

RANCANG BANGUN SISTEM KEMAHASISWAAN BERBASIS WEB DI STMIK"AMIKBANDUNG"

SKRIPSI

Diajukan Untuk Memenuhi Sebagian dari Syarat
Kelulusan Program Pendidikan Strata I pada Program Studi Teknik Informatika
Sekolah Tinggi Manajemen Informatika dan Komputer
"AMIKBANDUNG"



Disusun oleh:

YUDA SAPUTRA

2042069

**PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
"AMIKBANDUNG"**

LEMBAR PENGESAHAN
RANCANG BANGUN SISTEM KEMAHASISWAAN
BERBASIS WEB DI STMIK "AMIKBANDUNG"

Disusun oleh:

YUDA SAPUTRA

2042069

Telah disahkan di:

Bandung

Pada tanggal:

07 Februari 2025

Menyetujui,

Ketua Program Studi
Teknik Informatika

Pembimbing

Muhammad Abdul Mujib, M. T.
NIP. 2019090019

Shafira Febriani, M. T.
NIP. 2024070009

PROGRAM STUDI SISTEM INFORMASI
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
"AMIKBANDUNG"

SURAT PERNYATAAN

Saya menyatakan dan bertanggungjawab dengan sebenarnya bahwa skripsi ini adalah hasil karya saya sendiri kecuali cuplikan dan ringkasan yang masing-masing telah saya jelaskan sumbernya. Jika pada waktu selanjutnya ada pihak lain yang mengklaim bahwa skripsi ini sebagai karyanya, yang disertai dengan bukti-bukti yang cukup, maka saya bersedia untuk mempertanggungjawabkannya serta menerima seluruh konsekuensi dari pelanggaran tersebut.

Bandung, 07 Februari 2025

Yuda Saputra

ABSTRAK

Kemahasiswaan STMIK “AMIKBANDUNG” menghadapi tantangan dalam pengelolaan pengajuan angka kredit aktivitas mahasiswa, pengajuan proposal kegiatan, dan *tracer* alumni yang masih dilakukan secara manual dan terpisah. Hal ini menyebabkan inefisiensi dan kesulitan dalam pengelolaan data, serta proses verifikasi yang lambat. Pengembangan sistem ini menggunakan *Vue.js* untuk antarmuka pengguna, *Express.js* untuk server, dan MySQL sebagai basis data. Diharapkan, sistem ini dapat mengurangi beban kerja mahasiswa, dosen wali, dan pihak kemahasiswaan, serta meningkatkan partisipasi alumni dalam pengisian kuesioner *tracer*. Dengan demikian, sistem ini menciptakan platform yang lebih efisien dan efektif bagi semua pihak yang terlibat. Hasil penelitian ini berupa sebuah sistem informasi terintegrasi yang telah diuji menggunakan metode *black-box testing*. Sistem ini diharapkan dapat mempercepat proses verifikasi pengajuan aktivitas dan proposal, mengurangi beban kerja dosen wali dan bagian kemahasiswaan, serta meningkatkan partisipasi alumni dalam *tracer study*. Secara keseluruhan, sistem ini bertujuan untuk meningkatkan efisiensi dan efektivitas pengelolaan data aktivitas mahasiswa, proposal kegiatan, dan *tracer* alumni di STMIK “AMIKBANDUNG”.

Kata Kunci: Pengelolaan Pengajuan, Kemahasiswaan, *Website*

ABSTRACT

STMIK “AMIKBANDUNG” Student Affairs faces challenges in managing the submission of student activity credit scores, submission of activity proposals, and alumni tracers which are still done manually and separately. This causes inefficiency and difficulty in data management, as well as a slow verification process . The development of this system uses Vue.js for the user interface, Express.js for the server, and MySQL as a database. Hopefully, this system can reduce the workload of students, guardian lecturers, and student affairs, and increase alumni participation in filling out tracer questionnaires. Thus, this system creates a more efficient and effective platform for all parties involved. The result of this research is an integrated information system that has been tested using the black-box testing method. This system is expected to accelerate the verification process of activity and proposal submission, reduce the workload of the guardian lecturer and student affairs department, and increase alumni participation in the tracer study. Overall, this system aims to improve the efficiency and effectiveness of managing student activity data, activity proposals, and alumni tracer at STMIK “AMIKBANDUNG”.

Keywords: *Proposal Management, Student Affairs, Website.*

KATA PENGANTAR

Puji dan syukur penulis haturkan kehadiran Allah SWT, karena berkat rahmat dan hidayah-Nya-lah saya dapat menyelesaikan skripsi yang berjudul : “**Rancang Bangun Sistem Kemahasiswaan Berbasis Web di STMIK “AMIK BANDUNG”**”. Skripsi ini dibuat untuk memenuhi tugas akhir perkuliahan dan sebagai salah satu persyaratan untuk memperoleh gelar Sarjana Strata 1 di Program Studi Teknik Informatika STMIK “AMIK BANDUNG”. Selain itu, skripsi ini juga dibuat sebagai salah satu wujud implementasi dari ilmu yang didapatkan selama masa perkuliahan di Program Studi Teknik Informatika STMIK “AMIK BANDUNG”.

Penulis sadar bahwa laporan ini mungkin masih memiliki keterbatasan. Namun, penulis berharap laporan ini mampu memberikan gambaran yang komprehensif tentang usaha penulis dalam menciptakan sistem Kemahasiswaan Berbasis Web di STMIK “AMIK BANDUNG”.

Pada akhirnya, penulis ingin bertima kasih kepada semua yang telah memberikan dukungan. Oleh karena itu, tidak lupa penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak H. Asmui Mansur, M.Kom. selaku Ketua STMIK “AMIKBANDUNG” yang telah menyediakan sarana prasarana dan dukungan untuk memudahkan Penulis dalam proses pembelajaran dan penelitian.
2. Bapak Muhammad Abdul Mujib, M.T. selaku Ketua Program Studi Teknik Informatika STMIK “AMIKBANDUNG”. Sekaligus Dosen Pembimbing Terimakasih atas ilmu dan saran yang diberikan selama Penulis menjalani pendidikan di Program Studi Teknik Informatika.
3. Ibu Shafira Febriyani, M. T. selaku Dosen Pembimbing yang telah memberikan pengarahan selama penulisan tugas akhir ini.
4. Ibu Yolanda Georgia, S.Sn., M.Ds. selaku staff kemahasiswaan yang telah bersedia membantu penulis dalam memberikan informasi terkait kebutuhan kemahasiswaan dalam proses pembuatan sistem.
5. Seluruh Dosen, Staff dan Karyawan STMIK “AMIKBANDUNG” yang telah memberikan ilmu serta wawasan dan banyak membantu Penulis dalam menyelesaikan perkuliahan.
6. Rekan-rekan satu Program Studi S1 Teknik Informatika yang tidak bisa Penulis sebut satu persatu selalu memberikan bantuan dan motivasi dalam penyelesaian laporan ini.

DAFTAR ISI

LEMBAR PENGESAHAN	i
SURAT PERNYATAAN	ii
ABSTRAK.....	iii
<i>ABSTRACT.....</i>	iv
KATA PENGANTAR	v
DAFTAR ISI.....	vi
DAFTAR GAMBAR	ix
DAFTAR TABEL	xv
DAFTAR SIMBOL.....	xviii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Identifikasi Masalah.....	4
1.3 Rumusan Masalah.....	4
1.4 Tujuan	5
1.5 Batasan	5
1.6 Metode Penelitian	6
1.7 Waktu dan Tempat Penelitian.....	7
1.8 Sitematika Penulisan.....	8
BAB II TINJAUAN PUSTAKA.....	10
2.1 Konsep Sistem	10
2.1.1. Karakteristik Sistem.....	10
2.1.2 Klasifikasi Sistem.....	11
2.2 Pengembangan Sistem	12
2.3 Kredit Aktivitas Mahasiswa (AKAM)	14
2.4 Metode <i>System Development Life Cycle</i> (SDLC).....	15
2.5 MySQL (<i>My Structure Query Language</i>)	17

2.6	<i>Visual Studio Code</i>	17
2.6.1	<i>Vue.Js</i>	18
2.6.2	<i>Express.JS</i>	18
2.7	<i>Unified Modeling Language (UML)</i>	18
2.7.1	Diagram <i>Use Case (Use Case Diagram)</i>	19
2.7.2	Diagram Aktivitas (<i>Activity Diagram</i>).....	19
2.7.3	Diagram Kelas (<i>Class Diagram</i>)	19
2.7.4	Diagram Sekuen (<i>Sequence Diagram</i>).....	20
2.8	<i>Entity Relationship Diagram (ERD)</i>	20
2.9	Pengujian <i>Black box</i>	21
2.10	Unit Kegiatan Mahasiswa (UKM)	21
2.11	Pengertian Alumni	22
2.12	<i>Tracker</i> Alumni/Pelacakan Alumni.....	22
	BAB III ANALISIS DAN PERANCANGAN	23
3.1	Analisis Sistem Berjalan	23
3.2	Analisis Sistem yang Diusulkan	24
3.3	Analisis Kebutuhan Sistem yang Diusulkan.....	26
3.3.1.	Definisi Pengguna Sistem	26
3.3.2.	Kebutuhan Fungsional	27
3.3.3.	Kebutuhan Non Fungsional	28
3.3.4.	<i>Use Case Diagram</i>	29
3.3.5.	Skenario <i>Use Case</i>	33
3.3.6.	<i>Activity Diagram</i>	67
3.3.7.	<i>Squence Diagram</i>	99
3.3.8.	<i>Class Diagram</i>	118
3.3.9.	ERD (<i>Entity Relationship Diagram</i>).....	119
	BAB IV IMPLEMENTASI DAN PENGUJIAN	126

4.1	Implementasi Sistem Informasi Kemahasiswaan	126
4.1.1	<i>Flowchart</i> Implementasi Fitur	126
4.1.2	Arsitektur Sistem	129
4.1.3	Perangkat Yang Digunakan.....	129
4.1.4	Implementasi Antarmuka	130
4.2	Implementasi Status Pengajuan	140
4.3	Hasil Pengujian Sistem	141
	BAB V KESIMPULAN DAN PENUTUP	149
5.1	Kesimpulan	149
5.2	Saran	149
	DAFTAR PUSTAKA.....	150
	LAMPIRAN.....	153

DAFTAR GAMBAR

Gambar: 2. 1 Proses munculnya sistem baru	13
Gambar: 2. 2 Metode System Development Life Cycle (SDLC)	15
Gambar 3. 1 Skema Sistem Pengajuan Kredit Aktivitas yang berjalan	24
Gambar 3. 2 Skema Sistem yang Diusulkan.....	26
Gambar 3. 3 <i>Use Case Diagram</i> Mahasiswa	29
Gambar 3. 4 <i>Use Case Diagram</i> Dosen.....	30
Gambar 3. 5 <i>Use Case Diagram</i> Kemahasiswaan.....	31
Gambar 3. 6 <i>Use Case Diagram</i> Admin	32
Gambar 3. 7 <i>Activity Diagram Login</i> untuk aktor mahasiswa.....	67
Gambar 3. 8 <i>Activity Diagram Logout</i> untuk aktor mahasiswa.....	67
Gambar 3. 9 <i>Activity Diagram</i> Lihat Daftar Pengajuan Kredit Aktivitas untuk aktor mahasiswa	68
Gambar 3. 10 <i>Activity Diagram</i> Buat Pengajuan Kredit Aktivitas untuk aktor mahasiswa	68
Gambar 3. 11 <i>Activity Diagram</i> Membatalkan Pengajuan Kredit Aktivitas untuk aktor mahasiswa	69
Gambar 3. 12 <i>Activity Diagram</i> Lihat Daftar Pengajuan Proposal untuk aktor mahasiswa	69
Gambar 3. 13 <i>Activity Diagram</i> Buat Pengajuan Proposal untuk aktor mahasiswa....	70
Gambar 3. 14 <i>Activity Diagram</i> Membatalkan Pengajuan Proposal untuk aktor mahasiswa	70
Gambar 3. 15 <i>Activity Diagram</i> Lihat Daftar <i>Tracer Alumni</i> untuk aktor mahasiswa	71
Gambar 3. 16 <i>Activity Diagram</i> Kirim Bukti Pengisian <i>Tracer Alumni</i> untuk aktor mahasiswa	71
Gambar 3. 17 <i>Activity Diagram</i> Mengubah Bukti Pengisian <i>Tracer Alumni</i> untuk aktor mahasiswa	72
Gambar 3. 18 <i>Activity Diagram</i> Membatalkan Bukti Pengisian <i>Tracer Alumni</i> untuk aktor mahasiswa	72
Gambar 3. 19 <i>Activity Diagram Login</i> untuk aktor dosen	73
Gambar 3. 20 <i>Activity Diagram Logout</i> untuk aktor dosen	73
Gambar 3. 21 <i>Activity Diagram</i> Lihat Detail Bukti Aktivitas Mahasiswa untuk aktor dosen	74

Gambar 3. 22 <i>Activity Diagram</i> Edit Status Verifikasi Pengajuan Proposal untuk aktor dosen	74
Gambar 3. 23 <i>Activity Diagram Login</i> untuk aktor kemahasiswaan	75
Gambar 3. 24 <i>Activity Diagram Logout</i> untuk aktor kemahasiswaan	75
Gambar 3. 25 <i>Activity Diagram</i> Lihat Daftar Kategori Aktivitas untuk aktor kemahasiswaan	76
Gambar 3. 26 <i>Activity Diagram</i> Tambah Data Kategori Aktivitas untuk aktor kemahasiswaan	76
Gambar 3. 27 <i>Activity Diagram</i> Hapus Data Kategori Aktivitas untuk aktor kemahasiswaan	77
Gambar 3. 28 <i>Activity Diagram</i> Edit Data Kategori Aktivitas untuk aktor kemahasiswaan	77
Gambar 3. 29 <i>Activity Diagram</i> Lihat Daftar Pengajuan Aktivitas Mahasiswa untuk aktor kemahasiswaan	78
Gambar 3. 30 <i>Activity Diagram</i> Lihat Detail Bukti Aktivitas Mahasiswa untuk aktor kemahasiswaan	78
Gambar 3. 31 <i>Activity Diagram</i> Edit Status Verifikasi Pengajuan Aktivitas untuk aktor kemahasiswaan	79
Gambar 3. 32 <i>Activity Diagram</i> Lihat Daftar Pengajuan Proposal untuk aktor kemahasiswaan	79
Gambar 3. 33 <i>Activity Diagram</i> Lihat Detail Bukti pengajuan Proposal untuk aktor kemahasiswaan	80
Gambar 3. 34 <i>Activity Diagram</i> Edit Status Verifikasi Pengajuan Proposal untuk aktor kemahasiswaan	80
Gambar 3. 35 <i>Activity Diagram</i> Lihat Daftar <i>Tracer Alumni</i> untuk aktor kemahasiswaan	81
Gambar 3. 36 <i>Activity Diagram</i> Tambah Data <i>Tracer Alumni</i> untuk aktor kemahasiswaan	81
Gambar 3. 37 <i>Activity Diagram</i> Edit Data <i>Tracer Alumni</i> untuk aktor kemahasiswaan	82
Gambar 3. 38 <i>Activity Diagram</i> Hapus Data <i>Tracer Alumni</i> untuk aktor kemahasiswaan	82
Gambar 3. 39 <i>Activity Diagram Login</i> untuk aktor admin	83
Gambar 3. 40 <i>Activity Diagram Logout</i> untuk aktor admin	84

Gambar 3. 41 <i>Activity Diagram</i> Lihat Daftar Program Studi untuk aktor admin.....	84
Gambar 3. 42 <i>Activity Diagram</i> Tambah Data Daftar Program Studi untuk aktor admin	85
Gambar 3. 43 <i>Activity Diagram</i> Edit Data Program Studi untuk aktor admi	85
Gambar 3. 44 <i>Activity Diagram</i> Hapus Data Program Studi untuk aktor admin.....	86
Gambar 3. 45 <i>Activity Diagram</i> Lihat Daftar Tahun Akademik untuk aktor admin....	86
Gambar 3. 46 <i>Activity Diagram</i> Tambah Data Tahun Akademik untuk aktor admin..	87
Gambar 3. 47 <i>Activity Diagram</i> Edit Data Tahun Akademik untuk aktor admin	88
Gambar 3. 48 <i>Activity Diagram</i> Hapus Data Tahun Akademik untuk aktor admin.....	89
Gambar 3. 49 <i>Activity Diagram</i> Lihat Daftar Unit Kegiatan Mahasiswa untuk aktor admin.....	89
Gambar 3. 50 <i>Activity Diagram</i> Tambah Data Unit Kegiatan Mahasiswa untuk aktor admin.....	90
Gambar 3. 51 <i>Activity Diagram</i> Edit Data Unit Kegiatan Mahasiswa untuk aktor admin	91
Gambar 3. 52 <i>Activity Diagram</i> Hapus Data Unit Kegiatan Mahasiswa untuk aktor admin.....	92
Gambar 3. 53 47. <i>Activity Diagram</i> Lihat Daftar Dosen untuk aktor admin	92
Gambar 3. 54 <i>Activity Diagram</i> Tambah Data Dosen untuk aktor admin	93
Gambar 3. 55 <i>Activity Diagram</i> Edit Data Dosen untuk aktor admin	94
Gambar 3. 56 <i>Activity Diagram</i> Hapus Data Dosen untuk aktor admin.....	95
Gambar 3. 57 <i>Activity Diagram</i> Lihat Daftar Mahasiswa untuk aktor admin	95
Gambar 3. 58 5 <i>Activity Diagram</i> Tambah Data Mahasiswa untuk aktor admin	96
Gambar 3. 59 <i>Activity Diagram</i> Edit Data Mahasiswa untuk aktor admin	97
Gambar 3. 60 <i>Activity Diagram</i> Hapus Data Mahasiswa untuk aktor admin.....	98
Gambar 3. 61 <i>Squence Diagram</i> Login untuk aktor mahasiswa.....	99
Gambar 3. 62 <i>Squence Diagram</i> Logout untuk aktor mahasiswa.....	99
Gambar 3. 63 <i>Squence Diagram</i> Lihat Daftar Pengajuan Kredit Aktivitas untuk aktor mahasiswa.....	99
Gambar 3. 64 <i>Squence Diagram</i> Buat Pengajuan Kredit Aktivitas untuk aktor mahasiswa.....	100
Gambar 3. 65 <i>Squence Diagram</i> Membatalkan Pengajuan Kredit Aktivitas untuk aktor mahasiswa.....	100

Gambar 3. 66 <i>Squence Diagram</i> Lihat Daftar Pengajuan Proposal untuk aktor mahasiswa.....	100
Gambar 3. 67 <i>Squence Diagram</i> Buat Pengajuan Proposal untuk aktor mahasiswa.	101
Gambar 3. 68 <i>Squence Diagram</i> Membatalkan Pengajuan Proposal untuk aktor mahasiswa.....	101
Gambar 3. 69 <i>Squence Diagram</i> Lihat Daftar <i>Tracer Alumni</i> untuk aktor mahasiswa	101
Gambar 3. 70 <i>Squence Diagram</i> Kirim Bukti Pengisian <i>Tracer Alumni</i> untuk aktor mahasiswa.....	102
Gambar 3. 71 <i>Squence Diagram</i> Mengubah Bukti Pengisian <i>Tracer Alumni</i> untuk aktor mahasiswa	102
Gambar 3. 72 <i>Squence Diagram</i> Membatalkan Bukti Pengisian <i>Tracer Alumni</i> aktor mahasiswa.....	103
Gambar 3. 73 <i>Squence Diagram Login</i> untuk aktor dosen.....	103
Gambar 3. 74 <i>Squence Diagram Logout</i> untuk aktor dosen.....	103
Gambar 3. 75 <i>Squence Diagram</i> Lihat Detail Bukti Aktivitas Mahasiswa untuk aktor dosen	104
Gambar 3. 76 <i>Squence Diagram</i> Edit Status Verifikasi Pengajuan Proposal untuk aktor dosen	104
Gambar 3. 77 <i>Squence Diagram Login</i> untuk aktor kemahasiswaan	104
Gambar 3. 78 <i>Squence Diagram Logout</i> untuk aktor kemahasiswaan	105
Gambar 3. 79 <i>Squence Diagram</i> Lihat Daftar Kategori Aktivitas untuk aktor kemahasiswaan	105
Gambar 3. 80 <i>Squence Diagram</i> Tambah Data Kategori Aktivitas untuk aktor kemahasiswaan	105
Gambar 3. 81 <i>Squence Diagram</i> Hapus Data Kategori Aktivitas untuk aktor kemahasiswaan	106
Gambar 3. 82 <i>Squence Diagram</i> Edit Data Kategori Aktivitas untuk aktor kemahasiswaan	106
Gambar 3. 83 <i>Squence Diagram</i> Lihat Daftar Pengajuan Aktivitas Mahasiswa untuk aktor kemahasiswaan	106
Gambar 3. 84 <i>Squence Diagram</i> Lihat Detail Bukti Aktivitas Mahasiswa untuk aktor kemahasiswaan	107

Gambar 3. 85 <i>Squence Diagram</i> Edit Status Verifikasi Pengajuan Aktivitas untuk aktor kemahasiswaan	107
Gambar 3. 86 <i>Squence Diagram</i> Lihat Daftar Pengajuan Proposal untuk aktor kemahasiswaan	107
Gambar 3. 87 <i>Squence Diagram</i> Lihat Detail Bukti pengajuan Proposal untuk aktor kemahasiswaan	108
Gambar 3. 88 <i>Squence Diagram</i> Edit Status Verifikasi Pengajuan Proposal untuk aktor kemahasiswaan	108
Gambar 3. 89 <i>Squence Diagram</i> Lihat Daftar <i>Tracer Alumni</i> untuk aktor kemahasiswaan	109
Gambar 3. 90 <i>Squence Diagram</i> Tambah Data <i>Tracer Alumni</i> untuk aktor kemahasiswaan	109
Gambar 3. 91 <i>Squence Diagram</i> Edit Data <i>Tracer Alumni</i> untuk aktor kemahasiswaan	109
Gambar 3. 92 <i>Squence Diagram</i> Hapus Data <i>Tracer Alumni</i> untuk aktor kemahasiswaan	110
Gambar 3. 93 <i>Squence Diagram Login</i> untuk aktor admin	110
Gambar 3. 94 <i>Squence Diagram Logout</i> aktor admin	111
Gambar 3. 95 <i>Squence Diagram</i> Lihat Daftar Program Studi untuk aktor admin....	111
Gambar 3. 96 <i>Squence Diagram</i> Tambah Data Daftar Program Studi untuk aktor admin	111
Gambar 3. 97 <i>Squence Diagram</i> Edit Data Program Studi untuk aktor admin	112
Gambar 3. 98 <i>Squence Diagram</i> Hapus Data Program Studi untuk aktor admin.....	112
Gambar 3. 99 <i>Squence Diagram</i> Lihat Daftar Tahun Akademik untuk aktor admin.	112
Gambar 3. 100 <i>Squence Diagram</i> Tambah Data Tahun Akademik untuk aktor admin	113
Gambar 3. 101 <i>Squence Diagram</i> Edit Data Tahun Akademik untuk aktor admin ...	113
Gambar 3. 102 <i>Squence Diagram</i> Hapus Data Tahun Akademik untuk aktor admin	113
Gambar 3. 103 <i>Squence Diagram</i> Lihat Daftar Unit Kegiatan Mahasiswa aktor admin	114
Gambar 3. 104 <i>Squence Diagram</i> Tambah Data Unit Kegiatan Mahasiswa untuk aktor admin.....	114
Gambar 3. 105 <i>Squence Diagram</i> Edit Data Unit Kegiatan Mahasiswa untuk aktor admin.....	114

Gambar 3. 106 <i>Squence Diagram</i> Hapus Data Unit Kegiatan Mahasiswa untuk aktor admin.....	115
Gambar 3. 107 <i>Squence Diagram</i> Lihat Daftar Dosen untuk aktor admin.....	115
Gambar 3. 108 <i>Squence Diagram</i> Tambah Data Dosen untuk aktor admin	115
Gambar 3. 109 <i>Squence Diagram</i> Edit Data Dosen untuk aktor admin	116
Gambar 3. 110 <i>Squence Diagram</i> Hapus Data Dosen untuk aktor admin.....	116
Gambar 3. 111 <i>Squence Diagram</i> Lihat Daftar Mahasiswa untuk aktor admin	116
Gambar 3. 112 <i>Squence Diagram</i> Tambah Data Mahasiswa untuk aktor admin	117
Gambar 3. 113 <i>Squence Diagram</i> Edit Data Mahasiswa untuk aktor admin	117
Gambar 3. 114 <i>Squence Diagram</i> Hapus Data Mahasiswa untuk aktor admin.....	118
Gambar 3. 115 <i>Class Diagram</i> Sistem.....	118
Gambar 3. 116 <i>ERD Entity Relationship Diagram</i> Sistem	119
Gambar 4. 1 <i>Flowchart</i> Kredit Aktivitas Mahasiswa.....	126
Gambar 4. 2 <i>Flowchart</i> Pengajuan Proposal	127
Gambar 4. 3 <i>Flowchart Tracer</i> Almni	128
Gambar 4. 4 Arsitektur Sistem.....	129
Gambar 4. 5 Tampilan <i>Login</i> Mahasiswa	130
Gambar 4. 6 Tampilan Aktivitas Mahasiswa	131
Gambar 4. 7 Tampilan <i>Tracer</i> Alumni Mahasiswa	131
Gambar 4. 8 Tampilan Pengajuan Proposal UKM Mahasiswa.....	132
Gambar 4. 9 Tampilan <i>Login</i> Dosen	133
Gambar 4. 10 Tampilan Aktivitas Dosen	133
Gambar 4. 11 Tampilan <i>Login</i> Kemahasiswaan	134
Gambar 4. 12 Tampilan Aktivitas Kemahasiswaan	134
Gambar 4. 13 Tampilan Proposal UKM Kemahasiswaan	135
Gambar 4. 14 Tampilan Kemahasiswaan <i>Tracer</i> Alumni	136
Gambar 4. 15 Tampilan <i>Login</i> Admin	136
Gambar 4. 16 Tampilan Program Studi Admin.....	137
Gambar 4. 17 Tampilan Tahun Akademik Admin	137
Gambar 4. 18 Tampian Unit Kegiatan Mahasiswa Admin	138
Gambar 4. 19 Tampilan Dosen Admin.....	138
Gambar 4. 20 Tampilan Mahasiswa Admin	139
Gambar 4. 21 Tampilan Kemahasiswaan Admin.....	139
Gambar 4. 22 Informasi Status Pengajuan.....	140

DAFTAR TABEL

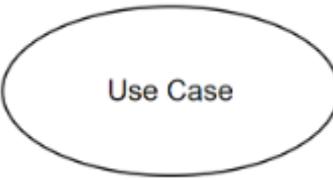
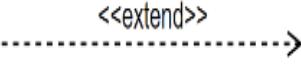
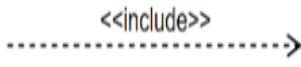
Tabel 1. 1 Jadwal Kegiatan Penelitian	7
Tabel 3. 1 Kategori Pengguna Sistem.....	26
Tabel 3. 2 Kebutuhan Fungsional Sistem	27
Tabel 3. 3 Kebutuhan Non Fungsional	28
Tabel 3. 4 Skenario <i>Use Case Login</i> mahasiswa	33
Tabel 3. 5 Skenario <i>Use Case Logout</i> Mahasiswa.....	34
Tabel 3. 6 Skenario <i>Use Case Lihat Daftar Pengajuan Kredit Aktivitas</i> untuk aktor mahasiswa.....	34
Tabel 3. 7 Skenario <i>Use Case Buat Pengajuan Kredit Aktivitas</i> untuk aktor mahasiswa	35
Tabel 3. 8 Skenario <i>Use Case Membatalkan Pengajuan Kredit Aktivitas</i> untuk aktor mahasiswa.....	36
Tabel 3. 9 Skenario <i>Use Case Lihat Daftar Pengajuan Proposal</i> untuk aktor mahasiswa	37
Tabel 3. 10 Skenario <i>Use Case Buat Pengajuan Proposal</i> untuk aktor mahasiswa	37
Tabel 3. 11 Skenario <i>Use Case Membatalkan Pengajuan Proposal</i> untuk aktor mahasiswa.....	38
Tabel 3. 12 Skenario <i>Use Case Lihat Daftar Tracer Alumni</i> untuk aktor mahasiswa .	39
Tabel 3. 13 Skenario <i>Use Case Kirim Bukti Pengisian Tracer Alumni</i> untuk aktor mahasiswa.....	39
Tabel 3. 14 Skenario <i>Use Case Mengubah Bukti Pengisian Tracer Alumni</i> untuk aktor mahasiswa.....	40
Tabel 3. 15 Skenario <i>Use Case Membatalkan Bukti Pengisian Tracer Alumni</i> untuk aktor mahasiswa	41
Tabel 3. 16 Skenario <i>Use Case Login</i> untuk aktor dosen	41
Tabel 3. 17 Skenario <i>Use Case Logout</i> untuk aktor dosen	42
Tabel 3. 18 Skenario <i>Use Case Lihat Detail Bukti Aktivitas Mahasiswa</i> uuntuk aktor dosen.....	42
Tabel 3. 19 Skenario <i>Use Case Edit Status Verifikasi Pengajuan Proposal</i> untuk aktor dosen.....	43
Tabel 3. 20 Skenario <i>Use Case Login</i> untuk aktor kemahasiswaan	44
Tabel 3. 21 Skenario <i>Use Case Logout</i> untuk aktor kemahasiswaan	44

Tabel 3. 22 Skenario <i>Use Case</i> Lihat Daftar Kategori Aktivitas untuk aktor kemahasiswaan	45
Tabel 3. 23 Skenario <i>Use Case</i> Tambah Data Kategori Aktivitas untuk aktor kemahasiswaan	45
Tabel 3. 24 Skenario <i>Use Case</i> Hapus Data Kategori Aktivitas untuk aktor kemahasiswaan	46
Tabel 3. 25 Skenario <i>Use Case</i> Edit Data Kategori Aktivitas untuk aktor kemahasiswaan	46
Tabel 3. 26 Skenario <i>Use Case</i> Lihat Daftar Pengajuan Aktivitas Mahasiswa untuk aktor kemahasiswaan	47
Tabel 3. 27 Skenario <i>Use Case</i> Lihat Detail Bukti Aktivitas Mahasiswa untuk aktor kemahasiswaan	47
Tabel 3. 28 Skenario <i>Use Case</i> Edit Status Verifikasi Pengajuan Aktivitas untuk aktor kemahasiswaan	48
Tabel 3. 29 Skenario <i>Use Case</i> Lihat Daftar Pengajuan Proposal untuk aktor kemahasiswaan	48
Tabel 3. 30 Skenario <i>Use Case</i> Lihat Detail Bukti pengajuan Proposal untuk aktor kemahasiswaan	49
Tabel 3. 31 Skenario <i>Use Case</i> Edit Status Verifikasi Pengajuan Proposal untuk aktor kemahasiswaan	49
Tabel 3. 32 Skenario <i>Use Case</i> Lihat Daftar <i>Tracer</i> Alumni untuk aktor kemahasiswaan	50
Tabel 3. 33 Skenario <i>Use Case</i> Tambah Data <i>Tracer</i> Alumni untuk aktor kemahasiswaan	51
Tabel 3. 34 Skenario <i>Use Case</i> Edit Data <i>Tracer</i> Alumni untuk aktor kemahasiswaan	51
Tabel 3. 35 Skenario <i>Use Case</i> Hapus Data <i>Tracer</i> Alumni untuk aktor kemahasiswaan	52
Tabel 3. 36 Skenario <i>Use Case Login</i> untuk aktor admin.....	52
Tabel 3. 37 Skenario <i>Use Case Logout</i> untuk aktor admin.....	53
Tabel 3. 38 Skenario <i>Use Case</i> Lihat Daftar Program Studi untuk aktor admin	53
Tabel 3. 39 Skenario <i>Use Case</i> Tambah Data Daftar Program Studi untuk aktor admin	54
Tabel 3. 40 Skenario <i>Use Case</i> Edit Data Program Studi untuk aktor admin	55

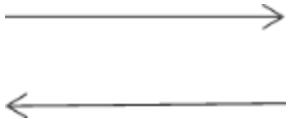
Tabel 3. 41 Skenario <i>Use Case</i> Hapus Data Program Studi untuk aktor admin	56
Tabel 3. 42 Skenario <i>Use Case</i> Lihat Daftar Tahun Akademik untuk aktor admin	56
Tabel 3. 43 Skenario <i>Use Case</i> Tambah Data Tahun Akademik untuk aktor admin ...	57
Tabel 3. 44 Skenario <i>Use Case</i> Edit Data Tahun Akademik untuk aktor admin	58
Tabel 3. 45 Skenario <i>Use Case</i> Hapus Data Tahun Akademik untuk aktor admin.....	59
Tabel 3. 46 Skenario <i>Use Case</i> Lihat Daftar Unit Kegiatan Mahasiswa untuk aktor admin	59
Tabel 3. 47 Skenario <i>Use Case</i> Tambah Data Unit Kegiatan Mahasiswa untuk aktor admin	60
Tabel 3. 48 Skenario <i>Use Case</i> Edit Data Unit Kegiatan Mahasiswa aktor admin	60
Tabel 3. 49 Skenario <i>Use Case</i> Hapus Data Unit Kegiatan Mahasiswa untuk aktor admin	61
Tabel 3. 50 Skenario <i>Use Case</i> Lihat Daftar Dosen untuk aktor admin	62
Tabel 3. 51 Skenario <i>Use Case</i> Tambah Data Dosen untuk aktor admin	62
Tabel 3. 52 Skenario <i>Use Case</i> Edit Data Dosen untuk aktor admin	63
Tabel 3. 53 Skenario <i>Use Case</i> Hapus Data Dosen untuk aktor admin	63
Tabel 3. 54 Skenario <i>Use Case</i> Lihat Daftar Mahasiswa untuk aktor admin	64
Tabel 3. 55 Skenario <i>Use Case</i> Tambah Data Mahasiswa untuk aktor admin.....	64
Tabel 3. 56 ekanrio <i>Use Case</i> Edit Data Mahasiswa untuk aktor admin	65
Tabel 3. 57 Skenario <i>Use Case</i> Hapus Data Mahasiswa untuk aktor admin	66
Tabel 3. 58 Struktur Tabel <i>Student</i>	119
Tabel 3. 59 Struktur Tabel <i>QuestionnaireSubmission</i>	120
Tabel 3. 60 Struktur Tabel <i>QuestionnaireAnswer</i>	120
Tabel 3. 61 Struktur Tabel <i>Lecturer</i>	121
Tabel 3. 62 Struktur Tabel <i>ProgramStudy</i>	121
Tabel 3. 63 Struktur Tabel <i>StudentActivity</i>	122
Tabel 3. 64 Struktur Tabel <i>ActivityCategory</i>	122
Tabel 3. 65 Struktur Tabel <i>StudentAffairs</i>	123
Tabel 3. 66 Struktur Tabel Admin	123
Tabel 3. 67 Struktur Tabel <i>AcademicYear</i>	124
Tabel 3. 68 Struktur Tabel UKM	124
Tabel 3. 69 Struktur Tabel <i>Proposal</i>	124
Tabel 3. 70 Struktur Tabel <i>ProposalFile</i>	125
Tabel 3. 71 <i>Blackbox Testing</i>	141

DAFTAR SIMBOL

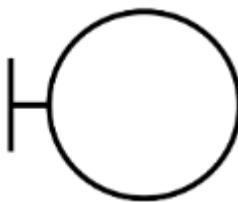
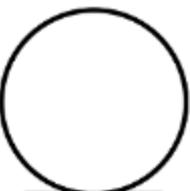
A. Use Case Diagram

No	Nama	Simbol	Deskripsi
1	<i>Use Case</i>		Fungsionalitas yang disediakan sistem sebagai unit yang saling bertukar pesan antar unit atau aktor; sering dinyatakan dengan menggunakan kata kerja, misal Input data.
2	Aktor		Orang, proses, atau sistem lain yang berinteraksi dengan sistem yang akan dikembangkan di luar sistem tersebut, sehingga meskipun simbol aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; sering dinyatakan menggunakan kata benda, misal admin, mahasiswa, dll.
3	Asosiasi		Menunjukkan komunikasi atau hubungan antara aktor dan <i>use case</i> atau antar <i>use case</i>
4	<i>Extend</i>		Hubungan <i>use case</i> tambahan ke <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; anak panah menuju <i>use case</i> yang dituju.
5	<i>Include</i>		Hubungan antara <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya, sehingga <i>use case</i> harus dijalankan terlebih dulu.

B. Activity Diagram

No	Nama	Simbol	Keterangan
1	<i>Swimlane</i>		Menunjukkan siapa yang bertanggung jawab dalam melakukan aktivitas dalam suatu diagram
2	<i>Initial State</i> (Titik Awal)		Aliran kerja dimulai
3	<i>Activity Final Node</i> (Titik Akhir)		Aliran kerja diakhiri
4	<i>Activity</i>		Langkah-langkah dalam sebuah <i>activity</i> . <i>Action</i> ini mencerminkan eksekusi dari suatu aksi.
5	<i>Decision Node</i>		Menunjukkan suatu keputusan yang mempunyai lebih dari satu aktivitas sesuai dari kondisi yang dituju.
6	<i>Control Flow</i>		<i>Line Connection</i> , Digunakan untuk menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu.

C. Squence Diagram

No	Nama	Gambar	Keterangan
1	<i>Lifeline</i>		Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2	<i>Boundary</i>		Menggambarkan interaksi antara satu atau lebih aktor dengan sistem. Memodelkan bagian dari sistem yang bergantung pada pihak lain disekitarnya dan merupakan pembatas sistem dengan lingkungan luar.
3	<i>Control</i>		Menggambarkan “perilaku mengatur”. Mengkoordinasikan perilaku sistem dan dinamika dari suatu sistem. Menangani tugas utama dan mengontrol alur kerja sistem.
4	<i>Entity</i>		Menggambarkan informasi yang harus disimpan oleh sistem (struktur data dari sebuah sistem)

5	<i>Message</i>		Spesifikasi dari komunikasi antarobjek yang memuat informasi-informasi tentang aktifitas yang terjadi
6	<i>Return Message</i>		Menggambarkan pesan/hubungan antar objek yang menunjukkan urutan kejadian yang terjadi.

D. Class Diagram

No	Nama	Simbol	Deskripsi					
1	Kelas	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>nama kelas</td></tr> <tr><td>+Attribute1</td></tr> <tr><td>+Attribute2</td></tr> <tr><td>+Operation1()</td></tr> <tr><td>+Operation2()</td></tr> </table>	nama kelas	+Attribute1	+Attribute2	+Operation1()	+Operation2()	Menggambarkan sebuah kelas yang terdiri dari <i>attribute</i> dan <i>method</i>
nama kelas								
+Attribute1								
+Attribute2								
+Operation1()								
+Operation2()								
2	Asosiasi		Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>					

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perguruan tinggi merupakan lembaga pendidikan yang memiliki tanggung jawab besar untuk mencetak Sumber Daya Manusia (SDM) yang kompeten, profesional, bermoral, dan mampu bersaing secara global. Untuk mewujudkan tujuan tersebut, perguruan tinggi perlu memastikan lulusannya memiliki kemampuan yang menyeluruh, baik dari sisi *hard skill* maupun *soft skill* [1]. Dalam mendukung hal ini, mahasiswa didorong untuk mengikuti dan mengadakan berbagai kegiatan yang bertujuan meningkatkan keterampilan tersebut. Sebagai bagian dari upaya ini, setiap aktivitas yang melibatkan mahasiswa akan dihargai melalui sistem penilaian Angka Kredit Aktivitas Mahasiswa (AKAM) [2], yang mencatat kontribusi mereka dalam pengembangan *hard skill* dan *soft skill*. Dengan demikian, mahasiswa dapat meningkatkan kemampuan mereka, baik melalui perkuliahan di kelas maupun kegiatan pendukung di luar kelas.

Setiap mahasiswa diwajibkan untuk melaporkan kegiatan yang telah diikuti atau prestasi yang berhasil diraih melalui Sistem Penilaian Angka Kredit Aktivitas Mahasiswa (AKAM). Laporan ini mencakup berbagai bidang, seperti penalaran dan keilmuan, organisasi dan kepemimpinan, pengabdian masyarakat, minat dan bakat, serta bidang lainnya. Mahasiswa juga perlu melampirkan bukti fisik yang sah, seperti sertifikat, surat keputusan, surat tugas, atau dokumen pendukung lain yang sesuai dengan aturan yang berlaku [3]. Proses validasi atas kegiatan mahasiswa dilakukan oleh Bagian Kemahasiswaan sebagai pihak yang berwenang sesuai dengan bidang kegiatannya.

Sekolah Tinggi Manajemen dan Informatika Komputer (STMIK) "AMIKBANDUNG" berupaya mendorong mahasiswanya untuk aktif mengikuti berbagai kegiatan kemahasiswaan sebagai sarana menggali pengalaman dan mengembangkan kemampuan. Melalui kegiatan ini, mahasiswa diharapkan dapat memiliki nilai tambah dan keterampilan yang lebih seimbang antara *hard skill* dan *soft skill* [4]. Sebagai bagian dari upaya tersebut, STMIK "AMIKBANDUNG" menetapkan sistem Aktivitas Kredit Mahasiswa untuk menilai dan mendokumentasikan aktivitas mahasiswa. Kebijakan ini diatur secara resmi dalam Surat Keputusan Nomor

024/SAB/Z.2/II/2020 tentang Peraturan Kredit Aktivitas Mahasiswa di STMIK "AMIKBANDUNG".

Aktivitas Kredit Mahasiswa adalah sistem yang dirancang untuk memberikan pengakuan atas prestasi mahasiswa dalam mengembangkan *soft skills* melalui kegiatan kemahasiswaan. Mahasiswa diwajibkan melaporkan aktivitas yang telah diikuti ke Bagian Kemahasiswaan untuk mendapatkan nilai kredit sesuai dengan jenis kegiatannya. Kredit yang terkumpul selama masa perkuliahan akan dicantumkan dalam Surat Keterangan Pendamping Ijazah (SKPI) yang diberikan kepada lulusan saat wisuda. SKPI memiliki peran penting sebagai dokumen pendukung syarat kelulusan. Namun, banyak mahasiswa STMIK "AMIKBANDUNG" yang masih belum mencapai target kredit aktivitas, terutama ketika sudah memasuki tahun akhir. Salah satu kendala utama adalah belum adanya sistem yang mampu mengelola dan memantau data kredit aktivitas mahasiswa secara terorganisir.

Berbagai penelitian sebelumnya menunjukkan bahwa pengelolaan data kredit aktivitas mahasiswa menggunakan sistem manual dapat menyebabkan berbagai permasalahan, antara lain ketidakakuratan data, keterlambatan dalam penghitungan kredit, serta ketidakteraturan dalam pelaporan kegiatan. Penelitian oleh Achmad Syarif dalam "Sistem Informasi Penilaian Prestasi Mahasiswa Program Studi Sekretari Berbasis Web" mengembangkan aplikasi berbasis web" [5] yang memungkinkan mahasiswa untuk mengajukan, memantau, dan memperbarui status angka kredit kegiatan mereka secara *real-time*. Sistem ini mengurangi risiko kesalahan input data dan meningkatkan transparansi pengelolaan kredit mahasiswa. Selain itu, aplikasi ini juga memungkinkan dosen untuk memberikan persetujuan secara digital, sehingga mempermudah proses verifikasi kegiatan mahasiswa.

Penelitian serupa dilakukan oleh Afriansyah [6] dalam "Sistem Informasi Transkrip Aktivitas Kemahasiswaan (SITAKIS) Fakultas Ilmu Komputer Universitas Lancang Kuning ". Dalam penelitian ini, sistem yang dikembangkan berbasis aplikasi *mobile*, memungkinkan mahasiswa untuk mengakses dan melaporkan kegiatan mereka kapan saja dan di mana saja. Aplikasi ini mempermudah dosen dalam memberikan persetujuan dan memantau kegiatan mahasiswa, yang pada akhirnya dapat meningkatkan efisiensi dan fleksibilitas dalam pengelolaan kredit aktivitas.

Sebuah studi lain yang dilakukan oleh Sur Yanti,dkk [7] berjudul "Pengukuran Capability Level Sistem Penilaian Prestasi Mahasiswa Stmik Akakom Menggunakan Framework Cobit 5", mengembangkan sistem berbasis *cloud* untuk mempermudah

penyimpanan dan pengelolaan data kegiatan mahasiswa. Dengan menggunakan sistem berbasis *cloud*, aplikasi ini memudahkan administrasi untuk mengakses dan mengelola data secara *real-time*, serta meminimalisir kehilangan data penting. Sistem ini juga menyediakan laporan analitik terkait keterlibatan mahasiswa dalam berbagai kegiatan, yang sangat berguna bagi pengambilan keputusan terkait kebijakan kemahasiswaan.

Berdasarkan penelitian-penelitian tersebut, pengembangan sistem informasi berbasis teknologi untuk pengelolaan angka kredit mahasiswa terbukti efektif dalam meningkatkan efisiensi, transparansi, dan akurasi data. Oleh karena itu, untuk mengatasi masalah yang ada di STMIK "AMIKBANDUNG", diperlukan pengembangan sistem informasi kemahasiswaan yang lebih efisien, terintegrasi, dan dapat diakses secara *real-time*. Sistem ini diharapkan dapat mempermudah pengelolaan dan monitoring angka kredit mahasiswa serta memastikan bahwa mahasiswa dapat memenuhi target kredit aktivitas tepat waktu. Dengan demikian, sistem ini akan meningkatkan pengalaman administrasi bagi mahasiswa, dosen, dan pihak kemahasiswaan, serta berkontribusi pada kelancaran proses kelulusan mahasiswa.

Selain itu, pengembangan sistem informasi kemahasiswaan ini juga mencakup fitur pengajuan proposal untuk keperluan skripsi dan *tracer* alumni. Fitur pengajuan proposal skripsi dirancang untuk memberikan kemudahan bagi mahasiswa dalam mengajukan, memantau, dan mengelola status proposal mereka secara efisien. Melalui sistem ini, mahasiswa dapat mengakses informasi terkait prosedur pengajuan, termasuk panduan penulisan proposal, persyaratan yang harus dipenuhi, serta tenggat waktu yang harus diperhatikan. Dengan adanya fitur ini, diharapkan mahasiswa dapat lebih terorganisir dalam proses pengajuan, sehingga mengurangi risiko keterlambatan dan kebingungan yang sering terjadi dalam tahap awal penyusunan skripsi.

Selain itu, fitur *tracer* alumni akan memungkinkan institusi untuk melacak jejak karier lulusan secara sistematis. Melalui aplikasi ini, alumni dapat mengisi survei mengenai pengalaman kerja, posisi yang dijabat, serta relevansi pendidikan yang diterima di STMIK "AMIKBANDUNG" terhadap dunia kerja. Data yang terkumpul dari fitur *Tracer* alumni ini akan memberikan wawasan berharga bagi pihak perguruan tinggi dalam mengevaluasi dan meningkatkan kualitas kurikulum serta program studi yang ditawarkan. Informasi ini juga dapat digunakan untuk membangun jaringan alumni yang lebih kuat, serta memberikan dukungan bagi mahasiswa yang sedang mencari peluang kerja.

Dengan demikian, pengembangan sistem kemahasiswaan berbasis web di STMIK "AMIKBANDUNG" yang terintegrasi ini tidak hanya akan meningkatkan efisiensi dan transparansi dalam pengelolaan kredit aktivitas, tetapi juga akan memberikan kontribusi signifikan terhadap pengelolaan akademik secara keseluruhan di STMIK "AMIKBANDUNG". Sistem ini diharapkan dapat mempermudah mahasiswa dalam memenuhi target kredit aktivitas, serta mendukung proses akademik yang lebih terstruktur dan terencana, baik untuk mahasiswa aktif maupun alumni. Dengan fitur-fitur tambahan ini, aplikasi diharapkan dapat menjadi solusi komprehensif yang mendukung pengembangan kompetensi mahasiswa dan alumni, serta memperkuat posisi STMIK "AMIKBANDUNG" dalam mencetak sumber daya manusia yang berkualitas dan siap bersaing di tingkat global.

1.2 Identifikasi Masalah

Berdasarkan latar belakang yang dikemukakan, maka dapat diidentifikasi berbagai permasalahan sebagai berikut:

1. Kebutuhan akan sistem informasi yang dapat mengelola pengajuan kredit aktivitas mahasiswa, pengajuan proposal skripsi, dan pelacakan alumni secara efisien dan transparan. Saat ini, pengelolaan kredit aktivitas mahasiswa di STMIK "AMIKBANDUNG" masih dilakukan secara manual, yang mengakibatkan kesulitan dalam memantau dan mengelola berbagai jenis kegiatan mahasiswa dan juga menghambat proses pengajuan proposal mahasiswa.
2. Tidak adanya sistem otomatisasi yang mampu meminimalkan kesalahan dalam pengolahan data pengajuan kredit aktivitas mahasiswa, pengajuan proposal skripsi, dan pelacakan alumni. Proses manual yang saat ini diterapkan sering kali menyebabkan kesalahan input data, keterlambatan dalam verifikasi, dan ketidakakuratan dalam penilaian kredit aktivitas serta berakibat pada keterlambatan pengajuan proposal.

1.3 Rumusan Masalah

Berdasarkan latar belakang yang dikemukakan, maka dapat dirumuskan permasalahan sebagai berikut:

1. Bagaimana merancang sistem informasi yang dapat mengelola pengajuan angka kredit aktivitas mahasiswa, pengajuan proposal Unit Kegiatan Mahasiswa, dan *tracer* alumni secara efisien dan transparan?
2. Bagaimana sistem informasi dapat memberikan akses yang mudah kepada mahasiswa mengenai status pengajuan kredit aktivitas, proposal unit kegiatan mahasiswa, serta informasi *tracer* alumni?
3. Bagaimana hasil pengujian dari sistem informasi kemahasiswaan STMIK "AMIKBANDUNG"?

1.4 Tujuan

Tujuan perancangan pada penelitian ini diantaranya:

1. Mengembangkan sistem berupa aplikasi *website* kemahasiswaan yang terintegrasi untuk pengelolaan pengajuan angka kredit aktivitas mahasiswa, pengajuan proposal Unit Kegiatan Mahasiswa, dan pelacakan alumni, yang mampu menggantikan proses manual dengan sistem berbasis digital yang efisien dan efektif.
2. Memberikan kemudahan kepada mahasiswa dalam mengakses informasi terkait status pengajuan kredit aktivitas mereka, pengajuan proposal unit kegiatan mahasiswa, serta informasi *Tracer* alumni, sehingga mahasiswa dapat dengan mudah memantau perkembangan pengajuan mereka.
3. Melakukan pengujian menyeluruh terhadap sistem berupa aplikasi *website* kemahasiswaan STMIK "AMIKBANDUNG" untuk memastikan bahwa sistem ini dapat memberikan layanan yang optimal.

1.5 Batasan

Adapun Batasan dalam Penelitian ini sebagai berikut:

1. Sistem hanya mencakup pengelolaan angka kredit aktivitas mahasiswa, pengajuan proposal skripsi, dan pelacakan alumni di STMIK "AMIKBANDUNG", tidak mencakup institusi lain.

2. Kegiatan yang dicatat sebagai angka kredit aktivitas adalah kegiatan akademik dan non-akademik yang telah diakui oleh pihak kampus, serta pengajuan proposal skripsi yang memenuhi syarat yang ditetapkan.
3. Fitur sistem meliputi pengajuan kredit aktivitas *online*, pengajuan proposal skripsi, persetujuan digital oleh dosen wali, pelacakan status pengajuan, dan akses informasi *real-time*. Sistem ini tidak mencakup integrasi dengan sistem eksternal lain di luar sistem administrasi kampus.
4. Pembuatan sistem informasi berupa aplikasi berbasis web untuk pengelolaan angka kredit aktivitas mahasiswa, pengajuan proposal skripsi, dan pelacakan alumni di STMIK "AMIKBANDUNG".
5. Metode pengembangan sistem yang digunakan adalah metode *Agile*, dengan pendekatan iteratif untuk memastikan fleksibilitas dalam memenuhi kebutuhan pengguna.
6. Aktor-aktor dalam sistem meliputi mahasiswa, dosen wali, dan pihak kemahasiswaan, masing-masing dengan peran spesifik dalam pelaporan, validasi, pengelolaan data, serta pelacakan alumni.

1.6 Metode Penelitian

Untuk mencapai tujuan penelitian dan menjawab rumusan masalah, penelitian ini menggunakan pendekatan *Research and Development (R&D)* yang dikombinasikan dengan metode *System Development Life Cycle (SDLC)*, khususnya model *Agile*. Metode ini dipilih karena pengembangan sistem dapat dilakukan secara bertahap dengan tahapan yang jelas. Tahapan dalam metode ini adalah sebagai berikut:

1. Analisis Kebutuhan

Melakukan analisis terhadap proses pengelolaan kredit poin, pengajuan proposal skripsi, dan pelacakan alumni yang sedang berjalan melalui wawancara, observasi, dan pengumpulan dokumen. Analisis ini bertujuan untuk mengidentifikasi kebutuhan pengguna, baik mahasiswa, dosen wali, maupun bagian kemahasiswaan.

2. Perancangan Sistem

Membuat desain sistem informasi menggunakan pendekatan *Use Case Diagram*, *ERD (Entity Relationship Diagram)*, dan rancangan antarmuka untuk

menggambarkan fitur-fitur utama dan alur kerja sistem, termasuk pengajuan proposal dan pelacakan alumni.

3. Implementasi Sistem

Mengembangkan sistem berbasis web menggunakan teknologi yang sesuai, seperti *PHP*, *Laravel*, atau *framework* lain, serta menggunakan *database* seperti MySQL. Sistem ini akan mencakup modul untuk pengelolaan angka kredit aktivitas mahasiswa, pengajuan proposal skripsi, dan pelacakan alumni.

4. Pengujian Sistem

Melakukan pengujian sistem dengan metode *black box testing* untuk memastikan sistem berjalan sesuai kebutuhan, serta mendapatkan umpan balik dari pengguna awal (mahasiswa dan dosen wali) terkait semua fitur, termasuk pengajuan proposal dan pelacakan alumni.

5. Evaluasi dan Dokumentasi

Mengevaluasi hasil pengujian dan menyusun dokumentasi akhir yang mencakup laporan implementasi, analisis umpan balik pengguna, serta rekomendasi pengembangan lanjutan untuk meningkatkan sistem, termasuk fitur-fitur terkait pengajuan proposal dan pelacakan alumni.

1.7 Waktu dan Tempat Penelitian

Penelitian ini dilakukan di STMIK “AMIKBANDUNG”, Jl. Jakarta No.28, Kebonwaru, Batununggal, Kota Bandung, Jawa Barat 40272.

Sebelum melaksanakan penelitian terlebih dahulu penulis melakukan wawancara sebanyak dua kali bersama dengan kemahasiswaan yaitu ibu Yolanda Georgia, S.Sn., M.Ds, dimana dalam wawancara tersebut penulis menanyakan beberapa hal terkait permasalahan yang sedang dihadapi saat ini serta kebutuhan apa saja yang perlu ditambahkan dalam sistem baik terkait aktor maupun fitur yang diperlukan.

Bersadarkan wawanacara yang dilakukan kemudian penulis melakukan penelitian dengan jadwal penelitian sebagai berikut:

Tabel 1. 1 Jadwal Kegiatan Penelitian

No	Jenis Kegiatan	Jadwal Kegiatan																							
		Agustus				September				Oktober				November				Desember				Januari			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4

1	Pengumpulan Data																			
3	Analisis Data																			
4	Perancan Aplikasi																			
5	Pengembangan sistem																			
6	Pengujian Aplikasi																			

1.8 Sitematika Penulisan

Laporan ini terbagi ke dalam beberapa bab yang mencakup berbagai aspek penting dari penelitian. Bab-bab tersebut meliputi Pendahuluan, Landasan Teori, Analisis dan Perancangan, Implementasi dan Pengujian, serta Penutup.

BAB I PENDAHULUAN

Berisi tentang Latar Belakang, Identifikasi Masalah, Rumusan Masalah, Tujuan Penelitian, Batasan Masalah, Metode Penelitian yang terdiri dari Teknik Pengumpulan Data dan Metode Perancangan, Tempat dan Waktu Penelitian serta Sistematika Penulisan

BAB II LANDASAN TEORI

Bab ini menguraikan konsep-konsep penting yang digunakan dalam penelitian ini, seperti konsep sistem, pengertian Kredit Aktivitas Mahasiswa, pengertian Unit Kegiatan Mahasiswa, pengertian Alumni dan pelacakan alumni, pengertian Metode System Development Life Cycle (*SDLC*), pengertian *Visual Studio Code, framework* yang digunakan dalam pengembangan sistem di ,yaitu *Vue.Js* dan *Express.JS*, serta pengertian MySQL (*My Structure Query Language*).

BAB III ANALISIS DAN PERANCANGAN

Bab ini membahas analisis dan perancangan sistem informasi kemahasiswaan di STMIK "AMIKBANDUNG". Penulis menggunakan pendekatan *System Development Life Cycle (SDLC)* dan mengumpulkan data melalui wawancara dengan pihak kemahasiswaan. Bab ini juga mencakup diagram alir (*flowmap*) sistem yang sedang berjalan dan yang diusulkan.

BAB IV IMPLEMENTASI DAN PENGUJIAN

Bab ini menjelaskan implementasi sistem informasi kemahasiswaan di STMIK "AMIKBANDUNG" serta pengujian menggunakan metode *Black box*. Pengujian dilakukan untuk memastikan bahwa sistem berfungsi sesuai dengan harapan pengguna.

BAB V KESIMPULAN DAN SARAN

Bab ini merangkum kesimpulan yang diperoleh dari pelaksanaan penelitian dan penyusunan laporan serta memberikan saran yang diperlukan terkait penggunaan atau pengembangan sistem di masa mendatang.

DAFTAR PUSTAKA

Memuat informasi dan teori literasi dari sumber-sumber yang dikutip, baik dari jurnal maupun situs web, yang digunakan dalam penyusunan penelitian ini.

BAB II

TINJAUAN PUSTAKA

2.1 Konsep Sistem

Sistem adalah suatu jaringan kerja dari prosedur prosedur yang saling berhubungan, berkumpul bersama sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu. Penjelasan di atas menjelaskan bahwa sistem bekerja dalam suatu jaringan kerja dari suatu prosedur yang saling berhubungan satu sama lain untuk menyelesaikan tujuan dan sasaran yang dimaksud. Sistem juga diartikan sebagai sekumpulan elemen yang bekerja sama dalam suatu kesatuan untuk melaksanakan suatu fungsi yang berguna. Dalam bukunya Jogiyanto sistem dapat didefinisikan dengan pendekatan prosedur dan pendekatan komponen. [9]

2.1.1. Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat – sifat tertentu yaitu [10]:

a. Komponen Sistem (*Component*).

Suatu sistem terdiri dari sejumlah komponen yang sering disebut dengan subsistem yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai bagian-bagian dari sistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

b. Batasan Sistem (*Boundary*).

Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luar. Batas sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (*scope*) sistem itu sendiri.

c. Lingkungan Luar Sistem (*Environments*).

Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar dari sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.

d. Penghubung Sistem (*Interface*).

Penghubung merupakan media yang menghubungkan antara subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya.

e. Masukan Sistem (*Input*).

Masukan yaitu energi yang dimasukan kedalam sistem, dimana dapat berupa masukan perawatan (*maintenance Input*) dan masukan sinyal (*signal Input*). Masukan perawatan adalah energi yang di-Input-kan supaya sistem dapat beroperasi, sedang masukan sinyal adalah energi yang diproses untuk didapatkan keluaran.

f. Keluaran Sistem (*Output*).

Keluaran yaitu hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembungan. Sedangkan informasi adalah keluaran yang dibutuhkan.

g. Pengolahan Sistem (*Process*).

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan merubah masukan menjadi keluaran.

h. Sasaran Sistem (*Objective*).

Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.

2.1.2 Klasifikasi Sistem

Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada didalam sistem tersebut. Oleh karena itu, sistem dapat diklasifikasikan dari beberapa sudut pandang diantaranya[11].

a. Sistem abstrak dan sistem fisik.

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologi, yaitu sistem yang berisi gagasan tentang hubungan manusia dengan Tuhan, sedangkan sistem fisik merupakan

sistem yang ada secara fisik, misalkan sistem komputer, sistem sekolah, sistem penjualan dan lain sebagainya.

b. Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang dan malam. Sistem buatan manusia adalah sistem yang melibatkan interaksi antara manusia dengan mesin. Yang disebut *human machine system*. Sistem informasi berbasis computer merupakan contoh sistem *human machine system* karena menyangkut penggunaan computer yang berinteraksi dengan manusia.

c. Sistem Deterministik dan Sistem Probabilistik Sistem

Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut dengan sistem deterministik. Sistem komputer adalah contoh dari sistem yang tingkah lakunya dapat dipastikan berdasarkan pemrograman komputer yang dijalankan. Sedangkan sistem yang bersifat probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.

d. Sistem Terbuka dan Sistem Tertutup

Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa campur tangan pihak luar. Sedangkan sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk subsistem lainnya.

