tesis dan data pengajuan judul yang ditampilkan pada halaman ini mencakup judul tesis, bidang minat tesis, nama dan nim mahasiswa dan dosen pembimbing atau calon dosen pembimbing. Implementasi antarmuka halaman profil mahasiswa ini dibuat berdasarkan perancangan antarmuka yang telah dilakukan pada bab perancangan. Hasil implementasi antarmuka halaman profil mahasiswa dapat dilihat pada Gambar 6.4



Gambar 6.4 Implementasi antarmuka halaman profil mahasiswa

BAB 7 PENGUJIAN

Bab ini membahas pengujian dari perangkat lunak yang telah dikembangkan sebelumnya. Pengujian dilakukan untuk menemukan kesalahan-kesalahan yang terjadi dan memastikan sistem yang dikembangkan telah sesuai dengan perancangan sebelumnya.

7.1 Pengujian *Black-box*

7.1.1 Pengujian Validasi Aktor Pengguna

Tabel 7.1 Kasus uji melakukan *login*

Nama Kasus Uji	Melakukan <i>login</i> .
Objek Uji	SIMT-2-001
Tujuan Pengujian	Pengujian dilakukan untuk memastikan aktor pengguna dapat melakukan pendaftaran <i>login</i> ke dalam sistem
Prosedur Uji	<ol style="list-style-type: none">1. Aktor memasukkan <i>email</i> dan <i>password</i>2. Aktor menekan tombol <i>login</i>3. Aktor berhasil <i>login</i> ke dalam sistem
Hasil Yang Diharapkan	Aktor berhasil <i>login</i> ke dalam sistem.
Hasil Pengujian	Aktor berhasil <i>login</i> ke dalam sistem.
Status Validasi	Valid.

Tabel 7.2 Kasus uji melakukan *login* alternatif 1: *email* dan *password* tidak terdaftar

Nama Kasus Uji	Melakukan <i>login</i> .
Objek Uji	SIMT-2-001
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor gagal <i>login</i> apabila memasukan <i>email</i> dan <i>password</i> yang tidak terdaftar didalam pengguna dapat melakukan pendaftaran <i>login</i> ke dalam sistem
Prosedur Uji	<ol style="list-style-type: none">1. Aktor memasukkan <i>email</i> dan <i>password</i> yang tidak terdaftar2. Aktor menekan tombol <i>login</i>3. Sistem menampilkan pesan kesalahan
Hasil Yang Diharapkan	Aktor gagal <i>login</i> ke dalam sistem dan sistem menampilkan pesan kesalahan.
Hasil Pengujian	Aktor gagal <i>login</i> ke dalam sistem dan sistem menampilkan pesan kesalahan.
Status Validasi	Valid.



Tabel 7.3 Kasus uji mendaftar sebagai mahasiswa

Nama Kasus Uji	Mendaftar sebagai mahasiswa.
Objek Uji	SIMT-2-002
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mendaftar sebagai mahasiswa.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memasukkan <i>email</i>, <i>password</i>, nama, nim dan nomer telpon 2. Aktor menekan tombol <i>register</i>. 3. Sistem menampilkan pesan registrasi berhasil.
Hasil Yang Diharapkan	Aktor berhasil mendaftar sebagai mahasiswa.
Hasil Pengujian	Aktor berhasil mendaftar sebagai mahasiswa.
Status Validasi	Valid.

Tabel 7.4 Kasus uji mendaftar sebagai mahasiswa alternatif 1: email tidak valid atau sudah terdaftar

Nama Kasus Uji	Mendaftar sebagai mahasiswa.
Objek Uji	SIMT-2-002
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem menampilkan pesan kesalahan apabila aktor memasukkan email yang tidak valid atau sudah terdaftar.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memasukkan <i>email</i> yang sudah terdaftar. 2. Sistem menampilkan pesan bahwa email tidak valid atau sudah terdaftar.
Hasil Yang Diharapkan	Sistem menampilkan pesan bahwa email tidak valid atau sudah terdaftar.
Hasil Pengujian	Sistem menampilkan pesan bahwa email tidak valid atau sudah terdaftar.
Status Validasi	Valid.

Tabel 7.5 Kasus uji mendaftar sebagai dosen

Nama Kasus Uji	Mendaftar sebagai dosen.
Objek Uji	SIMT-2-003
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mendaftar sebagai dosen.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memasukkan <i>email</i>, <i>password</i>, nama, NIP dan bidang minat. 2. Aktor menekan tombol <i>register</i>. 3. Sistem menampilkan pesan bahwa pendaftaran berhasil.



Hasil Yang Diharapkan	Sistem menampilkan pesan bahwa pendaftaran berhasil.
Hasil Pengujian	Sistem menampilkan pesan bahwa pendaftaran berhasil.
Status Validasi	Valid.

Tabel 7.6 Kasus uji mendaftar sebagai dosen alternatif 1: email tidak valid atau sudah terdaftar

Nama Kasus Uji	Mendaftar sebagai dosen.
Objek Uji	SIMT-2-002
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem menampilkan pesan kesalahan apabila aktor memasukkan <i>email</i> yang tidak valid atau sudah terdaftar.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memasukkan <i>email</i> yang sudah terdaftar. 2. Sistem menampilkan pesan bahwa email tidak valid atau sudah terdaftar.
Hasil Yang Diharapkan	Sistem menampilkan pesan bahwa email tidak valid atau sudah terdaftar.
Hasil Pengujian	Sistem menampilkan pesan bahwa email tidak valid atau sudah terdaftar.
Status Validasi	Valid.

Tabel 7.7 Kasus uji mengatur ulang password

Nama Kasus Uji	Mengatur ulang password.
Objek Uji	SIMT-2-003
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengatur ulang password.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memasukkan <i>email</i> yang digunakan untuk <i>login</i>. 2. Aktor menekan tombol <i>submit</i>. 3. Sistem menampilkan pesan berhasil mengirim password baru ke email yang telah dimasukkan. 4. Aktor mendapat <i>password</i> baru.
Hasil Yang Diharapkan	Sistem menampilkan pesan berhasil mengirim password baru ke email dan aktor dapat <i>password</i> baru.
Hasil Pengujian	Sistem menampilkan pesan berhasil mengirim password baru ke email dan aktor dapat <i>password</i> baru.
Status Validasi	Valid.



7.1.2 Pengujian Validasi Aktor Pegawai

Tabel 7.8 Kasus uji melihat profil mahasiswa

Nama Kasus Uji	Melihat profil mahasiswa.
Objek Uji	SIMT-2-005
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat melihat profil mahasiswa.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu mahasiswa 2. Sistem menampilkan daftar mahasiswa. 3. Aktor menekan nama mahasiswa. 4. Sistem menampilkan profil mahasiswa.
Hasil Yang Diharapkan	Sistem menampilkan profil mahasiswa.
Hasil Pengujian	Sistem menampilkan profil mahasiswa.
Status Validasi	Valid.

Tabel 7.9 Kasus uji melihat profil

Nama Kasus Uji	Melihat profil.
Objek Uji	SIMT-2-006
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat melihat profil dirinya sendiri.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan tombol profil pada <i>dropdown</i> nama pengguna. 2. Sistem menampilkan halaman profil.
Hasil Yang Diharapkan	Sistem menampilkan halaman profil.
Hasil Pengujian	Sistem menampilkan halaman profil.
Status Validasi	Valid.

Tabel 7.10 Kasus uji melihat data tesis

Nama Kasus Uji	Melihat data tesis.
Objek Uji	SIMT-2-007
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat melihat data tesis dari mahasiswa.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu mahasiswa 2. Sistem menampilkan daftar mahasiswa. 3. Aktor menekan nama mahasiswa. 4. Sistem menampilkan profil mahasiswa. 5. Aktor menekan judul tesis mahasiswa pada kolom data tesis. 6. Sistem menampilkan data tesis mahasiswa.
Hasil Yang Diharapkan	Sistem menampilkan data tesis mahasiswa.
Hasil Pengujian	Sistem menampilkan data tesis mahasiswa.
Status Validasi	Valid.

**Tabel 7.11 Kasus uji melihat data pengajuan judul dan dosen pembimbing**

Nama Kasus Uji	Melihat data pengajuan judul dan dosen pembimbing tesis.
Objek Uji	SIMT-2-008
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat melihat data pengajuan judul dan dosen pembimbing.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu mahasiswa 2. Sistem menampilkan daftar mahasiswa. 3. Aktor menekan nama mahasiswa. 4. Sistem menampilkan profil mahasiswa. 5. Aktor menekan judul tesis mahasiswa pada kolom data pengajuan. 6. Sistem menampilkan data pengajuan judul dan dosen pembimbing tesis.
Hasil Yang Diharapkan	Sistem menampilkan data pengajuan judul dan dosen pembimbing mahasiswa.
Hasil Pengujian	Sistem menampilkan data pengajuan judul dan dosen pembimbing mahasiswa.
Status Validasi	Valid.

Tabel 7.12 Kasus uji melihat data ujian proposal

Nama Kasus Uji	Melihat data ujian proposal.
Objek Uji	SIMT-2-009
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat melihat data ujian proposal mahasiswa.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu mahasiswa 2. Sistem menampilkan daftar mahasiswa. 3. Aktor menekan nama mahasiswa. 4. Sistem menampilkan profil mahasiswa. 5. Aktor menekan judul tesis kolom data tesis. 6. Sistem menampilkan data data tesis mahasiswa. 7. Aktor menekan <i>tab</i> ujian proposal. 8. Sistem menampilkan data ujian proposal.
Hasil Yang Diharapkan	Sistem menampilkan data ujian proposal mahasiswa.
Hasil Pengujian	Sistem menampilkan data ujian proposal mahasiswa.
Status Validasi	Valid.

Tabel 7.13 Kasus uji melihat data seminar hasil

Nama Kasus Uji	Melihat data seminar hasil.
Objek Uji	SIMT-2-010



Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat melihat data seminar hasil mahasiswa.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu mahasiswa 2. Sistem menampilkan daftar mahasiswa. 3. Aktor menekan nama mahasiswa. 4. Sistem menampilkan profil mahasiswa. 5. Aktor menekan judul tesis kolom data tesis. 6. Sistem menampilkan data data tesis mahasiswa. 7. Aktor menekan <i>tab</i> seminar hasil. 8. Sistem menampilkan data seminar hasil.
Hasil Yang Diharapkan	Sistem menampilkan data seminar hasil mahasiswa.
Hasil Pengujian	Sistem menampilkan data seminar hasil mahasiswa.
Status Validasi	Valid.

Tabel 7.14 Kasus uji melihat data ujian tesis

Nama Kasus Uji	Melihat data ujian tesis.
Objek Uji	SIMT-2-011
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat melihat data ujian tesis mahasiswa.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu mahasiswa 2. Sistem menampilkan daftar mahasiswa. 3. Aktor menekan nama mahasiswa. 4. Sistem menampilkan profil mahasiswa. 5. Aktor menekan judul tesis kolom data tesis. 6. Sistem menampilkan data data tesis mahasiswa. 7. Aktor menekan <i>tab</i> ujian tesis. 8. Sistem menampilkan data ujian tesis.
Hasil Yang Diharapkan	Sistem menampilkan data ujian tesis mahasiswa.
Hasil Pengujian	Sistem menampilkan data ujian tesis mahasiswa.
Status Validasi	Valid.

Tabel 7.15 Kasus uji melakukan logout

Nama Kasus Uji	Melakukan <i>logout</i> .
Objek Uji	SIMT-2-012
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat <i>logout</i> dari sistem.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan tombol <i>logout</i> pada <i>dropdown</i> nama pengguna. 2. Aktor berhasil <i>logout</i> dari sistem.
Hasil Yang Diharapkan	Aktor berhasil <i>logout</i> dari sistem.
Hasil Pengujian	Aktor berhasil <i>logout</i> dari sistem.
Status Validasi	Valid.

