

**PENGEMBANGAN SISTEM KENAIKAN PANGKAT DAN
JABATAN DOSEN FAKULTAS ILMU KOMPUTER UNIVERSITAS
BRAWIJAYA**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Muhammad Safreza
NIM: 155150200111038



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2019**

PENGESAHAN

PENGEMBANGAN SISTEM KENAIKAN PANGKAT DAN JABATAN DOSEN FAKULTAS
ILMU KOMPUTER UNIVERSITAS BRAWIJAYA

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Muhammad Safreza
NIM: 155150200111038

Skripsi ini telah diuji dan dinyatakan lulus pada
02 Mei 2019
Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II



Denny Sagita R., S.Kom, M.Kom
NIP: 19851124 201504 1 001



Agi Putra Kharisma, S.T, M.T
NIK: 2013048 604301 001

Mengetahui
Ketua Jurusan Teknik Informatika



Astoto Kurniawan, S.T, M.T, Ph.D
NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

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

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

Malang, 8 April 2019



Muhammad Safreza

NIM: 155150200111038

KATA PENGANTAR

Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat, taufik dan hidayah-Nya sehingga laporan skripsi yang berjudul “Pengembangan Sistem Kenaikan Pangkat dan jabatan Dosen Fakultas Ilmu Komputer Universitas Brawijaya” ini dapat terselesaikan. Penulis menyadari bahwa skripsi ini tidak akan berhasil tanpa bantuan dari beberapa pihak. Oleh karena itu, penulis ingin menyampaikan rasa hormat dan terima kasih kepada:

1. Bapak Bapak Denny Sagita Rusdianto, S.Kom, M.Kom dan Bapak Agi Putra Kharisma, S.T, M.T selaku Pembimbing skripsi yang telah dengan sabar membimbing dan mengarahkan penulis sehingga dapat menyelesaikan skripsi ini.
2. Bapak Agus Wahyu Widodo , S.T, M.Cs. selaku Ketua Program Studi Teknik Informatika Universitas Brawijaya.
3. Bapak Tri Astoto Kurniawan , S.T, M.T, Ph.D selaku Ketua Jurusan Teknik Informatika Universitas Brawijaya.
4. Bapak Wayan Firdaus Mamudy, S.Si, M.T, Ph.D selaku Dekan Fakultas Ilmu Komputer Universitas Brawijaya.
5. Seluruh Bapak dan Ibu dosen Fakultas Ilmu Komputer Universtas Brawijaya atas segala bimbingan dan ilmu yang diberikan kepada penulis.
6. Ibu dan keluarga dirumah terima kasih atas doa dan wejangannya, Sungguh berarti bagi penulis untuk menyelesaikan skripsi ini.
7. Teman-teman teknik informatika 2015 atas bantuan, motivasi dan memberikan waktu untuk berdiskusi selama ini.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih banyak kekurangan, sehingga saran dan kritik yang membangun sangat penulis harapkan. Akhir kata penulis berharap skripsi ini dapat membawa manfaat bagi semua pihak yang menggunakannya..

Malang, 8 April 2019

Penulis

muhammad.safreza1@gmail.com

ABSTRAK

Muhammad Safreza, Pengembangan Sistem Kenaikan Pangkat dan jabatan Dosen Fakultas Ilmu Komputer Universitas Brawijaya.

Pembimbing : Denny Sagita R., S.Kom, M.Kom dan Agi Putra Kharisma, S.T, M.T

Jabatan fungsional dosen adalah kedudukan yang menunjukkan tugas, tanggung jawab, wewenang dan hak seseorang Dosen dalam suatu satuan Pendidikan Tinggi yang dalam pelaksanaannya didasarkan pada keahlian tertentu serta bersifat mandiri. Sementara pangkat adalah kedudukan yang menunjukkan tingkat seseorang pegawai berdasarkan jabatannya dalam rangkaian susunan kepegawaian dan digunakan sebagai dasar penggajian. Di Fakultas Ilmu Komputer Universitas Brawijaya dari temuan lapangan beberapa dosen FILKOM bahwa dosen tersebut memiliki banyak aktivitas. Lalu pada saat proses validasi dan verifikasi berkas di kepegawaian FILKOM usulan sering terjadi kesalahan sehingga tidak memiliki waktu untuk memproses revisi lebih lanjut dalam mengumpulkan berkas 4 bidang dan berkas administrasi untuk usulan kenaikan pangkat dan/atau jabatan fungsional. Ditambah lagi dengan dosen masih melakukan penghitungan AK secara manual dengan melihat panduan sehingga memungkinkan kesalahan penghitungan bahkan fatalnya bisa salah dalam penghitungan AK pada setiap kegiatannya. Pada penelitian ini bertujuan membantu dosen FILKOM dalam mengumpulkan berkas 4 bidang, membantu validasi dan verifikasi berkas 4 bidang, memantau status usulan, dan juga membantu dosen dalam penghitungan AK kegiatan dosen secara otomatis berdasarkan pedoman penilaian AK. Pada penelitian ini menggunakan metode pengembangan perangkat lunak *prototyping*. Dan pada implementasinya menggunakan *webservice dummy* untuk mengambil data berkas BKD, staf dan dosen Universitas Brawijaya Khususnya Fakultas Ilmu Komputer. Hasil penelitian ini adalah dihasilkan 28 kebutuhan fungsional dan 1 kebutuhan non-fungsional yang diperoleh dari 1 proses iterasi *prototyping*. Dari hasil pengujian unit, integrasi dan uji *compatibility* lingkungan *browser*, didapatkan hasil sebesar 100% valid.

Kata kunci: pangkat, jabatan fungsional, *prototyping*

ABSTRACT

Muhammad Safreza, *Software Development System of Rank and Position of Lecturer at the Faculty of Computer Science Universitas Brawijaya.*

Advisor: Denny Sagita R., S.Kom, M.Kom and Agi Putra Kharisma, S.T, M.T

The functional position of a lecturer is a position which shows the duties, responsibilities, authority, and rights of a lecturer in a Higher Education unit which in its implementation is based on certain expertise and is independent. While rank is a position that shows the level of an employee based on his position in a series of staffing and is used as a base for payroll. In the Faculty of Computer Science, Universitas Brawijaya, from the field findings of several FILKOM lecturers, the lecturer had many activities. Then when the file validation and verification process in the FILKOM employee proposed a frequent error so that it did not have time to process further revisions in collecting files of activities in 4 fields and administrative files for proposed promotions and/or functional positions. Coupled with the lecturer still doing the AK calculation manually by looking at the guide so that it allows even fatal calculation errors can be mistaken in the calculation of AK in each activity. In this study aims to help FILKOM lecturers in collecting 4 fields of activities files, assisting in the validation and verification of 4 fields of activities files, monitoring the status of proposals, and also assisting lecturers in calculating credit number of lecturer activities automatically based on AK assessment guidelines. In this study using prototyping software development methods. And the implementation uses a dummy web service to retrieve BKD file data, staff and lecturers of Universitas Brawijaya, especially the Faculty of Computer Science. The results of this study are 28 functional requirements and 1 non-functional requirement obtained from 1 prototyping iteration process. From the results of unit testing, integration and compatibility testing of the browser environment, the results are 100% valid.

Keywords: Lecturer rank, functional position of a lecturer, *prototyping*

DAFTAR ISI

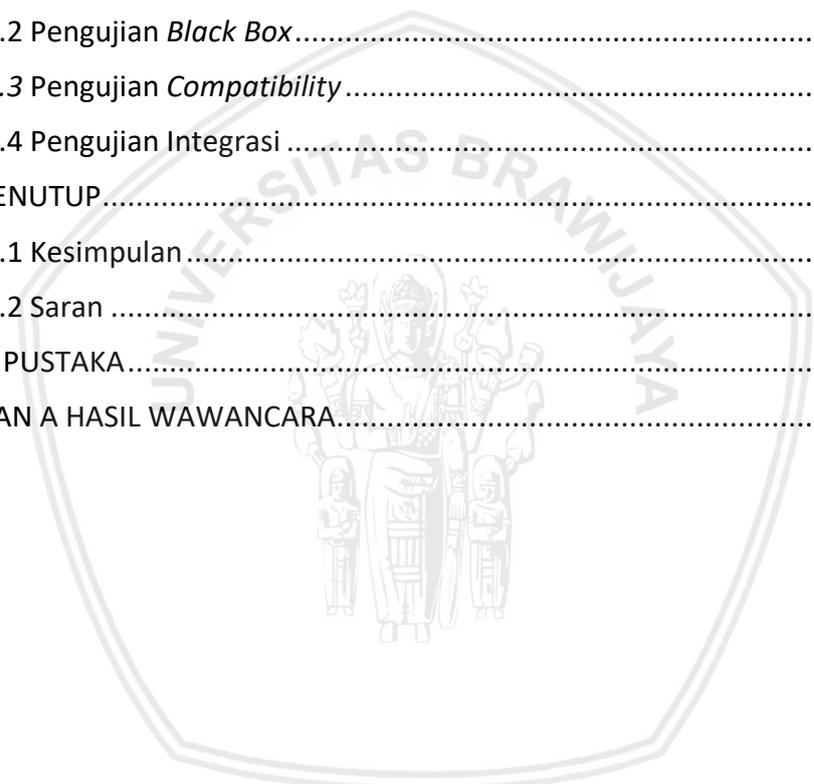
PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR	iv
ABSTRAK.....	v
ABSTRACT.....	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN	xv
BAB 1 PENDAHULUAN	1
1.1 Latar belakang	1
1.2 Rumusan masalah.....	2
1.3 Tujuan	2
1.4 Manfaat.....	2
1.5 Batasan masalah.....	3
1.6 Sistematika pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN	4
2.1 Kajian Pustaka.....	4
2.2 Kenaikan Pangkat dan Jabatan Fungsional Dosen	4
2.2.1 Angka Kredit.....	5
2.2.2 Bagian Kepegawaian.....	6
2.3 <i>Business Process Model and Notation</i>	6
2.4 <i>Prototyping</i>	7
2.5 <i>Unified Modeling Language (UML)</i>	7
2.5.1 <i>Usecase Diagram</i>	7
2.5.2 <i>Usecase Scenario Diagram</i>	9
2.5.3 <i>Sequence Diagram</i>	10
2.5.4 <i>Class Diagram</i>	10
2.6 <i>Webservice</i>	11
2.7 MVC.....	12



2.8 Framework CodeIgniter	12
BAB 3 METODOLOGI.....	13
3.1 Landasan Kepustakaan	13
3.2 Rekayasa Kebutuhan.....	14
3.3 Perancangan dan Implementasi Sistem.....	14
3.4 Pengujian.....	15
3.5 Kesimpulan Dan Saran	15
BAB 4 REKAYASA KEBUTUHAN.....	16
4.1 Elisitasi Kebutuhan.....	16
4.2 Analisis Kebutuhan Sistem	17
4.2.1 Iterasi 1	18
4.2.2 Iterasi 2	22
4.2.3 Identifikasi Aktor	22
4.2.4 Pendefinisian Kebutuhan.....	23
4.3 Use case Diagram.....	26
4.4 Use case Scenario Diagram	27
BAB 5 PERANCANGAN DAN IMPLEMENTASI	41
5.1 Perancangan Arsitektur Sistem	41
5.2 Perancangan Basis Data.....	41
5.2.1 Conceptual Data Model.....	42
5.2.2 Physical Data Model	42
5.3 Perancangan Sequence Diagram	43
5.3.1 Sequence diagram update status usulan.....	43
5.3.2 Sequence diagram lihat berkas BKD.....	43
5.3.3 Sequence diagram ajukan berkas BKD	44
5.4 Perancangan Class Diagram	44
5.5 Perancangan Algoritme Program	47
5.5.1 Algoritme update status usulan.....	47
5.5.2 Algoritme lihat berkas BKD.....	48
5.5.3 Algoritme Ajukan Berkas BKD	48
5.6 Perancangan Purwarupa	49
5.6.1 Purwarupa Login	49

5.6.2 Purwarupa Dashboard.....	49
5.6.3 Purwarupa Penambahan berkas.....	50
5.6.4 Purwarupa Berkas BKD.....	50
5.6.5 Purwarupa Data diri.....	51
5.6.6 Purwarupa Berkas usulan.....	51
5.6.7 Purwarupa Edit berkas usulan.....	52
5.6.8 Purwarupa Edit usulan.....	52
5.6.9 Purwarupa Riwayat usulan.....	53
5.6.10 Purwarupa Daftar dosen.....	53
5.6.11 Purwarupa daftar usulan dosen.....	54
5.6.12 Purwarupa Detil berkas usulan dosen.....	54
5.6.13 Purwarupa Penjelasan kesalahan.....	55
5.6.14 Purwarupa Update status usulan.....	55
5.6.15 Purwarupa <i>Edit</i> dupak.....	56
5.7 Implementasi Basis Data.....	56
5.8 Implementasi Kode Program.....	57
5.8.1 Kode program <i>update</i> status usulan.....	57
5.8.2 Kode program lihat berkas bkd.....	58
5.8.3 Kode program Ajukan Berkas BKD.....	58
5.9 Implementasi Antarmuka.....	59
5.9.1 Implementasi antarmuka Login.....	59
5.9.2 Implementasi antarmuka Dashboard.....	60
5.9.3 Implementasi Penambahan berkas.....	60
5.9.4 Implementasi antarmuka Berkas BKD.....	61
5.9.5 Implementasi antarmuka Data diri.....	61
5.9.6 Implementasi antarmuka Berkas usulan.....	62
5.9.7 Implementasi antarmuka Edit berkas usulan.....	62
5.9.8 Implementasi antarmuka Edit usulan.....	63
5.9.9 Implementasi antarmuka Riwayat usulan.....	63
5.9.10 Implementasi Daftar dosen.....	64
5.9.11 Implementasi antarmuka daftar usulan dosen.....	64
5.9.12 Implementasi antarmuka Detil berkas usulan dosen.....	65

5.9.13 Implementasi antarmuka Penjelasan kesalahan	65
5.9.14 Implementasi antarmuka Update status usulan.....	66
5.9.15 Implementasi antarmuka Edit dupak	66
BAB 6 PENGUJIAN.....	67
6.1 Pengujian <i>White Box</i>	67
6.1.1 Pengujian <i>white box update</i> status usulan.....	67
6.1.2 Pengujian <i>white box</i> lihat berkas bkd.....	69
6.1.3 Pengujian <i>white box</i> ajukan berkas BKD	72
6.2 Pengujian <i>Black Box</i>	73
6.3 Pengujian <i>Compatibility</i>	76
6.4 Pengujian Integrasi	77
BAB 7 PENUTUP.....	80
7.1 Kesimpulan	80
7.2 Saran	80
DAFTAR PUSTAKA.....	81
LAMPIRAN A HASIL WAWANCARA.....	83



DAFTAR TABEL

Tabel 2.1 Jenjang jabatan dan pangkat dosen.....	5
Tabel 2.2 Angka Kredit Kumulatif Perbidang	6
Tabel 2.3 Notasi Usecase Diagram	8
Tabel 2.4 Contoh Usecase Scenario Diagram	9
Tabel 2.5 Contoh Simbol Hubungan Antar Class	11
Tabel 3.1 Tabel definisi kode kebutuhan.....	14
Tabel 4.1 Inisialisasi kebutuhan Iterasi pertama	19
Tabel 4.2 Perubahan kebutuhan Iterasi kedua.....	22
Tabel 4.3 Tabel Identifikasi Aktor.....	23
Tabel 4.4 Tabel Kebutuhan Fungsional	23
Tabel 4.4 Tabel Kebutuhan Non Fungsional	26
Tabel 4.6 Tabel <i>use case scenario login</i>	27
Tabel 4.7 Tabel <i>use case scenario logout</i>	27
Tabel 4.8 Tabel <i>use case scenario</i> tambah usulan	28
Tabel 4.9 Tabel <i>use case scenario</i> edit usulan	28
Tabel 4.10 Tabel <i>use case scenario</i> hapus usulan.....	28
Tabel 4.11 Tabel <i>use case scenario</i> lihat usulan	29
Tabel 4.12 Tabel <i>use case scenario</i> cetak dupak	29
Tabel 4.13 Tabel <i>use case scenario</i> tambah berkas	30
Tabel 4.14 Tabel <i>use case scenario</i> hapus berkas administrasi.....	30
Tabel 4.15 Tabel <i>use case scenario</i> hapus berkas usulan.....	31
Tabel 4.16 Tabel <i>use case scenario</i> edit berkas usulan	31
Tabel 4.17 Tabel <i>use case scenario</i> lihat berkas usulan	32
Tabel 4.18 Tabel <i>use case scenario</i> lihat data diri.....	32
Tabel 4.19 Tabel <i>use case scenario</i> lihat berkas BKD	33
Tabel 4.20 Tabel <i>use case scenario</i> ajukan berkas BKD	33
Tabel 4.21 Tabel <i>use case scenario</i> lihat grafik AK.....	34
Tabel 4.22 Tabel <i>use case scenario</i> kirim pemberitahuan kenaikan pangkat dan jabatan	34
Tabel 4.23 Tabel <i>use case scenario</i> lihat daftar dosen.....	35

Tabel 4.24 Tabel <i>use case scenario</i> lihat usulan dosen	35
Tabel 4.25 Tabel <i>use case scenario</i> lihat detil dosen	35
Tabel 4.26 Tabel <i>use case scenario</i> validasi dan verifikasi usulan	36
Tabel 4.27 Tabel <i>use case scenario</i> validasi dan verifikasi berkas administrasi ...	36
Tabel 4.28 Tabel <i>use case scenario</i> batal validasi dan verifikasi berkas administrasi.....	37
Tabel 4.29 Tabel <i>use case scenario</i> validasi dan verifikasi berkas usulan.....	37
Tabel 4.30 Tabel <i>use case scenario</i> batal validasi dan verifikasi berkas usulan ...	38
Tabel 4.31 Tabel <i>use case scenario</i> kirim penjelasan kesalahan	38
Tabel 4.32 Tabel <i>use case scenario update</i> status usulan.....	39
Tabel 4.33 Tabel <i>use case scenario edit</i> dupak.....	39
Tabel 5.1 Tabel algoritme update status usulan.....	47
Tabel 5.2 Tabel algoritme lihat berkas bkd	48
Tabel 5.3 Tabel algoritme ajukan berkas BKD	48
Tabel 5.4 Tabel kode program update status usulan.....	57
Tabel 5.5 Tabel kode program lihat berkas bkd.....	58
Tabel 5.6 Tabel kode program ajukan berkas BKD	59
Tabel 6.1 Tabel <i>pseudocode white box</i> update status usulan	67
Tabel 6.2 Tabel kasus uji <i>white box</i> update status usulan	68
Tabel 6.3 Tabel <i>pseudocode white box</i> lihat berkas bkd.....	69
Tabel 6.4 Tabel kasus uji <i>white box</i> lihat berkas bkd	71
Tabel 6.5 Tabel <i>pseudocode white box</i> ajukan berkas BKD.....	72
Tabel 6.6 Tabel kasus uji <i>white box</i> ajukan berkas BKD	73
Tabel 6.7 Tabel pengujian <i>black box</i>	73
Tabel 6.8 Tabel <i>pseudocode</i> stubAjukanBerkasBKD()	77
Tabel 6.9 Tabel <i>pseudocode</i> pengujian fungsi stubAjukanBerkasBKD()	78
Tabel 6.10 Tabel <i>pseudocode</i> fungsi validasi_verifikasi() di kelas C_dosen.....	78
Tabel 6.11 Tabel <i>pseudocode</i> pengujian fungsi insert_berkas() di kelas M_berkas_usulan.....	79

DAFTAR GAMBAR

Gambar 2.1 Teknik Pemodelan Prototyping.....	7
Gambar 2.2 Contoh Sequence Diagram	10
Gambar 2.3 Contoh alur kerja RESTful Webservice.....	11
Gambar 3.1 Metodologi Penelitian.....	13
Gambar 4.1 Proses bisnis <i>As-is</i>	16
Gambar 4.2 Proses bisnis <i>To-be</i>	18
Gambar 4.3 Perancangan tampilan Tambah Berkas Sebelum	21
Gambar 4.4 Perancangan tampilan Tambah Berkas Sesudah.....	22
Gambar 4.5 <i>Use case</i> diagram	26
Gambar 5.1 Arsitektur sistem.....	41
Gambar 5.2 Conceptual Data Model.....	42
Gambar 5.3 Physical Data Model	42
Gambar 5.4 <i>Sequence Diagram update</i> status usulan.....	43
Gambar 5.5 <i>Sequence Diagram</i> lihat berkas BKD.....	44
Gambar 5.6 <i>Sequence Diagram</i> ajukan berkas BKD	44
Gambar 5.7 Perancangan Class Diagram.....	45
Gambar 5.8 Perancangan Implementasi <i>Class Diagram</i>	46
Gambar 5.9 Detil perancangan implementasi <i>Class Diagram</i>	47
Gambar 5.10 Purwarupa login	49
Gambar 5.11 Purwarupa dashboard	50
Gambar 5.12 Purwarupa penambahan berkas	50
Gambar 5.13 Purwarupa berkas BKD	51
Gambar 5.14 Purwarupa data diri.....	51
Gambar 5.15 Purwarupa berkas usulan	52
Gambar 5.16 Purwarupa Edit berkas usulan	52
Gambar 5.17 Purwarupa edit usulan.....	53
Gambar 5.18 Purwarupa riwayat usulan.....	53
Gambar 5.19 Purwarupa daftar dosen.....	54
Gambar 5.20 Purwarupa daftar usulan dosen.....	54
Gambar 5.21 Purwarupa detil berkas usulan dosen	55

Gambar 5.22 Purwarupa penjelasan kesalahan	55
Gambar 5.23 Purwarupa update status usulan	56
Gambar 5.24 Purwarupa edit dupak	56
Gambar 5.25 implementasi basis data	57
Gambar 5.26 Implementasi antarmuka login	59
Gambar 5.27 Implementasi antarmuka dashboard	60
Gambar 5.28 Implementasi antarmuka penambahan berkas	60
Gambar 5.29 Implementasi antarmuka berkas BKD	61
Gambar 5.30 Implementasi antarmuka data diri	61
Gambar 5.31 Implementasi antarmuka berkas usulan	62
Gambar 5.32 Implementasi antarmuka Edit berkas usulan	62
Gambar 5.33 Implementasi antarmuka edit usulan	63
Gambar 5.34 Implementasi antarmuka riwayat usulan	63
Gambar 5.35 Implementasi antarmuka daftar dosen	64
Gambar 5.36 Implementasi antarmuka daftar usulan dosen	64
Gambar 5.37 Implementasi antarmuka detil berkas usulan dosen	65
Gambar 5.38 Implementasi antarmuka penjelasan kesalahan	65
Gambar 5.39 Implementasi antarmuka update status usulan	66
Gambar 5.40 Implementasi antarmuka edit dupak	66
Gambar 6.1 <i>Flowgraph white box</i> update status usulan	68
Gambar 6.2 <i>Flowgraph white box</i> lihat berkas bkd	70
Gambar 6.3 <i>Flowgraph white box</i> ajukan berkas BKD	72
Gambar 6.4 Hasil Pengujian <i>Compatibility</i>	77
Gambar 6.5 Diagram hierarki pengujian integrasi	77
Gambar 6.6 Hasil Stub stubAjukanBerkasBKD()	78
Gambar 6.7 Hasil dari luaran integrasi fungsi insert_berkas dari M_berkas_usulan dan fungsi validasi_berifikasi_dari C_dosen	79

DAFTAR LAMPIRAN

LAMPIRAN A HASIL WAWANCARA.....	83
A.1 Hasil Wawancara Beberapa Dosen FILKOM	83
A.2 Hasil Wawancara Staf Kepegawaian FILKOM	85
A.3 Iterasi Perubahan Antar Muka	85



BAB 1 PENDAHULUAN

1.1 Latar belakang

Kenaikan jabatan fungsional atau jabatan akademik merupakan bentuk pemberian penghargaan pemerintah atas prestasi kerja yang dicapai setiap dosen yang telah mempunyai prestasi kerja sesuai dengan peraturan perundangan berhak mendapatkan penghargaan kenaikan jabatan fungsional (Kemenristekdikti, 2016). Komponen penilaian jabatan fungsional terdiri dari beberapa unsur meliputi Pendidikan, Penelitian, Pengabdian kepada masyarakat dan Penunjang yang selanjutnya dituliskan dengan 4 bidang. Setiap unsur 4 bidang memiliki jumlah penilaian dari setiap butir komponen kegiatan yang dilakukan dosen, disebut juga dengan angka kredit (AK). Untuk dapat menduduki jenjang jabatan akademik dan/atau pangkat tertentu, dosen wajib memenuhi AK dengan distribusi unsur utama dan penunjang tertentu (Kemenristekdikti, 2016).

Di Fakultas Ilmu Komputer Universitas Brawijaya selanjutnya dituliskan dengan FILKOM memiliki bagian yang mengurus prosedur kenaikan pangkat dan jabatan fungsional yaitu bagian kepegawaian FILKOM. Dosen FILKOM dapat melakukan kenaikan pangkat dan/atau jabatan fungsional melalui prosedur yang dimiliki oleh bagian kepegawaian. Namun dari data wawancara pada beberapa dosen FILKOM (Lampiran A.1) bahwa dosen tersebut memiliki banyak aktivitas. Lalu pada saat proses validasi dan verifikasi berkas di Kepagawaian FILKOM usulan sering terjadi kesalahan sehingga tidak memiliki waktu untuk memproses revisi lebih lanjut dalam mengumpulkan berkas 4 bidang dan berkas administrasi untuk usulan kenaikan pangkat dan/atau jabatan fungsional. Ditambah lagi dengan dosen masih melakukan penghitungan AK secara manual dengan melihat panduan sehingga memungkinkan kesalahan penghitungan bahkan fatalnya bisa salah dalam penghitungan AK pada setiap kegiatannya.

