



**RANCANG BANGUN *WORKFLOW* LAYANAN ADMINISTRASI
AKADEMIK DI UNIVERSITAS DINAMIKA**

TUGAS AKHIR



UNIVERSITAS
Dinamika

Oleh :

FADHLI DZIL IKHRAM ISVIANTO PRAKOSO

17410100019

FAKULTAS TEKNOLOGI DAN INFORMATIKA

UNIVERSITAS DINAMIKA

2022

**RANCANG BANGUN *WORKFLOW* LAYANAN ADMINISTRASI
AKADEMIK DI UNIVERSITAS DINAMIKA**

TUGAS AKHIR

**Diajukan sebagai syarat untuk menyelesaikan
Program Sarjana**



UNIVERSITAS
Dinamika

Oleh :

Nama : Fadhli Dzil Ikham Isvianto Prakoso

NIM : 17410100019

Program Studi : Sistem Informasi

**FAKULTAS TEKNOLOGI DAN INFORMATIKA
UNIVERSITAS DINAMIKA**

2022

TUGAS AKHIR

RANCANG BANGUN *WORKFLOW* LAYANAN ADMINISTRASI AKADEMIK DI UNIVERSITAS DINAMIKA

Dipersiapkan dan disusun oleh
Fadhli Dzil Ikham Isvianto Prakoso
NIM: 17410100019

Telah diperiksa, dibahas dan disetujui oleh Dewan Pembahas

Pada: Selasa, 23 Agustus 2022

Susunan Dewan Pembahas

Pembimbing:

- I. Erwin Sutomo, S.Kom., M.Eng.
NIDN. 0722057501
- II. I Gusti Ngurah Alit W. P., S.T., M.Eng.
NIDN. 0805058602

Pembahas:

- I. Julianto Lemantara, S.Kom., M.Eng.
NIDN. 0722108601

Digitally signed by
Erwin
DN: cn=Erwin,
o=Universitas
Dinamika, ou=PPTI,
email=sutomo@dina
mika.ac.id, c=ID
Date: 2022.08.23
15:51:27 +07'00'

Digitally signed
by I Gusti Ngurah
Alit Widana Putra
Date: 2022.08.24
08:11:43 +07'00'

Digitally signed
by Julianto
Date: 2022.08.24
09:07:56 +07'00'

Tugas Akhir ini telah diterima sebagai salah satu persyaratan

untuk memperoleh gelar Sarjana



Digitally signed by
Universitas Dinamika
Date: 2022.08.24
09:42:14 +07'00'

Tri Sagirani, S.Kom., M.MT.

NIDN. 0731017601

Dekan Fakultas Teknologi dan Informatika
UNIVERSITAS DINAMIKA



A person who never

Made a mistake

Never tried anything new.

- Albert Einstein

UNIVERSITAS
Dinamika



*Ku persembahkan kepada
Ibu & Bapak, Serta teman, sahabat yang*

Selalu memberikan semangat disetiap perjuangan ku

UNIVERSITAS
Dinamika

SURAT PERNYATAAN
PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH

Sebagai mahasiswa Universitas Dinamika, saya :

Nama : Fadhli Dzil Ikhran Isvianto Prakoso
NIM : 17410100019
Program Studi : S1 Sistem Informasi
Email : 17410100019@dinamika.ac.id
Alamat : Taman Pinang Indah D3 no.35, Sidoarjo
Jenis Karya : Tugas Akhir
Judul Karya : **RANCANG BANGUN *WORKFLOW* LAYANAN ADMINISTRASI
AKADEMIK DI UNIVERSITAS DINAMIKA**

Menyatakan dengan sesungguhnya bahwa:

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan seni, saya menyetujui memberikan kepada Universitas Dinamika Hak Bebas Royalti Non-Eksklusif (*Non-exclusive Royalti free righ*) atas seluruh isi/sebagian karya ilmiah saya tersebut di atas untuk disimpan, dialihmediakan dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencampurkan nama saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta.
2. Karya tersebut di atas adalah karya asli saya, bukan plagiat baik sebagian maupun keseluruhan Kutipan, karya atau pendapat orang lain yang ada dalam karya ilmiah ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya.
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiat pada karya ilmiah ini, maka saya bersedia untuk menerima pencabutan terhadap gelar keserjanaan yang telah diberikan kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, 06 Juli 2022

Yang menyatakan



Fadhli Dzil Ikhran Isvianto Prakoso

NIM: 17410100019

ABSTRAK

Universitas Dinamika memiliki bagian yang berfungsi sebagai tempat mahasiswa untuk melakukan pengajuan yang berhubungan dengan akademik maupun non-akademik, yaitu bagian Administrasi Akademik dan Kemahasiswaan (AAK) yang memiliki layanan pengajuan yang terdiri dari dispensasi sakit, yudisium, cuti/Berhenti Studi Tetap (BSS), dan Berhenti Studi Tetap (BST). Berdasarkan kuesioner I yang telah disebar ke mahasiswa Universitas Dinamika dengan jumlah responden sebanyak 30 orang, didapati indeks kepuasan mahasiswa sebesar 78% dengan *workflow* pengajuan layanan AAK sekarang. Disisi lain 84% mahasiswa menginginkan *workflow* pengajuan layanan di AAK yang dapat lebih mudah, cepat, dan membantu dalam proses pengajuan, persetujuan ke pihak terkait (dosen wali, kaprodi, bagian AAK), serta informasi progres dari pengajuan dengan bantuan aplikasi yang terintegrasi dengan kampus. Maka dari hasil kuesioner diatas, dibuat rancang bangun *workflow* pengajuan layanan AAK yang dapat membantu mahasiswa dalam mengajukan layanan tersebut dengan metode *Workflow Management System* (WMS). Metode ini dipilih karena dapat membuat *workflow* menjadi lebih sistematis dalam pendistribusian data pengajuan, informasi progres pengajuan, dan pengisian formulir pengajuan. Setelah mengadakan kuesioner II dan *demo* aplikasi untuk mendapatkan data mengenai kepuasan mahasiswa terhadap rancang bangun *workflow*, didapati hasil indeks kepuasan sebesar 89%. Dapat disimpulkan, rancang bangun *workflow* ini dapat membantu menjawab kebutuhan pengembangan pada *workflow* pengajuan layanan di AAK yang sekarang berjalan.

Kata Kunci: AAK, *workflow management system*, pengajuan, kuesioner

KATA PENGANTAR

Puji syukur selalu tercurahkan kepada Allah S.W.T. yang telah memberikan rahmat dan hidayah-Nya sehingga peneliti mampu menyelesaikan Tugas Akhir dengan judul “**RANCANG BANGUN *WORKFLOW* LAYANAN ADMINISTRASI AKADEMIK DI UNIVERSITAS DINAMIKA**”.

Penyelesaian tugas akhir ini tidak terlepas dari bantuan berbagai pihak yang telah memberikan banyak masukan, nasihat, saran, kritik, dan dukungan moral maupun materil kepada peneliti. Oleh karena itu peneliti menyampaikan rasa terima kasih kepada:

1. Bapak dan Ibu tercinta serta keluarga yang selalu mendoakan, mendukung dan memberikan semangat di setiap langkah dan aktivitas peneliti.
2. Bapak Erwin Sutomo, S.Kom., M.Eng. selaku Dosen Pembimbing 1 yang selalu membimbing, mendukung, dan memberikan motivasi kepada peneliti dalam menyelesaikan tugas akhir ini.
3. Bapak I Gusti Ngurah Alit W. P., S.T., M.Eng. selaku Dosen Pembimbing 2 dan juga selalu membimbing, mendukung, memberikan motivasi dan arahan kepada peneliti dalam menyelesaikan tugas akhir ini.
4. Bapak Julianto Lemantara, S.Kom., M.Eng. selaku Dosen Pembahas yang telah bersedia menjadi dosen pembahas dalam mengerjakan tugas akhir ini.
5. Teman-teman tercinta yang memberikan bantuan dan dukungan dalam menyelesaikan tugas akhir ini.
6. Pihak-pihak lain yang tidak dapat disebutkan satu-persatu yang telah memberikan bantuan dan dukungan kepada peneliti.

Semoga Allah S.W.T. memberikan balasan yang setimpal kepada semua pihak yang telah membantu dan memberikan bimbingan serta nasehat dalam proses menyelesaikan Tugas Akhir ini.

Peneliti menyadari bahwa Tugas Akhir yang dikerjakan ini masih banyak terdapat kekurangan sehingga kritik yang bersifat membangun dan saran dari semua pihak sangatlah diharapkan agar aplikasi ini dapat diperbaiki menjadi lebih baik lagi. Semoga Tugas Akhir ini dapat diterima dan bermanfaat bagi peneliti dan semua pihak.

Surabaya, 06 Juli 2022



Fadli Dzil Ikham Isvianto Prakoso



UNIVERSITAS
Dinamika

DAFTAR ISI

	Halaman
ABSTRAK	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	x
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan.....	2
1.5 Manfaat.....	3
BAB II LANDASAN TEORI	4
2.1 Penelitian Terdahulu.....	4
2.2 <i>Workflow Management System (WMS)</i>	5
2.3 <i>Framework</i> Pengembangan Aplikasi.....	6
2.3.1 PHP	6
2.3.2 <i>Laravel</i>	7
2.4 Model <i>SCRUM</i>	7
2.5 Metode <i>Agile</i>	8
BAB III METODOLOGI PENELITIAN	9
3.1 Tahap Awal.....	10
3.1.1 Analisis	10
3.1.2 Wawancara & Observasi	10
3.1.3 Analisis Proses Bisnis	10
3.1.4 Identifikasi Masalah.....	18
3.1.5 Analisis Kebutuhan Pengguna	19
3.1.6 <i>Product Backlog</i>	20
3.2 Tahap Pengembangan.....	21
3.2.1 <i>Sprint Planning</i>	21
3.2.2 <i>Sprint Backlog</i>	21
3.2.3 <i>Sprint</i>	23
3.3 Tahap Akhir.....	41
3.3.1 <i>Sprint Retrospective</i>	41
3.3.2 <i>Sprint Review</i>	41

BAB IV HASIL DAN PEMBAHASAN.....	42
4.1 <i>Sprint</i>	42
4.1.1 <i>Sprint</i> Iterasi Pertama	42
4.1.2 <i>Sprint</i> Iterasi Kedua	43
4.1.3 <i>Sprint</i> Iterasi Ketiga.....	44
4.2 Pembahasan Sistem	45
BAB V PENUTUP.....	47
5.1 Kesimpulan	47
5.2 Saran	47
DAFTAR PUSTAKA	48
LAMPIRAN.....	49



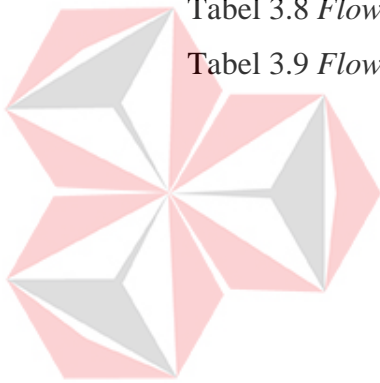
UNIVERSITAS
Dinamika

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Model <i>SCRUM</i> (Steen Lerche-Jensen, 2016)	7
Gambar 3.1 Metode Penelitian.....	9
Gambar 3.2 <i>Workflow</i> Pengajuan Yudisium.....	11
Gambar 3.3 <i>Workflow</i> Pengajuan Dispensasi sakit.....	13
Gambar 3.4 <i>Workflow</i> Pengajuan Cuti.....	15
Gambar 3.5 <i>Workflow</i> Pengajuan BST	17
Gambar 3.6 <i>Use case</i> pengajuan dispensasi sakit	24
Gambar 3.7 <i>Use case</i> pengajuan yudisium	25
Gambar 3.8 <i>Use case</i> pengajuan cuti/Berhenti Studi Sementara (BSS)	27
Gambar 3.9 <i>Use case</i> pengajuan Berhenti Studi Tetap (BST).....	28
Gambar 3.10 <i>Sequence Diagram</i> pengajuan dispensasi sakit bagian mahasiswa..	32
Gambar 3.11 <i>Sequence Diagram</i> pengajuan dispensasi sakit bagian dosen wali ..	32
Gambar 3.12 <i>Sequence Diagram</i> pengajuan dispensasi sakit bagian kaprodi	33
Gambar 3.13 <i>Sequence Diagram</i> pengajuan dispensasi sakit bagian AAK.....	34
Gambar 3.14 <i>Sequence Diagram</i> pengajuan yudisium bagian mahasiswa.....	35
Gambar 3.15 <i>Sequence Diagram</i> pengajuan yudisium bagian AAK.....	35
Gambar 3.16 <i>Sequence Diagram</i> pengajuan cuti/BSS bagian mahasiswa	36
Gambar 3.17 <i>Sequence Diagram</i> pengajuan cuti/BSS bagian dosen wali.....	37
Gambar 3.18 <i>Sequence Diagram</i> pengajuan cuti/BSS bagian AAK	37
Gambar 3.19 <i>Sequence Diagram</i> pengajuan BST bagian mahasiswa	38
Gambar 3.20 <i>Sequence Diagram</i> pengajuan BST bagian dosen wali.....	39
Gambar 3.21 <i>Sequence Diagram</i> pengajuan BST bagian kaprodi.....	40
Gambar 3.22 <i>Sequence Diagram</i> pengajuan BST bagian AAK	40
Gambar 4.1 Tampilan tabel <i>track</i> progres pengajuan cuti/BSS.....	43
Gambar 4.2 Tampilan halaman <i>upload</i> pada pengajuan yudisium.....	43
Gambar 4.3 Tampilan halaman <i>upload</i> pada pengajuan dispensasi sakit.....	44
Gambar 4.4 Tampilan notifikasi <i>sorting</i> mahasiswa di pengajuan BST.....	45
Gambar 4.5 Tampilan hasil <i>export data</i> pada pengajuan dispensasi sakit.....	45

DAFTAR TABEL

	Halaman
Tabel 2.1 Penelitian terdahulu 1.....	4
Tabel 2.2 Penelitian Terdahulu 2	5
Tabel 2.3 Penelitian Terdahulu 3	5
Tabel 3.1 Identifikasi Masalah.....	18
Tabel 3.2 Analisis Kebutuhan Pengguna	19
Tabel 3.3 <i>Product Backlog</i>	20
Tabel 3.4 <i>Sprint Backlog</i> Iterasi Pertama.....	22
Tabel 3.5 <i>Sprint Backlog</i> Iterasi Kedua	22
Tabel 3.6 <i>Sprint Backlog</i> Iterasi Ketiga	23
Tabel 3.7 <i>Flow of event</i> fitur <i>monitoring</i> pengajuan.....	29
Tabel 3.8 <i>Flow of event</i> fitur pengajuan dan <i>upload</i> berkas	30
Tabel 3.9 <i>Flow of event</i> fitur notifikasi <i>email</i>	31



UNIVERSITAS
Dinamika

DAFTAR LAMPIRAN

	Halaman
Lampiran 1 <i>Sprint</i>	49
Lampiran 2 Hasil Uji Coba <i>Sprint</i> Iterasi 1	49
Lampiran 3 Hasil Uji Coba <i>Sprint</i> Iterasi 2	51
Lampiran 4 Hasil Uji Coba <i>Sprint</i> Iterasi 3	53
Lampiran 5 Daftar Jawaban Responden Pada Kuesioner I.....	55
Lampiran 6 Daftar Jawaban Responden Pada Kuesioner II.....	58
Lampiran 7 <i>Flow of Event</i>	59
Lampiran 8 Hasil Turnitin.....	63
Lampiran 9 Biodata Penulis	64



UNIVERSITAS
Dinamika

BAB I

PENDAHULUAN

1.1 Latar Belakang

Bagian Administasi Akademik & Kemahasiswaan (AAK) Universitas Dinamika adalah salah satu bagian kampus yang bertugas melayani segala kebutuhan administrasi mahasiswa. Bagian ini bertanggung jawab melayani mahasiswa dalam hal akademik maupun non-akademik, termasuk dalam bidang pengajuan yang berkaitan dengan kampus. *Workflow* pada layanan AAK berpusat pada Sistem Informasi *Cyber Campus* (SICYCA) atau mendatangi langsung kantor AAK di lantai 2 Universitas Dinamika.

Pada *workflow* pengajuan yang sekarang, pengajuan dapat diproses dengan *form* pengajuan yang mahasiswa bisa dapatkan di kantor AAK atau mengunduh di *website* Sistem Informasi Cyber Campus (SICYCA). *Form* yang telah didapat kemudian diisi dengan data diri mahasiswa tergantung dari jenis pengajuannya yang terdiri dari dispensasi sakit, yudisium, cuti/Berhenti Studi Sementara (BST) dan Berhenti Studi Tetap (BST). Kemudian mahasiswa mendatangi pihak terkait, yaitu dosen wali dan kaprodi untuk meminta tanda tangan sekaligus memberikan keterangan terkait alasan mahasiswa melakukan pengajuan tersebut. Barulah mahasiswa bisa memberikan *form* pengajuan tersebut ke AAK untuk diproses.

Untuk mengetahui pendapat mahasiswa mengenai *workflow* pengajuan layanan di AAK yang sekarang berjalan, diadakan kuesioner kepada 30 mahasiswa Undika yang terdiri dari angkatan 2016-angkatan 2021. Hasil kuesioner I (dapat dilihat pada lampiran 14) menunjukkan rata-rata indeks kepuasan mahasiswa sebesar 78%. Namun, didapati juga rata-rata indeks harapan mahasiswa agar *workflow* yang sekarang dapat dikembangkan sebesar 84% dari segi pengajuan, mendapat persetujuan dari pihak terkait, serta informasi progres dari pengajuan yang telah diajukan secara langsung.