**Tabel 7.16 Kasus uji mengubah password**

Nama Kasus Uji	Mengubah <i>password</i> .
Objek Uji	SIMT-2-013
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengubah password.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan tombol ganti <i>password</i> pada <i>dropdown</i> nama pengguna. 2. Sistem menampilkan formulir ganti password. 3. Pengguna mengisi kolom password baru dan kolom ulangi password baru. 4. Pengguna menekan tombol <i>submit</i>. 5. Sistem menampilkan pesan password berhasil diubah.
Hasil Yang Diharapkan	Aktor berhasil mengubah password.
Hasil Pengujian	Aktor berhasil mengubah password.
Status Validasi	Valid.

Tabel 7.17 Kasus uji mengubah *password* alternatif 1: isi kolom *password* baru dan ulangi *password* baru tidak sama

Nama Kasus Uji	Mengubah <i>password</i> .
Objek Uji	SIMT-2-013
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem menampilkan pesan kesalahan saat isi kolom password baru dan ulangi password baru tidak sama.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan tombol ganti <i>password</i> pada <i>dropdown</i> nama pengguna. 2. Sistem menampilkan formulir ganti password. 3. Pengguna mengisi kolom password baru dan kolom ulangi password baru dengan isi yang berbeda. 4. Sistem menampilkan pesan bahwa kolom password baru dan ulangi password baru tidak sama.
Hasil Yang Diharapkan	Sistem menampilkan pesan kesalahan bahwa isi kolom password dan ulangi password tidak sama.
Hasil Pengujian	Sistem menampilkan pesan kesalahan bahwa isi kolom password dan ulangi password tidak sama.
Status Validasi	Valid.

7.1.3 Pengujian Validasi Aktor Mahasiswa

Tabel 7.18 Kasus uji mengubah *password*

Nama Kasus Uji	Mengubah <i>password</i> .
-----------------------	----------------------------



Objek Uji	SIMT-2-014
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengubah password.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan tombol ganti <i>password</i> pada <i>dropdown</i> nama pengguna. 2. Sistem menampilkan formulir ganti password. 3. Pengguna mengisi kolom password baru dan kolom ulangi password baru. 4. Pengguna menekan tombol <i>submit</i>. 5. Sistem menampilkan pesan password berhasil diubah.
Hasil Yang Diharapkan	Aktor berhasil mengubah password.
Hasil Pengujian	Aktor berhasil mengubah password.
Status Validasi	Valid.

Tabel 7.19 Kasus uji mengubah *password* alternatif 1: isi kolom *password* baru dan ulangi *password* baru tidak sama

Nama Kasus Uji	Mengubah <i>password</i> .
Objek Uji	SIMT-2-014
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem menampilkan pesan kesalahan saat isi kolom password baru dan ulangi password baru tidak sama.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan tombol ganti <i>password</i> pada <i>dropdown</i> nama pengguna. 2. Sistem menampilkan formulir ganti password. 3. Pengguna mengisi kolom password baru dan kolom ulangi password baru dengan isi yang berbeda. 4. Sistem menampilkan pesan bahwa kolom password baru dan ulangi password baru tidak sama.
Hasil Yang Diharapkan	Sistem menampilkan pesan kesalahan bahwa isi kolom password dan ulangi password tidak sama.
Hasil Pengujian	Sistem menampilkan pesan kesalahan bahwa isi kolom password dan ulangi password tidak sama.
Status Validasi	Valid.

Tabel 7.20 Kasus uji mengisi formulir pengajuan judul dan dosen pembimbing

Nama Kasus Uji	Mengisi formulir pengajuan judul dan dosen pembimbing
-----------------------	---



Objek Uji	SIMT-2-015
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengajukan judul dan dosen pembimbing.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu pengajuan judul. 2. Sistem menampilkan halaman pengajuan judul dan dosen pembimbing. 3. Aktor menekan tombol ajukan judul. 4. Sistem menampilkan formulir pengajuan judul dan dosen pembimbing. 5. Aktor mengisi formulir pengajuan judul dan dosen pembimbing dan menekan. 6. Aktor menekan tombol submit. 7. Sistem menyimpan data pengajuan judul dan menampilkan pengajuan judul berhasil disubmit.
Hasil Yang Diharapkan	Sistem menyimpan data pengajuan judul dan menampilkan pesan pengajuan judul berhasil disubmit.
Hasil Pengujian	Sistem menyimpan data pengajuan judul dan menampilkan pesan pengajuan judul berhasil disubmit.
Status Validasi	Valid.

Tabel 7.21 Kasus uji mengisi formulir pengajuan judul dan dosen pembimbing alternatif 1: kolom yang wajib diisi tidak diisi

Nama Kasus Uji	Mengisi formulir pengajuan judul dan dosen pembimbing
Objek Uji	SIMT-2-015
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa sistem menampilkan pesan kesalahan saat kolom yang wajib diisi tidak diisi.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu pengajuan judul. 2. Sistem menampilkan halaman pengajuan judul dan dosen pembimbing. 3. Aktor menekan tombol ajukan judul. 4. Sistem menampilkan formulir pengajuan judul dan dosen pembimbing. 5. Aktor tidak mengisi kolom yang wajib diisi pada formulir pengajuan judul dan dosen pembimbing. 6. Sistem menampilkan pesan kesalahan lengkapi formulir pengajuan.
Hasil Yang Diharapkan	Sistem menampilkan pesan kesalahan lengkapi formulir pengajuan.
Hasil Pengujian	Sistem menampilkan pesan kesalahan lengkapi formulir pengajuan.



Status Validasi	Valid.
------------------------	--------

Tabel 7.22 Kasus uji melihat data pengajuan judul dan dosen pembimbing

Nama Kasus Uji	Melihat data pengajuan judul dan dosen pembimbing
Objek Uji	SIMT-2-016
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat melihat data pengajuan judul dan dosen pembimbing.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu pengajuan judul. 2. Sistem menampilkan halaman pengajuan judul dan dosen pembimbing. 3. Aktor menekan judul pengajuan. 4. Sistem menampilkan data pengajuan judul dan dosen pembimbing.
Hasil Yang Diharapkan	Sistem menampilkan data pengajuan judul dan dosen pembimbing.
Hasil Pengujian	Sistem menampilkan data pengajuan judul dan dosen pembimbing.
Status Validasi	Valid.

Tabel 7.23 Mengisi formulir persetujuan ujian proposal

Nama Kasus Uji	Mengisi formulir persetujuan ujian proposal
Objek Uji	SIMT-2-017
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengajukan persetujuan ujian proposal.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu ujian proposal. 2. Sistem menampilkan halaman ujian proposal. 3. Aktor menekan ajukan ujian proposal tesis. 4. Sistem menampilkan formulir pengajuan ujian proposal. 5. Aktor menekan tombol ajukan persetujuan 6. Sistem menyimpan pengajuan ujian proposal dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan ujian proposal dan menampilkan pesan data persetujuan berhasil disimpan.
Hasil Pengujian	Sistem menyimpan data persetujuan ujian proposal dan menampilkan pesan data persetujuan berhasil disimpan.
Status Validasi	Valid.



Tabel 7.24 Kasus uji mengisi formulir pengajuan ujian proposal

Nama Kasus Uji	Mengisi formulir pengajuan ujian proposal
Objek Uji	SIMT-2-018
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengajukan ujian proposal.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu ujian proposal. 2. Sistem menampilkan halaman ujian proposal. 3. Aktor menekan judul tesis. 4. Sistem menampilkan formulir pengajuan ujian proposal. 5. Aktor mengisi formulir pengajuan ujian proposal 6. Aktor menekan tombol ajukan ujian proposal 7. Sistem menyimpan data pengajuan ujian proposal dan menampilkan pesan data pengajuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data pengajuan ujian proposal dan menampilkan pesan data pengajuan berhasil disimpan.
Hasil Pengujian	Sistem menyimpan data pengajuan ujian proposal dan menampilkan pesan data pengajuan berhasil disimpan.
Status Validasi	Valid.

Tabel 7.25 Kasus uji melihat data ujian proposal

Nama Kasus Uji	Melihat data ujian proposal.
Objek Uji	SIMT-2-019
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat melihat data ujian proposal.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu profil pada <i>dropdown</i> nama pengguna. 2. Sistem menampilkan halaman profil. 3. Aktor menekan judul tesis kolom data tesis. 4. Sistem menampilkan data data tesis mahasiswa. 5. Aktor menekan <i>tab</i> ujian proposal. 6. Sistem menampilkan data ujian proposal.
Hasil Yang Diharapkan	Sistem menampilkan data ujian proposal.
Hasil Pengujian	Sistem menampilkan data ujian proposal.
Status Validasi	Valid.

Tabel 7.26 Kasus uji mengisi formulir persetujuan seminar hasil

Nama Kasus Uji	Mengisi formulir persetujuan seminar hasil
-----------------------	--



Objek Uji	SIMT-2-020
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengajukan persetujuan seminar hasil
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu seminar hasil. 2. Sistem menampilkan halaman seminar hasil. 3. Aktor menekan ajukan seminar hasil. 4. Sistem menampilkan formulir persetujuan seminar hasil. 5. Aktor menekan tombol ajukan persetujuan 6. Sistem menyimpan persetujuan seminar hasil dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan seminar hasil dan menampilkan pesan data persetujuan berhasil disimpan.
Hasil Pengujian	Sistem menyimpan data persetujuan seminar hasil dan menampilkan pesan data persetujuan berhasil disimpan.
Status Validasi	Valid.

Tabel 7.27 Mengisi formulir pengajuan seminar hasil

Nama Kasus Uji	Mengisi formulir pengajuan seminar hasil
Objek Uji	SIMT-2-021
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengajukan seminar hasil.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu seminar hasil. 2. Sistem menampilkan halaman seminar hasil. 3. Aktor menekan judul tesis. 4. Sistem menampilkan formulir pengajuan seminar hasil. 5. Aktor mengisi formulir pengajuan seminar hasil. 6. Aktor menekan tombol ajukan seminar hasil. 7. Sistem menyimpan data pengajuan seminar hasil dan menampilkan pesan data pengajuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data pengajuan seminar hasil dan menampilkan pesan data pengajuan berhasil disimpan.
Hasil Pengujian	Sistem menyimpan data pengajuan seminar hasil dan menampilkan pesan data pengajuan berhasil disimpan.
Status Validasi	Valid.



Tabel 7.28 Kasus uji melihat data seminar hasil

Nama Kasus Uji	Melihat data seminar hasil.
Objek Uji	SIMT-2-022
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat melihat data seminar hasil.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu profil pada <i>dropdown</i> nama pengguna. 2. Sistem menampilkan halaman profil. 3. Aktor menekan judul tesis kolom data tesis. 4. Sistem menampilkan data data tesis mahasiswa. 5. Aktor menekan <i>tab</i> seminar hasil. 6. Sistem menampilkan data seminar hasil.
Hasil Yang Diharapkan	Sistem menampilkan data seminar hasil.
Hasil Pengujian	Sistem menampilkan data seminar hasil.
Status Validasi	Valid.

Tabel 7.29 Kasus ujia mengisi formulir persetujuan ujian tesis

Nama Kasus Uji	Mengisi formulir persetujuan ujian tesis
Objek Uji	SIMT-2-023
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengajukan persetujuan ujian tesis.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu ujian tesis. 2. Sistem menampilkan halaman ujian tesis. 3. Aktor menekan ajukan ujian tesis. 4. Sistem menampilkan formulir persetujuan ujian tesis. 5. Aktor menekan tombol ajukan persetujuan. 6. Sistem menyimpan persetujuan ujian tesis dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan ujian tesis dan menampilkan pesan data persetujuan berhasil disimpan.
Hasil Pengujian	Sistem menyimpan data persetujuan ujian tesis dan menampilkan pesan data persetujuan berhasil disimpan.
Status Validasi	Valid.

Tabel 7.30 Kasus uji mengisi formulir pengajuan ujian tesis

Nama Kasus Uji	Mengisi formulir pengajuan ujian tesis
Objek Uji	SIMT-2-024



Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengajukan ujian tesis.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu ujian tesis. 2. Sistem menampilkan halaman ujian tesis. 3. Aktor menekan judul tesis. 4. Sistem menampilkan formulir pengajuan ujian tesis. 5. Aktor mengisi formulir pengajuan ujian tesis. 6. Aktor menekan tombol ajukan ujian tesis. 7. Sistem menyimpan data pengajuan seminar hasil dan menampilkan pesan data pengajuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data pengajuan ujian tesis dan menampilkan pesan data pengajuan berhasil disimpan.
Hasil Pengujian	Sistem menyimpan data pengajuan ujian tesis dan menampilkan pesan data pengajuan berhasil disimpan.
Status Validasi	Valid.