Pada penelitian sebelumnya telah memberikan solusi tentang kenaikan jabatan fungsional dosen dilingkungan Universitas Brawijaya (UB). Penelitian tersebut memberikan solusi penghitungan AK yang lebih cepat, memberikan fitur notifikasi kenaikan jabatan dosen dan memberikan fitur memantau status untuk mengetahui proses pengajuan kenaikan jabatan fungsional. Namun pada penelitian tersebut dosen harus melakukan *input* berkas secara mandiri yang pada kenyataannya masih banyak dosen yang sibuk untuk melakukan pengusulan kenaikan pangkat dan/atau jabatan secara mandiri. Dari masalah di atas, penulis memiliki sebuah alternatif solusi dengan melakukan pengembangan suatu aplikasi sistem yang bertujuan untuk membantu proses pengusulan kenaikan pangkat dan/atau jabatan dosen FILKOM.

Pada penelitian ini menerapkan sistem berbasis *webservice* dalam pengambilan data BKD maupun data staf kepegawaian/dosen UB. *Webservice* digunakan sebagai jembatan sistem yang sudah ada dengan sistem yang lain tanpa mempengaruhi sistem utama. Sumber data model *webservice* yang digunakan pada sistem ini adalah data *dummy* staf pada Sistem Informasi Dosen

(SIADO) dan data *dummy* Sistem Informasi Pegawai (SIMPEG). Pada proses pengusulan kenaikan pangkat dan/atau jabatan, dosen harus melakukan pengumpulan berkas 4 bidang secara mandiri atau data *dummy* berkas kinerja dosen (BKD). Selanjutnya berkas 4 bidang tersebut diteruskan ke kepegawaian untuk divalidasi dan diverifikasi. Setelah divalidasi dan diverifikasi dosen akan menerima pemberitahuan *email* dan bukti lampiran DUPAK (Daftar Pengusul Penetapan Angka Kredit) yang berarti bahwa dosen bersangkutan dapat mengumpulkan berkas-berkas 4 bidang yang terlampir pada DUPAK. Selanjutnya kepegawaian melanjutkan usulan ke proses berikutnya yang nantinya dapat dipantau oleh dosen yang bersangkutan.

Untuk mewujudkan solusi yang tepat guna yang diajukan oleh penulis diperlukan hasil analisis kebutuhan yang sesuai dengan kebutuhan pengguna, perancangan yang sesuai dengan analisis, hasil implementasi yang sesuai dengan hasil perancangan dan dilakukan pengujian untuk memastikan sistem yang dibuat sesuai dengan kebutuhan pengguna.

1.2 Rumusan masalah

1. Bagaimana analisis kebutuhan yang digunakan untuk Sistem kenaikan pangkat dan jabatan dosen Fakultas Ilmu Komputer?
2. Bagaimana rancangan yang digunakan untuk Sistem kenaikan pangkat dan jabatan dosen Fakultas Ilmu Komputer?
3. Bagaimana implementasi dari Sistem kenaikan pangkat dan jabatan dosen Fakultas Ilmu Komputer?
4. Bagaimana pengujian dari Sistem kenaikan pangkat dan jabatan dosen Fakultas Ilmu Komputer?

1.3 Tujuan

1. Menganalisis kebutuhan Sistem kenaikan pangkat dan jabatan dosen Fakultas Ilmu Komputer
2. Merancang Sistem kenaikan pangkat dan jabatan dosen Fakultas Ilmu Komputer dengan hasil analisis kebutuhan.
3. Membuat Sistem kenaikan pangkat dan jabatan dosen Fakultas Ilmu Komputer.
4. Menguji Sistem kenaikan pangkat dan jabatan dosen Fakultas Ilmu Komputer.

1.4 Manfaat

Manfaat penelitian ini adalah ketika dosen yang memiliki kesibukan dapat terbantu dalam mengumpulkan berkas 4 bidang, memantau usulan melalui sistem, membantu validasi dan verifikasi usulan atau revisi usulan melalui sistem, dan meminimalisir kesalahan pada penghitungan AK, Sehingga meningkatkan efektifitas dan efisiensi prosedur pengusulan kenaikan pangkat dan jabatan dosen.

1.5 Batasan masalah

1. Ruang lingkup hanya pada proses kenaikan pangkat dan jabatan fungsional di lingkup kepegawaian FILKOM.
2. Pedoman penilaian AK menggunakan versi tahun 2014.

1.6 Sistematika pembahasan

Penyusunan perancangan sistematika penulisan sebagai berikut:

BAB 1 PENDAHULUAN

Pada bab 1 merumuskan latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika pembahasan.

BAB 2 LANDASAN KEPUSTAKAAN

Pada bab 2 menguraikan tentang dasar teori dan referensi yang mendasari pengembangan sistem kenaikan pangkat dan jabatan dosen FILKOM.

BAB 3 METODOLOGI PENELITIAN

Bab ini berisi penjelasan tentang rencana penelitian yaitu landasan kepastakaan, pemodelan proses bisnis, analisis kebutuhan, perancangan sistem, implementasi, pengujian dan kesimpulan.

BAB 4 REKAYASA KEBUTUHAN

Bab ini menganalisis kebutuhan terdiri dari analisis kebutuhan, iterasi, pendefinisian kebutuhan, pendefinisian aktor dan proses bisnis.

BAB 5 PERANCANGAN DAN IMPLEMENTASI

Bab ini menjelaskan tentang perancangan arsitektur sistem, perancangan basis data, perancangan *class diagram*, perancangan implemnetasi *class diagram*, perancangan *sequence diagram*, perancangan algoritme program, perancangan purwarupa, implementasi basis data, implemetasi kode program, implementasi antarmuka.

BAB 6 PENGUJIAN

Bab ini berisi hasil pengujian unit, pengujian integrase dan pengujian *compatibility browser*. bertujuan untuk memastikan fitur-fitur dalam aplikasi berjalan sesuai dengan luaran yang diharapkan.

BAB 7 PENUTUP

Bab ini menjelaskan tentang kesimpulan dan saran yang diperoleh dari kesesuaian analisis kebutuhan, perancangan sistem, implentasi, pengujian dengan rumusan masalah penelitian yang nantinya menjadi acuan bagi penelitian selanjutnya.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Pada penelitian ini peneliti mengambil beberapa penelitian terkait sebagai tinjauan sistem yang akan dikembangkan pada penelitian ini.

Penelitian pertama dilakukan oleh Muhammad Saifussolihin Putra Aditama Dengan judul penelitian “Pembangunan Sistem Informasi kenaikan jabatan fungsional Dosen Universitas Brawijaya”. Penelitian ini bertujuan mengatasi proses bisnis sistem informasi kenaikan jabatan fungsional dosen dalam mengatasi permasalahan lamanya penghitungan angka kredit dan kurangnya inisiatif dosen. Sehingga di dalam Sistem informasi ini dapat mengurangi proses penghitungan secara mandiri angka kredit dengan 5-10 menit pengerjaan dan terdapat notifikasi untuk menginformasikan dosen telah siap untuk naik jabatan jika memenuhi TMT (Tanggal Mulai Tugas) 1.5 tahun dari TMT terakhir. Pada sistem informasi ini memiliki alur dosen memasukan berkas 4 bidang lalu akan divalidasi pada pihak kepegawaian fakultas dan universitas. Dosen akan menerima pemberitahuan berkas akan status berkas yang telah divalidasi oleh bagian kepegawaian (Aditama, Agung Wicaksono and Pradana, 2018).

Penelitian kedua tentang pembangunan sistem informasi pengajuan kegiatan dan pemberian poin bagi FILKOM UB. Penelitian ini bertujuan untuk membantu mahasiswa FILKOM dalam mengajukan kegiatan untuk dijadikan sebuah poin yang dinamakan dengan Satuan Kredit Mahasiswa (SKM) (Rasyid, Pradana and Rusdianto, 2018).

Penelitian ketiga tentang pembangunan sistem informasi manajemen perizinan di Dinas KOMINFO (Kementrian Komunikasi dan Informatika) kota Pasuruan. Penelitian ini bertujuan untuk mengatasi kendala permohonan pengurusan izin tidak lengkap, sering tertukar dengan perizinan lain dan pengurusan izin yang lama. (Evi et al., 2019). Pada penelitian ini didapatkan contoh proses bisnis dalam penerapan pengusulan perizinan.

Dari penelitian ketiganya ditarik kesimpulan diharapkan sistem ini diharapkan dapat meningkatkan efektifitas dan efisiensi proses pengumpulan berkas dan dapat membantu dosen dalam segi pengumpulan dengan adanya *webservice dummy* BKD, *email* pemberitahuan untuk memantau status usulan, validasi dan verifikasi berkas 4 bidang pada sistem untuk dosen yang memiliki aktivitas di luar kampus, dan melakukan perhitungan otomatis berdasarkan pedoman AK untuk membantu penghitungan kegiatan.

2.2 Kenaikan Pangkat dan Jabatan Fungsional Dosen

Jabatan fungsional Dosen adalah kedudukan yang menunjukkan tugas, tanggung jawab, wewenang dan hak seseorang Dosen dalam suatu satuan Pendidikan Tinggi yang dalam pelaksanaannya didasarkan pada keahlian tertentu serta bersifat mandiri dan disesuaikan dengan angka kredit. AK adalah satuan

nilai dari butir kegiatan dan/atau akumulasi nilai butir-butir kegiatan yang harus dicapai oleh seorang dosen dalam rangka pembinaan karier jabatan dan kepangkatan. Sementara itu pangkat adalah kedudukan yang menunjukkan tingkat seseorang pegawai berdasarkan jabatannya dalam rangkaian susunan kepegawaian dan digunakan sebagai dasar penggajian (Rahman, 2017). Tabel 2.1 menunjukkan akumulasi jabatan fungsional AK dan pangkat atau golongan berdasarkan pemaparan presentasi bapak Taufan Sudrajat, S.E., M.Si. mengenai sosialisasi pengusulan jabatan fungsional dosen (Sudrajat, 2016). Untuk waktu kenaikan jabatan adalah minimal 2 tahun dari TMT jabatan terakhir. Untuk waktu pengumpulan berkas antara usulan kenaikan jabatan dan pangkat memiliki perbedaan waktu, jika usulan kenaikan jabatan tidak diberikan tenggat waktu pengumpulan berkas hanya minimal 2 tahun TMT, namun untuk usulan kenaikan pangkat harus dikumpulkan pada bulan Januari untuk TMT bulan April atau pada bulan Juli untuk TMT bulan Oktober. Hal tersebut dipaparkan oleh bidang kepegawaian FILKOM dalam wawancara mengenai kenaikan jabatan dan pangkat dosen (Lampiran A.2).

Tabel 2.1 Jenjang jabatan dan pangkat dosen

No	Jenjang jabatan	Jenjang pangkat / Golongan ruang	Angka kredit dipersyaratkan minimal
1	Asisten Ahli	Penata Muda Tk I, III/b	150
2	Lektor	Penata, III/c	200
		Penata Tk. I, III/d	300
3	Lektor Kepala	Pembina, IV/a	400
		Pembina Tk. I, IV/b	550
		Pembina Utama Muda, IV/c	700
4	Guru besar/Professor	Pembina Utama Madya, IV/d	850
		Pembina Utama, IV/e	1050

(Rahman, 2017)

2.2.1 Angka Kredit

Angka kredit adalah nilai dari tiap butir kegiatan tri dharma dosen yang dapat digunakan untuk kenaikan pangkat dan jabatan. Angka kredit yang dipersyaratkan untuk kenaikan jabatan harus memenuhi kredit kumulatif perbidangnya. Unsur utama yang harus dipenuhi dosen adalah unsur Pendidikan, penelitian, dan pengabdian yang harus memenuhi minimal 90% dari unsur utama, sedangkan unsur penunjang sekitar 10% maksimal (Sudrajat, 2016). Tiap bidang memiliki peraturan persentase minimal dan maksimal angka kredit yang dikumpulkan yang direpresentasikan Tabel 2.2.

Tabel 2.2 Angka Kredit Kumulatif Perbidang

No	Jenjang jabatan	Kualifikasi pendidikan	Unsur Utama 90%			Penunjang 10%
			Pendidikan	Penelitian	Pengabdian	
1	Asisten Ahli	Magister	≥ 55%	≥ 25%	≤ 10%	≤ 10%
2	Lektor	Magister	≥ 45%	≥ 35%	≤ 10%	≤ 10%
3	Lektor Kepala	Doktor atau Magister	≥ 40%	≥ 40%	≤ 10%	≤ 10%
4	Guru besar/Professor	Doktor	≥ 35%	≥ 45%	≤ 10%	≤ 10%

(Rahman, 2017)

2.2.2 Bagian Kepegawaian

Bagian kepegawaian adalah bagian atau unit yang menangani urusan staf dan dosen. Bagian kepegawaian khususnya di FILKOM menangani urusan dosen salah satunya kenaikan jabatan fungsional dosen. Berdasarkan SOP (*Standard Operational Procedure*) kenaikan jabatan fungsional dosen (Kemenristekdikti, 2018). Proses yang dilakukan administrasi kepegawaian dalam rangka kenaikan jabatan fungsional dosen adalah memeriksa, menghitung angka kredit, membuat surat pengantar kenaikan jabatan dan memberikan luaran berupa DUPAK.

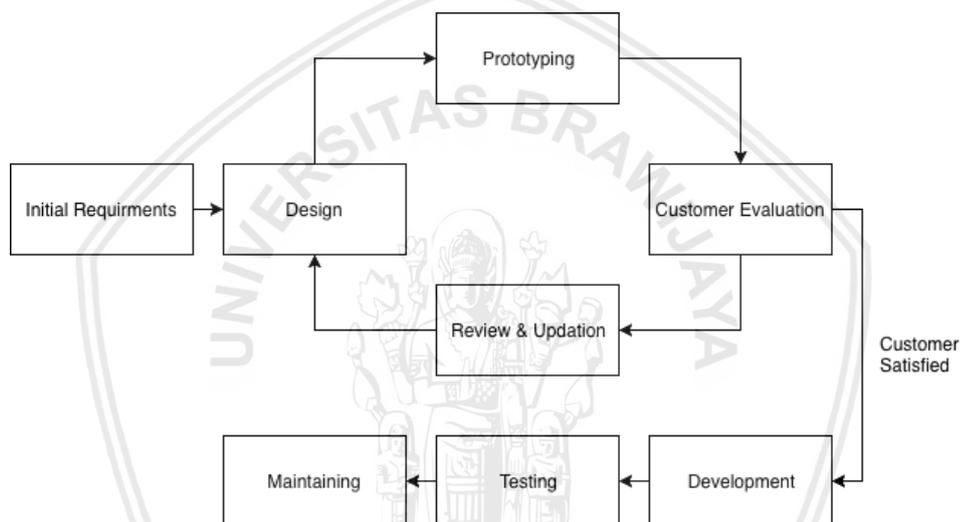
2.3 Business Process Model and Notation

Business Process Model and Notation atau lebih dikenal dengan BPMN adalah sebuah notasi grafis yang menggambarkan alur kerja. BPMN biasa digunakan untuk menggambarkan proses kerja suatu organisasi atau institusi. BPMN dapat menjembatani komunikasi antara desain proses bisnis dan implementasi. BPMN memiliki 3 tahap untuk pendekatan mengatasi masalah (Aditama, Agung Wicaksono and Pradana, 2018).

1. *Bisnis Proses As-is*
Tahap ini bertujuan untuk menggambarkan proses kerja atau alur dari suatu organisasi atau institusi dalam menjalankan peran yang dilakukan.
2. *Bisnis Proses To-be*
Tahap ini bertujuan untuk menggambarkan proses kerja atau alur dari suatu pemodelan sistem. Pemodelan sistem ini dibuat berdasarkan kondisi dari beberapa pihak atau pemangku kepentingan. Proses bisnis ini mereferensi dari proses bisnis *As-is*.
3. *Pendefinisian kebutuhan*
Setelah menganalisis proses bisnis yang ada maka selanjutnya mengidentifikasi masalah berdasarkan proses bisnis *to-be*. Oleh karena itu proses bisnis dapat menjadi acuan dalam mengidentifikasi kebutuhan.

2.4 Prototyping

Prototyping adalah suatu teknik pemodelan *Software development Life Cycle* (SDLC) yang lebih menekankan pada kepuasan pengguna dari sisi desain *prototype*. *Prototyping* sendiri memiliki tahap pengembangan diantaranya ada tahap iterasi desain dan review seperti digambarkan pada Gambar 2.1. *Prototyping* ini memiliki keunggulan pada sisi evaluasi desain pengguna sehingga *software* akan diimplementasi sampai memenuhi kriteria pengguna (Pressman, 2010). Kekurangan yang mungkin terjadi pada pengembangan model *prototyping* adalah dengan terburu-buru dalam pembuatan fungsi akan berdampak pada pemeliharaan sistem dan algoritma yang digunakan belum tentu efisien karena menyesuaikan dengan keinginan desain pengguna (Pressman, 2010).



Gambar 2.1 Teknik Pemodelan Prototyping

(Pressman, 2010)

2.5 Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah adalah suatu teknologi pemandu pemodelan proyek perangkat lunak yang berbasis *object-oriented* (Pressman, 2010). UML ini digunakan ketika pengembangan suatu sistem menggunakan paradigma *object-oriented* khususnya dalam tahap *object-oriented design* untuk pemodelannya. UML menekankan pada *behavior* aktor pada sistem. UML memiliki macam-macam diagram yang digunakan sesuai tahap-tahap pengembangan perangkat lunak, diantaranya diagram *usecase*, diagram *usecase scenario*, diagram *sequence*, dan diagram *class* (Larman, 2005) (Pressman, 2010) (Kurniawan, 2018).

2.5.1 Usecase Diagram

Usecase diagram adalah diagram yang menggambarkan pengguna atau aktor dalam menggunakan sistem. *Usecase* diagram termasuk dalam tahap

object-oriented design pada tahap pengembangan perangkat lunak dengan paradigma *object-oriented*. *Usecase* diagram memiliki beberapa notasi dalam merepresentasikan suatu sistem. Tabel 2.3 adalah tabel yang menggambarkan notasi dari *usecase* diagram.

1. *Usecase*

Usecase adalah sebuah deskripsi fungsi yang akan dijalankan pada sistem. Sebuah *usecase* harus menggunakan representasi kata kerja dalam penulisan pada teks deskripsinya di awal. *Usecase* memiliki 3 representasi notasi salah satunya berbentuk elips (Kurniawan, 2018).

2. Aktor

Aktor adalah sebuah entitas yang berinteraksi dengan *usecase*. Aktor disebut juga entitas luar sistem. Setiap aktor dapat berinteraksi dengan banyak *usecase* (Aditama, Agung Wicaksono and Pradana, 2018). Contoh dari aktor adalah server diluar sistem, pengguna/*stakeholder*, dan sistem lain.

3. *Association*

Association adalah hubungan antara aktor dengan *usecase*. *Association* digambarkan dengan garis lurus (Aditama, Agung Wicaksono and Pradana, 2018).

4. *Generalization*

Generalization adalah hubungan umum-khusus atau spesialisasi dari fungsi yang spesifik ditarik ke lebih umum atau *general*. Contohnya adalah fungsi aktor saat sebelum *login* dan aktor setelah *login* ialah sama sebagai *user* atau pengguna (Kurniawan, 2018).

5. *Extend*

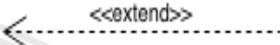
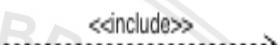
Extend adalah notasi relasi dari *usecase* yang hampir mirip dengan prinsip *inheritance* pada pemrograman berorientasi objek (Kurniawan, 2018). Fungsinya adalah setiap *usecase* memiliki fungsi yang hampir sama namun ada perbedaan yang membedakan *usecase* tersebut sehingga perbedaan itu ditarik menjadi sebuah *usecase* baru yang akan di *extend*.

6. *Include*

Include adalah notasi relasi yang memiliki fungsi syarat untuk *usecase* lain yang terhubung. *Include* digunakan ketika *usecase* yang digunakan ketika fungsi hampir sama dengan *usecase* yang lain sehingga ditarik menjadi *usecase* baru (Kurniawan, 2018).

Tabel 2.3 Notasi Usecase Diagram

No	Nama Notasi	Gambar Notasi
1	<i>Usecase</i>	

2	Aktor	
3	Association	
4	Generalization	
5	Exetnd	
6	Include	

(Pressman, 2010)

2.5.2 Usecase Scenario Diagram

Usecase scenario diagram adalah penjelasan detail secara deskriptif dari *usecase* diagram (Kurniawan, 2018). *Usecase scenario* diagram memiliki peran yang sangat penting karena berkaitan dengan *usecase* tertentu. *Usecase scenario* diagram setidaknya mendeskripsikan tujuan, aktor, identifikasi prakondisi, penjelasan aliran utama, aliran alternatif dan kondisi akhir. Untuk mendeskripsikan tujuan maka harus spesifik sesuai dengan luaran yang diinginkan. Untuk mendeskripsikan aktor maka harus sesuai konteks dan harus konsisten penggunaannya. Untuk mendeskripsikan identifikasi prakondisi dan kondisi akhir menggunakan secara spesifik dan kalimat pasif, misal aktor sudah teridentifikasi sebagai *teller*. Untuk mendeskripsikan aliran utama atau aliran alternatif maka harus dipisah antara pendeskripsian aliran utama dengan aliran alternatif, misal aliran utama “Sistem melakukan validasi dan menyimpan data” dan aliran alternatifnya “jika validasi salah maka sistem akan menampilkan halaman utama” (Kurniawan, 2018). Tabel 2.4 adalah contoh penggunaan *usecase scenario* diagram.

Tabel 2.4 Contoh Usecase Scenario Diagram

Kode	SRS_01_01
Tujuan	Aktor mengisi form data pribadi secara lengkap.
Aktor	Mahasiswa.
Prakondisi	Aktor telah <i>login</i> dan berada pada halaman akademik.
Aliran utama	1. Aktor mengisi data form pribadi secara lengkap. 2. Sistem melakukan validasi data form pribadi dan menyimpan

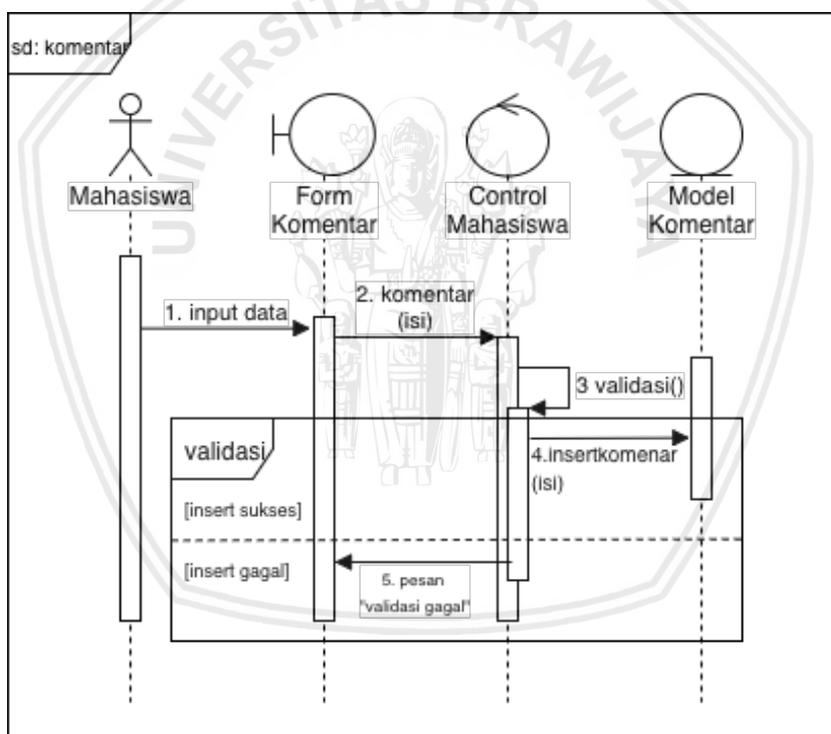


	data form pribadi. 3. Sistem menampilkan pesan “data berhasil disimpan”.
Aliran alternatif	Jika validasi salah, maka sistem menampilkan pesan kesalahan “data tidak valid”.
Kondisi akhir	Data pribadi aktor sudah diperbarui.

(Larman, 2005)

2.5.3 Sequence Diagram

Sequence diagram adalah mendeskripsikan gambaran interaksi antara objek pada sebuah *software* (Larman, 2005). Sequence diagram memiliki dua dimensi, yaitu dimensi vertikal yang mewakili waktu serta dimensi *horizontal* yang mewakili objek yang berbeda. Inisiasi urutan dimulai disudut kiri atas, dan waktu dari atas ke bawah. Garis vertikal disebut garis hidup objek (*lifeline*). Gambar 2.2 menunjukkan contoh dari *sequence diagram*.



Gambar 2.2 Contoh Sequence Diagram

(Larman, 2005)

2.5.4 Class Diagram

Class diagram adalah bagian dari perancangan perangkat lunak dengan paradigma *object-oriented*. Class diagram memiliki peranan dalam perancangan interaksi antar objek klas. Class diagram memiliki beberapa notasi diantaranya asosiasi dan *dependency*. Setiap *class* memiliki atribut, operasi dan *access-*

modifier (Pressman, 2010). Gambar 2.5 adalah contoh simbol-simbol dari hubungan antar *class* pada *class* diagram.