Berdasarkan hasil kuisioner diatas, diusulkan untuk dibuatkan *workflow* yang lebih efisien dengan bantuan aplikasi. *Workflow* ini diharapkan dapat membantu menjawab harapan mahasiswa dalam efisiensi proses pengajuan, pencarian persetujuan ke pihak terkait (dosen wali dan kaprodi), serta membantu mahasiswa melacak progres dari pengajuan mereka melalui notifikasi *email* dan aplikasi. Aplikasi ini akan menerapkan metode *workflow management system* (WMS) yang mengacu pada proses kerja yang sistematis. WMS memungkinkan alur pekerjaan menjadi lebih jelas, memudahkan proses otorisasi yang dapat dilakukan melalui sistem, persetujuan bisa diketahui secara cepat, dan dapat diakses dimana saja dan kapan saja jika terkoneksi dengan internet (Kezia Rafinska, 2020). Dengan begitu, *workflow* yang sedang berjalan sekarang dapat diefisienkan sesuai dengan harapan mahasiswa.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, dapat diambil rumusan masalah pada penelitian ini yaitu bagaimana cara membuat *workflow* pengajuan layanan di AAK menjadi lebih membantu mahasiswa dalam proses pengajuan, persetujuan, dan informasi progres pengajuan dengan bantuan aplikasi?

1.3 Batasan Masalah

Batasan masalah pada rancang bangun *workflow* layanan administrasi akademik di universitas dinamika adalah sebagai berikut:

1. Penelitian ini membahas dispensasi sakit, yudisium, cuti/BSS, dan BST.
2. Notifikasi hanya melalui *email* dan didalam aplikasi.
3. Pengujian aplikasi hanya menggunakan *black box testing*.

1.4 Tujuan

Berdasarkan latar belakang dan rumusan masalah di atas, maka tujuan penelitian ini adalah membuat *workflow* pengajuan layanan AAK yang lebih membantu mahasiswa dalam proses pengajuan, proses persetujuan, dan pelacakan progres pengajuan yang diajukan dengan bantuan aplikasi.

1.5 Manfaat

Dengan adanya Rancang Bangun *Workflow* Layanan Administrasi Akademik di Universitas Dinamika, diharapkan dapat memiliki manfaat sebagai berikut:

1. Membantu mahasiswa dalam melakukan proses pengajuan.
2. Membantu mahasiswa dalam proses persetujuan ke pihak terkait (dosen wali dan kaprodi).
3. Membantu mahasiswa dalam pelacakan progres pengajuan yang diajukan.



UNIVERSITAS
Dinamika

BAB II LANDASAN TEORI

Landasan teori digunakan untuk dijadikan dasar dalam memberikan solusi yang ditawarkan untuk menyelesaikan permasalahan yang telah dijelaskan sebelumnya, yaitu Rancang Bangun *Workflow* Layanan Administrasi Akademik di Universitas Dinamika.

2.1 Penelitian Terdahulu

Dalam penelitian laporan, perlu digali informasi dari penelitian-penelitian sebelumnya sebagai bahan perbandingan, baik mengenai kekurangan atau kelebihan yang sudah ada. Penelitian terdahulu ini menjadi salah satu acuan dalam memperkaya teori yang digunakan untuk mengkaji penelitian yang dilakukan. Berikut referensi penelitian terdahulu yang terdapat pada Tabel 2.1, Tabel 2.2, dan Tabel 2.3:

Tabel 2.1 Peneliti terdahulu 1

Nama Peneliti	Judul Peneliti	Hasil Peneliti
Yason Novrianto, Sulistiowati, Sulistiowati, Julianto Lemantara (2016)	Rancang Bangun Aplikasi <i>Workflow</i> Seleksi Penerima Beasiswa Pada Stikes Yayasan Dr. Soetomo Surabaya Berbasis Web	Penelitian ini menghasilkan aplikasi yang membantu proses pemilihan mahasiswa penerima beasiswa berbasis <i>workflow</i> .

Perbedaan: Penentuan kriteria penerima mahasiswa beasiswa pada *user* kemahasiswaan dilakukan secara *manual*. Sedangkan milik peneliti, penentuan kriteria mahasiswa dilakukan sebelum mahasiswa dapat mengisi *form* pengajuan *online*.

Tabel 2.2 Penelitian Terdahulu 2

Nama Peneliti	Judul Peneliti	Hasil Peneliti
Afifuddin, Muhajir, A. B. , Tjandrarini, Tegar Heru Susilo (2016)	Rancang Bangun Aplikasi Penanganan Komplain Menggunakan <i>Administrative Workflow</i> Sistem	Penelitian ini menghasilkan aplikasi yang menampung dan Memproses komplain perusahaan berbasis <i>administrative workflow</i> .
Perbedaan: Pada <i>wokrflow</i> penelitian ini, pengguna mengisi <i>form</i> lalu langsung diteruskan kepada orang lain untuk dilakukan otorisasi. Sedangkan pada <i>workflow</i> penelitian ini, akan ada notifikasi <i>email</i> yang terkirim setelah pengguna mengisi <i>form</i> .		

Tabel 2.3 Penelitian Terdahulu 3

Nama Peneliti	Judul Peneliti	Hasil Peneliti
Pebriantono Pebriantono, Vivine Nurcahyawati, Martinus Sony, Erstiawan (2016)	Rancang Bangun Aplikasi <i>Workflow</i> Pelaporan Pemakaian <i>Spare Part</i> Pada PT. Rukina Sukses Abadi	Penelitian ini menghasilkan aplikasi yang membantu perusahaan kapal menghitung penggunaan <i>spare part</i> kapal berbasis <i>workflow</i> .
Perbedaan: Pada <i>workflow</i> penelitian ini, ketika pengguna (bagian kapal) mengirimkan data permintaan <i>spare part</i> ke aplikasi, tidak ada notifikasi yang muncul didalam aplikasi. Sedangkan di penelitian ini, setiap pengguna berhasil mengirim atau melakukan <i>edit data</i> pengajuan dari aplikasi, akan ada notifikasi dari aplikasi yang mengindikasi bahwa proses telah berjalan/terjadi kendala.		

2.2 *Workflow Management System* (WMS)

Berdasarkan (Kezia Rafinska, 2020), *Workflow management system* (WMS) adalah sebuah istilah yang diberikan kepada perangkat lunak untuk memudahkan suatu alur pekerjaan atau yang biasa dikenal dengan istilah *workflow*. *Workflow* sendiri mengacu pada proses kerja yang sistematis dimana pekerjaan yang sudah diselesaikan oleh satu pihak dialihkan ke pihak lain untuk mendapatkan tindakan lanjutan berdasarkan prosedur tertentu.

Manfaat yang bisa didapat dari pengadaan WMS yaitu:

1. Kejelasan alur dan kelancaran proses distribusi pekerjaan, mulai dari pihak pertama sampai pihak yang berwenang untuk mengesahkan berkas sebelum dipublikasikan.
2. Kemudahan otorisasi, karena pekerjaan atau berkas tidak perlu lagi diantar secara manual dalam bentuk hardcopy, tetapi dapat diakses sepenuhnya menggunakan sistem.
3. Penolakan atau persetujuan pekerjaan dapat diketahui secara cepat, sesaat setelah pihak yang berwenang memberikan respon terhadap berkas yang sudah Anda kirimkan.
4. Dapat diakses dimana saja dan kapan saja, jika sistem ini sudah terkoneksi dengan internet.

2.3 Framework Pengembangan Aplikasi

(Robith Adani, 2020) berpendapat bahwa *framework* adalah kerangka kerja untuk mengembangkan aplikasi berbasis *website* maupun *desktop*. Kerangka kerja diciptakan untuk mempermudah kinerja dari *programmer*. Sehingga, seorang *programmer* tidak perlu untuk menuliskan kode secara berulang-ulang. Karena di dalamnya sendiri anda hanya perlu menyusun komponen-komponen pemrograman saja.

2.3.1 PHP

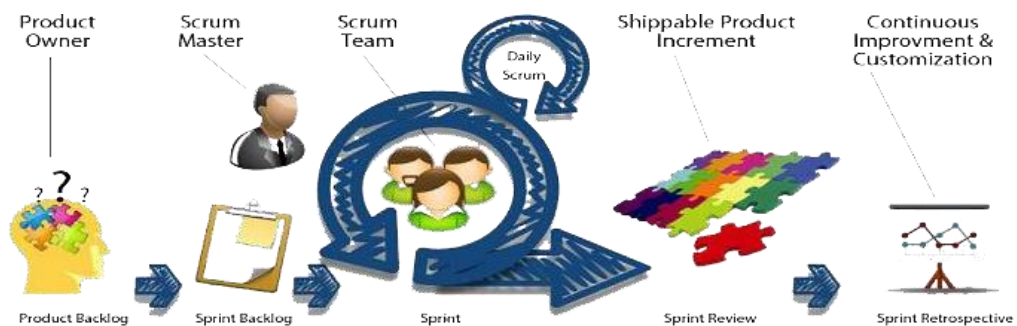
Menurut Kustiyaningsih (2011:114), PHP (atau resminya PHP: *Hypertext Preprocessor*) adalah skrip bersifat *server-side* yang ditambahkan ke dalam HTML. Pada prinsipnya *server* akan bekerja apabila ada permintaan dari *client*. Dalam hal ini *client* menggunakan kode-kode PHP untuk mengirimkan permintaan ke *server*.

2.3.2 Laravel

Menurut (Aminudin, 2015), Laravel dirilis dibawah lisensi MIT dengan kode sumber yang sudah disediakan oleh Github, sama seperti *framework-framework* yang lain, Laravel dibangun dengan konsep MVC (*Model-Controller-View*), kemudian Laravel dilengkapi juga *command line tool* yang bernama “Artisan” yang bisa digunakan untuk *packaging* dan instalasi *bundle* melalui *command prompt*. Maka tidak heran jika *framework* laravel berbasis PHP ini digadang-gadang menjadi primadona *framework* bagi *programmer* PHP untuk membuat aplikasi-aplikasi yang lebih elegan dan dinamis. Karena *framework* ini menekankan kesederhanaan dan leksibilitas pada desain-nya, maka dari itu pengguna *framework* ini meningkat dari tahun ke tahun.

2.4 Model SCRUM

Berdasarkan Y.L. Prasetio, et al. (2014:534-543), *SCRUM* merupakan model pengembangan perangkat lunak yang banyak digunakan baik pada skala industri maupun pendidikan. Kajian berkaitan dengan *SCRUM* diantaranya adalah dalam proses pengembangan sistem manajemen untuk melakukan analisis performance indicator, penelitian tersebut bertujuan untuk mengukur target dan pencapaian karyawan berdasarkan tujuan perusahaan melalui key performance indicator. Pada penelitian tersebut *SCRUM* yang digunakan terdiri dari *backlog* dan *sprint*. Dimana dalam proses *backlog* aktivitas dilakukan yaitu studi literatur baik berupa hasil penelitian maupun produk sebelumnya, penyebaran kuesioner, dan wawancara sedangkan *sprint* aktivitas yang dilakukan terdiri dari *sprint*, *release planning* dan *actual sprint*. Contoh Model *SCRUM* ada pada gambar 2.1:



Gambar 2.1 Model *SCRUM* (Steen Lerche-Jensen, 2016)

Alqudah, M. & Razali, R. (2016:828-837) berpendapat dalam proses pengembangan *SCRUM* menganut tiga aturan utama yaitu *product owner* (PO), *SCRUM master* (SM), dan *cross functional* yang didukung oleh lima aktivitas dalam proses pengembangan yaitu *backlog planning*, *sprint planning*, *daily meeting*, *reviews*, dan *sprint retrospective*.

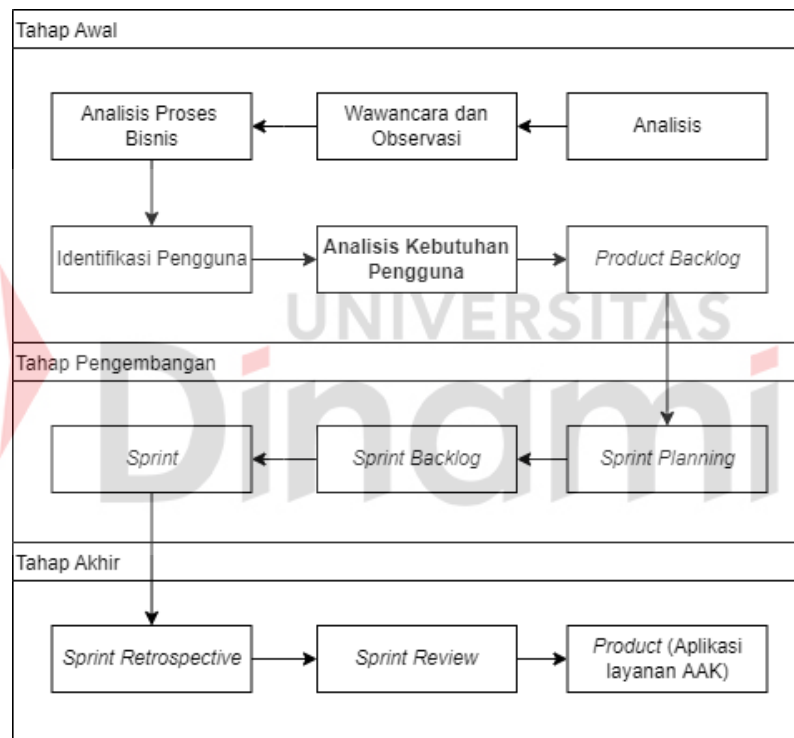
2.5 Metode Agile

Menurut (Adani, 2020), *Agile* adalah metodologi pengembangan perangkat lunak yang didasarkan pada proses pengerjaan yang dilakukan berulang dimana, aturan dan solusi yang disepakati dilakukan dengan kolaborasi antar tiap tim secara terorganisir dan terstruktur. *Agile* sendiri merupakan model pengembangan perangkat lunak dalam jangka pendek. Kemudian, membutuhkan adaptasi yang cepat dalam mengatasi setiap perubahan. Nilai terpenting dari *Agile* ini adalah memungkinkan sebuah tim dalam mengambil keputusan dengan cepat, kualitas dan prediksi yang baik, serta memiliki potensi yang baik dalam menangani setiap perubahan. Proses pengembangan *Agile* mengacu pada konsep dari *Agile Manifesto*. Konsep tersebut dikembangkan oleh 14 tokoh terkenal dalam industri perangkat lunak.

Iswari (2012:34) mengemukakan bahwa *agile* menggunakan asumsi bahwa *goals* yang ingin dicapai belum pernah diberkastasikan secara eksplisit dari *stakeholders*, sehingga analis harus bekerja dari diagram atau berkas-berkas mengenai perusahaan yang bersangkutan. *Agile* berfokus pada identifikasi awal dan abstraksi *goal* dari para *stakeholder*. Proses yang dilakukan dalam metode *agile* ini berkisar antara 1 sampai 4 minggu, yang disebut *Sprint*. Prinsip dari teknik *agile* adalah mengadopsi berkastasi seminimal mungkin untuk mengekspresikan kebutuhan (*requirement*) dalam bahasa yang natural dalam setiap sesi *sprint*. Selanjutnya, tantangan dalam elisitasi dari kebutuhan non-fungsional dilakukan dengan mendorong partisipasi maksimum dari *stakeholders* dalam sesi *sprint*.

BAB III METODOLOGI PENELITIAN

Dalam penelitian ini, peneliti menggunakan metode *agile* untuk mengidentifikasi workflow mulai dari tahap awal hingga tahap akhir. Penelitian ini menggunakan model SCRUM, untuk menelusuri alur masuk *form* administrasi dari pengisian *form* hingga persetujuan *form*, sehingga bisa ditarik garis workflow yang nantinya akan diimplentasikan kedalam aplikasi. Gambaran metode penelitian bisa dilihat pada gambar 3.1.



Gambar 3.1 Metode Penelitian

3.1 Tahap Awal

Tahap ini diperlukan untuk melihat aplikasi yang sudah ada. Peneliti bermaksud untuk mengumpulkan informasi tentang layanan administrasi AAK yang sudah berjalan sebagai dasar pembuatan aplikasi ini.

3.1.1 Analisis

Tahap analisis adalah tahap awal dimana pengembang sistem diperlukan suatu komunikasi yang bertujuan untuk memahami *software* yang dibutuhkan pengguna dan batasan *software*. Informasi ini biasanya dapat diperoleh melalui wawancara, *survey* (observasi) atau diskusi.

3.1.2 Wawancara & Observasi

Peneliti akan mengadakan wawancara dengan kepala Bagian AAK, Sekar Dewanti, untuk menggali informasi tentang sistem AAK yang selama ini berjalan dan kemana saja alur data yang dilalui oleh sebuah form/surat yang diajukan oleh mahasiswa.