Tabel 7.31 Kasus uji melihat data ujian tesis

Nama Kasus Uji	Melihat data ujian tesis.
Objek Uji	SIMT-2-025
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat melihat data ujian tesis.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu profil pada <i>dropdown</i> nama pengguna. 2. Sistem menampilkan halaman profil. 3. Aktor menekan judul tesis kolom data tesis. 4. Sistem menampilkan data data tesis mahasiswa. 5. Aktor menekan <i>tab</i> ujian tesis. 6. Sistem menampilkan data ujian tesis.
Hasil Yang Diharapkan	Sistem menampilkan data ujian tesis.
Hasil Pengujian	Sistem menampilkan data ujian tesis.
Status Validasi	Valid.

Tabel 7.32 Kasus uji melihat profil

Nama Kasus Uji	Melihat profil.
Objek Uji	SIMT-2-026



Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat melihat profilnya.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu profil pada <i>dropdown</i> nama pengguna. 2. Sistem menampilkan halaman profil.
Hasil Yang Diharapkan	Sistem menampilkan halaman profil.
Hasil Pengujian	Sistem menampilkan halaman profil.
Status Validasi	Valid.

Tabel 7.33 Kasus uji melakukan *logout*

Nama Kasus Uji	Melakukan <i>logout</i> .
Objek Uji	SIMT-2-027
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat <i>logout</i> dari sistem.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan tombol <i>logout</i> pada <i>dropdown</i> nama pengguna. 2. Aktor berhasil <i>logout</i> dari sistem.
Hasil Yang Diharapkan	Aktor berhasil <i>logout</i> dari sistem.
Hasil Pengujian	Aktor berhasil <i>logout</i> dari sistem.
Status Validasi	Valid.

7.1.4 Pengujian Validasi Aktor Dosen Pembimbing

Tabel 7.34 Kasus uji menyetujui menjadi dosen pembimbing

Nama Kasus Uji	Menyetujui menjadi dosen pembimbing
Objek Uji	SIMT-2-028
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat menyetujui menjadi dosen pembimbing.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu pengajuan judul. 2. Aktor menekan tombol setuju. 3. Sistem menampilkan konfirmasi persetujuan. 4. Aktor menekan tombol ok. 5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.



Tabel 7.35 Kasus uji megeloyal persetujuan ujian proposal

Nama Kasus Uji	Mengelola persetujuan ujian proposal
Objek Uji	SIMT-2-029
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengelola persetujuan ujian proposal.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu ujian proposal. 2. Aktor menekan tombol setuju untuk menyetujui persetujuan ujian proposal atau menekan tombol tolak dan mengisi alasan penolakan untuk menolak ujian proposal. 3. Sistem menampilkan konfirmasi persetujuan atau penolakan. 4. Aktor menekan tombol ok. 5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.

Tabel 7.36 Kasus uji mengelola persetujuan seminar hasil

Nama Kasus Uji	Mengelola persetujuan seminar hasil
Objek Uji	SIMT-2-030
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengelola persetujuan seminar hasil.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu seminar hasil. 2. Aktor menekan tombol setuju untuk menyetujui persetujuan seminar hasil atau menekan tombol tolak dan mengisi alasan penolakan untuk menolak seminar hasil. 3. Sistem menampilkan konfirmasi persetujuan atau penolakan. 4. Aktor menekan tombol ok. 5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan



Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.

Tabel 7.37 Kasus uji mengelola persetujuan ujian tesis

Nama Kasus Uji	Mengelola persetujuan ujian tesis
Objek Uji	SIMT-2-031
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengelola persetujuan ujian tesis.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu ujian tesis. 2. Aktor menekan tombol setuju untuk menyetujui persetujuan ujian tesis atau menekan tombol tolak dan mengisi alasan penolakan untuk menolak ujian tesis. 3. Sistem menampilkan konfirmasi persetujuan atau penolakan. 4. Aktor menekan tombol ok. 5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.

Tabel 7.38 Kasus uji menyetujui pengajuan ujian proposal

Nama Kasus Uji	Menyetujui pengajuan ujian proposal
Objek Uji	SIMT-2-032
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat menyetujui pengajuan ujian proposal.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu ujian proposal. 2. Aktor menekan tombol setuju. 3. Sistem menampilkan konfirmasi 4. Aktor menekan tombol ok. 5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan



Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.

Tabel 7.39 Kasus uji menyetujui pengajuan seminar hasil

Nama Kasus Uji	Menyetujui pengajuan seminar hasil
Objek Uji	SIMT-2-034
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat menyetujui pengajuan seminar hasil.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu seminar hasil. 2. Aktor menekan tombol setuju. 3. Sistem menampilkan konfirmasi 4. Aktor menekan tombol ok. 5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.

Tabel 7.40 Kasus uji menyetujui ujian tesis

Nama Kasus Uji	Menyetujui pengajuan ujian tesis
Objek Uji	SIMT-2-033
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat menyetujui pengajuan ujian tesis.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu ujian tesis. 2. Aktor menekan tombol setuju. 3. Sistem menampilkan konfirmasi 4. Aktor menekan tombol ok. 5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan



Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.

7.1.5 Pengujian Validasi Aktor Dosen Penguji

Tabel 7.41 Kasus uji menyetujui ujian proposal

Nama Kasus Uji	Menyetujui pengajuan ujian proposal
Objek Uji	SIMT-2-034
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat menyetujui pengajuan ujian proposal.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu ujian proposal. 2. Aktor menekan tombol setuju. 3. Sistem menampilkan konfirmasi 4. Aktor menekan tombol ok. 5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.

Tabel 7.42 Kasus uji menyetujui ujian tesis

Nama Kasus Uji	Menyetujui pengajuan ujian tesis
Objek Uji	SIMT-2-035
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat menyetujui pengajuan ujian tesis.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu ujian tesis. 2. Aktor menekan tombol setuju. 3. Sistem menampilkan konfirmasi 4. Aktor menekan tombol ok. 5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan



Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.

7.1.6 Pengujian Validasi Aktor Kaprodi

Tabel 7.43 Kasus uji mengelola pengajuan judul dan menentukan dosen pembimbing

Nama Kasus Uji	Mengelola pengajuan judul dan menentukan dosen pembimbing
Objek Uji	SIMT-2-037
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat menyetujui dan menentukan dosen pembimbing atau menolak pengajuan judul.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu pengajuan judul. 2. Aktor menekan tombol setuju dan memilih dosen pembimbing untuk menyetujui pengajuan atau menekan tombol tolak dan mengisi alasan penolakan untuk menolak pengajuan judul. 3. Sistem menampilkan konfirmasi 4. Aktor menekan tombol ok. 5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.

Tabel 7.44 Kasus uji mengelola persetujuan ujian proposal

Nama Kasus Uji	Mengelola persetujuan ujian proposal
Objek Uji	SIMT-2-038
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengelola persetujuan ujian proposal.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu ujian proposal.



	<p>2. Aktor menekan tombol setuju untuk menyetujui persetujuan ujian proposal atau menekan tombol tolak dan mengisi alasan penolakan untuk menolak ujian proposal.</p> <p>3. Sistem menampilkan konfirmasi persetujuan atau penolakan.</p> <p>4. Aktor menekan tombol ok.</p> <p>5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan</p>
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.

Tabel 7.45 Kasus uji mengelola persetujuan seminar hasil

Nama Kasus Uji	Mengelola persetujuan seminar hasil
Objek Uji	SIMT-2-039
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengelola persetujuan seminar hasil.
Prosedur Uji	<p>1. Aktor menekan menu seminar hasil.</p> <p>2. Aktor menekan tombol setuju untuk menyetujui persetujuan seminar hasil atau menekan tombol tolak dan mengisi alasan penolakan untuk menolak seminar hasil.</p> <p>3. Sistem menampilkan konfirmasi persetujuan atau penolakan.</p> <p>4. Aktor menekan tombol ok.</p> <p>5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan</p>
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.



Tabel 7.46 Kasus uji mengelola persetujuan ujian tesis

Nama Kasus Uji	Mengelola persetujuan ujian tesis
Objek Uji	SIMT-2-040
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengelola persetujuan ujian tesis.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu persetujuan ujian tesis. 2. Aktor menekan tombol setuju untuk menyetujui persetujuan ujian tesis atau menekan tombol tolak dan mengisi alasan penolakan untuk menolak ujian tesis. 3. Sistem menampilkan konfirmasi persetujuan atau penolakan. 4. Aktor menekan tombol ok. 5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.

Tabel 7.47 Kasus uji menyetujui pengajuan ujian proposal dan menentukan dosen penguji

Nama Kasus Uji	Menyetujui pengajuan ujian proposal dan menentukan dosen penguji
Objek Uji	SIMT-2-041
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat menyetujui ujian proposal dan menentukan dosen penguji.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu pengajuan ujian proposal. 2. Aktor menekan tombol setuju 3. Sistem menampilkan konfirmasi 4. Aktor memilih dosen penguji ujian proposal. 5. Aktor menekan tombol ok. 6. Sistem menyimpan data pengajuan ujian proposal.
Hasil Yang Diharapkan	Sistem menyimpan data pengajuan dan menampilkan pesan data pengajuan berhasil disimpan



Hasil Pengujian	Sistem menyimpan data pengajuan dan menampilkan pesan data pengajuan berhasil disimpan
Status Validasi	Valid.

Tabel 7.48 Kasus uji menyetujui pengajuan seminar hasil

Nama Kasus Uji	Menyetujui pengajuan seminar hasil
Objek Uji	SIMT-2-042
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat menyetujui pengajuan seminar hasil.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu pengajuan seminar hasil. 2. Aktor menekan tombol setuju 3. Sistem menampilkan konfirmasi 4. Aktor menekan tombol ok. 5. Sistem menyimpan data pengajuan seminar hasil.
Hasil Yang Diharapkan	Sistem menyimpan data pengajuan dan menampilkan pesan data pengajuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data pengajuan dan menampilkan pesan data pengajuan berhasil disimpan
Status Validasi	Valid.

Tabel 7.49 Kasus uji menyetujui pengajuan ujian tesis dan menentukan dosen penguji

Nama Kasus Uji	Menyetujui pengajuan ujian tesis dan menentukan dosen penguji
Objek Uji	SIMT-2-043
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat menyetujui ujian tesis dan menentukan dosen penguji.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu pengajuan ujian tesis. 2. Aktor menekan tombol setuju 3. Sistem menampilkan konfirmasi 4. Aktor memilih dosen penguji ujian tesis. 5. Aktor menekan tombol ok. 6. Sistem menyimpan data pengajuan ujian tesis.
Hasil Yang Diharapkan	Sistem menyimpan data pengajuan dan menampilkan pesan data pengajuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data pengajuan dan menampilkan pesan data pengajuan berhasil disimpan



Status Validasi	Valid.
------------------------	--------

Tabel 7.50 Kasus uji melihat profil dosen

Nama Kasus Uji	Melihat profil dosen
Objek Uji	SIMT-2-044
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat melihat profil dosen.
Prosedur Uji	1. Aktor menekan menu dosen. 2. Aktor nama dosen 3. Sistem menampilkan halaman profil dosen
Hasil Yang Diharapkan	Sistem menampilkan halaman profil dosen
Hasil Pengujian	Sistem menampilkan halaman profil dosen
Status Validasi	Valid.

7.1.7 Pengujian Validasi Aktor Akademik

Tabel 7.51 Kasus uji mengelola pengajuan judul dan dosen pembimbing

Nama Kasus Uji	Mengelola pengajuan judul dan dosen pembimbing
Objek Uji	SIMT-2-045
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat menyetujui pengajuan judul dosen pembimbing atau menolak pengajuan judul dan dosen pembimbing.
Prosedur Uji	1. Aktor menekan menu pengajuan judul. 2. Aktor menekan tombol setuju untuk menyetujui pengajuan atau menekan tombol tolak dan mengisi alasan penolakan untuk menolak pengajuan judul. 3. Sistem menampilkan konfirmasi 4. Aktor menekan tombol ok. 5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.