2.2 Pengembangan Sistem

Berdasarkan definisi yang diberikan oleh Kamus Besar Bahasa Indonesia (KBBI), pengembangan yang berasal dari kata dasar “kembang” berarti proses, cara, atau perbuatan mengembangkan.

Pengembangan sistem (*system development*) dapat berarti menyusun suatu sistem baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada. Sistem yang lama perlu diperbaiki atau diganti disebabkan karena beberapa hal, yaitu sebagai berikut ini

1. Adanya permasalahan-permasalahan (*problems*) yang timbul di sistem yang lama.

Permasalahan yang timbul dapat berupa:

a. Ketidakberesan

Ketidakberesan dalam sistem yang lama menyebabkan sistem yang lama tidak dapat beroperasi sesuai dengan yang diharapkan.

b. Pertumbuhan organisasi

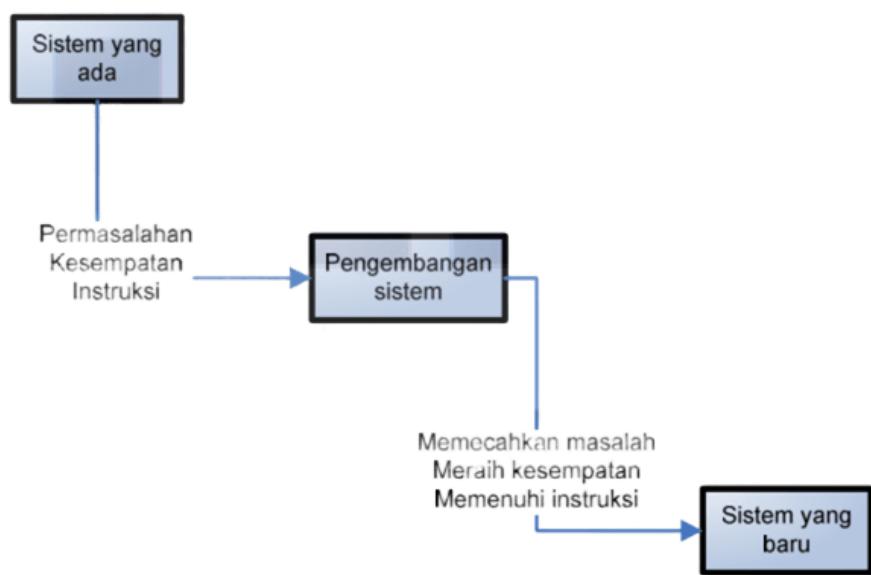
Pertumbuhan organisasi yang menyebabkan harus disusunnya sistem yang baru. Pertumbuhan organisasi, misalnya kebutuhan informasi yang semakin luas dan meningkat.

2. Untuk meraih kesempatan-kesempatan (*opportunities*)

Teknologi semakin berkembang dengan cepatnya, perangkat keras komputer, perangkat lunak dan teknologi komunikasi telah begitu cepat berkembang.

3. Adanya instruksi-instruksi (*directives*)

Penyusunan sistem yang baru dapat terjadi karena adanya instruksi-instruksi dari atas pimpinan ataupun dari luar organisasi, seperti misalnya peraturan pemerintah karena adanya permasalahan, kesempatan dan instruksi, maka sistem yang baru perlu dikembangkan untuk memecahkan permasalahan-permasalahan yang timbul, meraih kesempatan-kesempatan yang ada atau memenuhi instruksi yang diberikan.



Gambar: 2. 1 Proses munculnya sistem baru

Dengan adanya sistem baru diharapkan dapat memperoleh peningkatan-peningkatan yang berguna. Peningkatan-peningkatan ini berhubungan dengan *Performance, Information, Economy, Control, Efficiency* dan *Service*.

- a. *Performance* (kinerja), peningkatan terhadap kinerja (hasil kerja) sistem yang baru sehingga menjadi lebih efektif. Kinerja dapat diukur dari *throughput* dan *response time*. *Throughput* adalah jumlah dari pekerjaan yang di dapat saat tertentu. *Response time* adalah rata-rata waktu yang tertunda diantara dua transaksi atau pekerjaan ditambah dengan waktu response untuk menanggapi pekerjaan tersebut.
- b. *Information* (informasi), peningkatan terhadap informasi yang disajikan.
- c. *Economy* (ekonomis), peningkatan terhadap manfaat-manfaat atau keuntungan-keuntungan atau penurunan-penurunan biaya yang terjadi.
- d. *Control* (pengendalian), peningkatan terhadap pengendalian untuk mendeteksi dan memperbaiki kesalahan-kesalahan serta kecurangan-kecurangan yang ada dan akan terjadi.
- e. *Efficiency* (efisiensi), peningkatan terhadap efisiensi operasi. Efisiensi berbeda dengan ekonomis. Bila ekonomis berhubungan dengan sumber daya yang digunakan sedangkan efisiensi berhubungan dengan bagaimana sumber daya tersebut digunakan dengan pemborosan yang paling minimum.
- f. *Service* (pelayanan), peningkatan terhadap pelayanan yang diberikan oleh sistem.

2.3 Kredit Aktivitas Mahasiswa (AKAM)

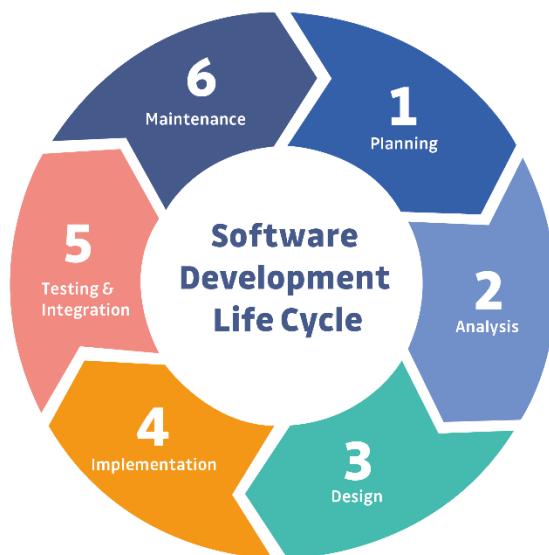
Kredit Aktivitas Mahasiswa merupakan sistem untuk menyatakan pengakuan prestasi pengembangan *softs kills* di bidang kemahasiswaan. Kegiatan yang dilakukan oleh mahasiswa harus dilaporkan ke Bagian Kemahasiswaan, agar dapat memperoleh nilai kredit sesuai dengan jenis kegiatannya. Kredit aktivitas yang telah dikumpulkan mahasiswa selama masa perkuliahan, dimuat dalam Surat Keterangan Pendamping Ijazah (SKPI) yang akan diterima oleh lulusan saat wisuda. Dengan peraturan ini, untuk lulus mahasiswa harus mengumpulkan sejumlah kredit aktivitas, sesuai dengan target yang telah ditetapkan tepat pada waktunya.

Setiap mahasiswa diwajibkan untuk melaporkan kegiatan yang telah diikuti atau prestasi yang berhasil diraih melalui Sistem Penilaian Angka Kredit Aktivitas Mahasiswa (AKAM). Laporan ini mencakup berbagai bidang, seperti penalaran dan

keilmuan, organisasi dan kepemimpinan, pengabdian masyarakat, minat dan bakat, serta bidang lainnya. Mahasiswa juga perlu melampirkan bukti fisik yang sah, seperti sertifikat, surat keputusan, surat tugas, atau dokumen pendukung lain yang sesuai dengan aturan yang berlaku[3]. Proses validasi atas kegiatan mahasiswa dilakukan oleh Bagian Kemahasiswaan sebagai pihak yang berwenang sesuai dengan bidang kegiatannya.

2.4 Metode *System Development Life Cycle* (SDLC)

SDLC merupakan suatu urutan dari beberapa proses secara bertahap didalam merancang dan mengembangkan sistem yang dikenal juga dengan nama *Information System Development* atau juga *Application Development*. Komponen SDLC meliputi perencanaan sistem, analisis sistem, rancangan sistem, pelaksanaan dan pengoperasian sistem[12]



Gambar: 2. 2 Metode System Development Life Cycle (SDLC)

Pengembangan sistem Angka Kredit Aktivitas Mahasiswa ini akan dirancang dengan tahapan sebagai berikut[13]:

a. Analisis kebutuhan *software*

Analisa dan pengumpulan kebutuhan sistem, yang berisi tentang informasi yang akan diteliti dilakukan pada tahap ini. Pendekatan penelitian ini diperoleh dari analisis kebutuhan-kebutuhan mekanisme pengumpulan Angka Kredit Aktivitas Mahasiswa (AKAM). *User* akan diperlihatkan kembali hasil dari analisa dan pengumpulan tersebut.

b. Desain

Pada proses desain, dilakukan dengan cara merumuskan *flowchart* atau detail algoritma prosedural dari hasil perencanaan tentang sistem penerimaan kas yang akan dikembangkan.

c. Pembuatan kode program

Mentranslasikan desain menggunakan bahasa pemrograman ke dalam program perangkat lunak. Hasil dari proses ini adalah program komputer berjalan sesuai dengan desain yang dirancang.

d. Pengujian

Proses selanjutnya adalah pengujian program perangkat lunak. Pengujian sistem sangat penting untuk memberikan jaminan kualitasnya dan membuktikan bahwa fungsinya telah beroperasi dengan benar (Ijudin & Saifudin, 2020). Uji coba dilakukan sebelum sistem informasi diterapkan pada objek sistem dengan cara menguji tingkat *error* dari sistem tersebut.

e. Pemeliharaan

Setelah perangkat lunak bisa digunakan, maka proses paling akhir dari siklus pengembangan perangkat lunak adalah melakukan pemeliharaan. Pada proses pemeliharaan terdapat kegiatan yang harus dilakukan, antara lain:

- *Corrective Maintenance* : mengoreksi atau revisi sistem informasi penerimaan kas yang dikembangkan karena terdeteksi *error* pada program yang digunakan.
- *Adaptive Maintenance* : penyesuaian penerapan sistem Angka Kredit Aktivitas Mahasiswa yang ada di lapangan dengan target pada program.
- *Perfective Maintenance* : Bila sistem informasi penerimaan kas berhasil digunakan oleh *user* dengan baik. Pemeliharaan bertujuan untuk meningkatkan kemampuannya seperti menambahkan beberapa fungsi, meningkatkan kinerja dan lainnya.

Pengembangan SDLC (*System Development Life Cycle*) diperlukan sebuah sistem yang mendukung untuk membuat dan menyimpan struktur basis data. Begitu juga dengan sistem yang mendukung dan merancang serta menampilkan sebuah sistem dengan penerapan DBMS (*Database Management System*). Maka yang digunakan untuk menampilkan sistem dan merancang adalah dengan MySQL (*My Structure Query Language*)

2.5 MySQL (*My Structure Query Language*)

MySQL merupakan RDBMS (*Relational Database Management System*) server. RDBMS adalah program yang memungkinkan pengguna *database* untuk membuat, mengelola, dan menggunakan data pada suatu model *relational*. Dengan demikian, tabel-tabel yang ada pada *database* memiliki relasi antara satu tabel dengan tabel lainnya.

Fungsi dari MySQL adalah untuk membuat dan mengelola *database* pada sisi server yang memuat berbagai informasi dengan menggunakan bahasa SQL. Fungsi lain yang dimiliki adalah memudahkan pengguna dalam mengakses data berisi informasi dalam bentuk string (teks), yang dapat diakses secara personal maupun publik dalam *web* [14].

2.6 Visual Studio Code

Visual Studio Code adalah *Software* yang sangat ringan, namun kuat editor kode sumbernya yang berjalan dari *desktop*. Muncul dengan *built-in* dukungan untuk *JavaScript*, naskah dan *Node.js* dan memiliki *array* beragam ekstensi yang tersedia untuk bahasa lain, termasuk *C ++*, *C #*, *Python*, dan *PHP*. Hal ini didasarkan sekitar *Github* ini Elektron, yang merupakan versi *cross-platform* dari Atom komponen *code-editing*, berdasarkan *JavaScript* dan *HTML5*. Editor ini adalah fitur lengkap lingkungan pengembangan terpadu (*IDE*) dirancang untuk pengembang yang bekerja dengan teknologi *cloud* yang terbuka *Microsoft*. *Visual Studio Code* menggunakan *open source* *NET* perkakas untuk memberikan dukungan untuk *ASP.NET C #* kode, membangun alat pengembang *Omnisharp .NET* dan *compiler Roslyn*. Antarmuka yang mudah untuk bekerja dengan, karena didasarkan pada gaya *explorer* umum, dengan panel di sebelah kiri, yang menunjukkan semua file dan *folder* Anda memiliki akses ke panel editor di sebelah kanan, yang menunjukkan isi dari file yang telah dibuka. Dalam hal ini, editor telah dikembangkan dengan baik, dan menyenangkan pada mata. Ia juga memiliki fungsi yang baik, dengan *intellisense* dan *autocomplete* bekerja dengan baik untuk *JSON*, *CSS*, *HTML*, dan *Node.js* [15].

Visual Studio Code telah dirancang untuk bekerja dengan alat-alat yang ada, dan *Microsoft* menyediakan dokumentasi untuk membantu pengembang bersama, dengan bantuan untuk bekerja dengan *ASP.NET 5*, *Node.js*, dan *Microsoft* naskah, serta alat-alat yang dapat digunakan untuk membantu membangun dan mengelola aplikasi

Node.js. *Visual Studio Code* benar-benar sedang ditargetkan pada pengembang *JavaScript* yang ingin alat pengembangannya lengkap untuk *scripting* server-side mereka dan yang mungkin ingin usaha dari *Node.js* untuk kerangka berbasis *NET*. *Visual Studio Code*, adalah lintas platform kode Editor ringan, yang dapat digunakan oleh siapa saja untuk membangun aplikasi untuk Web [16].

Dalam pembuatan aplikasi berbasis web ini penulis menggunakan beberapa *framework* yaitu:

2.6.1 *Vue.js*

Vue.js adalah kerangka kerja *JavaScript* yang bersifat progresif, bersumber terbuka untuk membangun antarmuka pengguna. Integrasi ke dalam proyek yang menggunakan pustaka *JavaScript* lainnya mudah dilakukan dengan *Vue* karena ia dirancang untuk bisa berjalan hanya di beberapa bagian halaman *web*[17].

Vue.js juga bisa berfungsi sebagai kerangka kerja aplikasi web yang mampu menjalankan aplikasi *single-page* yang canggih. Dengan menggunakan kerangka kerja *Vue.js* dapat direpresentasikan secara dini dari sisi *server* atau biasa disebut *server side rendering*[18].

2.6.2 *Express.JS*

Express.js adalah sebuah *framework web* yang menyediakan fitur-fitur untuk membuat aplikasi *web* yang cepat, ringan, dan fleksibel. *Express.js* mendukung *routing*, *middleware*, *template engine*, dan *API*. *Express js* juga kompatibel dengan banyak modul, *framework*, dan *library* lain yang berbasis *Node.js*. *Express.js* juga dikenal karena fleksibilitasnya, memungkinkan *developer* untuk memilih bagian mana dari *framework* yang ingin digunakan. Developer juga bisa memilih dari berbagai plugin atau *middleware* yang tersedia untuk memberikan fungsionalitas tambahan sesuai kebutuhan[19].

2.7 *Unified Modeling Language (UML)*

UML (*Unified Modeling Language*) adalah alat bantu yang sudah menjadi standar dalam dunia pengembangan sistem perangkat lunak berorientasi objek. *UML* menjadi bahasa yang handal dalam memvisualisasi rancangan sistem perangkat lunak. *UML* memungkinkan para pengembang sistem membuat blue print dalam bentuk yang baik dan mudah dimengerti sehingga bisa hasil rancangan bisa dikomunikasikan dengan pihak lain [20].

UML diagram memiliki tujuan utama untuk membantu tim pengembangan proyek berkomunikasi, mengeksplorasi potensi desain, dan memvalidasi desain arsitektur perangkat lunak atau membuat program. UML mempunyai tiga kategori utama yaitu struktur diagram, behaviour diagram dan interaction diagram. Dimana masing-masing kategori tersebut memiliki diagram yang menjelaskan arsitektur sistem dan saling terintegrasi

2.7.1 Diagram Use Case (*Use Case Diagram*)

Use Case Diagram menjelaskan hubungan fungsional yang diharapkan dari perancangan sebuah sistem. *Use Case Diagram* sangat berpengaruh dalam perancangan sebuah sistem atau dapat dikatakan *Use Case Diagram* adalah dasar dari perencanaan sebuah sistem. Identifikasi aktor mendeskripsikan interaksi antar aktor di dalam sistem.

2.7.2 Diagram Aktivitas (*Activity Diagram*)

Activity Diagram merupakan diagram yang mampu menjelaskan secara prosedural alur proses dari sebuah sistem. Dalam diagram ini dimungkinkan untuk mengevaluasi kemungkinan adanya lebih dari satu jalur yang terbentuk dan berjalan secara bersamaan. Penggambaran dari *Activity Diagram* dimulai dari initial node hingga berakhir di end node. Yang dapat menjadi catatan bahwa initial node dalam sebuah *Activity Diagram* diperbolehkan lebih dari satu. Hal ini dilakukan untuk mengakomodasi jika sistem yang dimodelkan memiliki lebih dari satu input.

2.7.3 Diagram Kelas (*Class Diagram*)

Class diagram adalah sebuah diagram yang menunjukkan hubungan antar class yang didalamnya terdapat atribut dan fungsi dari suatu objek. *Class diagram* mempunyai 3 relasi dalam penggunaannya, yaitu:

1. *Assosiation* adalah sebuah hubungan yang menunjukkan adanya interaksi antar class. Hubungan ini dapat ditunjukkan dengan garis dengan mata panah terbuka di ujungnya yang mengindikasikan adanya aliran pesan dalam satu arah.
2. *Generalizations* adalah sebuah hubungan antar class yang bersifat dari khusus ke umum.
3. *Constraint* adalah sebuah hubungan yang digunakan dalam sistem untuk memberi batasan pada sistem sehingga didapat aspek yang tidak fungsional

2.7.4 Diagram Sekuen (*Sequence Diagram*)

Sequence diagram adalah sebuah diagram yang menggambarkan kolaborasi dari objek-objek yang saling berinteraksi antar elemen dari suatu class. Berikut ini merupakan komponen dalam *sequence diagram*:

1. *Activations*: Menjelaskan tentang eksekusi dari fungsi yang dimiliki oleh suatu objek.
2. *Actor*: Menjelaskan tentang peran yang melakukan serangkaian aksi dalam suatu proses.
3. *Collaboration boundary*: Menjelaskan tentang tempat untuk lingkungan percobaan dan digunakan untuk memonitor objek.
4. *Parallel vertical lines*: Menjelaskan tentang suatu garis proses yang menunjuk pada suatu state.
5. *Process es*: Menjelaskan tentang tindakan/aksi yang dilakukan oleh aktor dalam suatu waktu.
6. *Window*: Menjelaskan tentang halaman yang sedang ditampilkan dalam suatu proses.
7. *Loop*: Menjelaskan tentang model logika yang berpotensi untuk diulang beberapa kali

2.8 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah jaringan yang menggambarkan tempat penyimpanan data sistem dengan gambaran yang lebih mudah dipahami. ERD adalah representasi grafis dari entitas, hubungan, dan batasan integritas yang dihasilkan dari aktivitas pengembangan. Relasi antar himpunan entitas dengan kardinalitas relasi ada tiga macam, yaitu:

- a. Relasi satu ke satu (*one to one*), satu *entity* hanya mempunyai satu relasi pada *entity* lainnya.
- b. Relasi satu ke banyak (*one to many*), satu *entity* dapat mempunyai relasi lebih dari satu pada *entity* lainnya.
- c. Relasi banyak ke banyak (*many to many*), banyak *entity* dapat berrelasi dengan banyak *entity* yang lain

2.9 Pengujian *Black box*

Black box testing adalah tahapan yang digunakan untuk menguji kelancaran program yang telah dibuat. pengujian *black box* testing penting dilakukan dengan tujuan agar tidak terjadi kesalahan alur program yang telah dibuat [21].

Black box Testing merupakan salah satu metode yang mudah digunakan karena hanya memerlukan batas bawah dan batas atas dari data yang di harapkan, Estimasi banyaknya data uji dapat dihitung melalui banyaknya field data entri yang akan diuji, aturan entri yang harus dipenuhi serta kasus batas atas dan batas bawah yang memenuhi. Dan dengan metode ini dapat diketahui jika fungsionalitas masih dapat menerima masukan data yang tidak diharapkan maka menyebabkan data yang disimpan kurang valid. Pengujian ini biasanya tidak dapat melakukan verifikasi proses sistem internal dan hanya hasil actual yang terlihat oleh pengguna sistem.[22]

Black box testing menemukan beberapa hal seperti fungsi yang tidak benar atau tidak ada, kesalahan antarmuka (*interface error*), kesalahan pada struktur data dan akses basis data, kesalahan performasi (*performance error*) dan kesalahan inisialisasi dan terminasi. Terdapat beberapa Teknik dari *black box testing* diantaranya adalah *equivalence partitioning*, *boundary value analysis* atau *limit testing*.[23]

Solusi praktis peningkatan akurasi perlu dilakukan segera guna memperbaiki celah *error* yang telah ditemukan, selanjutnya dilakukan pengujian keamanan secara intensif melalui jaringan internal (*whitebox penetration testing*) secara berkala oleh System Administrator atau Pengelola Sistem Informasi, khususnya bagi yang mengelola perangkat lunak tersebut dan Untuk mencapai tingkat akurasi, dimana semua parameter akurasi yang terkait aspek kerahasiaan, integritas data, dan avalibilitas data dapat terpenuhi, maka harus dipertimbangkan metode lain yang dapat dijadikan tolak ukur standar keamanan informasi[24]

2.10 Unit Kegiatan Mahasiswa (UKM)

UKM (singkatan dari Unit Kegiatan Mahasiswa) adalah lembaga kemahasiswaan tempat berhimpunnya para mahasiswa yang memiliki kesamaan minat, kegemaran, kreativitas, dan orientasi aktivitas penyaluran kegiatan ekstrakurikuler di dalam kampus. UKM merupakan organisasi kemahasiswaan yang mempunyai tugas merencanakan, melaksanakan, dan mengembangkan kegiatan ekstrakurikuler kemahasiswaan yang bersifat penalaran, minat dan kegemaran, kesejahteraan, dan

minat khusus sesuai dengan tugas dan tanggung jawabnya. Kedudukan lembaga ini berada pada wilayah universitas yang secara aktif mengembangkan sistem pengelolaan organisasi secara mandiri[25].

2.11 Pengertian Alumni

Alumni adalah individu, baik siswa maupun mahasiswa, yang telah menyelesaikan pendidikan pada suatu jenjang sesuai dengan peraturan yang berlaku di institusi pendidikan, sehingga mereka dapat disebut sebagai alumni. Setiap alumni memiliki tanggung jawab terhadap institusi yang telah memberikan gelar, dan dengan demikian, mereka diharapkan dapat menjaga nama baik institusi tersebut. Alumni dapat diartikan sebagai orang yang pernah menempuh pendidikan di suatu lembaga pendidikan, meskipun tidak harus lulus atau memperoleh gelar. Namun, dalam konteks penelitian ini, alumni merujuk pada mereka yang telah lulus dari salah satu program studi dan menerima ijazah atau sertifikasi dari perguruan tinggi[26].

2.12 *Tracker* Alumni/Pelacakan Alumni

Tracker alumni adalah sebuah studi mengenai lulusan lembaga penyelenggara pendidikan tinggi yang bertujuan untuk menyediakan informasi tentang alumni yang sudah bekerja atau belum bekerja, serta penilaian alumni terhadap lembaga penyelenggara pendidikan dan kepuasan pengguna alumni terhadap kompetensi yang dimiliki dalam dunia kerja. *Tracker* alumni bermanfaat sebagai sumber data bagi perguruan tinggi mengenai kondisi lulusan dan sebagai sarana evaluasi untuk memperbaiki serta meningkatkan kualitas pendidikan dan pelayanan, serta mewujudkan visi dan misi yang ingin dicapai. Fokus utama dari *tracker* alumni adalah memperoleh informasi dari lulusan yang sudah bekerja atau belum bekerja, yang dapat digunakan sebagai dasar pengambilan kebijakan untuk pengembangan atau penyempurnaan suatu institusi [26].

BAB III

ANALISIS DAN PERANCANGAN

3.1 Analisis Sistem Berjalan

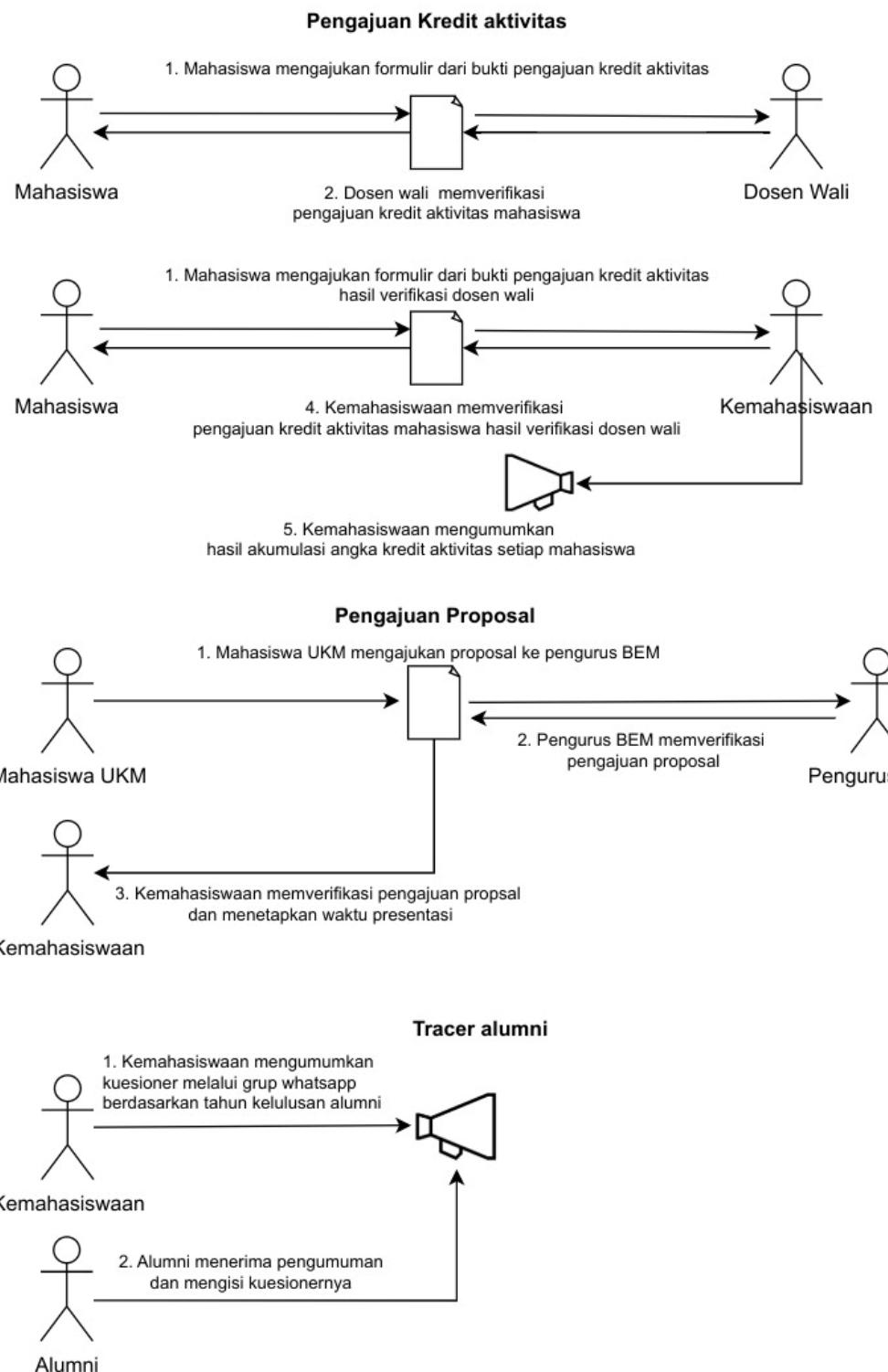
Saat ini, pengelolaan pengajuan angka kredit aktivitas mahasiswa, pengajuan proposal kegiatan, dan *Tracer* alumni di STMIK AMIK Bandung dilakukan melalui proses yang masih manual dan terpisah.

Untuk pengajuan angka kredit aktivitas mahasiswa, mahasiswa mengisi formulir yang mencakup informasi tentang aktivitas yang telah dilakukan, seperti sertifikat atau bukti pendaftaran. Setelah mengisi formulir, mahasiswa mencetak bukti aktivitas yang relevan dan menyerahkannya kepada dosen wali untuk diverifikasi. Proses ini sering kali mengalami keterlambatan, terutama pada proses verifikasi bisa memakan waktu yang lama karena mahasiswa perlu menemui dosen wali membuat janji temu terlebih dahulu baru bisa mengajukan verifikasi ke kemahasiswaan.

Pengajuan proposal kegiatan dilakukan secara manual, di mana mahasiswa mengajukan proposal kepada dosen pembimbing. Proposal ini hanya untuk keperluan pengarsipan dan tidak terintegrasi dengan sistem yang lebih luas, sehingga mempersulit pengelolaan dan pemantauan kegiatan.

Sementara itu, untuk *Tracer* alumni, STMIK AMIK Bandung menggunakan sistem web yang memungkinkan alumni untuk mengisi kuesioner secara online. Alumni menerima link untuk mengisi kuesioner dan diharuskan mengupload bukti bahwa mereka telah mengisi formulir tersebut. Meskipun sistem ini memudahkan pengumpulan data, masih terdapat tantangan dalam memastikan bahwa semua alumni berpartisipasi dan mengisi kuesioner dengan benar.

Secara keseluruhan, sistem yang berjalan saat ini memiliki beberapa kekurangan, termasuk keterlambatan dalam proses verifikasi, kurangnya integrasi antara pengajuan proposal dan pengelolaan data, serta tantangan dalam partisipasi alumni dalam *tracer*. Oleh karena itu, diperlukan pengembangan sistem yang lebih efisien dan terintegrasi untuk meningkatkan efektivitas pengelolaan aktivitas mahasiswa, proposal kegiatan, dan *tracer* alumni di STMIK AMIK Bandung, seperti gambar berikut ini.:



Gambar 3. 1 Skema Sistem Pengajuan Kredit Aktivitas yang berjalan

3.2 Analisis Sistem yang Diusulkan

Alur pengelolaan pengajuan angka kredit aktivitas mahasiswa, pengajuan proposal kegiatan, dan *tracer* alumni

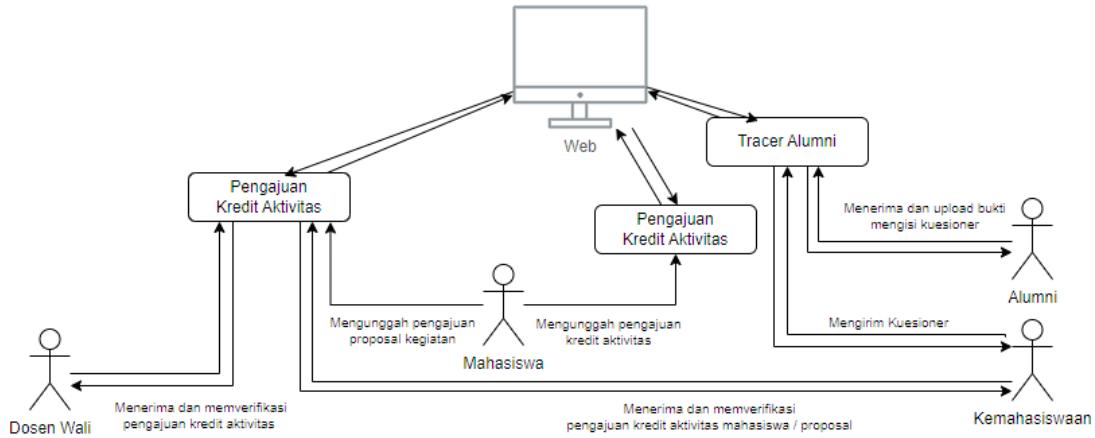
Dalam upaya meningkatkan efisiensi dan efektivitas pengelolaan pengajuan angka kredit aktivitas mahasiswa, pengajuan proposal kegiatan, dan *tracer* alumni, sistem yang diusulkan mencakup integrasi teknologi berbasis web.

Pengajuan Angka Kredit Aktivitas Mahasiswa: Sistem baru akan memungkinkan mahasiswa untuk mengajukan angka kredit secara online melalui platform yang terintegrasi. Mahasiswa dapat mengisi formulir pengajuan yang mencakup detail aktivitas dan mengunggah bukti pendukung, seperti sertifikat atau surat tugas. Proses verifikasi akan dilakukan secara digital oleh dosen wali dan pihak kemahasiswaan, yang akan mempercepat waktu respon dan mengurangi keterlambatan dalam proses verifikasi.

Pengajuan Proposal Kegiatan: Meskipun pengajuan proposal kegiatan tetap dilakukan secara manual, sistem yang diusulkan akan menyediakan fitur untuk mengarsipkan semua proposal yang diajukan. Dengan cara ini, semua dokumen akan tersimpan dalam satu sistem yang terorganisir, memudahkan akses dan pencarian di masa mendatang.

Tracer Alumni: Sistem baru juga akan menyediakan platform bagi alumni untuk mengisi kuesioner secara online. Alumni akan menerima link untuk mengisi kuesioner dan dapat mengunggah bukti pengisian, seperti screenshot atau dokumen lain yang relevan. Data yang terkumpul akan disimpan dalam *database* yang dapat diakses oleh pihak kampus untuk analisis lebih lanjut, membantu dalam evaluasi program dan pengembangan kurikulum.

Dengan implementasi sistem yang diusulkan ini, diharapkan STMIK AMIK Bandung dapat meningkatkan transparansi, efisiensi, dan kepuasan mahasiswa serta alumni dalam proses pengajuan dan pengelolaan data. Sistem ini akan memberikan kemudahan akses dan pengelolaan yang lebih baik, serta mendukung pengambilan keputusan yang lebih informasional bagi pihak kampus. Alur yang diusulkan seperti gambar berikut:



Gambar 3. 2 Skema Sistem yang Diusulkan

3.3 Analisis Kebutuhan Sistem yang Diusulkan

Penelitian kebutuhan mencakup analisis berbagai proses yang akan dilakukan oleh sistem yang direncanakan. Penelitian ini mengidentifikasi tiga kebutuhan utama sistem, yaitu kebutuhan data dan kebutuhan fungsional, yang diperoleh melalui metode observasi.

3.3.1. Definisi Pengguna Sistem

Pengguna sistem pengajuan angka kredit aktivitas mahasiswa di STMIK AMIK Bandung terdiri dari beberapa kategori, yaitu

Tabel 3. 1 Kategori Pengguna Sistem

No - DA	Aktor	Deskripsi
DA-01	Mahasiswa	Pengguna utama yang mengajukan angka kredit untuk aktivitas yang telah dilakukan, mengisi formulir pengajuan, memilih aktivitas, serta melihat status pengajuan.
DA-02	Dosen Wali	Bertanggung jawab untuk memverifikasi dan menyetujui pengajuan angka kredit yang diajukan oleh mahasiswa, serta memberikan catatan atau komentar pada pengajuan.
DA-03	Kemahasiswaan	Mengelola dan memverifikasi pengajuan angka kredit yang telah disetujui oleh dosen wali, memberikan informasi kepada mahasiswa mengenai status pengajuan.
DA-04	Admin Sistem	Mengelola sistem secara keseluruhan, termasuk pemeliharaan data, serta memastikan keamanan dan kinerja sistem secara umum.

Tabel 3.1 menjelaskan mengenai pembagian kategori dalam pembagian sistem yaitu mahasiswa, dosen wali, kemahasiswaan, dan admin sistem.

3.3.2. Kebutuhan Fungsional

Kebutuhan fungsional mencangkup fitur-fitur yang harus dimiliki oleh sistem, serta tindakan yang dapat dilakukan oleh sistem.

Tabel 3. 2 Kebutuhan Fungsional Sistem

No - KF	Deskripsi
Aktor	Mahasiswa
KF-01	<i>Login</i>
KF-02	<i>Logout</i>
KF-03	Lihat Daftar Pengajuan Kredit Aktivitas
KF-04	Buat Pengajuan Kredit Aktivitas
KF-05	Membatalkan Pengajuan Kredit Aktivitas
KF-06	Lihat Daftar Pengajuan Proposal
KF-07	Buat Pengajuan Proposal
KF-08	Membatalkan Pengajuan Proposal
KF-09	Lihat Daftar <i>Tracer</i> Alumni
KF-10	Kirim Bukti Pengisian <i>Tracer</i> Alumni
KF-11	Mengubah Bukti Pengisian <i>Tracer</i> Alumni
KF-12	Membatalkan Bukti Pengisian <i>Tracer</i> Alumni
Aktor	Dosen
KF-13	<i>Login</i>
KF-14	<i>Logout</i>
KF-15	Lihat Detail Bukti Aktivitas Mahasiswa
KF-16	Edit Status Verifikasi Pengajuan Proposal
Aktor	Kemahasiswaan
KF-17	<i>Login</i>
KF-18	<i>Logout</i>
KF-19	Lihat Daftar Kategori Aktivitas
KF-20	Tambah Data Kategori Aktivitas
KF-21	Hapus Data Kategori Aktivitas
KF-22	Edit Data Kategori Aktivitas
KF-23	Lihat Daftar Pengajuan Aktivitas Mahasiswa
KF-24	Lihat Detail Bukti Aktivitas Mahasiswa
KF-25	Edit Status Verifikasi Pengajuan Aktivitas
KF-26	Lihat Daftar Pengajuan Proposal
KF-27	Lihat Detail Bukti pengajuan Proposal
KF-28	Edit Status Verifikasi Pengajuan Proposal
KF-29	Lihat Daftar <i>Tracer</i> Alumni
KF-30	Tambah Data <i>Tracer</i> Alumni
KF-31	Edit Data <i>Tracer</i> Alumni
KF-32	Hapus Data <i>Tracer</i> Alumni

Aktor	Admin
KF-33	<i>Login</i>
KF-34	<i>Logout</i>
KF-35	Lihat Daftar Program Studi
KF-36	Tambah Data Daftar Program Studi
KF-37	Edit Data Program Studi
KF-38	Hapus Data Program Studi
KF-39	Lihat Daftar Tahun Akademik
KF-40	Tambah Data Tahun Akademik
KF-41	Edit Data Tahun Akademik
KF-42	Hapus Data Tahun Akademik
KF-43	Lihat Daftar Unit Kegiatan Mahasiswa
KF-44	Tambah Data Unit Kegiatan Mahasiswa
KF-45	Edit Data Unit Kegiatan Mahasiswa
KF-46	Hapus Data Unit Kegiatan Mahasiswa
KF-47	Lihat Daftar Dosen
KF-48	Tambah Data Dosen
KF-49	Edit Data Dosen
KF-50	Hapus Data Dosen
KF-51	Lihat Daftar Mahasiswa
KF-52	Tambah Data Mahasiswa
KF-53	Edit Data Mahasiswa
KF-54	Hapus Data Mahasiswa

3.3.3. Kebutuhan Non Fungsional

Kebutuhan non-fungsional meliputi asepek-aspek teknis kebutuhan sistem yang berfokus pada karakteristik perilaku sistem.

Tabel 3. 3 Kebutuhan Non Fungsional

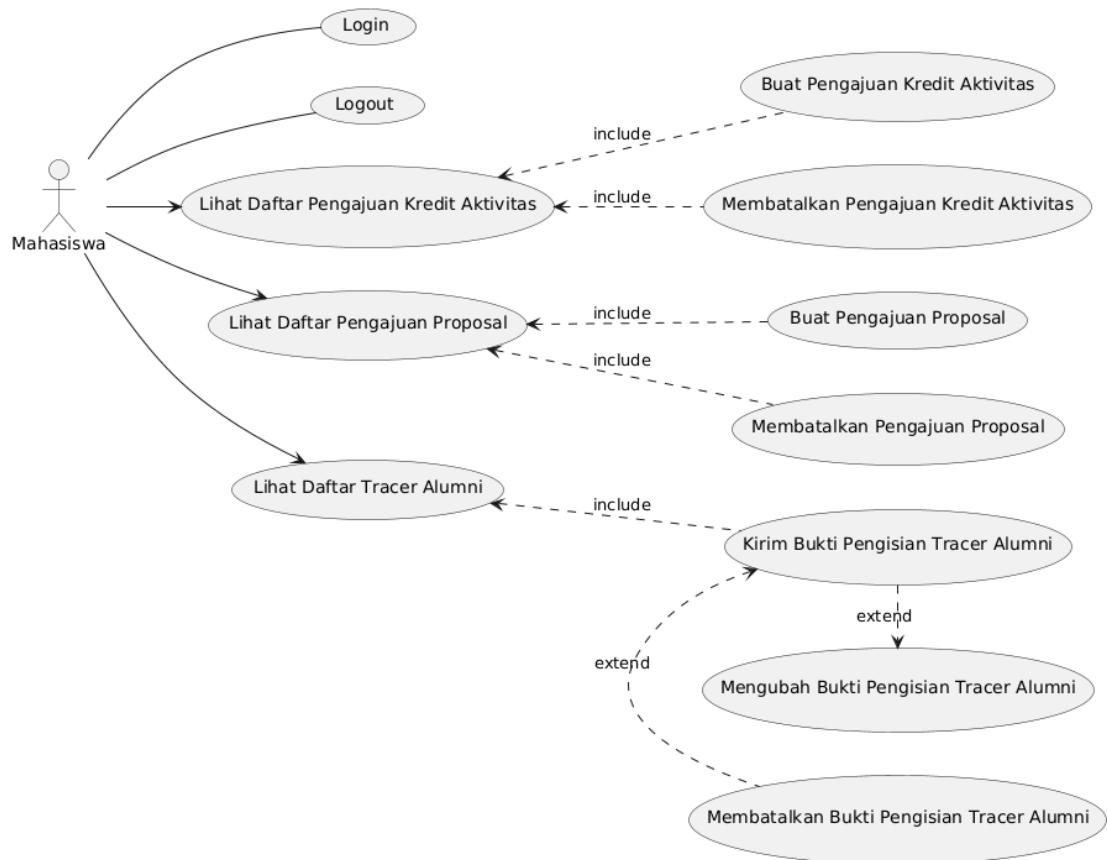
KNF	Deskripsi
KNF-01	Sistem dapat dijalankan pada beberapa <i>browser</i> seperti <i>Chrome</i> , <i>Safari</i> , <i>Mozilla Firefox</i> , dan <i>Microsoft Edge</i> .
KNF-02	Sistem terhubung ke jaringan internet dan <i>database</i> untuk menyimpan dan mengakses data pengajuan angka kredit.
KNF-03	Hanya pengguna yang terautentifikasi (mahasiswa, dosen wali, pihak kemahasiswaan, admin) yang dapat mengakses fitur yang sesuai dengan perannya.

3.3.4. Use Case Diagram

Berikut merupakan diagram *use case* dari pengembangan sistem kemahasiswaan yang menunjukkan fungsi-fungsi utama sistem dan hubungan interaksi antara pengguna dengan sistem yang dirancang.

a. Use Case Diagram Mahasiswa

Berikut merupakan *use case* diagram aktor mahasiswa:



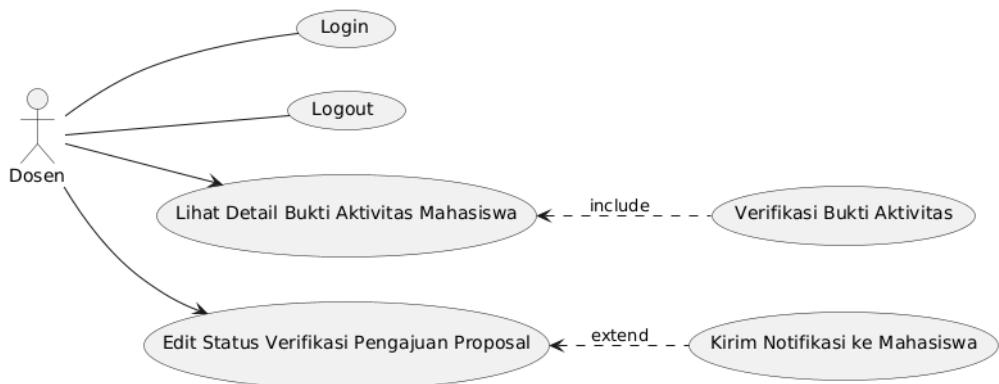
Gambar 3. 3 Use Case Diagram Mahasiswa

Pada gambar 3.3, dijelaskan tentang mahasiswa yang berinteraksi dengan sistem kemahasiswaan. Langkah pertama yang harus dilakukan mahasiswa adalah melakukan *Login* dengan meng-input data yang diperlukan. Setelah berhasil *login*, mahasiswa dapat memilih berbagai opsi, seperti Lihat Daftar Pengajuan Kredit Aktivitas, Buat Pengajuan Kredit Aktivitas, atau Membatalkan Pengajuan Kredit Aktivitas. Selanjutnya, mahasiswa juga dapat mengelola Pengajuan Proposal dengan memilih untuk Lihat Daftar Pengajuan Proposal, Buat Pengajuan Proposal, atau Membatalkan Pengajuan Proposal.

Terakhir, mahasiswa dapat mengakses fitur *Tracer Alumni*, di mana mereka dapat Lihat Daftar *Tracer Alumni*, Kirim Bukti Pengisian *Tracer Alumni*, Mengubah Bukti Pengisian *Tracer Alumni*, dan Membatalkan Bukti Pengisian *Tracer Alumni*.

b. Use Case Diagram Dosen

Berikut merupakan *use case diagram* aktor dosen:



Gambar 3.4 Use Case Diagram Dosen

Pada gambar 3.4, dijelaskan tentang dosen yang berinteraksi dengan sistem kemahasiswaan. Langkah pertama yang harus dilakukan dosen adalah melakukan *Login* dengan meng-input data yang diperlukan. Setelah berhasil *login*, dosen dapat mengakses beberapa opsi, yaitu Lihat Daftar Pengajuan Proposal, di mana dosen dapat melihat semua pengajuan proposal yang diajukan oleh mahasiswa, termasuk status dan detail dari masing-masing pengajuan. Selanjutnya, dosen dapat Lihat Detail Bukti Aktivitas Mahasiswa, yang memungkinkan mereka untuk memeriksa bukti aktivitas yang diajukan oleh mahasiswa untuk membantu dalam proses verifikasi. Terakhir, dosen memiliki kemampuan untuk Edit Status Verifikasi Pengajuan Proposal, di mana mereka dapat mengubah status verifikasi dari pengajuan proposal yang telah diajukan oleh mahasiswa, memberikan keputusan apakah proposal tersebut diterima atau ditolak.

c. Use Case Diagram Kemahasiswaan

Berikut merupakan *use case* diagram aktor kemahasiswaan:



Gambar 3. 5 Use Case Diagram Kemahasiswaan

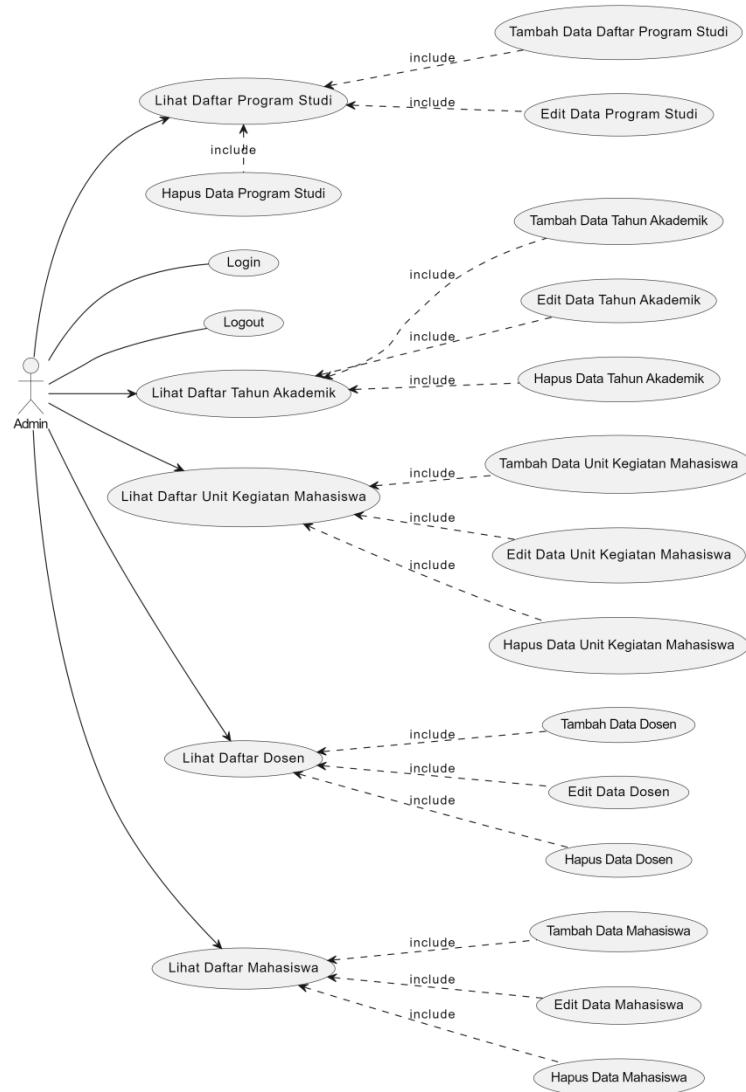
Pada gambar 3.4, dijelaskan tentang interaksi sistem kemahasiswaan. Langkah pertama yang harus dilakukan adalah melakukan *Login* dengan meng-input data yang diperlukan. Setelah berhasil *login*, pengguna dapat mengakses beberapa opsi, yaitu Lihat Daftar Kategori Aktivitas, di mana mereka dapat melihat semua kategori aktivitas yang tersedia. Pengguna juga memiliki kemampuan untuk Mengubah Data Kategori Aktivitas, yang memungkinkan mereka untuk memperbarui informasi yang ada, serta Hapus Data Kategori Aktivitas jika diperlukan.

Selanjutnya, pengguna dapat melihat Lihat Daftar Pengajuan Proposal, yang berisi semua proposal yang diajukan oleh mahasiswa. Mereka juga dapat Lihat Detail Bukti Aktivitas Mahasiswa untuk memeriksa bukti aktivitas yang diajukan oleh mahasiswa, serta Edit Status Verifikasi Pengajuan Proposal untuk mengubah status verifikasi dari pengajuan yang ada.

Selain itu, pengguna dapat mengakses fitur Lihat Daftar *Tracer* Alumni, di mana mereka dapat melihat data alumni yang terdaftar. Pengguna juga memiliki opsi untuk Tambah Data *Tracer* Alumni, Edit Data *Tracer* Alumni, dan Hapus Data *Tracer* Alumni sesuai kebutuhan.

d. Use Case Diagram Admin

Berikut merupakan *use case* diagram aktor admin:



Gambar 3. 6 Use Case Diagram Admin

Pada gambar 3.5, dijelaskan tentang interaksi admin dengan sistem kemahasiswaan. Langkah pertama yang harus dilakukan admin adalah melakukan *login* dengan menginput data yang diperlukan. Setelah berhasil *login*, admin dapat mengakses berbagai opsi untuk mengelola data dalam sistem.

Pertama, admin dapat Lihat Daftar Program Studi, yang memungkinkan mereka untuk melihat semua program studi yang terdaftar. Admin juga memiliki kemampuan untuk Tambah Data Program Studi, Edit Data Program Studi, dan Hapus Data Program Studi sesuai kebutuhan. Selanjutnya, admin dapat mengelola Tahun Akademik dengan opsi untuk Lihat Daftar Tahun Akademik, Tambah Data Tahun Akademik, Edit Data Tahun Akademik, dan Hapus Data Tahun Akademik. Ini penting untuk memastikan bahwa informasi akademik selalu diperbarui.

Admin juga dapat mengelola Unit Kegiatan Mahasiswa dengan melihat daftar yang ada, serta menambah, mengedit, atau menghapus data unit kegiatan mahasiswa sesuai kebutuhan. Selain itu, admin dapat mengelola data Dosen dengan opsi untuk Lihat Daftar Dosen, Tambah Data Dosen, Edit Data Dosen, dan Hapus Data Dosen. Ini memastikan bahwa informasi tentang dosen selalu akurat dan terkini. Terakhir, admin dapat mengelola data Mahasiswa dengan opsi untuk Lihat Daftar Mahasiswa, Tambah Data Mahasiswa, Edit Data Mahasiswa, dan Hapus Data Mahasiswa.

3.3.5. Skenario Use Case

1. Skenario Use Case Login (aktor mahasiswa).

Tabel 3. 4 Skenario Use Case Login mahasiswa

Identifikasi	
Kode	KF-01
Nama	<i>Login</i>
Aktor	Mahasiswa
Kondisi Awal	Aktor belum <i>login</i> ke sistem
Kondisi Akhir yang diharapkan	Aktor berhasil masuk ke halaman utama
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Membuka halaman <i>login</i>	
2. Pilih Role Mahasiswa	
3. Memasukkan <i>username</i> dan <i>password</i>	4. Memverifikasi data <i>login</i>
	5. Menampilkan halaman utama
Skenario Alternatif	
1. Memasukkan <i>username</i> dan <i>password</i> salah	2. Menampilkan pesan <i>error</i>

	3. Kembali ke halaman <i>login</i>
--	------------------------------------

Tabel 3.4 diatas menjelaskan mengenai skenario usecase *login* untuk mahasiswa agar dapat masuk ke halaman mahasiswa.