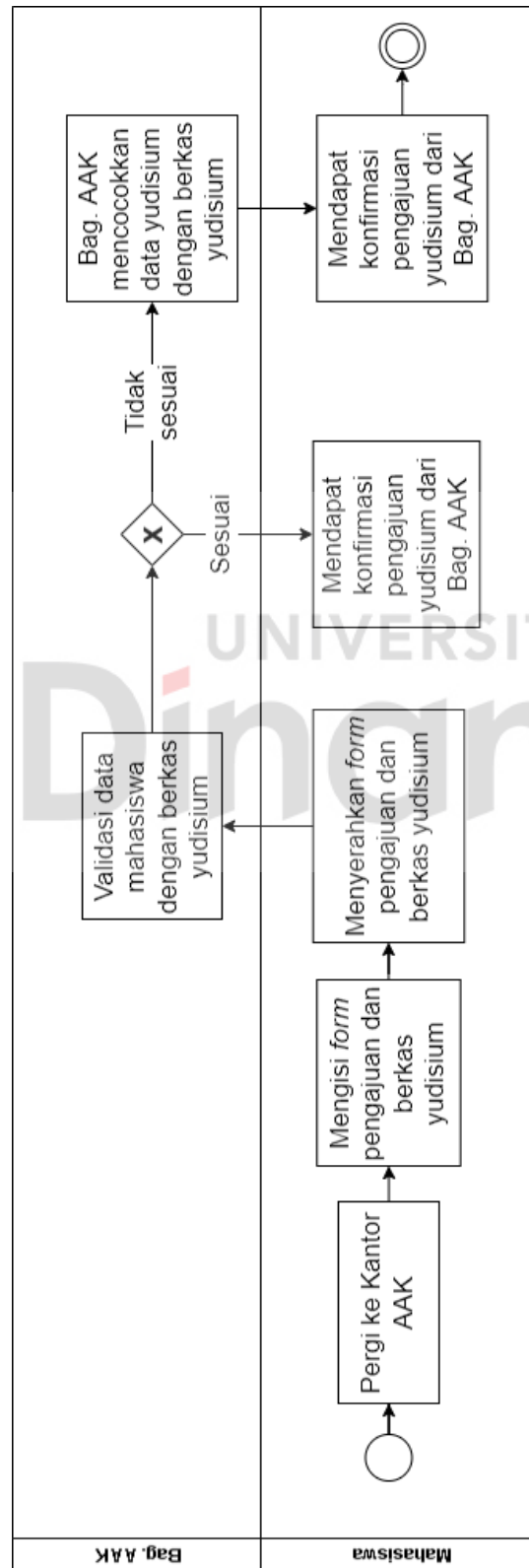
Peneliti ingin mengetahui apa pendapat mereka tentang proses bisnis yang terjadi di AAK. Bagian ini juga menunjukkan bagaimana peneliti mengambil data tersebut menggunakan metode kuisioner, data yang terlampir diambil dari 30 responden dari mahasiswa Universitas Dinamika Surabaya pada tanggal 6 Juli 2021 melalui Kuesioner I dan II. Hasil analisa kuesioner dapat dilihat pada lampiran 14 dan 15.

3.1.3 Analisis Proses Bisnis

Dari hasil analisis proses bisnis, didapati 4 proses bisnis utama, yaitu: pengajuan yudisium, pengajuan dispensasi sakit, pengajuan cuti, dan pengajuan BST pada layanan administrasi akademik.

Workflow pengajuan yudisium dimulai dari mahasiswa pergi ke kantor Administrasi Akademik dan Kemahasiswaan (AAK), lalu mengisi *form* pengajuan yudisium dengan membawa berkas-berkasnya. Selanjutnya, mahasiswa menyerahkan *form* pengajuan ke bagian AAK untuk dilakukan validasi data mahasiswa sekaligus pengumpulan berkas-berkasnya, jika data mahasiswa sesuai

dengan berkas, maka bagian AAK akan menginformasikan kepada mahasiswa bahwa mereka dapat melakukan yudisium dan menunggu kabar dari bagian AAK. Untuk gambar dari *workflow* pengajuan yudisium dapat dilihat pada gambar 3.2.



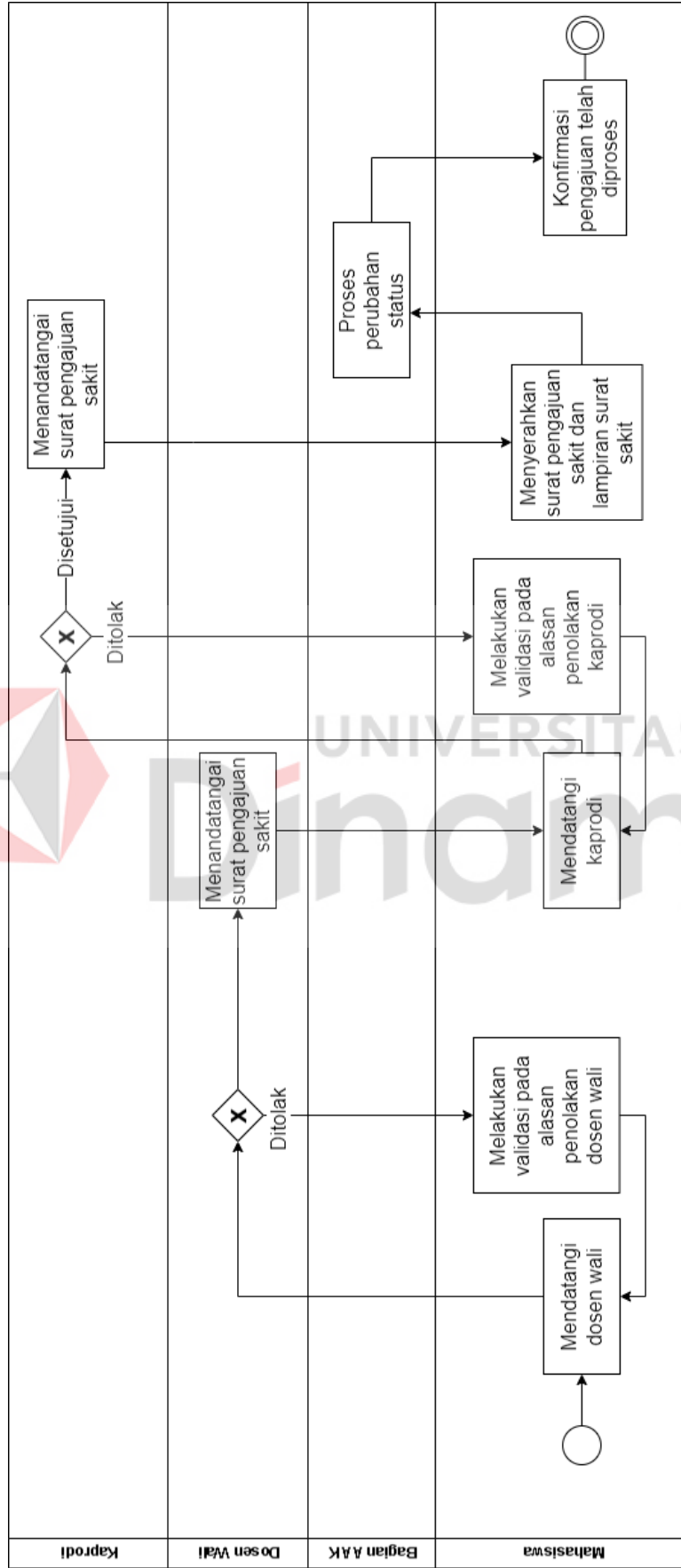
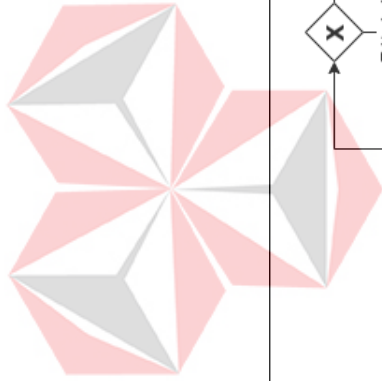
Gambar 3.2. *Workflow* Pengajuan Yudisium



Workflow pada pengajuan dispensasi sakit dimulai dari mahasiswa mendatangi dosen wali untuk meminta tanda tangan pada surat pengajuan sakit beserta lampiran surat sakit yang dibawa. Jika dosen wali menyetujui, surat pengajuan akan ditanda-tangani oleh dosen wali, namun jika dosen wali menolak maka mahasiswa dapat mengajukan ulang jika alasan penolakannya masih dapat diubah seperti surat sakit yang tanggalnya tidak sesuai. Setelah mendapat tanda tangan dari dosen wali, mahasiswa menemui kaprodi untuk meminta tanda tangan pada surat pengajuan sakit. Jika kaprodi menolak, maka mahasiswa dapat mengajukan ulang jika alasan penolakannya masih dapat diubah seperti surat sakit yang tanggalnya tidak sesuai. Jika kaprodi menyetujui, surat pengajuan akan ditanda-tangani dan mahasiswa bisa mengajukan dispensasi sakit ke kantor Administrasi Akademik dan Kemahasiswaan (AAK) untuk dilakukan penggantian status. Untuk gambar dari *workflow* dapat dilihat pada gambar 3.3.



UNIVERSITAS
Dinamika



Gambar 3.3 Workflow Pengajuan Dispensasi sakit

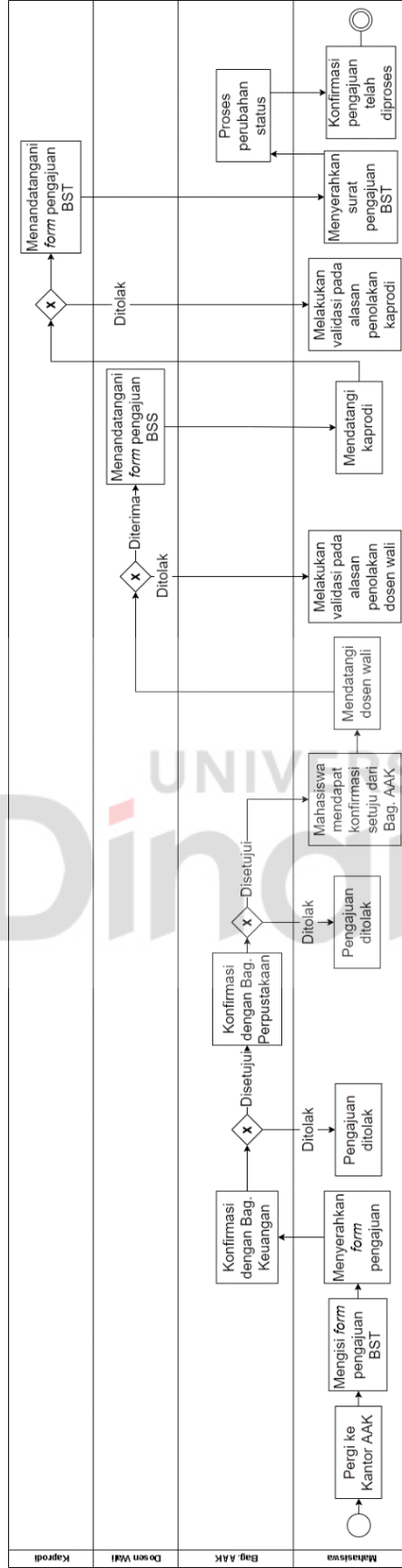
Workflow pada pengajuan cuti/Berhenti Studi Sementara (BSS) dimulai dari mahasiswa pergi ke kantor Administrasi Akademik dan Kemahasiswaan (AAK). Kemudian, mengisi formulir pengajuan cuti/BSS dan menyerahkannya ke kantor AAK. Bagian AAK melakukan validasi dengan menghubungi bagian keuangan untuk dilakukan pengecekan kondisi tanggungan Biaya Operasional Pendidikan (BOP) mahasiswa, jika didapati terdapat tanggungan, maka pengajuan ditolak oleh bagian AAK, jika tidak didapati tanggungan, bagian AAK akan melakukan pengecekan tanggungan buku pada bagian perpustakaan. Jika didapati mahasiswa terdapat tanggungan peminjaman buku, maka pengajuan ditolak oleh bagian AAK, namun jika tidak didapati tanggungan peminjaman buku, proses validasi dilanjutkan dengan melihat minggu perkuliahan, jika didapati minggu perkuliahan belum genap 4 minggu berjalan, maka pengajuan ditolak oleh bagian AAK, namun jika perkuliahan sudah berjalan lebih dari atau sama dengan 4 minggu, maka validasi terakhir adalah jumlah cuti. Jika didapati pengajuan cuti lebih dari 2 kali, maka pengajuan ditolak oleh bagian AAK, namun jika jumlah pengajuan cuti masih dibawah 2, maka mahasiswa akan mendapat konfirmasi setuju dari bagian AAK.

Mahasiswa melanjutkan ke dosen wali untuk meminta tanda tangan pada formulir pengajuan cuti/BSS, pada sesi ini, mahasiswa memberikan alasan mengenai pengajuan cutinya kepada dosen wali. Jika dosen wali menyetujui, maka formulir pengajuannya akan ditanda-tangani dan mahasiswa bisa melanjutkan ke kaprodi, namun jika dosen wali menolak, maka mahasiswa harus tetap melanjutkan kuliah. Setelah mendapat tanda tangan dari dosen wali, mahasiswa menyerahkan formulir pengajuan cuti/BSS ke bagian AAK untuk dilakukan proses perubahan status. Setelah selesai diproses, mahasiswa akan diberikan notifikasi *email* saat pengajuan cuti/BSS telah selesai diproses oleh bagian AAK. Untuk gambar *workflow* pengajuan BSS bisa dilihat pada gambar 3.4.

Workflow pada Berhenti Studi Tetap (BST) dimulai dari mahasiswa pergi ke kantor Administrasi Akademik dan Kemahasiswaan (AAK). Kemudian, mengisi formulir pengajuan BST dan menyerahkannya ke kantor AAK. Bagian AAK melakukan validasi dengan menghubungi bagian keuangan untuk dilakukan pengecekan kondisi tanggungan Biaya Oerasional Pendidikan (BOP) mahasiswa, jika didapati terdapat tanggungan, maka pengajuan ditolak oleh bagian AAK, jika tidak didapati tanggungan, bagian AAK akan melakukan pengecekan tanggungan buku pada bagian perpustakaan. Jika didapati mahasiswa terdapat tanggungan peminjaman buku, maka pengajuan ditolak oleh bagian AAK, namun jika tidak didapati tanggungan peminjaman buku, maka mahasiswa akan mendapat konfirmasi setuju dari bagian AAK.

Mahasiswa melanjutkan ke dosen wali untuk meminta tanda tangan pada formulir pengajuan cuti/BSS, pada sesi ini, mahasiswa memberikan alasan mengenai pengajuan cutinya kepada dosen wali. Jika dosen wali menyetujui, maka formulir pengajuannya akan ditanda-tangani dan mahasiswa bisa melanjutkan ke kaprodi, namun jika dosen wali menolak, maka mahasiswa harus tetap melanjutkan kuliah. Setelah mendapat tanda tangan dari dosen wali, mahasiswa mendatangi kaprodi untuk meminta tanda tangan pada formulir pengajuan. Proses yang sama terjadi dimana mahasiswa menceritakan alasan pengajuan BST kepada kaprodi, jika kaprodi menerima, formulir pengajuan akan ditanda-tangani dan mahasiswa bisa melanjutkan penyerahan formulir ke kantor AAK. Jika kaprodi menolak, mahasiswa harus kembali melanjutkan studi.

Setelah dari kaprodi, mahasiswa menyerahkan formulir pengajuan BST ke bagian AAK untuk dilakukan proses perubahan status. Setelah selesai diproses, mahasiswa akan diberikan notifikasi *email* saat pengajuan BST telah selesai diproses oleh bagian AAK. Untuk gambar *workflow* pengajuan BSS bisa dilihat pada gambar 3.5.



Gambar 3.5 Workflow Pengajuan BST

3.1.4 Identifikasi Masalah

Identifikasi dilakukan untuk mengetahui masalah pada proses bisnis yang ada, dari masalah tersebut, peneliti dapat mengambil solusi yang dapat diimplementasikan pada aplikasi yang tertuang pada tabel 3.1.

Tabel 3.1 Identifikasi Masalah

No.	Masalah	Dampak	Solusi
1	Proses pengajuan dilakukan mengambil dan mengembalikan <i>form</i> pengajuan ke kantor AAK.	Mahasiswa harus mendatangi kantor AAK untuk meminta dan mengembalikan <i>form</i> pengajuan.	Rancang bangun aplikasi layanan adminis-trasi dengan fitur <i>form online</i> yang dapat diisi mahasiswa untuk melakukan pengajuan <i>via online</i> .
2	Mahasiswa harus mencari pihak terkait untuk melalui proses persetujuan surat yang mereka ajukan jika mendatangi kampus.	Proses persetujuan pengajuan yang memakan waktu.	Rancang bangun aplikasi layanan administrasi akademik dengan <i>work-flow management system</i> (WMS) yang dapat me- <i>ngirim</i> berkas dan <i>form</i> pengajuan <i>via online</i> kepada pi-hak terkait.
3	Informasi progres pengajuan tidak di-informasikan kepada mahasiswa	Mahasiswa harus mendatangi Bagian AAK atau menghubungi <i>via</i> nomor telfon untuk menanyakan perihal pengajuan atau melalui <i>email</i> dengan waktu jawab yang tidak pasti.	Rancang bangun aplikasi layanan administrasi aka-demik yang me-nunjukkan status dari pengajuan ma-hasiswa melalui aplikasi dan <i>email</i> .

3.1.5 Analisis Kebutuhan Pengguna

Analisis ini dilakukan untuk mengetahui kebutuhan data apa saja yang akan digunakan oleh *user* sehingga bisa diketahui kebutuhan dari *user* serta memperkirakan hasil akhir yang didapat oleh *user*. Untuk lebih lanjut, dapat dilihat pada tabel 3.2.