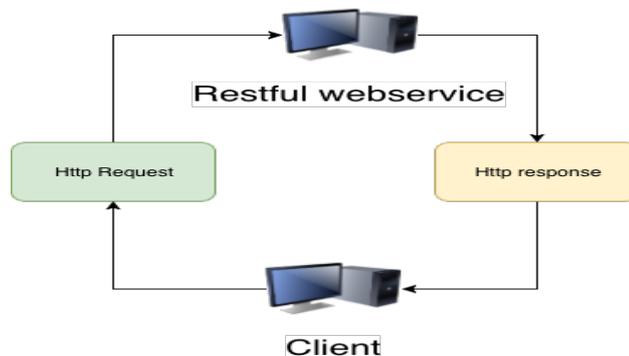
Tabel 2.5 Contoh Simbol Hubungan Antar Class

Asosiasi		hubungan statis antar class. Umumnya menggambarkan class yang memiliki atribut berupa class lain, atau class yang harus mengetahui eksistensi class lain.
Komposisi		merupakan bentuk khusus dari Agregasi di mana kelas yang menjadi <i>part</i> (bagian) baru dapat diciptakan setelah kelas yang menjadi <i>whole</i> (seluruhnya) dibuat dan ketika kelas yang menjadi <i>whole</i> dimusnahkan, maka kelas yang menjadi <i>part</i> ikut musnah.
Pewarisan		yaitu hubungan diwarisi klas lain yang mewarisi semua atribut <i>superclass</i> .
Agregasi		hubungan yang menyatakan suatu klas adalah bagian (“terdiri atas kumpulan”) dari klas yg lain, namun kedua klas ini bisa berdiri sendiri. Merupakan hubungan yang lebih kuat dari hubungan asosiasi.

(Larman, 2005)

2.6 Webservice

Webservice adalah standar dalam pertukaran data untuk aplikasi atau *platform* yang berbeda. Implementasi dari *webservice* ada 2 yaitu salah satunya *REST (Representational State Transfer)*. *REST* pada web service disebut juga dengan *RESTful Webservice*. Cara kerja *RESTful Webservice* ini dengan mengirimkan sebuah data *request* dari *client* melalui *protocol* HTTP. *Request client* yang digunakan diantaranya *GET, POST, DELETE, dan PUT* (Masse, 2011). Gambar 2.3. mendeskripsikan alur kerja dari *RESTful Webservice*.



Gambar 2.3 Contoh alur kerja RESTful Webservice

(Masse, 2011)

2.7 MVC

MVC (*Model, View, Controller*) merupakan suatu konsep arsitektur pengembangan perangkat lunak yang memisahkan bisnis logika, data logika dan presentasi logika. MVC memisahkan antara desain, data dan proses. MVC adalah konsep dasar yang harus diketahui sebelum mengenal *Framework CodeIgniter* (Pitt, 2012).

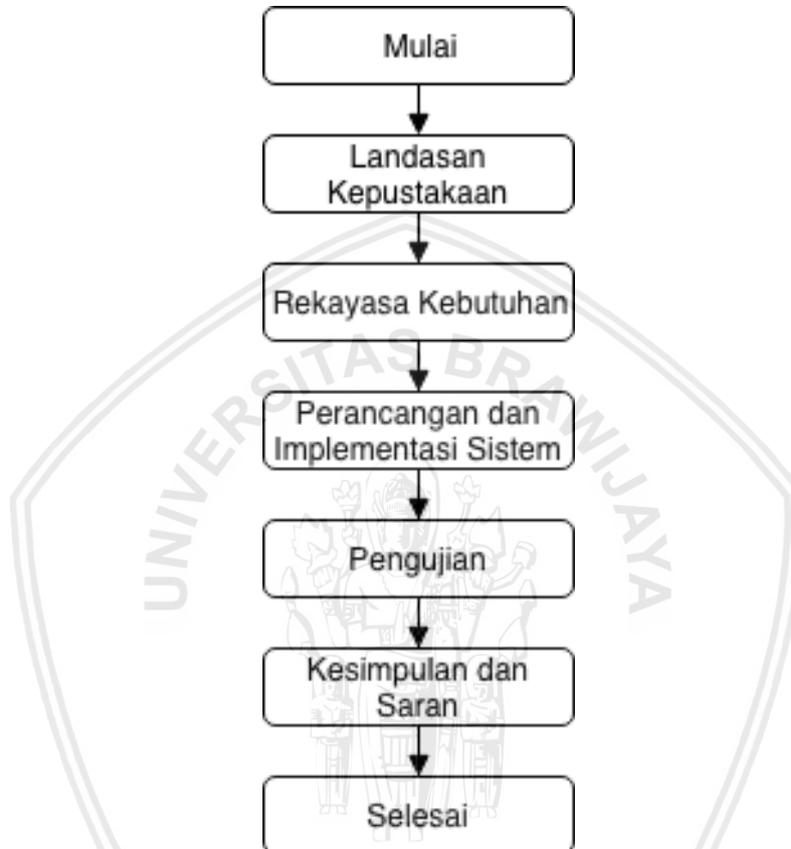
2.8 Framework CodeIgniter

Framework CodeIgniter adalah kerangka pengembangan aplikasi untuk pengembang yang membangun situs web menggunakan pemrograman *PHP*. Tujuannya adalah untuk memungkinkan Anda mengembangkan proyek lebih cepat daripada menulis kode dari awal, dengan menyediakan kumpulan pustaka yang kaya untuk tugas-tugas yang biasa dibutuhkan, serta antarmuka yang sederhana dan struktur logis untuk mengakses pustaka yang dibutuhkan. *CodeIgniter* memungkinkan pengembang secara kreatif fokus pada proyek dengan meminimalkan jumlah kode yang diperlukan untuk tugas yang diberikan (CodeIgniter, 2018).



BAB 3 METODOLOGI

Tahap ini membahas tentang tahapan pengerjaan penelitian. Penelitian dimulai dengan melakukan studi kepustakaan, pengumpulan data, pemodelan proses bisnis, analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan kesimpulan dan saran yang digambarkan pada gambar 3.1.



Gambar 3.1 Metodologi Penelitian

3.1 Landasan Kepustakaan

Landasan kepustakaan menjelaskan tentang acuan pembuatan penelitian skripsi ini. Landasan kepustakaan diambil dari berbagai media referensi diantaranya dokumen peraturan menteri, artikel jurnal, skripsi, buku dan dokumen presentasi. Teori yang dibahas antara lain:

1. Kenaikan pangkat dan jabatan fungsional dosen.
2. *Business process model and notation*.
3. Model pengembangan *prototyping*.
4. UML (*Unified Modeling Language*).
 - a. *Usecase diagram*.
 - b. *Usecase scenario diagram*.

- c. *Sequence* diagram.
 - d. *Class* Diagram.
5. *Webservice*
 6. *MVC (Model-View-Controller) Architecture*
 7. *Framework Code Igniter*

3.2 Rekayasa Kebutuhan

Proses rekayasa kebutuhan memiliki 2 tahap yaitu elisitasi kebutuhan dan analisis kebutuhan.

Elisitasi kebutuhan peniitian ini dilakukan dengan cara wawancara pada staf kepegawaian dan beberapa dosen FILKOM yang pernah melakukan usulan kenaikan pangkat atau jabatan. Pada tahap ini juga dilakukan pengumpulan dokumen-dokumen pendukung seperti SOP kenaikan jabatan dosen, dan pedoman penilaian AK kegiatan dosen. Elisitasi ini bertujuan untuk mendapatkan informasi keluhan dosen dan proses bisnis dari kenaikan pangkat dan jabatan.

Analisis kebutuhan adalah tahap perumusan permasalahan yang telah dibahas pada elisitasi kebutuhan. Analisis kebutuhan bertujuan untuk memetakan proses bisnis dan merancang sebuah solusi dari permasalahan yang ada pada tahap elisitasi. Pendefinisian kode kebutuhan sistem yaitu dengan kode acuan SPJ_XX_YY dijelaskan pada tabel 3.1.

Tabel 3.1 Tabel definisi kode kebutuhan

Kode	Deskripsi
SPJ	Merupakan identifikasi nama sistem.
XX	Merupakan sebuah angka mendefinisikan kebutuhan fungsional untuk 01 dan 02 untuk non-fungsional.
YY	Merupakan sebuah nomor tiap kebutuhan.

3.3 Perancangan dan Implementasi Sistem

Perancangan disini tujuannya adalah merancang kebutuhan perangkat lunak seperti perancangan arsitektur sistem, perancangan basis data, perancangan *class* diagram, perancangan *sequence* diagram, perancangan algoritme program, dan perancangan purwarupa. Perancangan sistem menggunakan paradigma pemodelan perangkat lunak *prototyping*. Sehingga terdapat tahap pembuatan purwarupa dengan melibatkan aktor. Pembuatan purwarupa dilakukan secara berulang-ulang dengan aktor hingga aktor tersebut menerima atas desain yang ditawarkan.

Implementasi sistem tujuannya melaksanakan pembuatan sistem sesuai dengan perancangan. Implementasi sistem terdiri dari beberapa tahap diantaranya Implementasi basis data, Implementasi *class* diagram, Implementasi

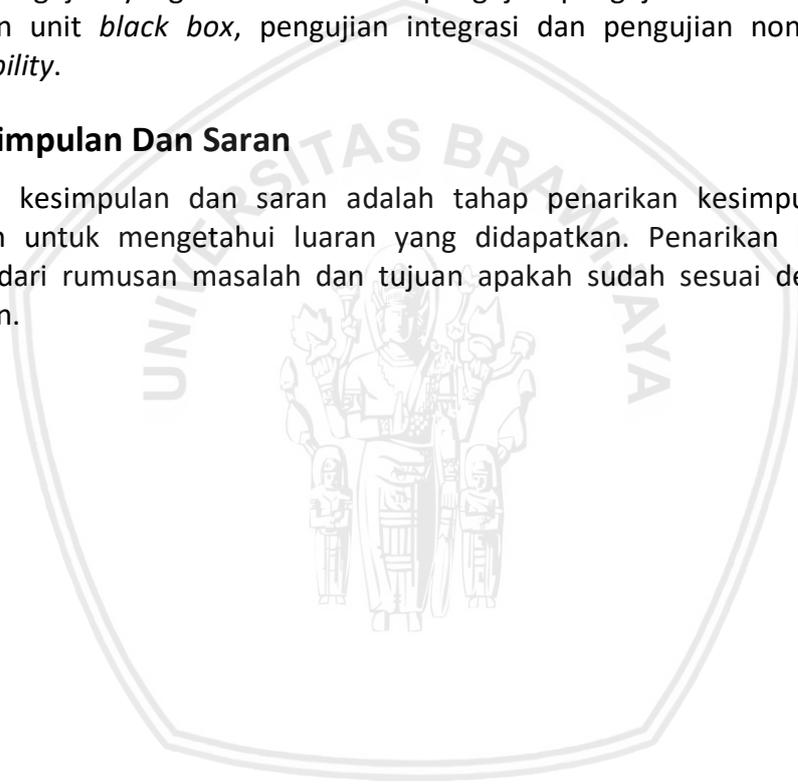
kode program, Implementasi antarmuka. Pada tahap implmentasi bertujuan untuk mengembangkan perangkat lunak berdasarkan perancangan yang telah dibuat. Implementasi program menggunakan *framework CodeIgniter* sebagai *environment* pengembangan sistem. *CodeIgniter* adalah *framework* bahasa pemrograman PHP. Pada pengembangannya akan digunakan 2 buah alamat, pertama alamat aplikasi kenaikan pangkat dan/atau jabatan fungsional, yang kedua adalah untuk *webservice* pengambilan data *dummy* SIADO dan SIMPEG. Pada implementasi basis data menggunakan *MySql*.

3.4 Pengujian

Pada tahap pengujian bertujuan untuk menguji hasil dari penelitian yang dibuat. Pengujian yang dilakukan adalah pengujian pengujian unit *white box* dan pengujian unit *black box*, pengujian integrasi dan pengujian non-fungsional *compatibility*.

3.5 Kesimpulan Dan Saran

Tahap kesimpulan dan saran adalah tahap penarikan kesimpulan dalam penelitian untuk mengetahui luaran yang didapatkan. Penarikan kesimpulan didapat dari rumusan masalah dan tujuan apakah sudah sesuai dengan hasil pengujian.

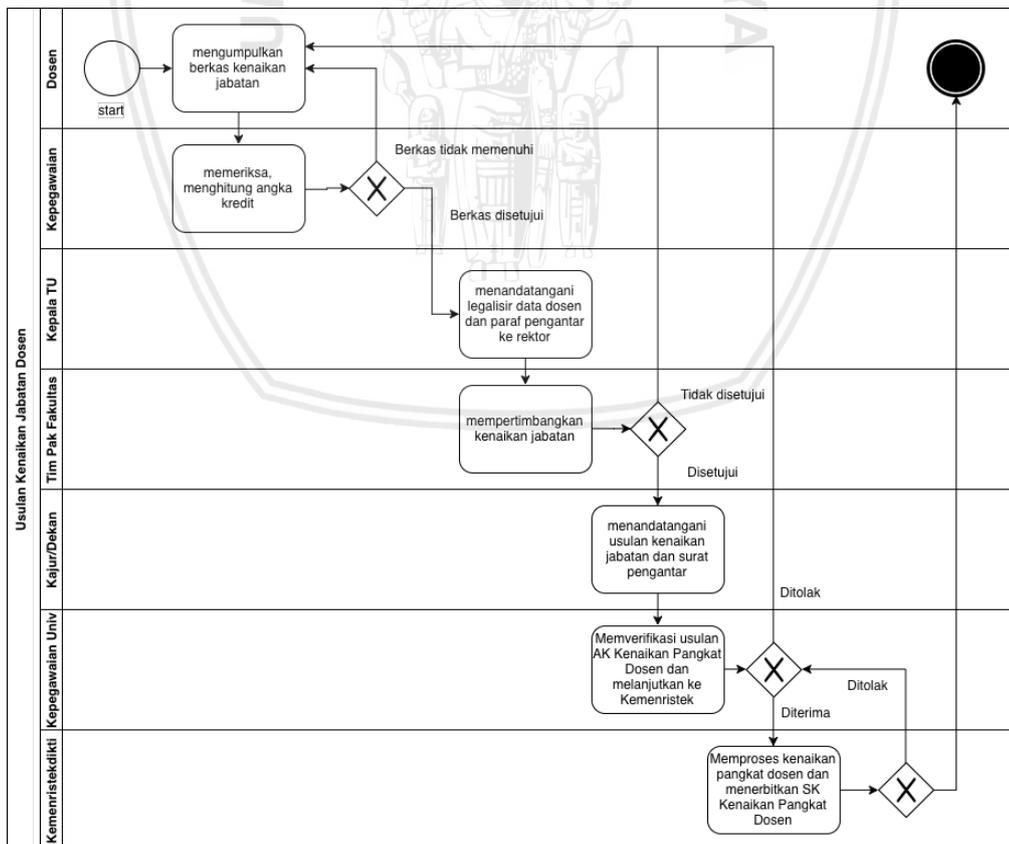


BAB 4 REKAYASA KEBUTUHAN

Pada bab rekayasa kebutuhan membahas tentang elisitasi kebutuhan dan analisis kebutuhan. Pada tahap elisitasi kebutuhan membahas hasil dari wawancara dan pengumpulan dokumen berupa proses bisnis *as-is*. Pada tahap analisis kebutuhan membahas iterasi perancangan, proses bisnis *to-be*, identifikasi aktor, pendefinisian kebutuhan fungsional dan non-fungsional.

4.1 Elisitasi Kebutuhan

Elisitasi kebutuhan adalah tahap yang pertama dilakukan dalam mengumpulkan data pendukung untuk penelitian ini. Data pendukung itu adalah wawancara beberapa dosen FILKOM dan staf kepegawaian dan SOP. Setelah dilakukan wawancara kepada dosen terdapat permasalahan utama adalah dosen banyak memiliki kesibukan sehingga tidak sempat dalam melakukan kenaikan pangkat dan/atau jabatan yang sifatnya tidak wajib. Selanjutnya wawancara kepada staf kepegawaian mengenai alur dari kenaikan pangkat dan/atau jabatan. Dari hasil wawancara tersebut didapatkan bahwa SOP kenaikan pangkat dan jabatan adalah sama namun yang berbeda adalah berkas administrasi dan waktu pengumpulannya. Gambar 4.1 menggambarkan proses dosen dalam mengusulkan kenaikan jabatan dosen hingga mendapatkan Sk jabatan



Gambar 4.1 Proses bisnis As-is

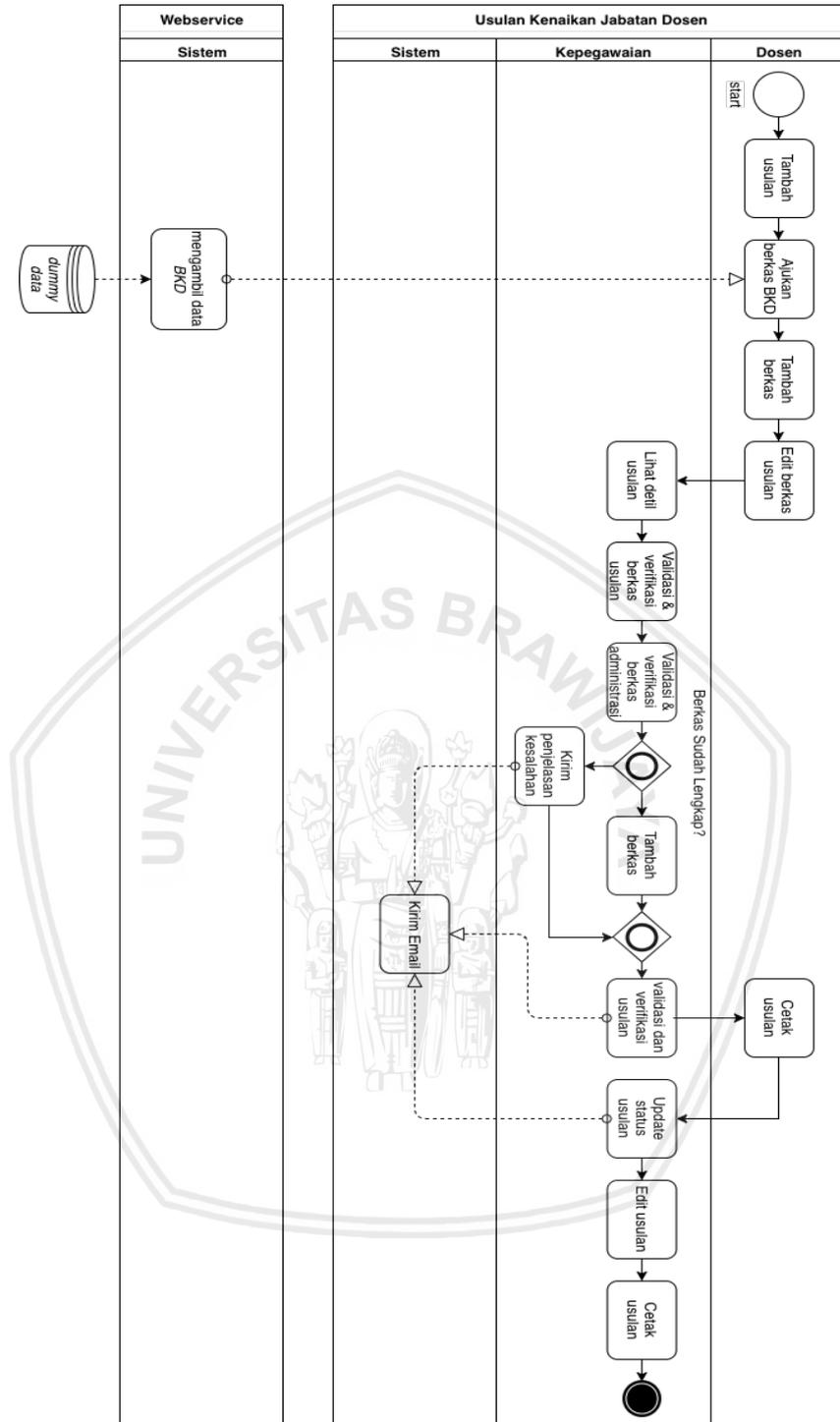


4.2 Analisis Kebutuhan Sistem

Dari hasil elisitasi kebutuhan maka penulis membuat suatu solusi untuk permasalahan tersebut, yaitu dengan melakukan pembuatan sistem kenaikan pangkat dan jabatan dosen FILKOM. Pada analisis kebutuhan sistem dilakukan beberapa proses identifikasi aktor dan pendefinisian kebutuhan fungsional dan non-fungsional. Dalam mendefinisikan aktor dan kebutuhan fungsional maka harus menganalisis proses bisnis sistem. Dimulai dari menganalisis proses bisnis yang sudah ada (proses bisnis *as-is*) dengan permasalahan pada elisitasi yang menghasilkan proses bisnis *to-be*. Proses bisnis *to-be* selain mereferensi dari proses bisnis *as-is* juga menggambarkan cara kerja sistem berdasarkan rencana solusi penelitian pada sistem yang dibuat. Pada gambar 4.2 menggambarkan alur sistem dimulai dari pengusulan hingga status selesai.

Alur dimulai dari dosen melakukan tambah usulan pangkat atau jabatan. Selanjutnya dosen dapat mengajukan berkas kegiatan BKD menjadi berkas yang diusulkan pada usulan. Dosen juga dapat menambah secara mandiri berkas kegiatan 4 bidang dan berkas administrasi. Setelah menambah berkas, dosen dapat mengatur berkas-berkas yang telah ditambahkan akan diusulkan ke usulan yang dikehendaki. Setelah itu proses dosen telah selesai dan menunggu konfirmasi kepegawaian untuk status usulannya.

Setelah itu alur kepegawaian melakukan validasi dan verifikasi pada berkas dosen. Jika terdapat kesalahan maka, kepegawaian dapat memberikan penjelasan kesalahan pada sistem yang nantinya akan diteruskan ke *email* dosen bersangkutan. Jika berkas masih kurang maka kepegawaian juga dapat menambahkan berkas untuk dosen dari arsip kepegawaian. Jika sudah dan memenuhi maka kepegawaian dapat melakukan validasi dan verifikasi usulan dan dosen tersebut diberi tahu melalui *email* dan dosen mengumpulkan berkas yang tervalidasi dan verifikasi pada bagian kepegawaian. Setelah itu kepegawaian melakukan *update* status usulan milik dosen dan status tersebut terkirim ke *email* masing-masing dosen. Jika status usulan selesai tandanya bahwa tahap dimana sk jabatan telah selesai maka kepegawaian melakukan *edit* dupak mengisi data pengusul, Kepala jurusan, Dekan/Rektor, dan data dari Tim penilai. Selanjutnya tahap terakhir kepegawaian dapat melakukan Cetak dupak untuk dilanjutkan pada tahap selanjutnya.



Gambar 4.2 Proses bisnis *To-be*

4.2.1 Iterasi 1

Pada Iterasi perancangan 1 dibuatlah inialisasi definisi kebutuhan pada tabel 4.1. Saat melakukan *review design* pada pengguna pada iterasi pertama terdapat perubahan pada kebutuhan Tambah Berkas, Pada kebutuhan Tambah Berkas yang awalnya dibuat dengan masukan Nama Berkas, Tanggal, Unsur Kegiatan dan *File* Berkas diubah menjadi Unsur Bidang, Kategori, Satuan Hasil,

Unsur AK, Nama Berkas, Tanggal, Jumlah Volume kegiatan, Jumlah Ak, Keterangan dan *File* Berkas.