Tabel 7.52 Kasus uji menyetujui pengajuan ujian proposal

Nama Kasus Uji	Menyetujui pengajuan ujian proposal
-----------------------	-------------------------------------



Objek Uji	SIMT-2-046
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat menyetujui pengajuan ujian proposal.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu ujian proposal. 2. Aktor menekan tombol setuju. 3. Sistem menampilkan konfirmasi 4. Aktor menekan tombol ok. 5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.

Tabel 7.53 Kasus uji menyetujui pengajuan seminar hasil

Nama Kasus Uji	Menyetujui pengajuan seminar hasil
Objek Uji	SIMT-2-047
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat menyetujui pengajuan seminar hasil
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu seminar hasil. 2. Aktor menekan tombol setuju. 3. Sistem menampilkan konfirmasi 4. Aktor menekan tombol ok. 5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.

Tabel 7.54 Kasus uji menyetujui pengajuan ujian tesis

Nama Kasus Uji	Menyetujui pengajuan ujian tesis
Objek Uji	SIMT-2-048
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat menyetujui pengajuan ujian tesis



Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu ujian tesis. 2. Aktor menekan tombol setuju. 3. Sistem menampilkan konfirmasi 4. Aktor menekan tombol ok. 5. Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Yang Diharapkan	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Hasil Pengujian	Sistem menyimpan data persetujuan dan menampilkan pesan data persetujuan berhasil disimpan
Status Validasi	Valid.

Tabel 7.55 Kasus uji mengisi nilai dan saran perbaikan ujian proposal

Nama Kasus Uji	Mengisi nilai dan saran perbaikan ujian proposal
Objek Uji	SIMT-2-049
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengisi nilai dan saran perbaikan ujian proposal.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu pelaksanaan ujian proposal. 2. Aktor menekan tombol insert nilai. 3. Sistem menampilkan form nilai dan saran perbaikan ujian proposal. 4. Aktor menekan tombol ok. 5. Sistem menyimpan data nilai dan saran perbaikan 6. Sistem menampilkan pesan berhasil
Hasil Yang Diharapkan	Sistem menyimpan data nilai dan saran perbaikan dan menampilkan pesan berhasil
Hasil Pengujian	Sistem menyimpan data nilai dan saran perbaikan dan menampilkan pesan berhasil
Status Validasi	Valid.

Tabel 7.56 Kasus uji mengisi nilai dan saran perbaikan seminar hasil

Nama Kasus Uji	Mengisi nilai dan saran perbaikan seminar hasil
Objek Uji	SIMT-2-050
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengisi nilai dan saran perbaikan seminar hasil.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu pelaksanaan seminar hasil. 2. Aktor menekan tombol insert nilai.



	<ol style="list-style-type: none"> 3. Sistem menampilkan form nilai dan saran perbaikan seminar hasil. 4. Aktor menekan tombol ok. 5. Sistem menyimpan data nilai dan saran perbaikan 6. Sistem menampilkan pesan berhasil
Hasil Yang Diharapkan	Sistem menyimpan data nilai dan saran perbaikan dan menampilkan pesan berhasil
Hasil Pengujian	Sistem menyimpan data nilai dan saran perbaikan dan menampilkan pesan berhasil
Status Validasi	Valid.

Tabel 7.57 Kasus uji mengisi nilai dan saran perbaikan ujian tesis

Nama Kasus Uji	Mengisi nilai dan saran perbaikan ujian tesis
Objek Uji	SIMT-2-051
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aktor dapat mengisi nilai dan saran perbaikan ujian tesis.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan menu pelaksanaan ujian tesis. 2. Aktor menekan tombol insert nilai. 3. Sistem menampilkan form nilai dan saran perbaikan ujian tesis. 4. Aktor menekan tombol ok. 5. Sistem menyimpan data nilai dan saran perbaikan 6. Sistem menampilkan pesan berhasil
Hasil Yang Diharapkan	Sistem menyimpan data nilai dan saran perbaikan dan menampilkan pesan berhasil
Hasil Pengujian	Sistem menyimpan data nilai dan saran perbaikan dan menampilkan pesan berhasil
Status Validasi	Valid.

7.2 Pengujian *Compatibility*

Pengujian *Compatibility* bertujuan untuk mengetahui apakah sistem yang telah dikembangkan dapat digunakan dengan baik dalam *web browser* yang berbeda-beda. Pengujian ini dilakukan menggunakan *software* SortSite versi 5.27.804.0. *Software* tersebut akan melakukan pemeriksaan terhadap struktur sistem yang *compatible* atau tidak *compatible* saat ketika diakses menggunakan *web browser* yang berbeda-beda. Pada Tabel 7.58 dijelaskan arti dari masing-masing simbol yang terdapat pada *software* SortSite

Tabel 7.58 Keterangan simbol pada *software* sortsite

	Tidak ada masalah <i>compatibility</i> yang ditemukan
	Terdapat konten yang tidak dapat ditampilkan atau fungsional yang tidak dapat dijalankan



	Terdapat masalah pada tata letak dan performa pada hampir keseluruhan struktur sistem
	Terdapat masalah pada tata letak dan performa pada sebagian struktu sistem

Pada Gambar 7.1 merupakan hasil dari pengujian compablity yang telah dijalankan pada sistem informasi manajemen tesis. Pada Gambar 7.1 menunjukan bahwa sistem informasi manajemen tesis yang telah dikembangkan memiliki masalah sebagian pada web browser komputer IE versi 9 dan web browser pada perangkat BlackBerry pada veris dibawah 7.1. Namun sebagian web broswer menunjukan bahwa sistem informasi manajemen tesis dapat berjalan dengan baik.

Browser Version	IE			Edge	Firefox	Safari		Opera	Chrome	iOS			Android		BlackBerry	
	9	10	11	15	56	≤ 10	11	47	61	≤ 9	10	11	≤ 3	4*	≤ 7.1	10.0
Critical Issues																
Major Issues																
Minor Issues																

Gambar 7.1 Hasil pengujian *compatibility*



BAB 8 PENUTUP

8.1 Kesimpulan

Berdasarkan penelitian yang sudah dilakukan, maka dapat diambil kesimpulan sebagai berikut:

1. Dari hasil analisis kebutuhan yang telah dilakukan didapatkan sebanyak 52 kebutuhan fungsional dan 1 kebutuhan non fungsional.
2. Dari 52 kebutuhan fungsional, seluruhnya telah diimplementasikan.
3. Hasil pengujian *Black-box* didapatkan hasil sebesar 100% valid dari 52 fitur yang diuji, menunjukkan bahwa sistem yang dikembangkan berjalan sesuai dengan ekspektasi pengguna. Hasil pengujian *compatibility*, sistem sebagian besar berjalan dengan baik pada web browser komputer dan web browser perangkat bergerak namun memiliki masalah sebagaimana pada web browser komputer IE versi 9 dan web browser pada perangkat BlackBerry pada versi dibawah 7.1.

8.2 Saran

1. Untuk pengembangan berikutnya, dapat ditambahkan fitur analisis mahasiswa yang sudah lama tidak ada perkembangan pengerjaan tesis dan fitur notifikasi lewat SMS.
2. Dapat dilakukan integrasi antara sistem informasi manajemen tesis dengan sistem informasi akademik yang sudah ada sehingga data mahasiswa saling terintegrasi.

DAFTAR PUSTAKA

- Adobe Systems Incorporated, 2013. Adobe Air (3.5 beta). [program komputer] Adobe Labs. Tersedia di: <<http://labs.adobe.com/technologies/flashruntimes/air/>> [Diakses 1 Mei 2013]
- Arief, M. Rudyanto. 2011. Pemrograman Web Dinamis menggunakan PHP dan MySQL. Yogyakarta: Andi Publisher.
- Auth0, 2016. *Introduction to JSON Web Tokens: Auth0* [online] Tersedia melalui: JWT.io <<https://jwt.io/introduction/>> [Diakses 9 Mei 2017].
- Ayuba, D., Ismail, A., Hamzah, M.I., 2013. *Evaluation of Page Response Time between Partial and Full Rendering in a Web-based Catalog System*. The 4th International Conference on Electrical Engineering and Informatics (ICEEI 2013), [online] Tersedia di: <<http://www.sciencedirect.com/science/article/pii/S2212017313004167>> [Diakses 8 Mei 2017].
- Brooks, D.R., 2007. *An Introduction to HTML and JavaScript for Scientists and Engineers*. London: Springer-Verlag London Limited.
- Bizagi, 2016. *Bizagi Process Modeler Pengguna's Guide*. [online] Tersedia di: <http://download.bizagi.com/docs/modeler/3100/en/Modeler_pengguna_Guide.pdf> [Diakses 5 July 2017].
- Fakultas Ilmu Administrasi UB, 2016. *Buku Pedoman Akademik*. [pdf] Fakultas Ilmu Administrasi UB. Tersedia di: <http://fia.ub.ac.id/wp-content/uploads/2017/02/Pedoman-Pendidikan-FIA-UB-2016-2017.pdf> [Diakses 1 Juni 2017]
- Businessballs, 2017. *Business process modelling*. [online] Tersedia di: <<http://www.businessballs.com/business-process-modelling.htm>> [Diakses 1 Juni 2017].
- David, C. 2017. *Introduction to cross browser testing*. Mozilla, [online] Tersedia di: <https://developer.mozilla.org/enUS/docs/Learn/Tools_and_testing/Cross_browser_testing/Introduction> [Diakses 5 Mei 2017].
- Deitel, P., Deitel, H., 2012. *Java™ How to Program*. 9th ed. Boston: Prentice Hall.
- Dennis, A., Wixom, B.H & Roth, R.M, 2012. *System Analysis and Design*. 5th ed. Hoboken: John Wiley & Sons.
- Ford, J.L., 2009. *Ajax Programming for the Absolute Beginner*. Boston: Course Technology.
- Hurwitz, J., 2009. *Service Oriented Architecture For Dummies*. Inidana: Wiley Publishing.
- Kadir, A. 2014. *Pengenalan Sistem Informasi Edisi Revisi*. Yogyakarta: Andi



Khadka, S. 2016. *Converting Multipage Application to Single Page Application*. S2. Helsinki Metropolia University.

Kluge, J., Kargl, F., Weber, M., 2007. *The Effects of The Ajax Technology on Web Application Usability*. International Conference on Web Information Systems and Technologies (WEBIST 2007), [online] Tersedia di: <https://www.uni-ulm.de/fileadmin/website_uni_ulm/iui.inst.100/institut/Papers/Prof_Web_er/webist2007.pdf> [Diakses 5 Mei 2017].

Laudon, K.C., Laudon, J.P., 2014. *Management Information Systems Managing The Digital Firm*. 13th ed. Harlow: Pearson Education Limited.

Mills-Senn, P. 2014. *How safe your digital documents*. University Business, [online] Tersedia di: <https://www.universitybusiness.com/article/how-safe-are-your-digital-documents> [Diakses 5 Mei 2017].

Monteiro, F. 2014. *Learning Single-page Web Application Development*. Birmingham: Packt Publishing Ltd.

Opengroup, 2016. *Service-Oriented Architecture*. [online] Tersedia di : <http://www.opengroup.org/soa/source-book/soa/p1.htm> [Diakses 14 Maret 2017].

Program Studi Magister Administrasi Bisnis, 2013. *Sejarah Magister Ilmu Administrasi Bisnis Fakultas Ilmu Administrasi Bisnis Universitas Brawijaya*. [online] Tersedia di: <<http://fia.ub.ac.id/s2bisnis/profil/sejarah>> [Diakses 14 Maret 2017].

Pressman, R.S., 2010. *Software Engineering: A Practitioner's Approach*. 7th ed. New York: McGraw-Hill.

Sommerville, I., 2011. *Software Engineering*. 9th ed. Boston: Pearson Education.