2. Skenario *Use Case Logout* (aktor mahasiswa).

Tabel 3. 5 Skenario *Use Case Logout* Mahasiswa

Identifikasi	
Kode	KF-03
Nama	Lihat Daftar Pengajuan Kredit Aktivitas
Aktor	Mahasiswa
Kondisi Awal	Mahasiswa sudah <i>login</i> ke sistem
Kondisi Akhir yang diharapkan	Mahasiswa dapat melihat daftar pengajuan kredit aktivitas yang telah dilakukan
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Pilih menu "Kredit Aktivitas"	2. Sistem mengambil data daftar pengajuan dari <i>database</i> 3. Sistem menampilkan daftar pengajuan
4. Mahasiswa melihat daftar pengajuan aktivitas	
Skenario Alternatif	
	2. Sistem tidak menemukan data pengajuan
	3. Sistem menampilkan pesan "Belum ada pengajuan kredit aktivitas"

Tabel 3.5 diatas menjelaskan mengenai skenario Skenario *Use Case Logout* untuk aktor mahasiswa

3. Skenario *Use Case Lihat Daftar Pengajuan Kredit Aktivitas* (aktor mahasiswa).

Tabel 3. 6 Skenario *Use Case Lihat Daftar Pengajuan Kredit Aktivitas* untuk aktor mahasiswa

Identifikasi	
Kode	KF-03
Nama	Lihat Daftar Pengajuan Kredit Aktivitas
Aktor	Mahasiswa
Kondisi Awal	Mahasiswa sudah <i>login</i> ke sistem

Kondisi Akhir yang diharapkan	Mahasiswa dapat melihat daftar pengajuan kredit aktivitas yang telah dilakukan
Skenario utama	
Aksi Aktor	Reaksi Sistem
5. Pilih menu "Kredit Aktivitas"	6. Sistem mengambil data daftar pengajuan dari <i>database</i> 7. Sistem menampilkan daftar pengajuan
8. Mahasiswa melihat daftar pengajuan aktivitas	
Skenario Alternatif	
	4. Sistem tidak menemukan data pengajuan
	5. Sistem menampilkan pesan "Belum ada pengajuan kredit aktivitas"

4. Skenario *Use Case* Buat Pengajuan Kredit Aktivitas (aktor mahasiswa).

Tabel 3. 7 Skenario *Use Case* Buat Pengajuan Kredit Aktivitas untuk aktor mahasiswa

Identifikasi	
Kode	KF-04
Nama	Buat Pengajuan Kredit Aktivitas
Aktor	Mahasiswa
Kondisi Awal	Mahasiswa sudah <i>login</i> ke sistem dan berada di halaman pengajuan kredit aktivitas
Kondisi Akhir yang diharapkan	Mahasiswa dapat melihat daftar pengajuan kredit aktivitas yang telah dilakukan
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Pilih menu "Kredit Aktivitas"	2. Sistem mengambil data daftar pengajuan dari <i>database</i> 3. Sistem menampilkan daftar pengajuan
4. Klik tombol buat pengajuan baru	5. Menampilkan form pengajuan baru
6. Mengisi form pengajuan baru	
7. Klik "Simpan"	8. Sistem menyimpan data pengajuan baru 9. Menampilkan daftar pengajuan aktivitas
Skenario Alternatif	

7. Klik tombol "Batal"	8. Sistem tidak menemukan data pengajuan
	9. Sistem menampilkan pesan "Belum ada pengajuan kredit aktivitas"

5. Skenario *Use Case* Membatalkan Pengajuan Kredit Aktivitas (aktor mahasiswa).

Tabel 3. 8 Skenario *Use Case* Membatalkan Pengajuan Kredit Aktivitas untuk aktor mahasiswa

Identifikasi	
Kode	KF-05
Nama	Membatalkan Pengajuan Kredit Aktivitas
Aktor	Mahasiswa
Kondisi Awal	Mahasiswa sudah <i>login</i> ke sistem dan memiliki pengajuan kredit aktivitas yang masih dalam status menunggu atau dalam proses
Kondisi Akhir yang diharapkan	Pengajuan kredit aktivitas dibatalkan, dan sistem memperbarui status pengajuan menjadi dibatalkan
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Mahasiswa mengakses halaman daftar pengajuan kredit aktivitas	2. Sistem menampilkan daftar pengajuan dengan status masing-masing
3. Mahasiswa memilih pengajuan dengan status menunggu	4. Sistem menampilkan detail pengajuan
5. Mahasiswa mengklik tombol "Batalkan Pengajuan"	6. Sistem menampilkan konfirmasi pembatalan
7. Mahasiswa mengonfirmasi dengan memilih "Ya"	8. Sistem mengubah status pengajuan menjadi dibatalkan
	9. Sistem menampilkan notifikasi bahwa pengajuan telah berhasil dibatalkan
Skenario Alternatif	
1. Mahasiswa memilih pengajuan dengan status disetujui atau ditolak	2. Sistem menampilkan pesan bahwa pengajuan yang telah disetujui atau ditolak tidak dapat dibatalkan

6. Skenario *Use Case* Lihat Daftar Pengajuan Proposal (aktor mahasiswa).

Tabel 3. 9 Skenario *Use Case* Lihat Daftar Pengajuan Proposal untuk aktor mahasiswa

Identifikasi	
Kode	KF-06
Nama	Lihat Daftar Pengajuan Proposal
Aktor	Mahasiswa
Kondisi Awal	Mahasiswa sudah <i>login</i> ke sistem dan berada di dashboard atau halaman utama
Kondisi Akhir yang diharapkan	Mahasiswa dapat melihat daftar pengajuan proposal beserta statusnya
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Mahasiswa mengakses halaman "Daftar Pengajuan Proposal"	2. Sistem menampilkan daftar proposal yang telah diajukan oleh mahasiswa
3. Mahasiswa dapat melihat detail dari setiap pengajuan proposal (termasuk status seperti menunggu, disetujui, atau ditolak)	4. Sistem menampilkan detail pengajuan proposal yang dipilih
Skenario Alternatif	
1. Mahasiswa mengakses halaman "Daftar Pengajuan Proposal"	2. Sistem menampilkan pesan bahwa belum ada pengajuan proposal yang dibuat

7. Skenario *Use Case* Buat Pengajuan Proposal (aktor mahasiswa).

Tabel 3. 10 Skenario *Use Case* Buat Pengajuan Proposal untuk aktor mahasiswa

Identifikasi	
Kode	KF-07
Nama	Buat Pengajuan Proposal
Aktor	Mahasiswa
Kondisi Awal	Mahasiswa sudah <i>login</i> ke sistem dan berada di halaman pengajuan proposal
Kondisi Akhir yang diharapkan	Pengajuan proposal berhasil dibuat dan tersimpan dalam sistem dengan status menunggu
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Mahasiswa mengakses halaman "Buat Pengajuan Proposal"	2. Sistem menampilkan formulir pengajuan proposal

3. Mahasiswa mengisi formulir pengajuan dengan data yang diperlukan (Judul proposal, jenis proposal, berkas pendukung.)	4. Sistem memvalidasi input yang dimasukkan
5. Mahasiswa mengklik tombol "Kirim Pengajuan"	6. Sistem menyimpan pengajuan ke <i>database</i> dengan status menunggu 7. Sistem menampilkan notifikasi bahwa pengajuan berhasil dikirim
Skenario Alternatif	
1. Mahasiswa mencoba mengklik tombol "Kirim Pengajuan" tanpa melengkapi semua data	2. Sistem menampilkan pesan <i>error</i> bahwa ada data yang belum diisi

8. Skenario *Use Case* Membatalkan Pengajuan Proposal (aktor mahasiswa).

Tabel 3. 11 Skenario *Use Case* Membatalkan Pengajuan Proposal untuk aktor mahasiswa

Identifikasi	
Kode	KF-08
Nama	Mmembatalkan Pengajuan Proposal
Aktor	Mahasiswa
Kondisi Awal	Mahasiswa sudah <i>login</i> ke sistem dan memiliki proposal dengan status menunggu yang dapat dibatalkan
Kondisi Akhir yang diharapkan	Proposal berhasil dibatalkan dan tidak lagi diproses
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Mahasiswa mengakses halaman "Daftar Pengajuan Proposal"	2. Sistem menampilkan daftar proposal yang telah diajukan
3. Mahasiswa memilih proposal yang ingin dibatalkan	4. Sistem menampilkan detail proposal
5. Mahasiswa mengklik tombol "Batalkan Pengajuan"	6. Sistem menampilkan konfirmasi pembatalan
7. Mahasiswa memilih "Ya" untuk konfirmasi pembatalan	8. Sistem mengubah status proposal menjadi dibatalkan dan menyimpan perubahan ke <i>database</i> 9. Sistem menampilkan notifikasi bahwa pengajuan berhasil dibatalkan
Skenario Alternatif	
1. Mahasiswa mengklik tombol "Batalkan Pengajuan"	2. Sistem menampilkan konfirmasi pembatalan

3. Mahasiswa memilih "Batal"	4. Sistem membatalkan proses pembatalan dan tetap menampilkan detail proposal
------------------------------	---

9. Skenario *Use Case* Lihat Daftar *Tracer Alumni* (aktor mahasiswa).

Tabel 3. 12 Skenario *Use Case* Lihat Daftar *Tracer Alumni* untuk aktor mahasiswa

Identifikasi	
Kode	KF-09
Nama	Lihat Daftar <i>Tracer Alumni</i>
Aktor	Mahasiswa
Kondisi Awal	Aktor sudah <i>login</i> ke sistem dan berada di halaman <i>tracer alumni</i>
Kondisi Akhir yang diharapkan	Aktor dapat melihat daftar <i>tracer alumni</i> yang tersedia dalam sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Aktor mengakses halaman " <i>Tracer Alumni</i> "	2. Sistem menampilkan daftar <i>tracer alumni</i> yang tersedia
Skenario Alternatif	
1. Aktor mengakses halaman " <i>Tracer Alumni</i> "	2. Sistem menampilkan pesan bahwa tidak ada data <i>tracer alumni</i> yang tersedia

10. Skenario *Use Case* Kirim Bukti Pengisian *Tracer Alumni* (aktor mahasiswa).

Tabel 3. 13 Skenario *Use Case* Kirim Bukti Pengisian *Tracer Alumni* untuk aktor mahasiswa

Identifikasi	
Kode	KF-10
Nama	Kirim Bukti Pengisian <i>Tracer Alumni</i>
Aktor	Mahasiswa
Kondisi Awal	Alumni telah mengisi <i>tracer alumni</i> dan memiliki bukti pengisian dalam bentuk file atau screenshot
Kondisi Akhir yang diharapkan	Bukti pengisian berhasil dikirim dan tersimpan dalam sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Alumni masuk ke halaman "Kirim Bukti <i>Tracer Alumni</i> "	2. Sistem menampilkan formulir unggah bukti pengisian

3. Alumni mengunggah file bukti pengisian berupa gambar	4. Sistem memvalidasi format dan ukuran file
5. Alumni mengklik tombol "Kirim"	6. Sistem menyimpan bukti pengisian ke <i>database</i>
	7. Sistem menampilkan notifikasi bahwa bukti pengisian berhasil dikirim
Skenario Alternatif	
1. Alumni mengunggah file dengan format atau ukuran yang tidak sesuai	2. Sistem menampilkan pesan <i>error</i> bahwa file tidak memenuhi syarat dan meminta alumni mengunggah ulang

11. Skenario *Use Case* Mengubah Bukti Pengisian *Tracer* Alumni (aktor mahasiswa).

Tabel 3. 14 Skenario *Use Case* Mengubah Bukti Pengisian *Tracer* Alumni untuk aktor mahasiswa

Identifikasi	
Kode	KF-11
Nama	Mengubah Bukti Pengisian <i>Tracer</i> Alumni
Aktor	Mahasiswa
Kondisi Awal	Alumni telah mengisi <i>tracer</i> alumni dan memiliki bukti pengisian dalam bentuk file atau screenshot
Kondisi Akhir yang diharapkan	Perubahan bukti pengisian berhasil dikirim dan tersimpan dalam sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Membuka halaman <i>tracer</i> alumni	2. Menampilkan data <i>tracer</i> alumni yang sudah diisi sebelumnya
3. Memilih opsi "Ubah" pada bukti pengisian yang ingin diubah	4. Menampilkan input file untuk mengubah bukti pengisian <i>tracer</i> alumni
5. pilih file bukti pengisian berupa gambar	6. Menyimpan perubahan yang dilakukan oleh alumni
7. Menekan tombol "Simpan"	8. Menampilkan pesan konfirmasi bahwa bukti pengisian <i>tracer</i> alumni berhasil diperbarui
Skenario Alternatif	
1. Alumni mengunggah file dengan format atau ukuran yang tidak sesuai	2. Sistem menampilkan pesan <i>error</i> bahwa file tidak memenuhi syarat dan meminta alumni mengunggah ulang

12. Skenario *Use Case* Membatalkan Bukti Pengisian *Tracer* Alumni (aktor mahasiswa).

Tabel 3. 15 Skenario *Use Case* Membatalkan Bukti Pengisian *Tracer* Alumni untuk aktor mahasiswa

Identifikasi	
Kode	KF-12
Nama	Membatalkan Bukti Pengisian <i>Tracer</i> Alumni
Aktor	Mahasiswa
Kondisi Awal	Alumni telah mengisi <i>tracer</i> alumni dan memiliki bukti pengisian dalam bentuk file atau screenshot
Kondisi Akhir yang diharapkan	Bukti pengisian berhasil dibatalkan dan bukti file bukti dihapus dalam sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Membuka halaman <i>tracer</i> alumni	2. Menampilkan data <i>tracer</i> alumni yang sudah diisi sebelumnya
3. Memilih opsi "batalkan" pada bukti pengisian yang ingin diubah	4. Menampilkan konfirmasi membatalkan bukti pengisian <i>tracer</i> alumni
5. Pilih opsi "Ya"	6. Menampilkan input file karena bukti sebelumnya sudah dihapus
Skenario Alternatif	
5. Pilih "Tidak"	6. Menutup tampilan konfirmasi membatalkan bukti pengisian <i>tracer</i> alumni

13. Skenario *Use Case Login* (aktor dosen).

Tabel 3. 16 Skenario *Use Case Login* untuk aktor dosen

Identifikasi	
Kode	KF-13
Nama	<i>Login</i>
Aktor	Dosen
Kondisi Awal	Aktor belum <i>login</i> ke sistem
Kondisi Akhir yang diharapkan	Aktor berhasil masuk ke halaman utama
Skenario utama	
Aksi Aktor	Reaksi Sistem

1. Membuka halaman <i>login</i>	
2. Pilih Role Dosen	
3. Memasukkan <i>username</i> dan <i>password</i>	4. Memverifikasi data <i>login</i>
	5. Menampilkan halaman utama
Skenario Alternatif	
1. Memasukkan <i>username</i> dan <i>password</i> salah	2. Menampilkan pesan <i>error</i>
	3. Kembali ke halaman <i>login</i>

14. Skenario *Use Case Logout* (aktor dosen).

Tabel 3. 17 Skenario *Use Case Logout* untuk aktor dosen

Identifikasi	
Kode	KF-14
Nama	<i>Logout</i>
Aktor	Dosen
Kondisi Awal	<i>Logout</i>
Kondisi Akhir yang diharapkan	Aktor berhasil <i>logout</i>
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Klik tombol <i>logout</i>	2. Menampilkan konfirmasi <i>logout</i>
3. Pilih opsi "Ya"	4. Menampilkan halaman <i>login</i>
	5. Menampilkan halaman utama
Skenario Alternatif	
3. Pilih opsi "Batal"	4. Sistem membatalkan proses <i>logout</i> , pengguna tetap <i>login</i>

15. Skenario *Use Case Lihat Detail Bukti Aktivitas Mahasiswa* (aktor dosen).

Tabel 3. 18 Skenario *Use Case Lihat Detail Bukti Aktivitas Mahasiswa* untuk aktor dosen

Identifikasi	
Kode	KF-15
Nama	Lihat Detail Bukti Aktivitas Mahasiswa
Aktor	Dosen

Kondisi Awal	Dosen sudah <i>login</i> ke sistem
Kondisi Akhir yang diharapkan	Dosen dapat melihat bukti dari daftar pengajuan kredit aktivitas yang telah dilakukan
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Pilih menu "Kredit Aktivitas"	2. Sistem mengambil data daftar pengajuan dari <i>database</i> 3. Sistem menampilkan daftar pengajuan
4. Dosen melihat daftar pengajuan aktivitas	
5. Klik lihat file dari daftar pengajuan aktivitas	6. Sistem menampilkan file bukti dari daftar pengajuan aktivitas yang dipilih

16. Skenario *Use Case* Edit Status Verifikasi Pengajuan Proposal (aktor dosen).

Tabel 3. 19 Skenario *Use Case* Edit Status Verifikasi Pengajuan Proposal untuk aktor dosen

Identifikasi	
Kode	KF-16
Nama	Edit Status Verifikasi Pengajuan Proposal
Aktor	Dosen
Kondisi Awal	Dosen sudah <i>login</i> ke sistem
Kondisi Akhir yang diharapkan	Dosen mengubah status pengajuan dari daftar pengajuan kredit aktivitas yang telah dilakukan
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Pilih menu "Kredit Aktivitas"	2. Sistem mengambil data daftar pengajuan dari <i>database</i> 3. Sistem menampilkan daftar pengajuan
4. Dosen melihat daftar pengajuan aktivitas	
5. Klik status dari daftar pengajuan yang dipilih	
6. Pilih opsi status yang diinginkan	7. Update status pengajuan disimpan di <i>database</i>

17. Skenario *Use Case Login* (aktor kemahasiswaan).

Tabel 3. 20 Skenario *Use Case Login* untuk aktor kemahasiswaan

Identifikasi	
Kode	KF-17
Nama	<i>Login</i>
Aktor	Kemahasiswaan
Kondisi Awal	Aktor belum <i>login</i> ke sistem
Kondisi Akhir yang diharapkan	Aktor berhasil masuk ke halaman utama
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Membuka halaman <i>login</i>	
2. Pilih Role Kemahasiswaan	
3. Memasukkan <i>username</i> dan <i>password</i>	4. Memverifikasi data <i>login</i> 5. Menampilkan halaman utama
Skenario Alternatif	
3. Memasukkan <i>username</i> dan <i>password</i> salah	4. Menampilkan pesan <i>error</i> 3. Kembali ke halaman <i>login</i>

18. Skenario *Use Case Logout* (aktor kemahasiswaan).

Tabel 3. 21 Skenario *Use Case Logout* untuk aktor kemahasiswaan

Identifikasi	
Kode	KF-18
Nama	<i>Logout</i>
Aktor	Kemahasiswaan
Kondisi Awal	<i>Logout</i>
Kondisi Akhir yang diharapkan	Aktor berhasil <i>logout</i>
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Klik tombol <i>logout</i>	2. Menampilkan konfirmasi <i>logout</i>
3. Pilih opsi "Ya"	4. Menampilkan halaman <i>login</i> 5. Menampilkan halaman utama
Skenario Alternatif	

3. Pilih opsi "Batal"	4. Sistem membatalkan proses <i>logout</i> , pengguna tetap <i>login</i>
-----------------------	--

19. Skenario *Use Case* Lihat Daftar Kategori Aktivitas (aktor kemahasiswaan).

Tabel 3. 22 Skenario *Use Case* Lihat Daftar Kategori Aktivitas untuk aktor kemahasiswaan

Identifikasi	
Kode	KF-19
Nama	Lihat Daftar Kategori Aktivitas
Aktor	Aktor telah <i>login</i> dan mengakses halaman kategori aktivitas.
Kondisi Awal	Aktor telah <i>login</i> ke sistem
Kondisi Akhir yang diharapkan	Daftar kategori aktivitas ditampilkan dengan informasi terkait.
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Membuka halaman kategori aktivitas	2. Menampilkan daftar kategori aktivitas
Skenario Alternatif	
1. Tidak ada kategori aktivitas yang tersedia	2. Menampilkan pesan "Tidak ada kategori aktivitas yang tersedia"

20. Skenario *Use Case* Tambah Data Kategori Aktivitas (aktor kemahasiswaan).

Tabel 3. 23 Skenario *Use Case* Tambah Data Kategori Aktivitas untuk aktor kemahasiswaan

Identifikasi	
Kode	KF-20
Nama	Tambah Data Kategori Aktivitas
Aktor	Kemahasiswaan
Kondisi Awal	Aktor telah <i>login</i> dan mengakses halaman kategori aktivitas.
Kondisi Akhir yang diharapkan	Data kategori aktivitas berhasil ditambah.
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Klik tombol tambah kategori aktivitas	2. Menampilkan form untuk tambah data kategori aktivitas
3. Isi form tambah data kategori aktivitas (nama dan poin)	4. Menyimpan data kategori aktivitas

21. Skenario *Use Case* Hapus Data Kategori Aktivitas (aktor kemahasiswaan).

Tabel 3. 24 Skenario *Use Case* Hapus Data Kategori Aktivitas untuk aktor kemahasiswaan

Identifikasi	
Kode	KF-21
Nama	Hapus Data Kategori Aktivitas
Aktor	Kemahasiswaan
Kondisi Awal	Aktor telah <i>login</i> dan mengakses halaman kategori aktivitas.
Kondisi Akhir yang diharapkan	Data kategori aktivitas berhasil dihapus.
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Memilih kategori yang akan dihapus	2. Menampilkan konfirmasi penghapusan data kategori aktivitas
3. Konfirmasi penghapusan kategori	4. Menghapus data kategori aktivitas dari sistem
5. Menampilkan daftar kategori aktivitas yang diperbarui	6. Menampilkan daftar kategori aktivitas yang sudah dihapus
Skenario Alternatif	
1. Data yang ingin dihapus sudah tidak ada	2. Menampilkan pesan "Data kategori aktivitas sudah tidak ada"

22. Skenario *Use Case* Edit Data Kategori Aktivitas (aktor kemahasiswaan).

Tabel 3. 25 Skenario *Use Case* Edit Data Kategori Aktivitas untuk aktor kemahasiswaan

Identifikasi	
Kode	KF-22
Nama	Mengubah Data Kategori Aktivitas
Aktor	Kemahasiswaan
Kondisi Awal	Aktor telah <i>login</i> dan mengakses halaman kategori aktivitas.
Kondisi Akhir yang diharapkan	Data kategori aktivitas berhasil diperbarui.
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Memilih kategori yang akan diubah	2. Menampilkan form untuk mengubah data kategori aktivitas
3. Mengubah data kategori aktivitas	4. Menyimpan perubahan data kategori aktivitas
5. Menampilkan konfirmasi perubahan	6. Menampilkan daftar kategori aktivitas yang telah diperbarui

Skenario Alternatif	
1. Data yang diubah tidak valid	2. Menampilkan pesan <i>error</i> "Data tidak valid"

23. Skenario *Use Case* Lihat Daftar Pengajuan Aktivitas Mahasiswa (aktor kemahasiswaan).

Tabel 3. 26 Skenario *Use Case* Lihat Daftar Pengajuan Aktivitas Mahasiswa untuk aktor kemahasiswaan.

Identifikasi	
Kode	KF-23
Nama	Lihat Daftar Pengajuan Aktivitas
Aktor	Kemahasiswaan
Kondisi Awal	Aktor telah <i>login</i> .
Kondisi Akhir yang diharapkan	Daftar pengajuan proposal ditampilkan.
Skenario utama	
Aksi Aktor	Reaksi Sistem
1.Membuka halaman pengajuan aktivitas	2.Menampilkan daftar pengajuan aktivitas
Skenario Alternatif	
1. Tidak ada pengajuan aktivitas yang tersedia	2. Menampilkan pesan "Tidak ada pengajuan aktivitas"

24. Skenario *Use Case* Lihat Detail Bukti Aktivitas Mahasiswa (aktor kemahasiswaan).

Tabel 3. 27 Skenario *Use Case* Lihat Detail Bukti Aktivitas Mahasiswa untuk aktor kemahasiswaan.

Identifikasi	
Kode	KF-24
Nama	Lihat Detail Bukti Aktivitas Mahasiswa
Aktor	Kemahasiswaan
Kondisi Awal	Aktor telah <i>login</i> ke sistem dan mengakses halaman bukti aktivitas mahasiswa.
Kondisi Akhir yang diharapkan	Detail bukti aktivitas mahasiswa ditampilkan.

Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Memilih bukti aktivitas mahasiswa	2. Menampilkan detail bukti aktivitas mahasiswa
Skenario Alternatif	
1. Bukti aktivitas mahasiswa tidak ditemukan	2. Menampilkan pesan "Bukti aktivitas mahasiswa tidak ditemukan"

25. Skenario *Use Case* Edit Status Verifikasi Pengajuan Aktivitas (aktor kemahasiswaan).

Tabel 3. 28 Skenario *Use Case* Edit Status Verifikasi Pengajuan Aktivitas untuk aktor kemahasiswaan.

Identifikasi	
Kode	KF-25
Nama	Edit Status Verifikasi Pengajuan Proposal
Aktor	Kemahasiswaan
Kondisi Awal	Aktor telah <i>login</i> ke sistem dan mengakses halaman bukti aktivitas mahasiswa.
Kondisi Akhir yang diharapkan	Status verifikasi pengajuan proposal berhasil diperbarui.
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Memilih proposal yang akan diverifikasi	2. Menampilkan form untuk mengubah status verifikasi proposal
3. Mengubah status verifikasi proposal	4. Menyimpan perubahan status verifikasi
5. Menampilkan daftar proposal dengan status yang diperbarui	6. Menampilkan daftar pengajuan proposal yang diperbarui
Skenario Alternatif	
1. Status verifikasi tidak valid	2. Menampilkan pesan <i>error</i> "Status verifikasi tidak valid"

26. Skenario *Use Case* Lihat Daftar Pengajuan Proposal (aktor kemahasiswaan).

Tabel 3. 29 Skenario *Use Case* Lihat Daftar Pengajuan Proposal untuk aktor kemahasiswaan

Identifikasi	
Kode	KF-26
Nama	Lihat Daftar Pengajuan Proposal
Aktor	Kemahasiswaan

Kondisi Awal	Aktor telah <i>login</i> ke sistem
Kondisi Akhir yang diharapkan	Daftar pengajuan proposal ditampilkan.
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Membuka halaman pengajuan proposal	2. Menampilkan daftar pengajuan proposal
Skenario Alternatif	
1. Tidak ada pengajuan proposal yang tersedia	2. Menampilkan pesan "Tidak ada pengajuan proposal"

27. Skenario *Use Case* Lihat Detail Bukti pengajuan Proposal (aktor kemahasiswaan).

Tabel 3. 30 Skenario *Use Case* Lihat Detail Bukti pengajuan Proposal untuk aktor kemahasiswaan

Identifikasi	
Kode	KF-27
Nama	Lihat Detail Bukti Aktivitas Mahasiswa
Aktor	Kemahasiswaan
Kondisi Awal	Aktor telah <i>login</i> ke sistem dan mengakses halaman bukti aktivitas mahasiswa.
Kondisi Akhir yang diharapkan	Detail bukti aktivitas mahasiswa ditampilkan.
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Memilih bukti aktivitas mahasiswa	2. Menampilkan detail bukti aktivitas mahasiswa
Skenario Alternatif	
1. Bukti aktivitas mahasiswa tidak ditemukan	2. Menampilkan pesan "Bukti aktivitas mahasiswa tidak ditemukan"

28. Skenario *Use Case* Edit Status Verifikasi Pengajuan Proposal (aktor kemahasiswaan).

Tabel 3. 31 Skenario *Use Case* Edit Status Verifikasi Pengajuan Proposal untuk aktor kemahasiswaan

Identifikasi	
Kode	KF-28
Nama	Edit Status Verifikasi Pengajuan Proposal
Aktor	Kemahasiswaan

Kondisi Awal	Aktor telah <i>login</i> dan mengakses halaman pengajuan proposal.
Kondisi Akhir yang diharapkan	Status verifikasi pengajuan proposal berhasil diperbarui.
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Memilih proposal yang akan diverifikasi	2. Menampilkan form untuk mengubah status verifikasi proposal
3. Mengubah status verifikasi proposal	4. Menyimpan perubahan status verifikasi
5. Menampilkan daftar proposal dengan status yang diperbarui	6. Menampilkan daftar pengajuan proposal yang diperbarui
Skenario Alternatif	
1. Status verifikasi tidak valid	2. Menampilkan pesan <i>error</i> "Status verifikasi tidak valid"

29. Skenario *Use Case* Lihat Daftar *Tracer Alumni* (aktor kemahasiswaan).

Tabel 3. 32 Skenario *Use Case* Lihat Daftar *Tracer Alumni* untuk aktor kemahasiswaan

Identifikasi	
Kode	KF-29
Nama	Lihat Daftar <i>Tracer Alumni</i>
Aktor	Kemahasiswaan
Kondisi Awal	Aktor telah <i>login</i> ke sistem.
Kondisi Akhir yang diharapkan	Daftar <i>tracer alumni</i> ditampilkan.
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Membuka halaman <i>tracer alumni</i>	2. Menampilkan daftar <i>tracer alumni</i>
Skenario Alternatif	
1. Tidak ada <i>tracer alumni</i> yang tersedia	2. Menampilkan pesan "Tidak ada <i>tracer alumni</i> "

30. Skenario *Use Case* Tambah Data *Tracer Alumni* (aktor kemahasiswaan).

Tabel 3. 33 Skenario *Use Case* Tambah Data *Tracer Alumni* untuk aktor kemahasiswaan

Identifikasi	
Kode	KF-30
Nama	Tambah Data <i>Tracer Alumni</i>
Aktor	Kemahasiswaan
Kondisi Awal	Aktor telah <i>login</i> ke sistem dan mengakses halaman <i>tracer alumni</i> .
Kondisi Akhir yang diharapkan	Data <i>tracer alumni</i> berhasil ditambahkan.
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Memilih opsi untuk menambah <i>tracer alumni</i>	2. Menampilkan form untuk menambah data <i>tracer alumni</i>
3. Mengisi form data <i>tracer alumni</i>	4. Menyimpan data <i>tracer alumni</i> yang baru
5. Menampilkan daftar <i>tracer alumni</i> yang diperbarui	6. Menampilkan daftar <i>tracer alumni</i> yang baru ditambahkan
Skenario Alternatif	
1. Data yang dimasukkan tidak valid	2. Menampilkan pesan <i>error</i> "Data tidak valid"

31. Skenario *Use Case* Edit Data *Tracer Alumni* (aktor kemahasiswaan).

Tabel 3. 34 Skenario *Use Case* Edit Data *Tracer Alumni* untuk aktor kemahasiswaan

Identifikasi	
Kode	KF-31
Nama	Edit Data <i>Tracer Alumni</i>
Aktor	Kemahasiswaan
Kondisi Awal	Aktor telah <i>login</i> ke sistem dan mengakses halaman daftar <i>tracer alumni</i> .
Kondisi Akhir yang diharapkan	Data <i>tracer alumni</i> berhasil diperbarui.
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Memilih <i>tracer alumni</i> yang akan diubah	2. Menampilkan form untuk mengubah data <i>tracer alumni</i>
3. Mengubah data <i>tracer alumni</i>	4. Menyimpan perubahan data <i>tracer alumni</i>
	5. Menampilkan daftar <i>tracer alumni</i> yang diperbarui

Skenario Alternatif	
1. Data yang dimasukkan tidak valid	2. Menampilkan pesan <i>error</i> "Data tidak valid"
3. Tidak ada <i>tracer</i> alumni yang dipilih	4. Menampilkan pesan <i>error</i> "Tracer alumni tidak ditemukan"

32. Skenario *Use Case* Hapus Data *Tracer* Alumni (aktor kemahasiswaan).

Tabel 3. 35 Skenario *Use Case* Hapus Data *Tracer* Alumni untuk aktor kemahasiswaan

Identifikasi	
Kode	KF-32
Nama	Edit Data <i>Tracer</i> Alumni
Aktor	Kemahasiswaan
Kondisi Awal	Aktor telah <i>login</i> ke sistem dan mengakses halaman daftar <i>tracer</i> alumni.
Kondisi Akhir yang diharapkan	Data <i>tracer</i> alumni berhasil dihapus.
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Memilih <i>tracer</i> alumni yang akan dihapus	2. Menampilkan konfirmasi penghapusan data <i>tracer</i> alumni
3. Mengonfirmasi penghapusan data	4. Menghapus data <i>tracer</i> alumni dari sistem
	5. Menampilkan daftar <i>tracer</i> alumni yang telah diperbarui
Skenario Alternatif	
1. Data yang dimasukkan tidak valid	2. Menampilkan pesan <i>error</i> "Data tidak valid"
3. Tidak ada <i>tracer</i> alumni yang dipilih	4. Menampilkan pesan <i>error</i> "Tracer alumni tidak ditemukan"

33. Skenario *Use Case* Login (aktor admin).

Tabel 3. 36 Skenario *Use Case* Login untuk aktor admin

Identifikasi	
Kode	KF-33
Nama	<i>Login</i>
Aktor	Admin
Kondisi Awal	Aktor belum <i>login</i> ke sistem
Kondisi Akhir yang diharapkan	Aktor berhasil masuk ke halaman utama

Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Membuka halaman <i>login</i>	
2. Pilih Role Mahasiswa	
3. Memasukkan <i>username</i> dan <i>password</i>	
	4. Memverifikasi data <i>login</i>
	5. Menampilkan halaman utama
Skenario Alternatif	
3. Memasukkan <i>username</i> dan <i>password</i> salah	
	4. Menampilkan pesan <i>error</i>
	5. Kembali ke halaman <i>login</i>

34. Skenario *Use Case Logout* (aktor admin).

Tabel 3. 37 Skenario *Use Case Logout* untuk aktor admin

Identifikasi	
Kode	KF-34
Nama	<i>Logout</i>
Aktor	Admin
Kondisi Awal	<i>Logout</i>
Kondisi Akhir yang diharapkan	Aktor berhasil <i>logout</i>
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Klik tombol <i>logout</i>	2. Menampilkan konfirmasi <i>logout</i>
3. Pilih opsi "Ya"	4. Menampilkan halaman <i>login</i>
	5. Menampilkan halaman utama
Skenario Alternatif	
3. Pilih opsi "Batal"	
	4. Sistem membatalkan proses <i>logout</i> , pengguna tetap <i>login</i>

35. Skenario *Use Case Lihat Daftar Program Studi* (aktor admin).

Tabel 3. 38 Skenario *Use Case Lihat Daftar Program Studi* untuk aktor admin

Identifikasi	
Kode	KF-35

Nama	Lihat Daftar Program Studi
Aktor	Admin
Kondisi Awal	Pengguna belum melihat daftar program studi
Kondisi Akhir yang Diharapkan	Pengguna dapat melihat daftar program studi yang tersedia
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Pengguna membuka halaman daftar program studi	2. Sistem menampilkan daftar program studi yang tersedia
Skenario Alternatif	
1. Pengguna melakukan pencarian atau <i>filter</i> berdasarkan jenjang/fakultas/akreditasi	2. Sistem menampilkan hasil pencarian atau <i>filter</i> sesuai dengan kriteria yang dimasukkan pengguna
3. Pengguna membuka halaman tetapi tidak ada data program studi yang tersedia	4. Sistem menampilkan pesan "Data program studi belum tersedia"

36. Skenario *Use Case* Tambah Data Daftar Program Studi (aktor admin).

Tabel 3. 39 Skenario *Use Case* Tambah Data Daftar Program Studi untuk aktor admin

Identifikasi	
Kode	KF-36
Nama	Tambah Data Daftar Program Studi
Aktor	Admin
Kondisi Awal	Data program studi belum ditambahkan ke sistem
Kondisi Akhir yang diharapkan	Data program studi berhasil ditambahkan ke sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin klik tombol data tambah program studi	2. Sistem menampilkan form tambah program studi
3. Admin mengisi data program studi (kode dan nama)	4. Sistem memvalidasi data yang dimasukkan
5. Admin mengklik tombol "Simpan"	6. Sistem menyimpan data program studi ke dalam <i>database</i>
	7. Sistem menampilkan pesan sukses dan daftar program studi yang telah diperbarui
Skenario Alternatif	

1. Admin mengisi form tetapi ada data yang tidak valid/kosong	2. Sistem membatalkan proses <i>logout</i> , pengguna tetap <i>login</i>
3. Admin membatalkan proses dengan mengklik tombol "Batal"	4. Sistem membatalkan proses penambahan dan kembali ke halaman daftar program studi

37. Skenario *Use Case* Edit Data Program Studi (aktor admin).

Tabel 3. 40 Skenario *Use Case* Edit Data Program Studi untuk aktor admin

Identifikasi	
Kode	KF-37
Nama	Edit Data Daftar Program Studi
Aktor	Admin
Kondisi Awal	Data program studi telah tersimpan di sistem
Kondisi Akhir yang diharapkan	Data program studi berhasil diperbarui di sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman daftar program studi	2. Sistem menampilkan daftar program studi yang tersedia
3. Admin memilih program studi yang akan diedit	4. Sistem menampilkan form edit dengan data program studi yang sudah ada
5. Admin mengubah data yang diperlukan (kode dan nama)	6. Sistem memvalidasi data yang dimasukkan
7. Admin mengklik tombol "Simpan"	8. Sistem memperbarui data program studi di <i>database</i>
	9. Sistem menampilkan pesan sukses dan daftar program studi yang telah diperbarui
Skenario Alternatif	
5. Admin mengisi form tetapi ada data yang tidak valid/kosong	6. Sistem menampilkan pesan <i>error</i> dan meminta admin mengisi data dengan benar
7. Admin membatalkan proses dengan mengklik tombol "Batal"	8. Sistem membatalkan perubahan dan kembali ke halaman daftar program studi tanpa menyimpan perubahan

38. Skenario *Use Case* Hapus Data Program Studi (aktor admin).

Tabel 3. 41 Skenario *Use Case* Hapus Data Program Studi untuk aktor admin

Identifikasi	
Kode	KF-38
Nama	Hapus Data Program Studi
Aktor	Admin
Kondisi Awal	Data program studi tersedia di sistem
Kondisi Akhir yang diharapkan	Data program studi berhasil dihapus dari sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman daftar program studi	2. Sistem menampilkan daftar program studi yang tersedia
3. Admin memilih program studi yang akan dihapus	4. Sistem menampilkan konfirmasi penghapusan
5. Admin mengonfirmasi dengan memilih opsi "Ya"	6. Sistem menghapus data program studi dari <i>database</i>
	7. Sistem menampilkan pesan sukses dan memperbarui daftar program studi
Skenario Alternatif	
5. Admin memilih opsi "Batal" pada konfirmasi	6. Sistem membatalkan proses penghapusan, data tetap ada di sistem
7. Sistem gagal menghapus data karena kendala teknis	8. Sistem menampilkan pesan <i>error</i> dan meminta admin mencoba lagi

39. Skenario *Use Case* Lihat Daftar Tahun Akademik (aktor admin).

Tabel 3. 42 Skenario *Use Case* Lihat Daftar Tahun Akademik untuk aktor admin

Identifikasi	
Kode	KF-39
Nama	Lihat Daftar Tahun Akademik
Aktor	Admin
Kondisi Awal	Pengguna belum melihat daftar tahun akademik
Kondisi Akhir yang Diharapkan	Pengguna dapat melihat daftar tahun akademik yang tersedia
Skenario utama	
Aksi Aktor	Reaksi Sistem

1. Admin membuka halaman daftar tahun akademik	2. Sistem menampilkan daftar tahun akademik yang tersedia
3. Admin dapat melakukan pencarian berdasarkan tahun atau status	4. Sistem menampilkan hasil pencarian sesuai kriteria
5. Admin memilih salah satu tahun akademik untuk melihat detailnya	6. Sistem menampilkan informasi detail tahun akademik
Skenario Alternatif	
3. Admin memasukkan kata kunci yang tidak ditemukan	4. Sistem menampilkan pesan "Data tidak ditemukan"
5. Admin kembali ke daftar tahun akademik tanpa memilih detail	6. Sistem kembali menampilkan daftar tahun akademik

40. Skenario *Use Case* Tambah Data Tahun Akademik (aktor admin).

Tabel 3. 43 Skenario *Use Case* Tambah Data Tahun Akademik untuk aktor admin

Identifikasi	
Kode	KF-40
Nama	Tambah Data Tahun Akademik
Aktor	Admin
Kondisi Awal	Data program studi tersedia di sistem Data tahun akademik belum ditambahkan ke sistem
Kondisi Akhir yang diharapkan	Data tahun akademik berhasil ditambahkan ke sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman manajemen tahun akademik	2. Sistem menampilkan daftar tahun akademik yang tersedia
3. Admin mengklik tombol "Tambah Tahun Akademik"	4. Sistem menampilkan form tambah tahun akademik
5. Admin mengisi data tahun akademik (misalnya: Tahun, Semester, Status Aktif/Tidak Aktif)	6. Sistem memvalidasi data yang dimasukkan
7. Admin mengklik tombol "Simpan"	8. Sistem menyimpan data tahun akademik ke dalam <i>database</i>
	9. Sistem menampilkan pesan sukses dan memperbarui daftar tahun akademik
Skenario Alternatif	
5. Admin mengisi form tetapi ada data yang tidak valid/kosong	6. Sistem menampilkan pesan <i>error</i> dan meminta admin mengisi data dengan benar

7. Admin membatalkan proses dengan mengklik tombol "Batal"	8. Sistem membatalkan proses penambahan dan kembali ke halaman daftar tahun akademik tanpa menyimpan data
--	---

41. Skenario *Use Case* Edit Data Tahun Akademik (aktor admin).

Tabel 3. 44 Skenario *Use Case* Edit Data Tahun Akademik untuk aktor admin

Identifikasi	
Kode	KF-41
Nama	Edit Data Tahun Akademik
Aktor	Admin
Kondisi Awal	Data tahun akademik telah tersimpan di sistem
Kondisi Akhir yang Diharapkan	Data tahun akademik berhasil diperbarui di sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman daftar tahun akademik	2. Sistem menampilkan daftar tahun akademik yang tersedia
3. Admin memilih tahun akademik yang akan diedit	4. Sistem menampilkan form edit dengan data yang sudah ada
5. Admin mengubah data yang diperlukan (Tahun, Semester, Status)	6. Sistem memvalidasi data yang dimasukkan
7. Admin mengklik tombol "Simpan"	8. Sistem memperbarui data tahun akademik di <i>database</i>
	9. Sistem menampilkan pesan sukses dan memperbarui daftar tahun akademik
Skenario Alternatif	
5. Admin mengisi form tetapi ada data yang tidak valid/kosong	6. Sistem menampilkan pesan <i>error</i> dan meminta admin mengisi data dengan benar
7. Admin membatalkan proses dengan mengklik tombol "Batal"	8. Sistem membatalkan perubahan dan kembali ke halaman daftar tahun akademik tanpa menyimpan perubahan

42. Skenario *Use Case* Hapus Data Tahun Akademik (aktor admin).

Tabel 3. 45 Skenario *Use Case* Hapus Data Tahun Akademik untuk aktor admin

Identifikasi	
Kode	KF-42
Nama	Hapus Data Tahun Akademik
Aktor	Admin
Kondisi Awal	Data tahun akademik tersedia di sistem
Kondisi Akhir yang Diharapkan	Data tahun akademik berhasil dihapus dari sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman daftar tahun akademik	2. Sistem menampilkan daftar tahun akademik yang tersedia
3. Admin memilih tahun akademik yang akan dihapus	4. Sistem menampilkan konfirmasi penghapusan
5. Admin mengonfirmasi dengan memilih opsi "Ya"	6. Sistem menghapus data tahun akademik dari <i>database</i>
	7. Sistem menampilkan pesan sukses dan memperbarui daftar tahun akademik
Skenario Alternatif	
5. Admin memilih opsi "Batal" pada konfirmasi	6. Sistem membatalkan proses penghapusan, data tetap ada di sistem

43. Skenario *Use Case* Lihat Daftar Unit Kegiatan Mahasiswa (aktor admin).

Tabel 3. 46 Skenario *Use Case* Lihat Daftar Unit Kegiatan Mahasiswa untuk aktor admin

Identifikasi	
Kode	KF-43
Nama	Lihat Daftar Unit Kegiatan Mahasiswa
Aktor	Admin
Kondisi Awal	Pengguna belum melihat daftar UKM
Kondisi Akhir yang Diharapkan	Pengguna dapat melihat daftar UKM yang tersedia
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman daftar UKM	2. Sistem menampilkan daftar UKM yang tersedia
3. Admin memilih salah satu UKM untuk melihat detailnya	4. Sistem menampilkan detail UKM seperti nama, deskripsi, dan kontak pengurus

5. Admin mengonfirmasi dengan memilih opsi "Ya"	6. Sistem menghapus data tahun akademik dari <i>database</i>
Skenario Alternatif	
5. Admin memilih opsi "Batal" pada konfirmasi	6. Sistem membatalkan proses penghapusan, data tetap ada di sistem

44. Skenario *Use Case* Tambah Data Unit Kegiatan Mahasiswa (aktor admin).

Tabel 3. 47 Skenario *Use Case* Tambah Data Unit Kegiatan Mahasiswa untuk aktor admin

Identifikasi	
Kode	KF-44
Nama	Tambah Data Unit Kegiatan Mahasiswa
Aktor	Admin
Kondisi Awal	Data UKM belum ditambahkan ke sistem
Kondisi Akhir yang Diharapkan	Data UKM berhasil ditambahkan ke sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman UKM	2. Sistem menampilkan daftar UKM yang tersedia
3. Admin mengklik tombol "Tambah UKM"	4. Sistem menampilkan form tambah UKM
5. Admin mengisi data UKM (Nama, Deskripsi, Kontak Pengurus)	6. Sistem memvalidasi data yang dimasukkan
7. Admin mengklik tombol "Simpan"	8. Sistem menyimpan data UKM ke dalam <i>database</i>
	9. Sistem menampilkan pesan sukses dan memperbarui daftar UKM

45. Skenario *Use Case* Edit Data Unit Kegiatan Mahasiswa (aktor admin).

Tabel 3. 48 Skenario *Use Case* Edit Data Unit Kegiatan Mahasiswa aktor admin

Identifikasi	
Kode	KF-45
Nama	Edit Data Unit Kegiatan Mahasiswa
Aktor	Admin
Kondisi Awal	Data UKM telah tersimpan di sistem
Kondisi Akhir yang Diharapkan	Data UKM berhasil diperbarui di sistem
Skenario utama	

Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman daftar UKM	2. Sistem menampilkan daftar UKM yang tersedia
3. Admin memilih UKM yang akan diedit	4. Sistem menampilkan form edit dengan data UKM yang sudah ada
5. Admin mengubah data yang diperlukan (Nama, Deskripsi, Kontak Pengurus)	6. Sistem memvalidasi data yang dimasukkan
7. Admin mengklik tombol "Simpan"	8. Sistem memperbarui data UKM di <i>database</i>
	9. Sistem menampilkan pesan sukses dan memperbarui daftar UKM

46. Skenario *Use Case* Hapus Data Unit Kegiatan Mahasiswa (aktor admin).

Tabel 3. 49 Skenario *Use Case* Hapus Data Unit Kegiatan Mahasiswa untuk aktor admin

Identifikasi	
Kode	KF-46
Nama	Hapus Data Unit Kegiatan Mahasiswa
Aktor	Admin
Kondisi Awal	Data UKM tersedia di sistem
Kondisi Akhir yang Diharapkan	Data UKM berhasil dihapus dari sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman daftar UKM	2. Sistem menampilkan daftar UKM yang tersedia
3. Admin memilih UKM yang akan dihapus	4. Sistem menampilkan konfirmasi penghapusan
5. Admin mengonfirmasi dengan memilih opsi "Ya"	6. Sistem menghapus data UKM dari <i>database</i>
	7. Sistem menampilkan pesan sukses dan memperbarui daftar UKM
Skenario Alternatif	
5. Admin memilih opsi "Batal" pada konfirmasi	6. Sistem membatalkan proses penghapusan, data tetap ada di sistem
7. Sistem gagal menghapus data karena kendala teknis	8. Sistem menampilkan pesan <i>error</i> dan meminta admin mencoba lagi

47. Skenario *Use Case* Lihat Daftar Dosen (aktor admin).

Tabel 3. 50 Skenario *Use Case* Lihat Daftar Dosen untuk aktor admin

Identifikasi	
Kode	KF-47
Nama	Lihat Daftar Dosen
Aktor	Admin, Mahasiswa
Kondisi Awal	Pengguna belum melihat daftar dosen
Kondisi Akhir yang Diharapkan	Pengguna dapat melihat daftar dosen yang tersedia
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Pengguna membuka halaman daftar dosen	2. Sistem menampilkan daftar dosen yang tersedia
3. Pengguna memilih salah satu dosen untuk melihat detailnya	4. Sistem menampilkan informasi detail dosen

48. Skenario *Use Case* Tambah Data Dosen (aktor admin).

Tabel 3. 51 Skenario *Use Case* Tambah Data Dosen untuk aktor admin

Identifikasi	
Kode	KF-48
Nama	Tambah Data Dosen
Aktor	Admin
Kondisi Awal	Data dosen belum ditambahkan ke sistem
Kondisi Akhir yang Diharapkan	Data dosen berhasil ditambahkan ke sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman manajemen dosen	2. Sistem menampilkan daftar dosen yang tersedia
3. Admin mengklik tombol "Tambah Dosen"	4. Sistem menampilkan form tambah dosen
5. Admin mengisi data dosen (Nama, NIP, Email, Password)	6. Sistem memvalidasi data yang dimasukkan
7. Admin mengklik tombol "Simpan"	8. Sistem menyimpan data dosen ke dalam database
	9. Sistem menampilkan pesan sukses dan memperbarui daftar dosen

49. Skenario *Use Case* Edit Data Dosen (aktor admin).

Tabel 3. 52 Skenario *Use Case* Edit Data Dosen untuk aktor admin

Identifikasi	
Kode	KF-49
Nama	Edit Data Dosen
Aktor	Admin
Kondisi Awal	Data dosen telah tersimpan di sistem
Kondisi Akhir yang Diharapkan	Data dosen berhasil diperbarui di sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman daftar dosen	2. Sistem menampilkan daftar dosen yang tersedia
3. Admin memilih dosen yang akan diedit data dosen	4. Sistem menampilkan form edit dengan data yang sudah ada
5. Admin mengubah data yang diperlukan (Nama, NIP, Email, Password)	6. Sistem memvalidasi data yang dimasukkan
7. Admin mengklik tombol "Simpan"	8. Sistem memperbarui data dosen di database
	9. Sistem menampilkan pesan sukses dan memperbarui daftar dosen

50. Skenario *Use Case* Hapus Data Dosen (aktor admin).

Tabel 3. 53 Skenario *Use Case* Hapus Data Dosen untuk aktor admin

Identifikasi	
Kode	KF-50
Nama	Hapus Data Dosen
Aktor	Admin
Kondisi Awal	Data dosen tersedia di sistem
Kondisi Akhir yang Diharapkan	Data dosen berhasil dihapus dari sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman daftar dosen	2. Sistem menampilkan daftar dosen yang tersedia
3. Admin memilih dosen yang akan dihapus	4. Sistem menampilkan konfirmasi penghapusan
5. Admin mengonfirmasi dengan memilih opsi "Ya"	6. Sistem menghapus data dosen dari database

	7. Sistem menampilkan pesan sukses dan memperbarui daftar dosen
Skenario Alternatif	
5. Admin memilih opsi "Batal" pada konfirmasi	6. Sistem membatalkan proses penghapusan, data tetap ada di sistem

51. Skenario *Use Case Lihat Daftar Mahasiswa* (aktor admin).

Tabel 3. 54 Skenario Use Case Lihat Daftar Mahasiswa untuk aktor admin

Identifikasi	
Kode	KF-51
Nama	Lihat Daftar Mahasiswa
Aktor	Admin
Kondisi Awal	Pengguna belum melihat daftar mahasiswa
Kondisi Akhir yang Diharapkan	Pengguna dapat melihat daftar mahasiswa yang tersedia
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman daftar mahasiswa	2. Sistem menampilkan daftar mahasiswa yang tersedia
3. Admin dapat melakukan pencarian berdasarkan nama, NIM, atau program studi	4. Sistem menampilkan hasil pencarian sesuai kriteria
5. Admin memilih salah satu mahasiswa untuk melihat detailnya	6. Sistem menampilkan informasi detail mahasiswa
Skenario Alternatif	
5. Pengguna kembali ke daftar mahasiswa tanpa memilih detail	6. Sistem kembali menampilkan daftar mahasiswa

52. Skenario *Use Case Tambah Data Mahasiswa* (aktor admin).

Tabel 3. 55 Skenario Use Case Tambah Data Mahasiswa untuk aktor admin

Identifikasi	
Kode	KF-52
Nama	Tambah Data Mahasiswa
Aktor	Admin
Kondisi Awal	Data mahasiswa belum ditambahkan ke sistem

Kondisi Akhir yang Diharapkan	Data mahasiswa berhasil ditambahkan ke sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman manajemen mahasiswa	2. Sistem menampilkan daftar mahasiswa yang tersedia
3. Admin mengklik tombol "Tambah Mahasiswa"	4. Sistem menampilkan form tambah mahasiswa
5. Admin mengisi data mahasiswa (Nama, NIM, Program Studi, Angkatan, Email, Kontak)	6. Sistem memvalidasi data yang dimasukkan
7. Admin mengklik tombol "Simpan"	8. Sistem menyimpan data mahasiswa ke dalam <i>database</i>
	9. Sistem menampilkan pesan sukses dan memperbarui daftar mahasiswa
Skenario Alternatif	
6. Admin mengisi data yang tidak valid (contoh: NIM sudah digunakan)	7. Sistem menampilkan pesan <i>error</i> dan meminta perbaikan data

53. Skenario *Use Case* Edit Data Mahasiswa (aktor admin).

Tabel 3. 56 Skenario *Use Case* Edit Data Mahasiswa untuk aktor admin

Identifikasi	
Kode	KF-53
Nama	Edit Data Mahasiswa
Aktor	Admin
Kondisi Awal	Data mahasiswa telah tersimpan di sistem
Kondisi Akhir yang Diharapkan	Data mahasiswa berhasil diperbarui di sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman daftar mahasiswa	2. Sistem menampilkan daftar mahasiswa yang tersedia
3. Admin memilih mahasiswa yang akan diedit	4. Sistem menampilkan form edit dengan data yang sudah ada
5. Admin mengubah data yang diperlukan (Nama, NIM, Program Studi, Angkatan, Email, Kontak)	6. Sistem memvalidasi data yang dimasukkan
7. Admin mengklik tombol "Simpan"	8. Sistem memperbarui data mahasiswa di <i>database</i>

	9. Sistem menampilkan pesan sukses dan memperbarui daftar mahasiswa
Skenario Alternatif	
6. Admin mengisi data yang tidak valid (contoh: format email salah)	7. Sistem menampilkan pesan <i>error</i> dan meminta perbaikan data

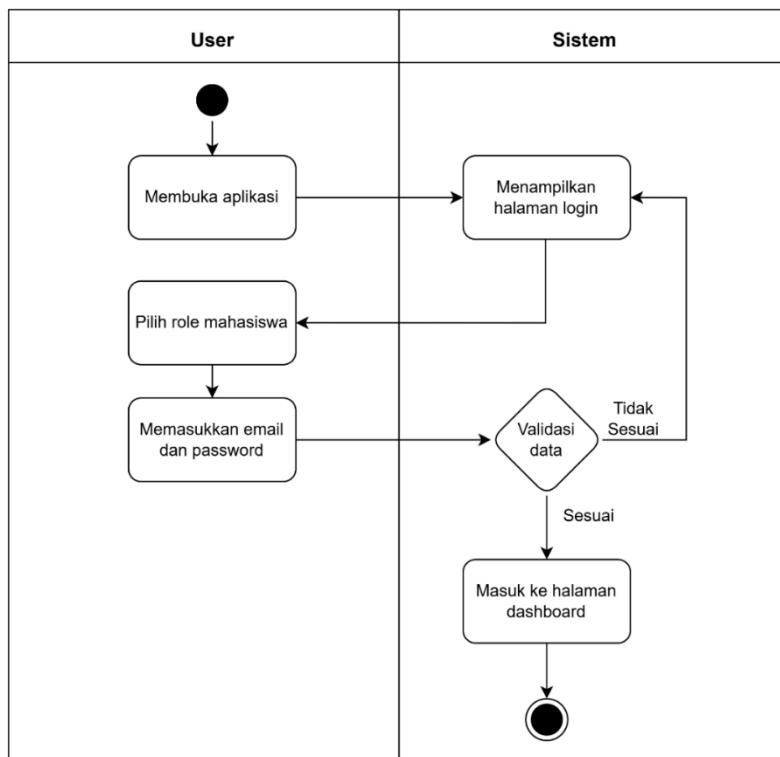
54. Skenario *Use Case* Hapus Data Mahasiswa (aktor admin).

Tabel 3. 57 Skenario *Use Case* Hapus Data Mahasiswa untuk aktor admin

Identifikasi	
Kode	KF-54
Nama	Hapus Data Mahasiswa
Aktor	Admin
Kondisi Awal	Data mahasiswa tersedia di sistem
Kondisi Akhir yang Diharapkan	Data mahasiswa berhasil dihapus dari sistem
Skenario utama	
Aksi Aktor	Reaksi Sistem
1. Admin membuka halaman daftar mahasiswa	2. Sistem menampilkan daftar mahasiswa yang tersedia
3. Admin memilih mahasiswa yang akan dihapus	4. Sistem menampilkan konfirmasi penghapusan
5. Admin mengonfirmasi dengan memilih opsi "Ya"	6. Sistem menghapus data mahasiswa dari <i>database</i>
	7. Sistem menampilkan pesan sukses dan memperbarui daftar mahasiswa
Skenario Alternatif	
5. Admin memilih opsi "Batal" pada konfirmasi	6. Sistem membatalkan proses penghapusan, data tetap ada di sistem
7. Sistem gagal menghapus data karena kendala teknis	8. Sistem menampilkan pesan <i>error</i> dan meminta admin mencoba lagi

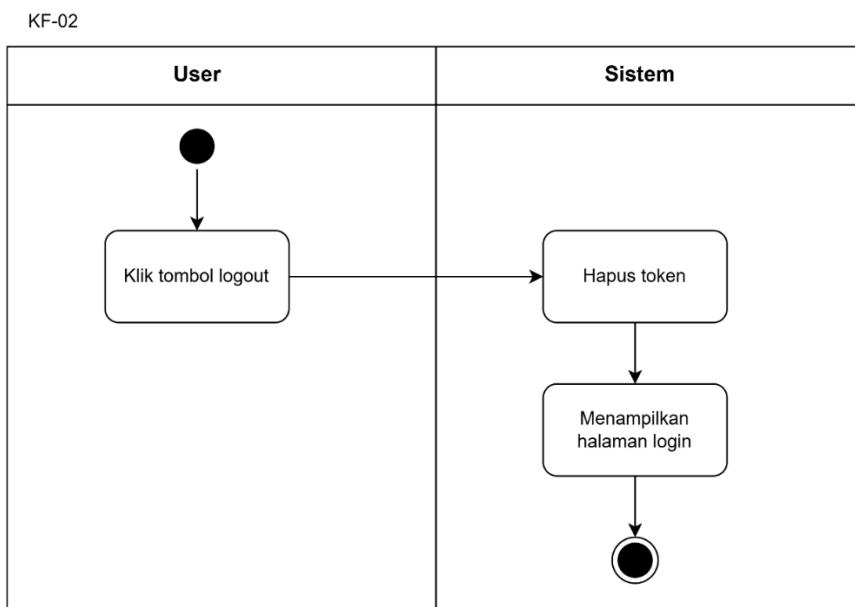
3.3.6. Activity Diagram

1. *Activity Diagram Login* (aktor mahasiswa).



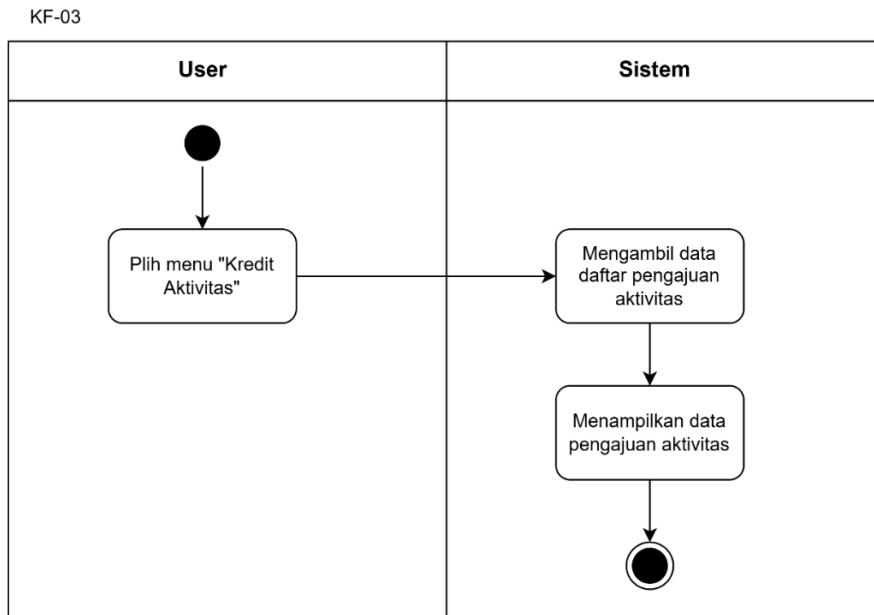
Gambar 3. 7 *Activity Diagram Login* untuk aktor mahasiswa

2. *Activity Diagram Logout* (aktor mahasiswa).



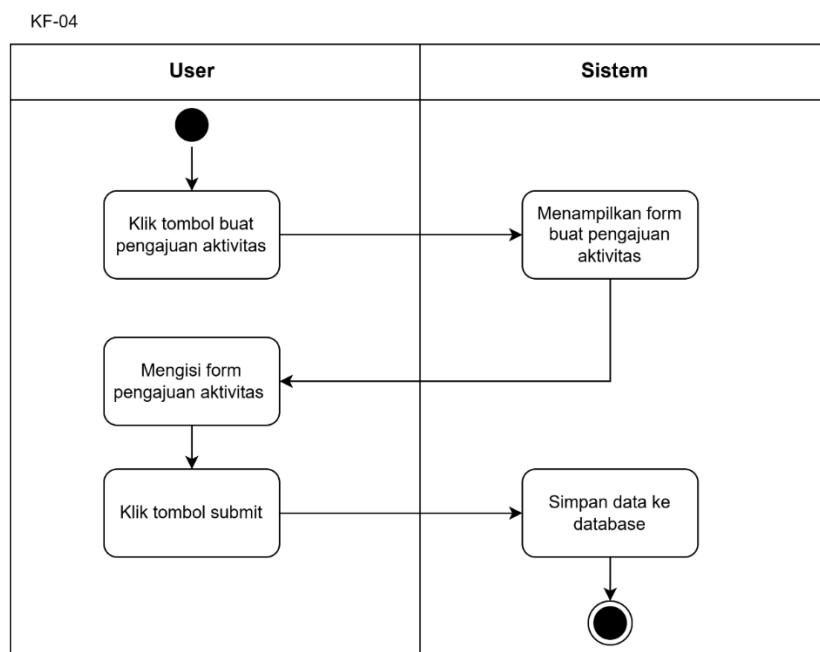
Gambar 3. 8 *Activity Diagram Logout* untuk aktor mahasiswa

3. *Activity Diagram* Lihat Daftar Pengajuan Kredit Aktivitas (aktor mahasiswa).



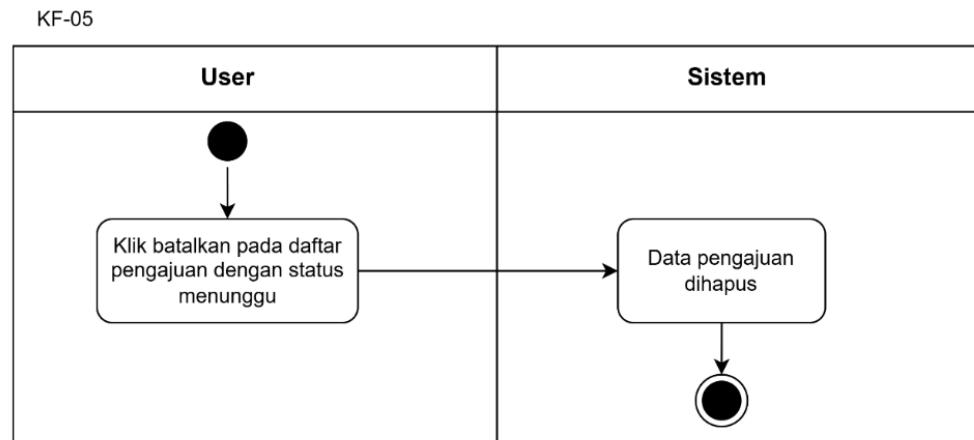
Gambar 3. 9 Activity Diagram Lihat Daftar Pengajuan Kredit Aktivitas untuk aktor mahasiswa

4. *Activity Diagram* Buat Pengajuan Kredit Aktivitas (aktor mahasiswa).



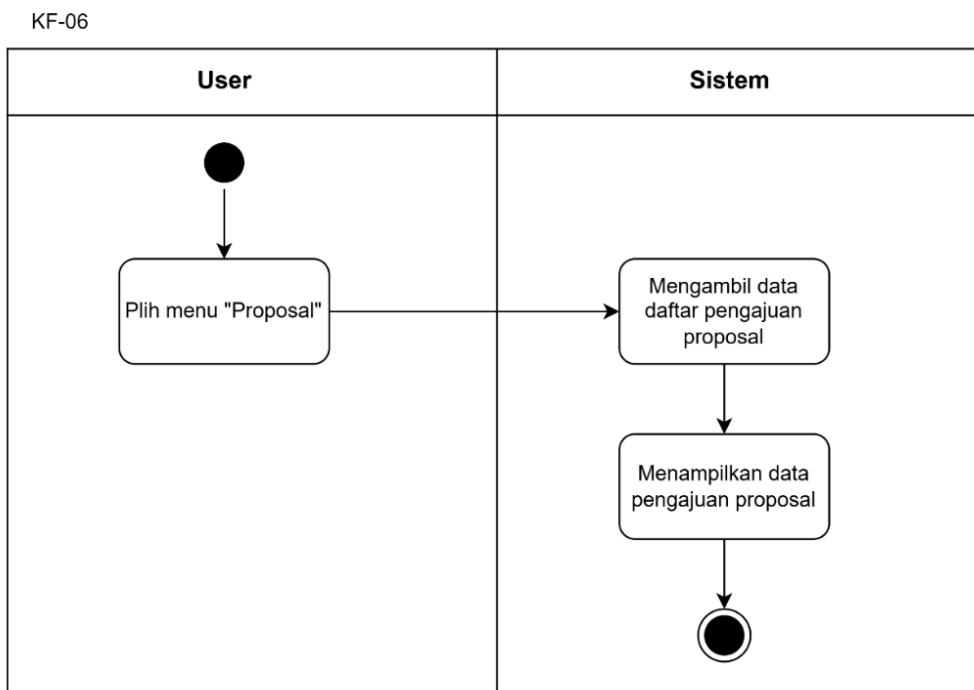
Gambar 3. 10 Activity Diagram Buat Pengajuan Kredit Aktivitas untuk aktor mahasiswa

5. *Activity Diagram* Membatalkan Pengajuan Kredit Aktivitas (aktor mahasiswa).