Tabel 3.2 Analisis Kebutuhan Pengguna

Pengguna	Kebutuhan Fungsi	Kebutuhan Data	Kebutuhan Informasi
Bagian AAK	Proses persetujuan	1. Data pengajuan mahasiswa 2. Berkas <i>file</i> pengajuan pada pengajuan tertentu	Informasi <i>detail</i> pengajuan
	<i>Export</i> laporan pengajuan	Data pengajuan mahasiswa	Laporan pengajuan mahasiswa
	<i>History</i> pengajuan mahasiswa	Data pengajuan mahasiswa	Laporan pengajuan mahasiswa
	Proses persetujuan	1. Data pengajuan mahasiswa 2. Berkas <i>file</i> pada pengajuan tertentu	Informasi <i>detail</i> pengajuan
Kaprodi	Notifikasi <i>email</i>	Data pengajuan mahasiswa	Konfirmasi pengajuan mahasiswa
	<i>History</i> pengajuan mahasiswa	Data pengajuan mahasiswa	Laporan pengajuan mahasiswa
Dosen Wali	Proses persetujuan	Data pengajuan mahasiswa	Informasi <i>detail</i> pengajuan
	Notifikasi <i>email</i>	Data pengajuan mahasiswa	Konfirmasi pengajuan mahasiswa
	<i>History</i> pengajuan mahasiswa	Data pengajuan mahasiswa	Laporan pengajuan mahasiswa



Pengguna	Kebutuhan Fungsi	Kebutuhan Data	Kebutuhan Informasi
Mahasiswa	Proses pengajuan/edit data	1. Data pengajuan 2. Berkas <i>file</i> pengajuan	1. Informasi absensi mhs 2. Informasi bop 3. Informasi tanggungan perpus 4. Informasi jumlah cuti
	Notifikasi <i>email</i>	Pengajuan yang telah disetujui/ditolak	Konfirmasi pengajuan yang telah disetujui/ditolak
	<i>History</i> pengajuan mahasiswa	Data pengajuan mahasiswa	Laporan pengajuan mahasiswa

3.1.6 Product Backlog

Product Backlog adalah daftar terurut semua hal yang telah diketahui hingga saat ini harus ada di dalam produk atau aplikasi. *Product Backlog* adalah satu-satunya sumber kebutuhan untuk semua perubahan yang perlu dilakukan terdapat produk. Dibawah ini adalah *product backlog* untuk aplikasi Bagian AAK dengan penentuan prioritas pekerjaan yang paling mudah dikerjakan yang tertuang pada tabel 3.3.

Tabel 3.3 *Product Backlog*

<i>Sprint</i>	ID	<i>Story</i>	Prioritas
<i>Sprint</i> ke 1	PB1	Mahasiswa, Dosen Wali, dan <i>Bagian AAK</i> dapat melakukan <i>login</i> .	1
	PB2	Semua pengguna dapat memonitor proses berjalannya berkas	2
<i>Sprint</i> ke 2	PB3	Mahasiswa dapat melakukan pengajuan dan <i>upload</i> berkas	3
	PB4	<i>Sistem</i> dapat menyortir berkas yang masuk sesuai dengan ketentuan yang ditetapkan	4

<i>Sprint</i>	ID	<i>Story</i>	Prioritas
<i>Sprint</i> ke 3	PB5	<i>Sistem</i> dapat memilah secara otomatis mahasiswa yang memenuhi/tidak memenuhi kriteria untuk mengakses layanan pengajuan	5
	PB6	<i>Sistem</i> dapat mengirim notifikasi berupa <i>email</i> sesaat setelah pengguna mengunggah, meng- <i>edit</i> , dan menyetujui atau menolak pengajuan	6
	PB7	<i>Sistem</i> dapat mengekspor data <i>history</i> pengajuan mahasiswa <i>via</i> Microsoft Excel	7

3.2 Tahap Pengembangan

Pada tahap ini, peneliti mulai merancang *sprint* untuk membantu perancangan *workflow* dengan mengambil data pada tahap pengembangan dan menyatukannya menjadi satu, sehingga didapat disimpulkan proses bisnis mana yang menjadi prioritas pengerjaan.

3.2.1 *Sprint Planning*

Pekerjaan yang akan dikerjakan di-*sprint* direncanakan pada saat *sprint planning*. Perencanaan ini dilakukan secara bersama oleh seluruh anggota *SCRUM team*. Pada pengembangan aplikasi, dilakukan 3 iterasi untuk dapat menyelesaikan aplikasi.

3.2.2 *Sprint Backlog*

Sprint backlog terdapat fungsi dari setiap *story* yang telah didapatkan dari *productbacklog* dan *sprint planning*. Pada *sprint backlog* menghasilkan fungsi dan goalnya masing-masing.

A. *Sprint Backlog* Iterasi Pertama

Sprint backlog pada iterasi pertama bertujuan pada pembuatan aplikasi sesuai *product backlog* dengan prioritas 1-2. Iterasi pertama ini difokuskan untuk beberapa fitur pengguna diantaranya adalah *login* bagi semua pengguna dan melakukan proses *track* proses pengajuan. Berikut detail penjelasan pada

tabel 15 mengenai *sprint backlog* iterasi pertama:

Tabel 3.4 *Sprint Backlog* Iterasi Pertama

ID	Task	Deskripsi Pekerjaan
PB1	Membuat halaman <i>login</i> .	Semua pengguna dapat melakukan proses <i>login</i> untuk mengakses halaman <i>home</i> sesuai dengan <i>role</i> masing-masing
PB2	Pembuatan tabel <i>track</i> progres pengajuan dan tabel <i>history</i> pengajuan	Agar mahasiswa, dosen wali, kaprodi, dan bagian aak dapat melakukan <i>tracking</i> pada progres pengajuan.

B. *Sprint Backlog* Iterasi Kedua

Sprint backlog pada iterasi kedua bertujuan merujuk pada *product backlog* dengan prioritas 3-4. Iterasi kedua ini difokuskan untuk beberapa fitur pendukung yaitu proses *upload* berkas pengajuan dan proses sortir berkas yang masuk ke sistem sesuai dengan aturan yang ditetapkan. Berikut detail penjelasan pada Tabel 3.5 mengenai *sprint backlog* iterasi kedua:

Tabel 3.5 *Sprint Backlog* Iterasi Kedua

ID	Task	Deskripsi Pekerjaan
PB3	Menambahkan <i>form</i> pengajuan dan fitur <i>upload</i> pada halaman yudisium dan dispensasi sakit	Mahasiswa dapat menyertakan berkas <i>file</i> pada saat <i>submit</i>
PB4	Menambahkan fitur <i>sorting</i> berkas pada fitur <i>upload</i>	Berkas yang masuk hanya yang sesuai dengan ketentuan AAK

C. *Sprint Backlog* Iterasi Ketiga

Sprint backlog pada iterasi ketiga bertujuan merujuk pada *product backlog* dengan prioritas 5-7. Iterasi ketiga ini difokuskan untuk beberapa fitur lainnya yaitu sortir mahasiswa yang dapat mengakses layanan tertentu (cuti dan BST), notifikasi *via email* saat pengguna melakukan aksi tertentu, serta *export history* pengajuan

mahasiswa bagi bagian AAK dalam bentuk Microsoft Excel. Berikut detail penjelasan pada Tabel 3.6 mengenai *sprint backlog iterasi* ketiga:

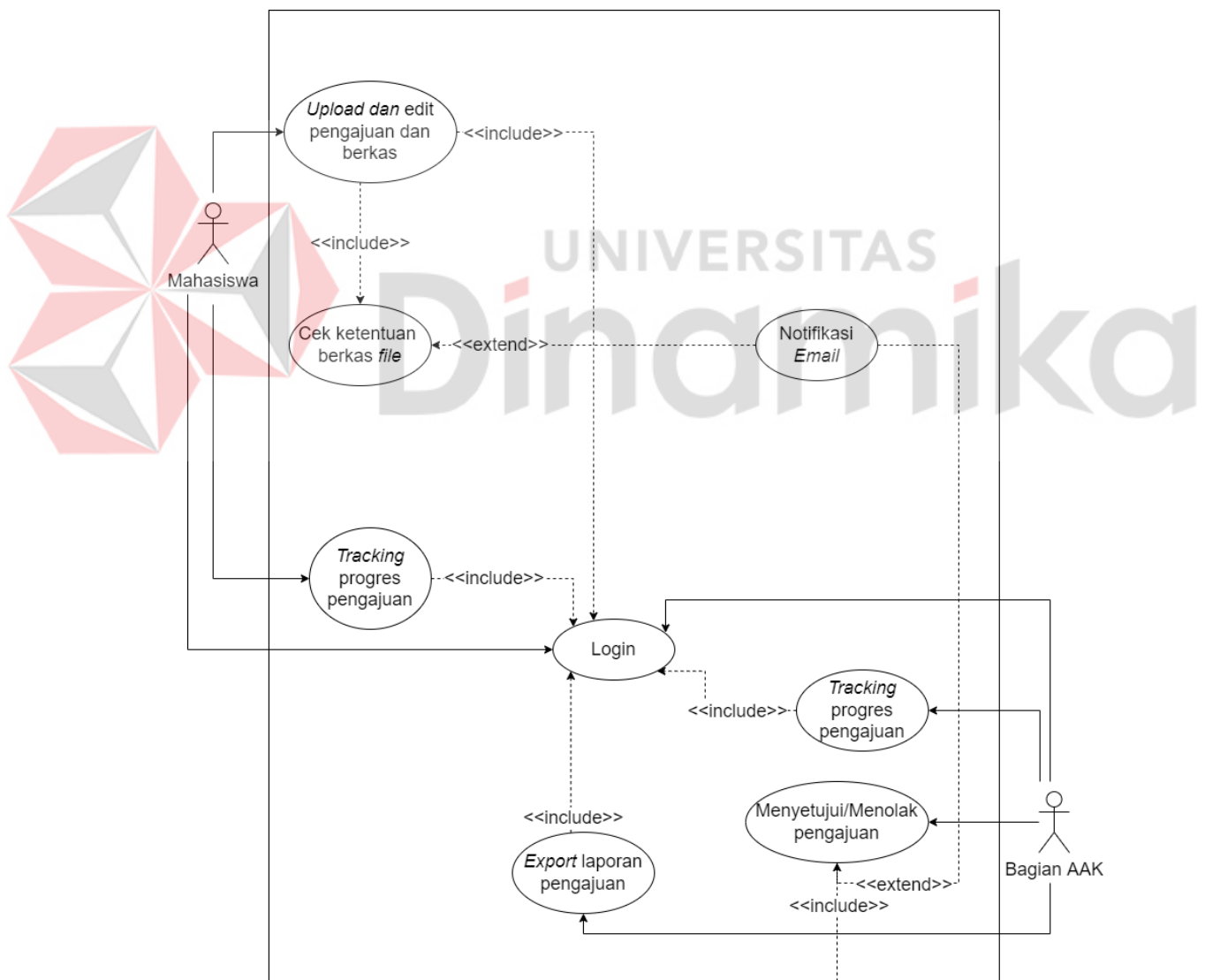
Tabel 3.6 *Sprint Backlog* Iterasi Ketiga

ID	Task	Deskripsi Pekerjaan
PB5	Menambahkan fitur <i>sorting</i> mahasiswa pada halaman cuti dan BST	Mahasiswa yang tidak sesuai dengan ketentuan AAK tidak dapat mengajukan pengajuan cuti dan BST
PB6	Menambahkan fitur unggah dan <i>edit</i> pengajuan ,fitur setuju/tolak pengajuan, dan menambahkan notifikasi <i>email</i>	Fitur unggah dan <i>edit</i> dibuat untuk mahasiswa dan diintegrasikan dengan notifikasi <i>email</i> . Fitur persetujuan dibuat untuk dosen wali,kaprodi, dan bag. AAK dan diintegrasikan dengan notifikasi <i>email</i> .
PB7	Menambahkan fitur <i>export data</i> tabel ke Microsoft Excel	Bagian AAK dapat melakukan rekap data dengan format Microsoft Excel.

3.2.3 Sprint

Tahap ini adalah pengerjaan pada *product backlog* yang telah dibuat sebelumnya. Pengerjaan dibagi menjadi 3 iterasi, yaitu: sprint iterasi pertama yang membahas *login* dan *monitoring* berkas, sprint iterasi kedua yang membahas *upload* dan *sorting* berkas, dan sprint interasi ketiga yang membahas pemilah mahasiswa, notifikasi *email*, dan *export history* pengajuan dengan menampilkan *use case diagram*, *flow of event*, dan *sequence diagram*.

- pengajuan, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
6. Dosen wali/kaprodi dapat melakukan *tracking* pada progres pengajuan, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
 7. Bagian AAK dapat melakukan proses *login*.
 8. Bagian AAK dapat melakukan proses persetujuan/penolakan pada pengajuan, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
 9. Bagian AAK dapat melakukan *tracking* pada progres pengajuan, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
 10. Notifikasi *email* adalah bagian dari proses cek ketentuan berkas di mahasiswa dan menyetujui/menolak pengajuan pada dosen wali/kaprodi dan bagian AAK.



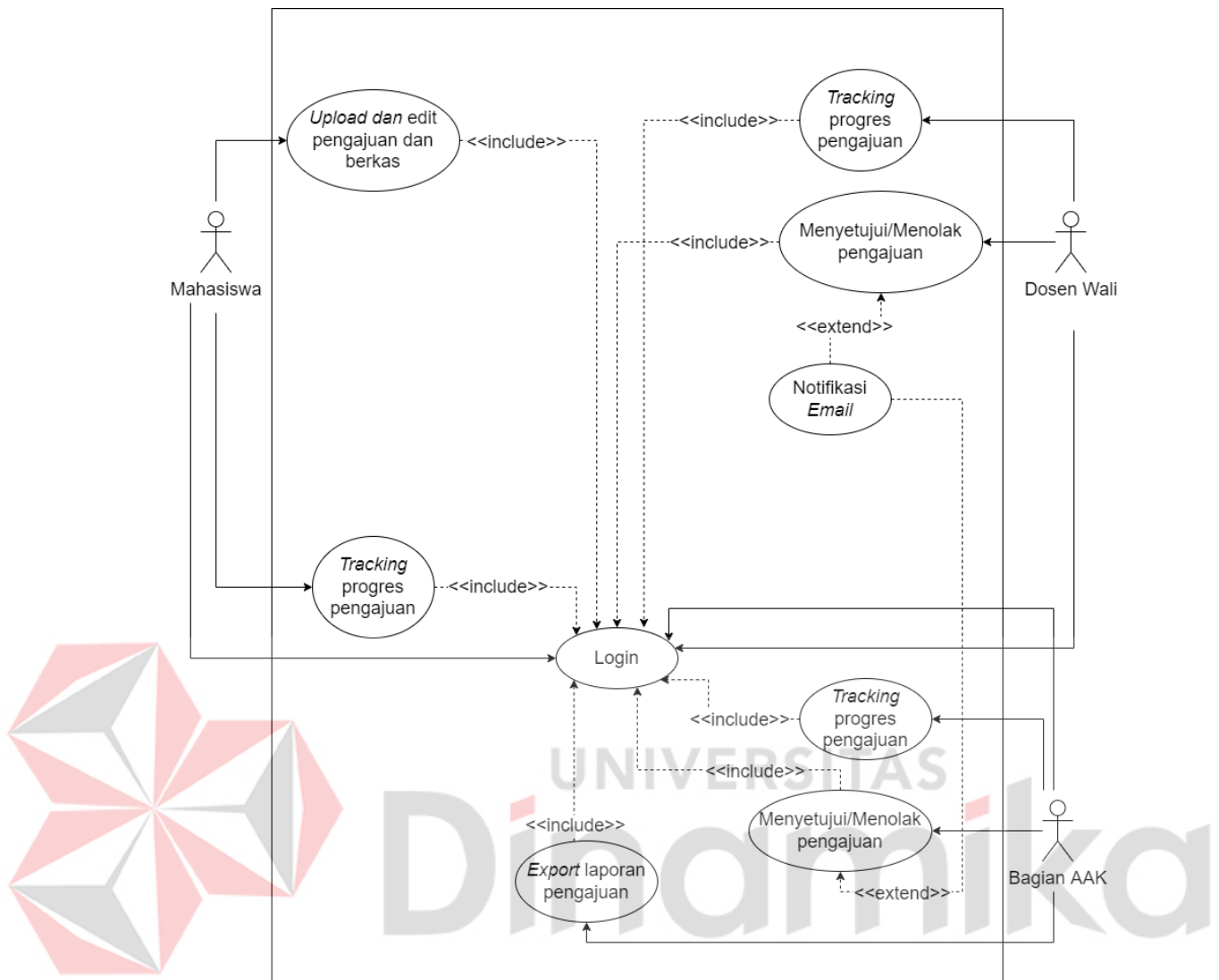
Gambar 3.7 Use case pengajuan yudisium

Keterangan:

1. Mahasiswa dapat melakukan *upload* dan *edit* data beserta berkasnya, proses tersebut diikuti dengan pengecekan ketentuan berkas *file*.
2. Untuk dapat mengakses layanan dispensasi sakit, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
3. Mahasiswa dapat melakukan *tracking* pada progres pengajuan dispensasi sakit, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
4. Bagian AAK dapat melakukan proses *login*.
5. Bagian AAK dapat melakukan proses persetujuan/penolakan pada pengajuan, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
6. Bagian AAK dapat melakukan *tracking* pada progres pengajuan, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
7. Notifikasi *email* adalah bagian dari proses cek ketentuan berkas di mahasiswa dan menyetujui/menolak pengajuan pada dosen wali/kaprodi dan bagian AAK.