Stair, R.M., Reynolds, G.W., 2012. *Fundamentals of Information Systems*. 6th ed. Boston: Course Technology.

Weske M., 2007. *Business Process Management Concepts, Languages, Architectures*, Berlin: Springer-Verlag.

Yoon C., Sussman A., Memon A., Porter A., 2008. *Effective and Scalable Software Compatibility Testing*, Maryland: University of Maryland.

W3C, 2004a. *Web Service Architecture*. [online] Tersedia di : <<http://www.w3.org/TR/ws-arch/>> [Diakses 13 Maret 2017]

W3C, 2004b. *Web Services Architecture Requirements*. [online] Tersedia di : <<https://www.w3.org/TR/wsa-reqs/>> [Diakses 13 Maret 2017]

W3C, 2005. *HTML & CSS*. [online] Tersedia di : <<https://www.w3.org/standards/webdesign/htmlcss/>> [Diakses 15 Maret 2017]



LAMPIRAN A HASIL WAWANCARA

Hasil wawancara dengan narasumber

Hari/tanggal : Jumat, 08-09-2017
 Waktu : 09.00
 Narasumber : Wempi Naviera
 Jabatan : Pegawai Akademik Magister Administrasi Bisnis

1. Apa saja tahapan tahapan yang harus dilalui mahasiswa untuk bisa menyelesaikan tesis?

Jawab: Mahasiswa pertama harus mengajukan judul dan dosen pembimbing dahulu. Setelah mendapat dosen pembimbing, mahasiswa bisa mulai menyusun. Setelah proposal diselesaikan, mahasiswa bisa menyerahkan proposal tersebut ke bagian akademik untuk diteruskan ke kaprodi. Kaprodi akan memeriksa kelayakan proposal lalu diserahkan ke reviewer untuk mengecek teknik penulisannya. Proposal yang sudah layak dapat diajukan ujian proposal. Progres selanjutnya setelah ujian proposal adalah seminar hasil dan ujian tesis.

2. Bagaimanakah detail dari tiap proses tersebut?

Jawab: Setiap progres yang ada dalam tahapan tesis diurus dengan menggunakan form isian. Untuk detail tahapannya dapat dilihat pada buku pedoman dan SOP yang ada pada fakultas ilmu administrasi.

3. Berapa jumlah dosen pembimbing tiap mahasiswa dan apakah masiswa dapat memilih dosen pembimbingnya sendiri?

Jawab: Setiap mahasiswa akan dibimbing oleh 2 dosen pembimbing. Mahasiswa hanya bisa mengajukan calon dosen pembimbingnya selanjutnya akan dipilihkan oleh kaprodi.

4. Apa saja tipe penelitian yang ada pada Magister Administrasi Bisnis ?

Jawab: Pemasaran, Keuangan, Sistem Informasi Manajemen, Manajemen Sumber Daya Manusia dan Pariwisata.

5. Apa saja syarat mahasiswa bisa mengajukan judul, ujian proposal, seminar hasil dan ujian tesis ?

Jawab: Untuk mengajukan judul dan dosen pembimbing mahasiwa harus memprogram tesis pada KRSnya, menyerahkan fotokopi sertifikat TPA OTO-Bappenas, fotokopi sertifikat TOEFL, fotokopi slip pembayaran. Untuk ujian proposal, syaratnya adalah membuat lembar persetujuan ujian proposal yang ditandatangani oleh 2 dosen pembimbing dan kaprodi. Untuk seminar hasil, syaratnya adalah membuat lembar persetujuan seminar hasil dan serta menyerahkan lembar revisi proposal yang sudah ditandatangani ke akademik. Untuk ujian tesis syaratnya adalah membuat lembar persetujuan ujian tesis yang ditandatangani oleh 2 dosen pembimbing da kaprodi serta menyerahkan lembar revisi dari seminar hasil.

6. Apa saja masalah yang timbul dari alur pengajuan judul hingga ujian tesis saat ini?

Jawab: Saat ini segala proses pengurusan akademik untuk tesis masih manual menggunakan form. Hal ini menyebabkan ada beberapa masalah yang timbul, terutama adalah masalah mengenai pengaksesan informasi. Saat ini segala informasi tentang tesis hanya terdapat di akademik. Dosen tidak mengetahui siapa saja mahasiswa bimbingannya serta sampai mana



progresnya saat ini. Kaprodi juga harus menemui pihak akademik secara langsung untuk mengetahui segala informasi mengenai tesis.

7. Bagaimana cara menangani masalah yang timbul dalam alur pengajuan judul hingga ujian tesis saat ini?

Jawab: Untuk masalah informasi harus menanyakan langsung pada pihak akademik.

8. Bagaimana cara merekap tiap progres tesis saat ini ?

Jawab: Saat ini direkap menggunakan Excell.

9. Bagaimana format form-form yang digunakan dalam tiap proses pengurusan tesis?

Jawab: Format form yang digunakan nanti akan saya berikan contohnya.

Narasumber

Wempi Naviera, M.AB



LAMPIRAN B FORMULIR PROSEDUR TESIS

B.1 Form Pengajuan Judul dan Dosen Pembimbing

KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
FAKULTAS ILMU ADMINISTRASI
UNIVERSITAS BRAWIJAYA
PROGRAM MAGISTER ILMU ADMINISTRASI BISNIS

UKP51 - 1

USULAN KOMISI PEMBIMBING

Kepada: Yth. Ketua
Program Studi Magister Administrasi Bisnis
Fakultas Ilmu Administrasi
Universitas Brawijaya

Yang bertanda tangan dibawah ini :

Nama Mahasiswa :

NIM :

Program Studi : Magister Administrasi Bisnis

Kekhususan/Minat :

Rencana Judul Tesis :

Linkage Penelitian

Dosen :

Tema :

mengajukan nama-nama dosen berikut ini sebagai Komisi Pembimbing saya:

1.
2.
3.
4.
5.



B.2 Form Persetujuan Komisi Pembimbing

KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA
FAKULTAS ILMU ADMINISTRASI
PROGRAM MAGISTER ADMINISTRASI BISNIS

UKPS1 - 2

PERSETUJUAN KOMISI PEMBIMBING

Dengan ini kami menyatakan menjadi pembimbing mahasiswa yang tertera sebagai berikut:

Nama : Chandra Riani
NIM : 156030200111001
Program Studi : Magister Administrasi Bisnis
Judul Tesis :

No.	Nama Dosen	Pembimbing	Tanda Tangan	Tanggal
1.	Prof. Dr. Endang Siti Astuti, M.Si	Ketua
2.	Dr. Hamidah Navati Utami, M.Si	Anggota

Malang,
Menyetujui,
Ketua Program Studi
Magister Adm. Bisnis



B.3 Form Ujian Proposal



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI UNIVERSITAS BRAWIJAYA

FAKULTAS ILMU ADMINISTRASI
Jl. MT. Haryono 163, Malang 65145, Indonesia
Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227
<http://fia.ub.ac.id> E-mail: fia@ub.ac.id

UP - 1

Perihal : Permohonan Ujian Proposal Penelitian Tesis

Kepada : Yth. Ketua Program Studi
Magister Administrasi Bisnis
Fakultas Ilmu Administrasi
Universitas Brawijaya

Bersama ini kami badapkan mahasiswa bimbingan kami

N a m a : Dedy Teguh Haryanto

No. Induk Mahasiswa : 146030202011003

Program Studi : Magister Administrasi Bisnis

Untuk melaksanakan ujian Proposal Tesis dengan

Judul : Implementasi Program Kemitraan Perusahaan dengan UKM untuk
Meningkatkan pertumbuhan Usaha (Studi Kemitraan PT Charoen
Pokphand Indonesia melalui PT Sinar Sarana Sentosa dengan
Peternak Ayam Broiler di Kabupaten Blitar)

yang akan diselenggarakan pada:

H a r i :

T a r i k h :

J a m :

Tempat : Gedung C (Magister) Ruang C.2.09 FIA UB

Demikian, atas perhatiannya kami ucapkan terima kasih.

Malang,
Ketua Komisi Pembimbing

Dr. Hamidah Nayati Utami, M.Si
NIP. 19721117 199802 2 001



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA
FAKULTAS ILMU ADMINISTRASI
 Jl. MT. Haryono 163, Malang 65145, Indonesia
 Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227
 http://fia.ub.ac.id E-mail: fia@ub.ac.id

UP - 2

Nomor : /UN10.F03.12.21/PN/2017
 Lampiran : -
 Perihal : Undangan Ujian Proposal Penelitian Tesis

Kepada : Yth. Dr. Hamidah Nayati Utami, M.Si
 (Ketua Komisi Pembimbing)
 Fakultas Ilmu Administrasi,
 Universitas Brawijaya

Sehubungan dengan rencana Ujian Proposal Penelitian Tesis dari mahasiswa berikut:

Nama : Dedy Teguh Haryanto
 No. Induk Mahasiswa : 146030202011003
 Program Studi : Magister Administrasi Bisnis

yang akan diselenggarakan pada :

Hari : |
 Tanggal :
 Jam : WIB
 Tempat : Gedung C (Magister) Ruang C.2.09 FIA UB

kami mohon kehadiran Bapak/Ibu pada Ujian Proposal Penelitian Tesis sesuai dengan jadwal di atas.
 Demikian atas perhatian dan kerjasamanya diucapkan terima kasih.

Malang
 Ketua,
 Prog. Studi Magister Adm. Bisnis,

Dr. Kusdi Raharjo, DEA
 NIP. 195701271984031001

Catatan:

1. DOSEN PENGUJI DAN KASUS DOSEN
2. Jika Dosen Penguji berhalangan akan digantikan penguji lain oleh KPS

**KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA**

FAKULTAS ILMU ADMINISTRASI

Jl. MT. Haryono 163, Malang 65145, Indonesia

Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227

http://fia.ub.ac.id E-mail: fia@ub.ac.id

UP - 3

PERSetujuan UJIAN PROPOSAL PENELITIAN TESIS

Yang bertandatangan dibawah ini, tim penguji menyetujui Ujian Proposal Penelitian Tesis mahasiswa

N a m a : Dedy Teguh Haryanto
Nomor Induk Mahasiswa : 146030202011003
Program Studi : Magister Administrasi Bisnis
Kekhususan : -
Judul : Implementasi Program Kemitraan Perusahaan dengan UKM untuk Meningkatkan pertumbuhan Usaha (Studi Kemitraan PT Charoen Pokphand Indonesia melalui PT Sinar Sarana Sentosa dengan Peternak Ayam Broiler di Kabupaten Blitar)

yang akan diselenggarakan pada :

H a r i :
T a n g a l :
J a m :
T e m p a t : Gedung C (Magister) Ruang C.2.09 FIA UB

No.	Nama-nama Tim Penguji	Tanda tangan	Tgl. Persetujuan
1.	Dr. Hamidah Nayati Utami, M.Si.....	<u>Ketua Komisi</u>
2.	Arik Prasetya, S.Sos, M.Si.,Ph.D	<u>Anggota Komisi</u>
3.	<u>Dosen Penguji</u>
4.	<u>Dosen Penguji</u>

Malang,
Ketua,
Prog. Studi Magister Adm. Bisnis,

Dr. Kusdi Raharjo, DEA
NIP. 195701271984031001

**KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA**

FAKULTAS ILMU ADMINISTRASI
 Jl. MT. Haryono 163, Malang 65145, Indonesia
 Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227
 http://fia.ub.ac.id E-mail: fia@ub.ac.id

UP - 4

Nomor : /UN10.F03.12.21/PN/2017
 Lembaran : -
 Perihal : Undangan Ujian Proposal Penelitian Tesis

Kesada : Yth.
 (Dosen Penguji)
 Fakultas Ilmu Administrasi,
 Universitas Brawijaya

Sehubungan dengan rencana Ujian Proposal Penelitian Tesis dari mahasiswa berikut.:

Nama : Dedy Teguh Haryanto
 Nomor Induk Mahasiswa : 146030202011003
 Program Studi : Magister Administrasi Bisnis

yang akan diselenggarakan pada :

Hari :
 Tanggal :
 Jam : WIB
 Tempat : Gedung C (Magister) Ruang C.2.09 FIA UB

kami mohon kehadiran Bapak/Ibu pada Ujian Proposal Penelitian Tesis sesuai dengan jadwal di atas.