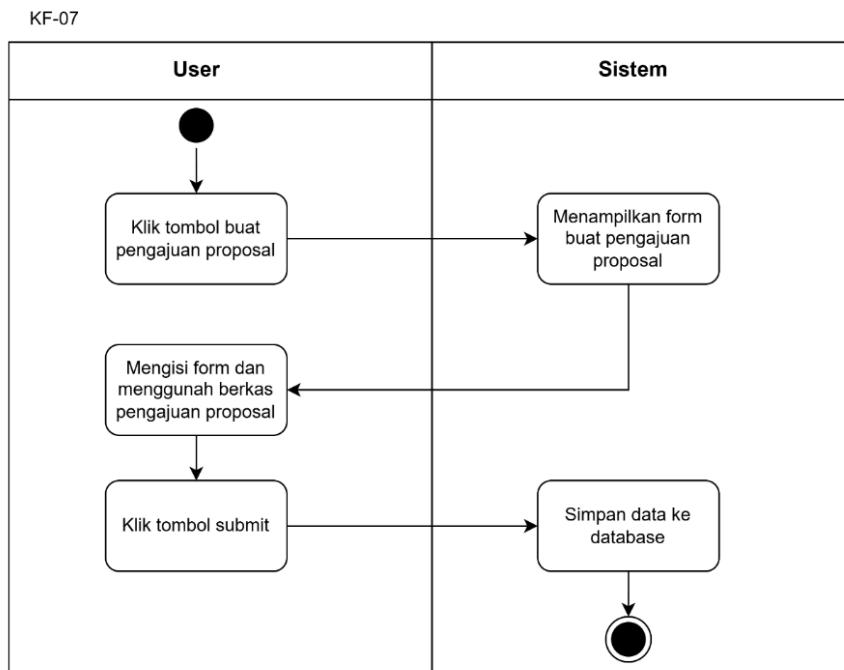
Gambar 3. 11 *Activity Diagram* Membatalkan Pengajuan Kredit Aktivitas untuk aktor mahasiswa

6. *Activity Diagram* Lihat Daftar Pengajuan Proposal (aktor mahasiswa).



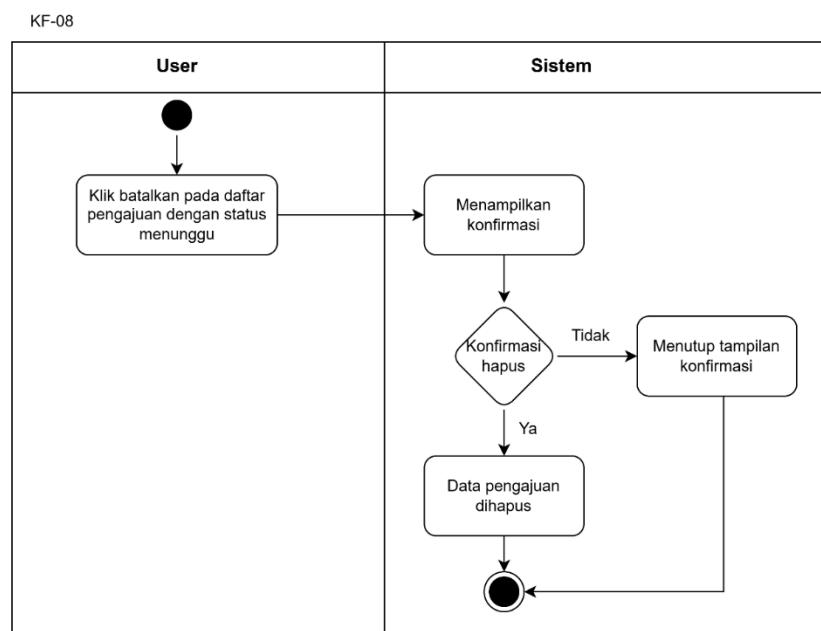
Gambar 3. 12 *Activity Diagram* Lihat Daftar Pengajuan Proposal untuk aktor mahasiswa

7. *Activity Diagram Buat Pengajuan Proposal* (aktor mahasiswa).



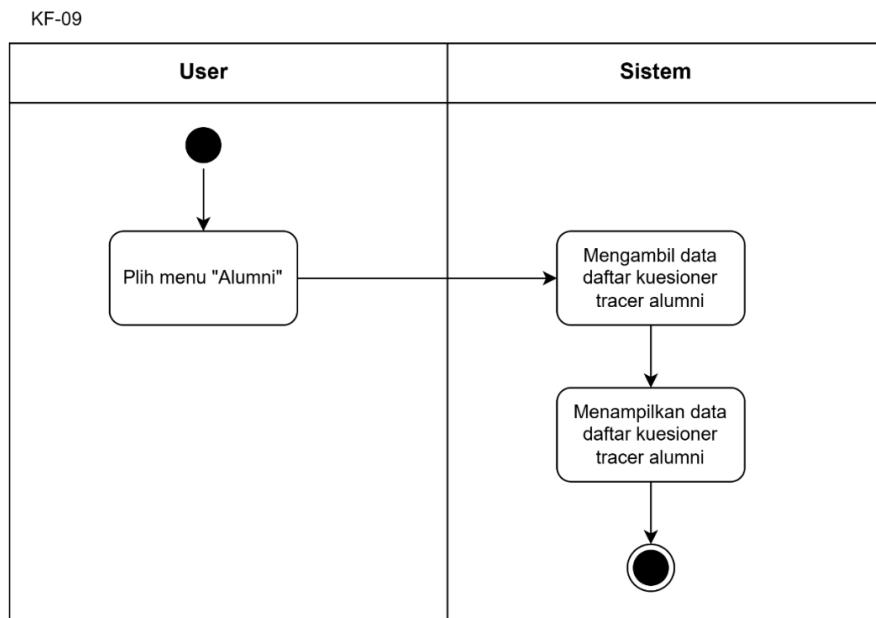
Gambar 3. 13 *Activity Diagram* Buat Pengajuan Proposal untuk aktor mahasiswa

8. *Activity Diagram Membatalkan Pengajuan Proposal* (aktor mahasiswa).



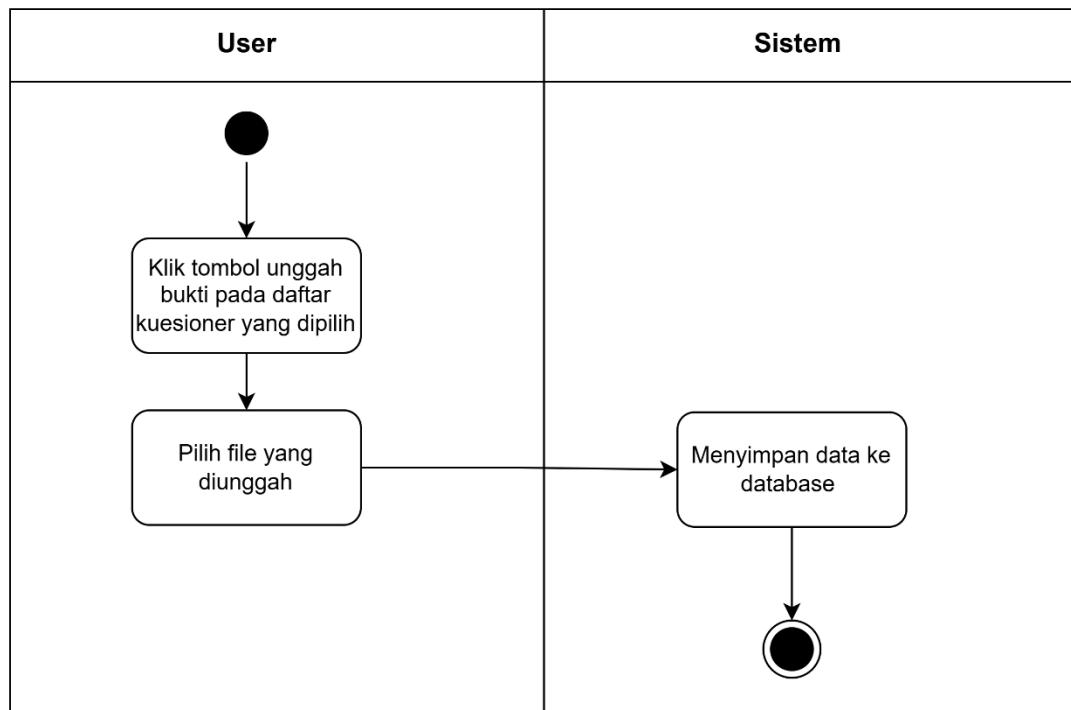
Gambar 3. 14 *Activity Diagram* Membatalkan Pengajuan Proposal untuk aktor mahasiswa

9. *Activity Diagram* Lihat Daftar Tracer Alumni (aktor mahasiswa).



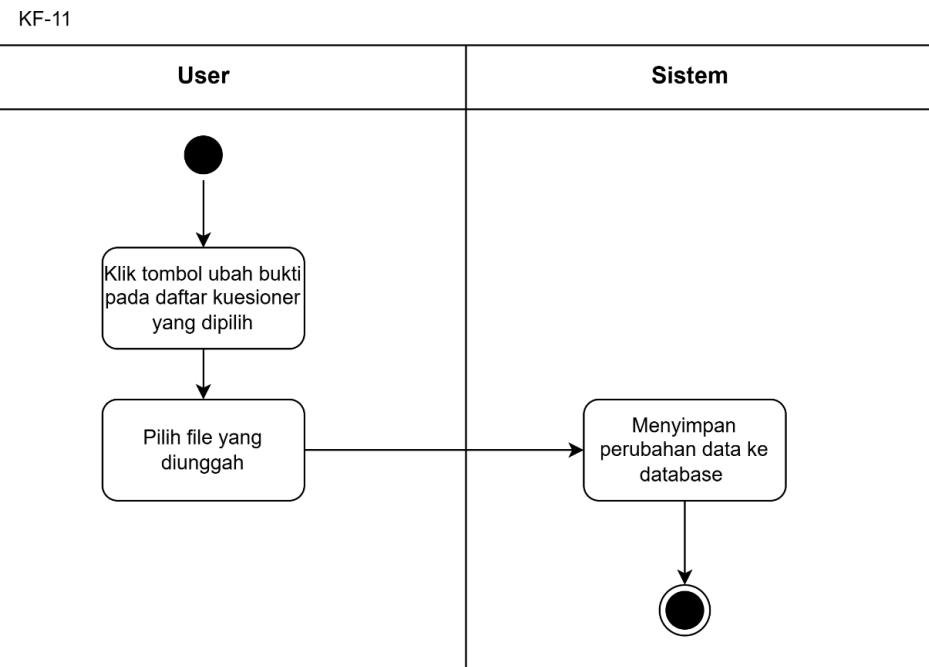
Gambar 3. 15 *Activity Diagram* Lihat Daftar Tracer Alumni untuk aktor mahasiswa

10. *Activity Diagram* Kirim Bukti Pengisian Tracer Alumni (aktor mahasiswa).



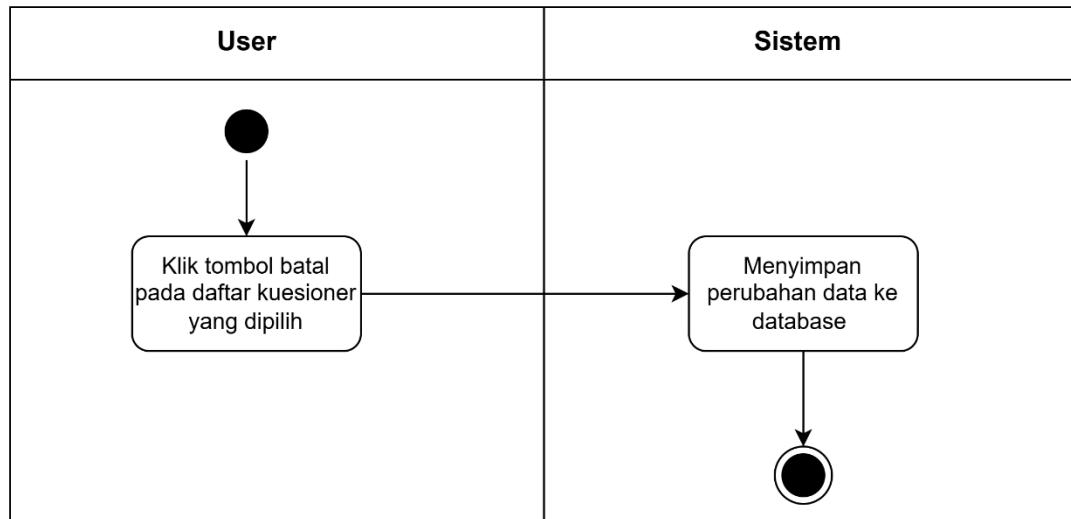
Gambar 3. 16 *Activity Diagram* Kirim Bukti Pengisian Tracer Alumni untuk aktor mahasiswa

11. *Activity Diagram* Mengubah Bukti Pengisian *Tracer Alumni* (aktor mahasiswa).



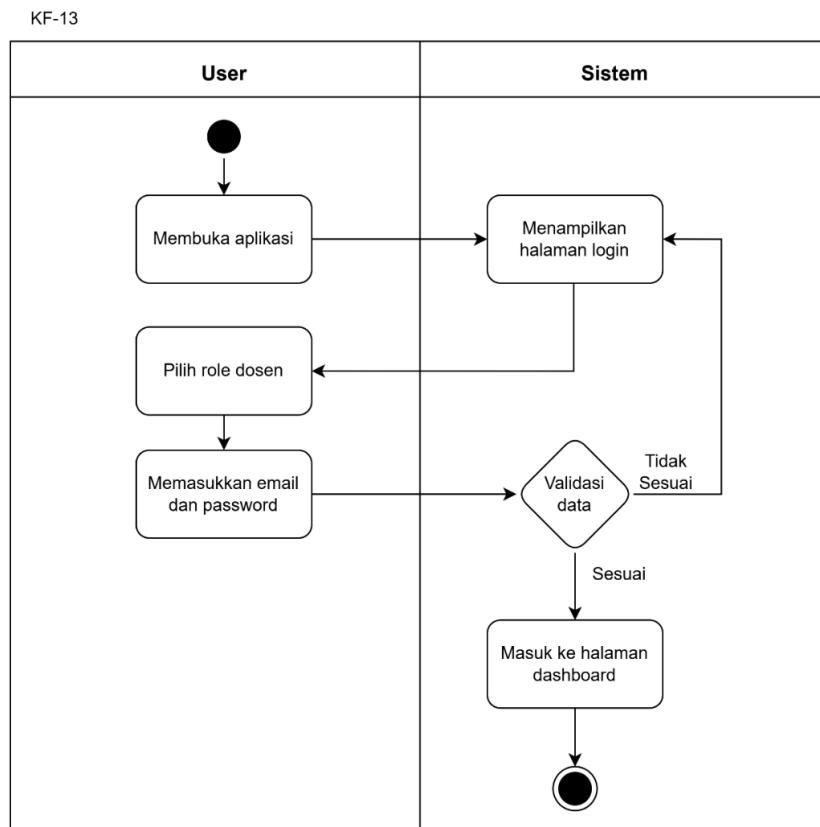
Gambar 3. 17 *Activity Diagram* Mengubah Bukti Pengisian *Tracer Alumni* untuk aktor mahasiswa

12. *Activity Diagram* Membatalkan Bukti Pengisian *Tracer Alumni* (aktor mahasiswa).



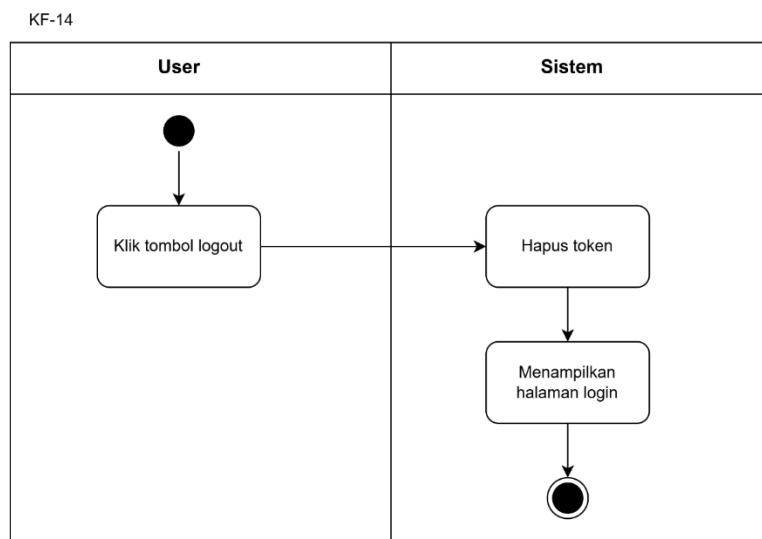
Gambar 3. 18 *Activity Diagram* Membatalkan Bukti Pengisian *Tracer Alumni* untuk aktor mahasiswa

13. *Activity Diagram Login* (aktor dosen)



Gambar 3. 19 *Activity Diagram Login* untuk aktor dosen

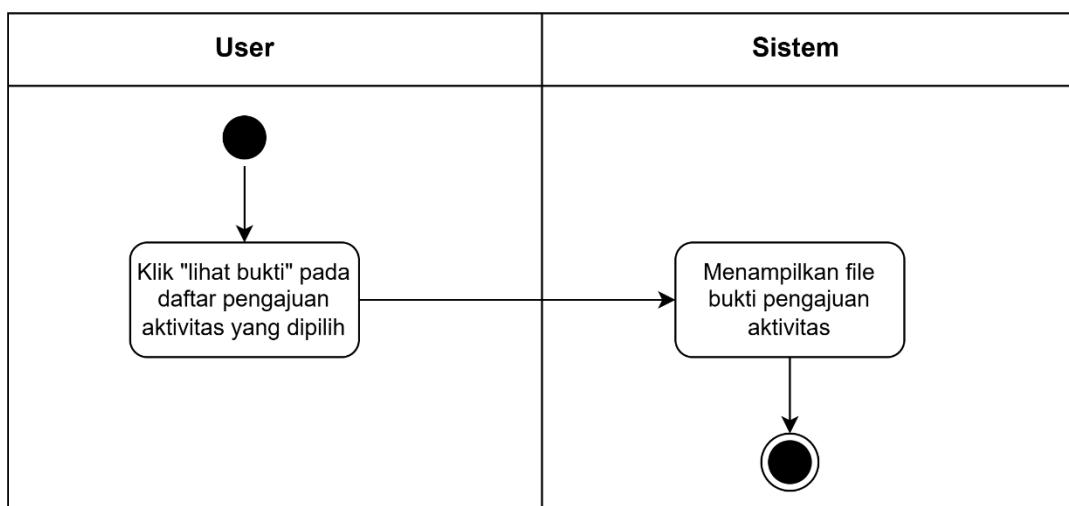
14. *Activity Diagram Logout* (aktor dosen).



Gambar 3. 20 *Activity Diagram Logout* untuk aktor dosen

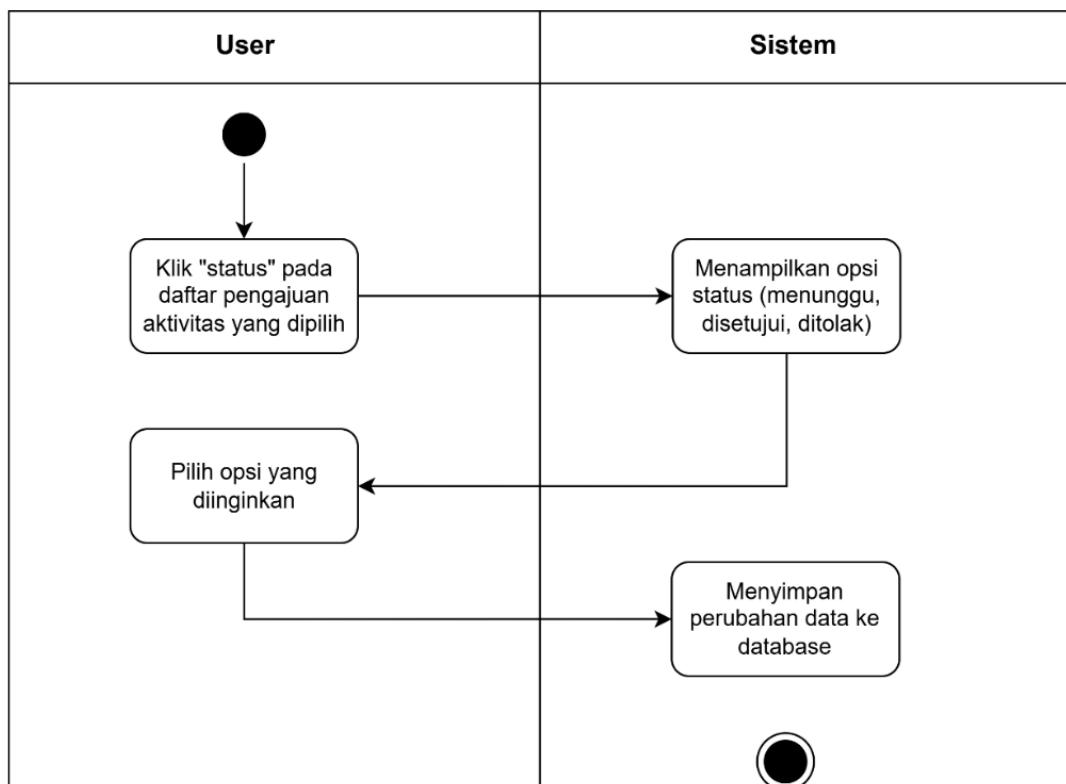
15. *Activity Diagram* Lihat Detail Bukti Aktivitas Mahasiswa (aktor dosen).

KF-15



Gambar 3. 21 *Activity Diagram* Lihat Detail Bukti Aktivitas Mahasiswa untuk aktor dosen

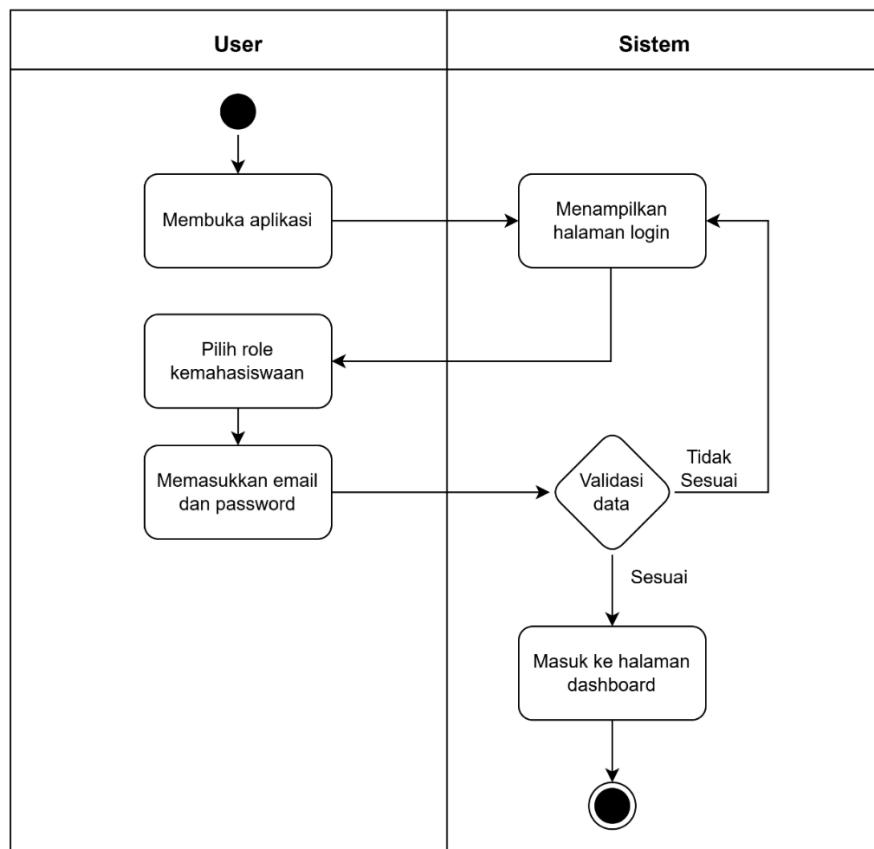
16. *Activity Diagram* Edit Status Verifikasi Pengajuan Proposal (aktor dosen).



Gambar 3. 22 *Activity Diagram* Edit Status Verifikasi Pengajuan Proposal untuk aktor dosen

17. *Activity Diagram Login* (aktor kemahasiswaan).

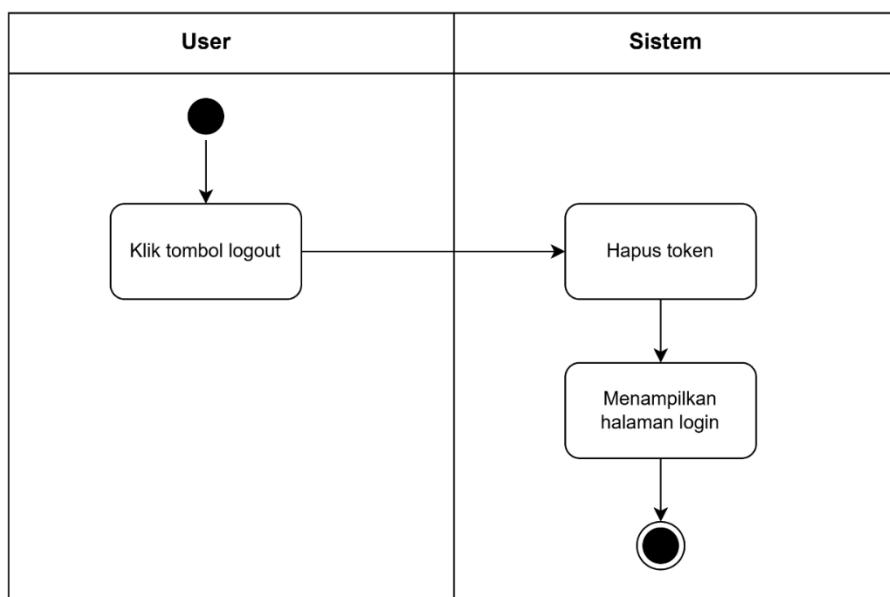
KF-17



Gambar 3. 23 *Activity Diagram Login* untuk aktor kemahasiswaan

18. *Activity Diagram Logout* (aktor kemahasiswaan).

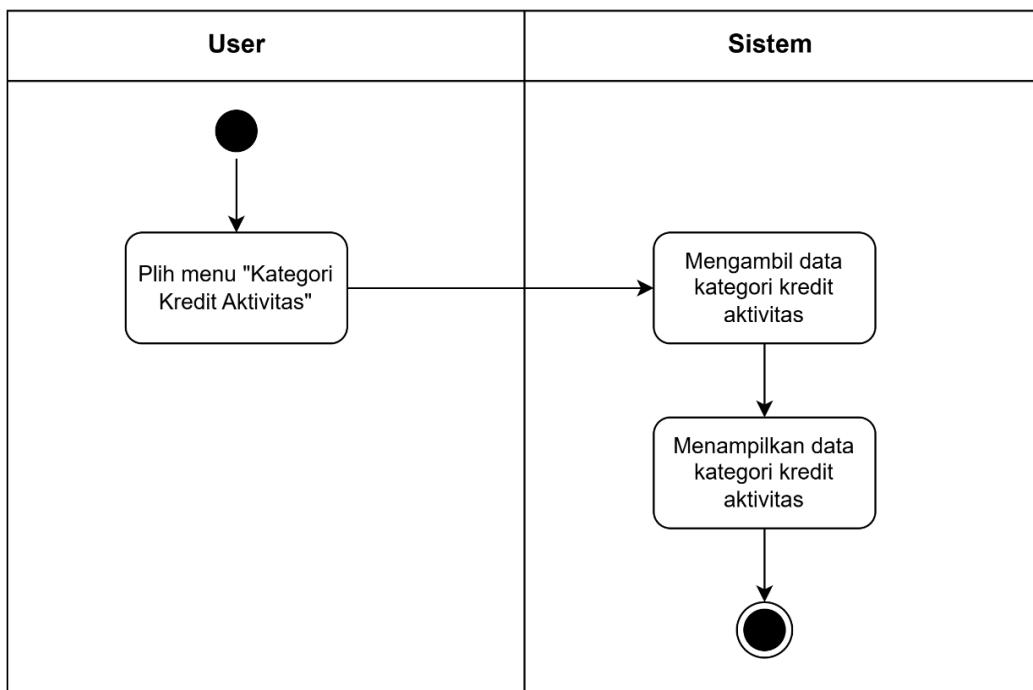
KF-18



Gambar 3. 24 *Activity Diagram Logout* untuk aktor kemahasiswaan

19. *Activity Diagram* Lihat Daftar Kategori Aktivitas (aktor kemahasiswaan).

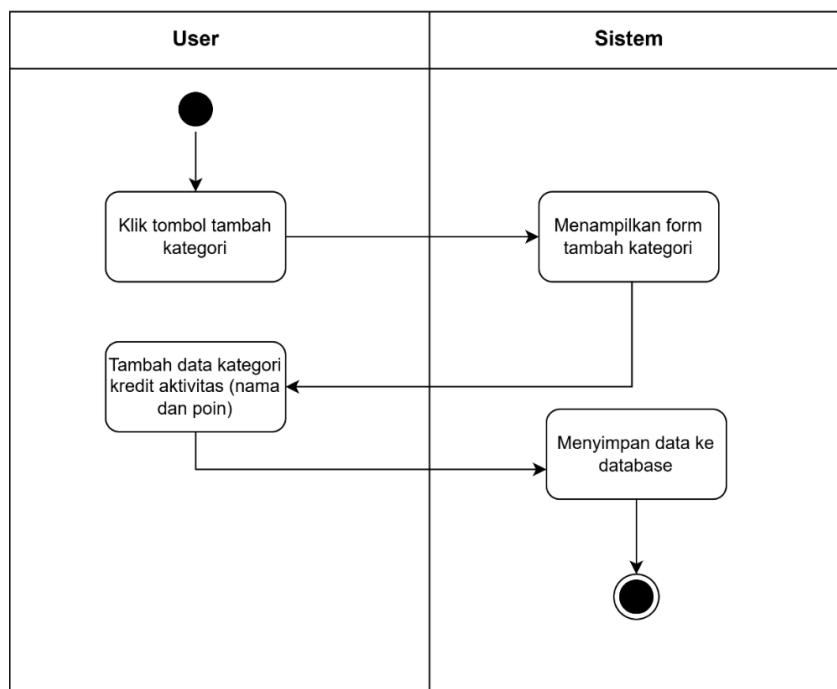
KF-19



Gambar 3. 25 *Activity Diagram* Lihat Daftar Kategori Aktivitas untuk aktor kemahasiswaan

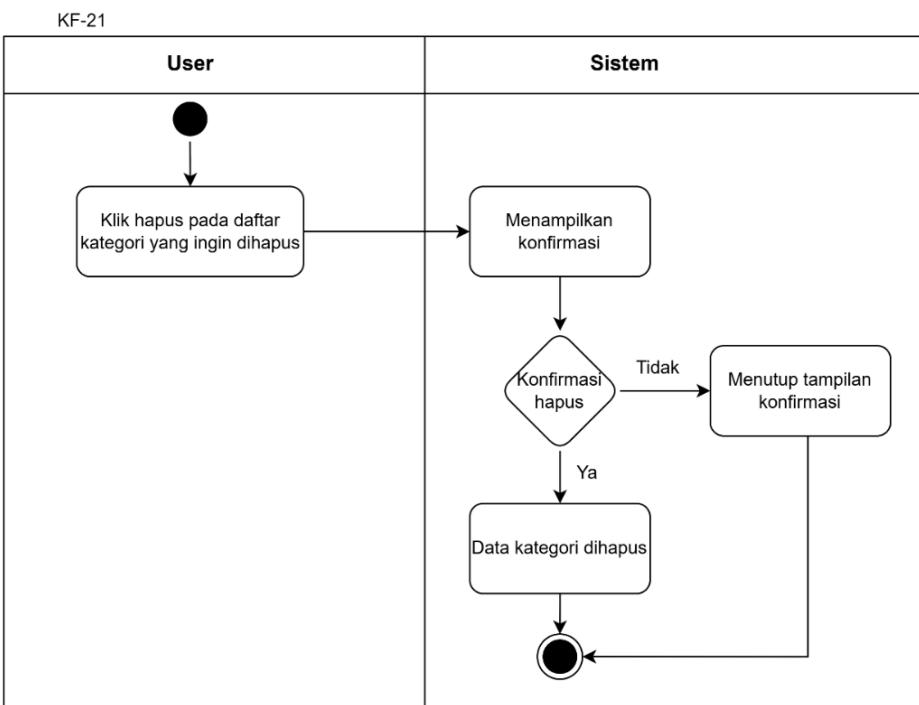
20. *Activity Diagram* Tambah Data Kategori Aktivitas (aktor kemahasiswaan).

KF-20



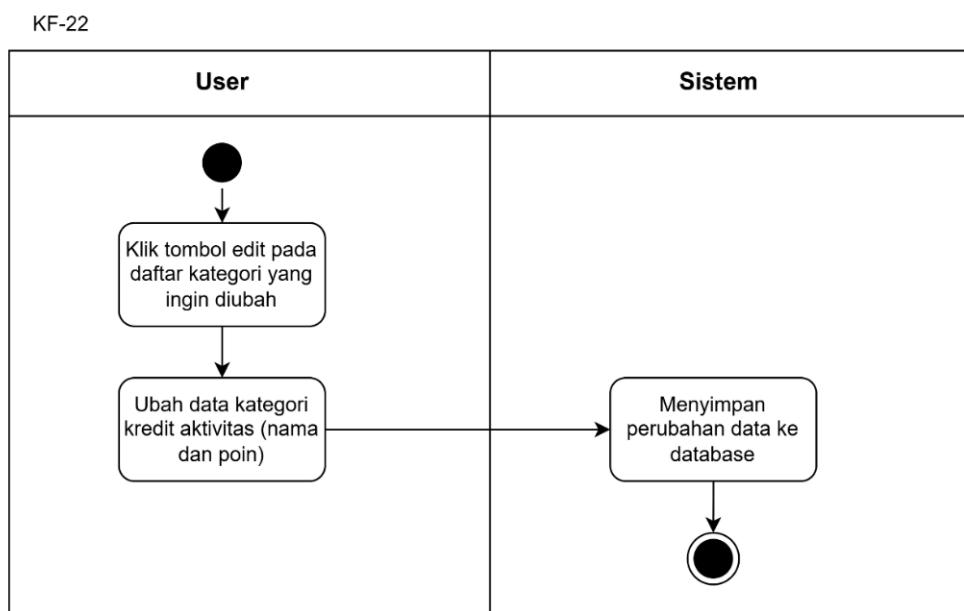
Gambar 3. 26 *Activity Diagram* Tambah Data Kategori Aktivitas untuk aktor kemahasiswaan

21. *Activity Diagram Hapus Data Kategori Aktivitas* (aktor kemahasiswaan).



Gambar 3. 27 Activity Diagram Hapus Data Kategori Aktivitas untuk aktor kemahasiswaan

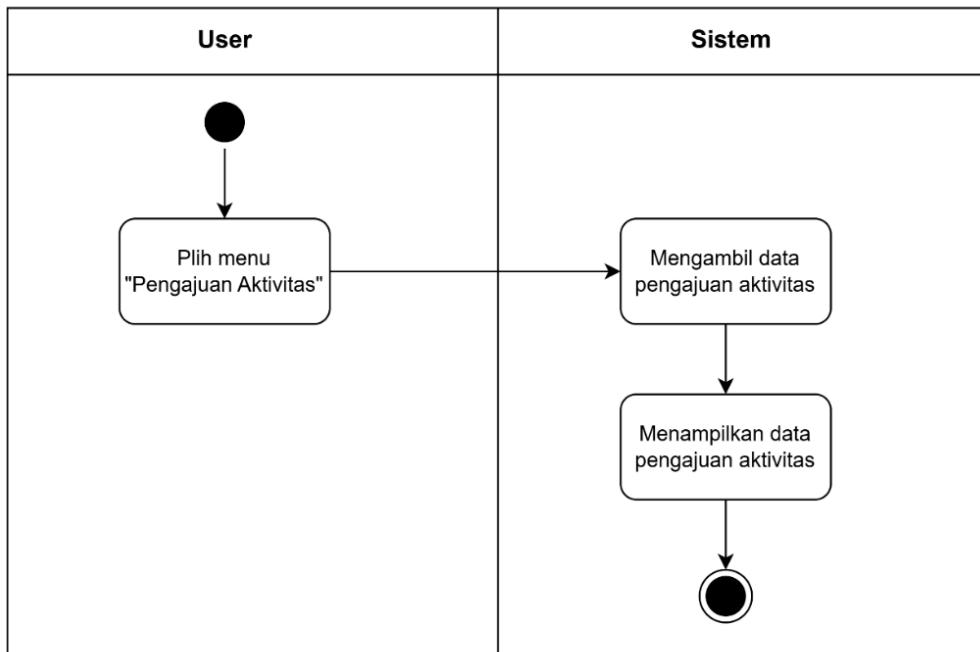
22. *Activity Diagram Edit Data Kategori Aktivitas* (aktor kemahasiswaan).



Gambar 3. 28 Activity Diagram Edit Data Kategori Aktivitas untuk aktor kemahasiswaan

23. *Activity Diagram* Lihat Daftar Pengajuan Aktivitas Mahasiswa (aktor kemahasiswaan).

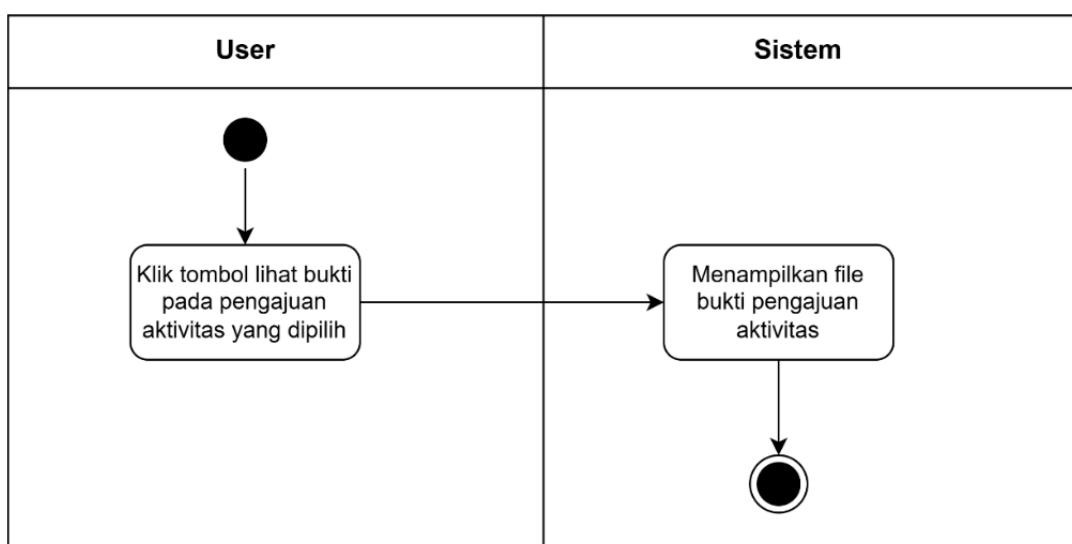
KF-23



Gambar 3. 29 *Activity Diagram* Lihat Daftar Pengajuan Aktivitas Mahasiswa untuk aktor kemahasiswaan

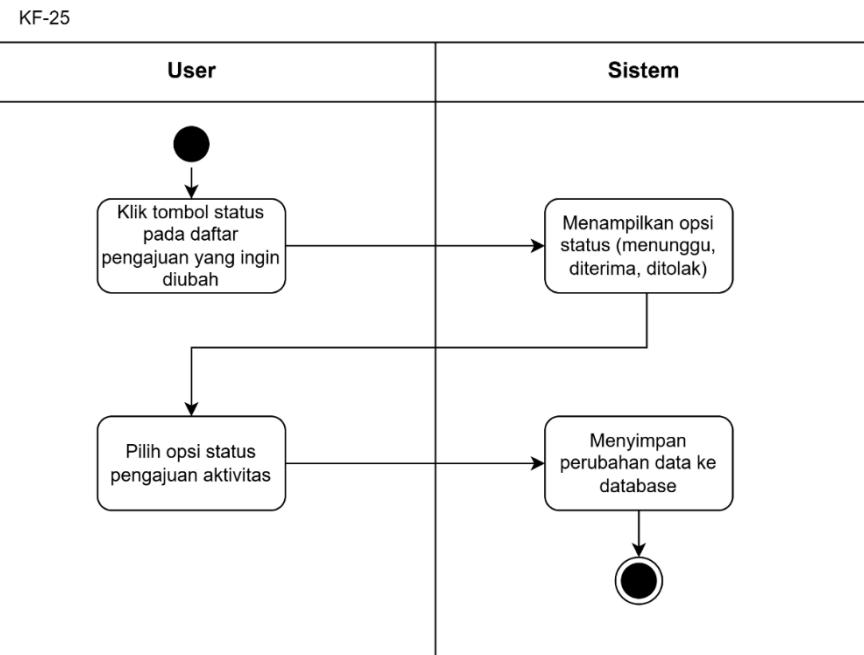
24. *Activity Diagram* Lihat Detail Bukti Aktivitas Mahasiswa (aktor kemahasiswaan).

KF-24



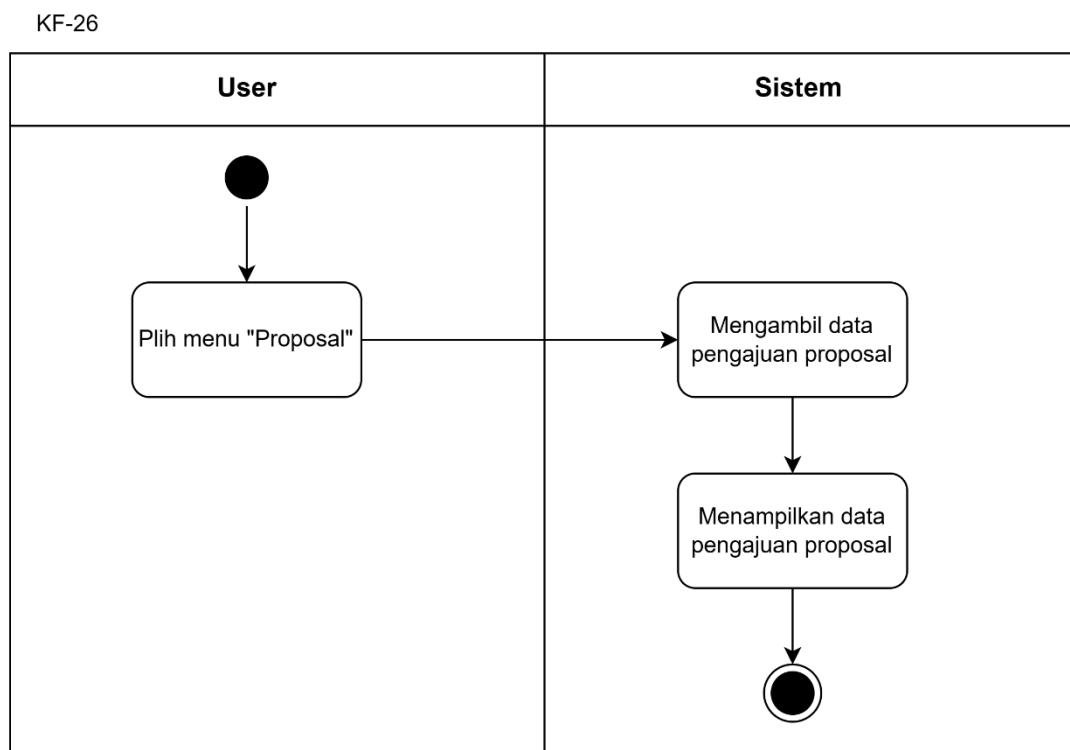
Gambar 3. 30 *Activity Diagram* Lihat Detail Bukti Aktivitas Mahasiswa untuk aktor kemahasiswaan

25. *Activity Diagram* Edit Status Verifikasi Pengajuan Aktivitas (aktor kemahasiswaan)



Gambar 3.31 *Activity Diagram* Edit Status Verifikasi Pengajuan Aktivitas untuk aktor kemahasiswaan

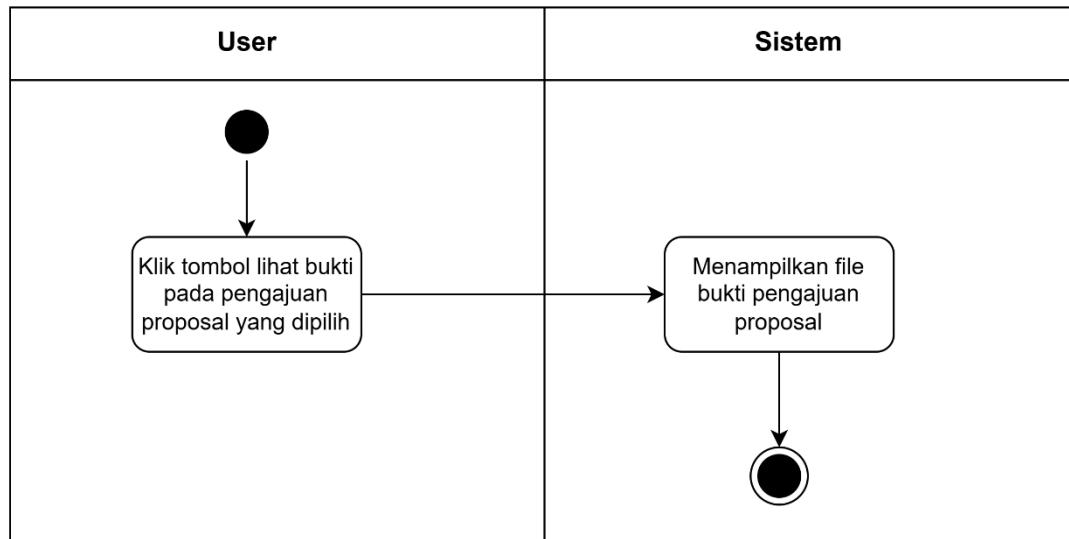
26. *Activity Diagram* Lihat Daftar Pengajuan Proposal (aktor kemahasiswaan).



Gambar 3.32 *Activity Diagram* Lihat Daftar Pengajuan Proposal untuk aktor kemahasiswaan

27. *Activity Diagram* Lihat Detail Bukti pengajuan Proposal (aktor kemahasiswaan).

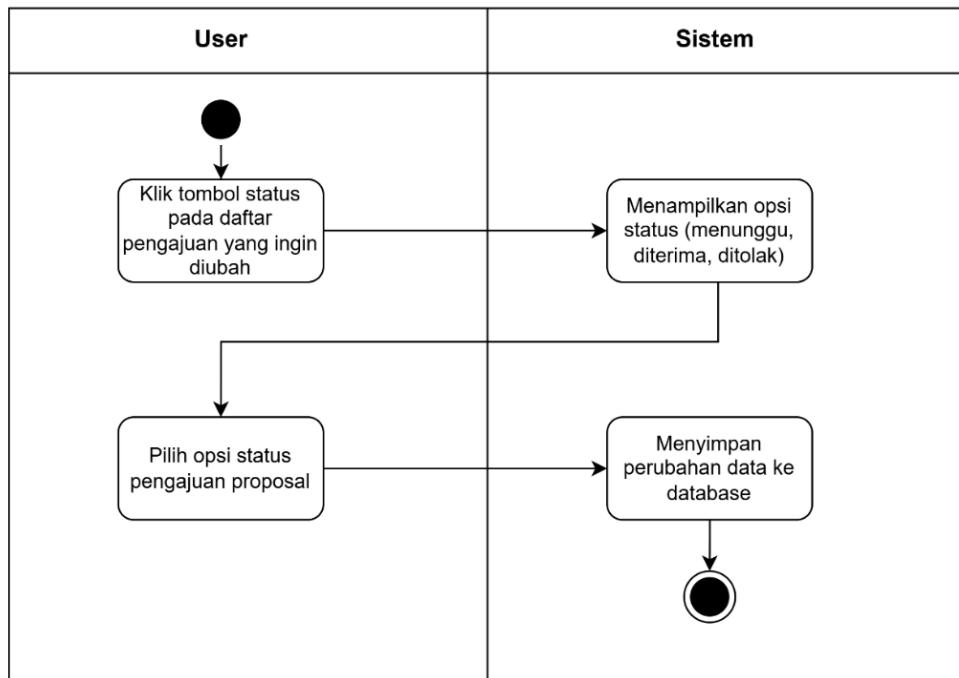
KF-27



Gambar 3. 33 *Activity Diagram* Lihat Detail Bukti pengajuan Proposal untuk aktor kemahasiswaan

28. *Activity Diagram* Edit Status Verifikasi Pengajuan Proposal (aktor kemahasiswaan).

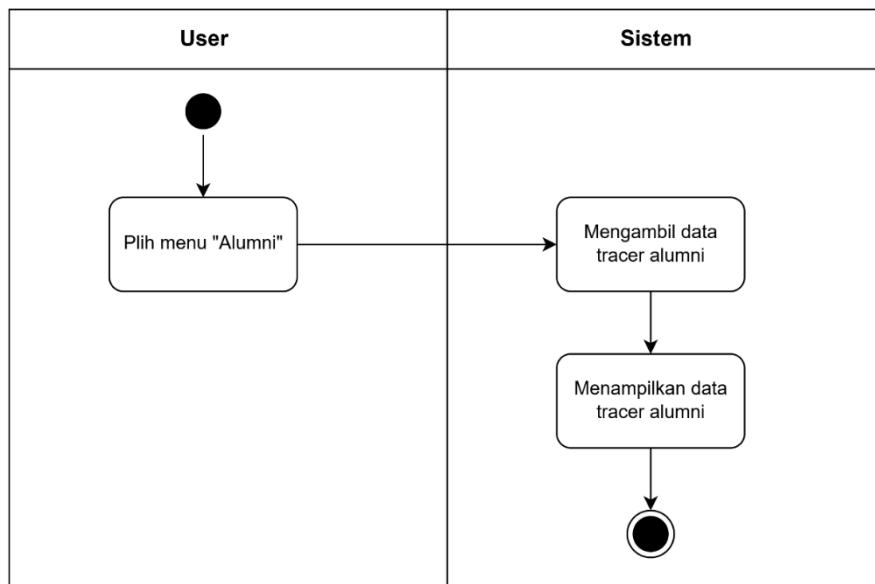
KF-28



Gambar 3. 34 *Activity Diagram* Edit Status Verifikasi Pengajuan Proposal untuk aktor kemahasiswaan

29. *Activity Diagram* Lihat Daftar Tracer Alumni (aktor kemahasiswaan).

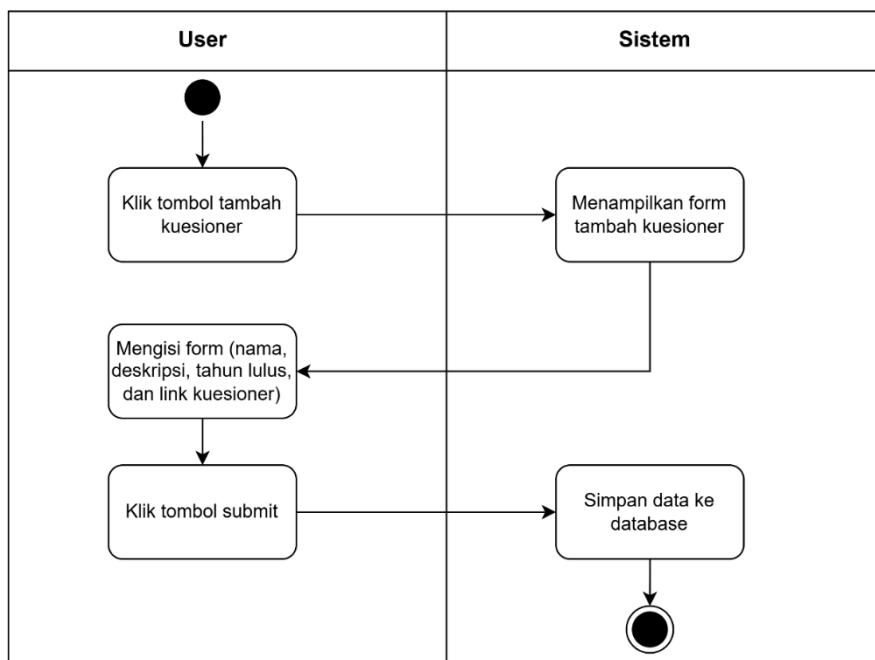
KF-29



Gambar 3. 35 *Activity Diagram* Lihat Daftar Tracer Alumni untuk aktor kemahasiswaan

30. *Activity Diagram* Tambah Data Tracer Alumni (aktor kemahasiswaan).

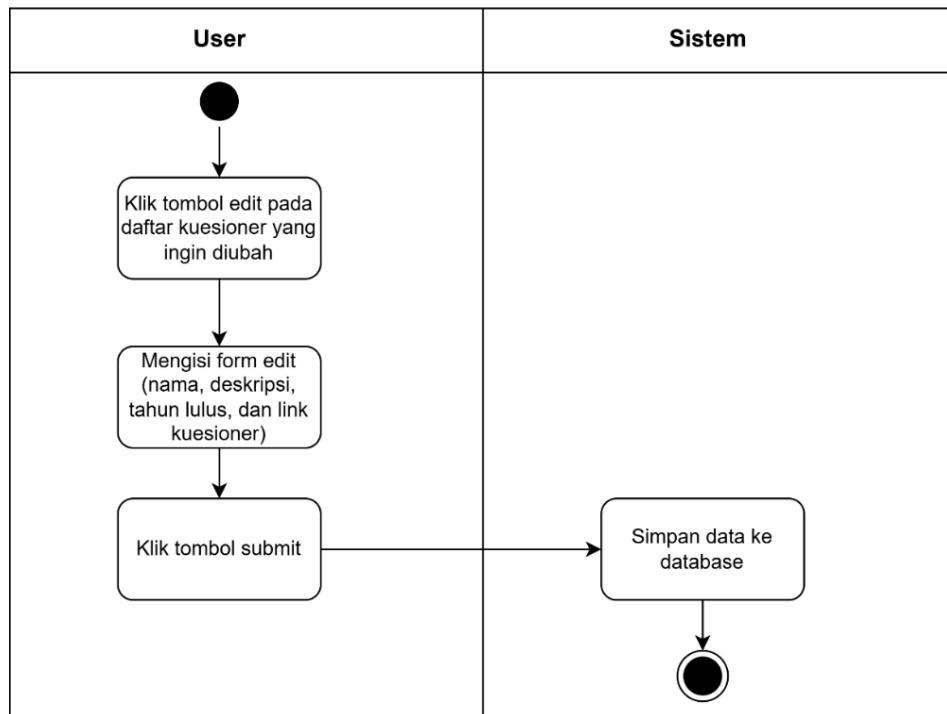
KF-30



Gambar 3. 36 *Activity Diagram* Tambah Data Tracer Alumni untuk aktor kemahasiswaan

31. *Activity Diagram Edit Data Tracer Alumni* (aktor kemahasiswaan)

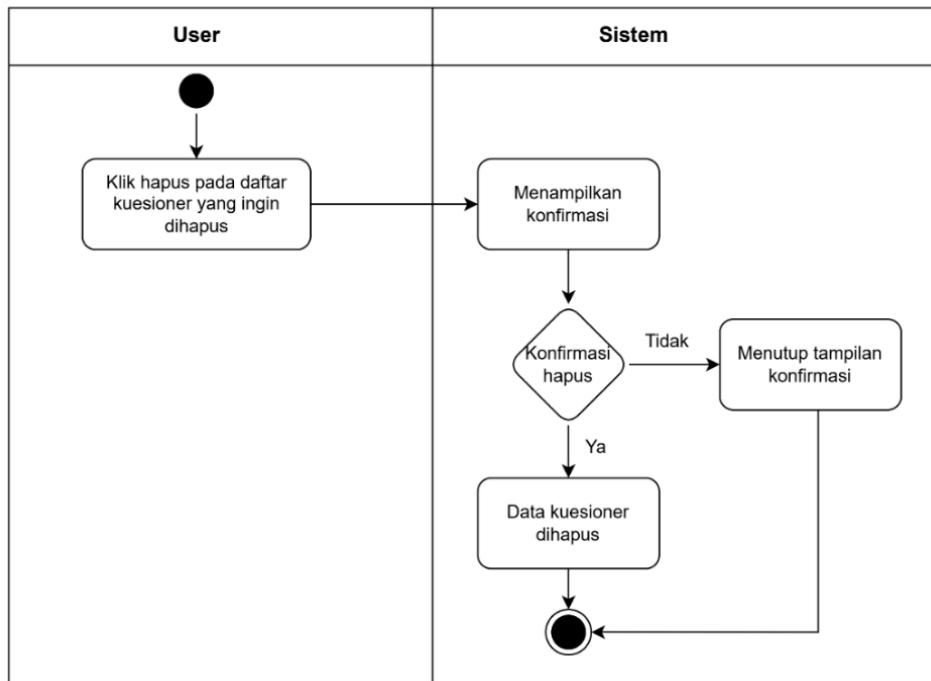
KF-31



Gambar 3. 37 *Activity Diagram Edit Data Tracer Alumni* untuk aktor kemahasiswaan

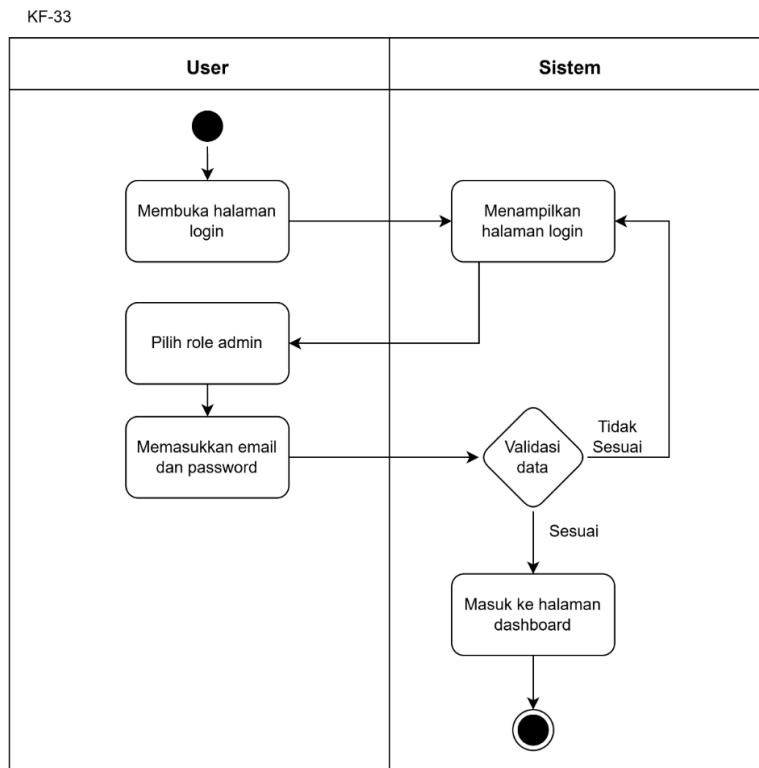
32. *Activity Diagram Hapus Data Tracer Alumni* (aktor kemahasiswaan).

KF-32



Gambar 3. 38 *Activity Diagram Hapus Data Tracer Alumni* untuk aktor kemahasiswaan

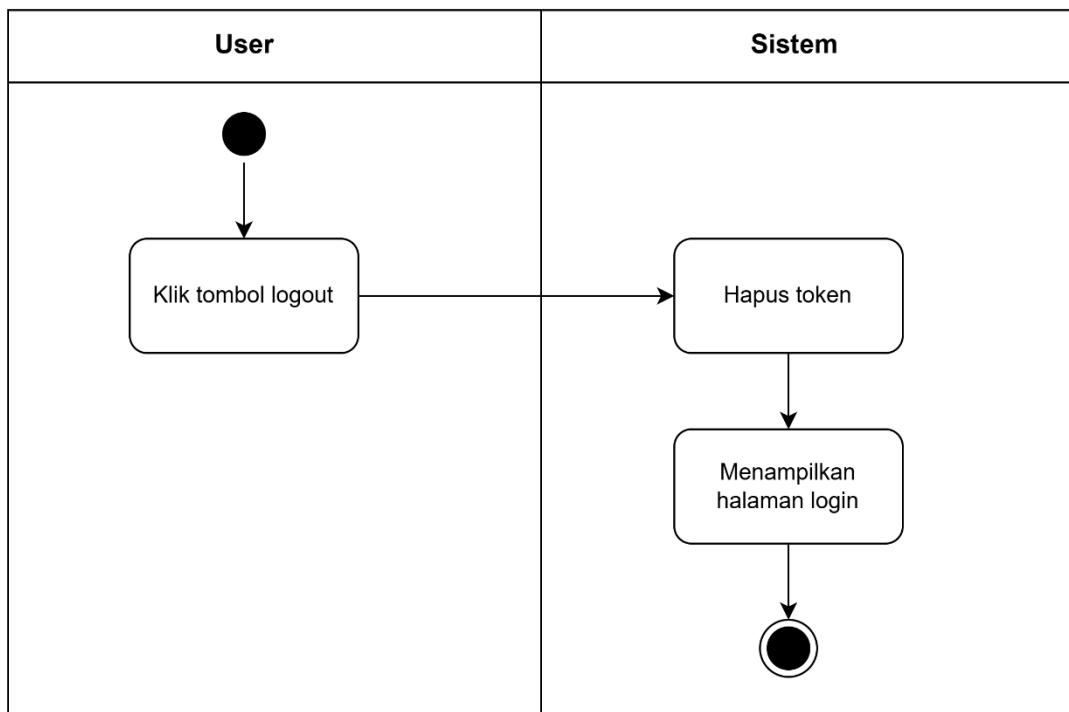
33. *Activity Diagram Login* (aktor admin).