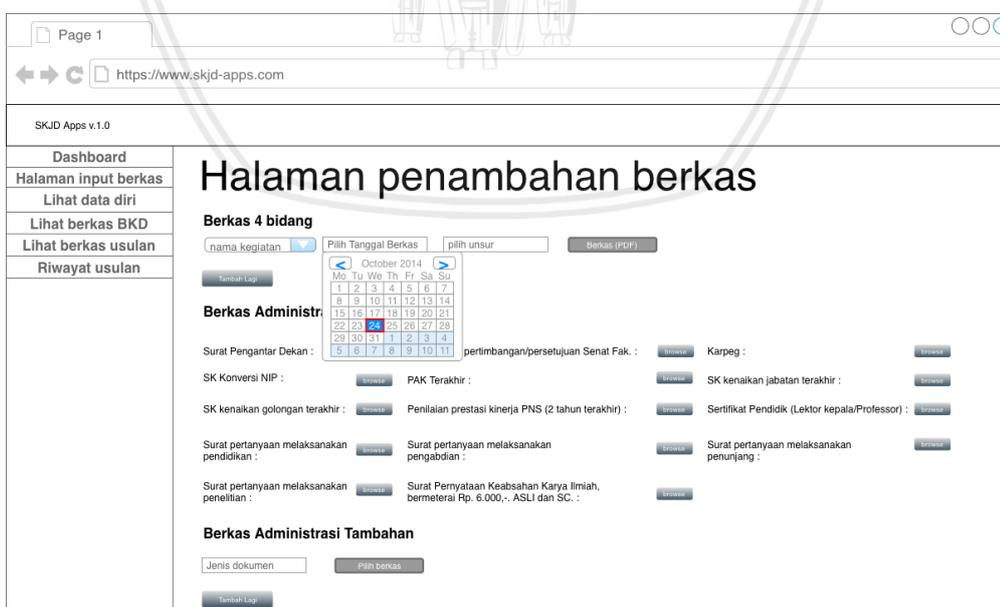
Tabel 4.1 Inisialisasi kebutuhan Iterasi pertama

Nama kebutuhan	Kebutuhan di <i>use case</i>	Spesifikasi
Pengguna dapat <i>login</i> .	<i>Login</i>	PL mampu mampu menampilkan untuk <i>input email, password</i> , dan tombol <i>login</i> untuk memproses verifikasi aktor.
Dosen dan kepegawaian dapat <i>logout</i> .	<i>Logout</i> .	PL mampu mampu menampilkan untuk aktor keluar dari sistem.
Dosen dapat tambah usulan.	Tambah usulan.	PL mampu mampu menampilkan untuk memasukan jenis usulan berupa <i>dropdown</i> dan tombol <i>save</i> untuk menyimpan.
Dosen dapat edit usulan.	<i>Edit</i> usulan.	PL mampu mampu menampilkan mengubah jenis usulan untuk dosen berupa <i>dropdown</i> dan memberi tombol <i>save</i> untuk mengubah.
Dosen dapat hapus usulan.	Hapus usulan.	PL mampu mampu menampilkan menghapus usulan dengan menggunakan tombol berwarna merah pada tabel.
Dosen dapat lihat usulan.	Lihat usulan.	PL mampu mampu menampilkan untuk menampilkan ID usulan, tanggal validasi dan verifikasi, jenis pengajuan, kum saat ini, target kum, status usulan, dan opsi cetak, edit dan hapus berupa tabel.
Dosen dan Kepegawaian dapat Cetak dupak.	Cetak dupak.	PL mampu mampu menampilkan untuk menCetak dupak berekstensi <i>.pdf</i> .
Dosen dan Kepegawaian dapat tambah berkas.	Tambah berkas.	PL mampu mampu menampilkan untuk aktor menambah berkas administrasi dan berkas 4 bidang berupa <i>form</i> Nama Berkas, Tanggal, Unsur Kegiatan dan <i>File</i> Berkas dan tombol unggah semua berkas untuk mengunggah semua berkas.
Dosen dapat hapus berkas administrasi.	Hapus berkas administrasi.	PL mampu mampu menampilkan menghapus berkas administrasi dengan menggunakan tombol berwarna merah pada tabel.
Dosen dapat hapus berkas 4 bidang.	Hapus berkas usulan.	PL mampu mampu menampilkan menghapus berkas 4 bidang dengan menggunakan tombol berwarna merah pada tabel.
Dosen dapat edit berkas 4 bidang.	Edit berkas usulan.	PL mampu mampu menampilkan <i>input</i> nama berkas, <i>dropdown</i> unsur berkas dan usulan, usulan, <i>datepicker</i> tanggal, unggah berkas dan memberi tombol <i>save</i> untuk mengubah.

Dosen dapat lihat berkas 4 bidang.	Lihat berkas usulan.	PL mampu menampilkan untuk menampilkan unsur, unsur bidang, AK, usulan, nama kegiatan, <i>file</i> , status berkas dan opsi edit dan hapus berupa tabel.
Dosen dapat lihat data diri.	Lihat data diri.	PL mampu menampilkan data diri dosen terdiri dari NIP/NIK, nama, jenis kelamin, Pendidikan terakhir, program studi, jurusan, pangkat/golongan, jabatan, tmt jabatan dan tmt pangkat. Serta menampilkan daftar tabel berkas administrasi yang terdiri ID Berkas, jenis berkas, status, <i>file</i> , dan opsi hapus.
Dosen dapat lihat berkas BKD.	Lihat berkas BKD.	PL mampu menampilkan untuk menampilkan <i>dropdown</i> set unsur 4 bidang, <i>datepicker</i> set tanggal berkas, nama kegiatan, kelompok kegiatan, sk penugasan bukti kinerja dan opsi ajukan berupa tabel.
Dosen dapat ajukan berkas BKD.	Ajukan berkas BKD.	PL mampu menampilkan tombol ajukan pada tabel berkas BKD berwarna hijau.
Dosen dapat lihat grafik AK.	Lihat grafik AK.	PL mampu menampilkan grafik batang kekurangan AK jabatan atau pangkat.
<i>Cron job</i> dapat mengekusi fungsi pemberitahuan kenaikan pangkat kepada dosen.	Kirim Pemberitahuan kenaikan pangkat dan jabatan	PL mampu menampilkan fungsi untuk pengiriman <i>email</i> pemberitahuan kenaikan pangkat atau jabatan kepada dosen.
Kepegawaian dapat lihat daftar dosen.	Lihat daftar dosen.	PL mampu menampilkan untuk menampilkan data jumlah dosen berdasarkan jabatan berupa grafik, jumlah berdasarkan jabatan, jumlah berdasarkan status pengajuan dan tabel berupa nama, nip/nik, program studi, jurusan dan opsi lihat usulan berupa tabel.
Kepegawaian dapat lihat daftar usulan dosen.	Lihat usulan dosen.	PL mampu menampilkan untuk menampilkan unsur, unsur bidang, AK, usulan, nama kegiatan, <i>file</i> , status berkas dan opsi detail berupa tabel.
Kepegawaian dapat lihat detail usulan.	Lihat detail usulan.	PL mampu menampilkan tombol validasi dan verifikasi usulan, tombol kirim penjelasan kesalahan, tombol tambah berkas, tombol edit usulan, tombol Cetak dupak. Serta menampilkan data diri dosen, daftar tabel berkas administrasi dan daftar tabel berkas 4 bidang.
Kepegawaian dapat validasi dan verifikasi usulan.	Validasi dan verifikasi usulan.	PL mampu menampilkan tombol validasi dan verifikasi usulan berwarna hijau.
Kepegawaian dapat validasi dan verifikasi berkas	Validasi dan verifikasi	PL mampu menampilkan tombol validasi dan verifikasi berkas berwarna hijau pada pilihan

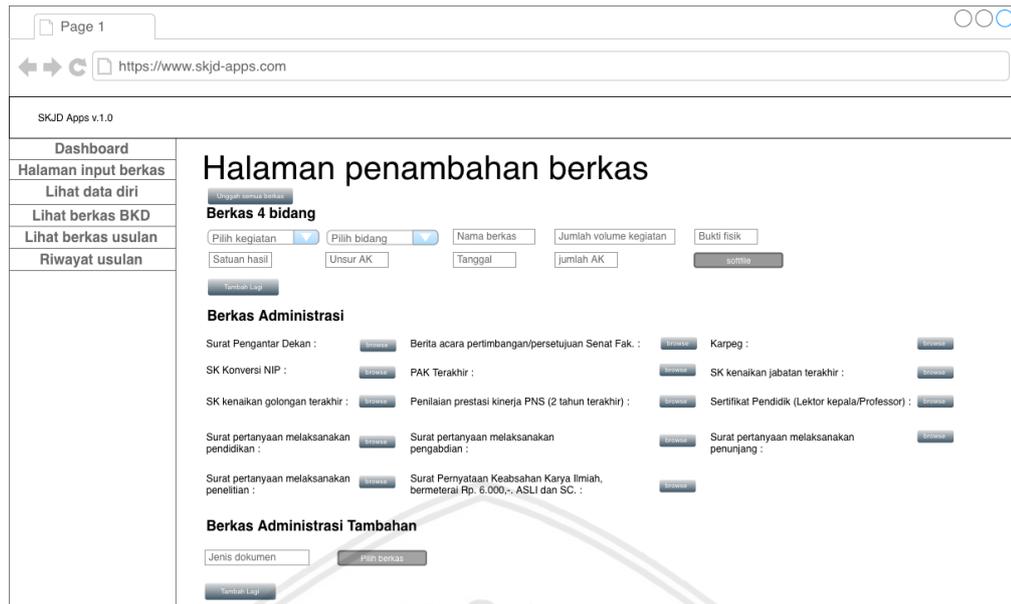
administrasi.	berkas administrasi.	opsi tabel administrasi.
Kepegawaian dapat batal validasi dan verifikasi berkas administrasi.	Batal validasi dan verifikasi berkas administrasi.	PL mampu menampilkan tombol batal validasi dan verifikasi berkas berwarna merah pada pilihan opsi tabel administrasi.
Kepegawaian dapat validasi dan verifikasi berkas 4 bidang.	Validasi dan verifikasi berkas usulan.	PL mampu menampilkan tombol validasi dan verifikasi berkas berwarna hijau pada pilihan opsi tabel berkas 4 bidang.
Kepegawaian dapat batal validasi dan verifikasi 4 bidang.	Batal validasi dan verifikasi berkas usulan.	PL mampu menampilkan tombol batal validasi dan verifikasi berkas berwarna merah pada pilihan opsi tabel berkas 4 bidang.
Kepegawaian dapat kirim penjelasan kesalahan.	Kirim penjelasan kesalahan	PL mampu menampilkan nama dosen tujuan, <i>textarea</i> pesan revisi kesalahan dan tombol <i>save</i> untuk mengirim pesan.
Kepegawaian dapat <i>update</i> status usulan.	<i>update</i> status usulan.	PL mampu menampilkan ID usulan, nama dosen, jenis pengajuan, <i>dropdown</i> status usulan yang jika memilih selesai, <i>datepicker</i> tmt pangkat dan jabatan, <i>input</i> AK dan tombol <i>save</i> memperbarui status usulan.
Kepegawaian dapat <i>edit</i> dupak.	<i>Edit</i> dupak	PL mampu menampilkan 3 <i>form</i> masukan yaitu data dekan/kajur, data pendukung usulan, data 4 bidang dari tim penilai untuk kepegawaian dan tombol <i>save</i> .

Gambar sebelum dilakukan iterasi pada gambar 4.3 sedangkan gambar setelah dilakukan iterasi pada gambar 4.4.



Gambar 4.3 Perancangan tampilan Tambah Berkas Sebelum





Gambar 4.4 Perancangan tampilan Tambah Berkas Sesudah

4.2.2 Iterasi 2

Pada Iterasi perancangan kedua pengguna telah menyetujui bahwa perancangan desain sistem telah sesuai dengan kebutuhan pengguna. Setelah itu maka dari hasil tersebut didapatkan hasil akhir dari kebutuhan final. Tabel 4.2 menjelaskan kebutuhan yang berubah berdasarkan spesifikasi.

Tabel 4.2 Perubahan kebutuhan Iterasi kedua

Nama kebutuhan	Kebutuhan di use case	Spesifikasi
Dosen dan Kepegawaian dapat tambah berkas.	Tambah berkas.	PL mampu menampilkan untuk aktor menambah berkas administrasi dan berkas 4 bidang berupa <i>form</i> Nama Berkas, Tanggal, Unsur Kegiatan dan <i>File</i> Berkas diubah menjadi Unsur Bidang, Kategori, Satuan Hasil, Unsur AK, Nama Berkas, Tanggal, Jumlah Volume kegiatan, Jumlah Ak, Keterangan dan <i>File</i> Berkas. dan tombol unggah semua berkas untuk mengunggah semua berkas.

4.2.3 Identifikasi Aktor

Pada tahap ini mendeskripsikan peran aktor yang terlibat dalam sistem. Identifikasi aktor didapat dari rumusan masalah dan dari proses bisnis. Tabel 4.3 mendeskripsikan peran aktor pada sistem kenaikan pangkat dan jabatan dosen FILKOM.



Tabel 4.3 Tabel Identifikasi Aktor

Nama Aktor	Deskripsi
Dosen	Merupakan aktor yang berperan dalam mengusulkan kenaikan pangkat dan/atau jabatan.
Kepegawaian	Merupakan aktor yang melegalisasi usulan, berkas administrasi, berkas 4 bidang, untuk dilanjutkan ke tahap pengusulan selanjutnya.

4.2.4 Pendefinisian Kebutuhan

Tabel 4.4 menjelaskan tentang pendefinisian kebutuhan fungsional yakni hasil akhir dari kebutuhan pada iterasi dan tabel 4.5 tentang kebutuhan non-fungsional. Penomoran SPJ adalah mendefinisikan nama sistem, Nomor pada posisi tengah menandakan kebutuhan fungsional atau non-fungsional dan dua angka terakhir adalah nomor dari setiap kebutuhan.

Tabel 4.4 Tabel Kebutuhan Fungsional

Kode kebutuhan	Nama kebutuhan	Kebutuhan di <i>use case</i>	Spesifikasi
SPJ_01_01	Pengguna dapat <i>login</i> .	<i>Login</i>	PL mampu mampu menampilkan untuk <i>input email, password</i> , dan tombol <i>login</i> untuk memproses verifikasi aktor.
SPJ_01_02	Dosen dan kepegawaian dapat <i>logout</i> .	<i>Logout</i> .	PL mampu mampu menampilkan untuk aktor keluar dari sistem.
SPJ_01_03	Dosen dapat tambah usulan.	Tambah usulan.	PL mampu mampu menampilkan untuk memasukan jenis usulan berupa <i>dropdown</i> dan tombol <i>save</i> untuk menyimpan.
SPJ_01_04	Dosen dapat edit usulan.	<i>Edit</i> usulan.	PL mampu mampu menampilkan mengubah jenis usulan untuk dosen berupa <i>dropdown</i> dan memberi tombol <i>save</i> untuk mengubah.
SPJ_01_05	Dosen dapat hapus usulan.	Hapus usulan.	PL mampu mampu menampilkan menghapus usulan dengan menggunakan tombol berwarna merah pada tabel.
SPJ_01_06	Dosen dapat lihat usulan.	Lihat usulan.	PL mampu mampu menampilkan untuk menampilkan ID usulan, tanggal validasi dan verifikasi, jenis pengajuan, kum saat ini, target kum, status usulan, dan opsi cetak, edit dan hapus berupa tabel.
SPJ_01_07	Dosen dan Kepegawaian dapat Cetak dupak.	Cetak dupak.	PL mampu mampu menampilkan untuk menCetak dupak berekstensi <i>.pdf</i> .
SPJ_01_08	Dosen dan Kepegawaian dapat tambah berkas.	Tambah berkas.	PL mampu mampu menampilkan untuk aktor menambah berkas administrasi dan berkas 4 bidang berupa <i>form</i> dan tombol unggah semua berkas untuk mengunggah semua berkas.



SPJ_01_09	Dosen dapat hapus berkas administrasi.	Hapus berkas administrasi.	PL mampu mampu menampilkan menghapus berkas administrasi dengan menggunakan tombol berwarna merah pada tabel.
SPJ_01_10	Dosen dapat hapus berkas 4 bidang.	Hapus berkas usulan.	PL mampu mampu menampilkan menghapus berkas 4 bidang dengan menggunakan tombol berwarna merah pada tabel.
SPJ_01_11	Dosen dapat edit berkas 4 bidang.	Edit berkas usulan.	PL mampu mampu menampilkan <i>input</i> nama berkas, <i>dropdown</i> unsur berkas dan usulan, usulan, <i>datepicker</i> tanggal, unggah berkas dan memberi tombol <i>save</i> untuk mengubah.
SPJ_01_12	Dosen dapat lihat berkas 4 bidang.	Lihat berkas usulan.	PL mampu mampu menampilkan untuk menampilkan unsur, unsur bidang, AK, usulan, nama kegiatan, <i>file</i> , status berkas dan opsi edit dan hapus berupa tabel.
SPJ_01_13	Dosen dapat lihat data diri.	Lihat data diri.	PL mampu menampilkan data diri dosen terdiri dari NIP/NIK, nama, jenis kelamin, Pendidikan terakhir, program studi, jurusan, pangkat/golongan, jabatan, tmt jabatan dan tmt pangkat. Serta menampilkan daftar tabel berkas administrasi yang terdiri ID Berkas, jenis berkas, status, <i>file</i> , dan opsi hapus.
SPJ_01_14	Dosen dapat lihat berkas BKD.	Lihat berkas BKD.	PL mampu mampu menampilkan untuk menampilkan <i>dropdown</i> set unsur 4 bidang, <i>datepicker</i> set tanggal berkas, nama kegiatan, kelompok kegiatan, sk penugasan bukti kinerja dan opsi ajukan berupa tabel.
SPJ_01_15	Dosen dapat ajukan berkas BKD.	Ajukan berkas BKD.	PL mampu menampilkan tombol ajukan pada tabel berkas BKD berwarna hijau.
SPJ_01_16	Dosen dapat lihat grafik AK.	Lihat grafik AK.	PL mampu menampilkan grafik batang kekurangan AK jabatan atau pangkat.
SPJ_01_17	<i>Cron job</i> dapat mengekusi fungsi pemberitahuan kenaikan pangkat kepada dosen.	Kirim Pemberitahuan kenaikan pangkat dan jabatan	PL mampu menampilkan fungsi untuk pengiriman <i>email</i> pemberitahuan kenaikan pangkat atau jabatan kepada dosen.
SPJ_01_18	Kepegawaian dapat lihat daftar dosen.	Lihat daftar dosen.	PL mampu mampu menampilkan untuk menampilkan data jumlah dosen berdasarkan jabatan berupa grafik, jumlah berdasarkan jabatan, jumlah berdasarkan status pengajuan dan tabel berupa nama, nip/nik, program studi, jurusan dan opsi lihat usulan berupa tabel.
SPJ_01_19	Kepegawaian dapat lihat daftar usulan	Lihat usulan dosen.	PL mampu mampu menampilkan untuk menampilkan unsur, unsur bidang, AK,

	dosen.		usulan, nama kegiatan, <i>file</i> , status berkas dan opsi detil berupa tabel.
SPJ_01_20	Kepegawaian dapat lihat detil usulan.	Lihat detil usulan.	PL mampu menampilkan tombol validasi dan verifikasi usulan, tombol kirim penjelasan kesalahan, tombol tambah berkas, tombol edit usulan, tombol Cetak dupak. Serta menampilkan data diri dosen, daftar tabel berkas administrasi dan daftar tabel berkas 4 bidang.
SPJ_01_21	Kepegawaian dapat validasi dan verifikasi usulan.	Validasi dan verifikasi usulan.	PL mampu menampilkan tombol validasi dan verifikasi usulan berwarna hijau.
SPJ_01_22	Kepegawaian dapat validasi dan verifikasi berkas administrasi.	Validasi dan verifikasi berkas administrasi.	PL mampu menampilkan tombol validasi dan verifikasi berkas berwarna hijau pada pilihan opsi tabel administrasi.
SPJ_01_23	Kepegawaian dapat batal validasi dan verifikasi berkas administrasi.	Batal validasi dan verifikasi berkas administrasi.	PL mampu menampilkan tombol batal validasi dan verifikasi berkas berwarna merah pada pilihan opsi tabel administrasi.
SPJ_01_24	Kepegawaian dapat validasi dan verifikasi berkas 4 bidang.	Validasi dan verifikasi berkas usulan.	PL mampu menampilkan tombol validasi dan verifikasi berkas berwarna hijau pada pilihan opsi tabel berkas 4 bidang.
SPJ_01_25	Kepegawaian dapat batal validasi dan verifikasi 4 bidang.	Batal validasi dan verifikasi berkas usulan.	PL mampu menampilkan tombol batal validasi dan verifikasi berkas berwarna merah pada pilihan opsi tabel berkas 4 bidang.
SPJ_01_26	Kepegawaian dapat kirim penjelasan kesalahan.	Kirim penjelasan kesalahan	PL mampu menampilkan nama dosen tujuan, <i>textarea</i> pesan revisi kesalahan dan tombol <i>save</i> untuk mengirim pesan.
SPJ_01_27	Kepegawaian dapat <i>update</i> status usulan.	<i>update</i> status usulan.	PL mampu menampilkan ID usulan, nama dosen, jenis pengajuan, <i>dropdown</i> status usulan yang jika memilih selesai, <i>datepicker</i> tmt pangkat dan jabatan, <i>input</i> AK dan tombol <i>save</i> memperbarui status usulan.
SPJ_01_28	Kepegawaian dapat <i>edit</i> dupak.	<i>Edit</i> dupak	PL mampu menampilkan 3 <i>form</i> masukan yaitu data dekan/kajur, data pendukung usulan, data 4 bidang dari tim penilai untuk kepegawaian dan tombol <i>save</i> .

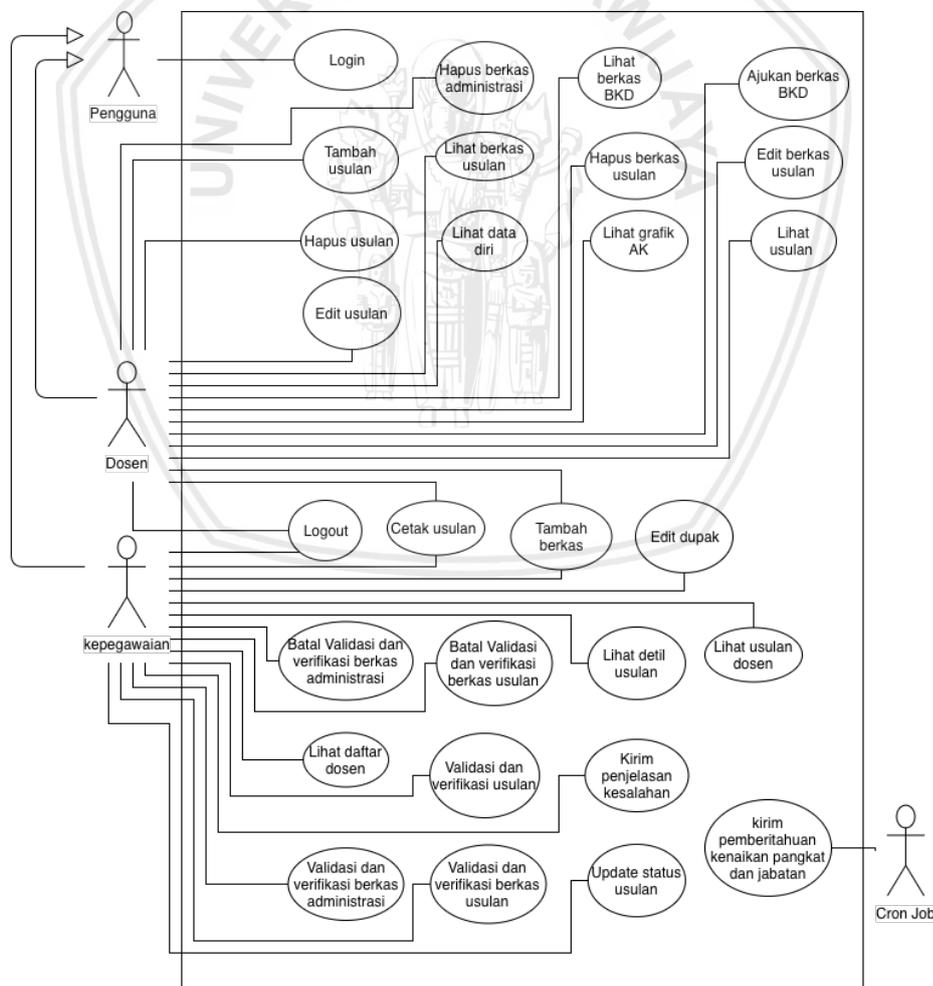
Kebutuhan non fungsional didefinisikan untuk membantu tercapainya tujuan dalam kebutuhan fungsional. Pengujian non fungsional akan menggunakan pengujian *compatibility*. Pengujian *compatibility* menguji sistem dapat berjalan di berbagai *browser*. Pada tabel 4.4 mendefinisikan kebutuhan non fungsional sistem.

Tabel 4.5 Tabel Kebutuhan Non Fungsional

Kode kebutuhan	Kategori	Nama kebutuhan	Spesifikasi
SPJ_02_01	<i>Compatibility</i>	Sistem dapat kompatibel di berbagai browser	PL mampu dijalankan pada pada <i>browser Safari 12, Microsoft Edge 18, Firefox 64, Chrome 71 dan Opera 57</i>

4.3 Use case Diagram

Use case diagram berfungsi untuk menggambarkan atau memetakan kebutuhan tiap aktor. Gambar 4.5 menggambarkan penggunaan *use case diagram* pada sistem kenaikan pangkat dan jabatan FILKOM.



Gambar 4.5 Use case diagram

4.4 Use case Scenario Diagram

Use case scenario diagram adalah sebuah diagram yang memberikan detail dari proses-proses pada setiap *use case*. Sehingga perlu dilakukan pendetilan proses di setiap *use case* yang digambarkan pada Gambar 4.5. Tabel 4.6 menjelaskan tentang proses dari *use case login*. *Login* melakukan proses masuknya pengguna pada sistem. Sistem nantinya akan mendefinisikan pengguna akan tergolong aktor kepegawaian atau dosen.

Tabel 4.6 Tabel use case scenario login

Kode	SPJ_01_01
Tujuan	Aktor masuk ke dalam sistem dan terdefinisi sebagai kepegawaian atau dosen.
Aktor	Pengguna.
Prakondisi	Aktor telah berada pada halaman <i>login</i> sistem.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor mengisi data <i>form email</i> dan <i>password</i>. 2. Aktor menekan tombol <i>login</i>. 3. Sistem melakukan proses identifikasi aktor. 4. Sistem mengarahkan ke halaman <i>dashboard</i>.
Aliran alternatif	3. Jika identifikasi salah, maka sistem menampilkan pesan kesalahan " <i>email</i> atau <i>password</i> salah".
Kondisi akhir	Aktor berhasil masuk ke dalam halaman <i>dashboard</i> .