Demikian atas perhatian dan kerjasamanya diucapkan terima kasih.

Malang
 Ketua,
 Prog. Studi Magister Adm. Bisnis,

Dr. Kusdi Raharjo, DEA
 NIP. 195701271984031001

Catatan:

1. Dosen Penguji ditunjuk oleh Dekan
2. Jika Dosen Penguji berhalangan akan digantikan penguji lain oleh KPS



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA
FAKULTAS ILMU ADMINISTRASI
 Jl. MT. Haryono 163, Malang 65145, Indonesia
 Telp. +62-341-553737, 568914, 558226 Fax : +62-341-558227
 http://fia.ub.ac.id E-mail: fia@ub.ac.id

UP - 8

BERITA ACARA UJIAN PROPOSAL PENELITIAN TESIS

Mahasiswa tersebut dibawah ini:

N a m a : Dedy Teguh Haryanto
Nomor Induk Mahasiswa : 146030202011003
Program Studi : Magister Administrasi Bisnis
Kekhususan : -
Judul : Implementasi Program Kemitraan Perusahaan dengan UKM untuk Meningkatkan pertumbuhan Usaha (Studi Kemitraan PT Charoen Pokphand Indonesia melalui PT Sinar Sarana Sentosa dengan Peternak Ayam Broiler di Kabupaten Blitar)

yang diselenggarakan pada :

H a r i :
T a n g g a l :
J a m : WIB
N i l a i :

N a m a		Nilai (Angka)	Keterangan *)
Ketua	1. Dr. Hamidah Nayati Utami, M.Si
Anggota	2. Arik Prasetya, S.Sos, M.Si., Ph.D
Anggota	3.
Anggota	4.
Rerata	

*) Catatan: Ujian/tes/serah terima proposal pada UP-8

Nilai huruf hanya untuk rerata dan diilakukan pada SK Rektor 078/SK/1005

Nilai Huruf	Skala	Bobot
>80 - 100	A	4,0
>75 - 80	B+	3,5
>69 - 75	B	3,0
>63 - 69	C+	2,5
>57 - 63	C	2,0
>50 - 57	D+	1,5
>44 - 50	D	1,0
0-44	E	0

*) Ujian yang dilakukan

Malang,
 Tim Penguji
 Ketua,

Dr. Hamidah Nayati Utami, M.Si
 NIP. 19721117 199802 2 001



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA
FAKULTAS ILMU ADMINISTRASI
 Jl. MT. Haryono 163, Malang 65145, Indonesia
 Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227
 http://fia.ub.ac.id E-mail: fia@ub.ac.id

UP - 2

NILAI UJIAN PROPOSAL PENELITIAN TESIS

Nama : Dedy Teguh Haryanto
 Nomor Induk Mahasiswa : 146030202011003
 Program Studi : Magister Administrasi Bisnis

KOMPONEN PENILAIAN	NILAI (ANGKA)
1. Penguasaan materi proposal
2. Kemampuan penalaran
Rerata

Malang,
 Pembimbing I,

Dr. Hamidah Nayati Utami, M.Si
 NIP. 19721117 199802 2 001



B.4 Form Seminar Hasil



KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI UNIVERSITAS BRAWIJAYA

FAKULTAS ILMU ADMINISTRASI

Jl. MT. Haryono 163, Malang 65145, Indonesia

Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227

http://fia.ub.ac.id

E-mail: fia@ub.ac.id

SMP 1

Perihal : Permohonan Seminar Hasil

Kepada : Yth. Ketua Program Studi
Magister Administrasi Bisnis FIA
Universitas Brawijaya
Malang

Bersama ini kami hadapkan mahasiswa bimbingan kami :

Nama : Nadiya Lifa Ningrum
No. Mahasiswa : 146030201111007
Program Studi : ~~Magister Administrasi Bisnis~~ Magister Administrasi Bisnis
Judul : Implementasi Community Development dan Knowledge Sharing
dalam Meningkatkan Kualitas Sumber Daya Manusia Wisata Pantai
Bangsring

untuk melaksanakan seminar Hasil Penelitian yang akan diselenggarakan pada :

Hari : Selasa
Tanggal : 06 Juni 2017
Jam :
Tempat : Gedung C (Magister) Ruang C.2.10 FIA UB

Demikian, atas perhatiannya kami ucapkan terima kasih.

Malang,
Komisi Pembimbing,
Ketua,

Dr. Hamidah Nayati Utami, M.Si
NIP. 19721117 199802 2 001



KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA

FAKULTAS ILMU ADMINISTRASI

Jl. MT. Haryono 163, Malang 65145, Indonesia

Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227

http://fia.ub.ac.id E-mail: fia@ub.ac.id

SHP 2

Nomor : /UN10.F03.12.21/PN/2017
Lampiran : -
Perihal : Undangan Seminar Hasil

Kepada : Yth. **Dr. Hamidah Nayati Utami, M.Si**
(Komisi Pembimbing)
Fakultas Ilmu Administrasi
Malang

Bersama ini kami mengharap kehadiran Bapak / Ibu pada Seminar Hasil Penelitian dari :

Nama : Nadiya Lifa Ningrum
No. Mahasiswa : 146030201111007
Program Studi : Magister ~~Administrasi Bisnis~~ **Administrasi Bisnis**

yang akan diselenggarakan pada :

Hari : selasa
Tanggal : 06 Juni 2017
Jam : 09.00 WIB
Tempat : Gedung C (Magister) Ruang C.2.10 FIA UB

Demikian, atas perhatiannya kami ucapkan terima kasih.

Malang,
Ketua,
Prog. Studi Magister Adm. Bisnis,

Dr. Kusdi Raharjo, DEA
NIP. 195701271984031001



**KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA**

FAKULTAS ILMU ADMINISTRASI
Jl. MT. Haryono, 163, Malang 65145, Indonesia
Telp. : +62-341-553737, 558914, 558226 Fax : +62-341-558227
http://fia.ub.ac.id E-mail: fia@ub.ac.id

SHP 3

PERSETUJUAN SEMINAR HASIL PENELITIAN TESIS

Yang bertanda tangan dibawah ini, Komisi Pembimbing menyetujui seminar hasil penelitian tesis mahasiswa :

N a m a : Nadiya Lifa Ningrum
No. Mahasiswa : 146030201111007
Program Studi : Magister Administrasi Bisnis
Kekhususan : Pengembangan SDM
Judul : Implementasi Community Development dan Knowledge Sharing dalam Meningkatkan Kualitas Sumber Daya Manusia Wisata Pantai Bangsring

Yang akan diselenggarakan pada :

H a r i : Selasa
Tanggal : 06 Juni 2017
J a m : 09.00
Tempat : Gedung C (Magister) Ruang C.2.10 FIA UB

No.	Komisi Pembimbing	Tanda tangan	Tgl. Persetujuan
1)	Dr. Hamidah Nayati Utami, M.Si..... Ketua
2)	Yusri Abdillah, S.Sos, M.Si., Ph.D Anggota

Malang,
Ketua,
Prog. Studi Magister Adm. Bisnis,
.....

Dr. Kusdi Raharjo, DEA
NIP. 195701271984031001

) Coret yang tidak perlu

**KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA****FAKULTAS ILMU ADMINISTRASI**

Jl. MT. Haryono 163, Malang 65145, Indonesia

Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227

http://fia.ub.ac.id E-mail: fia@ub.ac.id

SHP 4

Nama : /UN10.F03.12.21/PN/2017
Lampiran : -
Perihal : Pemberitahuan Seminar Hasil Penelitian

Kepada : Yth. Mahasiswa,
 Program Studi Magister Administrasi Bisnis,
 FIA Universitas Brawijaya
 Malang

Bersama ini kami menyampaikan kehadiran Saudara pada Seminar Hasil Penelitian dari.

Nama : Nadiya Lifa Ningrum
No. Mahasiswa : 146030201111007
Program Studi : Magister Administrasi Bisnis,
Judul : Implementasi Community Development dan Knowledge Sharing
dalam Meningkatkan Kualitas Sumber Daya Manusia Wisata Pantai
Bangsring

Yang akan dilaksanakan pada.

Ha ri : selasa
Tanggal : 06 Juni 2017
Ja m : 09.00 WIB
Tempat : Gedung C (Magister) Ruang C.2.10 FIA UB

Demikian, untuk diperhatikan.

Malang,
 Ketua,
 Prog. Studi Magister Adm. Bisnis,

Dr. Kusdi Raharjo, DEA
 NIP. 195701271984031001



KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA
FAKULTAS ILMU ADMINISTRASI
 Jl. MT. Haryono 163, Malang 65145, Indonesia
 Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227
 http://fia.ub.ac.id E-mail: fia@ub.ac.id

SHP 5

BERITA ACARA SEMINAR HASIL

Penyelenggaraan seminar hasil dari mahasiswa yang tersebut :

N a m a : Nadiya Lifa Ningrum
No. Mahasiswa : 146030201111007
Program Studi : Magister Administrasi Bisnis
Kekhususan : Pengembangan SDM
JudulTesis : Implementasi Community Development dan Knowledge Sharing dalam Meningkatkan Kualitas Sumber Daya Manusia Wisata Pantai Bangsring
Yang diselenggarakan pada :
H a r i : Selasa
T a r i x : 06 Juni 2017
J a m : 09.00 WIB
Tempat : Gedung C (Magister) Ruang C.2.10 FIA UB
Nilai :

Penilaian oleh Komisi		Nilai (Angka)	Nilai (Huruf)
Ketua :	Dr. Hamidah Nayati Utami, M.Si
Anggota :	Yusri Abdillah, S.Sos, M.Si., Ph.D
Berita			

Demikian untuk menjadi periksa adanya.

Nilai huruf hanya untuk neraca dan didasarkan pada SK Rektor 078/150/1005

Nilai Angka	Nilai Huruf	Nilai
>80 - 100	A	4,0
>75 - 80	B+	3,5
>69 - 75	B	3,0
>63 - 69	C+	2,5
>55 - 60	C	2,0
>50 - 55	D+	1,5
>44 - 50	D	1,0
0 - 44	E	0

Catatan: Nilai hasil diserahkan langsung kebagian pengajaran

Malang, 06 Juni 2017

Komis Pembimbing
Ketua,

Dr. Hamidah Nayati Utami, M.Si
 NIP. 19721117 199802 2 001



KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA
FAKULTAS ILMU ADMINISTRASI
 Jl. MT. Haryono 163, Malang 65145, Indonesia
 Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227
 http://fia.ub.ac.id E-mail: fia@ub.ac.id

SH 2.6

PENILAIAN SEMINAR HASIL

Berdasarkan hasil seminar hasil yang telah dilakukan oleh mahasiswa yang tersebut dibawah ini :

N a m a : Nadiya Lifa Ningrum
No. Mahasiswa : 146030201111007
Program Studi : Magister Administrasi Bisnis
Judul Tesis : Implementasi Community Development dan Knowledge Sharing dalam Meningkatkan Kualitas Sumber Daya Manusia Wisata Pantai Bangsring

Maka dapat dilakukan penilaian sebagai berikut :

Komponen Penilaian	Nilai (Angka)
1. Draf Naskah
2. Penyajian Naskah
3. Diskusi
Rerata Nilai

Demikian, untuk diketahui oleh pihak-pihak yang terkait.

Malang, 06 Juni 2017
 Komisi Pembimbing
 Ketua,

Dr. Hamidah Nayati Utami, M.Si
 NIP.19721117 199802 2 001

Catatan :
 Nilai seminar hasil diserahkan langsung kebagian pengajaran



**KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA**

FAKULTAS ILMU ADMINISTRASI
Jl. MT. Haryono 163, Malang 65145, Indonesia
Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227
http://fia.ub.ac.id E-mail: fia@ub.ac.id

SHP 7

PENILAIAN SEMINAR HASIL

Berdasarkan hasil seminar hasil yang telah dilakukan oleh mahasiswa yang tersebut dibawah ini :

N a m a : Nadiya Lifa Ningrum
No. Mahasiswa : 146030201111007
Program Studi : Magister **Administrasi Bisnis**
Judul Tesi : Implementasi Community Development dan Knowledge Sharing dalam Meningkatkan Kualitas Sumber Daya Manusia Wisata Pantai Bangsring

Maka dapat dilakukan penilaian sebagai berikut.