Gambar 3. 39 *Activity Diagram Login* untuk aktor admin

34. *Activity Diagram Logout* (aktor admin)

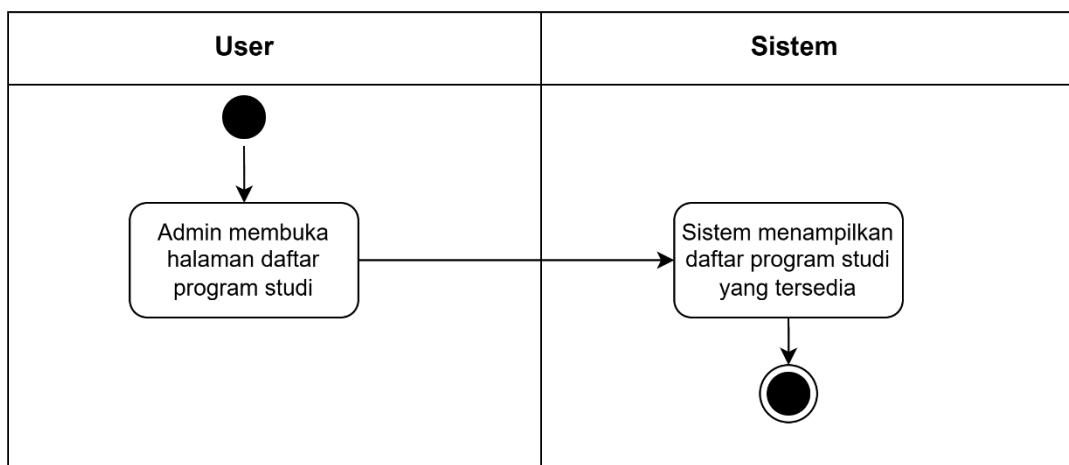
KF-34



Gambar 3. 40 *Activity Diagram Logout* untuk aktor admin

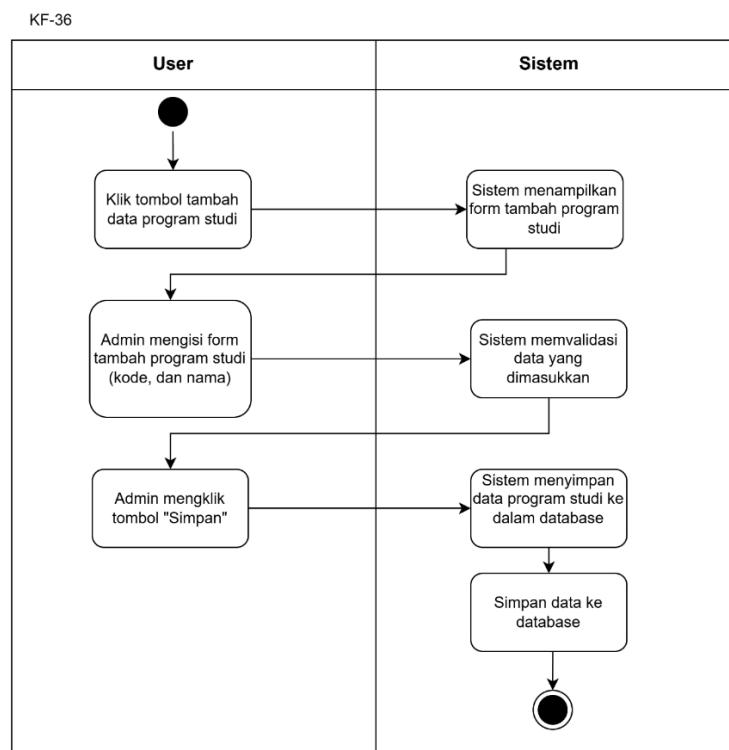
35. *Activity Diagram Lihat Daftar Program Studi* (aktor admin).

KF-35



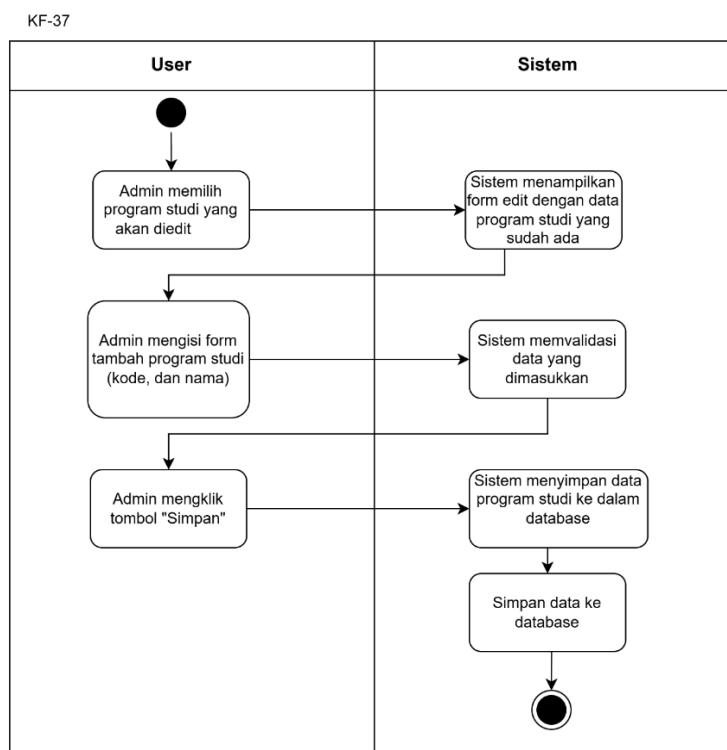
Gambar 3. 41 *Activity Diagram Lihat Daftar Program Studi* untuk aktor admin

36. *Activity Diagram* Tambah Data Daftar Program Studi (aktor admin).



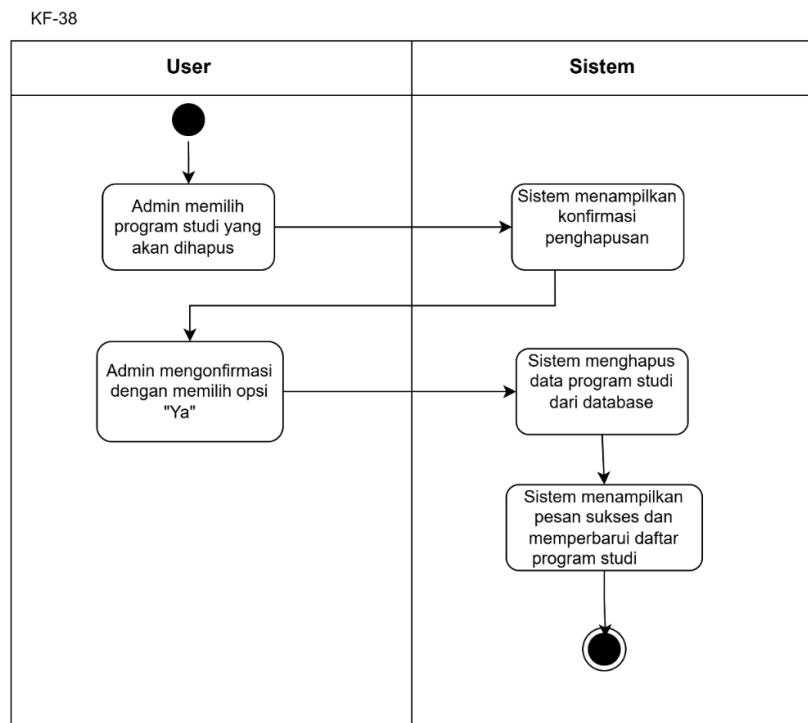
Gambar 3. 42 Activity Diagram Tambah Data Daftar Program Studi untuk aktor admin

37. *Activity Diagram* Edit Data Program Studi (aktor admin).



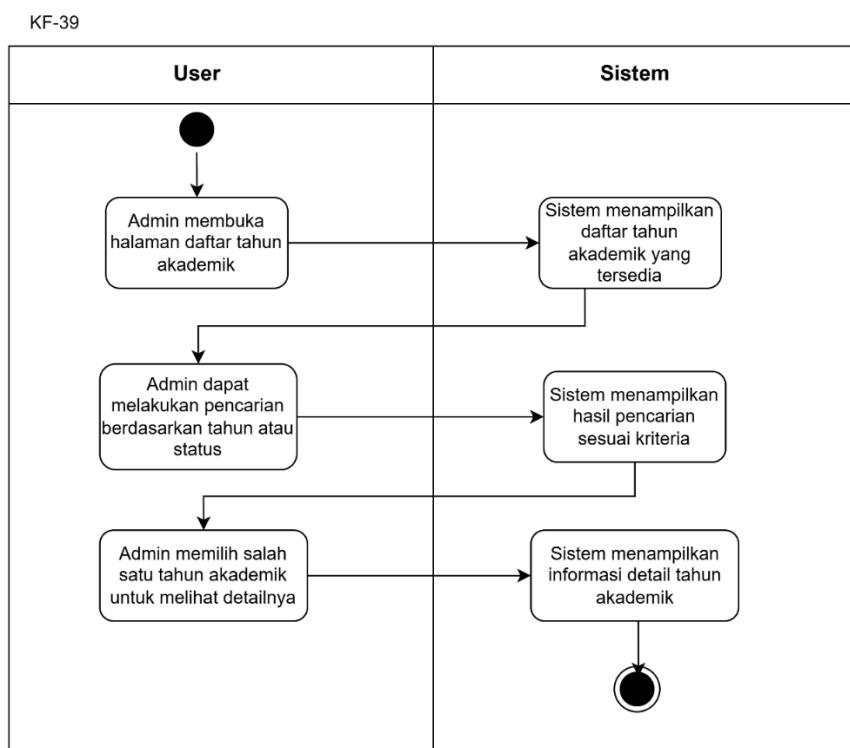
Gambar 3. 43 Activity Diagram Edit Data Program Studi untuk aktor admin

38. *Activity Diagram Hapus Data Program Studi (aktor admin).*



Gambar 3. 44 *Activity Diagram Hapus Data Program Studi untuk aktor admin*

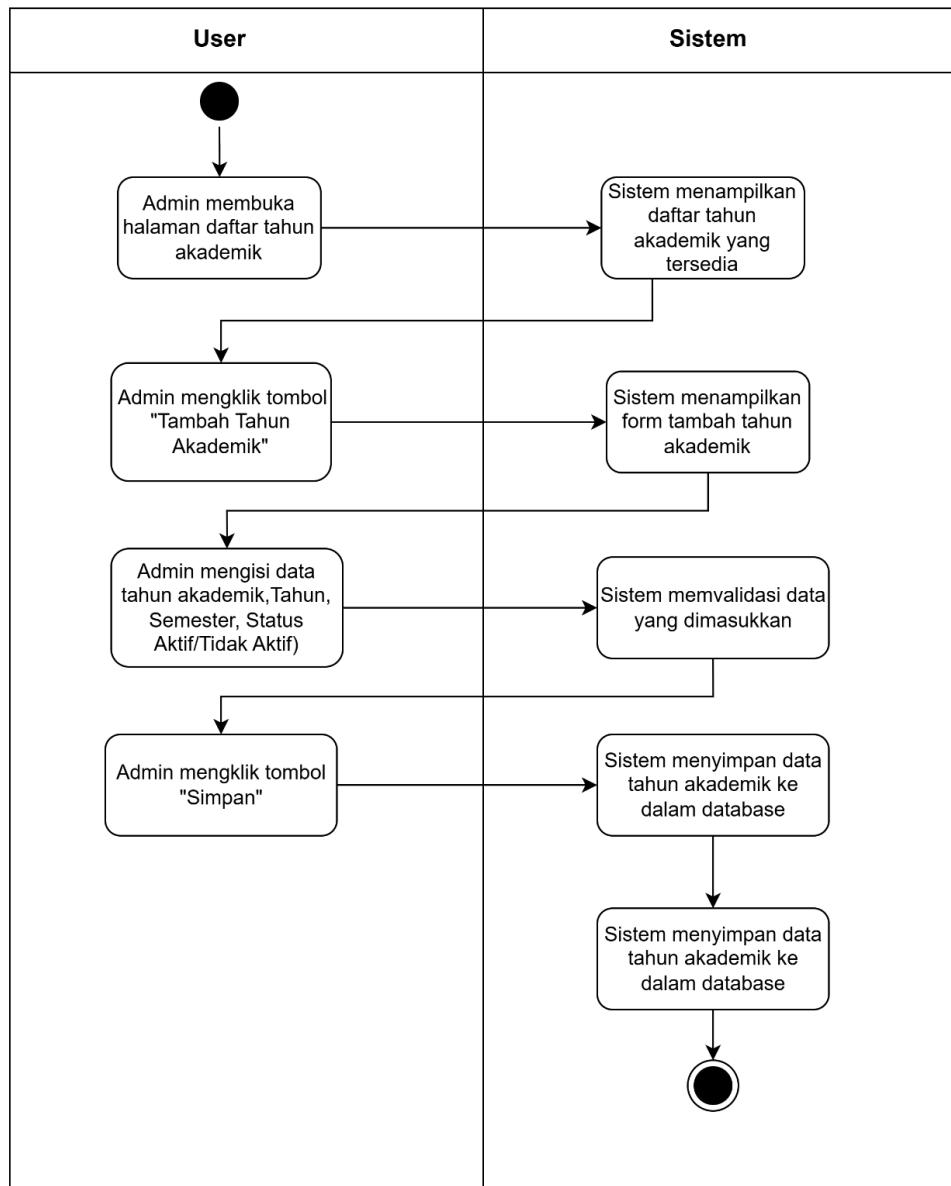
39. *Activity Diagram Lihat Daftar Tahun Akademik (aktor admin).*



Gambar 3. 45 *Activity Diagram Lihat Daftar Tahun Akademik untuk aktor admin*

40. *Activity Diagram* Tambah Data Tahun Akademik (aktor admin).

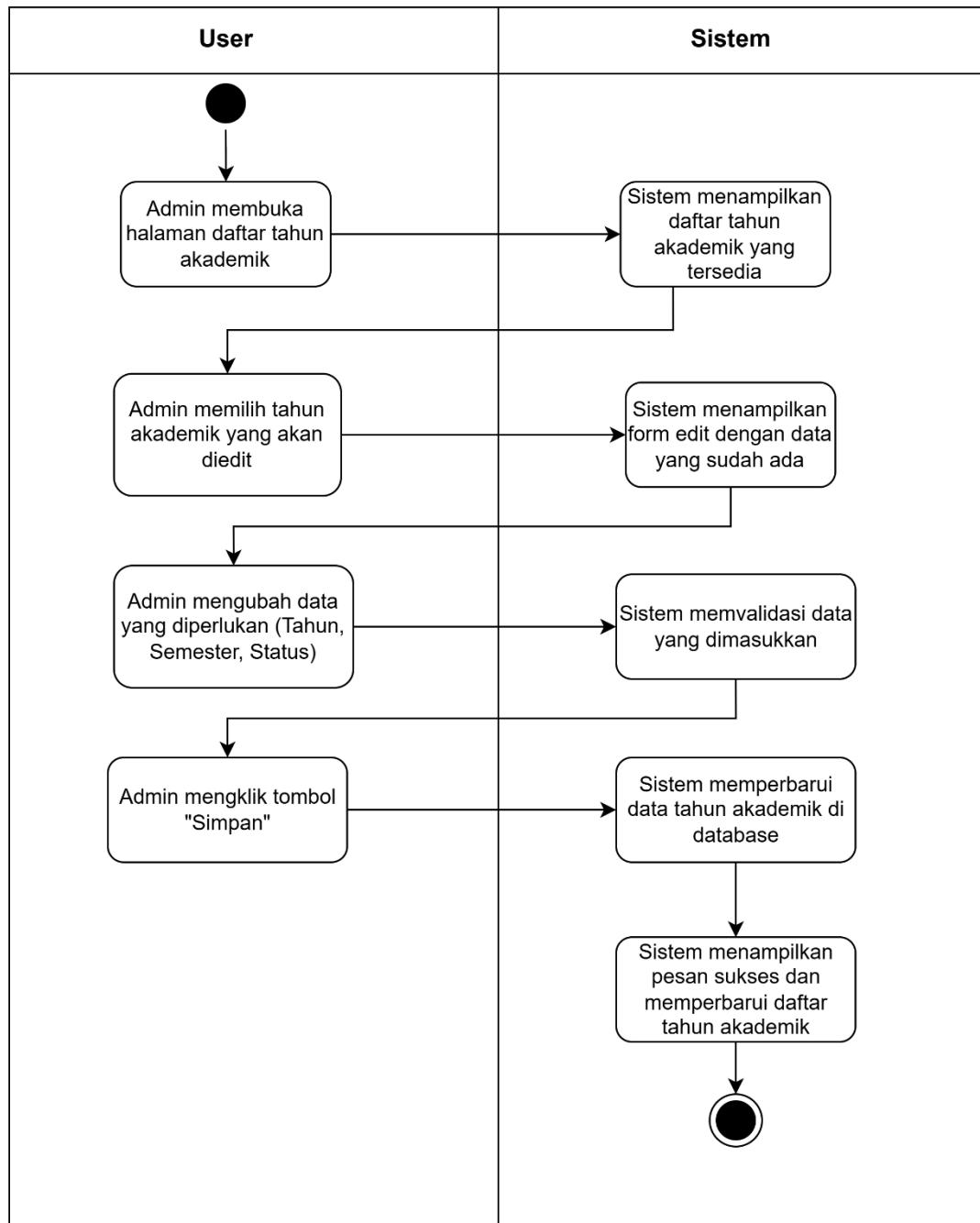
KF-40



Gambar 3. 46 *Activity Diagram* Tambah Data Tahun Akademik untuk aktor admin

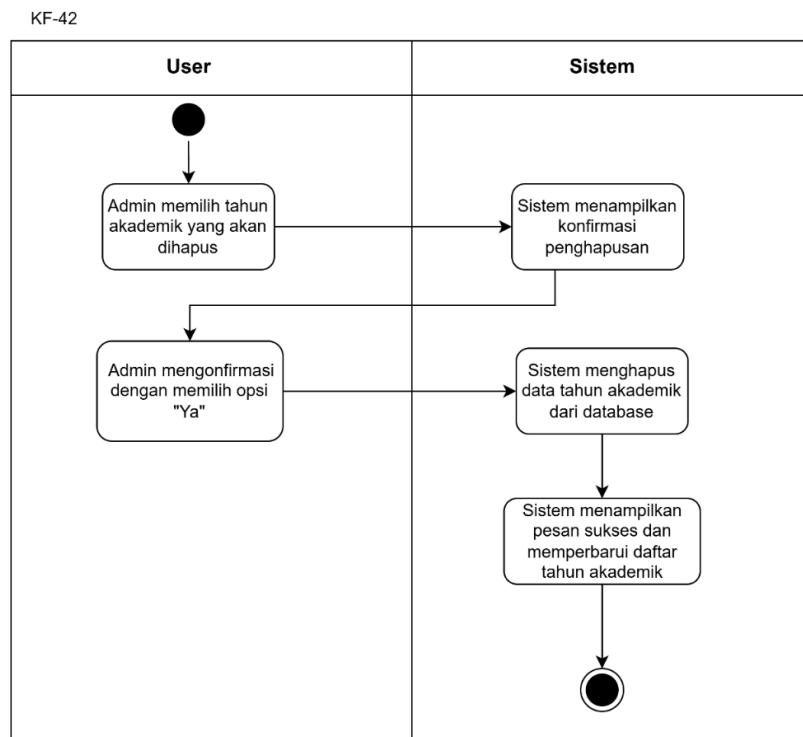
41. *Activity Diagram Edit Data Tahun Akademik (aktor admin).*

KF-41



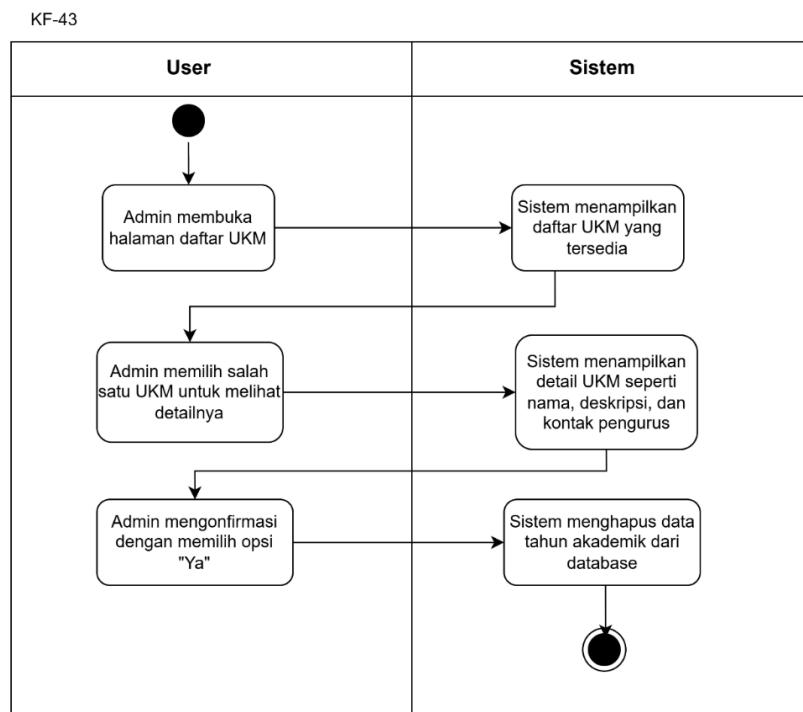
Gambar 3. 47 *Activity Diagram Edit Data Tahun Akademik untuk aktor admin*

42. *Activity Diagram* Hapus Data Tahun Akademik (aktor admin).



Gambar 3. 48 *Activity Diagram* Hapus Data Tahun Akademik untuk aktor admin

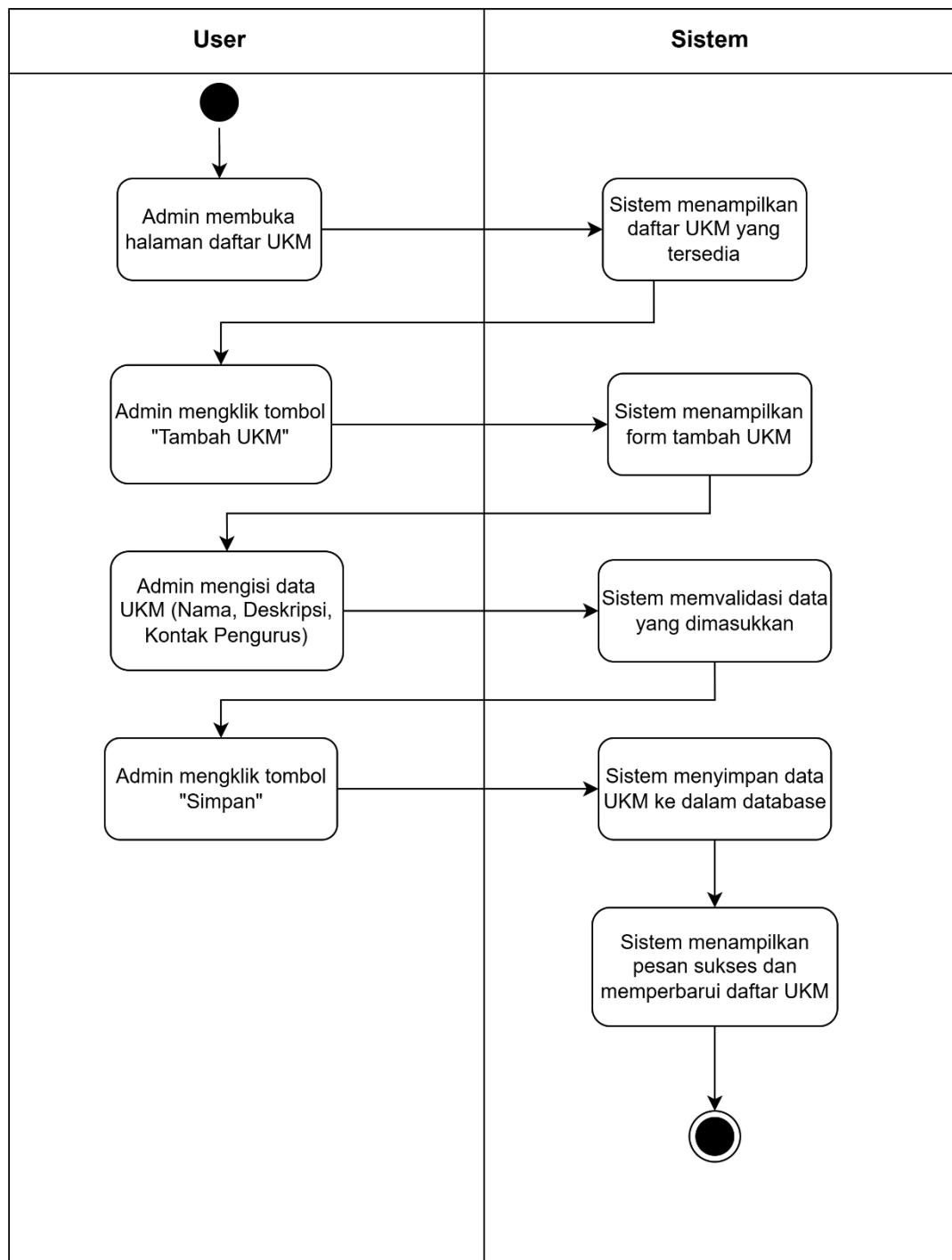
43. *Activity Diagram* Lihat Daftar Unit Kegiatan Mahasiswa (aktor admin).



Gambar 3. 49 *Activity Diagram* Lihat Daftar Unit Kegiatan Mahasiswa untuk aktor admin

44. *Activity Diagram Tambah Data Unit Kegiatan Mahasiswa (aktor admin).*

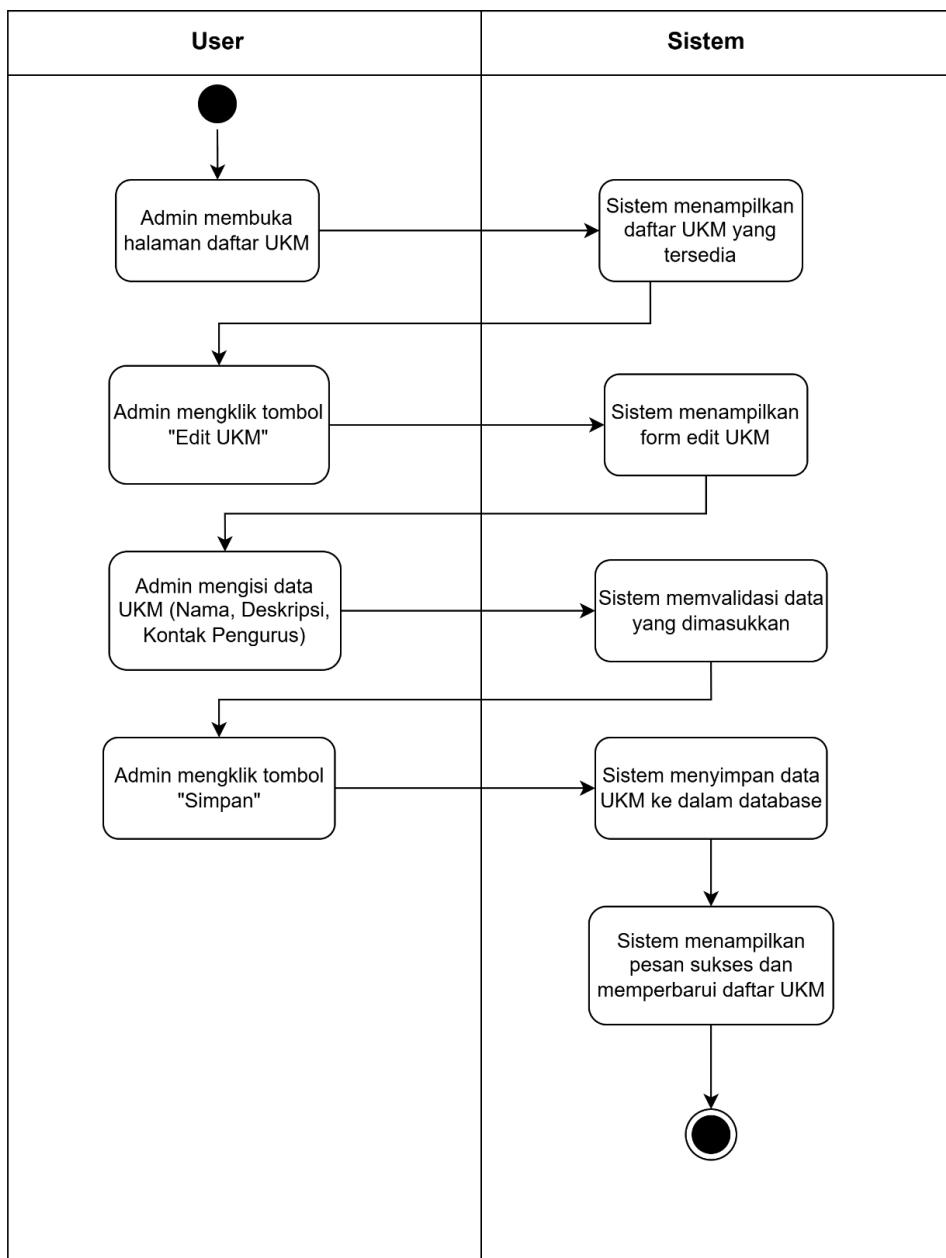
KF-44



Gambar 3. 50 Activity Diagram Tambah Data Unit Kegiatan Mahasiswa untuk aktor admin

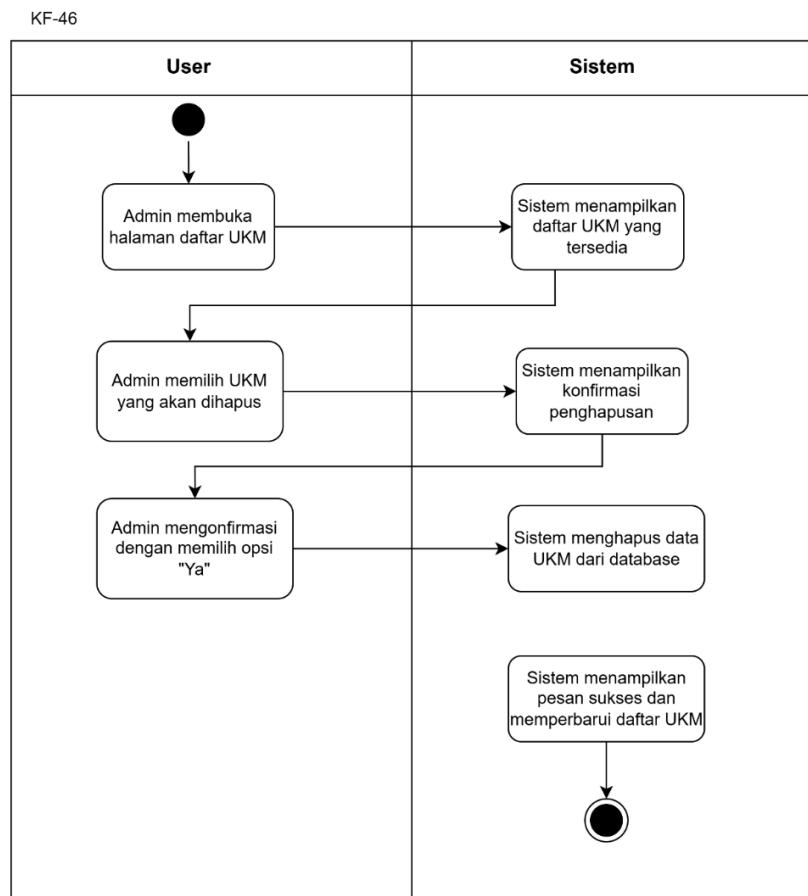
45. *Activity Diagram Edit Data Unit Kegiatan Mahasiswa (aktor admin).*

KF-45



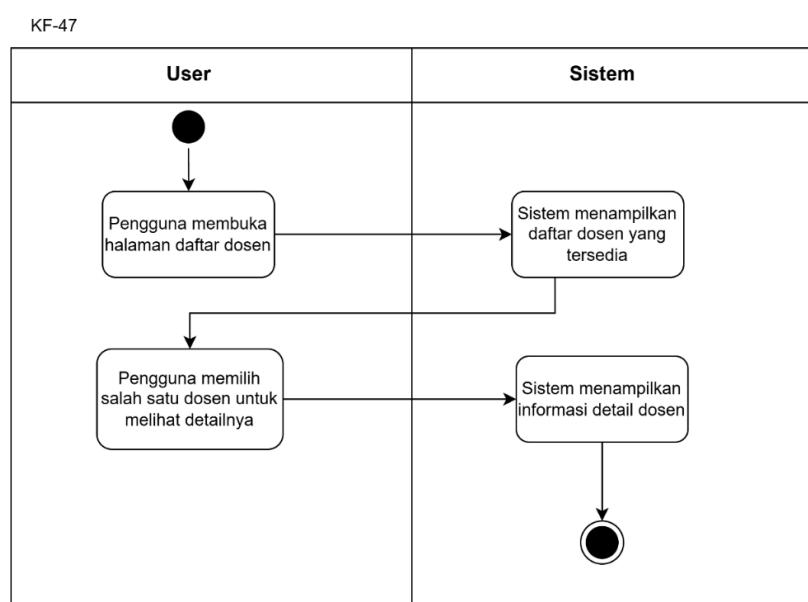
Gambar 3. 51 *Activity Diagram* Edit Data Unit Kegiatan Mahasiswa untuk aktor admin

46. *Activity Diagram Hapus Data Unit Kegiatan Mahasiswa (aktor admin).*



Gambar 3. 52 *Activity Diagram Hapus Data Unit Kegiatan Mahasiswa untuk aktor admin*

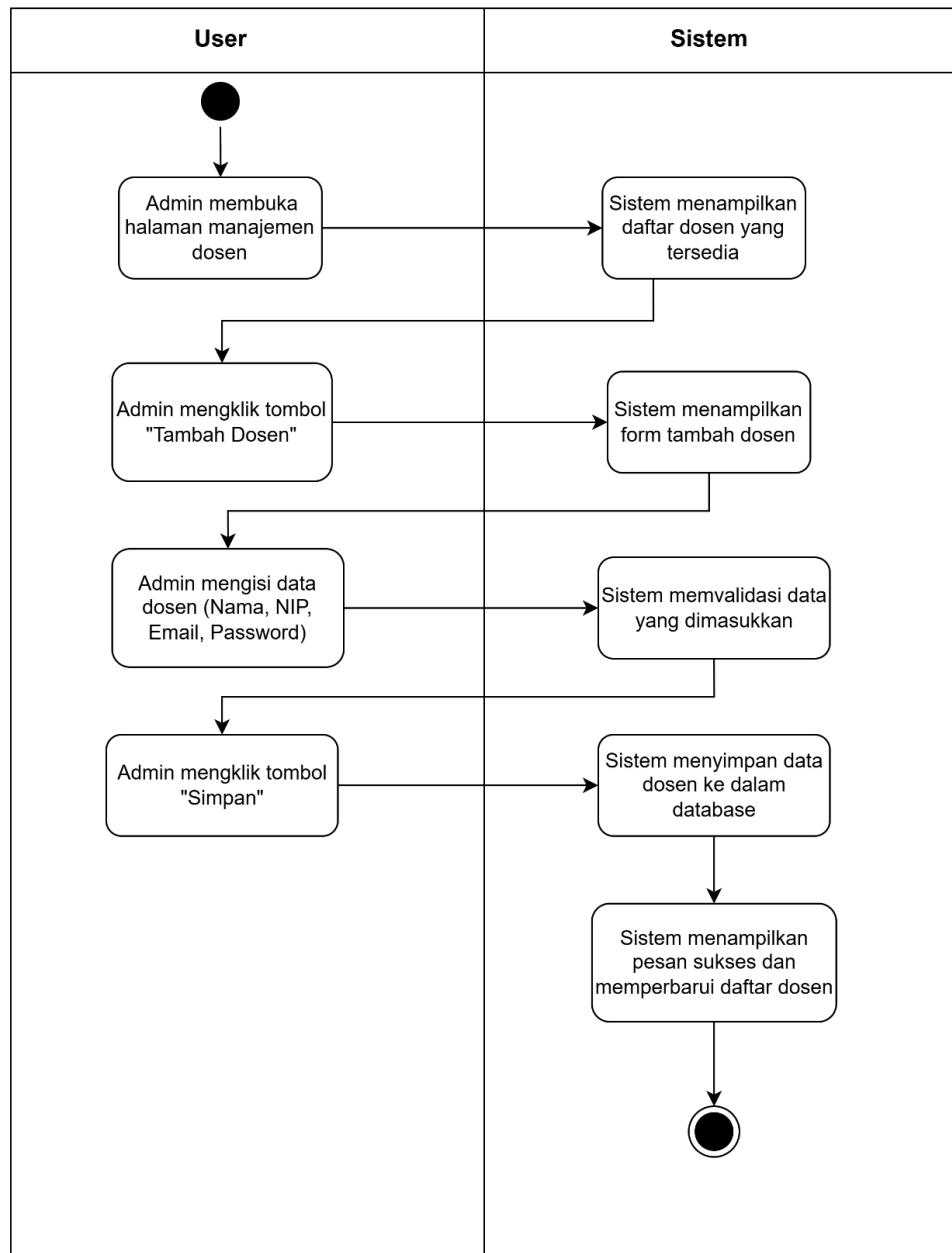
47. *Activity Diagram Lihat Daftar Dosen (aktor admin).*



Gambar 3. 53 47. *Activity Diagram Lihat Daftar Dosen untuk aktor admin*

48. *Activity Diagram Tambah Data Dosen (aktor admin).*

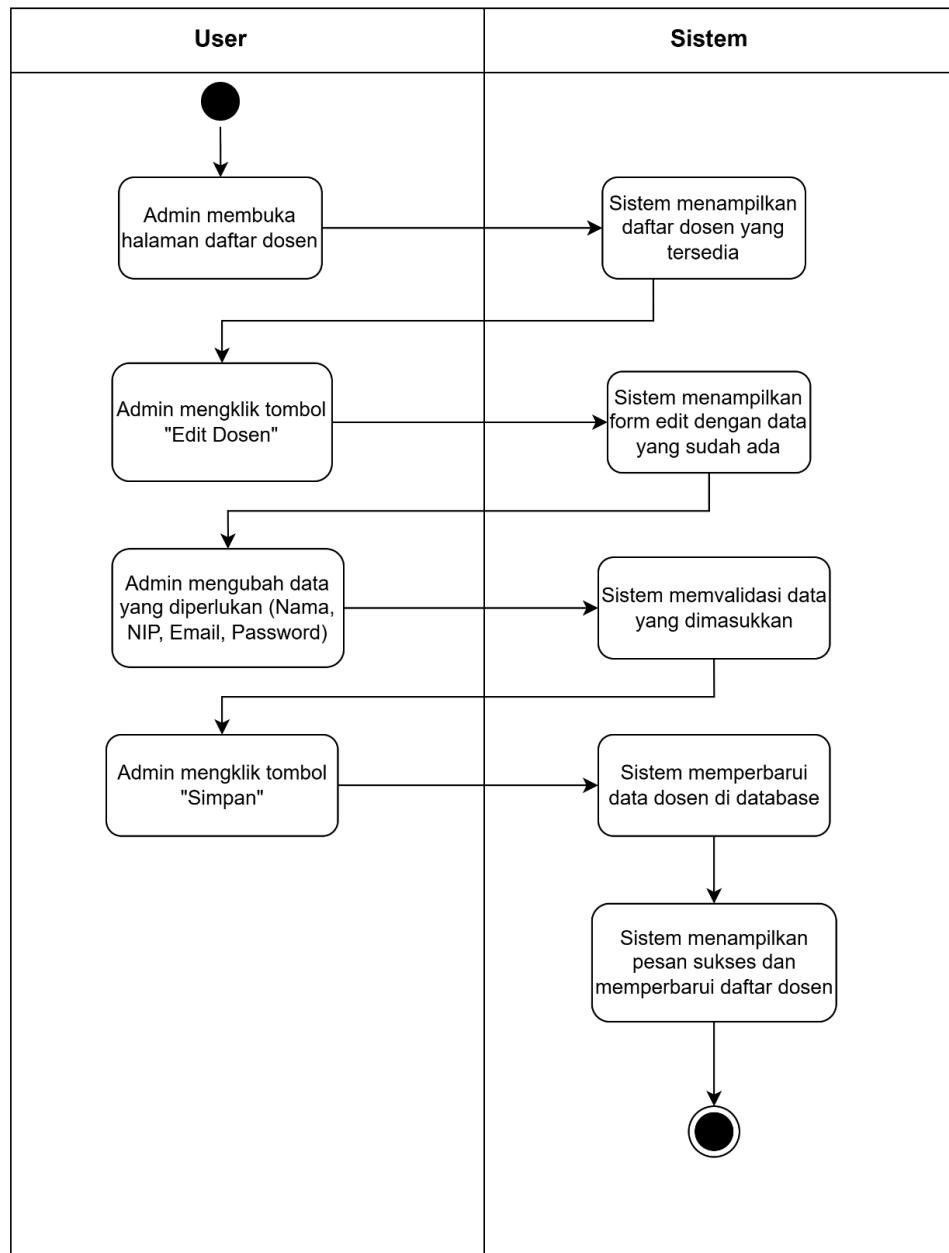
KF-48



Gambar 3. 54 Activity Diagram Tambah Data Dosen untuk aktor admin

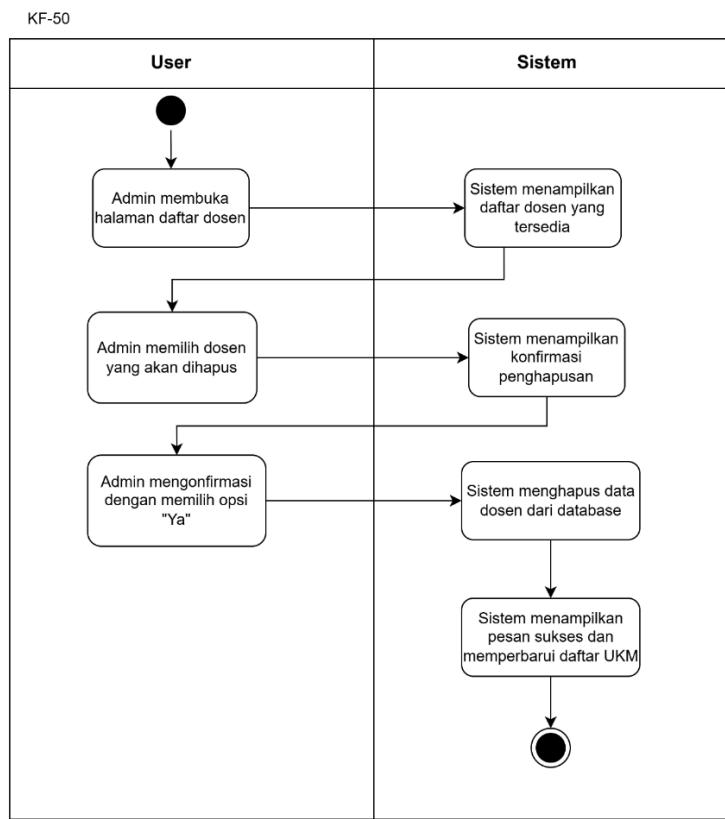
49. *Activity Diagram Edit Data Dosen (aktor admin)*

KF-49



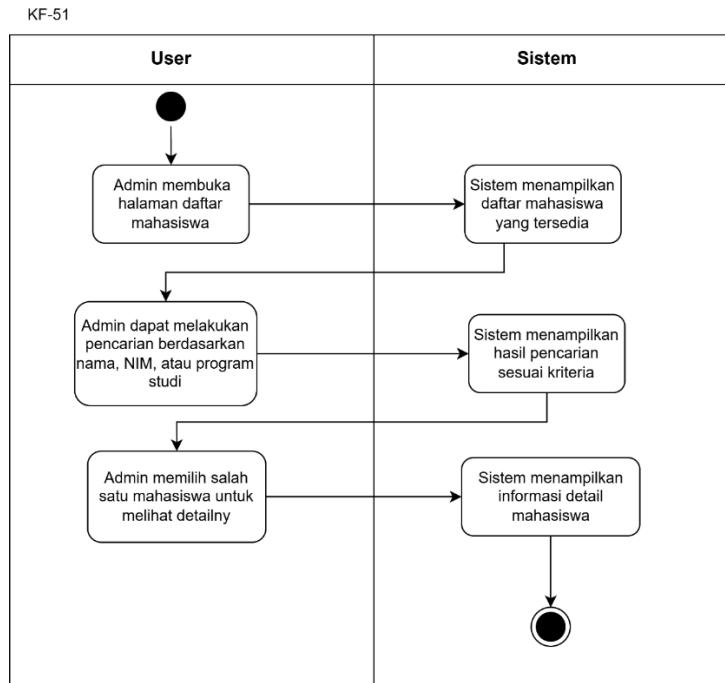
Gambar 3. 55 *Activity Diagram Edit Data Dosen untuk aktor admin*

50. *Activity Diagram Hapus Data Dosen* (aktor admin).



Gambar 3. 56 Activity Diagram Hapus Data Dosen untuk aktor admin

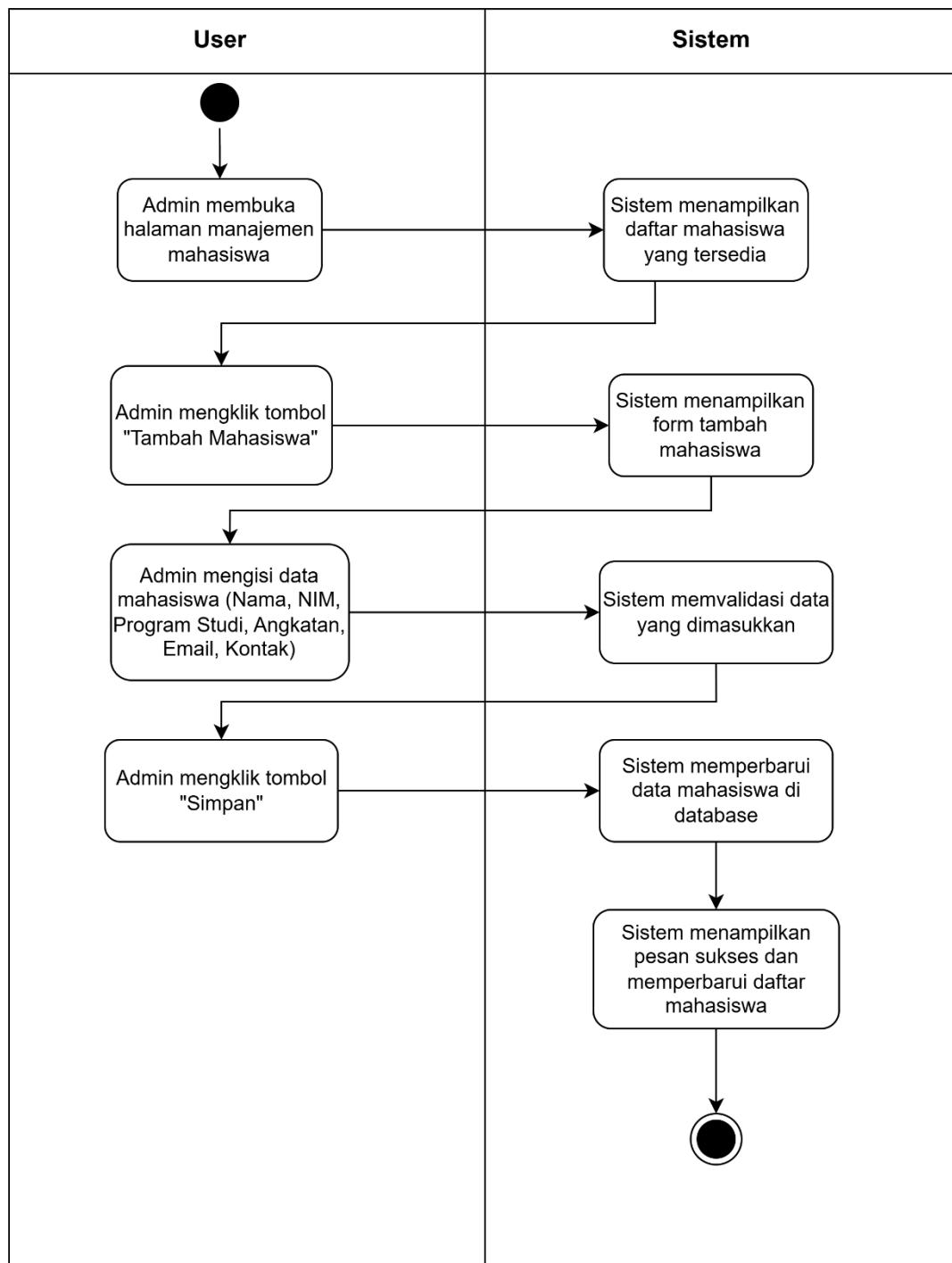
51. *Activity Diagram Lihat Daftar Mahasiswa* (aktor admin).



Gambar 3. 57 Activity Diagram Lihat Daftar Mahasiswa untuk aktor admin

52. *Activity Diagram* Tambah Data Mahasiswa (aktor admin).

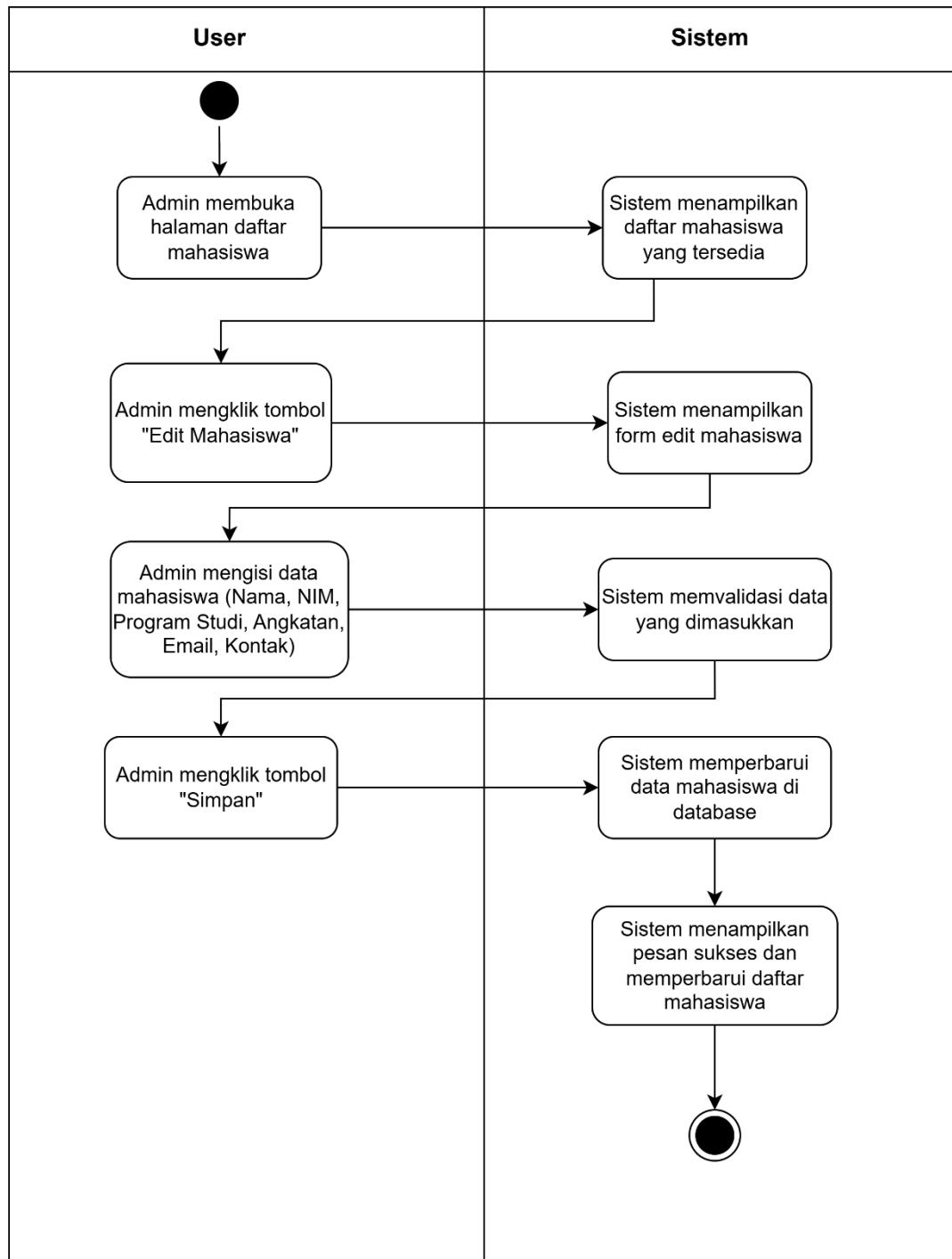
KF-52



Gambar 3. 58 5*Activity Diagram* Tambah Data Mahasiswa untuk aktor admin

53. *Activity Diagram Edit Data Mahasiswa (aktor admin).*

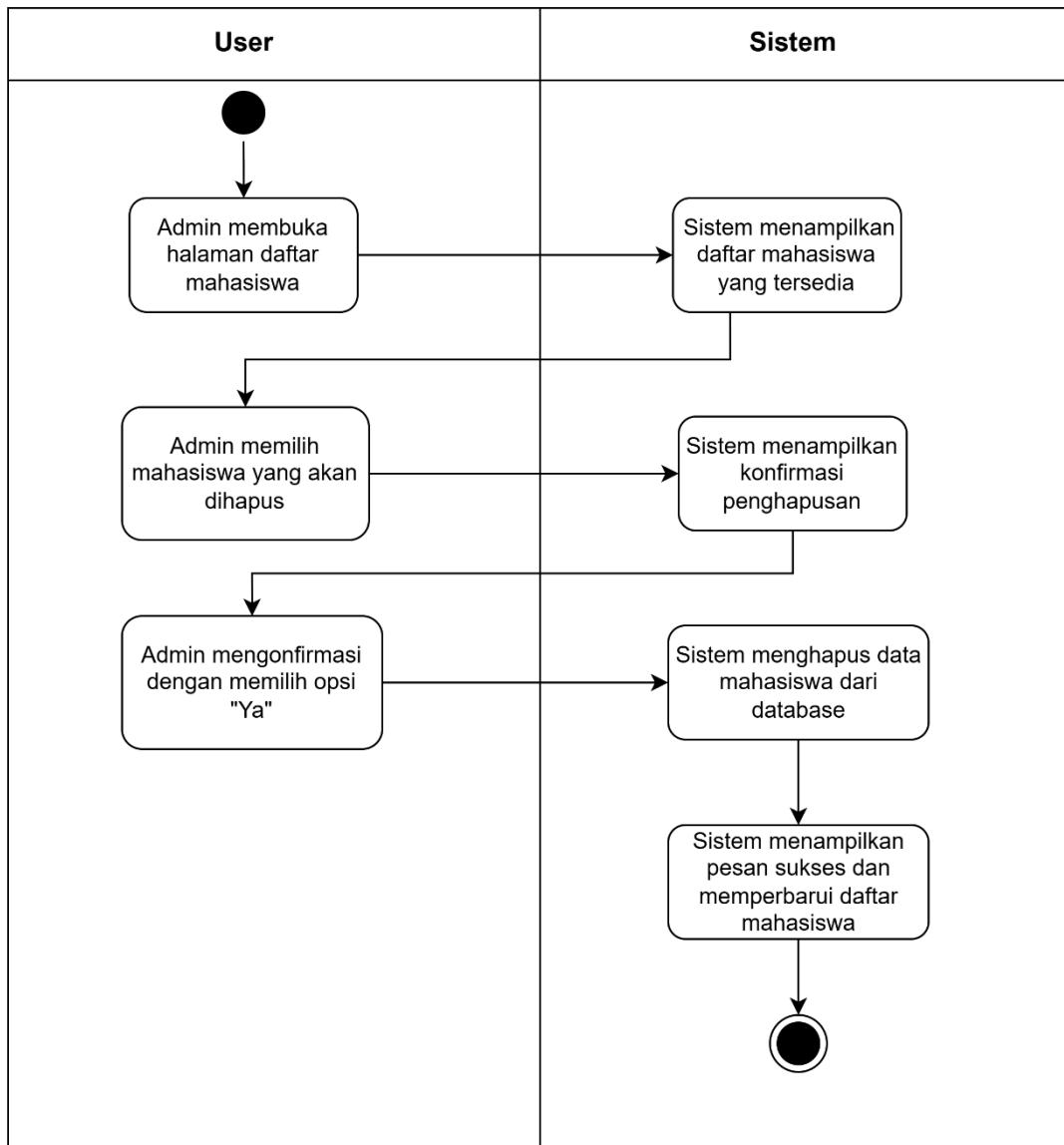
KF-53



Gambar 3. 59 Activity Diagram Edit Data Mahasiswa untuk aktor admin

54. *Activity Diagram Hapus Data Mahasiswa* (aktor admin).

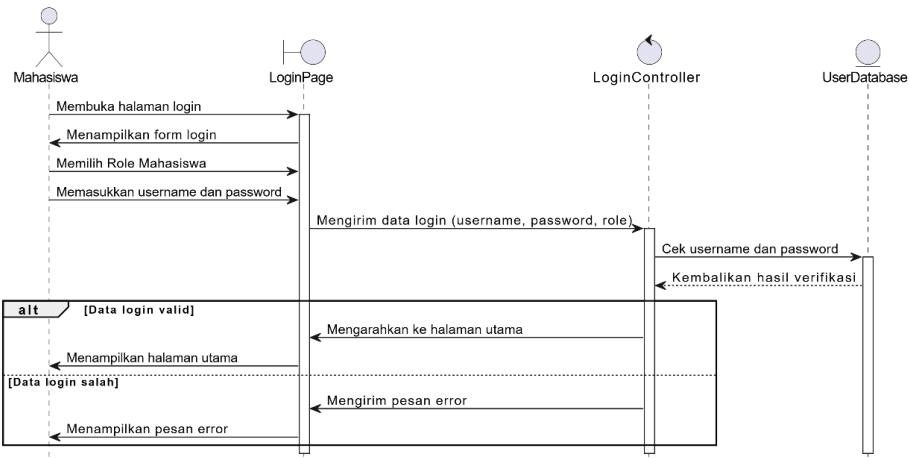
KF-54



Gambar 3. 60 *Activity Diagram Hapus Data Mahasiswa* untuk aktor admin

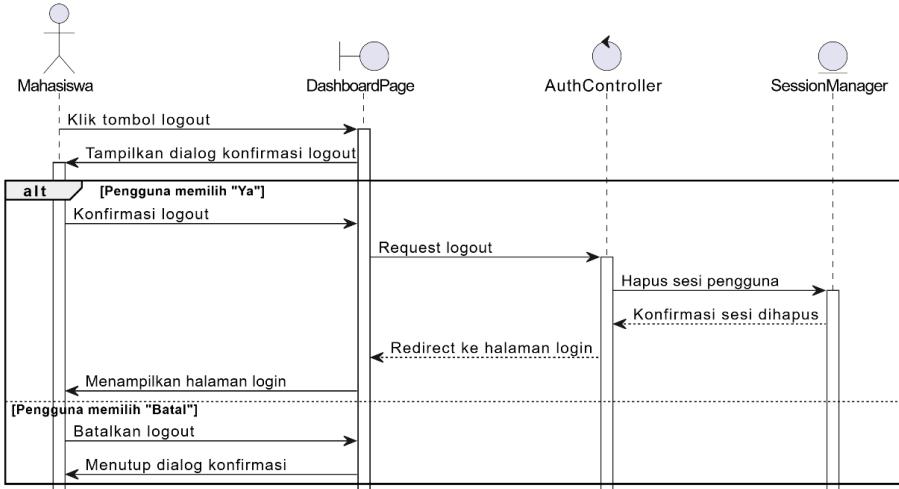
3.3.7. Sequence Diagram

1. Sequence Diagram Login (aktor mahasiswa).



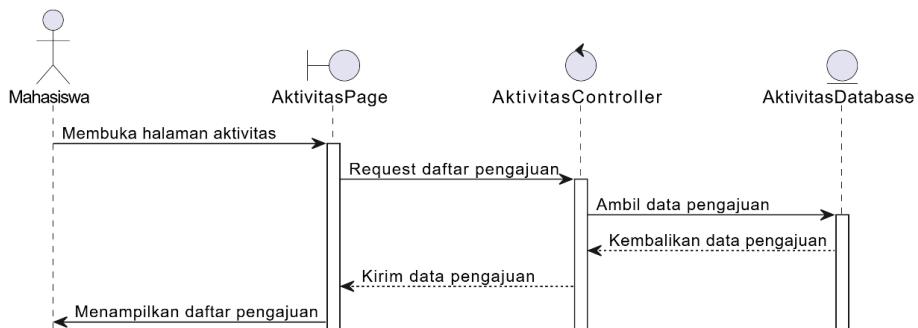
Gambar 3. 61 Sequence Diagram Login untuk aktor mahasiswa

2. Sequence Diagram Logout (aktor mahasiswa).



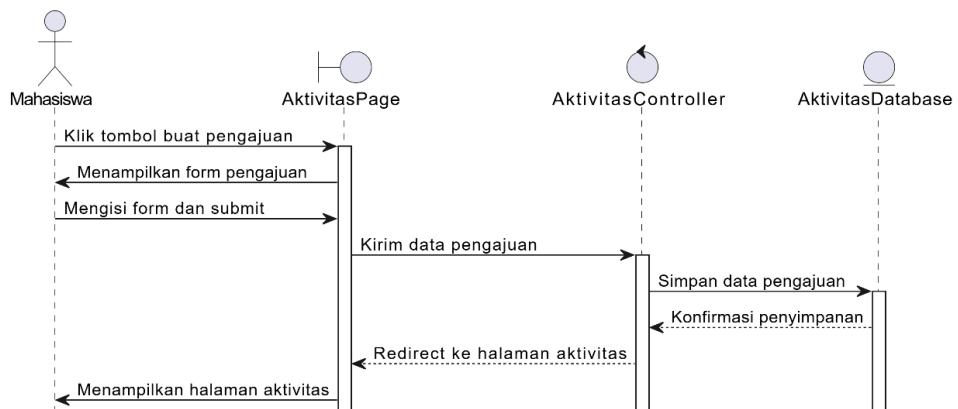
Gambar 3. 62 Sequence Diagram Logout untuk aktor mahasiswa

3. Sequence Diagram Lihat Daftar Pengajuan Kredit Aktivitas (aktor mahasiswa).



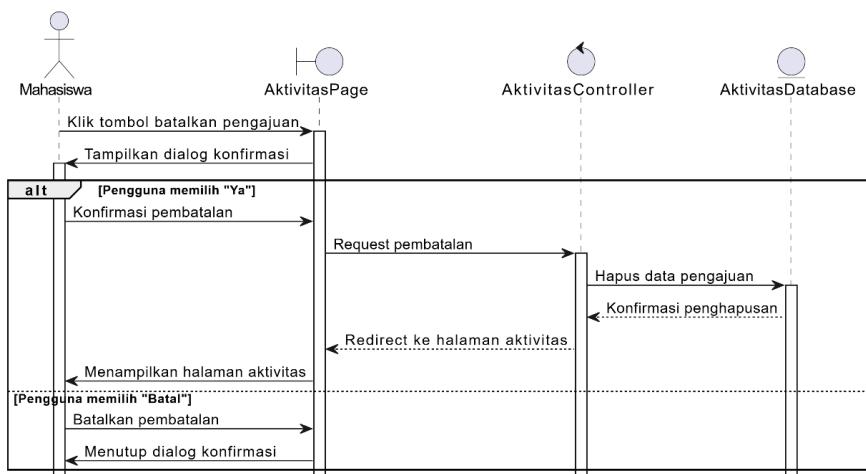
Gambar 3. 63 Sequence Diagram Lihat Daftar Pengajuan Kredit Aktivitas untuk aktor mahasiswa

4. *Squence Diagram* Buat Pengajuan Kredit Aktivitas (aktor mahasiswa).