Tabel 4.7 menjelaskan tentang proses dosen atau kepegawaian keluar menjadi sesi pengguna.

Tabel 4.7 Tabel use case scenario logout

Kode	SPJ_01_02
Tujuan	Aktor keluar ke halaman <i>login</i> .
Aktor	Dosen dan kepegawaian
Prakondisi	Aktor telah berada pada halaman <i>dashboard</i> .
Aliran utama	<ol style="list-style-type: none"> 1. Aktor menekan menu <i>logout</i>. 2. Sistem mengarahkan ke halaman <i>login</i>.
Aliran alternatif	-
Kondisi akhir	Aktor berada pada halaman <i>login</i> .

Tabel 4.8 menjelaskan tentang proses dari *use case* tambah usulan. Tambah usulan digunakan ketika dosen ingin melakukan pengusulan kenaikan pangkat dan/atau jabatan.

Tabel 4.8 Tabel *use case scenario* tambah usulan

Kode	SPJ_01_03
Tujuan	Aktor menambah usulan pada <i>database</i> .
Aktor	Dosen.
Prakondisi	Aktor telah berada pada halaman tambah usulan.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor memilih jenis usulan dan menekan tombol <i>save</i>. 2. Sistem menambahkan usulan pada <i>database</i>. 3. Sistem mengarahkan ke halaman riwayat usulan.
Aliran alternatif	-
Kondisi akhir	Sistem telah menambah usulan pada <i>database</i> dan tampil pada halaman riwayat usulan

Tabel 4.9 menjelaskan tentang proses dari *use case edit* usulan. *Edit* usulan adalah proses melakukan perubahan pada jenis usulan.

Tabel 4.9 Tabel *use case scenario* edit usulan

Kode	SPJ_01_04
Tujuan	Aktor mengubah jenis usulan pada <i>database</i> .
Aktor	Dosen.
Prakondisi	Aktor telah berada pada halaman edit usulan.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor memilih jenis usulan berkas dan menekan tombol <i>save</i>. 2. Sistem menyimpan perubahan jenis usulan. 3. Sistem mengarahkan ke halaman riwayat usulan.
Aliran alternatif	-
Kondisi akhir	Sistem telah mengubah usulan pada <i>database</i> dan perubahan tampil pada daftar riwayat usulan.

Tabel 4.10 menjelaskan tentang proses dari *use case* hapus usulan. Hapus usulan adalah proses melakukan penghapusan usulan yang dipilih pada *database*.

Tabel 4.10 Tabel *use case scenario* hapus usulan

Kode	SPJ_01_05
-------------	-----------

Tujuan	Aktor menghapus usulan pada <i>database</i> .
Aktor	Dosen
Prakondisi	Aktor telah berada pada halaman riwayat usulan.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor memilih usulan dan menekan tombol hapus. 2. Sistem menampilkan peringatan apakah usulan dihapus atau tidak. 3. Sistem akan menghapus usulan pada <i>database</i>.
Aliran alternatif	Jika memilih batal maka tidak akan diproses.
Kondisi akhir	Sistem telah menghapus usulan pada <i>database</i> dan usulan tidak tampil pada daftar riwayat usulan.

Tabel 4.11 menjelaskan tentang proses dari *use case* lihat usulan. lihat usulan adalah proses dalam melihat daftar usulan yang dimiliki dosen.

Tabel 4.11 Tabel *use case scenario* lihat usulan

Kode	SPJ_01_06
Tujuan	Sistem menampilkan daftar usulan aktor.
Aktor	Dosen.
Prakondisi	<ol style="list-style-type: none"> 1. Aktor telah berada pada halaman <i>dashboard</i>.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor menekan menu riwayat usulan. 2. Sistem mengambil semua data usulan dari <i>database</i> 3. Sistem menampilkan daftar usulan.
Aliran alternatif	-
Kondisi akhir	Sistem berhasil menampilkan daftar usulan aktor berbentuk tabel.

Tabel 4.12 menjelaskan tentang proses dari *use case input* cetak dupak. Cetak dupak adalah proses menampilkan dupak milik dosen berekstensi pdf.

Tabel 4.12 Tabel *use case scenario* cetak dupak

Kode	SPJ_01_07
Tujuan	Sistem dapat menampilkan dupak berekstensi pdf.
Aktor	Dosen dan kepegawaian.
Prakondisi	Aktor telah memilih usulan yang akan di cetak.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol cetak dupak. 2. Sistem sistem memproses usulan ke berkas pdf.

	3. Sistem dupak berekstensi pdf.
Aliran alternatif	-
Kondisi akhir	Sistem berhasil menampilkan dupak berekstensi pdf pada halaman baru.

Tabel 4.13 menjelaskan tentang proses dari *use case* tambah berkas. Tambah berkas adalah proses melakukan penambahan berkas usulan yang terdiri dari berkas 4 bidang dan berkas administrasi

Tabel 4.13 Tabel *use case scenario* tambah berkas

Kode	SPJ_01_08
Tujuan	Aktor dapat menyimpan data berkas administrasi dan/atau berkas 4 bidang pada <i>database</i> .
Aktor	Dosen dan kepegawaian
Prakondisi	Aktor telah berada pada halaman penambahan berkas.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor mengisi secara lengkap <i>form</i> penambahan berkas administrasi dan/atau berkas 4 bidang dan menekan tombol <i>save</i>. 2. Sistem menyimpan data berkas administrasi dan/atau 4 bidang pada <i>database</i>.
Aliran alternatif	-
Kondisi akhir	Sistem berhasil menyimpan berkas administrasi dan/atau berkas 4 bidang pada <i>database</i> .

Tabel 4.14 menjelaskan tentang proses dari *use case* hapus berkas administrasi. Hapus berkas administrasi adalah proses penghapusan berkas administrasi milik dosen.

Tabel 4.14 Tabel *use case scenario* hapus berkas administrasi

Kode	SPJ_01_09
Tujuan	Aktor dapat menghapus berkas administrasi pada <i>database</i> .
Aktor	Dosen.
Prakondisi	Aktor telah berada pada halaman data diri.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor memilih berkas administrasi dan mekan tombol hapus. 2. Sistem menampilkan peringatan apakah akan dihapus atau tidak. 3. Sistem menghapus berkas administrasi pada <i>database</i>.

Aliran alternatif	Jika pilih batal maka sistem tidak akan menghapus berkas administrasi.
Kondisi akhir	Sistem berhasil menghapus berkas administrasi pada <i>database</i> dan menghapus pada daftar berkas administrasi pada halaman data diri.

Tabel 4.15 menjelaskan tentang proses dari *use case* hapus berkas usulan. Hapus berkas usulan adalah proses penghapusan berkas usulan milik dosen.

Tabel 4.15 Tabel *use case scenario* hapus berkas usulan

Kode	SPJ_01_10
Tujuan	Aktor dapat menghapus berkas usulan pada <i>database</i> .
Aktor	Dosen.
Prakondisi	Aktor telah berada pada halaman berkas usulan.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor memilih berkas administrasi dan mekan tombol hapus. 2. Sistem menampilkan peringatan apakah akan dihapus atau tidak. 3. Sistem menghapus berkas usulan pada <i>database</i>.
Aliran alternatif	Jika pilih batal maka sistem tidak akan menghapus berkas usulan.
Kondisi akhir	Sistem berhasil menghapus berkas usulan pada <i>database</i> dan menghapus pada daftar berkas usulan pada halaman lihat berkas usulan.

Tabel 4.16 menjelaskan tentang proses dari *use case* kirim penjelasan kesalahan. Kirim penjelasan kesalahan usulan melakukan proses mengirimkan kejelasan revisi berkas usulan.

Tabel 4.16 Tabel *use case scenario* edit berkas usulan

Kode	SPJ_01_11
Tujuan	Aktor mengubah berkas usulan pada <i>database</i> .
Aktor	Dosen.
Prakondisi	Aktor berada pada halaman edit berkas usulan.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor mengisi <i>form</i> dengan lengkap dan menekan tombol <i>save</i>. 2. Sistem menyimpan perubahan berkas usulan pada <i>database</i>.
Aliran alternatif	-
Kondisi akhir	Sistem berhasil mengubah data berkas usulan pada <i>database</i> dan mengubah data pada daftar berkas usulan pada halaman

	lihat berkas usulan.
--	----------------------

Tabel 4.17 menjelaskan tentang proses dari *use case* lihat berkas usulan. Lihat berkas usulan adalah proses melihat berkas 4 bidang yang akan diusulkan untuk kenaikan pangkat dan/atau jabatan.

Tabel 4.17 Tabel *use case scenario* lihat berkas usulan

Kode	SPJ_01_12
Tujuan	Aktor dapat melihat daftar berkas usulan.
Aktor	Dosen.
Prakondisi	Aktor telah berada pada halaman <i>dashboard</i> .
Aliran utama	<ol style="list-style-type: none"> 1. Aktor menekan menu lihat berkas usulan. 2. Sistem mengambil data berkas usulan dari <i>database</i>. 3. Sistem menampilkan daftar berkas usulan.
Aliran alternatif	-
Kondisi akhir	Sistem berhasil menampilkan daftar berkas usulan berupa tabel.

Tabel 4.18 menjelaskan tentang proses dari *use case* lihat data diri. Lihat data diri adalah proses sistem dalam menampilkan data diri dosen dan daftar berkas administrasi milik dosen.

Tabel 4.18 Tabel *use case scenario* lihat data diri

Kode	SPJ_01_13
Tujuan	Aktor dapat melihat data diri dan daftar berkas administrasi.
Aktor	Dosen.
Prakondisi	Aktor telah berada pada halaman <i>dashboard</i> .
Aliran utama	<ol style="list-style-type: none"> 1. Aktor menekan menu lihat data diri. 2. Sistem mengambil data diri dan daftar berkas administrasi 3. Sistem menampilkan data diri dan daftar berkas administrasi.
Aliran alternatif	-
Kondisi akhir	Sistem berhasil menampilkan data diri aktor dan daftar berkas administrasi aktor.

Tabel 4.19 menjelaskan tentang proses dari *use case* lihat berkas BKD. Lihat berkas BKD adalah proses sistem menampilkan berkas 4 bidang milik dosen yang diambil dari data *dummy webservice* SIADO.

Tabel 4.19 Tabel *use case scenario* lihat berkas BKD

Kode	SPJ_01_14
Tujuan	Aktor dapat melihat daftar berkas BKD.
Aktor	Dosen.
Prakondisi	Aktor telah berada pada halaman <i>dashboard</i> .
Aliran utama	<ol style="list-style-type: none"> 1. Aktor menekan menu lihat berkas BKD. 2. Sistem mengambil data BKD dari <i>webservice</i> dan berkas usulan dari <i>database</i>. 3. Sistem menyeleksi berkas yang sudah diusulkan dari <i>webservice</i> BKD dengan berkas yang sudah diusulkan. 4. Sistem menampilkan daftar berkas BKD yang belum diusulkan.
Aliran alternatif	<ol style="list-style-type: none"> 3. Jika berkas data BKD belum diusulkan akan ditampilkan semua. 4. Sistem menampilkan semua daftar berkas BKD.
Kondisi akhir	Sistem berhasil menampilkan daftar berkas BKD aktor.

Tabel 4.20 menjelaskan tentang proses dari *use case* ajukan berkas BKD. Ajukan berkas BKD adalah proses mengajukan berkas 4 bidang BKD untuk diajukan menjadi berkas usulan.

Tabel 4.20 Tabel *use case scenario* ajukan berkas BKD

Kode	SPJ_01_15
Tujuan	Aktor mengajukan berkas BKD menjadi berkas usulan.
Aktor	Dosen.
Prakondisi	Aktor berada pada halaman berkas BKD.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor memilih berkas BKD, memasukan <i>dropdown search</i> kegiatan unsur 4 bidang, mengisi jumlah volume, jumlah ak dan menekan tombol centang hijau. 2. Sistem melakukan validasi masukan kegiatan unsur 4 bidang, jumlah volume dan jumlah ak. 3. Sistem memasukan data pada <i>database</i>.
Aliran alternatif	2. Jika masukan kegiatan unsur 4 bidang, jumlah volume dan jumlah ak tidak lengkap maka akan menampilkan peringatan "Pilih unsur bidang dan isi jumlah volume kegiatan dan jumlah AK".

Kondisi akhir	Sistem berhasil memasukan data BKD ke <i>database</i> berkas usulan dan ditampilkan di halaman lihat berkas usulan.
----------------------	---

Tabel 4.21 menjelaskan tentang proses dari *use case* lihat grafik AK. Lihat grafik AK adalah proses dosen melihat kekurangan AK yang dimiliki untuk mengusulkan pada pangkat atau jabatan tertentu.

Tabel 4.21 Tabel *use case scenario* lihat grafik AK

Kode	SPJ_01_16
Tujuan	Aktor dapat melihat grafik AK sesuai dengan AK yang dimiliki.
Aktor	Dosen.
Prakondisi	Aktor telah <i>login</i> .
Aliran utama	1. Aktor menekan menu <i>dashboard</i> . 2. Sistem memproses AK aktor menjadi grafik.
Aliran alternatif	-
Kondisi akhir	Sistem menampilkan kalkulasi AK aktor di grafik jabatan dan grafik pangkat.

Tabel 4.22 menjelaskan tentang proses dari *use case* kirim pemberitahuan kenaikan pangkat dan jabatan. kirim pemberitahuan kenaikan pangkat dan jabatan adalah proses melakukan pengiriman pemberitahuan pada dosen 2 tahun tepat dari TMT melalui *email* masing-masing dosen.

Tabel 4.22 Tabel *use case scenario* kirim pemberitahuan kenaikan pangkat dan jabatan

Kode	SPJ_01_17
Tujuan	Aktor dapat mengeksekusi pemberitahuan untuk kenaikan pangkat atau jabatan.
Aktor	Cron Job
Prakondisi	Aktor telah <i>online</i> .
Aliran utama	1. Aktor mengesekusi kode alamat fungsi. 2. Sistem mencari dosen yang hari ini tepat 2 tahun dikirmkan <i>email</i> . 3. Sistem mengirimkan email ke dosen yang dipilih.
Aliran alternatif	-
Kondisi akhir	Sistem telah mengirimkan <i>email</i> pemberitahuan kenaikan pangkat atau jabatan pada <i>email</i> masing-masing dosen.

Tabel 4.23 menjelaskan tentang proses dari *use case* lihat daftar dosen. Lihat daftar dosen adalah proses kepegawaian melihat daftar dosen FILKOM.

Tabel 4.23 Tabel *use case scenario* lihat daftar dosen

Kode	SPJ_01_18
Tujuan	Aktor dapat melihat daftar dosen FILKOM.
Aktor	Kepegawaian
Prakondisi	Aktor telah melakukan <i>login</i> .
Aliran utama	1. Aktor menekan menu daftar dosen. 2. Sistem menampilkan jumlah dosen berdasarkan jabatan, status pengajuan dan daftar dosen dari <i>database</i> .
Aliran alternatif	-
Kondisi akhir	Sistem berhasil menampilkan daftar dosen berupa tabel.

Tabel 4.24 menjelaskan tentang proses dari *use case* lihat usulan dosen. Lihat daftar dosen adalah proses kepegawaian melihat daftar usulan dosen.

Tabel 4.24 Tabel *use case scenario* lihat usulan dosen

Kode	SPJ_01_19
Tujuan	Aktor dapat melihat daftar usulan dosen FILKOM.
Aktor	Kepegawaian
Prakondisi	Aktor telah melakukan <i>login</i> .
Aliran utama	1. Aktor menekan menu daftar usulan. 2. Sistem menampilkan daftar usulan dosen dari <i>database</i> .
Aliran alternatif	-
Kondisi akhir	Sistem dapat melihat daftar usulan dosen berupa tabel.

Tabel 4.25 menjelaskan tentang proses dari *use case* lihat detail dosen. Lihat detail dosen adalah proses kepegawaian melihat data diri dosen, daftar berkas administrasi dan daftar berkas usulan dosen.

Tabel 4.25 Tabel *use case scenario* lihat detail dosen

Kode	SPJ_01_20
Tujuan	Aktor dapat melihat detail usulan dosen FILKOM.
Aktor	Kepegawaian

Prakondisi	Aktor berada pada halaman daftar usulan.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor memilih usulan dan menekan tombol detail. 2. Sistem menampilkan data diri dosen, daftar berkas administrasi dosen dan daftar berkas usulan dosen yang diambil dari <i>database</i>.
Aliran alternatif	-
Kondisi akhir	Sistem berhasil menampilkan data diri dosen, daftar berkas administrasi dosen berupa tabel dan daftar berkas usulan dosen berupa tabel.

Tabel 4.26 menjelaskan tentang proses dari *use case* validasi dan verifikasi usulan. Validasi dan verifikasi usulan adalah proses kepegawaian dalam menyetujui usulan dosen untuk diteruskan ke tahap selanjutnya.

Tabel 4.26 Tabel *use case scenario* validasi dan verifikasi usulan

Kode	SPJ_01_21
Tujuan	Aktor dapat merubah status menjadi “di kepegawaian sudah divalidasi”.
Aktor	Kepegawaian
Prakondisi	Aktor berada pada halaman detail berkas usulan dosen.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol validasi dan verifikasi usulan. 2. Sistem memberikan peringatan apakah divalidasi dan verifikasi usulan atau batal. 3. Sistem akan merubah status usulan menjadi menjadi “di kepegawaian sudah divalidasi”. 4. Sistem akan mengirimkan <i>email</i> pemberitahuan. 5. Sistem akan memberikan peringatan sukses dikirm.
Aliran alternatif	Jika memilih batal maka tidak akan diproses selanjutnya.
Kondisi akhir	Sistem berhasil merubah status usulan menjadi menjadi “di kepegawaian sudah divalidasi”.

Tabel 4.27 menjelaskan tentang proses dari *use case* validasi dan verifikasi berkas administrasi. Validasi dan verifikasi berkas administrasi adalah proses kepegawaian menyetujui berkas administrasi telah sesuai ketentuan.

Tabel 4.27 Tabel *use case scenario* validasi dan verifikasi berkas administrasi

Kode	SPJ_01_22
Tujuan	Aktor dapat merubah status berkas menjadi “sudah divalidasi”.



Aktor	Kepegawaian
Prakondisi	Aktor berada pada halaman detil berkas usulan dosen.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor memilih berkas dan menekan tombol validasi dan verifikasi pada tabel berkas administrasi. 2. Sistem mengubah status berkas administrasi di <i>database</i> menjadi "sudah divalidasi".
Aliran alternatif	-
Kondisi akhir	Sistem berhasil merubah status berkas menjadi menjadi "sudah divalidasi".

Tabel 4.28 menjelaskan tentang proses dari *use case* batal validasi dan verifikasi berkas administrasi. Batal validasi dan verifikasi berkas administrasi adalah proses kepegawaian batal menyetujui berkas administrasi telah sesuai ketentuan.

Tabel 4.28 Tabel *use case scenario* batal validasi dan verifikasi berkas administrasi

Kode	SPJ_01_23
Tujuan	Aktor dapat merubah status berkas menjadi "belum divalidasi".
Aktor	Kepegawaian
Prakondisi	Aktor berada pada halaman detil berkas usulan dosen.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor memilih berkas dan menekan tombol batal validasi dan verifikasi pada tabel berkas administrasi. 2. Sistem mengubah status berkas administrasi di <i>database</i> menjadi "belum divalidasi".
Aliran alternatif	-
Kondisi akhir	Sistem berhasil merubah status berkas menjadi menjadi "belum divalidasi".

Tabel 4.29 menjelaskan tentang proses dari *use case* validasi dan verifikasi berkas usulan. Validasi dan verifikasi berkas usulan adalah proses kepegawaian menyetujui berkas usulan telah sesuai ketentuan.

Tabel 4.29 Tabel *use case scenario* validasi dan verifikasi berkas usulan

Kode	SPJ_01_24
Tujuan	Aktor dapat merubah status berkas menjadi "sudah divalidasi".
Aktor	Kepegawaian
Prakondisi	Aktor berada pada halaman detil berkas usulan dosen.

Aliran utama	<ol style="list-style-type: none"> 1. Aktor memilih berkas dan menekan tombol validasi dan verifikasi pada tabel berkas 4 bidang. 2. Sistem mengubah status berkas usulan di <i>database</i> menjadi "sudah divalidasi".
Aliran alternatif	-
Kondisi akhir	Sistem berhasil merubah status berkas menjadi menjadi "sudah divalidasi".

Tabel 4.30 menjelaskan tentang proses dari *use case* batal validasi dan verifikasi berkas usulan. Batal validasi dan verifikasi berkas usulan adalah proses kepegawaian batal menyetujui berkas usulan telah sesuai ketentuan.

Tabel 4.30 Tabel *use case scenario* batal validasi dan verifikasi berkas usulan

Kode	SPJ_01_25
Tujuan	Aktor dapat merubah status berkas menjadi "belum divalidasi".
Aktor	Kepegawaian
Prakondisi	Aktor berada pada halaman detail berkas usulan dosen.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor memilih berkas dan menekan tombol batal validasi dan verifikasi pada tabel berkas 4 bidag. 2. Sistem mengubah status berkas administrasi di <i>database</i> menjadi "belum divalidasi".
Aliran alternatif	-
Kondisi akhir	Sistem berhasil merubah status berkas menjadi menjadi "belum divalidasi".

Tabel 4.31 menjelaskan tentang proses dari *use case* kirim penjelasan kesalahan. Kirim penjelasan kesalahan adalah proses kepegawaian memberitahu kesalahan usulan kepada dosen dengan mengirimkan email kepada dosen yang bersangkutan.

Tabel 4.31 Tabel *use case scenario* kirim penjelasan kesalahan

Kode	SPJ_01_26
Tujuan	Aktor dapat mengirimkan <i>email</i> penjelasan kesalahan.
Aktor	Kepegawaian
Prakondisi	Aktor berada pada halaman penjelasan kesalahan.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor memasukan pesan revisi kesalahan berkas dan menekan tombol <i>save</i>. 2. Sistem mengirimkan penjelasan kesalahan melalui <i>email</i>.

Aliran alternatif	-
Kondisi akhir	Sistem berhasil mengirim <i>email</i> penjelasan kesalahan.

Tabel 4.32 menjelaskan tentang proses dari *use case update* status usulan. *Update* status usulan adalah proses kepegawaian melakukan pemberitahuan secara berkala kepada dosen dimana usulanya sedang diproses.

Tabel 4.32 Tabel *use case scenario update* status usulan

Kode	SPJ_01_27
Tujuan	Aktor dapat mengubah status usulan pada database dan mengirim pemberitahuan lewat <i>email</i> .
Aktor	Kepegawaian
Prakondisi	Aktor berada pada halaman <i>update</i> status usulan.
Aliran utama	<ol style="list-style-type: none"> 1. Aktor mengisi data pada <i>form update</i> status usulan secara lengkap dan menekan tombol <i>save</i>. 2. Jika status usulan "selesai" maka menyimpan data status usulan dosen dan data jabatan terbaru dosen. 3. Sistem mengirim <i>email</i> dan menampilkan pesan "sudah dikirim".
Aliran alternatif	<ol style="list-style-type: none"> 2. Jika status usulan selain "selesai" maka menyimpan data status usulan dosen saja. 3. Sistem menampilkan pesan peringatan "sudah dikirim".
Kondisi akhir	Sistem berhasil mengubah status usulan pada database dan mengirim pemberitahuan lewat <i>email</i> .

Tabel 4.33 menjelaskan tentang proses dari *use case edit dupak*. *Edit dupak* adalah proses kepegawaian mengubah atau menambah data tim penilai, data ketua jurusan, data dekan/rektor dan data pendukung pengisian dupak.

Tabel 4.33 Tabel *use case scenario edit dupak*

Kode	SPJ_01_28
Tujuan	Aktor dapat mengubah mengubah data pada dupak.
Aktor	Kepegawaian
Prakondisi	Aktor berada pada halaman detil usulan
Aliran utama	<ol style="list-style-type: none"> 1. Mengisi data tim penilai, data ketua jurusan, data dekan/rektor dan data pendukung pengisian dupak secara lengkap. 2. Sistem memproses dan memasukan pada <i>database</i>.

Aliran alternatif	-
Kondisi akhir	Sistem berhasil mengubah data pada dupak dan perubahan ditampilkan pada dupak berekstensi pdf.

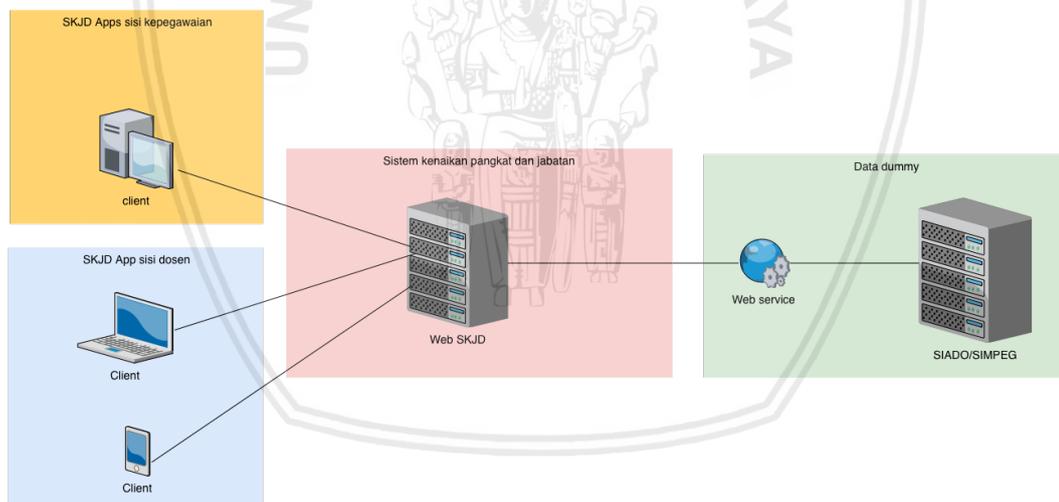


BAB 5 PERANCANGAN DAN IMPLEMENTASI

Bab 5 adalah tahap perancangan dan implementasi sistem berdasarkan kebutuhan yang di paparkan pada bab sebelumnya. Bab ini terdiri dari perancangan arsitektur, purwarupa, *database*, *class diagram framewrok*, *sequence diagram*, dan algoritme program.