Komponen Penilaian	Nilai (Ganda)
1. Draf Naskah
2. Penyajian Naskah
3. Diskusi
Rerata Nilai

Demikian, untuk diketahui oleh pihak-pihak yang terkait.

Malang, 06 Juni 2017
Komisi Pembimbing,
Anggota

Yusri Abdillah, S.Sos, M.Si, Ph.D
NIP. 19741227 200312 1 002

Catatan :
Nilai seminar hasil diserahkan langsung kebagian pengajaran



B.5 Form Ujian Tesis



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI UNIVERSITAS BRAWIJAYA

FAKULTAS ILMU ADMINISTRASI

Jl. MT. Haryono 163, Malang 65145, Indonesia

Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227

<http://fia.ub.ac.id>

E-mail: fia@ub.ac.id

UT - 1

Perihal : Permohonan Ujian Tesis

Kepada : Yth. Ketua
Program Magister Administrasi Bisnis
Fakultas Ilmu Administrasi,
Universitas Brawijaya.

Bersama ini kami hadapkan mahasiswa bimbingan kami:

Nama : Ridha Rahmatiyah

No. Mahasiswa : 146030201111016

Program Studi : Magister Administrasi Bisnis

Judul Tesis : Pengaruh Brand Satisfaction, Terhadap Brand Attachment, Brand Trust,
Dan Brand Loyalty

Untuk melaksanakan Ujian Tesis, dan kami mohon untuk diselenggarakan pada :

Hari : Jum'at

Tanggal : 12 Mei 2017

Jam : 08.00 WIB

Tempat : Gedung C (Magister) Ruang C.2.09 FIA UB

Demikian, atas perhatiannya kami ucapkan terima kasih.

Malang,
Komisi Pembimbing
Ketua,

Dr. Drs. Edy Yulianto, M.P
NIP. 196007281986011001



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA
FAKULTAS ILMU ADMINISTRASI
Jl. MT. Haryono, 163, Malang 65145, Indonesia
Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227
http://fia.ub.ac.id E-mail: fia@ub.ac.id

UT - 2

Nomor : /UN10.F03.12.21/PN/2017
Lampiran : -
Perihal : Tim Penguji Tesis

Kepada : Yth. Prof. Dr. Suharyono, MA
(Tim Penguji)
Fakultas Ilmu Administrasi,
Universitas Brawijaya.

Sehubungan dengan rencana ujian tesis dari mahasiswa berikut:

Nama : Ridha Rahmatiyah
No. Mahasiswa : 146030201111016
Program Studi : Magister Administrasi Bisnis

yang akan diselenggarakan pada :

Hari : Jum'at
Tanggal : 12 Mei 2017
Jam : 08.00 WIB
Tempat : Gedung C (Magister) Ruang C.2.09 FIA UB

kami mohon kehadiran Bapak/Ibu pada Ujian Tesis sesuai dengan jadwal di atas.
Demikian atas perhatian dan kerjasamanya diucapkan terima kasih.

Malang,
Ketua,
Prog. Studi Magister Adm. Bisnis,

Dr. Kusdi Raharjo, DEA
NIP. 195701271984031001

**KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA****FAKULTAS ILMU ADMINISTRASI**

Jl. MT. Haryono 163, Malang 65145, Indonesia

Telp. : +62-341-563737, 568914, 558226 Fax : +62-341-558227

http://fia.ub.ac.id

E-mail: fia@ub.ac.id

UF - 3

PERSETUJUAN UJIAN TESIS

Yang bertanda tangan dibawah ini, tim penguji menyetujui Ujian Tesis mahasiswa :

N a m a : Ridha Rahmatiyah
No. Mahasiswa : 146030201111016
Program Studi : Magister Administrasi Bisnis
Kebudayaan : Kebijakan Bisnis
Judul Tesis : Pengaruh Brand Satisfaction, Terhadap Brand Attachment, Brand Trust, Dan Brand Loyalty

yang akan dilaksanakan pada:

H a g i : Jum'at
Tanggal : 12 Mei 2017
J a m : 08.00 WIB
Tempat : Gedung C (Magister) Ruang C.2.09 FIA UB

No.	Nama Tim Penguji	Tanda tangan	Tgl. Persetujuan
1)	Dr. Drs. Edy Yulianto, M.P.....	Ketua
2)	Andriani Kusumawati, S.Sos, M.Si , DBA.....	Anggota
3)	Prof. Dr. Suharyono, MA.....	Dosen Penguji
4)	Edriana Pangestuti, SE, M.Si, DBA.....	Dosen Penguji

Malang,
Ketua,
Prog. Studi Magister Adm. Bisnis,

Dr. Kusdi Raharjo, DEA
NIP. 195701271984031001



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA
FAKULTAS ILMU ADMINISTRASI
Jl. MT. Haryono 163, Malang 65145, Indonesia
Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227
http://fia.ub.ac.id E-mail: fia@ub.ac.id

UI - 4

Nomor : /UN10.F03.12.21/PN/2017
Lampiran : -
Perihal : Undangan Ujian Tesis
Kepada : Yth. Dr. Drs. Edy Yulianto, M.P
(Komisi Pembimbing)
~~Fakultas Ilmu Administrasi~~
Malang

Sehubungan dengan rencana ujian tesis dari mahasiswa berikut:

Nama : Ridha Rahmatiyah
No. Mahasiswa : 146030201111016
Program Studi : Magister ~~Administrasi~~ Bisnis

yang akan diselenggarakan pada :

Hari : Jum'at
Tanggal : 12 Mei 2017
Jam : 08.00WIB
Tempat : Gedung C (Magister) Ruang C.2.09 FIA UB

kami mohon kehadiran Bapak/Ibu pada Ujian Tesis sesuai dengan jadwal di atas.
~~Demikian atas perhatian dan kerjasamanya diucapkan terima kasih.~~

Malang,
Ketua,
Prog. Studi Magister Adm. Bisnis,

Dr. Kusdi Raharjo, DEA
NIP. 195701271984031001



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA
FAKULTAS ILMU ADMINISTRASI
 Jl. MT. Haryono 163, Malang 65145, Indonesia
 Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227
 http://fia.ub.ac.id E-mail: fia@ub.ac.id

UT - 8

BERITA ACARA UJIAN TESIS

Mahasiswa Program Magister tersebut dibawah ini :

N a m a : Ridha Rahmatiyah
No. Mahasiswa : 146030201111016
Program Studi : Magister Administrasi Bisnis
Kekhususan : Kebijakan Bisnis
Judul : Pengaruh Brand Satisfaction, Terhadap Brand Attachment, Brand Trust, Dan Brand Loyalty

yang diselenggarakan pada :

H a r i : Jum'at
T a r i k h : 12 Mei 2017
J a m : 08.00 WIB
T e m p a t : Gedung C (Magister) Ruang C.2.09 FIA UB FIA - UB
N i l a i :

N a m a		N i l a i (Angka)	N i l a i (Huruf*)
Ketua	1. Dr. Drs. Edy Yulianto, M.P		
Anggota	2. Andriani Kusumawati, S.Sos, M.Si, DBA		
Anggota	3. Prof. Dr. Suharyono, MA		
Anggota	4. Edriana Pangestuti, SE, M.Si, DBA		
Rerata			

Catatan: Lulus/lulus dengan syarat **/tidak lulus ***)

Malang, 12 Mei 2017
 Tim Penguji
 Ketua,

Dr. Drs. Edy Yulianto, M.P
 NIP. 196007281986011001

*) Nilai huruf hanya untuk rerata dan didasarkan pada SK Rektor 076/SK/1005

Nilai Angka	Nilai Huruf	Bidat
>80 - 100	A	4.0
>75 - 80	B+	3.5
>60 - 75	B	3.0
>60 - 69	C+	2.5
>55 - 60	C	2.0
>50 - 55	D+	1.5
>45 - 50	D	1.0
0-44	F	

*) Mahasiswa harus mengikuti ujian tesik, paling cepet 1 bulan setelah ujian pertama.



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA
FAKULTAS ILMU ADMINISTRASI
Jl. MT. Haryono 163, Malang 65145, Indonesia
Telp. : +62-341-553737, 558914, 558226 Fax : +62-341-558227
http://fia.ub.ac.id E-mail: fia@ub.ac.id

UT - 7

NILAI UJIAN TESIS

Nama : Ridha Rahmatiyah
No. Mahasiswa : 146030201111016
Program Studi : Magister Administrasi Bisnis
Judul Tesis : Pengaruh Brand Satisfaction, Terhadap Brand Attachment, Brand Trust, Dan Brand Loyalty

KOMPONEN PENILAIAN	NILAI (ANGKA)
1. Draft tesis
2. Penilaian tesis
3. Kemampuan mempertahankan tesis
RERATA

Malang, 12 Mei 2017
Penguji,

Dr. Drs. Edy Yulianto, M.P
NIP.196007281986011001



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
UNIVERSITAS BRAWIJAYA
FAKULTAS ILMU ADMINISTRASI
Jl. MT. Haryono 163, Malang 65145, Indonesia
Telp. : +62-341-553737, 568914, 558226 Fax : +62-341-558227
http://fia.ub.ac.id E-mail: fia@ub.ac.id

UT - 7

Saran UT
Ridha Rahmatiyah
146030201111016
Reg.1 - Kecipakan Biru - Garis

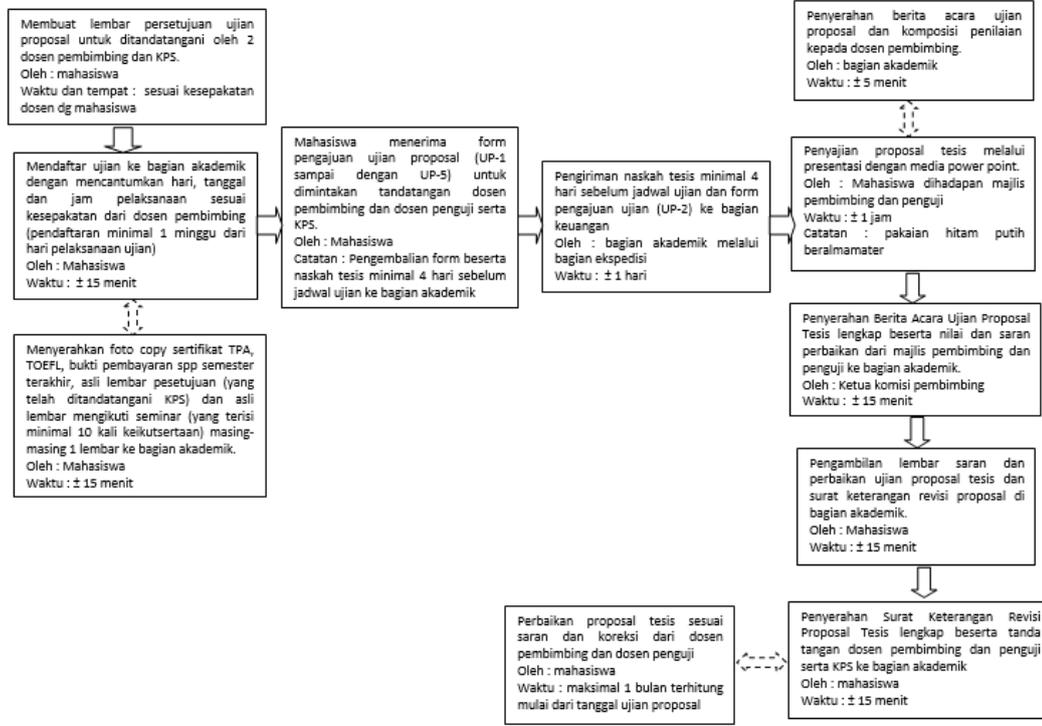
Saran Perbaikan Tesis

Area for providing thesis improvement suggestions, consisting of multiple horizontal lines.