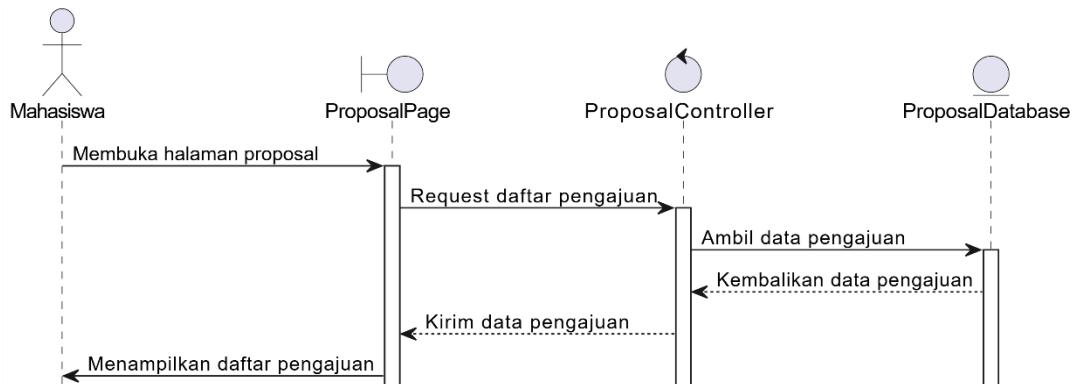
Gambar 3. 64 *Squence Diagram* Buat Pengajuan Kredit Aktivitas untuk aktor mahasiswa

5. *Squence Diagram* Membatalkan Pengajuan Kredit Aktivitas (aktor mahasiswa).



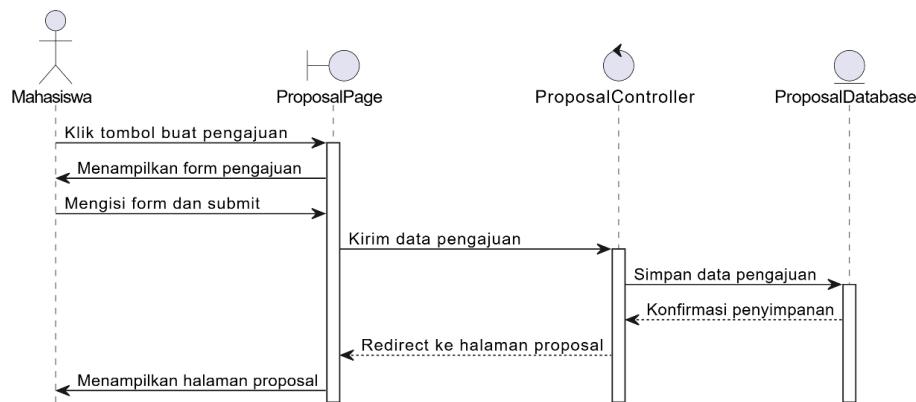
Gambar 3. 65 *Squence Diagram* Membatalkan Pengajuan Kredit Aktivitas untuk aktor mahasiswa

6. *Squence Diagram* Lihat Daftar Pengajuan Proposal (aktor mahasiswa).



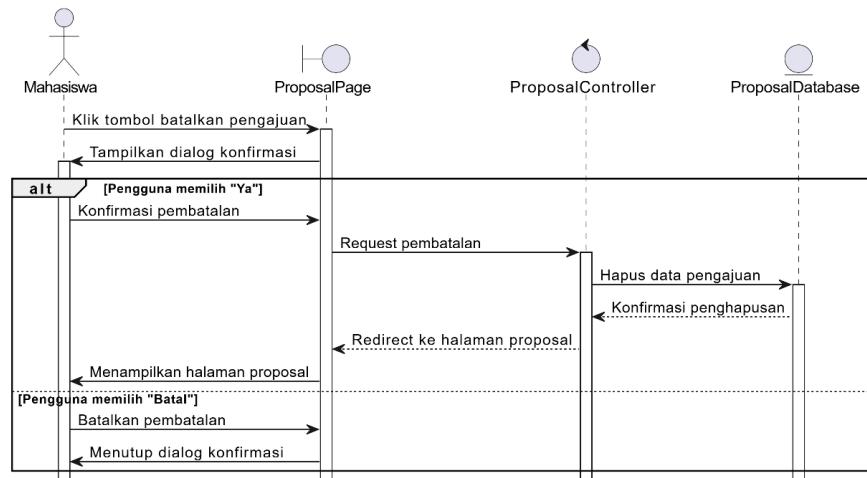
Gambar 3. 66 *Squence Diagram* Lihat Daftar Pengajuan Proposal untuk aktor mahasiswa

7. *Squence Diagram* Buat Pengajuan Proposal (aktor mahasiswa).



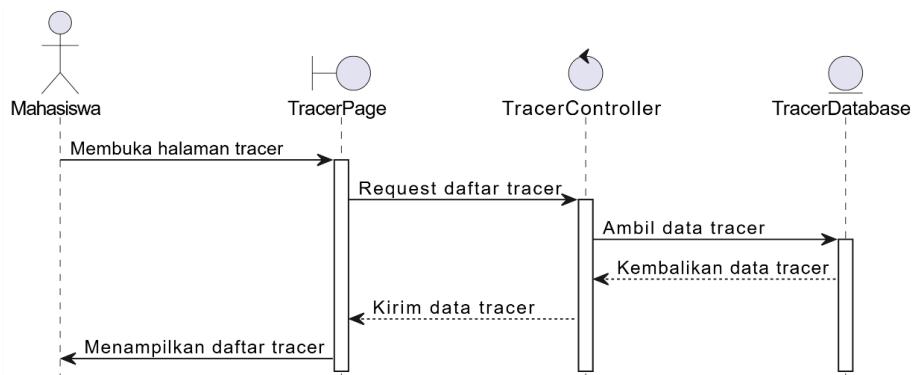
Gambar 3. 67 Squence Diagram Buat Pengajuan Proposal untuk aktor mahasiswa

8. *Squence Diagram* Membatalkan Pengajuan Proposal (aktor mahasiswa).



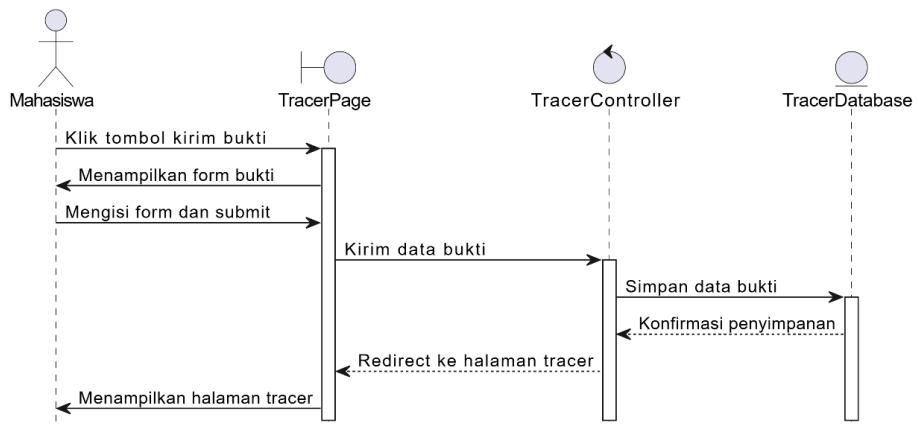
Gambar 3. 68 Squence Diagram Membatalkan Pengajuan Proposal untuk aktor mahasiswa

9. *Squence Diagram* Lihat Daftar Tracer Alumni (aktor mahasiswa).



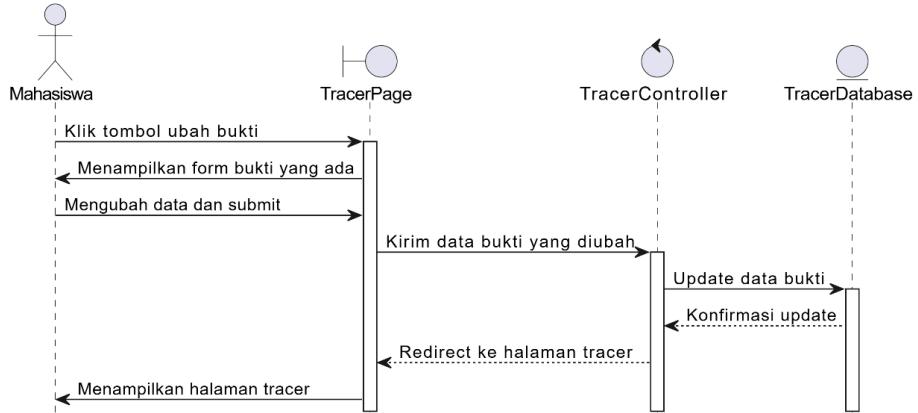
Gambar 3. 69 Squence Diagram Lihat Daftar Tracer Alumni untuk aktor mahasiswa

10. *Squence Diagram* Kirim Bukti Pengisian *Tracer Alumni* (aktor mahasiswa).



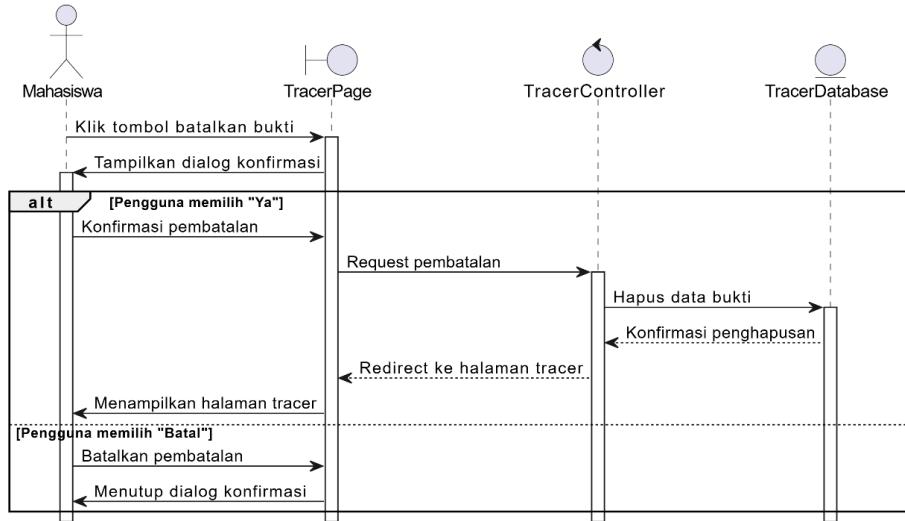
Gambar 3. 70 *Squence Diagram* Kirim Bukti Pengisian *Tracer Alumni* untuk aktor mahasiswa

11. *Squence Diagram* Mengubah Bukti Pengisian *Tracer Alumni* (aktor mahasiswa).



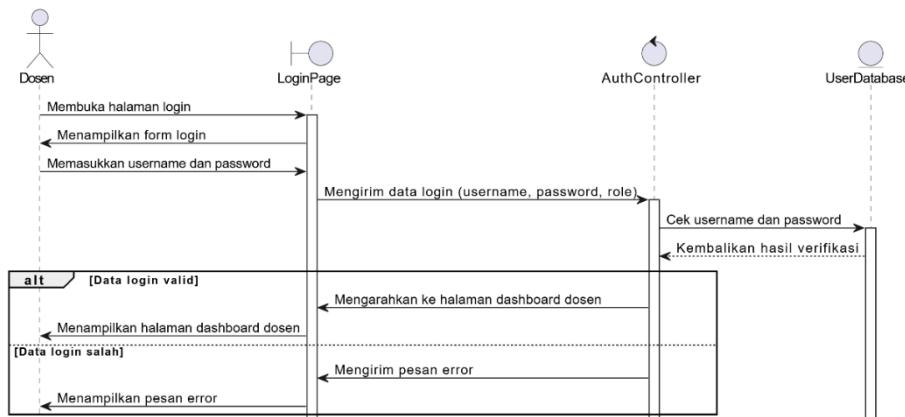
Gambar 3. 71 *Squence Diagram* Mengubah Bukti Pengisian *Tracer Alumni* untuk aktor mahasiswa

12. *Squence Diagram* Membatalkan Bukti Pengisian *Tracer* Alumni (aktor mahasiswa)



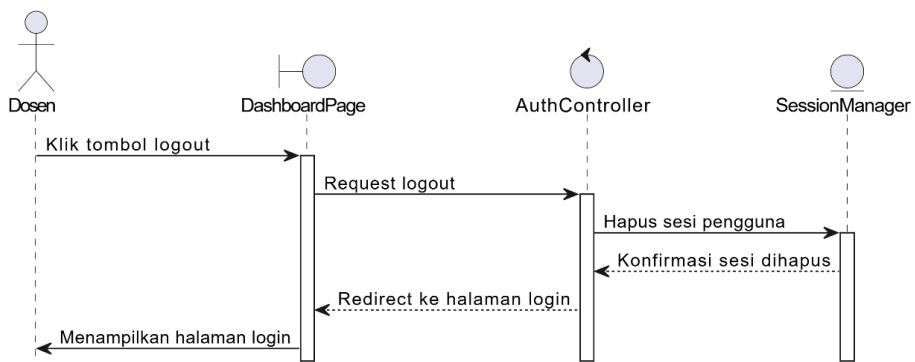
Gambar 3. 72 *Squence Diagram* Membatalkan Bukti Pengisian *Tracer* Alumni aktor mahasiswa

13. *Squence Diagram Login* (aktor dosen).



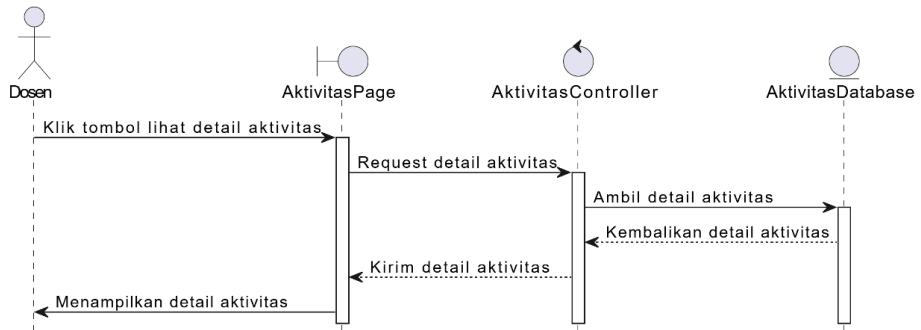
Gambar 3. 73 *Squence Diagram Login* untuk aktor dosen

14. *Squence Diagram Logout* (aktor dosen).



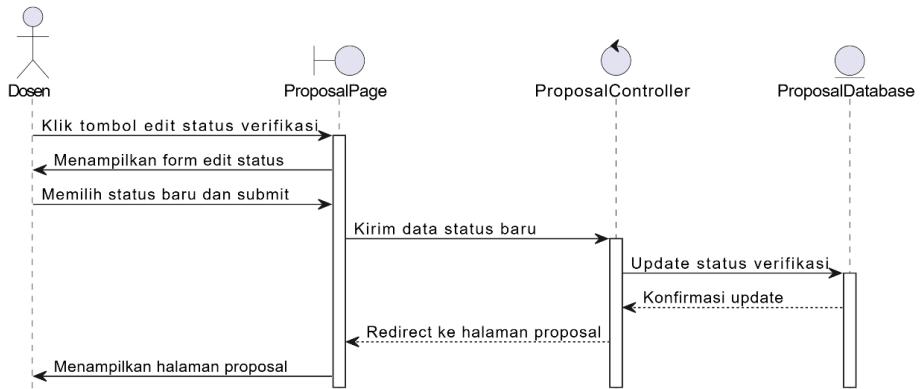
Gambar 3. 74 *Squence Diagram Logout* untuk aktor dosen

15. *Squence Diagram* Lihat Detail Bukti Aktivitas Mahasiswa (aktor dosen).



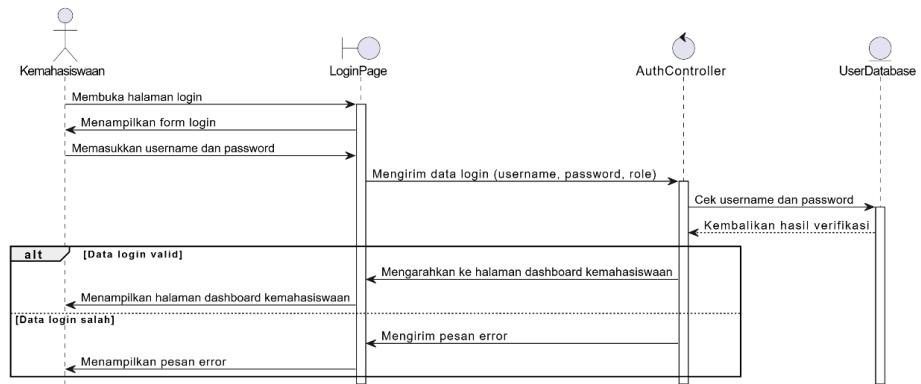
Gambar 3. 75 *Squence Diagram* Lihat Detail Bukti Aktivitas Mahasiswa untuk aktor dosen

16. *Squence Diagram* Edit Status Verifikasi Pengajuan Proposal (aktor dosen).



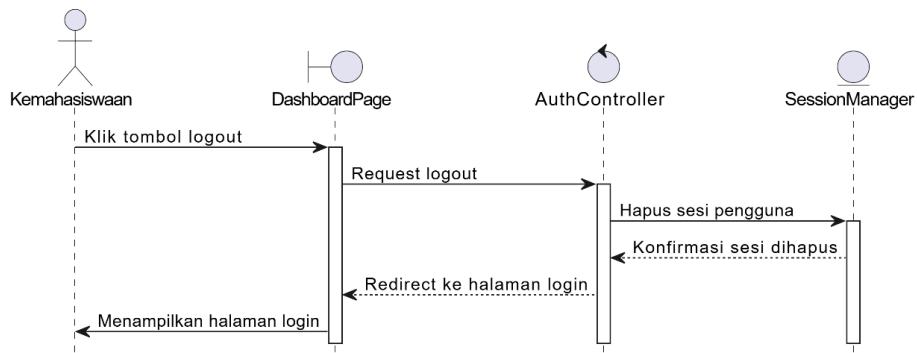
Gambar 3. 76 *Squence Diagram* Edit Status Verifikasi Pengajuan Proposal untuk aktor dosen

17. *Squence Diagram* Login (aktor kemahasiswaan).



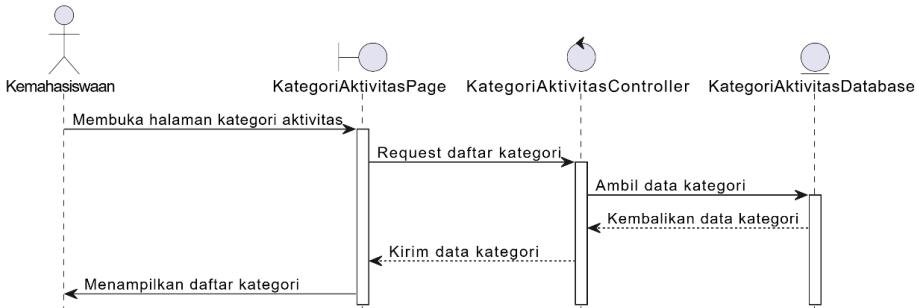
Gambar 3. 77 *Squence Diagram* Login untuk aktor kemahasiswaan

18. *Squence Diagram Logout* (aktor kemahasiswaan).



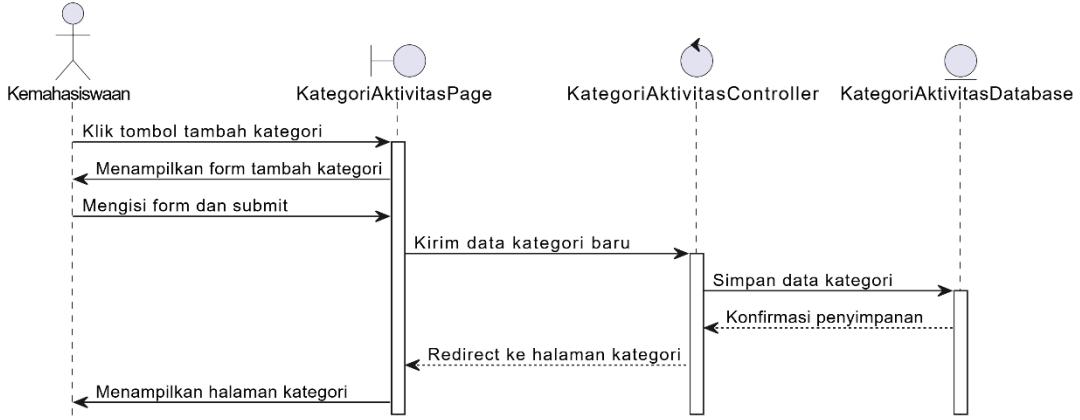
Gambar 3. 78 Squence Diagram Logout untuk aktor kemahasiswaan

19. *Squence Diagram Lihat Daftar Kategori Aktivitas* (aktor kemahasiswaan).



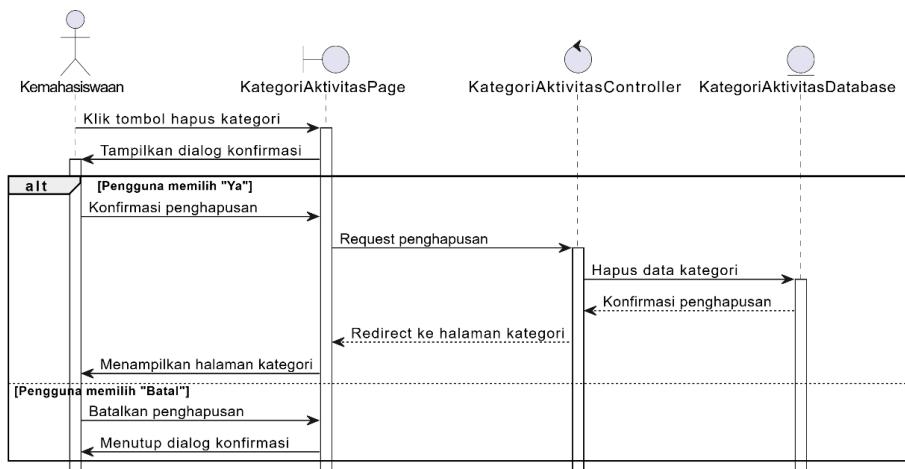
Gambar 3. 79 Squence Diagram Lihat Daftar Kategori Aktivitas untuk aktor kemahasiswaan

20. *Squence Diagram Tambah Data Kategori Aktivitas* (aktor kemahasiswaan).



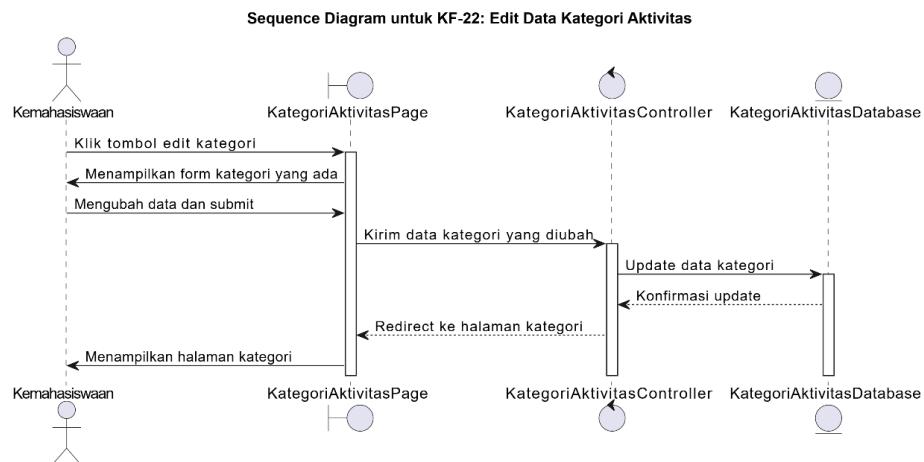
Gambar 3. 80 Squence Diagram Tambah Data Kategori Aktivitas untuk aktor kemahasiswaan

21. *Squence Diagram* Hapus Data Kategori Aktivitas (aktor kemahasiswaan).



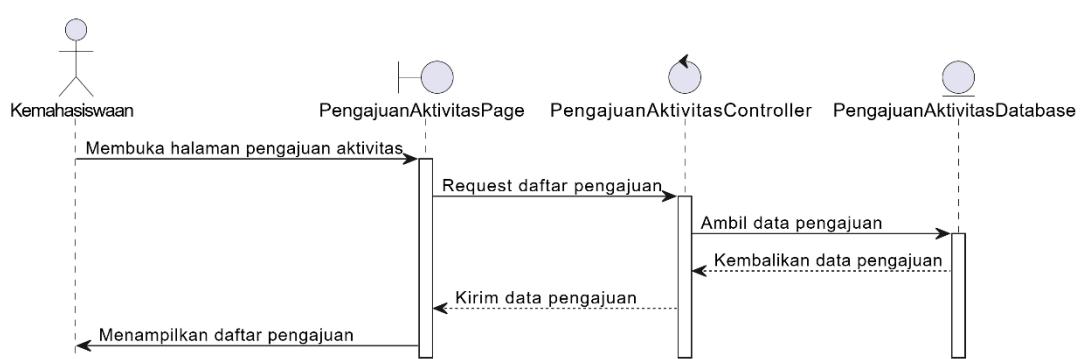
Gambar 3. 81 *Squence Diagram* Hapus Data Kategori Aktivitas untuk aktor kemahasiswaan

22. *Squence Diagram* Edit Data Kategori Aktivitas (aktor kemahasiswaan).



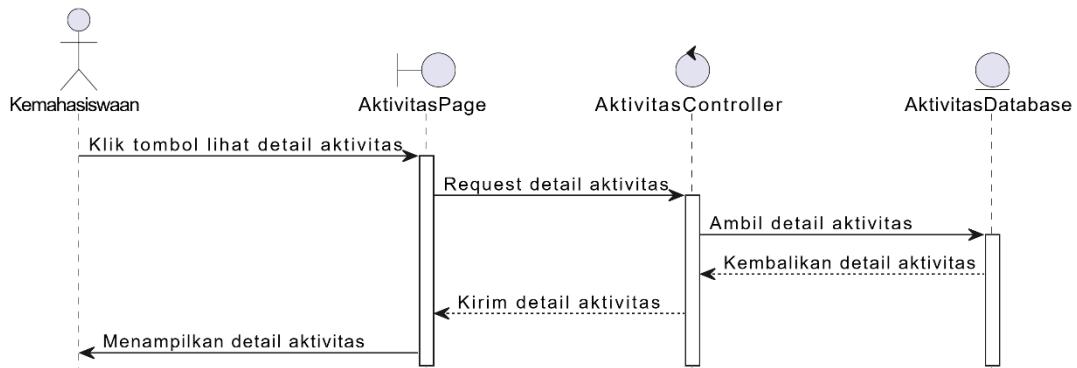
Gambar 3. 82 *Squence Diagram* Edit Data Kategori Aktivitas untuk aktor kemahasiswaan

23. *Squence Diagram* Lihat Daftar Pengajuan Aktivitas Mahasiswa (aktor kemahasiswaan).



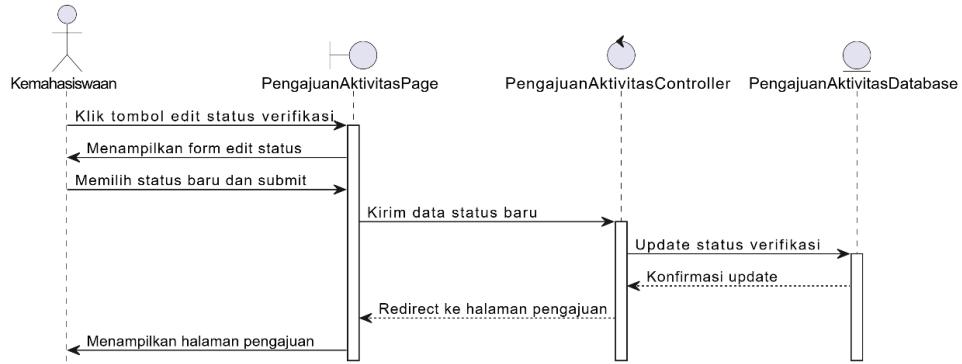
Gambar 3. 83 *Squence Diagram* Lihat Daftar Pengajuan Aktivitas Mahasiswa untuk aktor kemahasiswaan

24. *Squence Diagram* Lihat Detail Bukti Aktivitas Mahasiswa (aktor kemahasiswaan).



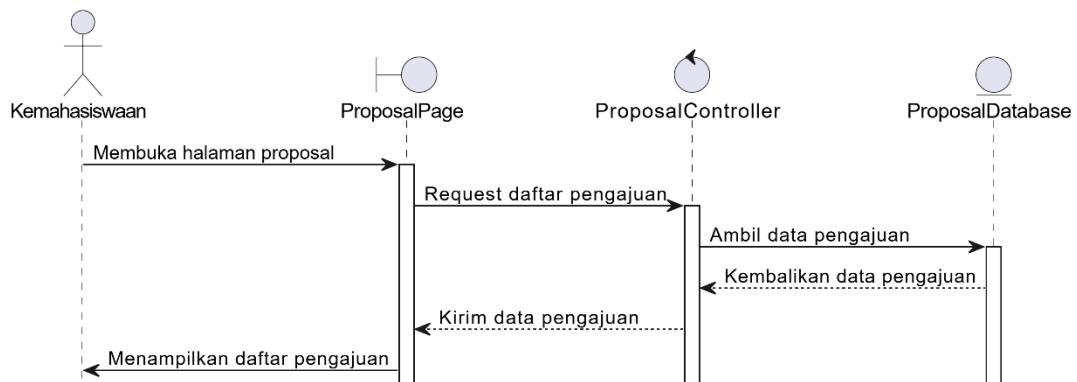
Gambar 3. 84 *Squence Diagram* Lihat Detail Bukti Aktivitas Mahasiswa untuk aktor kemahasiswaan

25. *Squence Diagram* Edit Status Verifikasi Pengajuan Aktivitas (aktor kemahasiswaan).



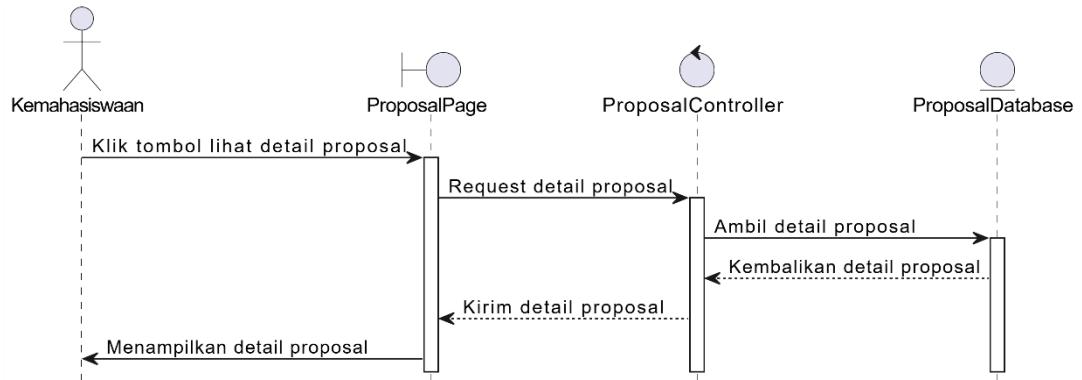
Gambar 3. 85 *Squence Diagram* Edit Status Verifikasi Pengajuan Aktivitas untuk aktor kemahasiswaan

26. *Squence Diagram* Lihat Daftar Pengajuan Proposal (aktor kemahasiswaan).



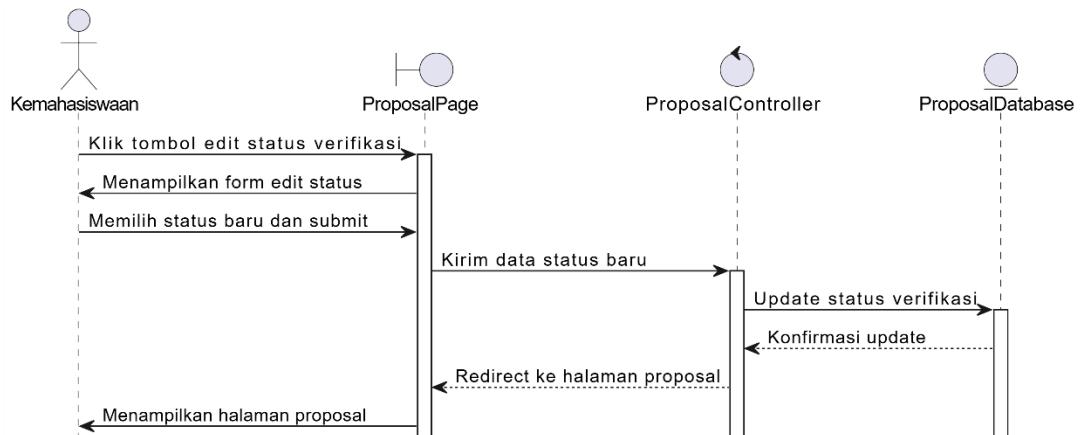
Gambar 3. 86 *Squence Diagram* Lihat Daftar Pengajuan Proposal untuk aktor kemahasiswaan

27. *Squence Diagram* Lihat Detail Bukti pengajuan Proposal (aktor kemahasiswaan).



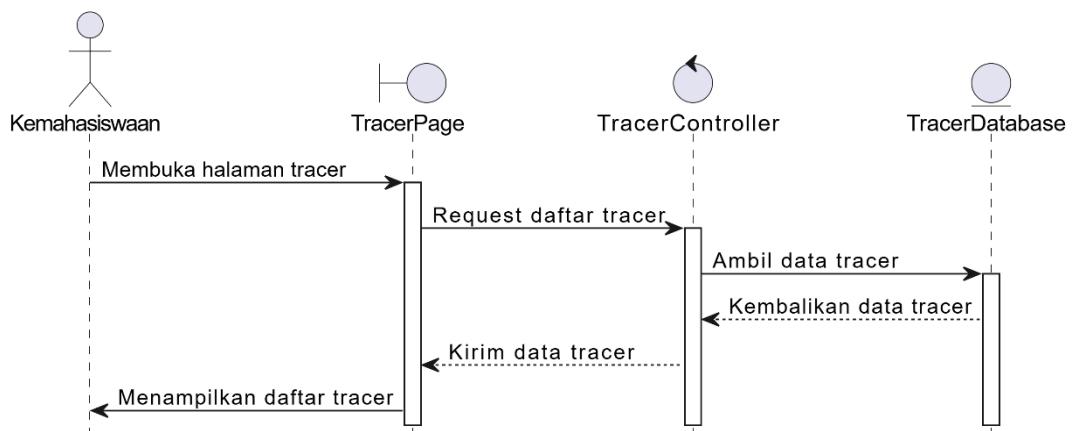
Gambar 3. 87 *Squence Diagram* Lihat Detail Bukti pengajuan Proposal untuk aktor kemahasiswaan

28. *Squence Diagram* Edit Status Verifikasi Pengajuan Proposal (aktor kemahasiswaan).



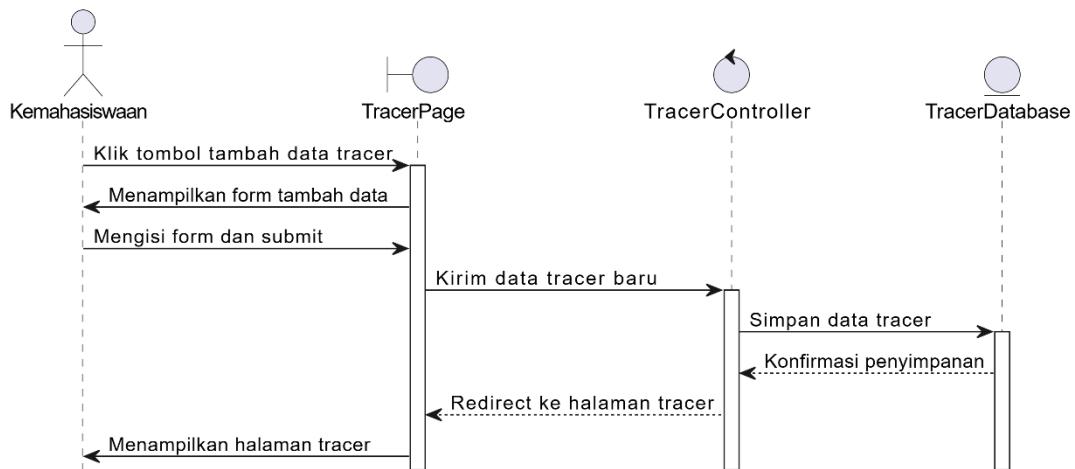
Gambar 3. 88 *Squence Diagram* Edit Status Verifikasi Pengajuan Proposal untuk aktor kemahasiswaan

29. Sequence Diagram Lihat Daftar Tracer Alumni (aktor kemahasiswaan).



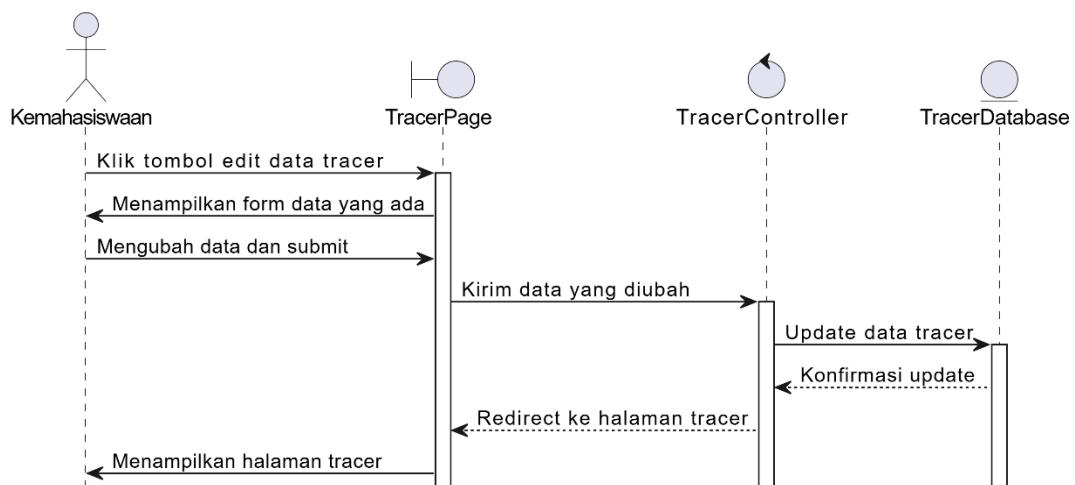
Gambar 3.89 Sequence Diagram Lihat Daftar Tracer Alumni untuk aktor kemahasiswaan

30. Sequence Diagram Tambah Data Tracer Alumni (aktor kemahasiswaan).



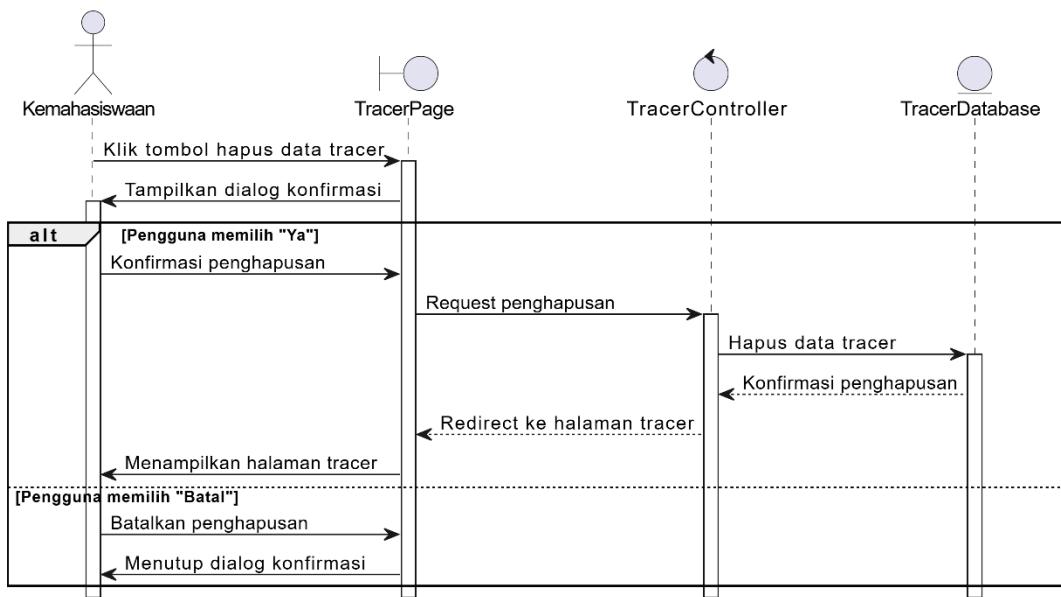
Gambar 3.90 Sequence Diagram Tambah Data Tracer Alumni untuk aktor kemahasiswaan

31. Sequence Diagram Edit Data Tracer Alumni (aktor kemahasiswaan).



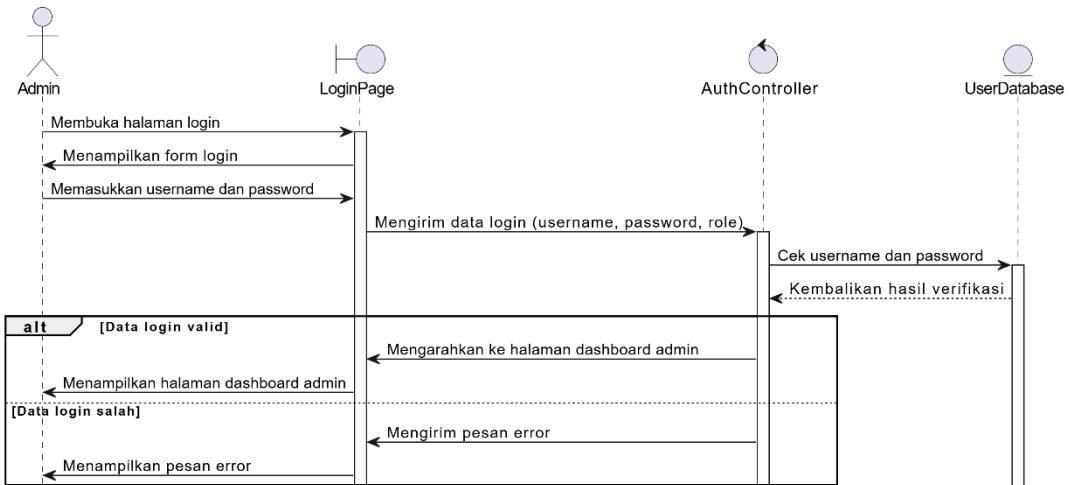
Gambar 3.91 Sequence Diagram Edit Data Tracer Alumni untuk aktor kemahasiswaan

32. Sequence Diagram Hapus Data Tracer Alumni (aktor kemahasiswaan).



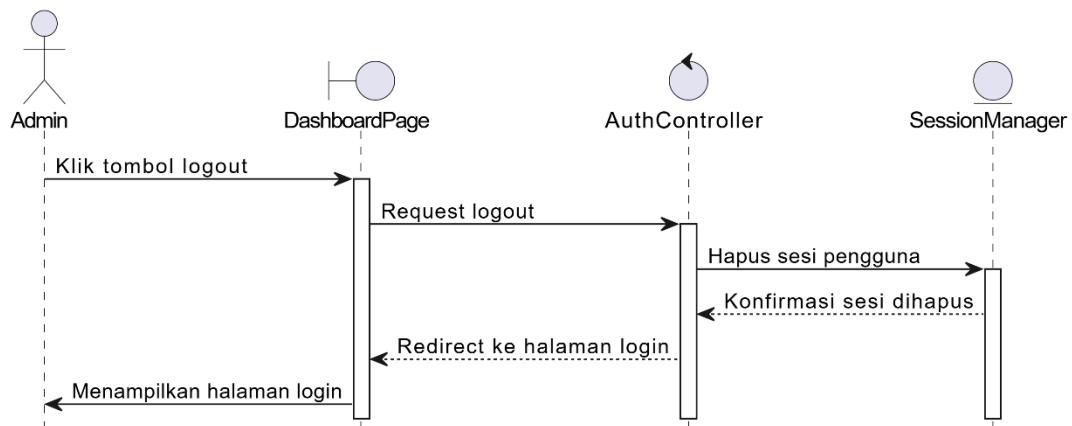
Gambar 3.92 Sequence Diagram Hapus Data Tracer Alumni untuk aktor kemahasiswaan

33. Sequence Diagram Login (aktor admin).



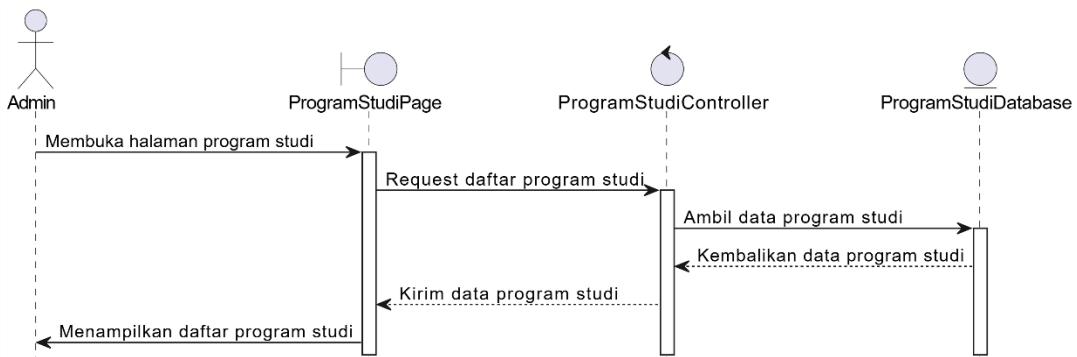
Gambar 3.93 Sequence Diagram Login untuk aktor admin

34. Sequence Diagram Logout (aktor admin).



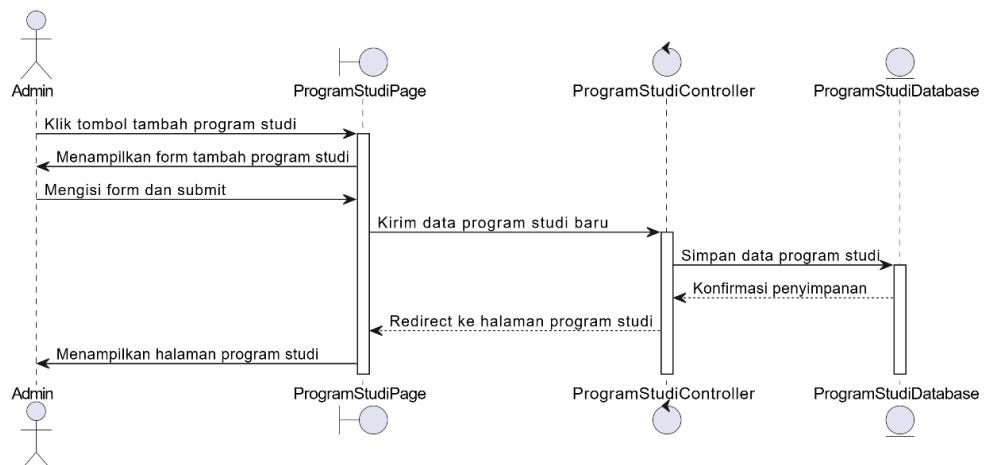
Gambar 3. 94 Squence Diagram Logout aktor admin

35. Sequence Diagram Lihat Daftar Program Studi (aktor admin).



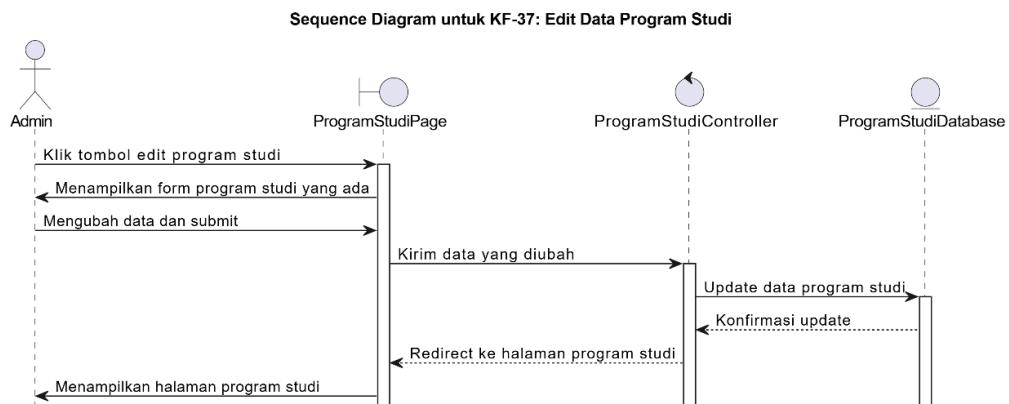
Gambar 3. 95 Squence Diagram Lihat Daftar Program Studi untuk aktor admin

36. Sequence Diagram Tambah Data Daftar Program Studi (aktor admin)



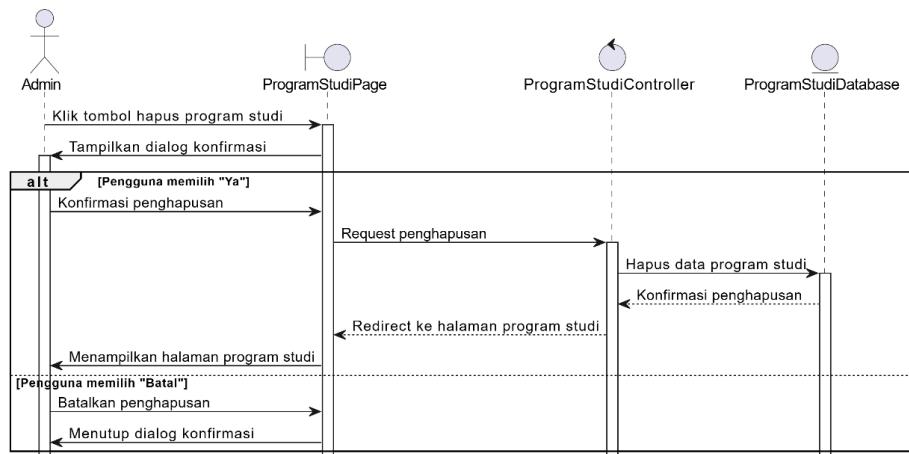
Gambar 3. 96 Squence Diagram Tambah Data Daftar Program Studi untuk aktor admin

37. Sequence Diagram Edit Data Program Studi (aktor admin)



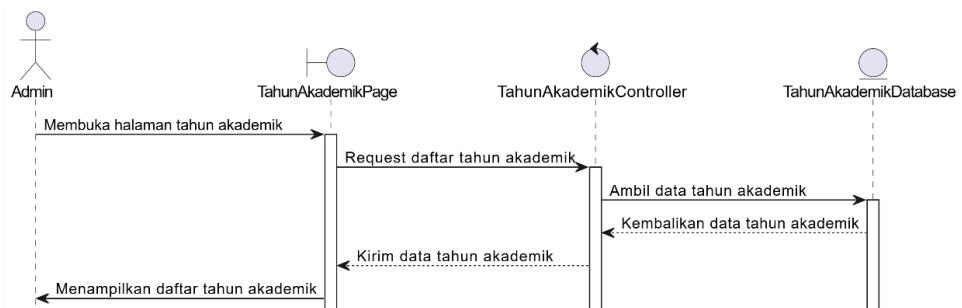
Gambar 3. 97 Sequence Diagram Edit Data Program Studi untuk aktor admin

38. Sequence Diagram Hapus Data Program Studi (aktor admin).



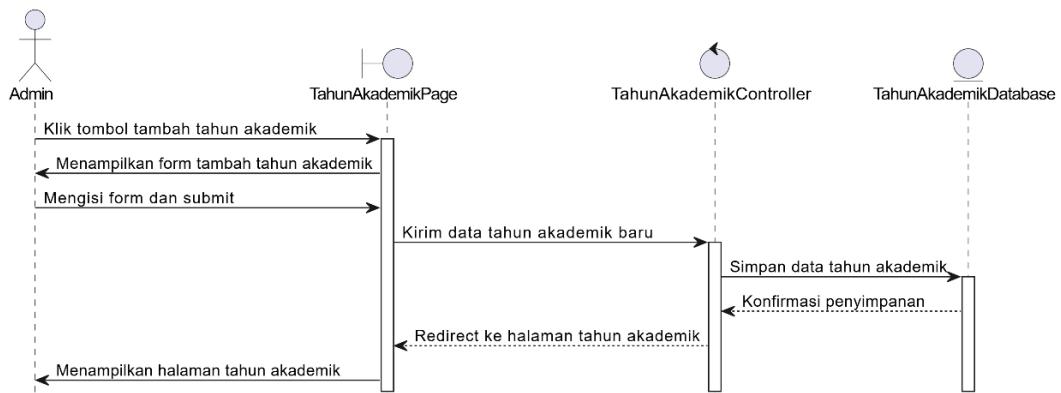
Gambar 3. 98 Squence Diagram Hapus Data Program Studi untuk aktor admin

39. Sequence Diagram Lihat Daftar Tahun Akademik (aktor admin).



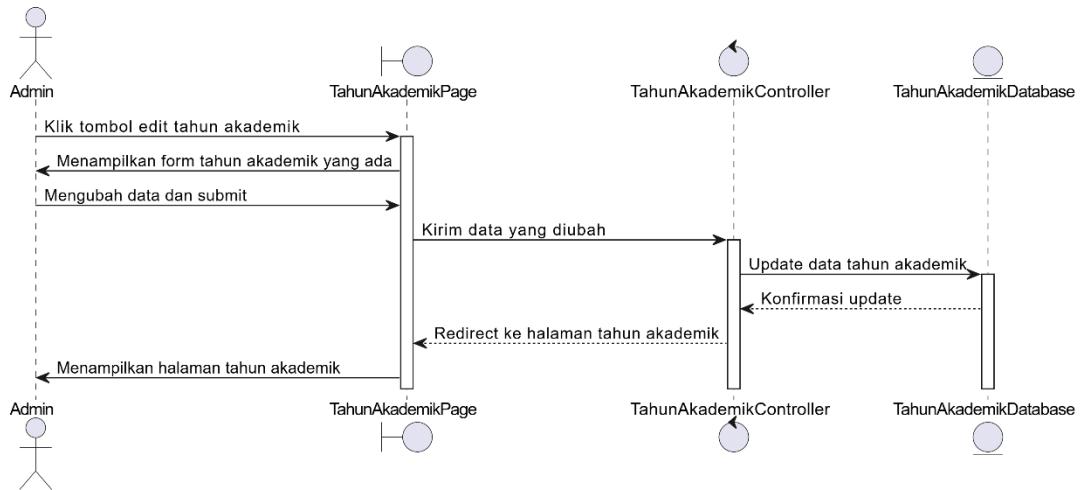
Gambar 3. 99 Squence Diagram Lihat Daftar Tahun Akademik untuk aktor admin

40. *Squence Diagram* Tambah Data Tahun Akademik (aktor admin).



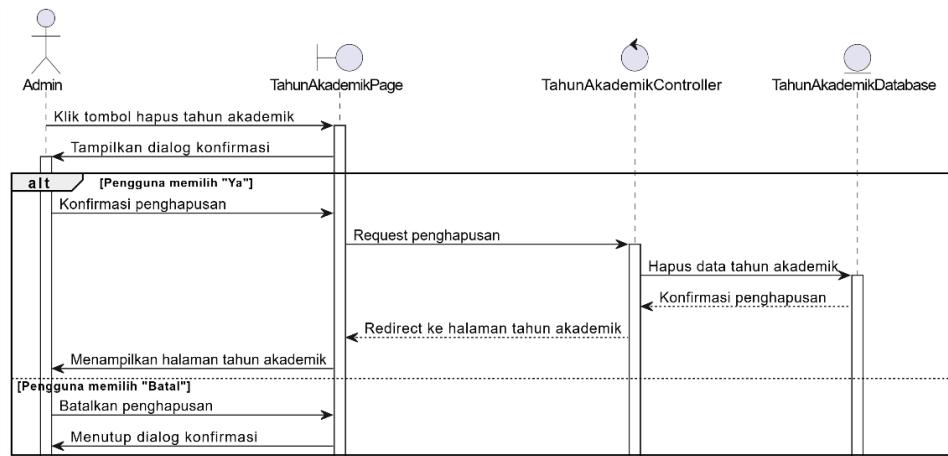
Gambar 3. 100 *Squence Diagram* Tambah Data Tahun Akademik untuk aktor admin

41. *Squence Diagram* Edit Data Tahun Akademik (aktor admin).



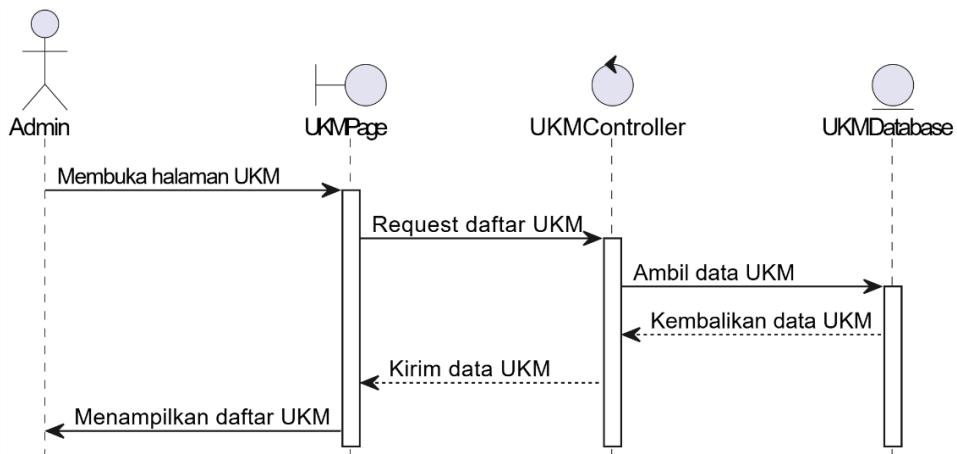
Gambar 3. 101 *Squence Diagram* Edit Data Tahun Akademik untuk aktor admin

42. *Squence Diagram* Hapus Data Tahun Akademik (aktor admin).