5.1 Perancangan Arsitektur Sistem

Perancangan arsitektur sistem adalah tahap dalam melakukan perncangan sistem dari sudut pandang arsitektur teknologi. Tahap ini dipengaruhi oleh faktor kondisi lapangan. Pada kondisi lapangan sistem di UB menerapkan sistem monolitik. Untuk menjembatani antara data dari *database* SIADO dan SIMPEG maka digunakan data *dummy* yang diakses melalui *webservice* untuk menjembatani dengan Sistem kenaikan pangkat dan jabatan (SKJD Apps). Gambar 5.1 menjelaskan Aplikasi SKJD Apps mengakses data pendukung dari SIADO/SIMPEG melalui *webservice* dan menyimpan data yang memiliki hubungan dengan kenaikan pangkat dan/atau jabatan melalui basis datanya sendiri.



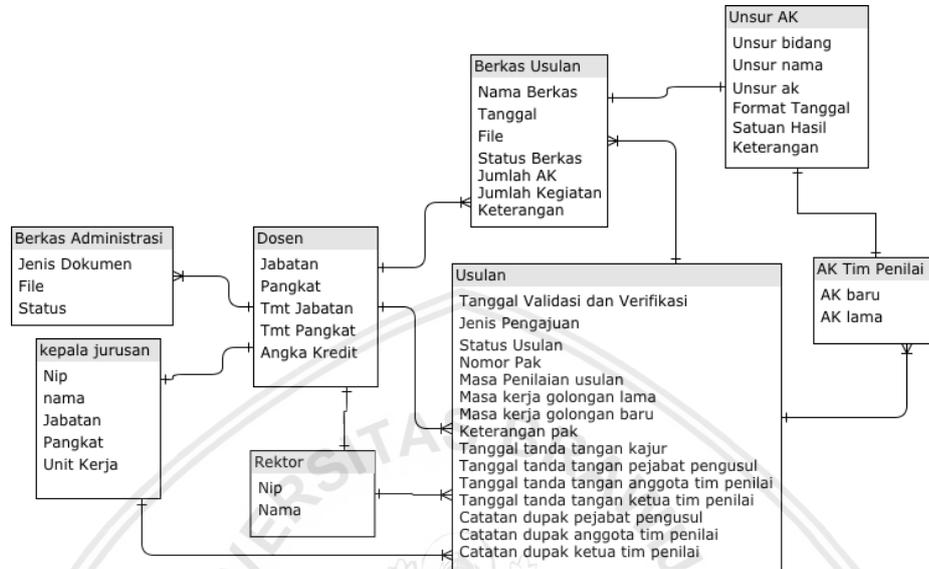
Gambar 5.1 Arsitektur sistem

5.2 Perancangan Basis Data

Perancangan basis data adalah tahap dalam melakukan implementasi pada sistem basis data. Perancangan basis data hanya pada lingkup sistem, dan tidak melakukan perancangan disisi sistem *dummy* yakni SIADO dan SIMPEG. Perancangan basis data menjelaskan *Conceptual data model* dan *Physiscal data model* sebagai acuan dalam implementasi basis data.

5.2.1 Conceptual Data Model

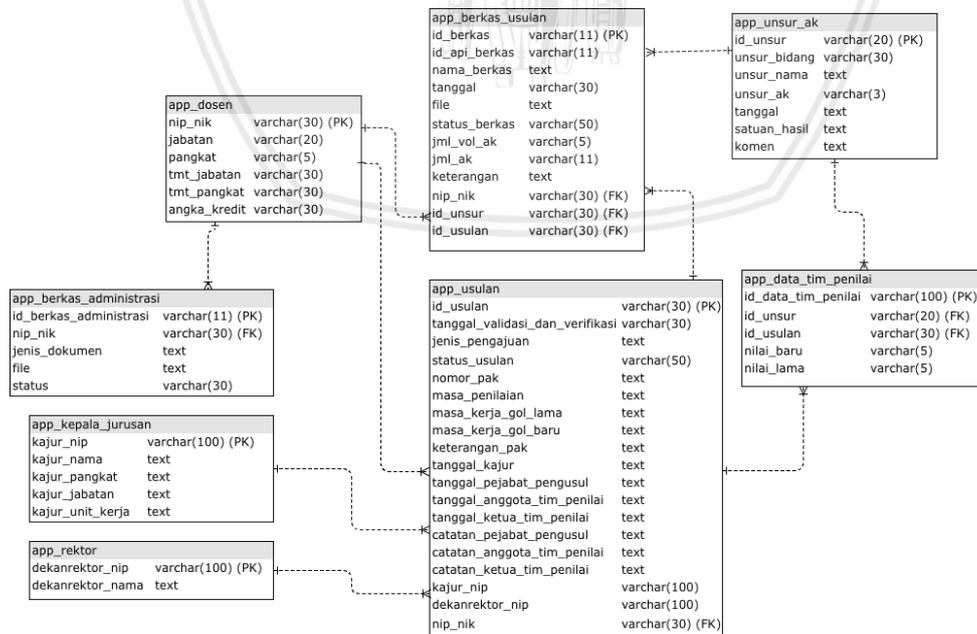
Conceptual data model (CDM) adalah sebuah representasi data dari objek kebutuhan. Gambar 5.2 merepresentasikan CDM pada perancangan basis data Sistem kenaikan pangkat dan jabatan dosen FILKOM.



Gambar 5.2 Conceptual Data Model

5.2.2 Physical Data Model

Setelah melakukan perancangan CDM maka dilakukan tahap *Physical Data Model* (PDM). Gambar 5.3 merepresentasikan PDM yang merepresentasikan basis data relasional yang sebenarnya.

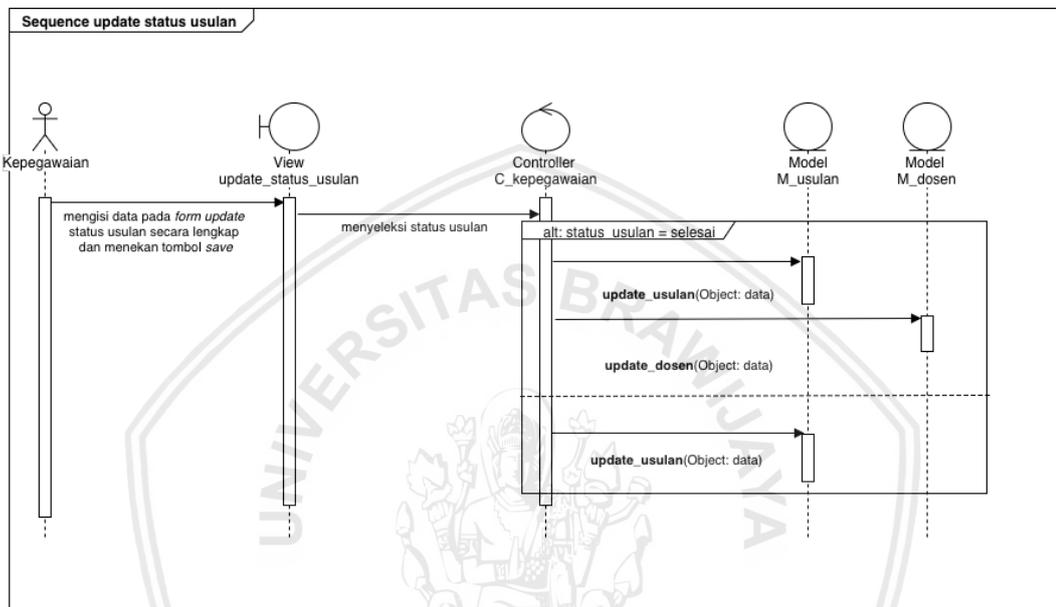


Gambar 5.3 Physical Data Model

5.3 Perancangan Sequence Diagram

5.3.1 Sequence diagram update status usulan

Gambar 5.4 merupakan *sequence diagram update* status usulan digunakan untuk kepegawaian memberikan informasi status usulan dosen. Jika status usulan dosen adalah selesai maka data dosen yang akan di *update* adalah status usulan, tmt terbaru dan jabatan/pangkat terbaru. Dan jika update status usulan selain selesai maka yang disimpan hanya status usulannya saja.

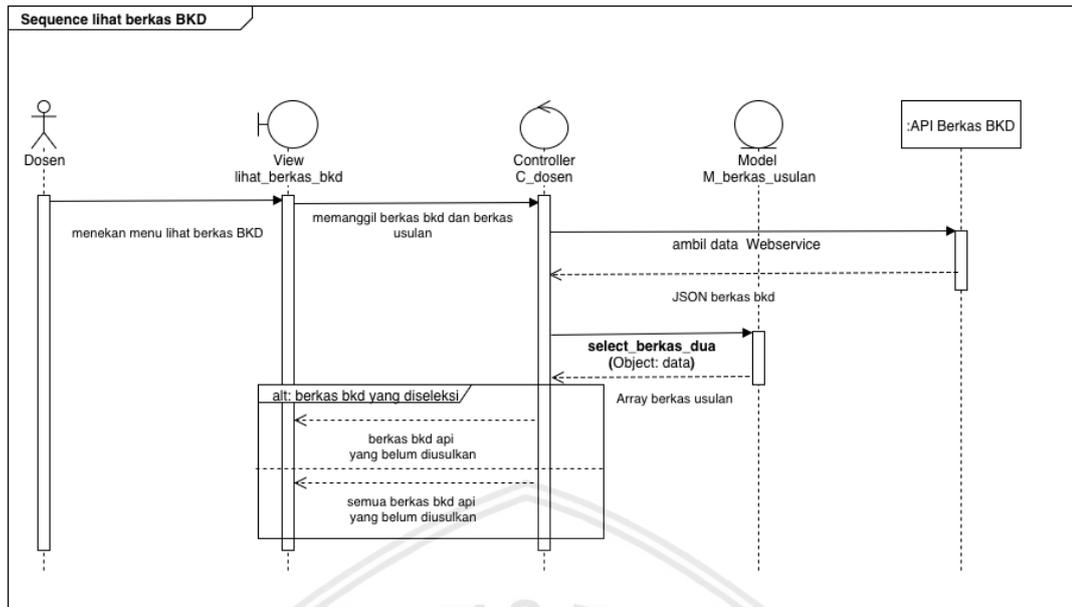


Gambar 5.4 Sequence Diagram update status usulan

5.3.2 Sequence diagram lihat berkas BKD

Gambar 5.5 merupakan *sequence diagram* lihat berkas BKD digunakan untuk dosen melihat berkas BKD dari *webservice*. Lihat berkas BKD menampilkan berkas BKD yang belum diusulkan. Jika sudah diusulkan maka tidak akan tampil pada tabel.

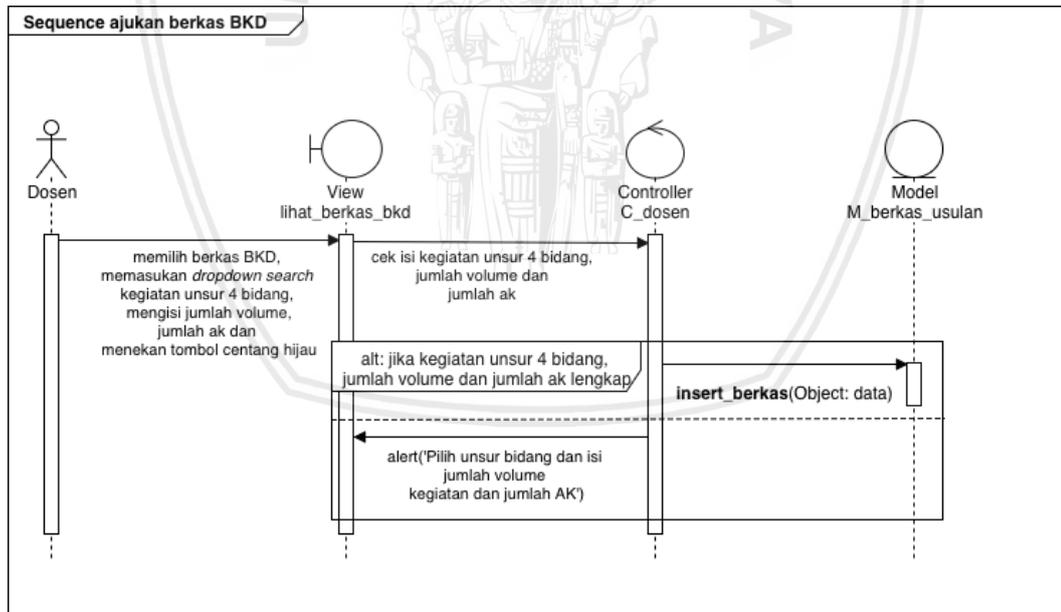




Gambar 5.5 Sequence Diagram lihat berkas BKD

5.3.3 Sequence diagram ajukan berkas BKD

Gambar 5.6 merupakan *sequence* diagram ajukan berkas BKD digunakan untuk dosen mengusulkan berkas yang terdapat pada *database* BKD SIADO.

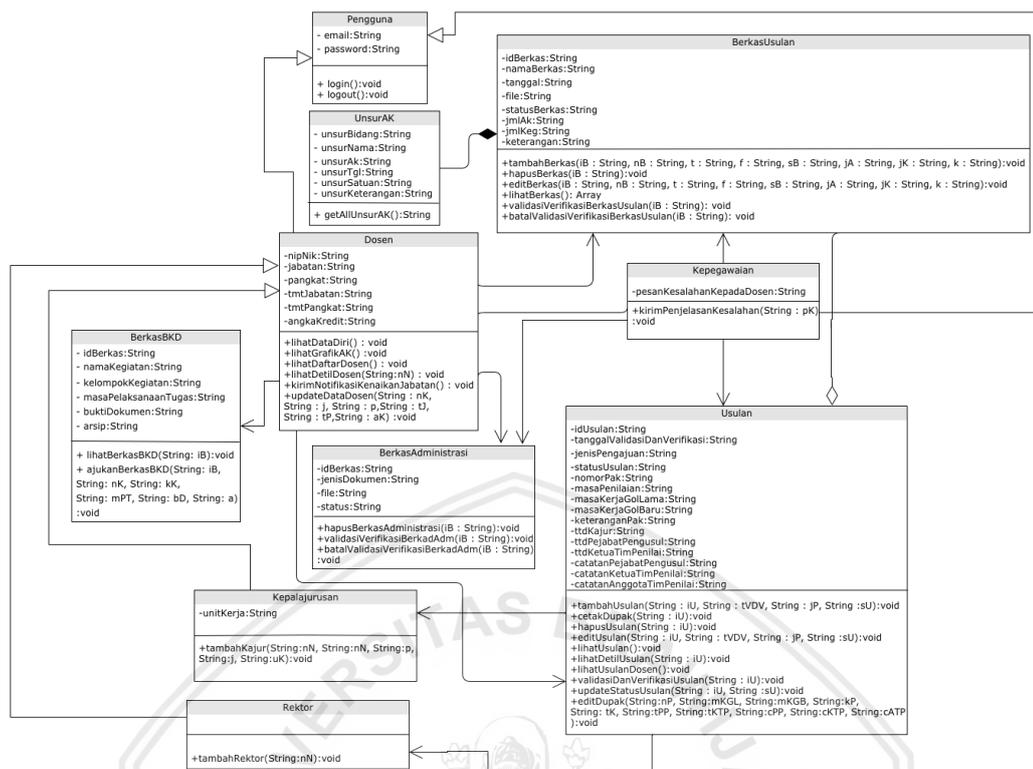


Gambar 5.6 Sequence Diagram ajukan berkas BKD

5.4 Perancangan Class Diagram

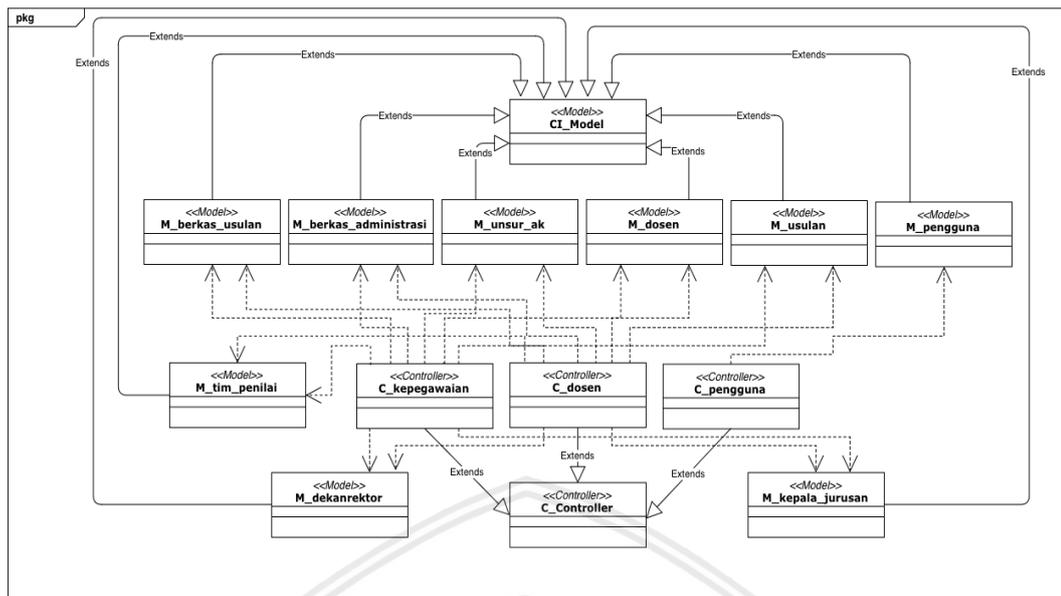
Perancangan class diagram adalah tahap dalam menentukan relasi antar objek dan informasi terperinci seperti atribut dan perilaku objek. Perancangan class diagram yang digambarkan gambar 5.7 digunakan sebagai dasar dalam pembuatan implementasi *class* diagram yang menggunakan *framework*.





Gambar 5.7 Perancangan Class Diagram

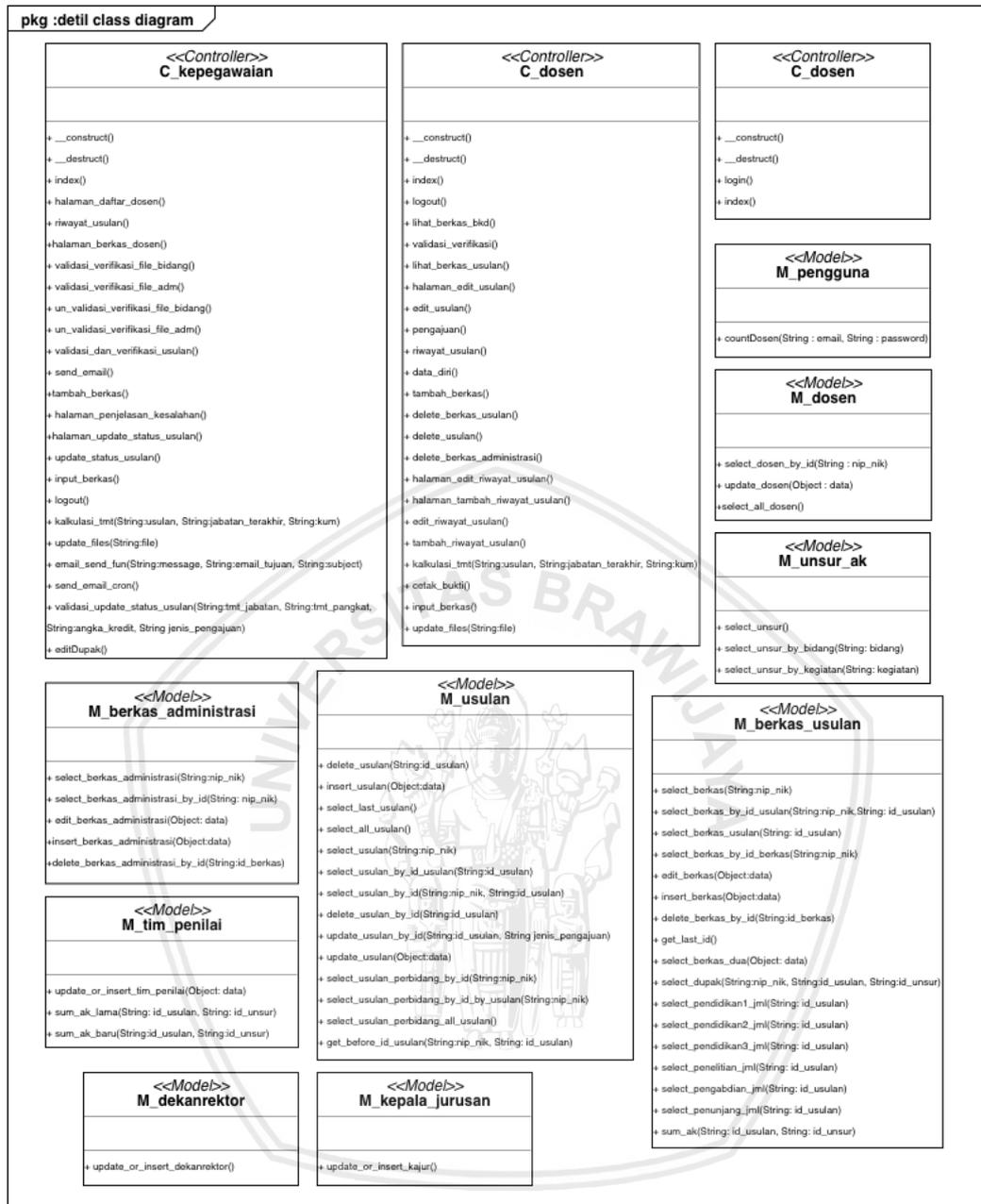
Selanjutnya yaitu penyesuaian Implementasi Class diagram, Impmentasi class diagram sistem adalah class diagram yang menyesuaikan dengan lingkungan sistem atau *framework*. Pada penelitian ini menggunakan *framework CodeIgniter* untuk merepresentasikan sistem. Gambar 5.8 mengimplementasikan class diagram untuk menggambarkan class diagram kebutuhan di proses menjadi class diagram sistem atau *framework*. Class diagram perancangan dijadikan sebagai acuan namun pada implementasi ada perubahan diantaranya pendefinisian nama fungsi dan atribut. Untuk implementasi php maka fungsi yang digunakan menggunakan *underscore* sebagai pemisah nama fungsi dan untuk atribut *framework* sudah memudahkan dengan langsung mengambil data dari *database* sehingga tidak perlu melakukan pendefinisian atribut lagi.



Gambar 5.8 Perancangan Implementasi Class Diagram

Implementasi class diagram dibuat karena dalam lingkungan *framework* Codeigniter sudah memberikan kemudahan dalam identifikasi atribut sehingga model tidak perlu mendefinisikan atribut karena langsung dapat melakukan *query* dari basis data tanpa mendefinisikan atribut. Hasil dari Implementasi Class diagram diatas akan didetilkkan dengan fungsi-fungsi yang digambarkan pada Gambar 5.9. Perbedaan *class* diagram pada perancangan dan implementasi adalah terletak pada nama fungsi. Fungsi yang diimplementasikan pada *controller* merujuk pada fungsi perancangan *class* diagram. Hal tersebut di satukan oleh sebuah class bernama *controller* yang memiliki tugas mengatur transaksi yang dilakukan pada sistem. Sementara model berguna sebagai jembatan antara logika sistem dengan basis data. *Controller* memiliki perilaku yang dimiliki oleh tiap aktor. Terdapat 3 aktor yaitu dosen, kepegawaian dan pengguna. Controller juga menjadi penghubung dengan antarmuka atau *view*. *View* di *Code igniter* dapat dibedakan berdasarkan aktor. Aktor dosen hanya memiliki akses untuk *view* dosen dan mengaturnya pada *controller* dosen. Aktor kepegawaian hanya memiliki akses untuk *view* kepegawaian dan mengaturnya pada *controller* kepegawaian. Sedangkan aktor pengguna hanya memiliki akses untuk *view* pengguna dan mengontrolnya di *controller* pengguna. Model memberikan luaran berupa data yang diolah oleh *controller* untuk ditampilkan pada *view*. *Method* pada model berupa sebuah *request query* yang nantinya akan mendapatkan luaran berupa data yang dikehendaki.





Gambar 5.9 Detil perancangan implementasi *Class Diagram*

5.5 Perancangan Algoritme Program

5.5.1 Algoritme *update* status usulan

Algoritme *update* status usulan diambil dari fungsi *update_status_usulan()* berdasarkan fungsi yang di definisikan pada *sequence* diagram *update* status usulan di tabel 5.1.