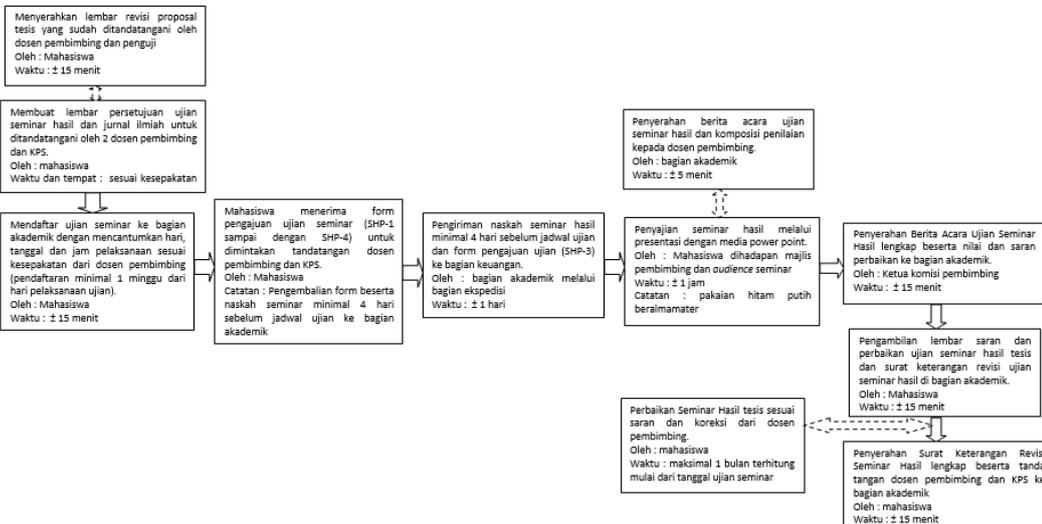
Demikian, untuk menjadikan periksa adanya.

LAMPIRAN C PROSEDUR TESIS

C.1 Prosedur Ujian Proposal

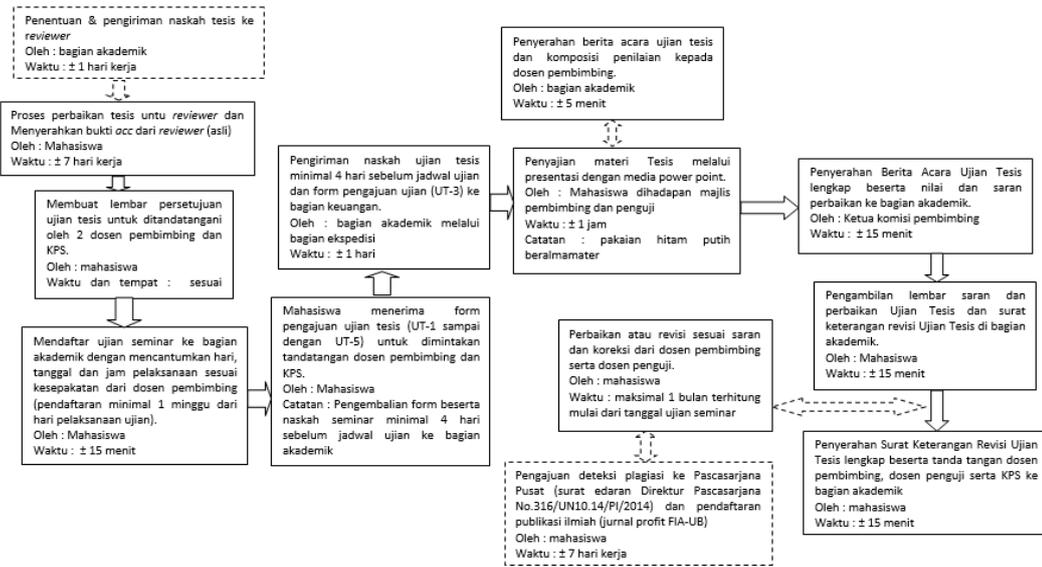


C.2 Prosedur Seminar Hasil





C.3 Prosedur Ujian Tesis





LAMPIRAN D KODE SUMBER

D.1 Kode Sumber Vue komponen pengajuan_proposal_list

```

1 <template lang="html">
2   <div class="layout-content-main">
3     <div class="">
4       <Card :bordered="false">
5         <p slot="title">Pengajuan Judul dan dosen Pembimbing</p>
6         <span slot="extra">
7           <span>
8             <router-link v-if="bisa_pengajuan_proposal"
9               to="/pengajuan-judul/Pengajuan-Proposal">
10              <Button type="primary" icon="plus">
11                Ajukan Judul
12              </Button>
13            </router-link>
14          </span>
15          <Spin v-if="ajax_fetch_data"></Spin>
16          <template v-else>
17            <div v-if="pengajuan_proposal.length == 0" class="tidak-
18              ada-data">
19              <div style="text-align:center">
20                <Icon type="document-text"></Icon>
21                <br>
22                <span>Data pengajuan judul tidak ditemukan</span>
23              </div>
24            <PengajuanProposalCard
25              v-else
26              v-for="(item,index) in pengajuan_proposal"
27              :key="item.id_pengajuan"
28              :prop_index="index"
29              :prop_data_pengajuan="item"
30              prop_type="full mode"
31              :prop_filter="fiter_pengajuan_proposal">
32            </PengajuanProposalCard>
33          </template>
34        </Card>
35        <!-- komponen modal pilih dosen -->
36        <modal-pilih-dosen

```



```
37     v-if="$store.state.profile.pegawai"
38     tipe="persetujuan_proposal"
39     :modal_dosen="$store.state.modal_pilih_pembimbing.show"
40     :dosen="$store.state.modal_pilih_pembimbing.dosen_pilihan"
41     :row_dosen="$store.state.modal_pilih_pembimbing.row_dosen">
42     </modal-pilih-dosen>
43 </div>
44 </div>
45 </template>
46
47 <script>
48 import PengajuanProposalCard from './Pengajuan_Proposal_Card.vue';
49 import ModalPilihDosen from './Modal-pilih-dosen.vue';
50 export default {
51   name: 'pengajuan_proposal_list',
52   components: {
53     PengajuanProposalCard,
54     ModalPilihDosen
55   },
56   data: function() {
57     return {
58       ajax_fetch_data: '',
59       fiter_pengajuan_proposal: 'all',
60       pengajuan_proposal: [],
61     }
62   },
63   created: function() {
64     this.set_tipe_modal();
65     this.get_pengajuan_proposal();
66   },
67   computed: {
68     bisa_pengajuan_proposal: function() {
69       if(this.$store.state.profile.status_pengajuan_proposal ==
70       'pengajuan baru') {
71         return true;
72       }
73       else {
74         return false;
75       }
76     },
```



```
77     methods: {
78         get_pengajuan_proposal: function(){
79             var self = this;
80             self.ajax_fetch_data = true;
81             this.$store.dispatch('get_pengajuan_proposal').then((progress) =>
82             {
83                 console.log('pengajuan proposal from ajax');
84                 self.pengajuan_proposal = progress;
85                 if(self.pengajuan_proposal.length == 0){
86                     self.ajax_fetch_data = false;
87                 }
88                 else{
89                     self.pengajuan_proposal.forEach(function(){
90                         self.$store.commit('push_modal_dosen_pembimbing');
91                     });
92                     self.get_dosen();
93                 }
94             });
95         },
96         get_dosen: function(){
97             var self = this;
98             if(this.$store.state.dosen.length == 0){
99                 console.log(this.$store.state.dosen.length);
100                 console.log('get dosen from ajax');
101                 this.$store.dispatch('get_all_dosen').then(() => {
102                     self.ajax_fetch_data = false;
103                 });
104             }
105             else{
106                 self.ajax_fetch_data = false;
107             }
108         },
109         set_tipe_modal: function(){
110             this.$store.commit('set_tipe_modal','peretujuan_proposal');
111         }
112     }
113 }
114 }
115 </script>
```



D.2 Kode Sumber Vue komponen pengajuan_proposal_form

```

1 <template lang="html">
2   <form @submit.prevent="submit_pengajuan">
3     <div class="form-group">
4       <label for="judul_tesis">Judul Tesis</label>
5       <input v-model="form_pengajuan.judul_tesis">
6     </div>
7     <div class="form-group">
8       <label for="tipe_penelitian">Tipe Penelitian</label>
9       <select v-model="form_pengajuan.tipe_penelitian"
10        name="tipe_penelitian">
11         <option value="" disabled>Pilih tipe penelitian</option>
12         <option v-for="bm in bidang_minat"
13           :value="bm.id_bidang_minat">{{ bm.nama_bidang_minat }}</option>
14       </select>
15     </div>
16     <div class="form-group linkage-group">
17       <label for="judul_tesis">Linkage Peneltian</label>
18       <label for="linkage_dosen">Dosen</label>
19       <input v-model="form_pengajuan.linkage_dosen"
20        name="linkage_dosen">
21       <label for="linkage_tema">Tema</label>
22       <input v-model="form_pengajuan.linkage_tema">
23     </div>
24     <div class="form-row">
25       <div class="col-md-12">
26         <label for="dosen_1">Calon Dosen Pembimbing 1</label>
27         <input v-model="form_pengajuan.dosen_1.nama_pegawai"
28          name="dosen_pembimbing_1">
29       </div>
30     </div>
31     <div class="form-row">
32       <div class="col-md-12">
33         <label for="dosen_2">Calon Dosen Pembimbing 2<span
34          class="required-star">*</span></label>
35         <input v-model="form_pengajuan.dosen_2.nama_pegawai"
36          name="dosen_pembimbing_2">
37       </div>
38     </div>
39     <div class="form-row">
40       <div class="col-md-12">
41         <label for="dosen_3">Calon Dosen Pembimbing
42         (Opsional)</label>

```



```
36     <input v-model="form_pengajuan.dosen_3.nama_pegawai"
37     type="text">
38   </div>
39 </div>
40 <div class="form-row">
41   <div class="col-md-12">
42     <label for="dosen_4">Calon Dosen Pembimbing
43     (Opsional)</label>
44     <input v-model="form_pengajuan.dosen_4.nama_pegawai">
45   </div>
46 </div>
47 <div class="form-row">
48   <div class="col-md-12">
49     <label for="dosen_5">Calon Dosen Pembimbing
50     (Opsional)</label>
51     <input v-model="form_pengajuan.dosen_5.nama_pegawai">
52   </div>
53 </div>
54 <Button :loading="ajax_submit" html-type="submit"
55 </form>
56 </template>
57 <script>
58 import ModalPilihDosen from './Modal-pilih-dosen.vue';
59 export default {
60   name: 'PengajuanProposalForm',
61   props: ['id_proposal'],
62   data: function() {
63     return {
64       row_dosen: '',
65       bidang_minat: [],
66       form_pengajuan: {
67         judul_tesis: '',
68         tipe_penelitian: '',
69         linkage_dosen: '',
70         linkage_tema: '',
71         dosen_1: '',
72         dosen_2: '',
73         dosen_3: '',
74         dosen_4: '',
75         dosen_5: '',
76       },
77       ajax_submit: false,
```



```
75     ajax_fetch_data: true,
76     modal_dospem: {
77         nomor_modal: '',
78         status_modal: false
79     }
80 }
81 },
82 components: {
83     ModalPilihDosen
84 },
85 methods: {
86     submit_pengajuan: function(){
87         var self = this;
88         $.ajax({
89             type: 'POST',
90             url:
91 "http://localhost/skripsi/index.php/Pengajuan_Proposal",
92             data: {token: this.$store.state.token, data:
93 self.form_pengajuan},
94             beforeSend: function(){
95                 self.ajax_submit = true;
96             },
97             success: function(response){
98                 if(response.status == 'Berhasil'){
99                     self.$Notice.success({
100                         title: 'Pengajuan berhasil disubmit!',
101                     });
102                     self.ajax_submit = true;
103                     self.$router.push('/pengajuan-judul');
104                 }
105             }
106         })
107     },
108 }
109 }
110 }
111 }
112 }
113 }
114 }
115 }
116 }
117 }
118 </script>
```