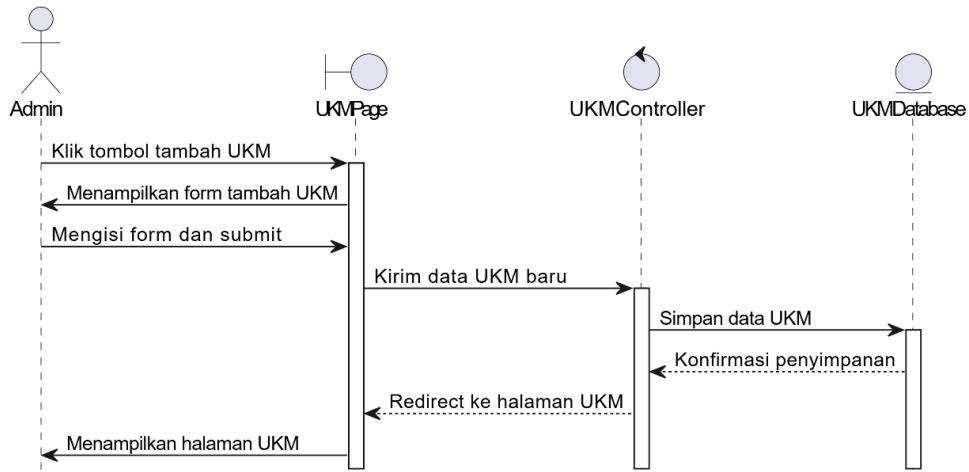
Gambar 3. 102 *Squence Diagram* Hapus Data Tahun Akademik untuk aktor admin

43. *Squence Diagram* Lihat Daftar Unit Kegiatan Mahasiswa (aktor admin)



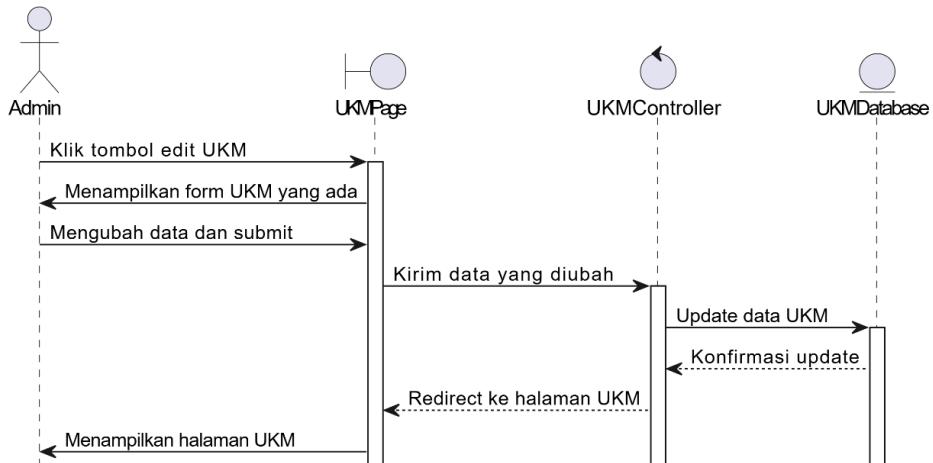
Gambar 3. 103 *Squence Diagram* Lihat Daftar Unit Kegiatan Mahasiswa aktor admin

44. *Squence Diagram* Tambah Data Unit Kegiatan Mahasiswa (aktor admin).



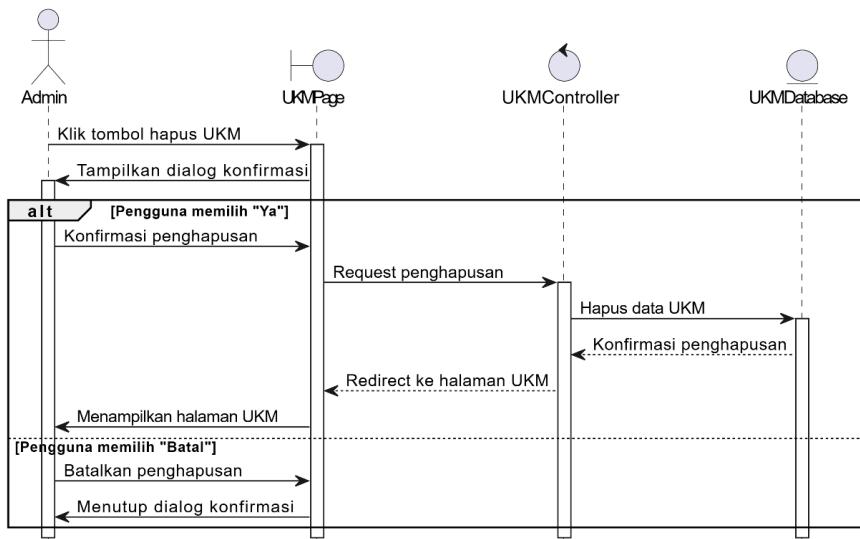
Gambar 3. 104 *Squence Diagram* Tambah Data Unit Kegiatan Mahasiswa untuk aktor admin

45. *Squence Diagram* Edit Data Unit Kegiatan Mahasiswa (aktor admin).



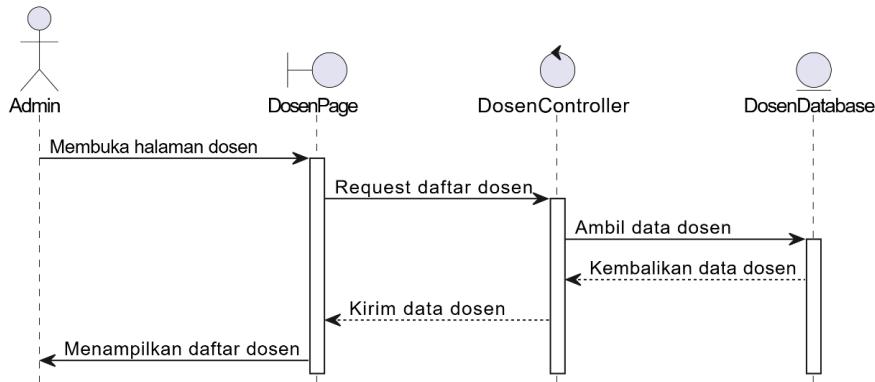
Gambar 3. 105 *Squence Diagram* Edit Data Unit Kegiatan Mahasiswa untuk aktor admin

46. *Squence Diagram* Hapus Data Unit Kegiatan Mahasiswa (aktor admin).



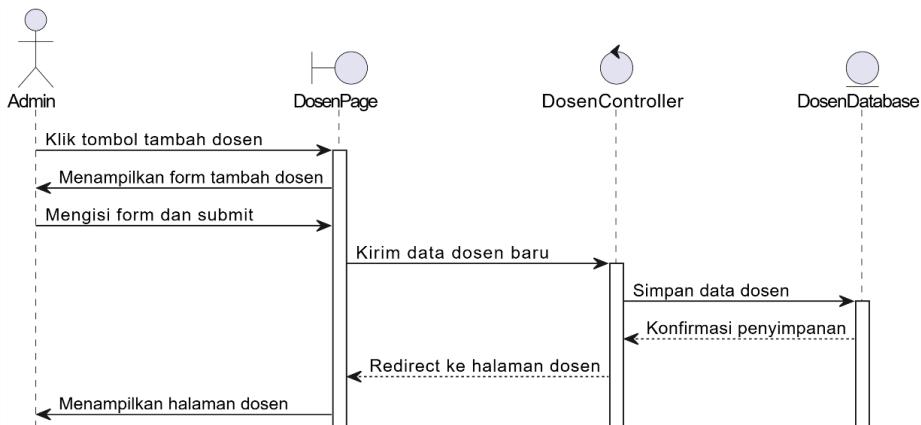
Gambar 3. 106 *Squence Diagram* Hapus Data Unit Kegiatan Mahasiswa untuk aktor admin

47. *Squence Diagram* Lihat Daftar Dosen (aktor admin)



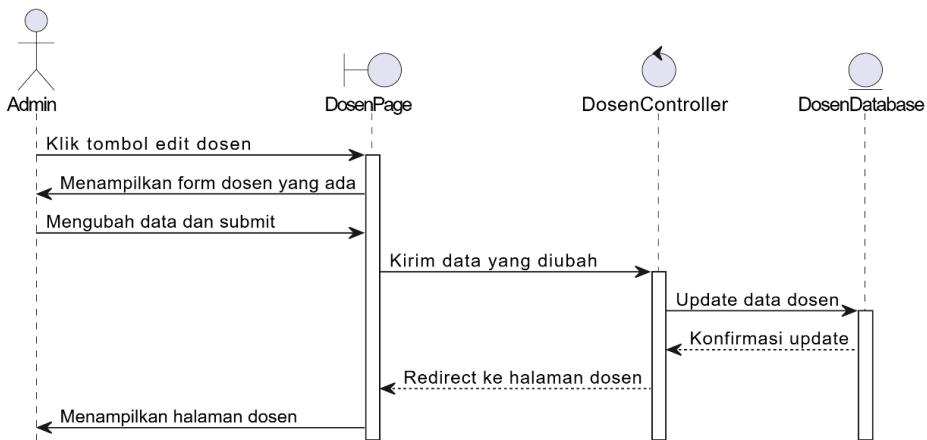
Gambar 3. 107 *Squence Diagram* Lihat Daftar Dosen untuk aktor admin

48. *Squence Diagram* Tambah Data Dosen (aktor admin).



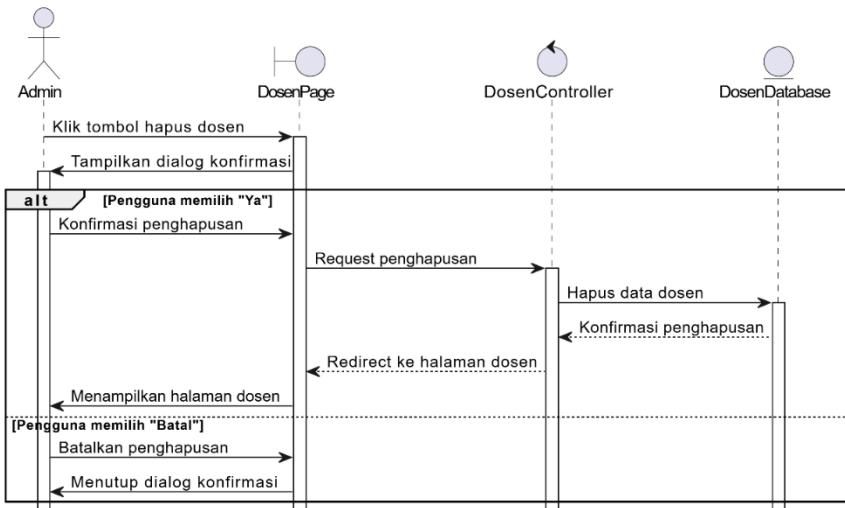
Gambar 3. 108 *Squence Diagram* Tambah Data Dosen untuk aktor admin

49. Sequence Diagram Edit Data Dosen (aktor admin).



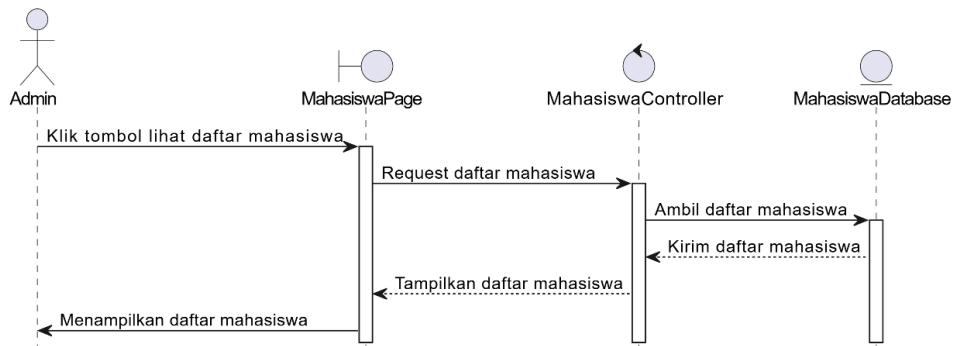
Gambar 3. 109 Sequence Diagram Edit Data Dosen untuk aktor admin

50. Sequence Diagram Hapus Data Dosen (aktor admin).



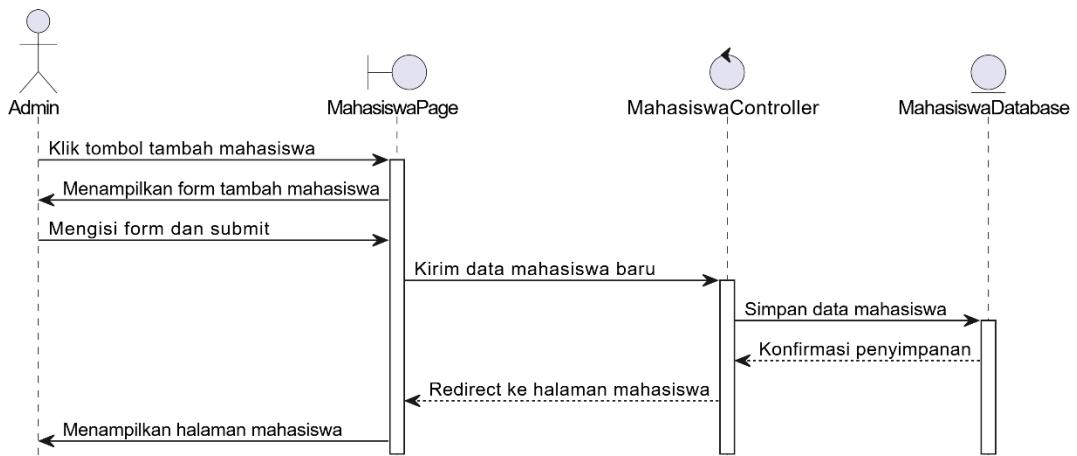
Gambar 3. 110 Sequence Diagram Hapus Data Dosen untuk aktor admin

51. Sequence Diagram Lihat Daftar Mahasiswa (aktor admin)



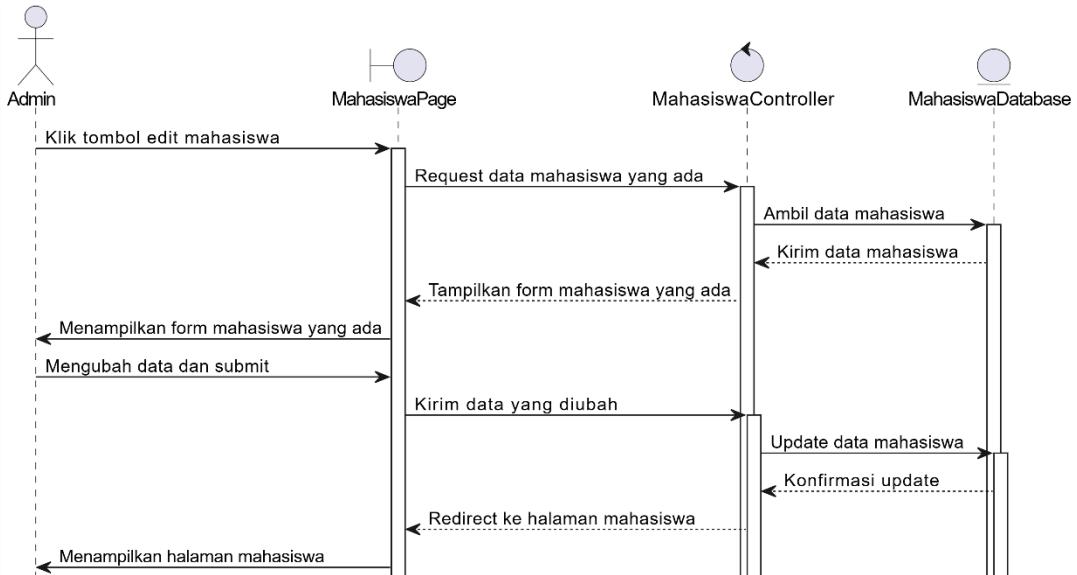
Gambar 3. 111 Sequence Diagram Lihat Daftar Mahasiswa untuk aktor admin

52. *Squence Diagram* Tambah Data Mahasiswa (aktor admin).



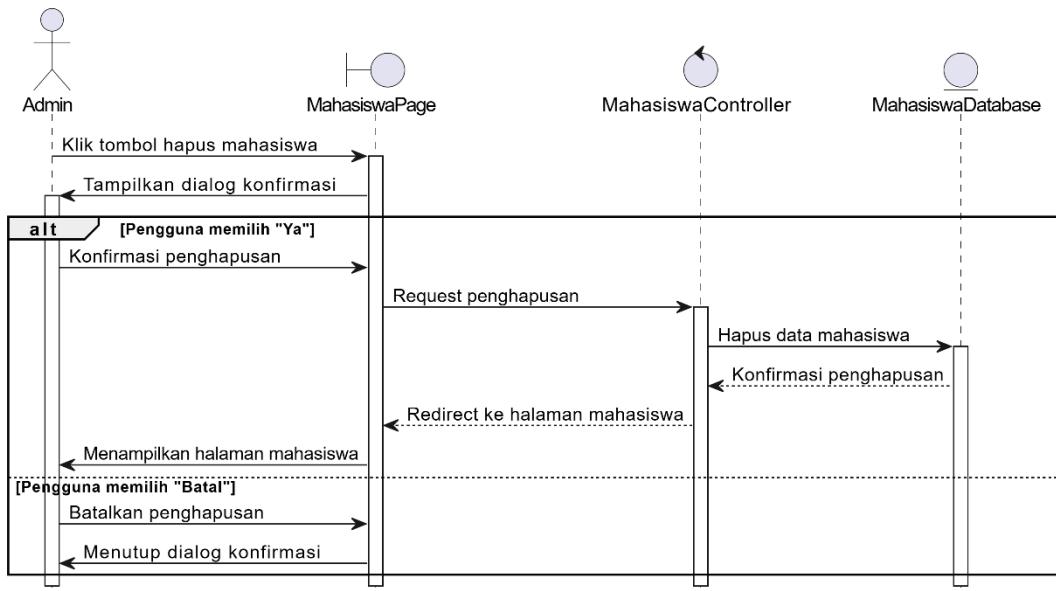
Gambar 3. 112 *Squence Diagram* Tambah Data Mahasiswa untuk aktor admin

53. *Squence Diagram* Edit Data Mahasiswa (aktor admin).



Gambar 3. 113 *Squence Diagram* Edit Data Mahasiswa untuk aktor admin

54. Sequence Diagram Hapus Data Mahasiswa (aktor admin).

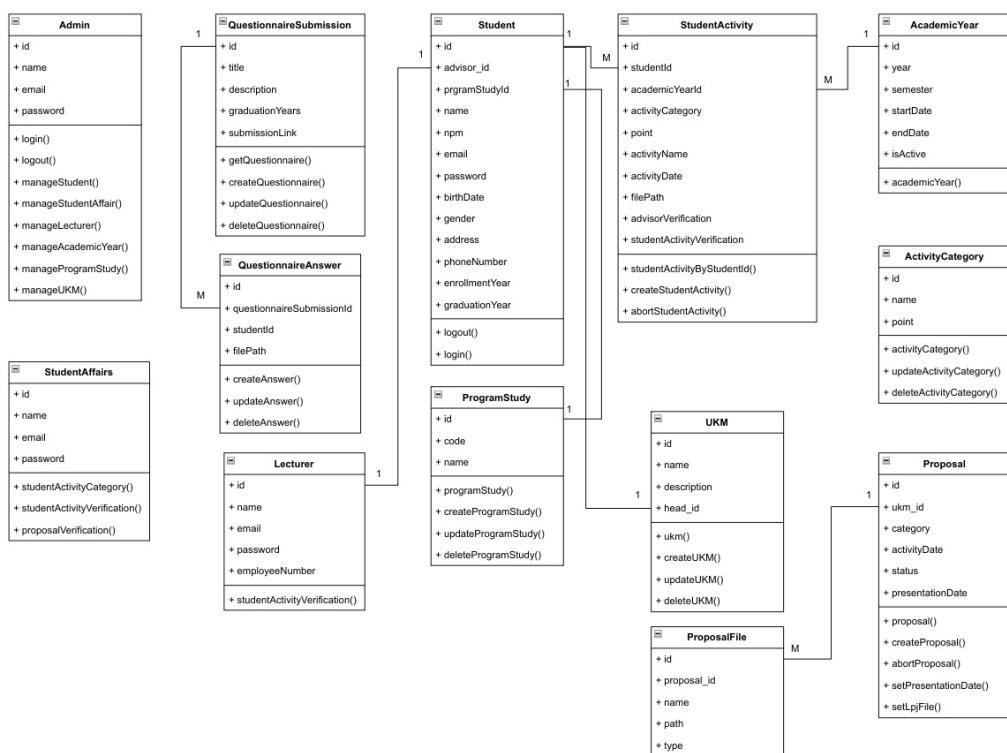


Gambar 3. 114 Squence Diagram Hapus Data Mahasiswa untuk aktor admin

3.3.8. Class Diagram

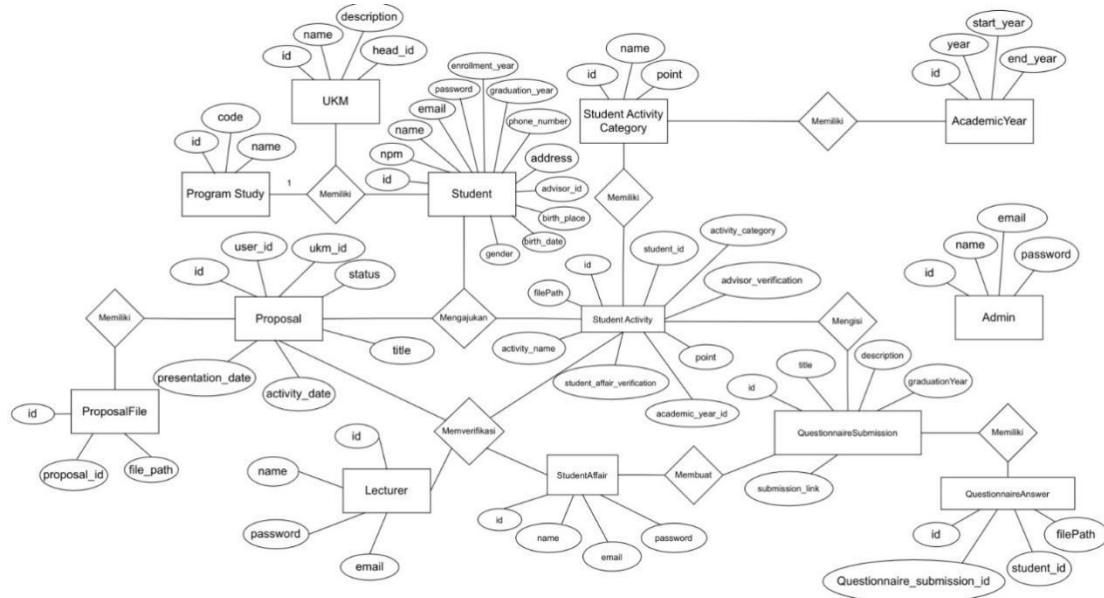
Berikut merupakan gambaran tentang kelas-kelas dalam perancangan sistem yang akan dibangun, termasuk atribut dan metode yang dimiliki oleh setiap kelas, serta hubungan antar kelas.

Gambar 3. 115 Class Diagram Sistem



3.3.9. ERD (*Entity Relationship Diagram*)

Entity Relationship Diagram (ERD) adalah representasi visual dari struktur *database* yang menggambarkan entitas serta hubungan antar entitas tersebut. Gambar di bawah ini menunjukkan ERD hasil dari desain arsitektur *database* yang telah dibuat.



Gambar 3. 116 ERD *Entity Relationship Diagram* Sistem

Entitas *Student* merepresentasikan tabel yang menyimpan data-data mahasiswa. Tabel ini berisi informasi detail tentang setiap mahasiswa termasuk data pribadi, akademik, dan relasi dengan entitas lainnya.

Tabel 3. 58 Struktur Tabel *Student*

No	Nama Field	Tipe Data	Keterangan
1	<i>id</i>	<i>Integer</i>	<i>Primary key</i> untuk identifikasi unik mahasiswa
2	<i>name</i>	<i>String</i>	Nama lengkap mahasiswa
3	<i>npm</i>	<i>String</i>	Nomor Pokok Mahasiswa (unik)
4	<i>email</i>	<i>String</i>	Alamat email mahasiswa (unik)
5	<i>password</i>	<i>String</i>	Password untuk autentikasi
6	<i>advisorId</i>	<i>Integer</i>	<i>Foreign key</i> yang merujuk ke id dosen pembimbing
7	<i>birthDate</i>	<i>String</i>	Tanggal lahir mahasiswa
8	<i>birthPlace</i>	<i>String</i>	Tempat lahir mahasiswa
9	<i>gender</i>	<i>Enum</i>	Jenis kelamin (MALE/FEMALE)
10	<i>address</i>	<i>String</i>	Alamat tempat tinggal

11	<i>phoneNumber</i>	<i>String</i>	Nomor telepon (unik)
12	<i>enrollmentYear</i>	<i>Integer</i>	Tahun masuk kuliah
13	<i>graduationYear</i>	<i>Integer</i>	Tahun lulus (opsional)
14	<i>programId</i>	<i>Integer</i>	<i>Foreign key</i> yang merujuk ke program studi
15	<i>createdAt</i>	<i>DateTime</i>	Waktu <i>record</i> dibuat
16	<i>updatedAt</i>	<i>DateTime</i>	Waktu terakhir <i>record</i> diperbarui

Entitas *QuestionnaireSubmission* merepresentasikan tabel yang menyimpan data pengumpulan kuesioner. Tabel ini mencatat informasi tentang kuesioner yang perlu diisi oleh mahasiswa.

Tabel 3. 59 Struktur Tabel *QuestionnaireSubmission*

No	Nama Field	Tipe Data	Keterangan
1	<i>id</i>	<i>Integer</i>	<i>Primary key</i> untuk identifikasi unik <i>submission</i>
2	<i>title</i>	<i>String</i>	Judul kuesioner
3	<i>description</i>	<i>String</i>	Deskripsi kuesioner
4	<i>graduationYears</i>	<i>String</i>	Tahun kelulusan yang ditargetkan
5	<i>submissionLink</i>	<i>String</i>	Link pengumpulan kuesioner
6	<i>createdAt</i>	<i>DateTime</i>	Waktu <i>record</i> dibuat
7	<i>updatedAt</i>	<i>DateTime</i>	Waktu terakhir <i>record</i> diperbarui

Entitas *QuestionnaireAnswer* merepresentasikan tabel yang menyimpan jawaban kuesioner dari mahasiswa. Tabel ini menghubungkan mahasiswa dengan kuesioner yang telah mereka isi.

Tabel 3. 60 Struktur Tabel *QuestionnaireAnswer*

No	Nama Field	Tipe Data	Keterangan
1	<i>id</i>	<i>Integer</i>	<i>Primary key</i> untuk identifikasi unik jawaban
2	<i>questionnaireSubmissionId</i>	<i>Integer</i>	<i>Foreign key</i> yang merujuk ke <i>submission</i>

3	<i>studentId</i>	<i>Integer</i>	<i>Foreign key yang merujuk ke mahasiswa</i>
4	<i>filePath</i>	<i>String</i>	<i>Path file jawaban kuesioner</i>
5	<i>createdAt</i>	<i>DateTime</i>	Waktu <i>record</i> dibuat
6	<i>updatedAt</i>	<i>DateTime</i>	Waktu terakhir <i>record</i> diperbarui

Entitas *Lecturer* merepresentasikan tabel yang menyimpan data-data dosen. Tabel ini berisi informasi tentang dosen yang mengajar dan membimbing mahasiswa.

Tabel 3. 61 Struktur Tabel *Lecturer*

No	Nama Field	Tipe Data	Keterangan
1	<i>id</i>	<i>Integer</i>	<i>Primary key</i> untuk identifikasi unik dosen
2	<i>name</i>	<i>String</i>	Nama lengkap dosen
3	<i>email</i>	<i>String</i>	Alamat email dosen (unik)
4	<i>password</i>	<i>String</i>	Password untuk autentikasi
5	<i>employeeNumber</i>	<i>String</i>	Nomor pegawai dosen (unik)
6	<i>createdAt</i>	<i>DateTime</i>	Waktu <i>record</i> dibuat
7	<i>updatedAt</i>	<i>DateTime</i>	Waktu terakhir <i>record</i> diperbarui
8	<i>deletedAt</i>	<i>DateTime</i>	Waktu <i>record</i> dihapus (soft delete)

Entitas *ProgramStudy* merepresentasikan tabel yang menyimpan data program studi. Tabel ini berisi informasi tentang program studi yang ada di institusi.

Tabel 3. 62 Struktur Tabel *ProgramStudy*

No	Nama Field	Tipe Data	Keterangan
1	<i>id</i>	<i>Integer</i>	<i>Primary key</i> untuk identifikasi unik program studi
2	<i>code</i>	<i>String</i>	Kode program studi
3	<i>name</i>	<i>String</i>	Nama program studi
4	<i>faculty</i>	<i>String</i>	Nama fakultas (opsional)
5	<i>degree</i>	<i>Enum</i>	Jenjang pendidikan (D3/S1/S2/S3)
6	<i>createdAt</i>	<i>DateTime</i>	Waktu <i>record</i> dibuat
7	<i>updatedAt</i>	<i>DateTime</i>	Waktu terakhir <i>record</i> diperbarui

Entitas *StudentActivity* merepresentasikan tabel yang menyimpan data aktivitas mahasiswa. Tabel ini mencatat berbagai kegiatan yang diikuti oleh mahasiswa.

Tabel 3. 63 Struktur Tabel *StudentActivity*

No	Nama Field	Tipe Data	Keterangan
1	<i>id</i>	<i>Integer</i>	<i>Primary key</i> untuk identifikasi unik aktivitas
2	<i>studentId</i>	<i>Integer</i>	<i>Foreign key</i> yang merujuk ke mahasiswa
3	<i>academicYearId</i>	<i>Integer</i>	<i>Foreign key</i> yang merujuk ke tahun akademik
4	<i>activityCategory</i>	<i>String</i>	Kategori aktivitas
5	<i>point</i>	<i>Integer</i>	Poin yang diperoleh dari aktivitas
6	<i>activityName</i>	<i>String</i>	Nama aktivitas
7	<i>activityDate</i>	<i>DateTime</i>	Tanggal aktivitas
8	<i>filePath</i>	<i>String</i>	<i>Path</i> file bukti aktivitas
9	<i>advisorVerification</i>	<i>Enum</i>	Status verifikasi dosen (<i>PENDING/APPROVED/REJECTED</i>)
10	<i>studentAffairVerification</i>	<i>Enum</i>	Status verifikasi bagian kemahasiswaan
11	<i>comments</i>	<i>String</i>	Komentar terkait aktivitas
12	<i>createdAt</i>	<i>DateTime</i>	Waktu <i>record</i> dibuat
13	<i>updatedAt</i>	<i>DateTime</i>	Waktu terakhir <i>record</i> diperbarui

Entitas *ActivityCategory* merepresentasikan tabel yang menyimpan kategori-kategori aktivitas. Tabel ini mendefinisikan jenis-jenis aktivitas dan poinnya.

Tabel 3. 64 Struktur Tabel *ActivityCategory*

No	Nama Field	Tipe Data	Keterangan
1	<i>id</i>	<i>Integer</i>	<i>Primary key</i> untuk identifikasi unik kategori
2	<i>name</i>	<i>String</i>	Nama kategori (unik)
3	<i>points</i>	<i>Integer</i>	Jumlah poin untuk kategori ini
4	<i>createdAt</i>	<i>DateTime</i>	Waktu <i>record</i> dibuat

5	updatedAt	DateTime	Waktu terakhir <i>record</i> diperbarui
6	deletedAt	DateTime	Waktu <i>record</i> dihapus (soft delete)

Entitas *StudentAffairs* merepresentasikan tabel yang menyimpan data pegawai bagian kemahasiswaan. Tabel ini berisi informasi tentang staf yang mengelola urusan kemahasiswaan.

Tabel 3. 65 Struktur Tabel *StudentAffairs*

No	Nama Field	Tipe Data	Keterangan
1	<i>id</i>	<i>Integer</i>	<i>Primary key</i> untuk identifikasi unik staf
2	<i>name</i>	<i>String</i>	Nama lengkap staf
3	<i>email</i>	<i>String</i>	Alamat email staf (unik)
4	<i>password</i>	<i>String</i>	Password untuk autentikasi
5	<i>createdAt</i>	<i>DateTime</i>	Waktu <i>record</i> dibuat
6	<i>updatedAt</i>	<i>DateTime</i>	Waktu terakhir <i>record</i> diperbarui

Entitas Admin merepresentasikan tabel yang menyimpan data administrator sistem. Tabel ini berisi informasi tentang pengguna yang memiliki akses penuh ke sistem.

Tabel 3. 66 Struktur Tabel Admin

No	Nama Field	Tipe Data	Keterangan
1	<i>id</i>	<i>Integer</i>	<i>Primary key</i> untuk identifikasi unik admin
2	<i>name</i>	<i>String</i>	Nama lengkap admin
3	<i>email</i>	<i>String</i>	Alamat email admin (unik)
4	<i>password</i>	<i>String</i>	Password untuk autentikasi
5	<i>createdAt</i>	<i>DateTime</i>	Waktu <i>record</i> dibuat
6	<i>updatedAt</i>	<i>DateTime</i>	Waktu terakhir <i>record</i> diperbarui

Entitas *Academic Year* merepresentasikan tabel yang menyimpan data tahun akademik. Tabel ini mencatat periode-periode akademik beserta statusnya.

Tabel 3. 67 Struktur Tabel Academic Year

No	Nama Field	Tipe Data	Keterangan
1	<i>id</i>	<i>Integer</i>	<i>Primary key</i> untuk identifikasi unik tahun akademik
2	<i>year</i>	<i>String</i>	Tahun akademik
3	<i>semester</i>	<i>Enum</i>	Semester (GANJIL/GENAP)
4	<i>startDate</i>	<i>Date</i>	Tanggal mulai semester
5	<i>endDate</i>	<i>Date</i>	Tanggal akhir semester
6	<i>isActive</i>	<i>Boolean</i>	Status aktif tahun akademik
7	<i>createdAt</i>	<i>DateTime</i>	Waktu <i>record</i> dibuat
8	<i>updatedAt</i>	<i>DateTime</i>	Waktu terakhir <i>record</i> diperbarui

Entitas UKM merepresentasikan tabel yang menyimpan data Unit Kegiatan Mahasiswa. Tabel ini berisi informasi tentang organisasi kemahasiswaan.

Tabel 3. 68 Struktur Tabel UKM

No	Nama Field	Tipe Data	Keterangan
1	<i>id</i>	<i>Integer</i>	<i>Primary key</i> untuk identifikasi unik UKM
2	<i>name</i>	<i>String</i>	Nama UKM (unik)
3	<i>description</i>	<i>String</i>	Deskripsi UKM
4	<i>headId</i>	<i>Integer</i>	<i>Foreign key</i> yang merujuk ke ketua UKM
5	<i>createdAt</i>	<i>DateTime</i>	Waktu <i>record</i> dibuat
6	<i>updatedAt</i>	<i>DateTime</i>	Waktu terakhir <i>record</i> diperbarui

Entitas Proposal merepresentasikan tabel yang menyimpan data proposal kegiatan UKM. Tabel ini mencatat proposal-proposal yang diajukan oleh UKM.

Tabel 3. 69 Struktur Tabel Proposal

No	Nama Field	Tipe Data	Keterangan
1	<i>id</i>	<i>Integer</i>	<i>Primary key</i> untuk identifikasi unik proposal
2	<i>ukmId</i>	<i>Integer</i>	<i>Foreign key</i> yang merujuk ke UKM
3	<i>title</i>	<i>String</i>	Judul proposal
4	<i>category</i>	<i>Enum</i>	Kategori proposal (INTERNAL/EXTERNAL)

5	<i>activityDate</i>	<i>DateTime</i>	Tanggal kegiatan
6	<i>status</i>	<i>Enum</i>	Status proposal (PENDING/APPROVED/REJECTED)
7	<i>presentationDate</i>	<i>DateTime</i>	Tanggal presentasi (opsional)
8	<i>comments</i>	<i>String</i>	Komentar terkait proposal
9	<i>createdAt</i>	<i>DateTime</i>	Waktu <i>record</i> dibuat
10	<i>updatedAt</i>	<i>DateTime</i>	Waktu terakhir <i>record</i> diperbarui

Entitas *ProposalFile* merepresentasikan tabel yang menyimpan file-file yang terkait dengan proposal. Tabel ini menyimpan informasi tentang dokumen pendukung proposal.

Tabel 3. 70 Struktur Tabel *ProposalFile*

No	Nama Field	Tipe Data	Keterangan
1	<i>id</i>	<i>Integer</i>	<i>Primary key</i> untuk identifikasi unik file
2	<i>proposalId</i>	<i>Integer</i>	<i>Foreign key</i> yang merujuk ke proposal
3	<i>name</i>	<i>String</i>	Nama file
4	<i>path</i>	<i>String</i>	<i>Path</i> penyimpanan file
5	<i>type</i>	<i>String</i>	Tipe file (opsional)
6	<i>createdAt</i>	<i>DateTime</i>	Waktu <i>record</i> dibuat

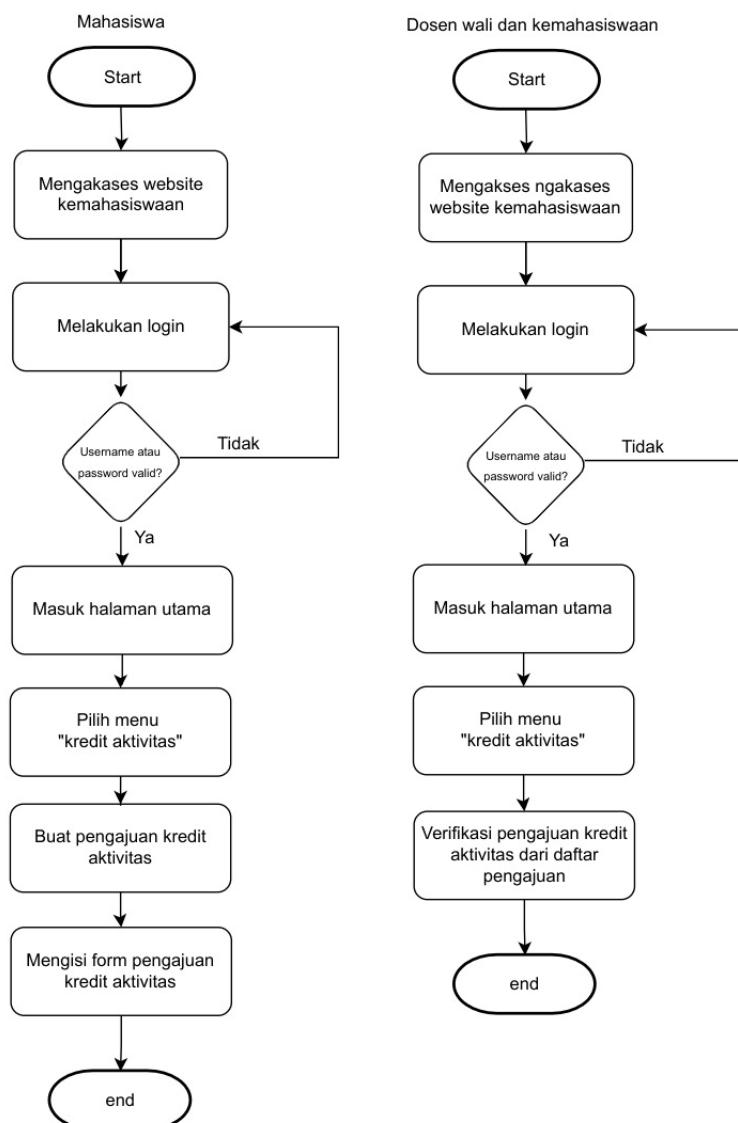
BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem Informasi Kemahasiswaan

4.1.1 Flowchart Implementasi Fitur

a. Flowchart Kredit Aktivitas Mahasiswa

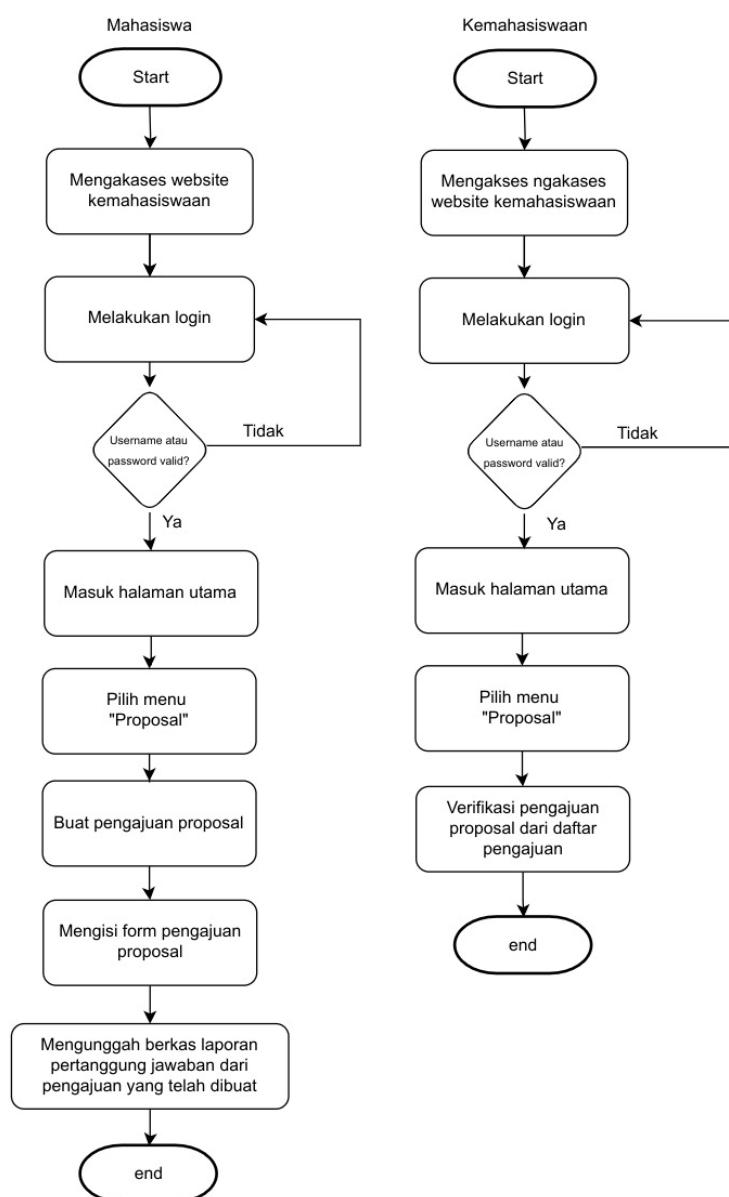


Gambar 4. 1 Flowchart Kredit Aktivitas Mahasiswa

Flowchart di atas menggambarkan proses pengajuan kredit aktivitas mahasiswa dan verifikasi oleh dosen wali. Proses dimulai ketika mahasiswa mengakses website kemahasiswaan dan melakukan *login* dengan memasukkan *username* dan *password*.

Setelah sistem memvalidasi *login*, jika berhasil, mahasiswa akan diarahkan ke halaman utama. Di sana, mahasiswa memilih menu "kredit aktivitas" untuk memulai pengajuan kredit aktivitas. Selanjutnya, mereka mengisi form pengajuan yang diperlukan sebelum proses pengajuan selesai. Di sisi lain, dosen wali juga mengakses website yang sama dan melakukan *login*. Setelah *login* berhasil, dosen wali masuk ke halaman utama dan memilih menu "kredit aktivitas" untuk memverifikasi pengajuan kredit aktivitas yang telah diajukan oleh mahasiswa.

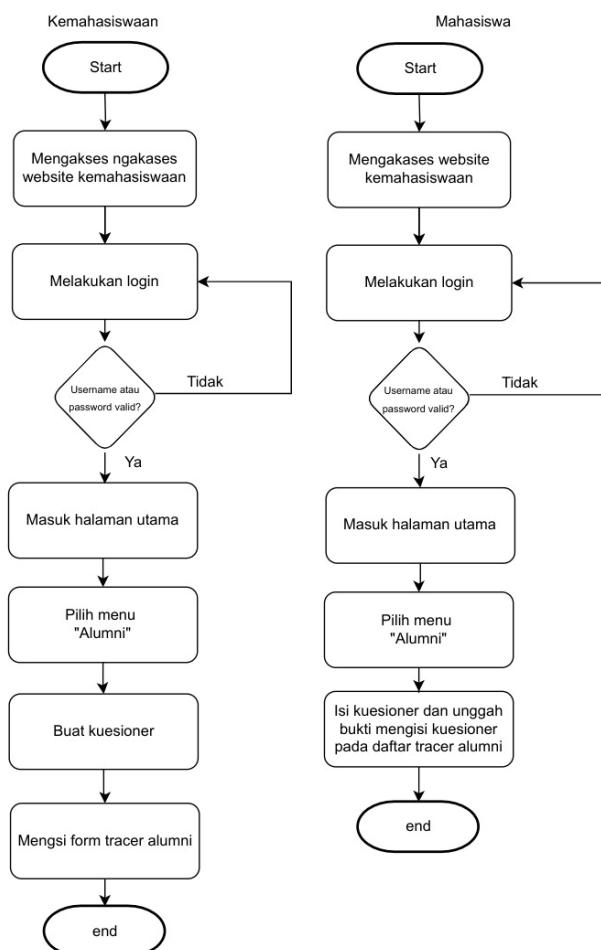
b. *Flowchart Pengajuan Proposal*



Gambar 4. 2 Flowchart Pengajuan Proposal

Flowchart di atas menggambarkan proses pengajuan proposal oleh mahasiswa dan verifikasi oleh pihak kemahasiswaan. Proses dimulai ketika mahasiswa mengakses website kemahasiswaan dan melakukan *login* dengan memasukkan *username* dan *password*. Setelah sistem memvalidasi *login*, jika berhasil, mahasiswa akan diarahkan ke halaman utama. Di halaman utama, mahasiswa memilih menu "Proposal" untuk memulai pengajuan. Mereka kemudian membuat pengajuan proposal dengan mengisi form yang diperlukan dan mengunggah berkas laporan pertanggungjawaban terkait pengajuan tersebut. Setelah semua langkah selesai, proses pengajuan berakhir. Di sisi lain, pihak kemahasiswaan juga mengakses website yang sama dan melakukan *login*. Setelah *login* berhasil, mereka masuk ke halaman utama dan memilih menu "Proposal" untuk memverifikasi pengajuan proposal yang telah diajukan oleh mahasiswa.

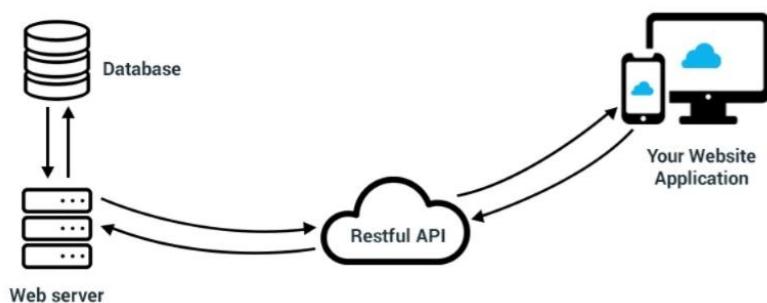
c. *Flowchart Tracer Almni*



Gambar 4. 3 *Flowchart Tracer Almni*

Flowchart di atas menggambarkan proses pengisian *tracer* alumni oleh mahasiswa dan pengelolaan oleh pihak kemahasiswaan. Proses dimulai ketika pihak kemahasiswaan mengakses website kemahasiswaan dan melakukan *login* dengan memasukkan *username* dan *password*. Setelah *login* berhasil, mereka akan diarahkan ke halaman utama, di mana mereka memilih menu "Alumni" untuk mengelola data alumni. Di sisi mahasiswa, mereka juga mengakses website yang sama dan melakukan *login*. Setelah *login* berhasil, mahasiswa masuk ke halaman utama dan memilih menu "Alumni." Di sini, mahasiswa diminta untuk mengisi kuesioner dan mengunggah bukti pengisian kuesioner pada daftar *tracer* alumni.

4.1.2 Arsitektur Sistem



Gambar 4. 4 Arsitektur Sistem

Arsitektur sistem yang ditunjukkan dalam gambar 4.4, menggambarkan interaksi antara beberapa komponen utama dalam aplikasi web menggunakan *RESTful API*. Di pusat arsitektur ini terdapat *database*, yang berfungsi sebagai tempat penyimpanan data. *Web server* menerima permintaan dari klien, seperti aplikasi web di komputer atau ponsel, dan mengelola komunikasi antara klien dan *database*. *RESTful API* berperan sebagai jembatan yang memungkinkan aplikasi untuk berkomunikasi dengan server melalui protokol *HTTP*, mendefinisikan cara data dapat diambil, ditambahkan, atau diperbarui. Proses dimulai ketika pengguna melakukan tindakan di aplikasi, yang kemudian mengirimkan permintaan ke *web server*. *Web server* meneruskan permintaan tersebut ke *RESTful API*, yang berinteraksi dengan *database* untuk mendapatkan data yang diperlukan. Setelah itu, data dikembalikan ke aplikasi klien untuk ditampilkan kepada pengguna.

4.1.3 Perangkat Yang Digunakan

Saat pengembangan sistem antrian ini, perangkat yang digunakan agar sistem yang dibuat dapat berjalan dengan baik dengan penerapan sistem yang diperlukan terbagi menjadi dua yaitu perangkat lunak dan perangkat keras

4.1.3.1 Spesifikasi Perangkat Lunak

Perangkat lunak yang digunakan dalam pengembangan sistem antrian ini adalah sebagai berikut:

1. Sistem operasi : Debian GNU/Linux 12 (bookworm)
2. Text editor : *Visual Studio Code*
3. Aplikasi pemodelan sistem : *Draw.io*
4. Bahasa Pemrograman : *JavaScript*
5. DBMS : MySQL

4.1.3.2 Spesifikasi Perangkat Keras

Perangkat keras yang digunakan dalam pengembangan sistem antrian ini adalah laptop Acer Aspire E5-475G dengan spesifikasi sebagai berikut:

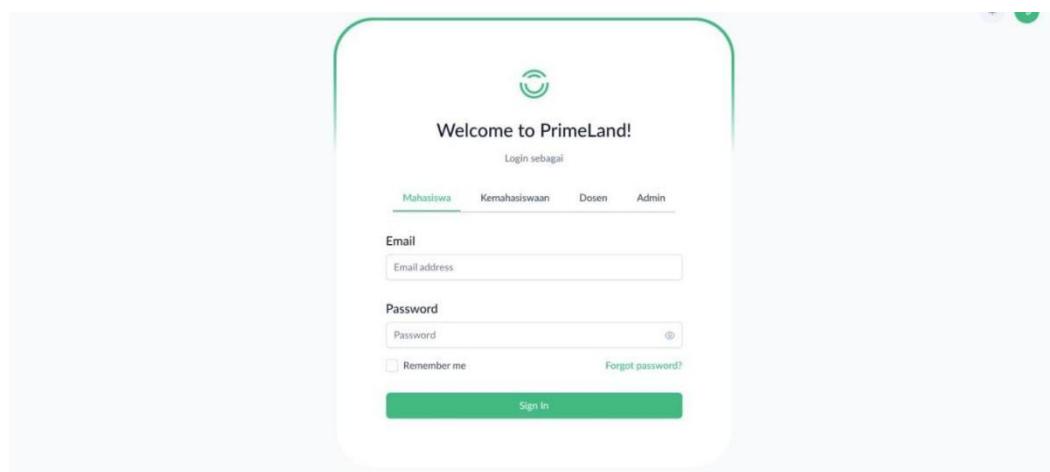
1. Processor : Intel i5-7200U (4 Core) - 3.100 GHz
2. Memory RAM : 12 GB 2400 Mhz DDR 4
3. Storage : SSD 240 GB & 500 GB HDD

4.1.4 Implementasi Antarmuka

Implementasi antar muka diimplementasikan dengan menampilkan setiap halaman dari program yang dikembangkan. Penulis telah mengembangkan antarmuka ini dengan tampilan yang berbeda untuk mahasiswa, dosen, kemahasiswaan, dan admin.

4.1.4.1 Implementasi Antarmuka Mahasiswa

1. Tampilan *Login* Mahasiswa



Gambar 4. 5 Tampilan *Login* Mahasiswa

Pada gambar 4.1 ini menampilkan tampilan *login* dari mahasiswa yang dimana untuk tampilan *login* mahasiswa terdapat *form email* dan *form password* untuk diinput oleh mahasiswa agar masuk ke halaman mahasiswa.

3. Tampilan Aktivitas Mahasiswa

The screenshot shows the Mahasiswa (Student) activities dashboard. At the top, it displays the user profile (Yuda Saputra, 2042069), the accumulated credit activities (32/70 Points), and the advisor information (Dr. John Doe, NIP. 1209371283). Below this, there is a section titled 'Daftar Aktivitas' (List of Activities) with filters for academic year (2024/2025 GENAP), advisor status, and activity status. The table lists activities with columns for year, category, name, points, advisor verification, and student verification. A green button '+ Buat Pengajuan' (Create Application) is located at the top right of the activity list.

Gambar 4. 6 Tampilan Aktivitas Mahasiswa

Tampilan aktivitas mahasiswa pada Gambar 4.3 terdapat indikator akumulasi kredit aktivitas yang menunjukkan progres poin yang telah diperoleh serta informasi mengenai dosen wali, termasuk nama dan NIP, juga ditampilkan terdapat tabel daftar aktivitas mahasiswa yang memuat informasi, seperti tahun akademik, status dosen wali, status kemahasiswaan, kategori aktivitas, nama aktivitas, poin yang diperoleh, serta verifikasi dari dosen wali dan pihak kemahasiswaan. Selain itu, terdapat tombol pengajuan yang memungkinkan mahasiswa untuk mengajukan aktivitas baru.

3. Tampilan Tracer Alumni (Mahasiswa)

The screenshot shows the Mahasiswa (Student) Tracer Alumni dashboard. It displays a table of questionnaires taken by alumni, with columns for status, name of the survey taker, survey link, and evidence of filling out. The table shows six entries, each with a preview thumbnail and edit/delete icons. The status column includes 'Belum Mengisi' and 'Sudah Mengisi'. The survey links are mostly test URLs like http://test.com or http://sdafhaflaksldf.com.

Gambar 4. 7 Tampilan Tracer Alumni Mahasiswa

Tampilan *tracer* alumni pada **Gambar 4.4** menunjukkan daftar kuesioner yang dapat diisi oleh mahasiswa. Di bagian atas, terdapat status yang menunjukkan apakah mahasiswa sudah mengisi kuesioner atau belum, dengan beberapa kuesioner berstatus "Sudah Mengisi" dan lainnya "Belum Mengisi". Tabel ini memuat informasi penting, seperti nama kuesioner, link kuesioner yang dapat diakses, serta opsi untuk mengunggah bukti pengisian kuesioner. Setiap entri dalam tabel juga dilengkapi dengan tombol untuk mengedit atau menghapus bukti yang telah diunggah.

3. Tampilan Pengajuan Proposal UKM Mahasiswa

The screenshot shows the 'Proposal Kegiatan' section of the STMK AMIK BANDUNG application. The interface includes a sidebar with user information (Yuda Saputra, 2042069) and navigation links (HOME, Dashboard, MAHASISWA, Aktivitas, Proposal, Tracer Alumni). The main content area displays a table of proposals with the following data:

Status	Judul Proposal	Jenis Kegiatan	Waktu Kegiatan	Waktu Presentasi	Komentar	Aksi
Menunggu	lomba futsal	EXTERNAL	13/1/2025	31/1/2025		(edit)
Disetujui	lomba lagi	EXTERNAL	15/1/2025			(edit)
Ditolak	lagi lagi	INTERNAL	8/1/2025			(edit)

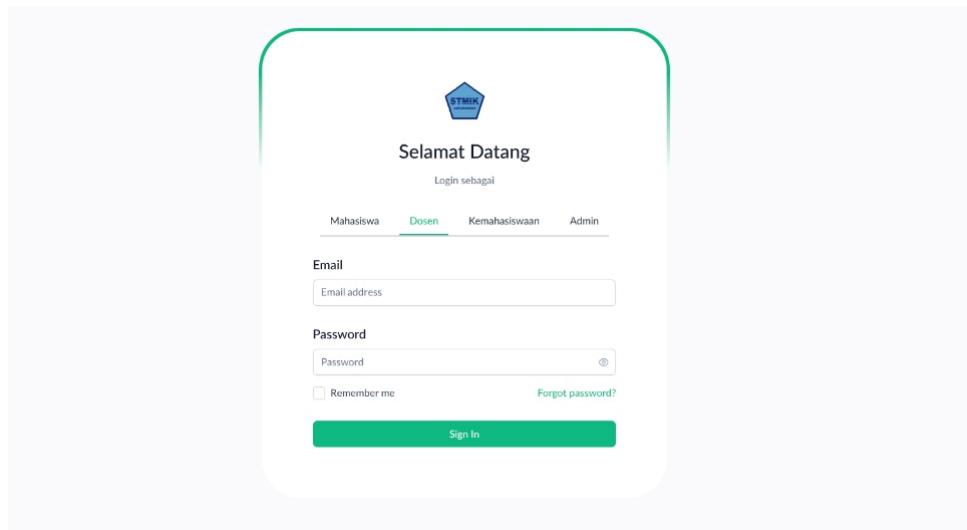
Below the table, there is a section for attachments related to the first proposal, showing file names like 'TTD LAPORAN-ENRICHMENT, YUDA-SAPUTRA revisi pengesahan.pdf', 'output.pdf', and 'myhumandesign-chart (2).pdf'. Each attachment has a 'Lihat Berkas' (View File) button next to it. The bottom of the page includes a footer with 'SAKAI by PrimeVue'.

Gambar 4. 8 Tampilan Pengajuan Proposal UKM Mahasiswa

Tampilan proposal kegiatan pada **Gambar 4.5** menunjukkan daftar proposal yang diajukan oleh mahasiswa. Di bagian atas, terdapat opsi untuk memilih status dan jenis kegiatan, serta status proposal yang saat ini ditampilkan sebagai "Menunggu". Tabel ini memuat informasi penting, termasuk judul proposal, jenis kegiatan, waktu kegiatan, dan waktu presentasi. Di bawahnya, terdapat daftar berkas yang diunggah untuk lomba futsal, lengkap dengan nama berkas yang dapat diakses. Status dari proposal juga ditampilkan, dengan beberapa berkas yang telah disetujui dan ditolak. Selain itu, terdapat tombol **Tambah Proposal** untuk memungkinkan mahasiswa mengajukan proposal baru.

4.1.4.2 Implementasi Antarmuka Dosen

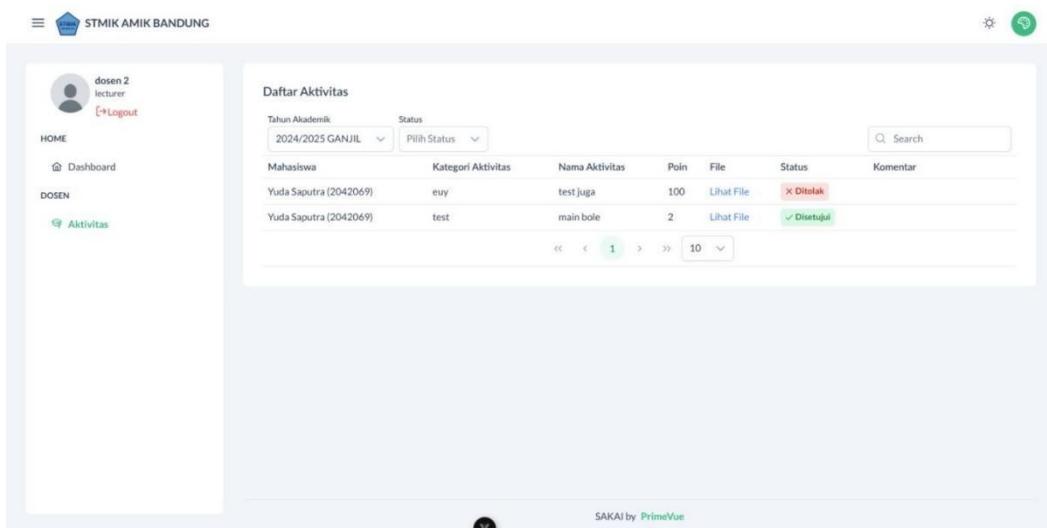
1. Tampilan *Login* Dosen



Gambar 4. 9 Tampilan *Login* Dosen

Pada gambar 4.5 ini menampilkan tampilan *login* dari dosen yang dimana untuk tampilan *login* dosen terdapat *form email* dan *form password* untuk diinput oleh mahasiswa agar masuk ke halaman dosen.