UNIVERSITAS
Dinamika

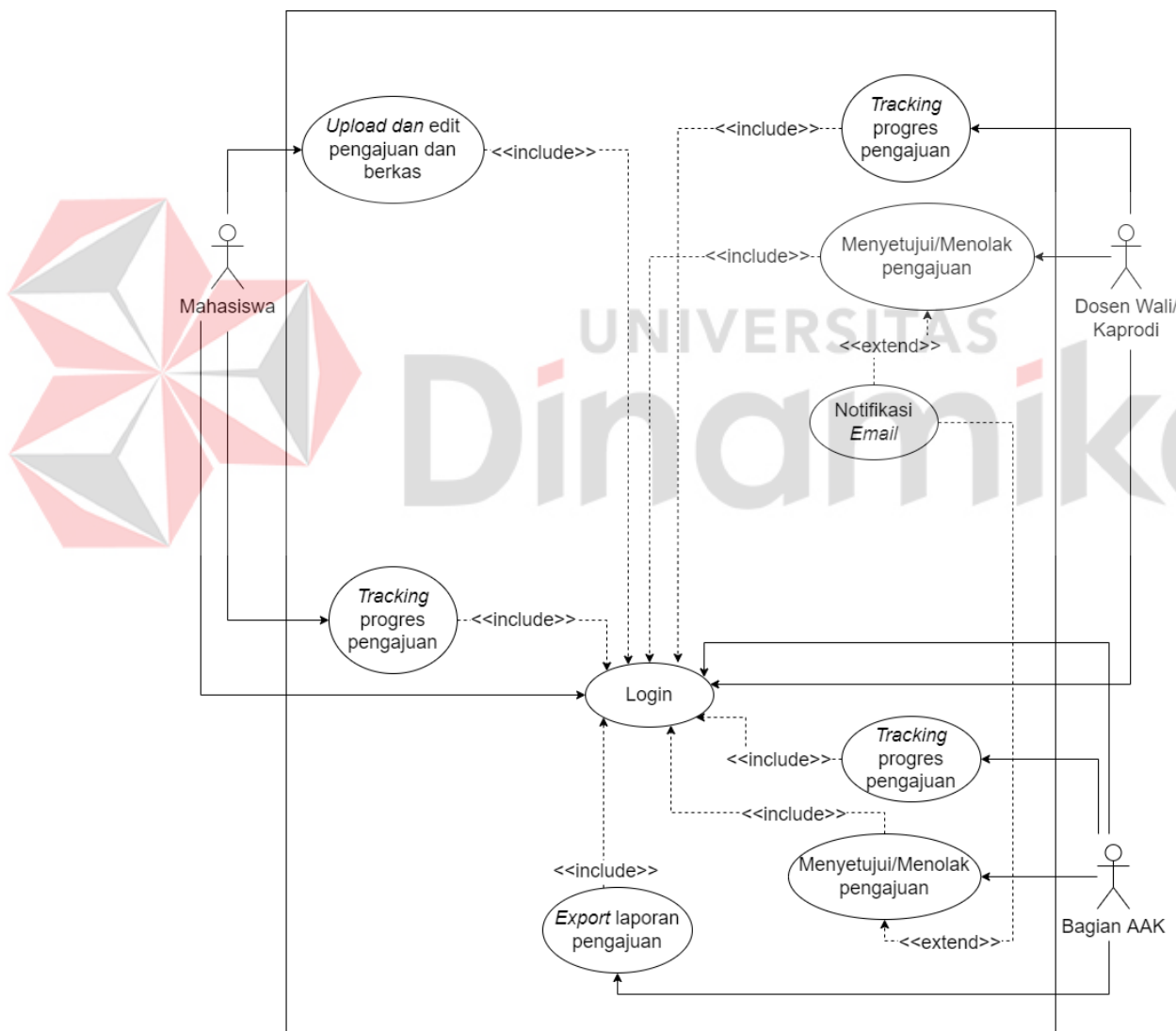


Gambar 3.8 Use case pengajuan cuti/Berhenti Studi Sementara (BSS)

Keterangan:

1. Mahasiswa dapat melakukan *upload* dan *edit* data beserta berkasnya, proses tersebut diikuti dengan pengecekan ketentuan berkas *file*.
2. Untuk dapat mengakses layanan dispensasi sakit, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
3. Mahasiswa dapat melakukan *tracking* pada progres pengajuan dispensasi sakit, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
4. Dosen wali dapat melakukan proses *login*.
5. Dosen wali dapat melakukan proses persetujuan/penolakan pada pengajuan, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
6. Dosen wali dapat melakukan *tracking* pada progres pengajuan, untuk dapat

- mengakses fitur tersebut harus melakukan proses *login*.
7. Bagian AAK dapat melakukan proses *login*.
 8. Bagian AAK dapat melakukan proses persetujuan/penolakan pada pengajuan, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
 9. Bagian AAK dapat melakukan *tracking* pada progres pengajuan, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
 10. Notifikasi *email* adalah bagian dari proses cek ketentuan berkas di mahasiswa dan menyetujui/menolak pengajuan pada dosen wali/kaprodi dan bagian AAK.



Gambar 3.9 Use case pengajuan Berhenti Studi Tetap (BST)

Keterangan:

1. Mahasiswa dapat melakukan *upload* dan *edit* data beserta berkasnya, proses tersebut diikuti dengan pengecekan ketentuan berkas *file*.
2. Untuk dapat mengakses layanan dispensasi sakit, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
3. Mahasiswa dapat melakukan *tracking* pada progres pengajuan dispensasi sakit, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
4. Dosen wali/kaprodi dapat melakukan proses *login*.
5. Dosen wali/kaprodi dapat melakukan proses persetujuan/penolakan pada pengajuan, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
6. Dosen wali/kaprodi dapat melakukan *tracking* pada progres pengajuan, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
7. Bagian AAK dapat melakukan proses *login*.
8. Bagian AAK dapat melakukan proses persetujuan/penolakan pada pengajuan, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
9. Bagian AAK dapat melakukan *tracking* pada progres pengajuan, untuk dapat mengakses fitur tersebut harus melakukan proses *login*.
10. Notifikasi *email* adalah bagian dari proses cek ketentuan berkas di mahasiswa dan menyetujui/menolak pengajuan pada dosen wali/kaprodi dan bagian AAK.

B. Analisis Use Case Diagram

Analisis pada diagram *use case* dilakukan berdasarkan pada analisis kebutuhan pengguna dan *sprint planning* dengan menggunakan *flow of event*. Untuk hasil analisis, terdapat di lampiran 13 dan tabel 3.7-3.9.

- Fitur *Monitoring* Pengajuan

Tabel 3.7 *Flow of event* fitur *monitoring* pengajuan

ID: UC 02	Prioritas: Penting
Pengguna: mahasiswa, Dosen Wali, Kaprodi, dan bagian AAK	
Deskripsi: sistem dapat menampilkan progres dari pengajuan kepada mahasiswa melalui halaman pengajuan yang mahasiswa ajukan. Dosen wali dan kaprodi dapat melihat progres dari pengajuan yang telah mereka setujui/tolak pada halaman pengajuan	

Pemicu: mahasiswa mengajukan pengajuan melalui aplikasi, mahasiswa meng-*upload* pengajuan yang telah di-*edit*, dosen wali menyetujui/menolak pengajuan, kaprodi menyetujui/menolak pengajuan, dan bagian AAK menyetujui/menolak pengajuan

Syarat:

1. Perlunya jaringan internet
 2. Mahasiswa melakukan proses *login*
 3. Dosen wali melakukan proses *login*
 4. Kaprodi melakukan proses *login*
 5. *Staff* AAK melakukan proses *login*
-

Jalan normal:

1. Mahasiswa berhasil *upload* atau *edit* pengajuan
 2. Dosen wali menyetujui/menolak pengajuan
 3. Kaprodi menyetujui/menolak pengajuan
 4. *Staff* AAK menyetujui/menolak pengajuan
-

Jalan alternatif: -

Post condition:

Sistem dapat memperlihatkan proses pada pengajuan didalam tabel *track* progres pengajuan di mahasiswa dan pada halaman pengajuan di dosen wali, kaprodi, dan *staff* AAK.

Pengecualian:-

- Fitur Pengajuan Dan *Upload* Berkas

Tabel 3.8 *Flow of event* fitur pengajuan dan *upload* berkas

ID: UC 03

Prioritas: Penting

Pengguna: Mahasiswa

Deskripsi: sistem memungkinkan mahasiswa untuk melakukan pengajuan di 4 layanan AAK beserta meng-*upload* berkas pada layanan dispensasi sakit dan yudisium

Pemicu: mahasiswa mengakses pengajuan dan meng-*upload* berkas pengajuan

Syarat:

1. Perlunya jaringan internet
-

Jalan normal:

1. Mahasiswa megakses layanan pengajuan
 2. Mahasiswa mengisi *form* pengajuan *online*
 3. Mahasiswa menekan *button upload* pada halaman pengajuan
 4. Mahasiswa memilih *file* yang ingin di *upload*
 5. Mahasiswa menekan *button submit*
-

Jalan alternatif: -

Post condition:

file berhasil di-*upload* dan siap untuk proses *submit*.

Pengecualian: -

- Fitur Notifikasi *Email*

Tabel 3.9 *Flow of event* fitur notifikasi *email*

ID: UC 06

Prioritas: Penting

Pengguna: Mahasiswa, Dosen Wali, Kaprodi, Bagian AAK

Deskripsi: sistem mengirim notifikasi *email* ke pengguna tertentu apabila aksi tertentu telah dilakukan oleh pengguna tertentu, dan pada aplikasi akan muncul tabel *track* progres pengajuan sebagai indikasi dari status pengajuan kepada mahasiswa, dosen wali, dan kaprodi.

Pemicu:

1. Mahasiswa melakukan pengajuan
 2. Dosen wali menyetujui/menolak pengajuan
 3. Kaprodi menyetujui/menolak pengajuan
 4. *Staff* AAK menyetujui/menolak pengajuan
-

Syarat:

1. Perlunya jaringan internet
-

Jalan normal:

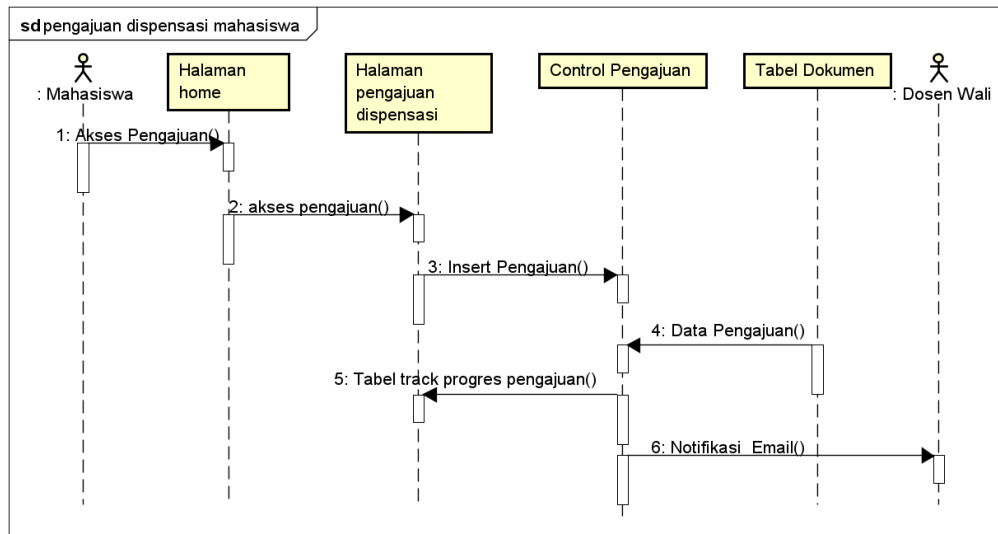
1. Mahasiswa mengunggah atau *edit* data pengajuan
 2. Dosen wali menyetujui atau menolak pengajuan
 3. Kaprodi menyetujui atau menolak pengajuan
 4. bag. AAK menyetujui/menolak pengajuan.
-

Jalan alternatif: -

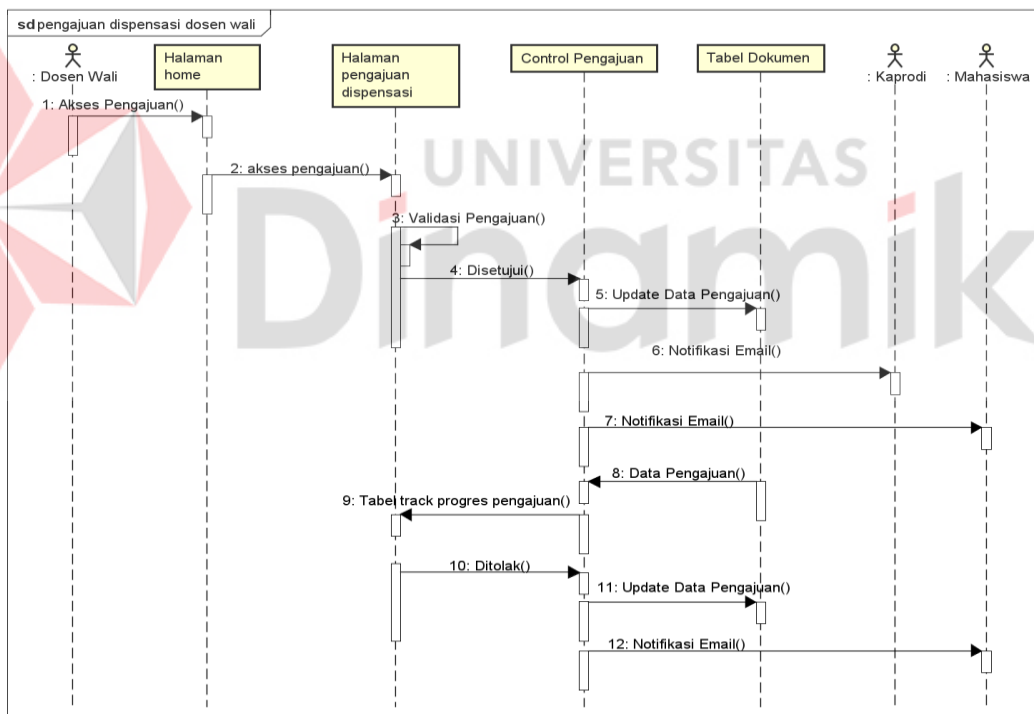
Post condition: notifikasi *email* akan terkirim kepada pengguna sesuai dengan aksi yang dijalankan oleh pengguna.

Pengecualian: -

C. Sequence Diagram



Gambar 3.10 *Sequence Diagram* pengajuan dispensasi sakit bagian mahasiswa

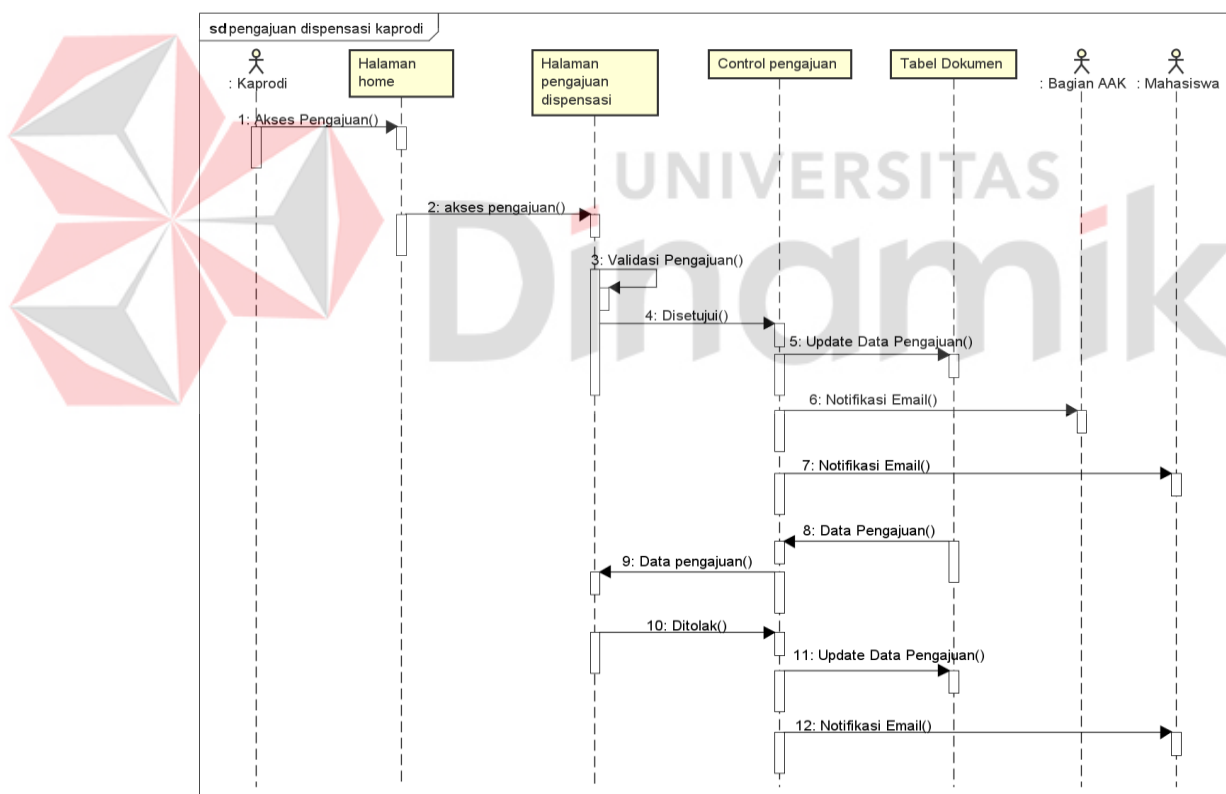


Gambar 3.11 *Sequence Diagram* pengajuan dispensasi sakit bagian dosen wali

Pada bagian mahasiswa dimulai dari melakukan akses pengajuan dan masuk ke halaman *home*. Selanjutnya, mahasiswa melakukan akses pengajuan ke halaman pengajuan dispensasi. Kemudian melakukan *insert* pengajuan yang setelah di-*submit* akan langsung masuk ke *database*. *Control* pengajuan akan menarik data pengajuan yang akan ditampilkan ke halaman pengajuan dispensasi melalui tabel

track progres pengajuan. Dari *control* pengajuan juga akan mengirim notifikasi *email* kepada dosen wali.