Tabel 5.1 Tabel algoritme *update* status usulan

Algoritme 1: Fungsi <i>Update</i> status usulan	
1	void <i>update_status_usulan</i> () {
2	initialize nip/nik, id usulan, status usulan, jenis



```

3 | pengajuan, email tujuan, tmt pangkat, tmt jabatan, AK
4 |
5 | data dosen = validasi usulan (tmt jabatan, tmt pangkat, AK,
6 | jenis pengajuan)
7 | data dosen add nip/nik
8 | data usulan add nip/nik, id usulan, status usulan
9 | if status usulan == "selesai" then
10 |   set data usulan to database
11 |   set data dosen to database
12 |   set status usulan and email tujuan to send email
13 | else
14 |   set data usulan to database
15 |   set status usulan and email tujuan to send email
16 | end if
17 | }

```

5.5.2 Algoritme lihat berkas BKD

Algoritme lihat berkas BKD diambil dari fungsi *lihat_berkas_bkd()* berdasarkan fungsi yang di definisikan pada *sequence* diagram lihat usulan di tabel 5.2.

Tabel 5.2 Tabel algoritme lihat berkas bkd

Algoritme 2: Fungsi lihat berkas bkd

```

1 | void lihat_berkas_bkd () {
2 |   inisialize nip/nik
3 |   berkasFromApi = get from api with nip/nik
4 |   initialize arrayFilter
5 |   berkasFromApp = get from database berkas usulan with nip/nik
6 |   if berkasFromApi > 0 then
7 |     loop x from 0 until end of index berkasFromApi
8 |     temporary = false
9 |     loop y from 0 until end of index berkasFromApp
10 |      if berkasAppApi at x == berkasAppApi at y then
11 |        temporary = true break
12 |      endIf
13 |    endLoop
14 |    if temporary == false then
15 |      push berkasFromApi at x to arrayFilter
16 |    endIf
17 |  endLoop
18 | else
19 |   arrayFilter = berkasFromApi
20 | endIf
21 | show view
22 | }

```

5.5.3 Algoritme Ajukan Berkas BKD

Algoritme ajukan berka BKD diambil dari fungsi *validasi_verifikasi()* berdasarkan fungsi yang di definisikan pada *sequence* diagram ajukan berkas BKD di tabel 5.3.

Tabel 5.3 Tabel algoritme ajukan berkas BKD

Algoritme 3: Fungsi ajukan berkas bkd

```

1 | void riwayat_usulan () {
2 |   inisialize nip/nik, id berkas, nama berkas, file, status
3 |   berkas, id unsur, tanggal, keterangan, jml vol, jml ak
4 |   inisialize array data berkas BKD
5 |

```



```

6   add nip/nik, id berkas, nama berkas, file, status berkas, id
7   unsur, tanggal, keterangan, jml vol, jml ak to data berkas
8   BKD
9
10  if id unsur not empty and jml vol not empty and jml ak not
11  empty then
12    set data berkas BKD to store database
13  else
14    alert "Pilih unsur bidang dan isi jumlah volume kegiatan
15    dan jumlah AK"
16  end if
17  redirect to lihat berkas bkd
18  }

```

5.6 Perancangan Purwarupa

5.6.1 Purwarupa Login

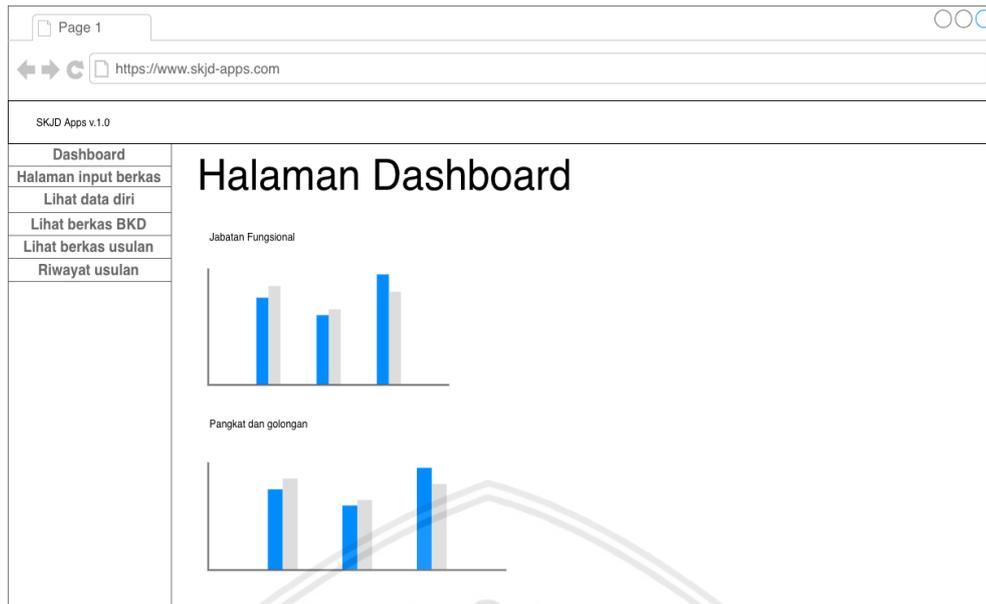
Purwarupa *login* digunakan pada fungsionalitas *login*. Purwarupa *login* digambarkan pada gambar 5.10 terdiri dari *email*, *password*, dan tombol *save*. Pengguna dari fungsionalitas ini adalah Dosen dan Kepegawaian.



Gambar 5.10 Purwarupa login

5.6.2 Purwarupa Dashboard

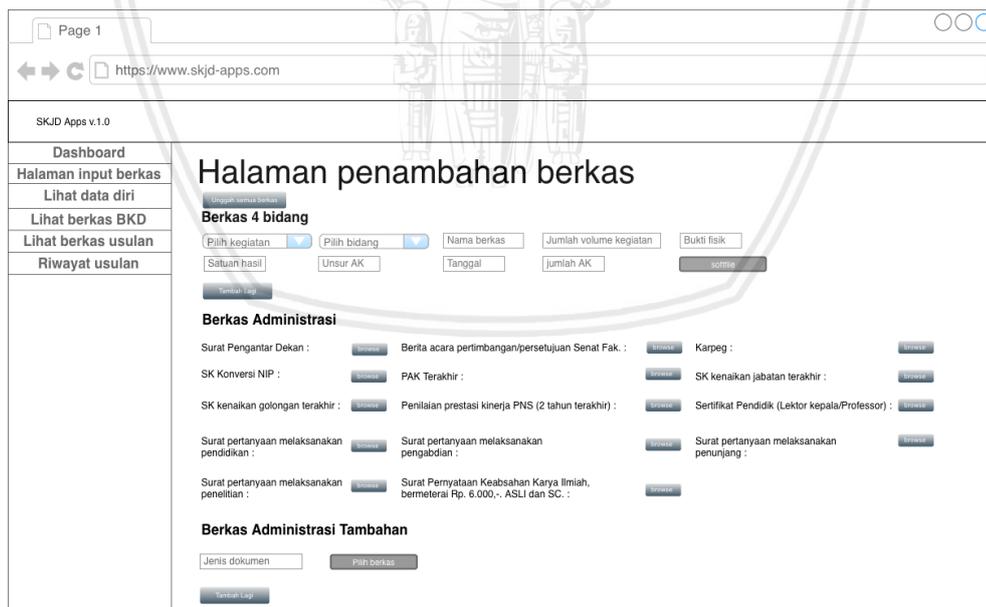
Purwarupa *dashboard* digunakan pada fungsionalitas *lihat grafik AK*. Purwarupa *dashboard* digambarkan pada gambar 5.11 terdiri dari grafik kekurangan AK Jabatan fungsional dan grafik kekurangan AK pangkat.



Gambar 5.11 Purwarupa dashboard

5.6.3 Purwarupa Penambahan berkas

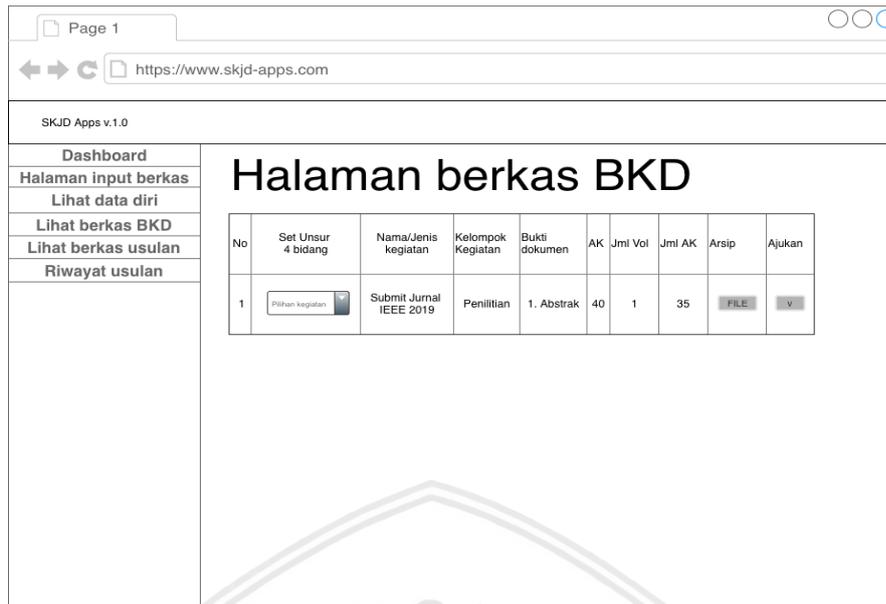
Purwarupa penambahan berkas digambarkan pada gambar 5.12 pada baris pertama untuk menambah berkas 4 bidang, administrasi, dan berkas administrasi tambahan.



Gambar 5.12 Purwarupa penambahan berkas

5.6.4 Purwarupa Berkas BKD

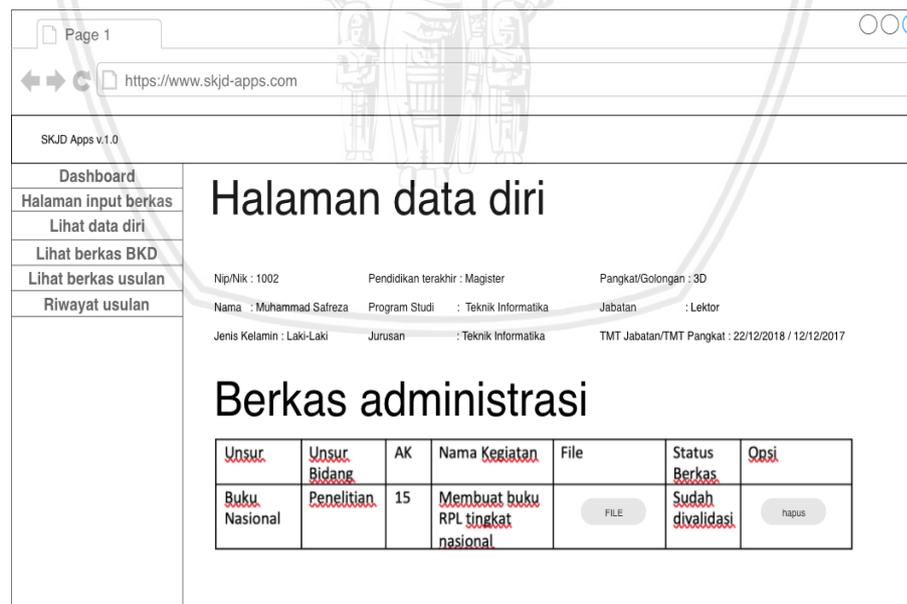
Purwarupa berkas BKD digunakan pada fungsionalitas lihat berkas BKD. Purwarupa berkas BKD digambarkan pada gambar 5.13 terdiri dari tabel yang tang berisi daftar berkas BKD yang diambil dari *webservice*.



Gambar 5.13 Purwarupa berkas BKD

5.6.5 Purwarupa Data diri

Purwarupa data diri digambarkan pada gambar 5.14 pada baris pertama menampilkan data diri pada baris kedua menampilkan daftar berkas administrasi.

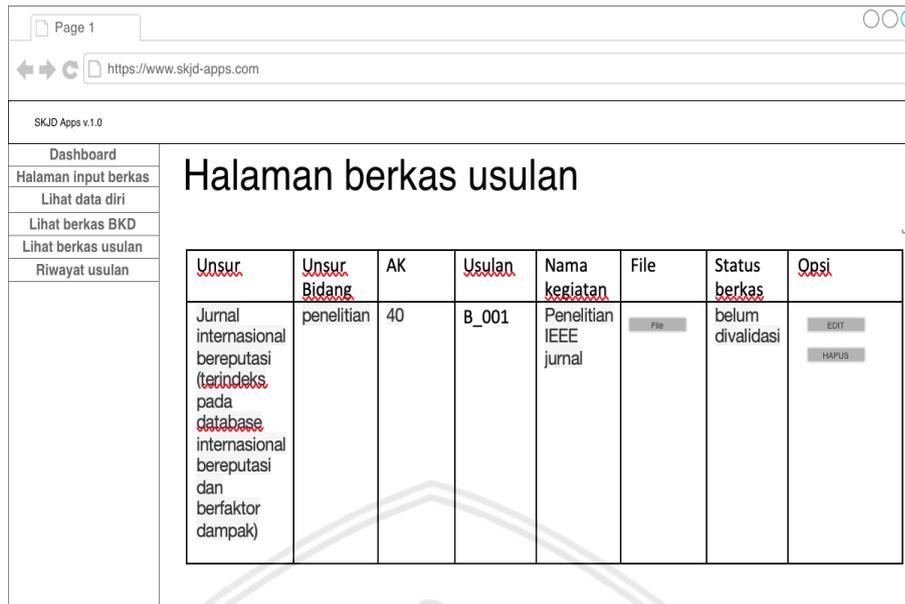


Gambar 5.14 Purwarupa data diri

5.6.6 Purwarupa Berkas usulan

Purwarupa berkas usulan digunakan pada fungsionalitas lihat berkas usulan. Purwarupa digambarkan pada gambar 5.15 terdiri dari tabel yang tang berisi daftar berkas usulan yang diambil dari BKD atau *input* dosen.

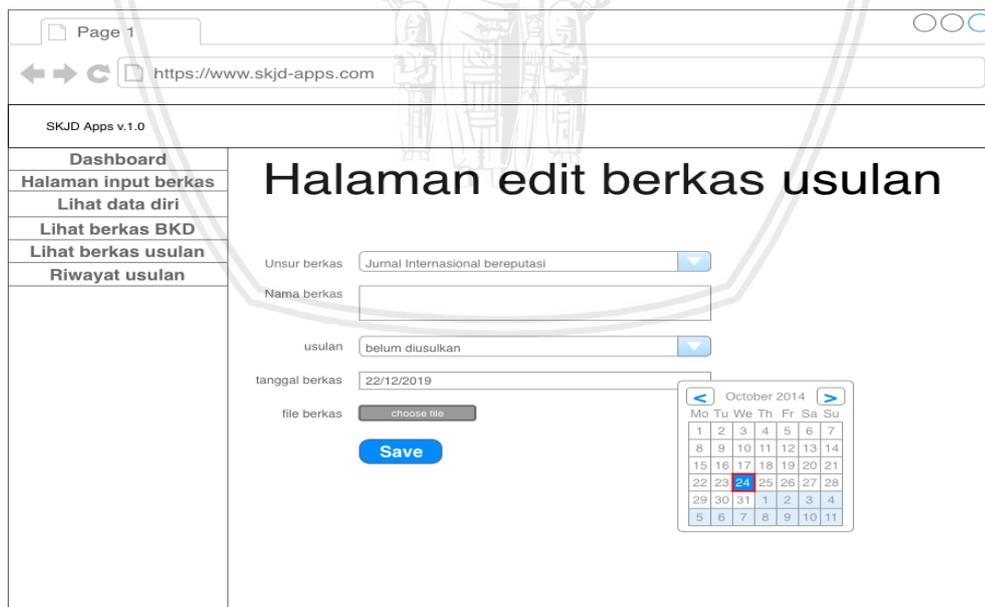




Gambar 5.15 Purwarupa berkas usulan

5.6.7 Purwarupa Edit berkas usulan

Purwarupa *edit* berkas usulan digunakan pada fungsionalitas *edit* berkas usulan. Purwarupa *edit* berkas usulan digambarkan pada gambar 5.16 yang digunakan untuk mengubah data berkas usulan.

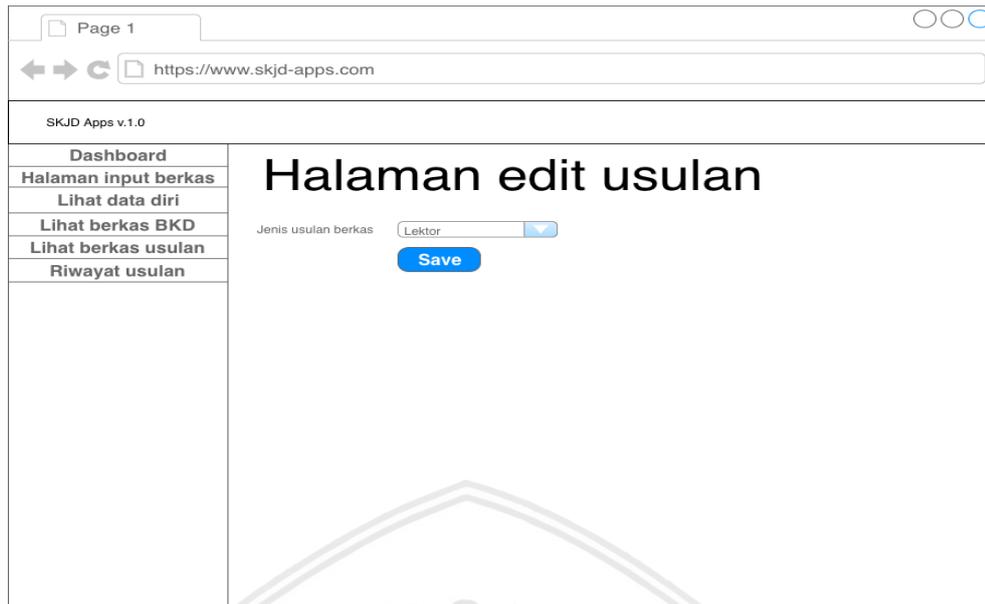


Gambar 5.16 Purwarupa Edit berkas usulan

5.6.8 Purwarupa Edit usulan

Purwarupa *edit* usulan digambarkan pada gambar 5.17 adalah representasi dari fungsionalitas *edit* usulan. *Edit* usulan dosen adalah mengubah jenis usulan seperti asisten ahli, lektor, dan lain-lain.

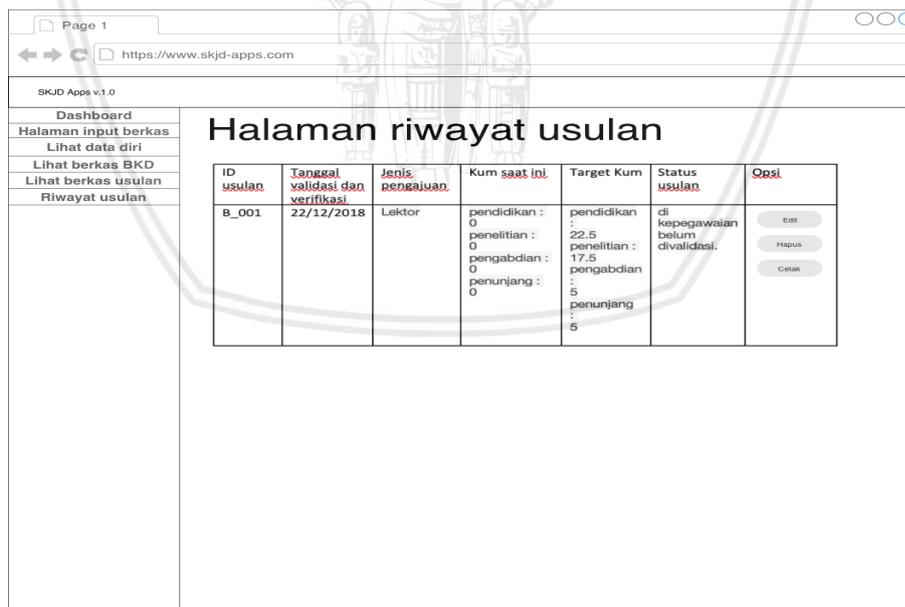




Gambar 5.17 Purwarupa edit usulan

5.6.9 Purwarupa Riwayat usulan

Purwarupa riwayat usulan digunakan pada fungsionalitas lihat usulan. Purwarupa riwayat usulan digambarkan pada gambar 5.18 yang digunakan untuk melihat daftar usulan.

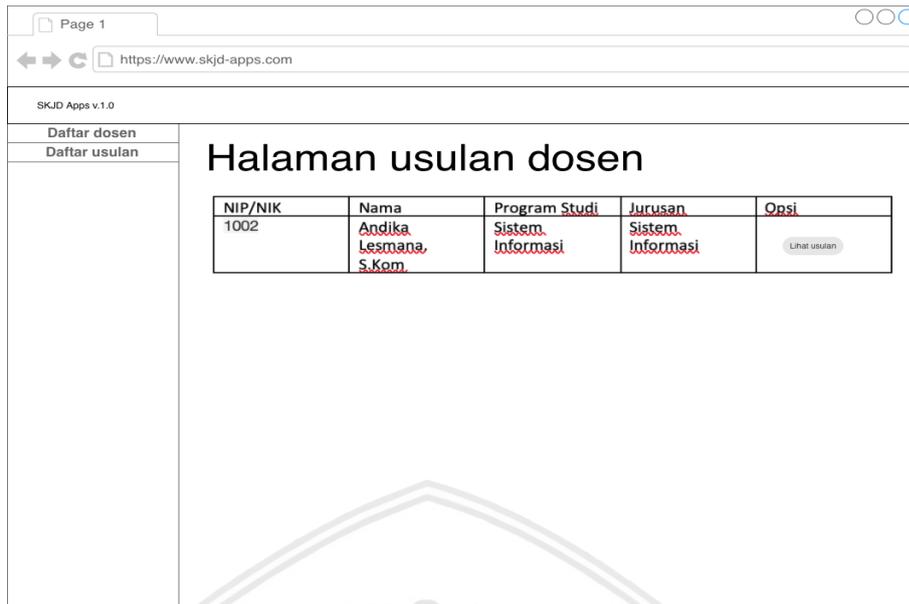


Gambar 5.18 Purwarupa riwayat usulan

5.6.10 Purwarupa Daftar dosen

Purwarupa daftar dosen digambarkan pada gambar 5.19 adalah representasi dari fungsionalitas lihat daftar dosen. daftar dosen adalah menampilkan seluruh data dosen yang diambil dari *webservice*.

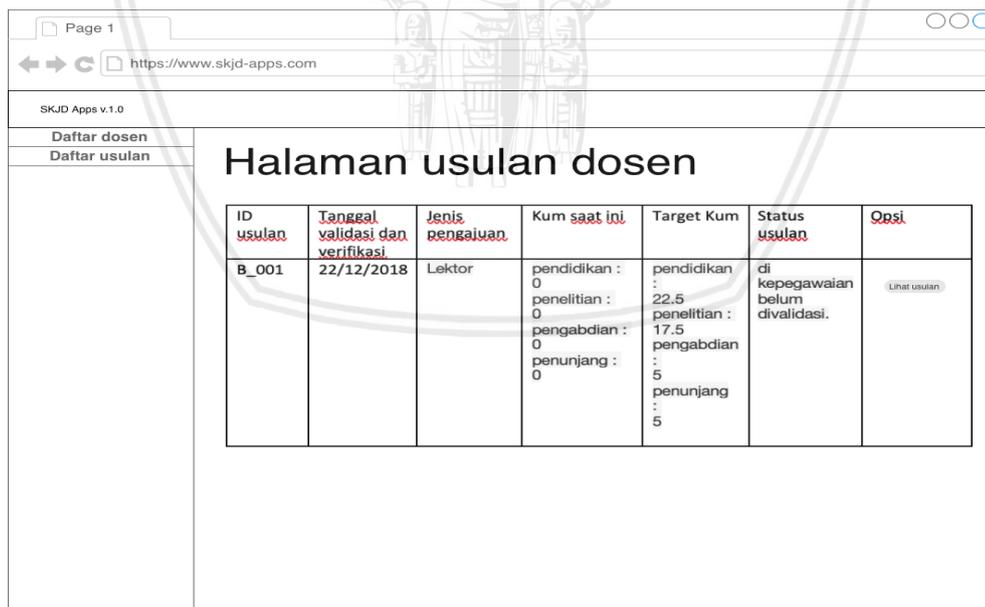




Gambar 5.19 Purwarupa daftar dosen

5.6.11 Purwarupa daftar usulan dosen

Purwarupa daftar usulan dosen digunakan pada fungsionalitas lihat usulan dosen. Purwarupa daftar usulan dosen digambarkan pada gambar 5.20 yang digunakan untuk melihat daftar usulan dosen.



Gambar 5.20 Purwarupa daftar usulan dosen

5.6.12 Purwarupa Detil berkas usulan dosen

Purwarupa detil berkas usulan dosen digambarkan pada gambar 5.21 adalah representasi dari fungsionalitas lihat detil berkas usulan. Detil berkas usulan dosen adalah menampilkan data diri, data administrasi dan berkas usulan.