2. Tampilan Aktivitas Dosen



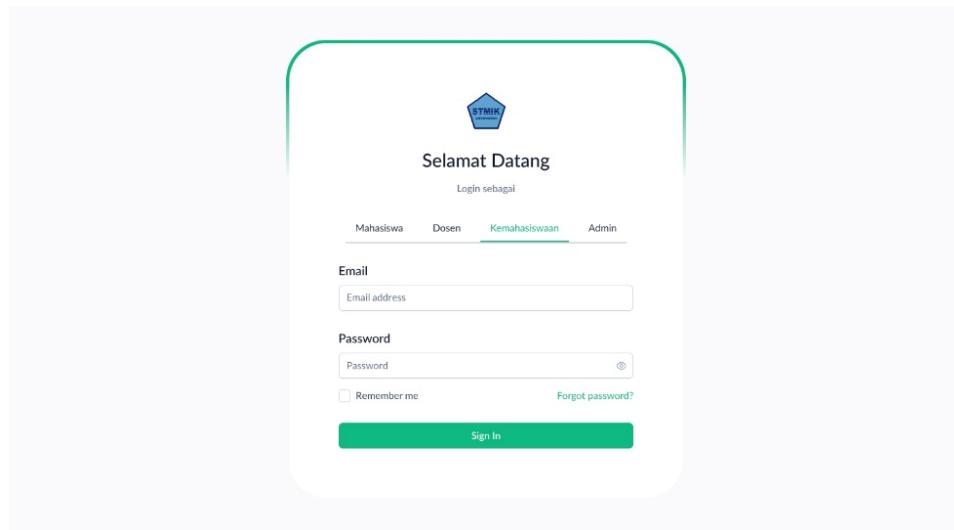
Gambar 4. 10 Tampilan Aktivitas Dosen

Tampilan aktivitas dosen pada Gambar 4.7 menunjukkan daftar aktivitas yang telah diajukan oleh mahasiswa serta terdapat opsi untuk memilih status aktivitas. Tabel ini memuat informasi penting, termasuk nama mahasiswa, kategori aktivitas, nama aktivitas, poin yang diperoleh, serta status pengajuan. Setiap entri dalam tabel

dilengkapi dengan opsi untuk melihat file yang diunggah. Status dari setiap aktivitas dapat dipilih oleh dosen apakah disetujui atau ditolak.

4.1.4.3 Implementasi Antarmuka Kemahasiswaan

1. Tampilan *Login* Kemahasiswaan



Gambar 4. 11 Tampilan *Login* Kemahasiswaan

Pada gambar 4.7 ini menampilkan tampilan *login* dari kemahasiswaan yang dimana untuk tampilan *login* kemahasiswaan terdapat *form email* dan *form password* untuk diinput oleh kemahasiswaan agar masuk ke halaman kemahasiswaan.

2. Tampilan Aktivitas Kemahasiswaan

A screenshot of a web-based application for managing student activities. At the top left is a sidebar with a user icon, the text 'student-affairs', and a red 'Logout' button. The sidebar also includes links for 'HOME', 'Dashboard', 'KEMAHASISWAAN', 'Kredit Aktivitas' (which is highlighted in green), 'Proposal Kegiatan', and 'Tracer Alumni'. The main content area has a header 'Daftar Aktivitas' with dropdown menus for 'Tahun Akademik' (set to '2024/2025 GANJIL') and 'Status' (set to 'Pilih Status'). Below this is a table titled 'Daftar Aktivitas' showing one activity entry: 'Yuda Saputra (2042069)' in the 'Mahasiswa' column, 'test' in 'Kategori Aktivitas', 'main bole' in 'Nama Aktivitas', '2' in 'Poin', and a red 'X Ditolak' button in the 'Status' column. There are navigation buttons for page numbers (1, 2, 3, etc.) and a search bar. At the bottom of this section is a table titled 'Kategori Aktivitas' with three entries: 'test', 'Seminar Eksternal', and 'Seminar Internal'. Each entry has a 'Poin' value (2, 3, 2 respectively) and three circular icons with icons for edit, delete, and another action. A green '+ Tambah Kategori' button is located at the top right of this table. The footer of the page includes the text 'SAKAI by PrimeVue'.

Gambar 4. 12 Tampilan Aktivitas Kemahasiswaan

Tampilan aktivitas kemahasiswaan pada Gambar 4.9 menunjukkan daftar aktivitas yang telah diajukan oleh mahasiswa. Di bagian atas, terdapat informasi mengenai tahun akademik yang sedang berlangsung, serta opsi untuk memilih status aktivitas. Terdapat tabel yang memuat informasi penting, termasuk nama mahasiswa, kategori aktivitas, nama aktivitas, poin yang diperoleh, serta status pengajuan serta terdapat status mengajuan yang dapat dipilih oleh kemahasiswaan, dengan opsi untuk melihat file yang diunggah. Di bawah tabel, terdapat bagian kategori aktivitas yang mencantumkan nama kategori dan poin yang terkait, seperti kategori, "Seminar Eksternal", dan "Seminar Internal" dengan masing-masing poin. Terdapat juga tombol Tambah Kategori yang memungkinkan pengguna untuk menambahkan kategori baru.

3. Tampilan Proposal UKM Kemahasiswaan

Status	Judul Proposal	Jenis Kegiatan	Waktu Kegiatan	Waktu Presentasi	Komentar
Pending	Seminar Teknologi Terkini	Internal	15/1/2025, 10.00.00		
Berkas untuk Seminar Teknologi Terkini Nama Berkas Surat Permohonan Kegiatan SK Panitia Kegiatan Surat Pernyataan Pertanggung Jawaban Kegiatan Surat Permohonan Dispensasi Mahasiswa Surat Perizinan Tempat Kegiatan Surat Pemberitahuan Kegiatan Kemahasiswaan Surat Pemberitahuan Kegiatan Wakil Ketua Surat Pemberitahuan Kegiatan Sarpras Surat Pemberitahuan Kegiatan BEM Surat Pemberitahuan Kegiatan Pembina Term of Reference (TOR) Kegiatan					
> Approved Konferensi Nasional Eksternal 20/2/2025, 09.00.00 19/2/2025, 15.00.00 Semua berkas lengkap					

Gambar 4. 13 Tampilan Proposal UKM Kemahasiswaan

Tampilan proposal unit kegiatan mahasiswa pada **Gambar 4.10** menunjukkan daftar berkas yang diperlukan untuk kegiatan seminar. Tabel ini mencakup berbagai dokumen, seperti surat permohonan kegiatan, SK panitia, dan surat pemberitahuan terkait, dengan opsi untuk **Lihat Berkas** pada setiap entri. Selain itu, terdapat tombol untuk menerima atau menolak proposal, memungkinkan pengguna untuk mengelola status pengajuan dengan mudah.

4. Tampilan Kemahasiswaan *Tracer Alumni*

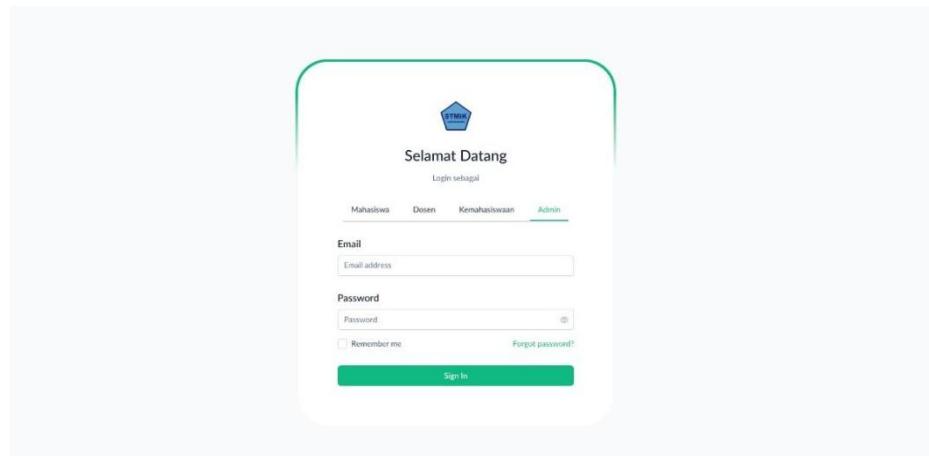
The screenshot shows a web-based application interface for managing tracer alumni. On the left, there's a sidebar with navigation links: HOME, KEMAHASISWAAN (with sub-links for Kredit Aktivitas and Proposal Kegiatan), and Tracer Alumni. The main content area is titled "Tracer Alumni" and displays a table of questionnaires. The table has columns for "Nama Kuesioner", "Deskripsi", "Rentang Tahun Kelulusan", and "Link Kuesioner". Each row contains a link to a specific questionnaire page. At the bottom of the table, there are navigation buttons for "Showing 1 to 6 of 6 entries" and a page number "10". A green button labeled "+ Tambah Tracer Alumni" is located at the top right of the table. The footer of the page says "SAKAI by PrimeVue".

Gambar 4. 14 Tampilan Kemahasiswaan *Tracer Alumni*

Tampilan kemahasiswaan *tracer* alumni pada **Gambar 4.11** memungkinkan pihak kemahasiswaan untuk melihat kuesioner yang telah diisi oleh alumni. Dalam tabel yang disediakan, terdapat informasi mengenai nama kuesioner, deskripsi, dan rentang tahun kelulusan, yang memberikan gambaran jelas tentang kuesioner yang relevan. Pihak kemahasiswaan dapat mengakses link kuesioner untuk melihat hasil pengisian alumni, serta memantau partisipasi alumni dalam program *tracer*. Dengan fitur ini, kemahasiswaan dapat mengevaluasi feedback dari alumni, yang berguna untuk perbaikan program dan kegiatan di masa mendatang. Selain itu, kemahasiswaan juga memiliki opsi untuk mengedit atau menghapus kuesioner yang ada, sehingga dapat mengelola data dengan lebih efektif.

4.1.4.4 Implementasi Antarmuka Admin

1. Tampilan *Login Admin*



Gambar 4. 15 Tampilan *Login Admin*

Pada gambar 4.11 ini menampilkan tampilan *login* dari dosen yang dimana untuk tampilan *login* dosen terdapat *form email* dan *form password* untuk diinput oleh dosen agar masuk ke halaman mahasiswa.

3. Tampilan Program Studi Admin

Kode	Nama
IF	Teknik Informatika
SI	Sistem Informasi
DKV	Desain Komunikasi Visual

Gambar 4. 16 Tampilan Program Studi Admin

Tampilan program studi pada **Gambar 4.14** menunjukkan daftar program studi yang dikelola oleh admin. Tabel ini mencakup informasi penting, seperti kode program studi dan nama program studi, yang memudahkan admin dalam mengelola data. Setiap entri dalam tabel dilengkapi dengan opsi untuk mengedit atau menghapus program studi yang ada. Terdapat juga tombol *+ New* yang memungkinkan admin untuk menambahkan program studi baru. Dengan tampilan ini, admin dapat dengan mudah memantau dan mengelola program studi yang tersedia di institusi, memastikan bahwa informasi yang ditampilkan selalu akurat dan terkini.

4. Tampilan Tahun Akademik Admin

Year	Semester	Status
2024/2025	GENAP	Active
2024/2025	GANJIL	Inactive

Gambar 4. 17 Tampilan Tahun Akademik Admin

Tampilan tahun akademik pada **Gambar 4.15** menunjukkan daftar tahun akademik, tabel ini mencakup informasi penting, seperti tahun akademik, semester, dan status aktif atau tidaknya tahun akademik tersebut. Setiap entri dalam tabel dilengkapi dengan opsi untuk menghapus tahun akademik yang ada. Terdapat juga tombol **+ New** yang memungkinkan admin untuk menambahkan tahun akademik baru.

5. Tampilan Unit Kegiatan Mahasiswa Admin

Nama	Deskripsi	Ketua
HIMATIF	Himpunan Mahasiswa Teknik Informatika	No Head Assigned
HIMSI	Himpunan Mahasiswa Sistem Informasi	mahasiswa ke 7 (20420603)

Gambar 4. 18 Tampilan Unit Kegiatan Mahasiswa Admin

Tampilan unit kegiatan mahasiswa pada **Gambar 4.16** menunjukkan daftar UKM (Unit Kegiatan Mahasiswa) yang dikelola oleh admin. Tabel ini mencakup informasi penting, seperti nama UKM, deskripsi, dan ketua yang bertanggung jawab. Setiap entri dalam tabel dilengkapi dengan opsi untuk mengedit atau menghapus UKM yang ada. Terdapat juga tombol **+ New** yang memungkinkan admin untuk menambahkan UKM baru.

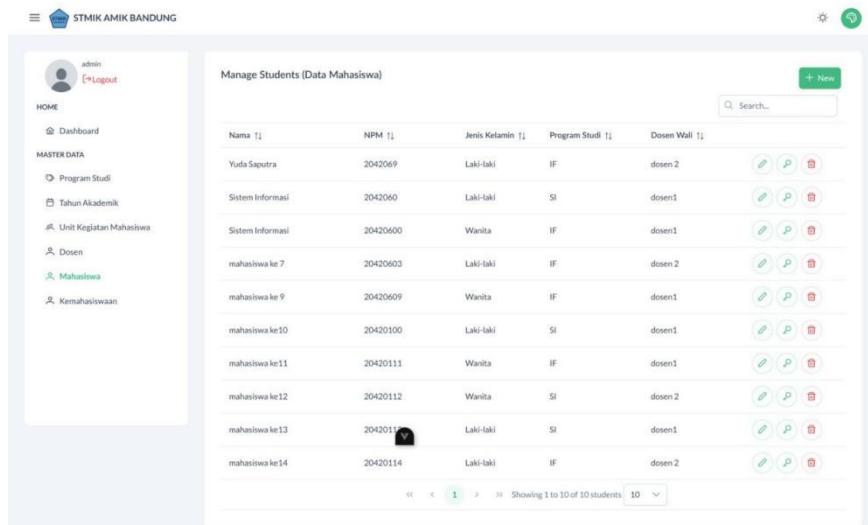
6. Tampilan Dosen Admin

Name	Email	Employee Number
dosen1	dosen1@stmk-amikbandung.ac.id	1023712093
dosen 2	dosen2@stmk-amikbandung.ac.id	098908923

Gambar 4. 19 Tampilan Dosen Admin

Tampilan dosen pada **Gambar 4.17** menunjukkan daftar dosen yang dikelola oleh admin. Tabel ini mencakup informasi penting, seperti nama dosen, alamat email, dan nomor pegawai. Setiap entri dalam tabel dilengkapi dengan opsi untuk mengedit atau menghapus data dosen yang ada. Terdapat juga tombol **+ New** yang memungkinkan admin untuk menambahkan dosen baru ke dalam sistem.

7. Tampilan Mahasiswa Admin

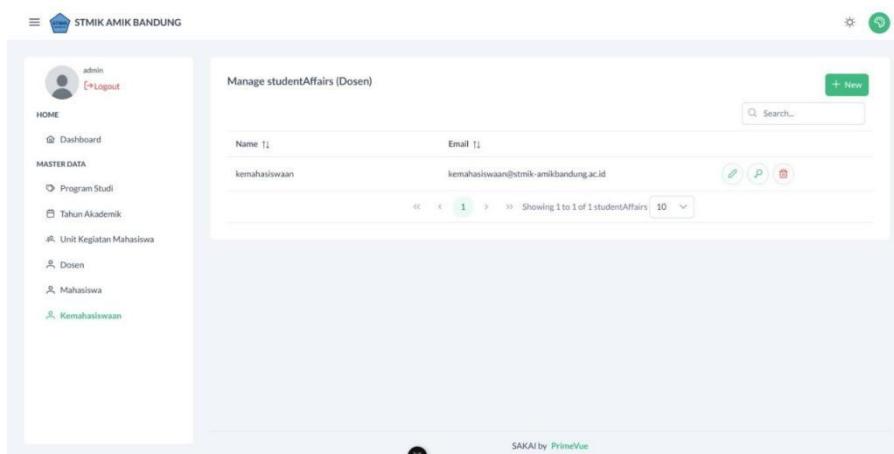


Nama	NPM	Jenis Kelamin	Program Studi	Dosen Wali
Yuda Saputra	2042069	Laki-laki	IF	dosen 2
Sistem Informasi	2042060	Laki-laki	SI	dosen1
Sistem Informasi	20420600	Wanita	IF	dosen1
mahasiswa ke 7	20420603	Laki-laki	IF	dosen 2
mahasiswa ke 9	20420609	Wanita	IF	dosen1
mahasiswa ke10	20420100	Laki-laki	SI	dosen1
mahasiswa ke11	20420111	Wanita	IF	dosen1
mahasiswa ke12	20420112	Wanita	SI	dosen 2
mahasiswa ke13	20420113	Laki-laki	SI	dosen1
mahasiswa ke14	20420114	Laki-laki	IF	dosen 2

Gambar 4. 20 Tampilan Mahasiswa Admin

Tampilan mahasiswa pada **Gambar 4.18** menunjukkan daftar data mahasiswa yang dikelola oleh admin. Tabel ini mencakup informasi penting, seperti nama mahasiswa, NPM (Nomor Pokok Mahasiswa), jenis kelamin, program studi, dan dosen wali. Setiap entri dalam tabel dilengkapi dengan opsi untuk mengedit atau menghapus data mahasiswa yang ada. Terdapat juga tombol **+ New** yang memungkinkan admin untuk menambahkan mahasiswa baru ke dalam sistem.

8. Tampilan Kemahasiswaan Admin



Name	Email
kemahasiswaan	kemahasiswaan@stmk-amikbandung.ac.id

Gambar 4. 21 Tampilan Kemahasiswaan Admin

Tampilan kemahasiswaan pada **Gambar 4.19** menunjukkan data terkait pengelolaan kemahasiswaan yang dikelola oleh admin. Tabel ini mencakup informasi penting, seperti nama kemahasiswaan dan alamat email yang terkait. Terdapat opsi untuk mengedit atau menghapus data kemahasiswaan yang ada. Selain itu, tombol **+ New** memungkinkan admin untuk menambahkan entri kemahasiswaan baru ke dalam sistem.

4.2 Implementasi Status Pengajuan

Dalam mengimplementasikan informasi status pengajuan pada sistem informasi kemahasiswaan, penulis menerapkan **filter** dan **sorting** untuk mengurutkan daftar pengajuan berdasarkan tahun akademik dan status pengajuan "menunggu" yang ditampilkan terlebih dahulu. Informasi status ini, dari sudut pandang aktor mahasiswa, akan mengurutkan status "menunggu" terlebih dahulu dengan tahun akademik yang sedang berjalan atau aktif.

Sementara dari sudut pandang aktor dosen dan kemahasiswaan, sistem juga menyediakan opsi untuk melihat semua pengajuan, termasuk yang telah disetujui atau ditolak. Dengan demikian, dosen dan pihak kemahasiswaan dapat dengan mudah memantau dan mengelola pengajuan yang masuk, serta memberikan umpan balik yang diperlukan. Penerapan fitur ini tidak hanya meningkatkan efisiensi dalam pengelolaan data pengajuan, tetapi juga memastikan transparansi dalam proses evaluasi dan keputusan yang diambil.

Dengan adanya sistem *filter* dan *sorting* ini, diharapkan mahasiswa dapat lebih cepat menemukan informasi yang relevan mengenai status pengajuan mereka, sehingga dapat mengambil langkah selanjutnya dengan lebih tepat dan efisien. Hal ini juga berkontribusi pada peningkatan pengalaman pengguna dalam menggunakan sistem informasi kemahasiswaan.

Kategori Aktivitas	Nama Aktivitas	Poin	File	Status	Komentar
Mahasiswa	workshop luar	3	Lihat File	Δ Menunggu	
mahasiswa ke 7 (20240602)	Seminar Eksternal	3	Lihat File	Δ Menunggu	
Yuda Saputra (2042069)	Bijak Menggunakan Social Media	3	Lihat File		
Yuda Saputra (2042069)	Prospek Kerja Data Analysis	3	Lihat File	X Ditolak	
Yuda Saputra (2042069)	Seminar Internal	2	Lihat File	✓ Disetujui	
Yuda Saputra (2042069)	Seminar Kepelimpahan	2	Lihat File	✓ Disetujui	

Gambar 4.22 Informasi Status Pengajuan

Pada gambar 4.22 diatas menunjukkan informasi status pengajuan mahasiswa sehingga informasi dapat langsung lihat oleh mahasiswa.

4.3 Hasil Pengujian Sistem

Dalam pengembangan sistem informasi kemahasiswaan di STMIK "AMIKBANDUNG", tujuan utamanya adalah untuk meningkatkan efisiensi dan transparansi dalam pengelolaan pengajuan kredit aktivitas mahasiswa, pengajuan proposal skripsi, dan pelacakan alumni. Dengan adanya sistem ini, diharapkan berbagai permasalahan yang dihadapi dalam proses manual dapat diselesaikan dengan lebih efisien, sehingga kegiatan operasional dapat berjalan dengan baik dan memberikan kemudahan bagi mahasiswa. Berikut adalah proses pengujian yang dilakukan oleh penulis dengan metode blackbox, yang bertujuan untuk memastikan bahwa semua fitur sistem berfungsi sesuai dengan yang diharapkan dan memenuhi kebutuhan pengguna.

Tabel 3. 71 *Blackbox Testing*

No	Fitur	Skenario	Hasil yang diharapkan	Hasil pengujian	Test
Mahasiswa					
1	<i>Login</i>	mengisi email dan password yang valid	<i>Login</i> berhasil	<i>Login</i> berhasil, Masuk ke halaman utama	valid
		mengisi email dan password yang tidak valid	Gagal <i>login</i>	Gagal <i>login</i> , menampilkan informasi email atau <i>password</i> salah	valid
2	<i>Logout</i>	klik tombol " <i>Logout</i> "	berhasil <i>logout</i>	berhasil <i>logout</i> , kembali ke halaman <i>login</i>	valid
3	Lihat aktivitas mahasiswa	klik menu aktivitas mahasiswa	menampilkan data daftar aktivitas mahasiswa	berhasil menampilkan data daftar aktivitas mahasiswa	valid
4	buat pengajuan aktivitas mahasiswa	klik tombol buat pengajuan aktivitas	menampilkan form pengajuan aktivitas	nerhasil menampilkan form pengajuan aktivitas	valid
		mengisi form pengajuan aktivitas	menyimpan data pengajuan aktivitas	berhasil menyimpan data pengajuan aktivitas	valid
5	batalkan pengajuan	klik tombol "batalkan" pada	menampilkan konfirmasi	berhasil menampilkan	valid

	aktivitas mahasiswa	daftar pengajuan yang dipilih	batalkan pengajuan	konfirmasi batalkan pengajuan	
		pilih "ya"	membatalkan pengajuan	berhasil membatalkan pengajuan	valid
6	Lihat Daftar Pengajuan Proposal	klik menu proposal	menampilkan daftar data proposal	berhasil menampilkan data daftar proposal	valid
7	Buat Pengajuan Proposal	klik tombol buat pengajuan proposal	menampilkan form pengajuan proposal	berhasil menampilkan form pengajuan proposal	valid
		mengisi form pengajuan proposal	menyimpan data pengajuan proposal	berhasil menyimpan data pengajuan proposal	valid
8	Membatalkan Pengajuan Proposal	klik tombol "batalkan" pada daftar pengajuan yang dipilih	menampilkan konfirmasi batalkan pengajuan	berhasil menampilkan konfirmasi batalkan pengajuan	valid
		pilih "ya"	membatalkan pengajuan	berhasil membatalkan pengajuan	valid
9	Lihat Daftar <i>Tracer</i> Alumni	klik menu alumni	menampilkan daftar data kuesioner <i>tracer</i> alumni	berhasil menampilkan data daftar kuesioner <i>tracer</i> alumni	valid
10	Kirim Bukti Pengisian <i>Tracer</i> Alumni	klik tombol "upload bukti" dan pilih file pada daftar kuesioner yang dipilih	menyimpan data bukti mengisi kuesioner	berhasil menyimpan data bukti mengisi kuesioner	valid
11	Mengubah Bukti Pengisian <i>Tracer</i> Alumni	klik tombol "edit" dan pilih file pada daftar kuesioner yang dipilih	mengedit data bukti mengisi kuesioner	berhasil mengedit data bukti mengisi kuesioner	valid
12	Membatalkan Bukti Pengisian <i>Tracer</i> Alumni	klik tombol "batalkan" dan pilih file pada daftar kuesioner yang dipilih	menampilkan konfirmasi batalkan bukti pengisian <i>tracer</i> alumni	berhasil menampilkan konfirmasi batalkan bukti pengisian <i>tracer</i> alumni	valid
		pilih "ya"	membatalkan data bukti mengisi kuesioner	berhasil membatalkan data bukti mengisi kuesioner	valid

Dosen					
13	<i>Login</i>	mengisi email dan passoword yang valid	<i>Login</i> berhasil	<i>Login</i> berhasil, Masuk ke halaman utama	valid
		mengisi email dan passoword yang tidak valid	Gagal <i>login</i>	Gagal <i>login</i> , menampilkan informasi email atau <i>password</i> salah	valid
14	<i>Logout</i>	klik tombol " <i>Logout</i> "	berhasil <i>logout</i>	berhasil <i>logout</i> , kembali ke halaman <i>login</i>	valid
15	Lihat Detail Bukti Aktivitas Mahasiswa	pilih menu aktivitas mahasiswa	menampilkan daftar data pengajuan aktivitas mahasiswa	berhasill menampilkan data daftar pengajuan aktivitas mahasiswa	valid
		klik tombol "lihat bukti" pada daftar pengajuan aktivitas mahasiswa yang dipilih	menampilkan file bukti pengajuan aktivitas mahasiswa	berhasill menampilkan file bukti pengajuan aktivitas mahasiswa	valid
16	Edit Status Verifikasi Pengajuan Proposal	klik status pada daftar pengajuan aktivitas mahasiswa yang dipilih, lalu pilih opsi status pengajuan	mengubah status verifikasi dosen wali pengajuan aktivitas mahasiswa	berhasill mengubah status verifikasi dosen wali pengajuan aktivitas mahasiswa	valid
Kemahasiswaan					
17	<i>Login</i>	mengisi email dan passoword yang valid	<i>Login</i> berhasil	<i>Login</i> berhasil, Masuk ke halaman utama	valid
		mengisi email dan passoword yang tidak valid	Gagal <i>login</i>	Gagal <i>login</i> , menampilkan informasi email atau <i>password</i> salah	valid
18	<i>Logout</i>	klik tombol " <i>Logout</i> "	berhasil <i>logout</i>	berhasil <i>logout</i> , kembali ke halaman <i>login</i>	valid
19	Lihat Daftar Kategori Aktivitas	pilih menu "kategori aktivitas"	menampilkan halaman kategori aktivitas beserta	berhasil menampilkan halaman kategori aktivitas beserta	valid

			daftar data kategori aktivitas	daftar data kategori aktivitas	
20	Tambah Data Kategori Aktivitas	klik tombol "tambah kategori"	menampilkan form tambah kategori aktivitas	berhasil menampilkan form tambah kategori aktivitas	valid
21	Hapus Data Kategori Aktivitas	klik tombol "hapus" pada daftar kategori aktivitas yang dipilih	mengahpus data kategori pada daftar kategori aktivitas yang dipilih	berhasil mengahpus data kategori pada daftar kategori aktivitas yang dipilih	valid
22	Edit Data Kategori Aktivitas	klik tombol "edit" pada daftar kategori aktivitas yang dipilih, lalu isi perubahan data kategori aktivitas	mengedit data kategori pada daftar kategori aktivitas yang dipilih	berhasil mengedit data kategori pada daftar kategori aktivitas yang dipilih	valid
23	Lihat Daftar Pengajuan Aktivitas Mahasiswa	pilih menu aktivitas mahasiswa	menampilkan daftar data pengajuan aktivitas mahasiswa	berhasill menampilkan data daftar pengajuan aktivitas mahasiswa	valid
24	Lihat Detail Bukti Aktivitas Mahasiswa	klik tombol "lihat bukti" pada daftar pengajuan aktivitas mahasiswa yang dipilih	menampilkan file bukti pengajuan aktivitas mahasiswa	berhasill menampilkan file bukti pengajuan aktivitas mahasiswa	valid
25	Edit Status Verifikasi Pengajuan Aktivitas	klik status pada daftar pengajuan aktivitas mahasiswa yang dipilih, lalu pilih opsi status pengajuan	mengubah status verifikasi kemahasiswaan pengajuan aktivitas mahasiswa	berhasill mengubah status verifikasi kemahasiswaan pengajuan aktivitas mahasiswa	valid
26	Lihat Daftar Pengajuan Proposal	pilih menu proposal	menampilkan halaman proposal beserta daftar data pengajuan proposal	berhasil menampilkan halaman proposal beserta daftar data pengajuan proposal	valid
27	Lihat Detail Bukti pengajuan Proposal	klik tombol "lihat bukti" pada daftar pengajuan proposal yang dipilih	menampilkan file bukti pengajuan proposal	berhasil menampilkan file bukti pengajuan proposal	valid

28	Edit Status Verifikasi Pengajuan Proposal	klik status pada daftar pengajuan proposal yang dipilih, lalu pilih opsi status proposal	mengubah status pengajuan proposal pada daftar data pengajuan proposal yang dipilih	berhasil mengubah status pengajuan proposal pada daftar data pengajuan proposal yang dipilih	valid
29	Lihat Daftar Tracer Alumni	klik menu "tracer alumni"	menampilkan halaman <i>tracer</i> alumni beserta daftar data kuesioner	berhasil menampilkan halaman <i>tracer</i> alumni beserta daftar data kuesioner	valid
30	Tambah Data Tracer Alumni	klik tombol "buat kuesioner"	menampilkan form tambah kuesioner	berhasil menampilkan form tambah kuesioner	valid
		mengisi form kuesioner	menyimpan data kuesioner	berhasil menyimpan data kuesioner	valid
31	Edit Data Tracer Alumni	klik tombol "edit" pada daftar data kuesioner yang dipilih, lalu ubah data kuesioner	menyimpan perubahan data kuesioner	berhasil menyimpan perubahan data kuesioner	valid
32	Hapus Data Tracer Alumni	klik tombol "hapus" pada daftar kuesioner yang dipilih	menampilkan kofirmasi hapus	berhasil menampilkan kofirmasi hapus	valid
		pilih "ya"	menghapus kuesioner	berhasil menghapus kuesioner	valid
Admin					
33	Login	Mengisi email dan passoword yang valid	Login berhasil	Login berhasil, Masuk ke halaman utama	valid
		mengisi email dan passoword yang tidak valid	Gagal login	Gagal login, menampilkan informasi email atau password salah	valid
34	Logout	klik tombol "Logout"	berhasil logout	berhasil logout, kembali ke halaman login	valid
35	Lihat Daftar Program Studi	pilih menu "program studi"	menampilkan halaman program studi berserta	berhasil menampilkan halaman program	valid

			daftar data program studi	studi berserta daftar data program studi	
36	Tambah Data Daftar Program Studi	Klik tombol "Tambah Program Studi" dan isi form yang muncul	Menyimpan data program studi baru	Berhasil menyimpan data program studi baru	Valid
37	Edit Data Program Studi	Klik tombol "Edit" pada daftar program studi yang dipilih, lalu ubah data	Menyimpan perubahan data program studi	Berhasil menyimpan perubahan data program studi	Valid
38	Hapus Data Program Studi	Klik tombol "Hapus" pada daftar program studi yang dipilih	Menampilkan konfirmasi hapus dan menghapus data program studi	Berhasil menampilkan konfirmasi hapus dan menghapus data program studi	Valid
39	Lihat Daftar Tahun Akademik	Klik menu "Tahun Akademik"	Menampilkan halaman tahun akademik beserta daftar data tahun akademik	Berhasil menampilkan halaman tahun akademik beserta daftar data tahun akademik	Valid
40	Tambah Data Tahun Akademik	Klik tombol "Tambah Tahun Akademik" dan isi form yang muncul	Menyimpan data tahun akademik baru	Berhasil menyimpan data tahun akademik baru	Valid
41	Edit Data Tahun Akademik	Klik tombol "Edit" pada daftar tahun akademik yang dipilih, lalu ubah data	Menyimpan perubahan data tahun akademik	Berhasil menyimpan perubahan data tahun akademik	Valid
42	Hapus Data Tahun Akademik	Klik tombol "Hapus" pada daftar tahun akademik yang dipilih	Menampilkan konfirmasi hapus dan menghapus data tahun akademik	Berhasil menampilkan konfirmasi hapus dan menghapus data tahun akademik	Valid
43	Lihat Daftar Unit Kegiatan Mahasiswa	Klik menu "Unit Kegiatan Mahasiswa"	Menampilkan halaman unit kegiatan mahasiswa beserta daftar data UKM	Berhasil menampilkan halaman unit kegiatan mahasiswa beserta daftar data UKM	Valid

44	Tambah Data Unit Kegiatan Mahasiswa	Klik tombol "Tambah UKM" dan isi form yang muncul	Menyimpan data unit kegiatan mahasiswa baru	Berhasil menyimpan data unit kegiatan mahasiswa baru	Valid
45	Edit Data Unit Kegiatan Mahasiswa	Klik tombol "Edit" pada daftar UKM yang dipilih, lalu ubah data	Menyimpan perubahan data unit kegiatan mahasiswa	Berhasil menyimpan perubahan data unit kegiatan mahasiswa	Valid
46	Hapus Data Unit Kegiatan Mahasiswa	Klik tombol "Hapus" pada daftar UKM yang dipilih	Menampilkan konfirmasi hapus dan menghapus data unit kegiatan mahasiswa	Berhasil menampilkan konfirmasi hapus dan menghapus data unit kegiatan mahasiswa	Valid
47	Lihat Daftar Dosen	Klik menu "Dosen"	Menampilkan halaman dosen beserta daftar data dosen	Berhasil menampilkan halaman dosen beserta daftar data dosen	Valid
48	Tambah Data Dosen	Klik tombol "Tambah Dosen" dan isi form yang muncul	Menyimpan data dosen baru	Berhasil menyimpan data dosen baru	Valid
49	Edit Data Dosen	Klik tombol "Edit" pada daftar dosen yang dipilih, lalu ubah data	Menyimpan perubahan data dosen	Berhasil menyimpan perubahan data dosen	Valid
50	Hapus Data Dosen	Klik tombol "Hapus" pada daftar dosen yang dipilih	Menampilkan konfirmasi hapus dan menghapus data dosen	Berhasil menampilkan konfirmasi hapus dan menghapus data dosen	Valid
51	Lihat Daftar Mahasiswa	Klik menu "Mahasiswa"	Menampilkan halaman mahasiswa beserta daftar data mahasiswa	Berhasil menampilkan halaman mahasiswa beserta daftar data mahasiswa	Valid
52	Tambah Data Mahasiswa	Klik tombol "Tambah Mahasiswa" dan isi form yang muncul	Menyimpan data mahasiswa baru	Berhasil menyimpan data mahasiswa baru	Valid

53	Edit Data Mahasiswa	Klik tombol "Edit" pada daftar mahasiswa yang dipilih, lalu ubah data	Menyimpan perubahan data mahasiswa	Berhasil menyimpan perubahan data mahasiswa	Valid
54	Hapus Data Mahasiswa	Klik tombol "Hapus" pada daftar mahasiswa yang dipilih	Menampilkan konfirmasi hapus dan menghapus data mahasiswa	Berhasil menampilkan konfirmasi hapus dan menghapus data mahasiswa	Valid

BAB V

KESIMPULAN DAN PENUTUP

5.1 Kesimpulan

Berdasarkan hasil yang didapat dari penelitian dan pembahasan yang telah dilakukan, dapat dibuat kesimpulan sebagai berikut:

1. Sistem informasi yang dirancang untuk mengelola pengajuan angka kredit aktivitas mahasiswa, pengajuan proposal unit kegiatan mahasiswa, dan *tracer* alumni di STMIK "AMIKBANDUNG" telah berhasil dikembangkan. Sistem ini mampu memberikan solusi yang efisien menggantikan proses manual yang selama ini digunakan, serta meningkatkan kemampuan pengelolaan data yang lebih baik.
2. Implementasi sistem informasi ini telah memudahkan mahasiswa dalam mengakses informasi terkait status pengajuan kredit aktivitas mereka, pengajuan proposal unit kegiatan mahasiswa, dan informasi *tracer* alumni. Dengan adanya sistem yang terintegrasi, mahasiswa dapat dengan lebih mudah memantau perkembangan pengajuan mereka.
3. Hasil pengujian sistem menunjukkan bahwa sistem informasi kemahasiswaan STMIK "AMIKBANDUNG" berfungsi dengan baik dalam mendukung proses administrasi akademik, sehingga dapat digunakan untuk meningkatkan pengalaman mahasiswa

5.2 Saran

Dari penelitian yang telah dilakukan mengenai sistem informasi kemahasiswaan di STMIK "AMIKBANDUNG", masih terdapat beberapa aspek yang perlu diperhatikan untuk pengembangan lebih lanjut seperti penambahan fitur notifikasi real-time yang sebaiknya diprioritaskan dalam pengembangan sistem selanjutnya, agar mahasiswa dapat menerima informasi terkini mengenai status pengajuan mereka dengan lebih cepat dan efisien. Fitur ini akan sangat membantu dalam meningkatkan komunikasi antara mahasiswa dan pihak administrasi.

DAFTAR PUSTAKA

- [1] A. J. Oktasari and D. Kurniadi, “Perancangan Sistem Informasi Manajemen Kegiatan Mahasiswa Berbasis Web,” *Voteteknika (Vocational Tek. Elektron. dan Inform.*, vol. 7, no. 4, p. 149, 2020, doi: 10.24036/voteteknika.v7i4.106536.
- [2] M. L. Gaol and Sardjono, “Rancang Bangun Sistem Informasi Perhitungan Satuan Kredit Kegiatan Mahasiswa (SKKM),” vol. 17, pp. 2614–5405, 2023, [Online]. Available: <https://journal.uniku.ac.id/index.php/ilkom>
- [3] I. G. Agus Satya Dharma, D. Care Khrisne, and I. M. Arsa Suyadnya, “Rancang Bangun Sistem Informasi Satuan Kredit Partisipasi Mahasiswa Fakultas Teknik Universitas Udayana,” *J. SPEKTRUM*, vol. 4, no. 2, p. 81, 2018, doi: 10.24843/spektrum.2017.v04.i02.p11.
- [4] R. Kurniawan and D. Prananda Putra, “Perancangan *User Interface* Sistem Kredit Aktivitas Mahasiswa STMIK ‘AMIKBANDUNG’ Berbasis Website Menggunakan Metode *User Centered Design (UCD)*,” *J. Inf. Technol.*, vol. 4, no. 1, pp. 23–30, 2022, doi: 10.47292/joint.v4i1.77.
- [5] A. Syarif, “Prototipe Sistem Informasi Penilaian Prestasi Mahasiswa Program Studi Sekretari Berbasis Web,” *J. Sekr. Adm.*, vol. 18, no. 2, pp. 46–56, 2020, [Online]. Available: <https://journal.budiluhur.ac.id/index.php/serasi/article/view/1259>
- [6] A. Afriansyah, “Sistem Informasi Transkrip Aktivitas Kemahasiswaan (SITAKIS) Fakultas Ilmu Komputer Universitas Lancang Kuning,” *J. Sains, Teknol. dan Ind.*, vol. 18, no. 1, p. 11, 2020, doi: 10.24014/sitekin.v18i1.11137.
- [7] S. Yanti, D. Kriestasnto, and H. A. Triyanto, “Pengukuran Capability Level Sistem Penilaian Prestasi Mahasiswa Stmik Akakom Menggunakan *Framework Cobit 5*,” *J. Inform. Komputer, Bisnis dan Manaj.*, vol. 21, no. 1, pp. 1–11, 2023, doi: 10.61805/fahma.v21i1.20.
- [8] Y. Hanafi, K. Anam, M. A. Ikhsan, M. Saefi, and T. N. Diyana, “Pesantren Transformatif M E M B A N G U N E K O S I S T E M D I G I T A L,” p. 4, 2022.
- [9] Y. Hanafi, K. Anam, M. A. Ikhsan, M. Saefi, and T. N. Diyana, “Pesantren Transformatif M E M B A N G U N E K O S I S T E M D I G I T A L,” p. 4, 2022.
- [10] A. Heryana, “Teori dan Jenis Sistem,” *Artik. Kepemimp. Berfikir Sist.*, no. 4, p.

39, 2021, doi: 10.13140/RG.2.2.12029.49126.

- [11] F. Fachruddin, M. Syukri, A. Maulidya, and D. Syahputra, “Klasifikasi Sistem dan Hubungan sebagai Inti dari Sistem,” *Transform. Manag. Journal Islam. Educ. Manag.*, vol. 3, no. 2, pp. 535–542, 2023, doi: 10.47467/manageria.v3i2.3182.
- [12] S. R. Rahadani, Z. Arifin, and E. Sumarya, “Analisa Dan Desain Sistem Informasi Manajemen Dengan Metode Sdlc (System Development Life Cycle) Guna Mengurangi Waktu Kerusakan Mesin Welding,” *PROFISIENSI J. Progr. Stud. Tek. Ind.*, vol. 9, no. 2, pp. 291–297, 2021, doi: 10.33373/profis.v9i2.3698.
- [13] A. Abdul Wahid, “Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi,” *J. Ilmu-ilmu Inform. dan Manaj. STMIK*, no. November, pp. 1–5, 2020.
- [14] F. Rifandi, Tri Viqi Adriansyah, and Rina Kurniawati, “Website Gallery Development Using Tailwind CSS Framework,” *J. E-Komtek*, vol. 6, no. 2, pp. 205–214, 2022, doi: 10.37339/e-komtek.v6i2.937.
- [15] N. Wilyanto, J. Firnando, B. Franko, S. P. Tanzil, H. C. Tan, and E. Hartati, “Pembuatan Website Menggunakan *Visual Studio Code* di SMA Xaverius 3 Palembang,” *Fordicate*, vol. 3, no. 1, pp. 1–8, 2023.
- [16] Djoni Haryadi Setiabudi and Ibnu Gunawan, “Studi Penggunaan Visual Studio 6.0 Untuk Pengembangan Sistem Informasi Berkelas Enterprise,” *J. Inform.*, vol. 4, no. 1, pp. 27–34, 2003, [Online]. Available: <http://puslit2.petra.ac.id/ejournal/index.php/inf/article/view/15829>
- [17] A. Christopher, S. Anjelina, and Lina, “Analisis Performa *Framework Frontend Javascript* Bagi Programmer,” *J. Ilmu Komput.*, vol. 7, no. 2, pp. 98–108, 2022.
- [18] I. K. A. H. Putra, D. Pramana, and N. L. P. Srinadi, “Sistem Manajemen Arsip Menggunakan *Framework Laravel* dan *Vue.Js* (Studi Kasus : BPKAD Provinsi Bali),” *J. Sist. Dan Inform.*, pp. 97–104, 2017.
- [19] Clinton Hatta Pradigi, Tintin Harlina, and Solehatin, “Implementasi Express Js Untuk Membangun Rest Api Website Stikom Pgri Banyuwangi,” *Jikom J. Inform. dan Komput.*, vol. 9, no. 2, pp. 118–122, 2022, doi: 10.55794/jikom.v9i2.68.
- [20] Normah, B. Rifai, S. Vambudi, and R. Maulana, “Analisa Sentimen Perkembangan Vtuber Dengan Metode Support Vector Machine Berbasis SMOTE,” *J. Tek. Komput. AMIK BSI*, vol. 8, no. 2, pp. 174–180, 2022, doi:

10.31294/jtk.v4i2.

- [21] Y. D. Wijaya and M. W. Astuti, “Pengujian Blackbox Sistem Informasi Penilaian Kinerja Karyawan Pt Inka (Persero) Berbasis Equivalence Partitions,” *J. Digit. Teknol. Inf.*, vol. 4, no. 1, p. 22, 2021, doi: 10.32502/digital.v4i1.3163.
- [22] S. L. M. Sitio, D. Y. Tanu, S. Solihin, A. Saifudin, and T. Desyani, “Pengujian Blackbox pada Website Open Jurnal Universitas Pamulang Menggunakan Teknik Cause-Effect Relationship Testing,” *J. Inform. Univ. Pamulang*, vol. 8, no. 1, pp. 102–106, 2023, doi: 10.32493/informatika.v8i1.26885.
- [23] W. N. Cholifah, Y. Yulianingsih, and S. M. Sagita, “Pengujian Black Box Testing pada Aplikasi Action & Strategy Berbasis Android dengan Teknologi Phonegap,” *STRING (Satuan Tulisan Ris. dan Inov. Teknol.)*, vol. 3, no. 2, p. 206, 2018, doi: 10.30998/string.v3i2.3048.
- [24] Y. F. Achmad and A. Yulfitri, “Pengujian Sistem Pendukung Keputusan Menggunakan Black Box Testisng Studi Kasus E-Wisudawan Di Institus Sains Dan Teknologi Al-Kamal,” *J. Ilmu Komput.*, vol. 5, p. 42, 2020.
- [25] T. Hartono, “UNIT KEGIATAN MAHASISWA DAN PERANNYA DALAM MEMBENTUK KARAKTER MAHASISWA (Studi Kasus di Unit Kegiatan Mahasiswa Jam’iyyah al-Qurra’ wa al-Huffazh Al-Furqan Institut Agama Islam Negeri Salatiga),” *POTENSIJA J. Kependidikan Islam*, vol. 5, no. 1, p. 99, 2019, doi: 10.24014/potensia.v5i1.6521.
- [26] Q. Mardzotillah and M. Ridwan, “Sistem *Tracer Study* Dan Persebaran Alumni Berbasis Web Di Universitas Islam Syekh-Yusuf Tangerang,” *Jutis (Jurnal Tek. Inform.)*, vol. 8, no. 1, pp. 90–106, 2020, [Online]. Available: <http://ejournal.unis.ac.id/index.php/jutis/article/view/705>

LAMPIRAN

Lampiran 1: Wawancara Pertama Dengan Pihak Kemahasiswaan STMIK“AMIK BANDUNG”

Hari/Tanggal : Selasa, 19 November 2024
 Tempat : STMIK”AMIKBANDUNG”
 Pukul : 09.00 WIB – Selesai

No	Uraian Wawancara	Pihak
1.	Yuda Saputra (Mahasiswa) <ol style="list-style-type: none"> Apa saja permasalahan yang sedang dihadapi oleh kemahasiswaan saat ini? 	Pewawancara: Mahasiswa STMIK”AMIK Bandung”
2.	Ibu Yolanda Georgia, S.Sn., M.Ds. (Kemahasiswaan) <ol style="list-style-type: none"> Ada banyak permalsahan yang sedang dihadapi oleh kemahasiswaan yaitu Angka Kredit Aktivitas Mahasiswa (AKAM), pemberitahuan dan pengisian berkas oleh mahasiswa yang masih menggunakan google form, serta masalah pendaftaran. Namun yang perlu diutamakan saat ini adalah Angka Kredit Aktivitas Mahasiswa (AKAM). 	Narasumber: Kemahasiswaan STMIK”AMIK Bandung”
3.	Yuda Saputra (Mahasiswa) <ol style="list-style-type: none"> Baik, jadi untuk sekarang yang lebih <i>urgen</i> adalah AKAM itu sendiri bu? 	Pewawancara: Mahasiswa STMIK”AMIK Bandung”
4.	Ibu Yolanda Georgia, S.Sn., M.Ds. (Kemahasiswaan) <ol style="list-style-type: none"> Betul, karena kegiatan AKAM ini sendiri dilakukan tidak hanya satu kali dalam satu tahun namun 2 kali karena dilakukan pada akhir semester dan untuk seluruh mahasiswa. 	Narasumber: Kemahasiswaan STMIK”AMIK Bandung”
5.	Yuda Saputra (Mahasiswa) <ol style="list-style-type: none"> Baik untuk itu dalam penelitian isi saya akan focus pada pengembangan AKAM terlebih dahulu. Faktor apa yang mendorong sehingga aplikasi kemahasiswaan ini perlu dibuat? 	Pewawancara: Mahasiswa STMIK”AMIK Bandung”
6.	Ibu Yolanda Georgia, S.Sn., M.Ds. (Kemahasiswaan) <ol style="list-style-type: none"> Pada saat ini mahasiswa dituntut untuk aktif selain hard skill juga soft skill. Serta selain dari pembelajaran mata kuliah juga mahasiswa dituntut untuk proaktif secara organisasi juga kegiatan liar termasuk dalam segi prestasi. 	Narasumber: Kemahasiswaan STMIK”AMIK Bandung”

No	Uraian Wawancara	Pihak
	<p>3. AKAM sendiri merupakan sistem yang merekap setiap kegiatan mahasiswa sehingga mendorong untuk rutin beraktivitas dan tidak hanya sebatas pada belajar.</p> <p>4. Sehingga berdasarkan hal tersebut terdapat beberapa kendala:</p> <ul style="list-style-type: none"> a. Kendala yang pertama adalah pengajuan AKAM dilakukan setiap akhir semester sehingga dalam periode satu tahun akan dilakukan 2 kali, AKAM terebut akan direkap dan dimasukkan dalam transkip SKIP (Surat Keterangan Pendamping ijazah). Namun selama ini rekап dilakukan secara manual sehingga pihak kemahasiswaan cukup kesulitan dikarnakan kegiatan yang diajukan mahasiswa cenderung banyak dan harus direkap ulang secara manual ke digital selama 8 semester serta jumlah mahasiswa yang harus direkap berjumlah ratusan; b. Kendala kedua adalah Dosen wali tidak berada di kampus setiap harinya sehingga Ketika mahasiswa ingin mengajukan verifikasi kegiatan, sulit untuk bertemu dan dosen sulit untuk melakukan verifikasi karena berkas yang diajukan harus berupa dokumen fisik, apabila dokumen pengajuan telah berupa dokumen digital maka akan memudahkan dosen wali dalam melakukan verifikasi karena dapat dilakukan Dimana saja dan kapan saja tidak mengharuskan untuk ke kampus serta lebih ramah lingkungan karena tidak perlu mencetak dokumen dan da pihak kemahasiswaan sendiri dapat langsung mengirim notifikasi kepada mahasiswa. c. Proses pemindahan Format SKPI juga sudah lebih mudah jika terformat digital sedari awal. 	
7.	<p>Yuda Saputra (Mahasiswa)</p> <p>1. Pada pengembangan aplikasi kemahasiswaan ini ada noda berapa <i>actor</i> yang terlibat dan siapa saja?</p>	Pewawancara: Mahasiswa STMIK"AMIK Bandung"
8.	<p>Ibu Yolanda Georgia, S.Sn., M.Ds. (Kemahasiswaan)</p> <p>1. Aktor yang terlibat ada 3 yaitu mahasiswa, dosen wali, serta kemahasiswaan itu sendiri.</p>	Narasumber: Kemahasiswaan STMIK"AMIK Bandung"
9.	<p>Yuda Saputra (Mahasiswa)</p> <p>1. Apa saja yang dapat dilakukan oleh setiap <i>actor</i> didalam sistem atau aplikasi yang akan dikembangkan?</p>	Pewawancara: Mahasiswa STMIK"AMIK Bandung"
11.	<p>Ibu Yolanda Georgia, S.Sn., M.Ds. (Kemahasiswaan)</p> <p>1. Untuk mahasiswa lebih kearah memilih kategori kegiatan, kemudian mengupload sesuai dengan kategori yang dipilih, serta masiswa itu setelah diverifikasi mendapat notifikasi perolehan</p>	Narasumber: Kemahasiswaan STMIK"AMIK Bandung"

No	Uraian Wawancara	Pihak
	<p>poin yang diterima dengan rekapan poin untuk seluruh semester yang sudah diajukan.</p> <p>2. Dosen wali khusus untuk melakukan pengecekan dokumen dengan nada <i>button checklist</i> jika mahasiswa telah mengirimkan dokumen, serta dosen sendiri memiliki fitur upload sertifikat tersendiri.</p> <p>3. Kemahasiswaan akan melakukan pengecekan atas aktivitas <i>checklist</i> yang sudah dilakukan oleh dosen wali dan akan melakukan verifikasi dengan <i>button checklist</i> juga, sehingga ada 2 <i>button checklist</i>, satu untuk dosen wali dan satu untuk kemahasiswaan.</p> <p>4. Jika seluruh proses telah selesai maka kemahasiswaan akan mengirimkan notifikasi kepada mahasiswa.</p>	
12.	<p>Yuda Saputra (Mahasiswa)</p> <p>1. Apakah untuk setiap berkas yang di <i>upload</i> memiliki poin tersendiri?</p>	<p>Pewawancara:</p> <p>Mahasiswa STMIK "AMIK Bandung"</p>
13.	<p>Ibu Yolanda Georgia, S.Sn., M.Ds. (Kemahasiswaan)</p> <p>1. Betul, jadi masing-masing berkas yang diupload memiliki kategori tersendiri misalnya seminar, workshop internal, dan kegiatan lainnya dengan jumlah sertifikat yang tidak dibatasi.</p> <p>2. Kalau bisa Ketika telah dichecklist oleh dosen wali dan kemahasiswaan maka poinnya dapat otomatis terakumulasi berdasarkan kategori yang telah dipilih oleh mahasiswa.</p>	<p>Narasumber:</p> <p>Kemahasiswaan STMIK "AMIK Bandung"</p>
14.	<p>Yuda Saputra (Mahasiswa)</p> <p>1. Apakah ada aturan mengenai waktu penerbitan sertifikat apabila sertifikat pada tahun kemarin baru diupload pada semester terbaru?</p>	<p>Pewawancara:</p> <p>STMIK "AMIK Bandung"</p>
15.	<p>Ibu Yolanda Georgia, S.Sn., M.Ds. (Kemahasiswaan)</p> <p>1. Sertifikat yang diupload merupakan sertifikat yang kegiatannya dilakukan pada saat semester aktif saja, kalau bisa pada fiturnya ada tanggal kegiatan, jadi sebelum mahasiswa mengupload berkas mereka terlebih dahulu harus memilih tanggal kegiatan dan apabila tanggal yang dipilih lewat dari semester aktif pada saat itu maka berkas tidak dapat diupload, apakah bisa dilakukan?</p>	<p>Narasumber:</p> <p>STMIK "AMIK Bandung"</p>
16.	<p>Yuda Saputra (Mahasiswa)</p> <p>1. Mungkin karena berkasnya berbentuk file upload maka kemahasiswaan perlu melakukan pengecekan Kembali karena meskipun ada fitur yang demikian namun berkas yang diupload bisa saja merupakan berkas dengan tanggal yang berbeda dari tanggal yang dipilih oleh mahasiswa dalam aplikasi sehingga perlu adanya verifikasi Kembali.</p>	<p>Pewawancara:</p> <p>Mahasiswa STMIK "AMIK Bandung"</p>
17.	<p>Ibu Yolanda Georgia, S.Sn., M.Ds. (Kemahasiswaan)</p>	<p>Narasumber:</p>

No	Uraian Wawancara	Pihak
	<p>1. Baik, dengan begitu dari kemahasiswaan akan melakukan pengecekan dan checklist.</p> <p>2. Terkait notifikasi point yang diperoleh oleh mahasiswa dibuatkan script yang isinya poin poin setiap semester dan kemudian akumulasi dari semua semester yang telah ditempuh oleh mahasiswa, contohnya semester 1 ada berapa poin, semester 2 ada berapa poin , dan akumulasi dari semster 1-8 ada berapa poin.</p>	Kemahasiswaan STMIK”AMIK Bandung”
18.	<p>Yuda Saputra (Mahasiswa)</p> <p>1. Baik, jika demikian maka kemahasiswaan yang memberikan kabar kepada mahasiswa terkait waktu pengupload an sehingga dari informasi itu tombol upload pada aplikasi dapat langsung diaktifkan.</p>	Pewawancara: Mahasiswa STMIK”AMIK Bandung”
19.	<p>Ibu Yolanda Georgia, S.Sn., M.Ds. (Kemahasiswaan)</p> <p>1. Baik dengan satu cacatan lagi bahwa pada saat mahasiswa melakukan upload berkas mereka harus terlebih dahulu menuliskan nama kegiatannya serta ada fitur yang sebelumnya telah kita bahas yaitu mengenai tanggal.</p>	Narasumber: Kemahasiswaan STMIK”AMIK Bandung”
20.	<p>Yuda Saputra (Mahasiswa)</p> <p>1. Baik bu, jika sudah selesai maka saya akhir wawancara ini, terimakasih atas waktu dan kesempatanntya.</p>	Pewawancara: Mahasiswa STMIK”AMIK Bandung”

Lampiran 2: Wawancara Kedua Dengan Pihak Kemahasiswaan STMIK “AMIK BANDUNG”

Tanggal : 03, Januari 2025

Responden : Ibu Yolanda Georgia, S.Sn., M.Ds.

Media : Google Meet (Online)

No	Pertanyaan
1	Apakah ada peraturan khusus mengenai batas maksimal poin kredit aktivitas yang bisa dikumpulkan dalam satu semester atau tahun ajaran?
2	Jika terdapat mahasiswa baru, dosen baru, atau terjadi perubahan dosen wali dari seorang mahasiswa, siapa yang akan bertanggung jawab untuk mendaftarkannya ke dalam sistem dan merelasikannya dengan dosen wali? Apakah tugas ini dilakukan oleh bagian kemahasiswaan, atau diperlukan admin khusus untuk menangani hal tersebut?
3	Karena kredit aktivitas yang dapat dikumpulkan adalah berdasarkan tanggal aktivitas saat di periode tahun ajaran saat pengumpulan, apakah mahasiswa dapat mengumpulkan kredit aktivitas kapan saja? Atau ada waktu tertentu untuk pengumpulan?
4	Apakah akan ada jenis aktivitas baru atau perubahan, baik dari nama aktivitas maupun dari poin yang diberikan untuk jenis aktivitas tertentu? Jika ada, siapa yang akan mengurnya, apakah bagian kemahasiswaan atau admin?
5	Apakah sistem perlu memiliki fitur laporan rekapitulasi? Apakah laporan ini hanya untuk mahasiswa (personal) atau juga untuk admin/dosen wali (kolektif)?

No	Jawaban
1	Untuk pengajuan minimal 70 poin dan tidak ada maksimal namun bisa berubah seiring waktu
2	tugas untuk mendaftarkan mahasiswa baru dan merelasikannya dengan dosen wali akan dilakukan oleh admin. Ada pembicaraan mengenai admin yang menambah tahun ajaran dan mengelola data mahasiswa.
3	Pengumpulan kredit aktivitas setiap haru harus dilakukan setiap akhir semester dan tidak dilain waktu dan tidak harus bersamaan karena ada periode pembukaannya.
4	Untuk jenis aktivitas baru belum ada namun kepastiannya pasti akan dan yang akan mengurnya adalah admin itu sendiri.
5	Sistem memerlukan fitur laporan rekapitulasi, namun itu hanya dikhususkan untuk kemahasiswaan saja.

Lampiran 3: Wawancara Ketiga Dengan Pihak Kemahasiswaan STMIK “AMIK BANDUNG”

Tanggal : 25 Januari 2025

Responden : Ibu Yolanda Georgia, S.Sn., M.Ds.

Tempat : STMIK”AMIKBANDUNG”

Pukul : 09.00 WIB – Selesai

No	Pertanyaan
1	Berhubungan dengan pengajuan proposal, apakah ada syarat tertentu yang harus dipenuhi dalam mengajukan proposal kegiatan baru?
2	Apakah ada batasan ukuran file yang dapat diunggah oleh mahasiswa saat mengajukan kegiatan?
3	Bagaimana alur pengajuan proposal UKM? Apakah ada tahapan tertentu yang harus dilalui?
4	Bagaimana yang dimaksud dengan <i>tracer</i> alaumni?

No	Jawaban
1	Sebelum UKM mengajukan prosal kegiatan baru maka meraka harus mengajukan Laporan Pertanggungjawaban terlebih dahulu dari kegiatan sebelumnya yang telah disetujui.
2	Batasa dalam mengunggah file baik pengajuan kredit maupun proposal adalah 2 MB.
3	Alur pengajuan proposal adalah melalui BEM dan penajuannya itu manual, unit kegiatan mahasiswa mengajukan melalui bebm kemudian bem menyalurkan proposalnya ke pihak kemahasiswaan dan apabila di setujui makan proposal akan dimaskukkan ke dalam sistem sebagai kebutuhan pengarsipan saja, dan alam pengajuan proposal harus memuat : <ol style="list-style-type: none"> 1. Proposal kegiata 2. Surat pertanggungjawaban 3. Permohonan dispensasi 4. SK panitia 5. Daftar peserta 6. Pemberitahuan kegiatan kepada pihak terkait
4	Untuk fitur <i>tracer</i> alumni yaitu untuk penyebaran link kuesioner yang akan disi oleh alumni untuk tahun kelulusan tertentu berdasarkan kebutuhan <i>tracker study</i> kemahasiswaan.