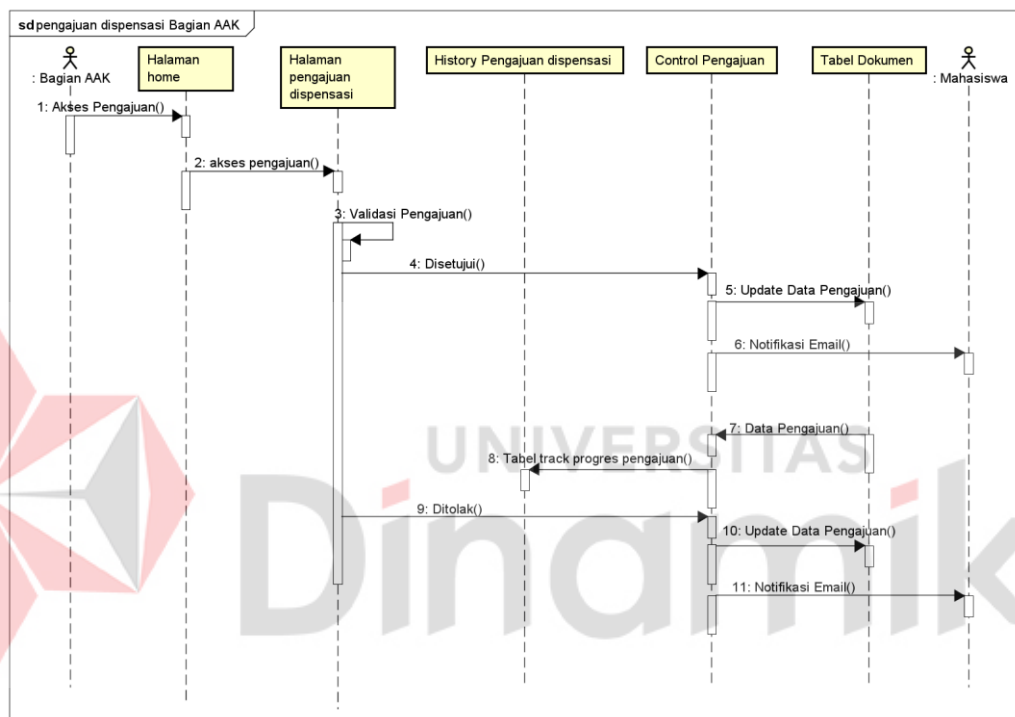
Pengajuan dispensasi pada dosen wali dimulai dari dosen wali melakukan akses pengajuan ke halaman *home*, kemudian melakukan akses pengajuan ke halaman pengajuan dispensasi. Disini, dosen wali melakukan verifikasi pengajuan. Jika pengajuan disetujui, maka data pengajuan pada *database* akan di-*update*. *Control* pengajuan akan mengirimkan notifikasi *email* kepada kaprodi dan konfirmasi *email* pengajuan diterima ke mahasiswa. Kemudian, data pengajuan di *database* akan ditampilkan ke halaman pengajuan oleh *control* pengajuan. Jika pengajuan ditolak, data pengajuan di *database* akan di-*update* dan pada *control* pengajuan akan mengirimkan notifikasi konfirmasi *email* kepada mahasiswa bahwa pengajuan telah ditolak dan harap melakukan *edit* data pengajuan.



Gambar 3.12 *Sequence Diagram* pengajuan dispensasi sakit bagian kaprodi

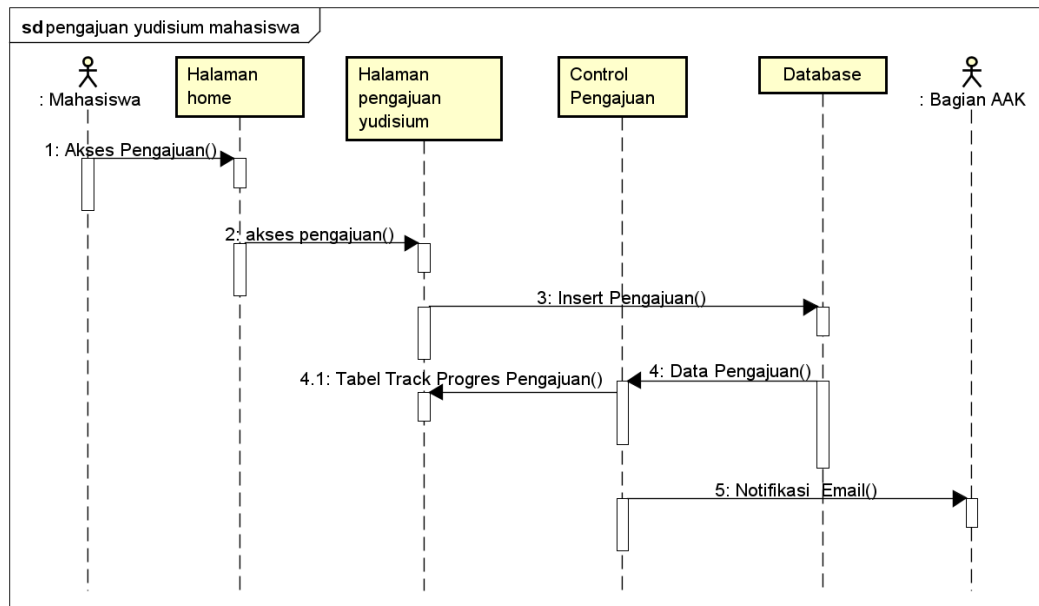
Pengajuan dispensasi pada bagian kaprodi dimulai dari melakukan akses pengajuan menuju halaman *home*. Kemudian melakukan akses pengajuan ke halaman pengajuan dispensasi, disini kaprodi melakukan validasi pengajuan. Jika pengajuan disetujui, maka *data* pengajuan pada *database* akan ter-*update*, *control*

pengajuan akan mengirimkan notifikasi *email* kepada bagian AAK dan notifikasi konfirmasi *email* pengajuan akan terkirim ke mahasiswa. Data pengajuan di *database* akan diambil oleh *control* pengajuan dan ditampilkan di halaman pengajuan dispensasi. Jika pengajuan ditolak, data pengajuan di *database* akan di-*update* dan pada *control* pengajuan akan mengirimkan notifikasi konfirmasi *email* kepada mahasiswa bahwa pengajuan telah ditolak dan harap melakukan *edit* data pengajuan.

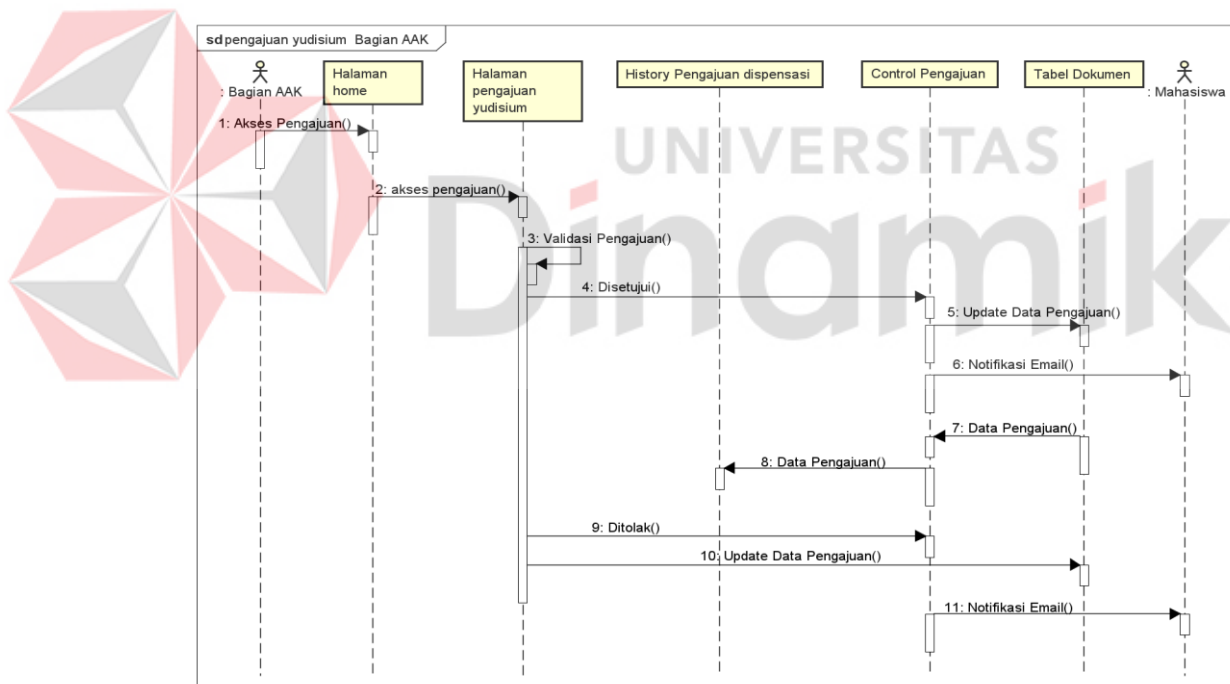


Gambar 3.13 *Sequence Diagram* pengajuan dispensasi sakit bagian AAK

Pengajuan dispensasi pada bagian Administrasi dan Akademi Kemahasiswaan (AAK) dimulai dari melakukan akses pengajuan menuju halaman *home*. Kemudian melakukan akses pengajuan ke halaman pengajuan dispensasi, disini bagian AAK melakukan validasi pengajuan. Jika pengajuan disetujui, maka *data pengajuan* pada *database* akan ter-*update*, *control* pengajuan akan mengirimkan notifikasi konfirmasi *email* pengajuan ke mahasiswa. Data pengajuan di *database* akan diambil oleh *control* pengajuan dan ditampilkan pada halaman pengajuan *history* pengajuan dispensasi. Jika pengajuan ditolak, data pengajuan di *database* akan di-*update* dan pada *control* pengajuan akan mengirimkan notifikasi konfirmasi *email* kepada mahasiswa bahwa pengajuan telah ditolak dan harap melakukan *edit* data pengajuan.



Gambar 3.14 *Sequence Diagram* pengajuan yudisium bagian mahasiswa

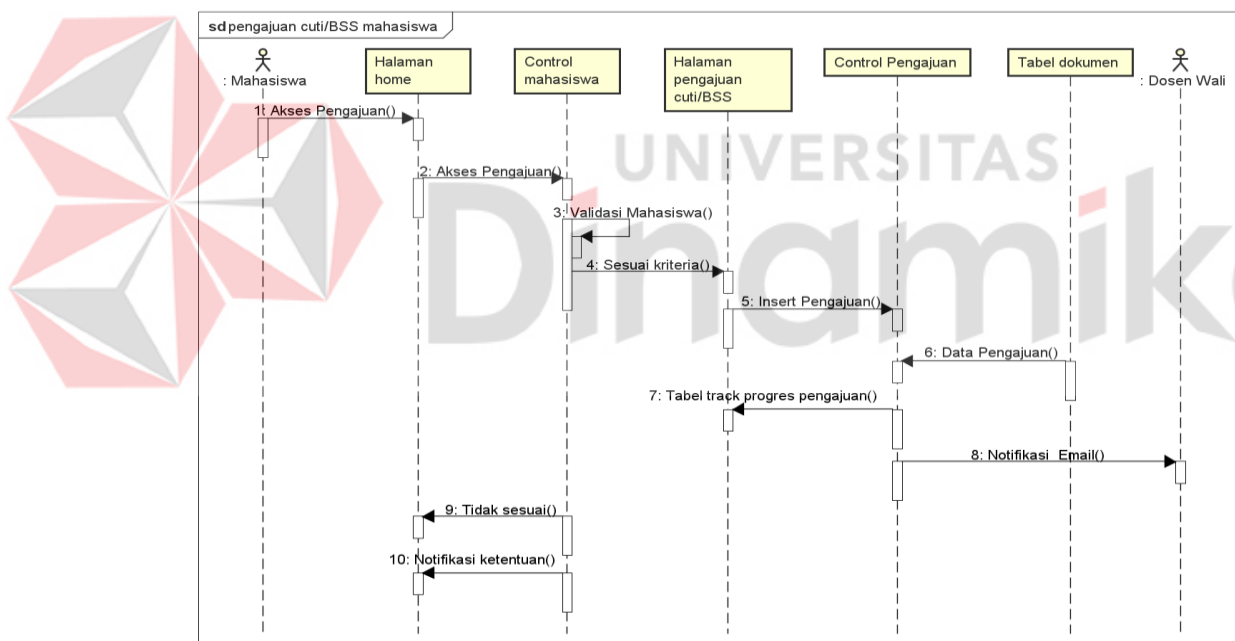


Gambar 3.15 *Sequence Diagram* pengajuan yudisium bagian AAK

Pengajuan yudisium pada mahasiswa dimulai dari melakukan akses pengajuan ke halaman *home*. Kemudian melakukan akses pengajuan ke halaman pengajuan yudisium. Mahasiswa melakukan *insert* pengajuan dan setelah data pengajuan di-*submit* akan masuk ke *database*. Data pengajuan akan diambil oleh *control* pengajuan dari *database* dan ditampilkan pada halaman pengajuan yudisium di tabel *track* progress pengajuan. *Control* pengajuan akan mengirim

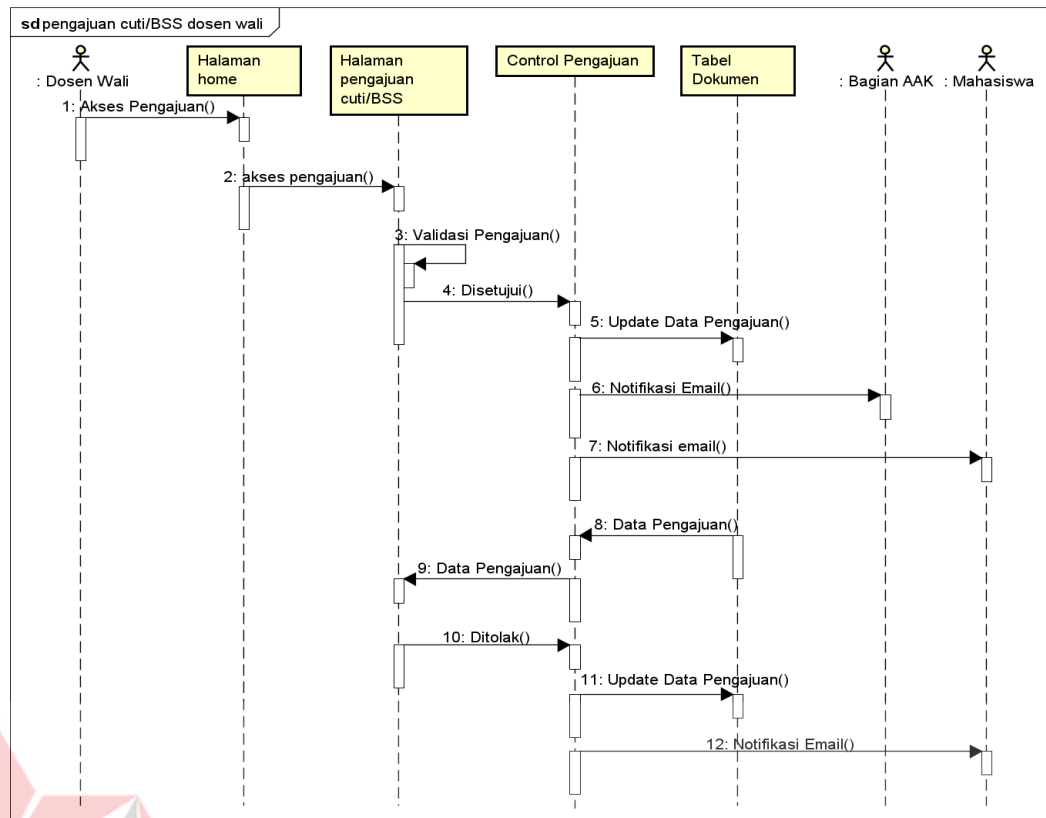
notifikasi *email* ke bagian Administrasi Akademik dan Kemahasiswaan (AAK).

Pengajuan yudisium pada bagian Administrasi Akademik dan Kemahasiswaan (AAK) dimulai dari melakukan akses pengajuan menuju halaman *home*. Kemudian melakukan akses pengajuan ke halaman pengajuan yudisium, disini bagian AAK melakukan validasi pengajuan. Jika pengajuan disetujui, maka *data pengajuan* pada *database* akan *ter-update*, *control* pengajuan akan mengirimkan notifikasi konfirmasi *email* pengajuan ke mahasiswa. Data pengajuan di *database* akan diambil oleh *control* pengajuan dan ditampilkan pada halaman pengajuan *history* pengajuan yudisium. Jika pengajuan ditolak, data pengajuan di *database* akan di-*update* dan pada *control* pengajuan akan mengirimkan notifikasi konfirmasi *email* kepada mahasiswa bahwa pengajuan telah ditolak dan harap melakukan *edit* data pengajuan.

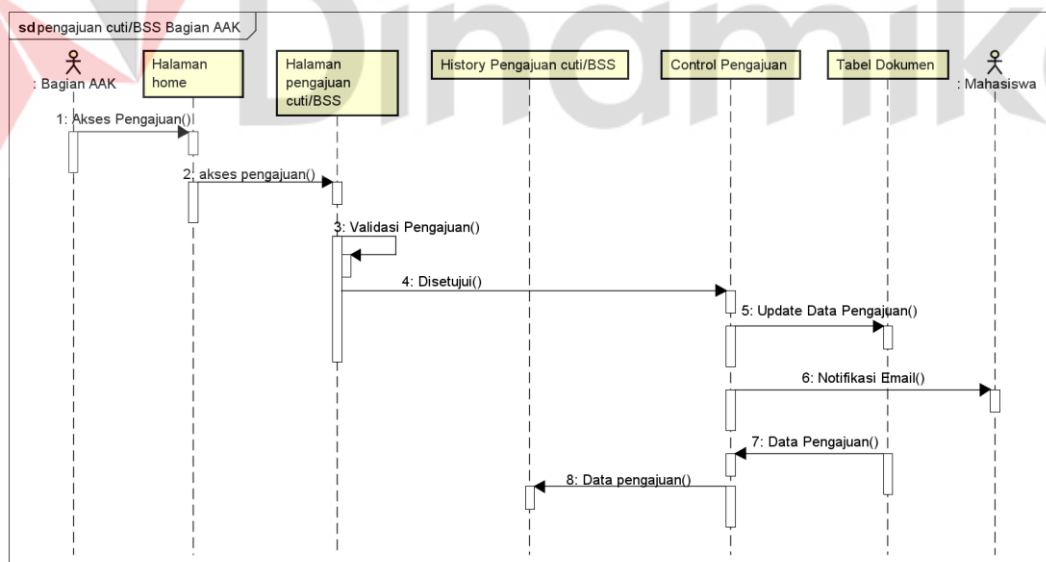


Gambar 3.16 *Sequence Diagram* pengajuan cuti/BSS bagian mahasiswa

Pada bagian mahasiswa dimulai dari melakukan akses pengajuan dan masuk ke halaman *home*. Selanjutnya mahasiswa melakukan akses pengajuan ke halaman pengajuan cuti/BSS. Kemudian melakukan *insert* pengajuan yang setelah di-*submit* akan langsung masuk ke *database*. *Control* pengajuan akan menarik data pengajuan yang akan ditampilkan ke halaman pengajuan dispensasi melalui tabel *track* progres pengajuan. Dari *control* pengajuan akan mengirim *email* kepada dosen wali.



Gambar 3.17 *Sequence Diagram* pengajuan cuti/BSS bagian dosen wali

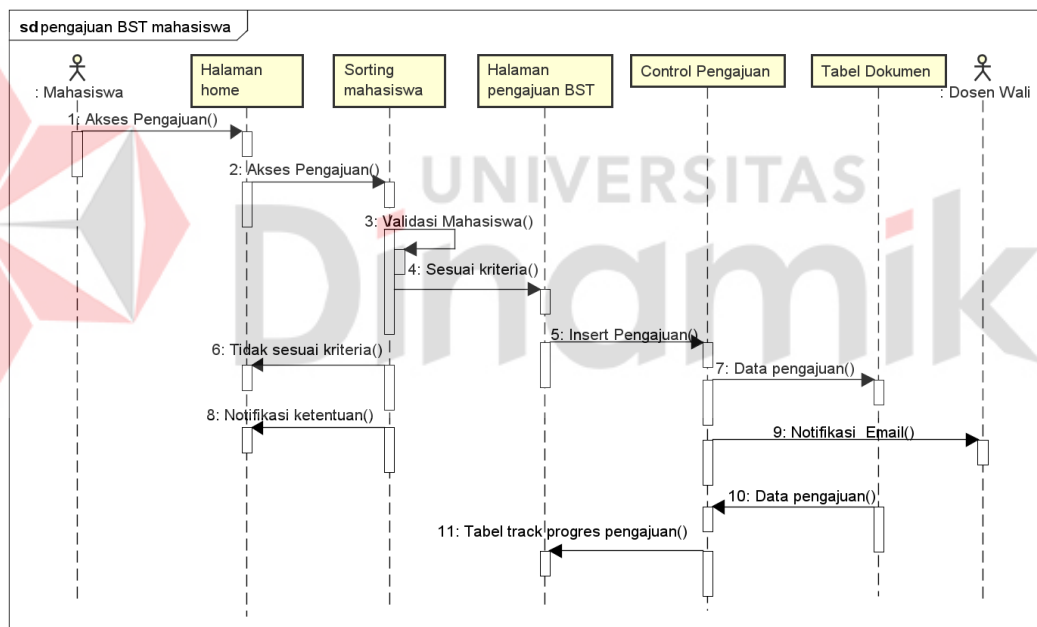


Gambar 3.18 *Sequence Diagram* pengajuan cuti/BSS bagian AAK

Pengajuan cuti/Berhenti Studi Sementara (BSS) pada dosen wali dimulai dari dosen wali melakukan akses pengajuan ke halaman *home*, kemudian melakukan akses pengajuan ke halaman pengajuan cuti/BSS. Disini, dosen wali melakukan verifikasi pengajuan. Jika pengajuan disetujui, maka data pengajuan pada *database* akan di-*update*. *Control*

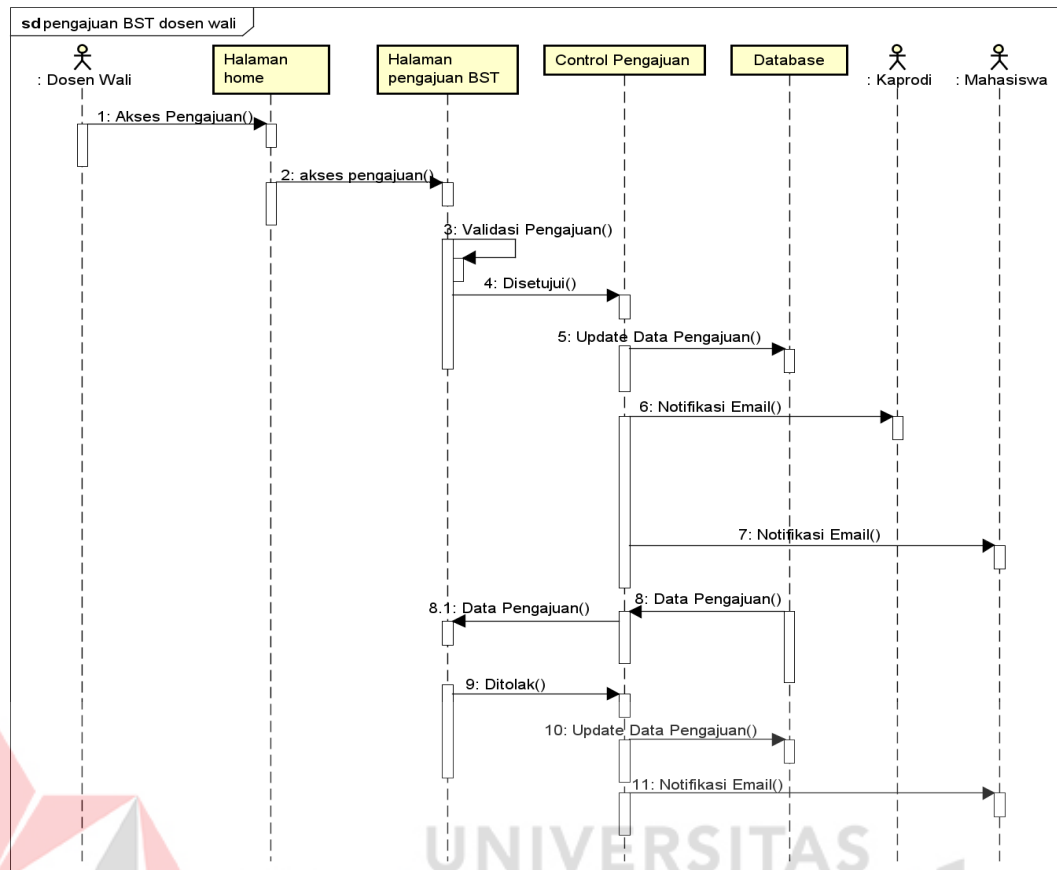
pengajuan akan mengirimkan notifikasi *email* kepada kaprodi dan konfirmasi *email* pengajuan diterima ke mahasiswa. Kemudian, data pengajuan di *database* akan ditampilkan ke halaman pengajuan oleh *control* pengajuan. Jika pengajuan ditolak, data pengajuan di *database* akan di-*update* dan pada *control* pengajuan akan mengirimkan notifikasi konfirmasi *email* kepada mahasiswa bahwa pengajuan telah ditolak.

Pengajuan cuti/BSS pada bagian Administrasi dan Akademi Kemahasiswaan (AAK) dimulai dari melakukan akses pengajuan menuju halaman *home*. Kemudian melakukan akses pengajuan ke halaman pengajuan cuti/BSS, disini bagian AAK melakukan validasi pengajuan. Jika pengajuan disetujui, maka *data pengajuan* pada *database* akan ter-*update*, *control* pengajuan akan mengirimkan notifikasi konfirmasi *email* pengajuan ke mahasiswa. Data pengajuan di *database* akan diambil oleh *control* pengajuan dan ditampilkan pada halaman pengajuan *history* pengajuan dispensasi.



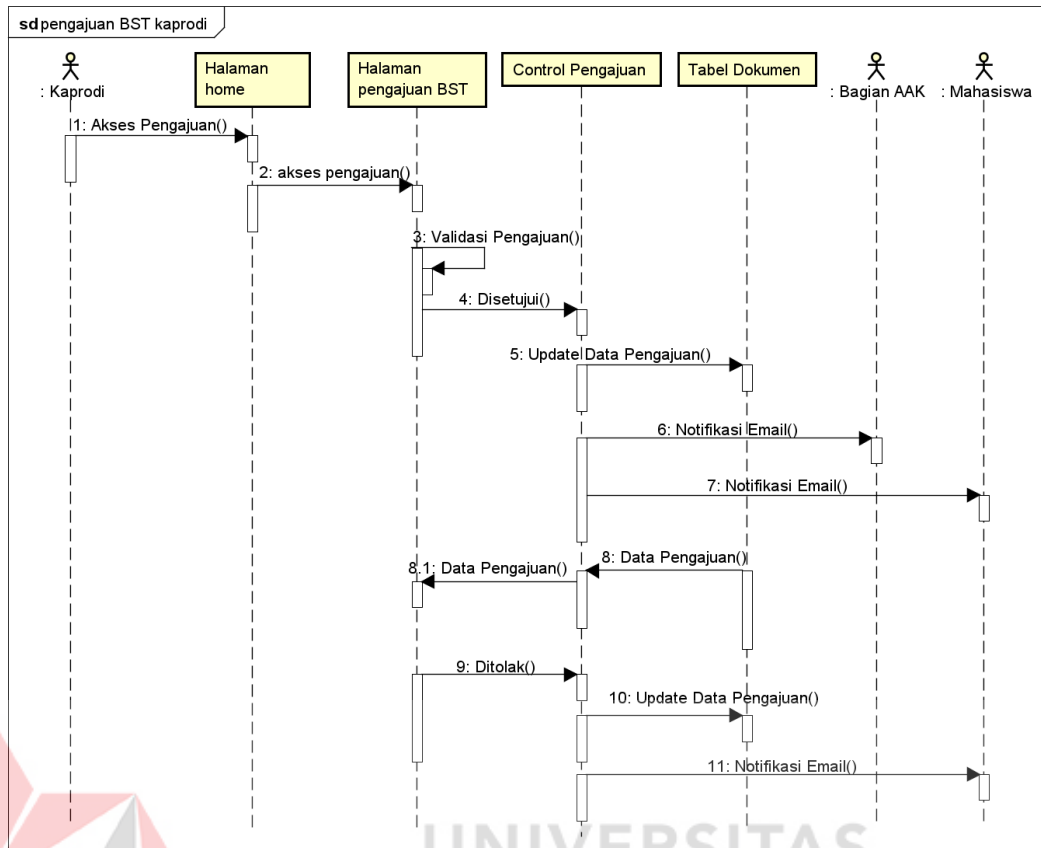
Gambar 3.19 *Sequence Diagram* pengajuan BST bagian mahasiswa

Pada bagian mahasiswa dimulai dari melakukan akses pengajuan dan masuk ke halaman *home*. Selanjutnya, mahasiswa melakukan akses pengajuan ke halaman pengajuan BST. Kemudian melakukan *insert* pengajuan yang setelah di-*submit* akan langsung masuk ke *database*. *Control* pengajuan akan menarik data pengajuan yang akan ditampilkan ke halaman pengajuan dispensasi melalui tabel *track* progres pengajuan. Dari *control* pengajuan juga akan mengirim notifikasi *email* kepada dosen wali.

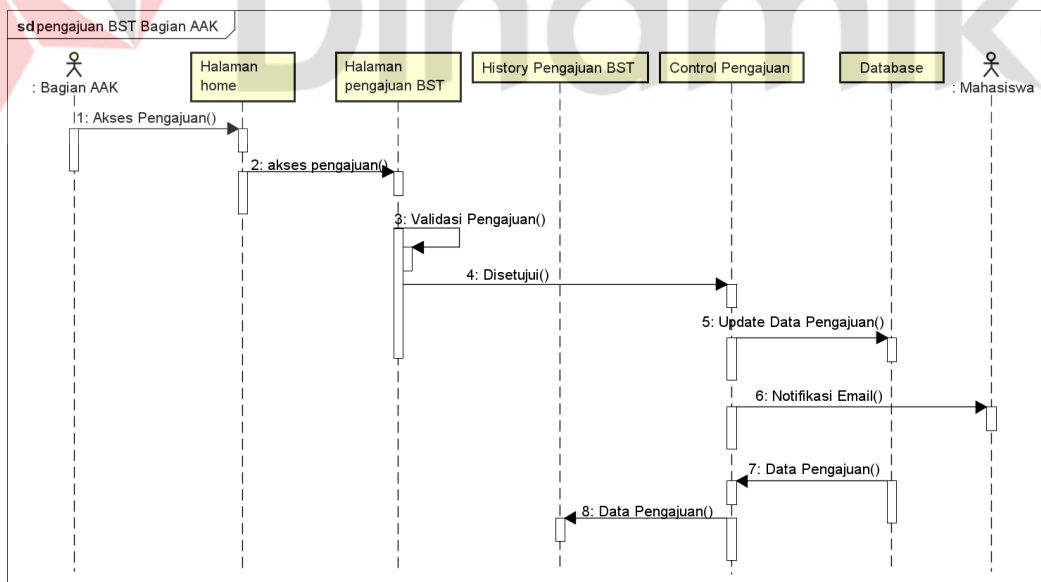


Gambar 3.20 *Sequence Diagram* pengajuan BST bagian dosen wali

Pengajuan Berhenti Studi Tetap (BST) pada dosen wali dimulai dari dosen wali melakukan akses pengajuan ke halaman *home*, kemudian melakukan akses pengajuan ke halaman pengajuan BST. Disini, dosen wali melakukan verifikasi pengajuan. Jika pengajuan disetujui, maka data pengajuan pada *database* akan di-*update*. *Control* pengajuan akan mengirimkan notifikasi *email* kepada kaprodi dan konfirmasi *email* pengajuan diterima ke mahasiswa. Kemudian, data pengajuan di *database* akan ditampilkan ke halaman pengajuan oleh *control* pengajuan. Jika pengajuan ditolak, data pengajuan di *database* akan di-*update* dan pada *control* pengajuan akan mengirimkan notifikasi konfirmasi *email* kepada mahasiswa bahwa pengajuan telah ditolak.



Gambar 3.21 Sequence Diagram pengajuan BST bagian kaprodi



Gambar 3.22 Sequence Diagram pengajuan BST bagian AAK

Pengajuan Berhenti Studi Tetap (BST) pada kaprodi dimulai dari kaprodi melakukan akses pengajuan ke halaman *home*, kemudian melakukan akses pengajuan ke halaman pengajuan BST. Disini, kaprodi melakukan verifikasi pengajuan. Jika pengajuan disetujui,

maka data pengajuan pada *database* akan di-*update*. *Control* pengajuan akan mengirimkan notifikasi *email* kepada bagian AAK dan konfirmasi *email* pengajuan diterima ke mahasiswa. Kemudian, data pengajuan di *database* akan ditampilkan ke halaman pengajuan oleh *control* pengajuan. Jika pengajuan ditolak, data pengajuan di *database* akan di-*update* dan pada *control* pengajuan akan mengirimkan notifikasi konfirmasi *email* kepada mahasiswa bahwa pengajuan telah ditolak.

Pengajuan BST pada bagian Administrasi dan Akademi Kemahasiswaan (AAK) dimulai dari melakukan akses pengajuan menuju halaman *home*. Kemudian melakukan akses pengajuan ke halaman pengajuan BST, disini bagian AAK melakukan validasi pengajuan. Jika pengajuan disetujui, maka *data pengajuan* pada *database* akan ter-*update*, *control* pengajuan akan mengirimkan notifikasi konfirmasi *email* pengajuan ke mahasiswa. Data pengajuan di *database* akan diambil oleh *control* pengajuan dan ditampilkan pada halaman pengajuan *history* pengajuan dispensasi.

3.3 Tahap Akhir

Tahap akhir adalah dimana proses pengembangan aplikasi sudah mencapai tahapakhir atau sudah selesai dimana aplikasi dapat berjalan dengan baik tanpa ada masalah.

3.3.1 *Sprint Retrospective*

Tahap ini adalah evaluasi dari tahap pengembangan yang telah dikerjakan. Tahap ini memiliki tujuan agar kekurangan pada *sprint* sebelumnya dapat dideteksi dan dijadikan bahan untuk koreksi pada tahap pengembangan.

3.3.2 *Sprint Review*

Tahap ini adalah evaluasi dari *product backlog* yang telah dikerjakan menggunakan metode *Blackbox testing*, gunanya untuk mengetahui apakah sistem yang telah dibangun mengeluarkan *output* yang diharapkan.

BAB IV

HASIL DAN PEMBAHASAN

Pada bab IV akan dibahas hasil implementasi sistem yang terdiri atas 3 jenis iterasi. Iterasi pertama membahas *login* untuk mahasiswa, dosen wali, kaprodi, dan bagian AAK dan tampilan *monitoring* jalannya berkas untuk semua pengguna. Iterasi kedua membahas *upload* berkas serta sortir berkas yang masuk sesuai ketentuan. Iterasi ketiga membahas tentang sortir mahasiswa yang berhak untuk mengakses layanan tertentu, notifikasi *email* saat pengguna melakukan aksi-aksi tertentu, dan *export data* pada laporan pengajuan di bagian AAK via Microsoft Excel. Detil setiap iterasi dibahas pada bagian berikut.

4.1 *Sprint*

Setelah tahap perancangan *sprint* dilakukan, langkah selanjutnya adalah implementasi *sprint* kedalam sistem sesuai dengan *product backlog* yang telah dibuat. Tahap pengerjaan ini dicatat di aplikasi Trello agar mudah untuk memantau urutan kerja yang dilakukan. Tampilan tahap pengerjaan dapat dilihat di lampiran 1.

Semua *product backlog* dipindah kedalam kolom “*To Do*”, kemudian *product backlog* yang sedang dikerjakan akan dipindah ke kolom “*Doing*”, dan yang sudah dikerjakan akan dipindah ke kolom “*Done*”.

4.1.1 *Sprint* Iterasi Pertama

Sprint Iterasi pertama menghasilkan halaman *login* untuk mahasiswa, Dosen Wali, Kaprodi, dan Bagian AAK serta tampilan tabel *track progres* untuk mahasiswa, dosen wali, dan kaprodi.

1. *Sprint Review* Iterasi Pertama

Uji coba dilakukan untuk fitur *login* dan tampilan tabel *track progres* pengajuan untuk mahasiswa, dosen wali, dan kaprodi. Hasilnya, fitur sukses berjalan sesuai dengan yang diharapkan. Untuk hasil uji coba dapat dilihat pada lampiran 8. Untuk tampilan tabel *track progres* pengajuan dapat dilihat pada gambar 4.1.

Cuti(BSS)		Home / Pengajuan Layanan-Cuti(BSS)		
NIM - Nama	Jurusan	Alasan Pengajuan	Tanggal Masuk	Status
17410100038 - Ahmad Rizqa	S1 Sistem Informasi	kerja	2022-08-20 10:44:13	Proses ke Dosen Wali

Gambar 4.1 Tampilan tabel *track* progres pengajuan cuti/BSS

Gambar diatas adalah alur kerja yang terjadi setelah mahasiswa melakukan pengajuan pada cuti/Berhenti Studi Sementara (BSS), *control* pengajuan akan menarik data pengajuan dari tabel dokumen di *database* dan menampilkan pengajuan yang baru saja di-*submit*. Pada kolom “status”, tulisan akan berubah jika pihak terkait (dosen wali, kaprodi, dan bagian AAK) telah menyetujui atau menolak. Jika pihak terkait menolak pengajuan mahasiswa, maka akan ada kolom tambahan yang muncul yaitu kolom “aksi” yang berisikan tombol “*edit data*” agar mahasiswa dapat melakukan *edit data* pada pengajuannya.

4.1.2 *Sprint Iterasi Kedua*

Uji coba dilakukan untuk fitur *upload* dan sortir berkas pengajuan. Hasilnya, fitur sukses berjalan sesuai dengan yang diharapkan. Untuk hasil uji coba dapat dilihat pada lampiran 9. Untuk tampilan *login* dan tabel *track* progres pengajuan dapat dilihat pada gambar 4.2 dan 4.3.

Yudisium
Home / Pengajuan Layanan-Yudisium

<table border="0" style="width: 100%;"> <tr> <td style="width: 50%;"> NIM <input type="text" value="17410100038"/> </td> <td style="width: 50%;"> Nama <input type="text" value="Ahmad Rizqa"/> </td> </tr> <tr> <td> Tempat Lahir <input type="text" value="Sidoarjo"/> </td> <td> Tanggal Lahir <input type="text" value="12/12/1998"/> </td> </tr> <tr> <td colspan="2"> No. KTP <input type="text" value="3515099"/> </td> </tr> <tr> <td colspan="2"> Alamat <input type="text" value="Taman Cahaya Asri X-90, Sidoarjo"/> </td> </tr> <tr> <td> No. Telfon <input type="text"/> </td> <td> E-mail Dinamika <input type="text" value="17410100038@dinamika.ac.id"/> </td> </tr> </table>	NIM <input type="text" value="17410100038"/>	Nama <input type="text" value="Ahmad Rizqa"/>	Tempat Lahir <input type="text" value="Sidoarjo"/>	Tanggal Lahir <input type="text" value="12/12/1998"/>	No. KTP <input type="text" value="3515099"/>		Alamat <input type="text" value="Taman Cahaya Asri X-90, Sidoarjo"/>		No. Telfon <input type="text"/>	E-mail Dinamika <input type="text" value="17410100038@dinamika.ac.id"/>	<table border="0" style="width: 100%;"> <tr> <td style="width: 50%;"> Upload Bukti SSKM* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small> </td> <td style="width: 50%;"> Upload Sertifikat Toefl/Toeic* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small> </td> </tr> <tr> <td> Upload Ijazah SMA* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small> </td> <td> Upload Akta Kelahiran* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small> </td> </tr> <tr> <td> Upload Kartu Keluarga* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small> </td> <td> Upload Kartu Tanda Mahasiswa* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small> </td> </tr> <tr> <td> Upload Kartu Tanda Penduduk* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small> </td> <td></td> </tr> </table>	Upload Bukti SSKM* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small>	Upload Sertifikat Toefl/Toeic* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small>	Upload Ijazah SMA* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small>	Upload Akta Kelahiran* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small>	Upload Kartu Keluarga* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small>	Upload Kartu Tanda Mahasiswa* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small>	Upload Kartu Tanda Penduduk* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small>	
NIM <input type="text" value="17410100038"/>	Nama <input type="text" value="Ahmad Rizqa"/>																		
Tempat Lahir <input type="text" value="Sidoarjo"/>	Tanggal Lahir <input type="text" value="12/12/1998"/>																		
No. KTP <input type="text" value="3515099"/>																			
Alamat <input type="text" value="Taman Cahaya Asri X-90, Sidoarjo"/>																			
No. Telfon <input type="text"/>	E-mail Dinamika <input type="text" value="17410100038@dinamika.ac.id"/>																		
Upload Bukti SSKM* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small>	Upload Sertifikat Toefl/Toeic* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small>																		
Upload Ijazah SMA* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small>	Upload Akta Kelahiran* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small>																		
Upload Kartu Keluarga* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small>	Upload Kartu Tanda Mahasiswa* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small>																		
Upload Kartu Tanda Penduduk* <input type="button" value="Choose File"/> No file chosen <small>Max. 2 MB dan format .JPG, .JPEG, .PNG, dan .PDF</small>																			

Submit
Activate Window
Go to Settings to activate Windows

Gambar 4.2 Tampilan halaman *upload* pada pengajuan yudisium

Gambar diatas adalah tampilan *menu upload* berkas pada pengajuan yudisium, dibawah tombol “*Choose file*” terdapat ketentuan *file* yang bisa diterima oleh sistem yang menggantikan alur kerja pengajuan dimana mahasiswa harus membawa berkas yudisium. Adanya fitur ini dapat membantu mahasiswa untuk melakukan pengajuan terbukti dari hasil kuesioner II dengan indeks kepuasan sebesar 89%.

The screenshot shows a web form titled "Dispensasi" with a breadcrumb "Home / Pengajuan Layanan-Dispensasi". The form contains the following fields:

- NIM:** 17410100038
- Nama:** Ahmad Rizqa
- Jurusan:** S1 Sistem Informasi
- Fakultas:** Teknik dan Informatika
- Tanggal Awal:** dd/mm/yyyy
- Tanggal Akhir:** dd/mm/yyyy
- Alasan Pengajuan:** A large text area for entering the reason for the dispensation.
- Upload berkas dispensasi*:** A section with a "Choose File" button, the text "No file chosen", and "Max. 2 MB dengan format .PDF".
- Submit:** A blue button at the bottom of the form.

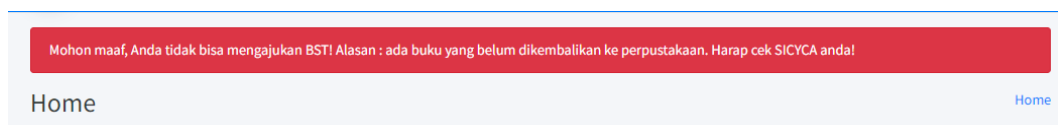
Gambar 4.3 Tampilan halaman *upload* pada pengajuan dispensasi sakit

Gambar diatas adalah tampilan *menu upload* berkas pada pengajuan dispensasi sakit, dibawah tombol “*Choose file*” terdapat ketentuan *file* yang bisa diterima oleh sistem yang menggantikan alur kerja pengajuan dimana mahasiswa harus membawa berkas yudisium. Adanya fitur ini dapat membantu mahasiswa untuk melakukan pengajuan terbukti dari hasil kuesioner II dengan indeks kepuasan sebesar 89%.

4.1.3 *Sprint Iterasi Ketiga*

Uji coba dilakukan untuk sortir mahasiswa yang berhak untuk mengakses layanan tertentu, notifikasi *email* saat pengguna melakukan aksi-aksi tertentu, dan *export data* pada laporan pengajuan di Bagian AAK *via* Microsoft Excel.. Hasilnya, fitur sukses berjalan sesuai dengan yang diharapkan. Untuk hasil uji coba dapat

dilihat pada lampiran 10. Untuk tampilan *login* dan tabel *track* progres pengajuan dapat dilihat pada gambar 4.4-4.5 .



Gambar 4.4 Tampilan notifikasi *sorting* mahasiswa di pengajuan BST

Gambar diatas adalah notifikasi pada fitur *sorting* mahasiswa yang muncul ketika mahasiswa memiliki kriteria yang tidak sesuai dengan sistem pada pengajuan Berhenti Studi Tetap (BST). Fitur tersebut ada untuk memastikan bahwa mahasiswa yang dapat melakukan pengajuan adalah yang telah memenuhi kriteria pengajuan BST, yaitu bebas Biaya Operasional Pendidikan (BOP) dan tidak ada tanggungan peminjaman perpustakaan.

NIM	Nama	E-mail Mahasiswa	Jurusan	Jenis	Status	Tanggal masuk	Tanggal selesai
1	Jumari Martaka Firgantoro M.Kom.	jailani.kanda@example.org	S1 Sistem Informasi	BST	selesai	2021-07-15	2021-07-17
2	Titi Riyanti	qaryani@example.org	S1 Akutansi	BST	selesai	2021-09-03	2021-09-05
3	5 Amalia Rahayu S.Gz	rmahendra@example.org	S1 Desain Komunikasi Visual	BST	selesai	2021-07-21	2021-07-25
4	153 Asmuni Najmudin S.Pd	tri16@example.com	S1 Desain Komunikasi Visual	BST	selesai	2021-04-25	2021-04-27
5	218 Panca Jefri Uwais S.I.Kom	slestari@example.com	S1 Akutansi	BST	ditolak	2021-08-09	2021-08-15
6	93 Yono Iswahyudi	chelsea.halim@example.org	S1 Akutansi	BST	selesai	2020-07-16	2021-07-17
7	547 Lukman Hidayanto	habibi.galak@example.net	S1 Sistem Informasi	BST	selesai	2020-07-22	2021-07-30

Gambar 4.5 Tampilan hasil *export* data pada pengajuan dispensasi sakit

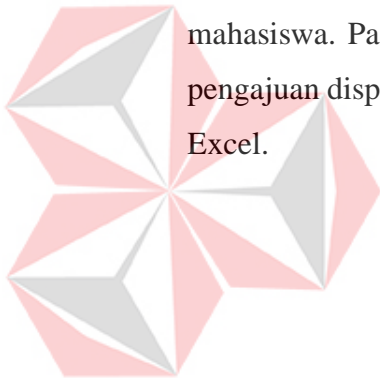
Gambar diatas adalah hasil *export* data pengajuan dari bagian AAK pada pengajuan dispensasi sakit dengan format Microsoft Excel.

4.2 Pembahasan Sistem

Peneliti melakukan kuesioner II kepada 30 mahasiswa mengenai *workflow* layanan Administrasi Akademik dan Kemahasiswaan (AAK) yang telah dikembangkan dengan bantuan aplikasi. Berdasarkan hasil kuesioner II, didapati indkes kepuasan terhadap rancang bangun *workflow* aplikasi layanan AAK yang terdiri dari proses pengajuan, proses persetujuan dari pihak terkait (dosen wali, kaprodi, dan bagian AAK), dan proses *tracking* progres pengajuan melalui notifikasi *email* dan tabel *track* progres pengajuan di dalam aplikasi sebesar 89%. Pada kuesioner I, didapati indeks kepuasan mahasiswa terhadap *workflow* yang

sedang berjalan sebesar 78.44%. Dari hasil tersebut, didapati peningkatan kepuasan mahasiswa terhadap *workflow* yang sekarang sebesar 10.56% yang menandakan rancang bangun *workflow* sudah bisa dikatakan lebih baik dari segi proses pengajuan, proses persetujuan dari pihak terkait, serta proses *tracking* progres pengajuan melalui notifikasi *email* dan tabel *track* progres pengajuan di dalam aplikasi. Untuk informasi lebih lanjut, hasil kuesioner II dapat dilihat pada lampiran 15.

Hasil dari pengujian menggunakan *black box testing*, sistem dapat melakukan *upload* pada pengajuan dan berkas pengajuan. Sistem dapat memilah mahasiswa mana yang dapat melakukan pengajuan cuti/Berhenti Studi Sementara (BSS) dan Berhenti Studi Tetap (BST). Notifikasi *email* berjalan dengan lancar dan menyampaikan informasi terkait pengajuan ke pengguna. Tabel *track* progres pengajuan juga dapat menampilkan status dari pengajuan yang telah diajukan oleh mahasiswa. Pada halaman AAK, sistem berhasil melakukan *export* pada *history* pengajuan dispensasi sakit, yudisium, cuti/BSS, dan BST kedalam format Microsoft Excel.



UNIVERSITAS
Dinamika

BAB V

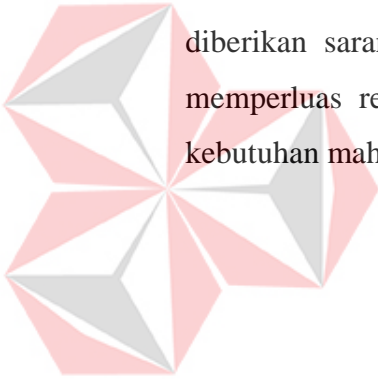
PENUTUP

5.1 Kesimpulan

Berdasarkan hasil kuesioner II yang membahas kemudahan, kecepatan, dan kemampuan aplikasi dalam membantu mahasiswa, didapati indeks kepuasan sebesar 89%. Maka dapat disimpulkan, rancang bangun *workflow* aplikasi layanan AAK telah dinilai oleh mahasiswa dapat membantu dalam proses pengajuan, persetujuan kepada pihak terkait (dosen wali dan kaprodi), serta pelacakan progres pengajuan.

5.2 Saran

Berdasarkan hasil dari implementasi dan evaluasi yang telah dicoba, dapat diberikan saran untuk pengembangan lebih lanjut pada sistem ini yaitu bisa memperluas responden untuk mendapatkan hasil yang lebih akurat mengenai kebutuhan mahasiswa mengenai rancang bangun *wokrflow* untuk kedepannya.



UNIVERSITAS
Dinamika

DAFTAR PUSTAKA

- Adani, M. R. (2020, August 8). Metode Agile: Pengertian, Jenis, Manfaat, Dan 12 Prinsipnya. <https://www.sekawanmedia.co.id/blog/metode-agile-development>. Diakses tanggal 10 Juli 2022.
- Aminudin. (2015). *Cara Efektif Belajar Framework Laravel*. Lokomedia.
- Alqudah, M. & Razali, R., 2016. A Review of Scaling Agile Methods in Large Software Development. *International Journal on Advanced Science, Engineering and Information Technology*, 6(6), pp. 828-837.
- Kustiyahningsih, Yeni. (2011). *Pemrograman Basis Data Berbasis Web Menggunakan PHP & MySQL*. Jakarta: Graha Ilmu.
- Muhajir, A., Tjandrarini, A., & Heru Susilo, T. (2016). *Rancang Bangun Aplikasi Penanganan Komplain Menggunakan Administrative Workflow System*. *Jurnal Sistem informasi dan Komputer Akuntansi*.
- Novrianto, Y., Sulistiowati, S., & Lemantara, J. (2016). *Rancang Bangun Aplikasi Workflow Seleksi Penerima Beasiswa pada STIKES Yayasan Dr. Soetomo Surabaya Berbasis Web*. *Jurnal Sistem informasi dan Komputer Akuntansi*.
- Pebriantono, P., Nurcahyawati, V., & Sony Erstiawan, M. (2016). *Rancang Bangun Aplikasi Workflow Pelaporan Pemakaian Spare Part Kapal Berbasis Web Pada Pt. Rukina Sukses Abadi*.
- Prasetio, Y. L. et al., 2014. Pengembangan Aplikasi Penjadwalan Wisata Harian pada Smartphone dengan Pendekatan Scrum. *ComTech: Computer, Mathematics and Engineering Applications*, 5(2), pp. 534- 543.
- Rafinska, K. (2020, Juni 17). Internet. *Workflow Management System: Istilah & Manfaatnya dalam perusahaan*. <https://www.online-pajak.com/seputar-e-filing/workflow-management-system>. Diakses tanggal 8 Juli 2022.
- Robith Adani, M. (2020, Agustus 07). Internet. <https://www.sekawanmedia.co.id/blog/pengertian-framework>. Diakses tanggal 8 Juli 2022.
- Satvika Iswari, N. (2012). *Tinjauan Proses Elisitasi Kebutuhan Perangkat Lunak Menggunakan Metode Agile*. *Jurnal Ilmu Teknik Informatika UMN*.