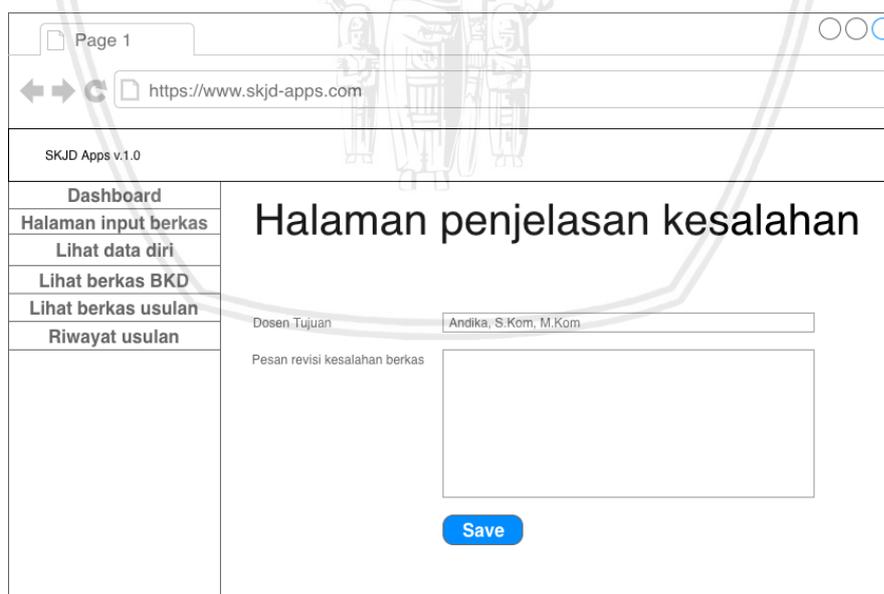




Gambar 5.21 Purwarupa detail berkas usulan dosen

5.6.13 Purwarupa Penjelasan kesalahan

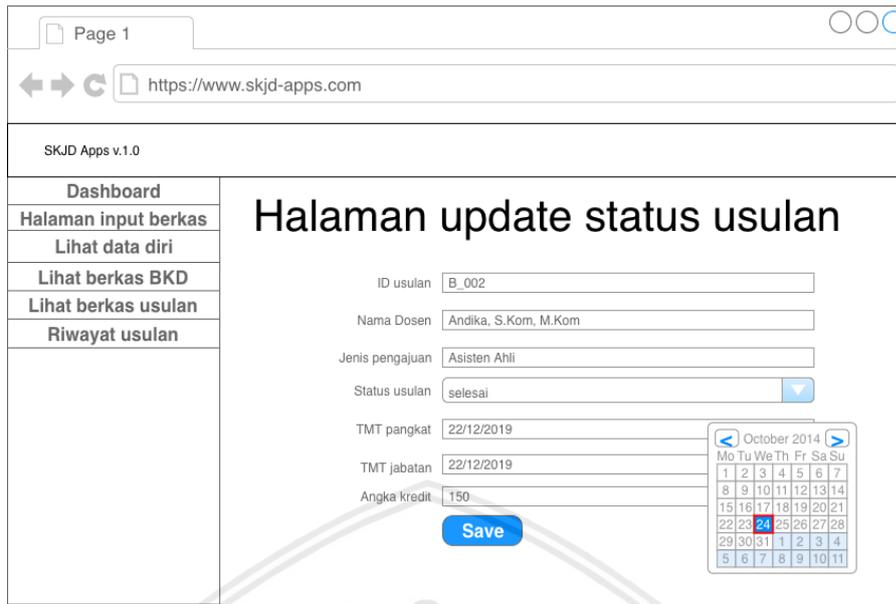
Purwarupa penjelasan kesalahan digunakan pada fungsionalitas kirim penjelasan kesalahan. Purwarupa penjelasan kesalahan digambarkan pada gambar 5.22 yang digunakan untuk mengirim penjelasan kesalahan.



Gambar 5.22 Purwarupa penjelasan kesalahan

5.6.14 Purwarupa Update status usulan

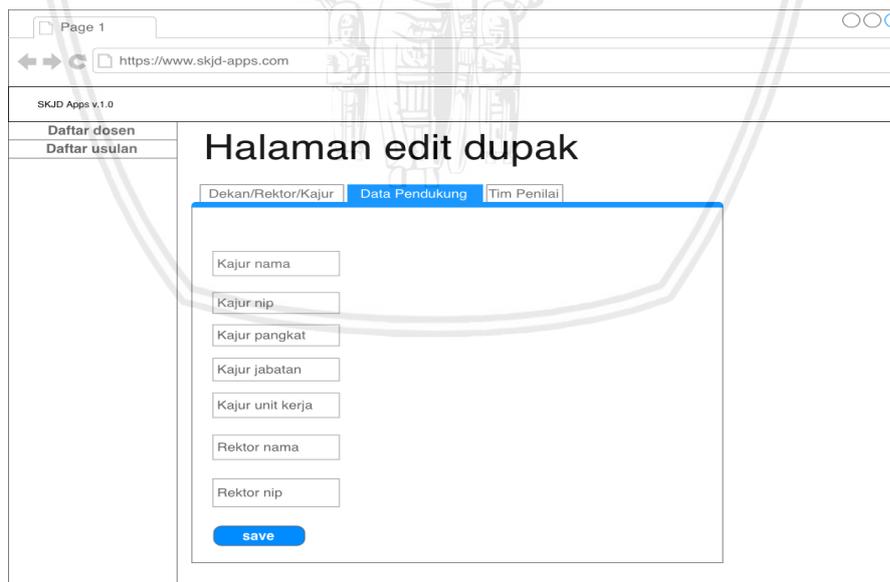
Purwarupa *update* status usulan digambarkan pada gambar 5.23 adalah representasi dari fungsionalitas *update* usulan. *Update* status usulan adalah menampilkan mengirim informasi terbaru terkait usulan kepada dosen.



Gambar 5.23 Purwarupa update status usulan

5.6.15 Purwarupa *Edit* dupak

Purwarupa *edit* dupak digambarkan pada gambar 5.24 adalah representasi dari fungsionalitas *Edit* dupa. *edit* dupak adalah mengisi data terkait kelengkapan dupak oleh kepegawaian.

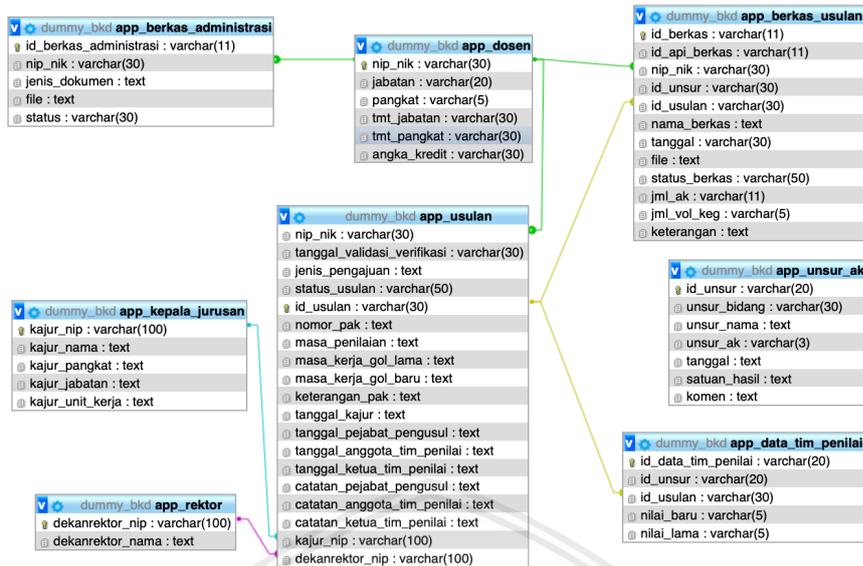


Gambar 5.24 Purwarupa edit dupak

5.7 Implementasi Basis Data

Implementasi basis data digunakan untuk implementasi perancangan dan menerapkan pada sistem. Gambar 5.25 Implementasi menggunakan basis data *Mysql* pada *server*.





Gambar 5.25 implementasi basis data

5.8 Implementasi Kode Program

Implementasi kode program menggunakan Bahasa pemrograman PHP. Pemrograman PHP adalah Bahasa pemrograman framework Code Igniter yang selanjutnya pada penelitian ini sistem yang dibuat menggunakan PHP versi 7. Kode program yang didefinisikan selanjutnya di tulis pada controller sesuai dengan perilaku aktor.

5.8.1 Kode program *update* status usulan

Kode program update status usulan diimplementasikan menjadi fungsi *update_status_usulan()* berdasarkan fungsi yang di definisikan pada perancangan algoritme update status usulan di tabel 5.4.

Tabel 5.4 Tabel kode program update status usulan

Kode Program 1: Fungsi Update status usulan	
1	public function update_status_usulan() {
2	\$nip_nik = \$this->input->post("nip_nik");
3	\$id_usulan = \$this->input->post("id_usulan");
4	\$status_usulan = \$this->input->post("status_usulan");
5	\$jenis_pengajuan = \$this->input->post("jenis_pengajuan");
6	\$email_tujuan = \$this->input->post("email_tujuan");
7	\$tmt_pangkat = \$this->input->post("tmt_pangkat");
8	\$tmt_jabatan = \$this->input->post("tmt_jabatan");
9	\$angka_kredit = \$this->input->post("angka_kredit");
10	\$data_dosen = array();
11	\$data_dosen['id_usulan'] = \$id_usulan;
12	\$data_dosen['status_usulan'] = \$status_usulan;
13	\$data = array(
14	'id_usulan' => \$id_usulan,
15	'status_usulan' => \$status_usulan,
16);
17	\$data_dosen = validasi_update_status_usulan(\$tmt_jabatan,

```

18     $tmt_pangkat,$angka_kredit,$jenis_pengajuan);
19
20     if ($status_usulan == "selesai"){
21         $this->M_dosen->update_dosen($data_dosen);
22         $this->email_send_fun('Status berkas anda
23         adalah '.$status_usulan, $email_tujuan, 'INFO TERBARU
24         USULAN PANGKAT DAN JABATAN');
25     }else{
26         $this->M_usulan->update_usulan($data);
27         $this->email_send_fun('Status berkas anda
28         adalah '.$status_usulan, $email_tujuan,
29         'INFO TERBARU USULAN PANGKAT DAN JABATAN');
30     }

```

5.8.2 Kode program lihat berkas bkd

Kode program lihat berkas bkd diimplementasikan di fungsi *lihat_berkas_bkd()* berdasarkan fungsi yang di definisikan pada perancangan *algoritme* lihat usulan di tabel 5.5.

Tabel 5.5 Tabel kode program lihat berkas bkd

Kode program 2: Fungsi lihat berkas bkd	
1	public function lihat_berkas_bkd () {
2	\$id = \$this->session->userdata("nip_nik");
3	\$global_function = new Custom();
4	\$unsur = \$this->M_unsur_ak->select_unsur()->result();
5	\$buffer_berkas=\$global_function->
6	curl_get('http://api.skjd-apps.com/api/berkas/id/'. \$id);
7	\$berkas_from_api = json_decode(\$buffer_berkas);
8	\$afterFilter = array();
9	\$berkas_from_app=\$this->M_berkas_usulan->
10	select_berkas_dua(\$id)->result();
11	if(count(\$berkas_from_app)>0) {
12	for (\$x=0; \$x<count(\$berkas_from_api);\$x++) {
13	\$temp = false;
14	for (\$y=0; \$y<count(\$berkas_from_app);\$y++) {
15	if(\$berkas_from_api[\$x]->
16	id_berkas==\$berkas_from_app[\$y]->id_api_berkas){
17	\$temp = true; break;
18	}
19	}
20	if (\$temp == false){
21	array_push(\$afterFilter,\$berkas_from_api[\$x]);
22	}
23	}
24	}else{ \$afterFilter = \$berkas_from_api; }
25	\$this->data['data_berkas'] = \$afterFilter;
26	\$this->data['data_berkas_app'] = \$berkas_from_app;
27	\$this->data['unsur'] = \$unsur;
28	\$this->data["title"] = \$this->variable["title"];
29	\$this->data['content']=\$this->load->
30	view('dosen/V_lihat_berkas_bkd', \$this->data, true);
31	\$this->load->view("V_dashboard_dosen", \$this->data);
32	}

5.8.3 Kode program Ajukan Berkas BKD

Kode program ajukan berka BKD diimplementasikan pada fungsi *validasi_verifikasi()* berdasarkan fungsi yang di definisikan pada algoritme ajukan berkas BKD di tabel 5.6.

Tabel 5.6 Tabel kode program ajukan berkas BKD

Kode program 3: Fungsi ajukan berkas bkd	
1	public function validasi_verifikasi(){
2	\$nip_nik = \$this->input->post('nip_nik');
3	\$id_berkas = \$this->input->post('id_berkas');
4	\$nama_berkas = \$this->input->post('nama_berkas');
5	\$file = \$this->input->post('file');
6	\$status_berkas = \$this->input->post('status_berkas');
7	\$id_unsur = \$this->input->post('idunsur');
8	\$tanggal = \$this->input->post('waktu');
9	\$keterangan = \$this->input->post('keterangan');
10	\$jml_vol = \$this->input->post('jml_vol');
11	\$jml_ak = \$this->input->post('jml_ak');
12	\$data = array(
13	'nip_nik' => \$nip_nik,
14	'nama_berkas' => \$nama_berkas,
15	'id_api_berkas' => \$id_berkas,
16	'id_unsur' => \$id_unsur,
17	'status_berkas' => \$status_berkas,
18	'file' => \$file,
19	'tanggal' => \$tanggal,
20	'keterangan' => \$keterangan,
21	'jml_vol_keg' => \$jml_vol,
22	'jml_ak' => \$jml_ak,
23);
24	if (!empty(\$id_unsur) && !empty(\$jml_vol) &&
25	!empty(\$jml_ak)) {
26	\$this->M_berkas_usulan->insert_berkas(\$data);
27	} else { echo "<script>alert('Pilih unsur bidang dan isi
28	jumlah volume kegiatan dan jumlah AK');</script>";
29	} redirect(base_url() . "dosen/lihat-berkas-bkd");
30	}

5.9 Implementasi Antarmuka

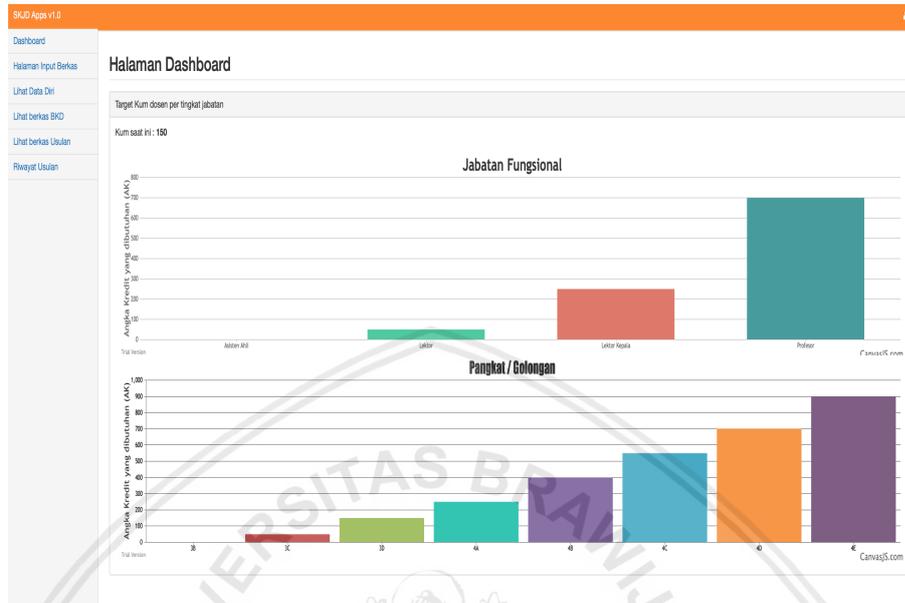
5.9.1 Implementasi antarmuka Login

Implementasi *login* digunakan pada fungsionalitas *login*. Implementasi *login* digambarkan pada gambar 5.26 terdiri dari *email*, *password*, dan tombol *save*.

Gambar 5.26 Implementasi antarmuka login

5.9.2 Implementasi antarmuka Dashboard

implementasi *dashboard* digunakan pada fungsionalitas *lihat grafik AK*. implementasi *dashboard* digambarkan pada gambar 5.27.



Gambar 5.27 Implementasi antarmuka dashboard

5.9.3 Implementasi Penambahan berkas

Implementasi penambahan berkas digambarkan pada gambar 5.28 pada baris pertama untuk menambah berkas 4 bidang, administrasi, dan berkas administrasi tambahan.

Berkas Pendukung

Unsur Bidang: Kegiatan: Satuan Hasil: Unsur AK:

Nama berkas: Tanggal: Jumlah Volume Kegiatan: Jumlah AK:

Keterangan/Bukti fisik: Sertifikat Berkas:

Berkas Administrasi

Surat Pengantar Dekan	Berita acara pertimbangan/persetujuan Senat Fak.	Karpeg
SK Konversi NIP	PKW Terakhr	SK kenaikan jabatan terakhir
SK kenaikan golongan terakhir	Penilaian prestasi kinerja PNS (2 tahun terakhir)	Sertifikat Pendidik (Lektor kepala/Professor)
Surat pertanyaan melaksanakan pendidikan	Surat pertanyaan melaksanakan pengabdian	Surat pertanyaan melaksanakan perijinan
Surat pertanyaan melaksanakan penelitian	Surat Pernyataan Keabsahan Karya Ilmiah, bernominal Rp. 6.000,-, ASU dan SC.	

Berkas Administrasi Tambahan

Jenis Dokumen: Berkas (PDF):

Gambar 5.28 Implementasi antarmuka penambahan berkas

5.9.4 Implementasi antarmuka Berkas BKD

Implementasi berkas BKD digunakan pada fungsionalitas lihat berkas BKD. Implementasi antarmuka berkas BKD digambarkan pada gambar 5.29 terdiri dari tabel yang tang berisi daftar berkas BKD yang diambil dari *webservice*.

No	Set Unsur 4 Bidang	Nama/Jenis Kegiatan	Kelompok Kegiatan	Bukti Dokumen	AK	Jml Vol	Jml AK	Arsip	Ajukan
1	Pilih kegiatan	Mengajar ASD TIF	pendidikan	1. SK		0	0	file	Y
2	Pilih kegiatan	Submit Jurnal IEEE 2019	penelitian	1. Abstrak		0	0	file	Y
3	Pilih kegiatan	E Jurnal SIET 2018	penelitian	1. Abstrak		0	0	file	Y

Gambar 5.29 Implementasi antarmuka berkas BKD

5.9.5 Implementasi antarmuka Data diri

Implementasi data diri digambarkan pada gambar 5.30 pada baris pertama menampilkan data diri pada baris kedua menampilkan daftar berkas administrasi.

ID Berkas Administrasi	Jenis Berkas	Status	File	Opsi
C.005	Surat Pengantar Dekan	belum divalidasi	File	Hapus

Gambar 5.30 Implementasi antarmuka data diri

5.9.6 Implementasi antarmuka Berkas usulan

Implementasi berkas usulan digambarkan pada gambar 5.31 terdiri dari tabel yang tang berisi daftar berkas usulan yang diambil dari BKD atau *input* dosen.

The screenshot shows the 'Halaman Data diri' (Self-Data Page) in the SKID Apps v1.0 interface. It includes a sidebar menu with options like 'Dashboard', 'Halaman Input Berkas', 'Lihat Data Diri', 'Lihat berkas BKD', 'Lihat berkas Usulan', and 'Riwayat Usulan'. The main content area is divided into sections: 'Data dosen', 'Data Diri', and 'Berkas administrasi'. The 'Data Diri' section displays personal details for a user with NIP/Nik 1002, name Muhammad Saheza, S.Kom, M.Kom, and a position of Asisten Ahli. The 'Berkas administrasi' section features a table with columns for ID, Jenis Berkas, Status, File, and Opsi. A single entry is visible with ID C.0005, Jenis Berkas Surat Pengantar Dekan, and Status belum divalidasi. A search bar and pagination controls are also present.

ID Berkas Administrasi	Jenis Berkas	Status	File	Opsi
C.0005	Surat Pengantar Dekan	belum divalidasi	File	Hapus

Gambar 5.31 Implementasi antarmuka berkas usulan

5.9.7 Implementasi antarmuka Edit berkas usulan

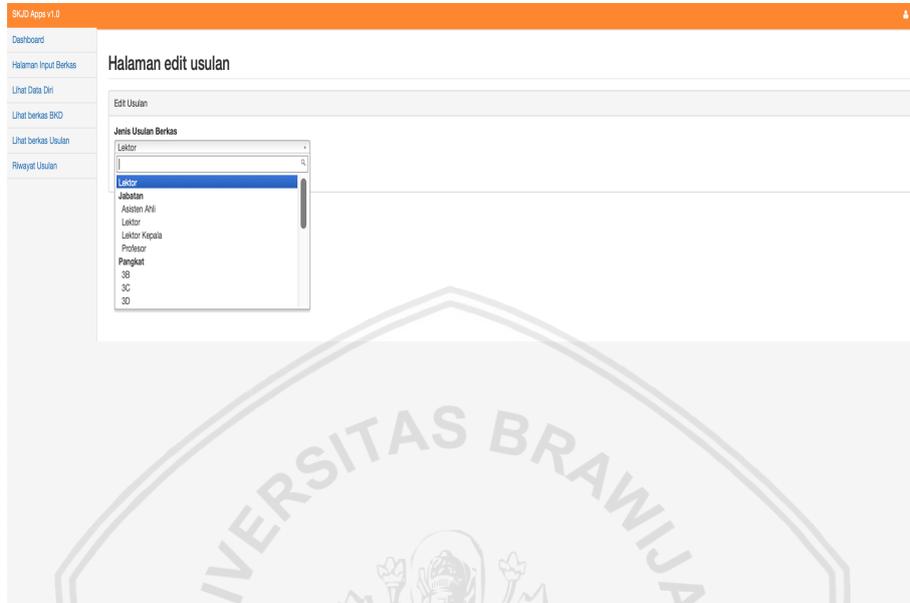
Implementasi *edit* berkas usulan digunakan pada fungsionalitas *edit* berkas usulan. Implementasi *edit* berkas usulan digambarkan pada gambar 5.32 yang digunakan untuk mengubah data berkas usulan.

The screenshot shows the 'Halaman edit berkas usulan' (Edit Proposal File Page) in the SKID Apps v1.0 interface. It includes a sidebar menu with options like 'Dashboard', 'Halaman Input Berkas', 'Lihat Data Diri', 'Lihat berkas BKD', 'Lihat berkas Usulan', and 'Riwayat Usulan'. The main content area is divided into sections: 'Edit Berkas Usulan', 'Unsur Berkas', 'Nama Berkas', 'Usulan', and 'File Berkas'. The 'Unsur Berkas' section contains a text area with the text '(perkuliahan/tutorial/pekerjaan praktikum dan membimbing mngajar serta menyelenggarakan pendidikan di laboratorium, praktik keguruan, bengkel/studio/kebabaroneban/teknologi pengajaran dan praktik lapangan (setiap semester) b...'. The 'Nama Berkas' section has a text input field containing 'Mengajar ASD 2013'. The 'Usulan' section has a dropdown menu set to 'belum divalidasi' and a date input field containing '13/02/2019'. The 'File Berkas' section has a text input field containing 'no file selected' and a 'Save' button.

Gambar 5.32 Implementasi antarmuka Edit berkas usulan

5.9.8 Implementasi antarmuka Edit usulan

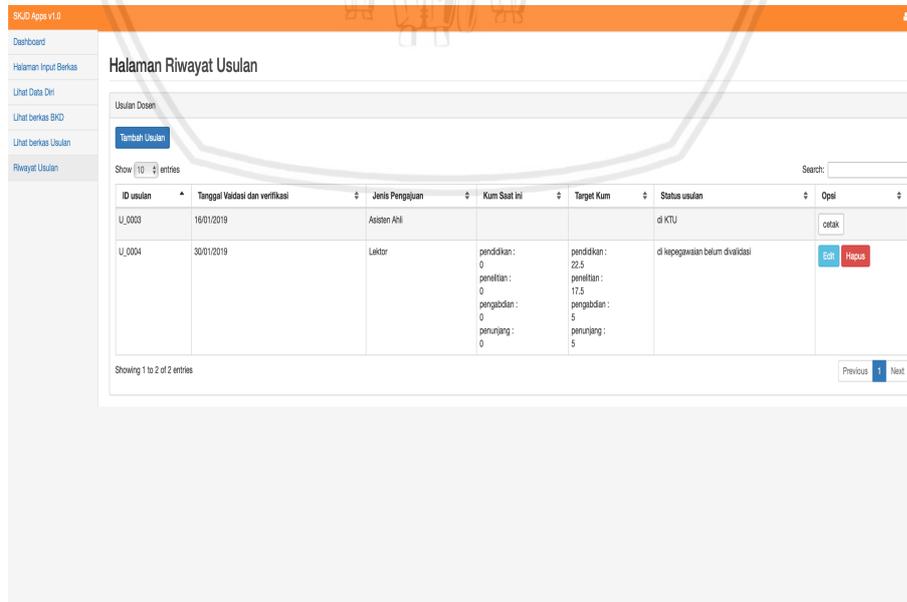
Implementasi *edit* usulan digambarkan pada gambar 5.33 adalah representasi dari fungsionalitas *edit* usulan. *Edit* usulan dosen adalah mengubah jenis usulan seperti asisten ahli, lektor, dan lain-lain.



Gambar 5.33 Implementasi antarmuka edit usulan

5.9.9 Implementasi antarmuka Riwayat usulan

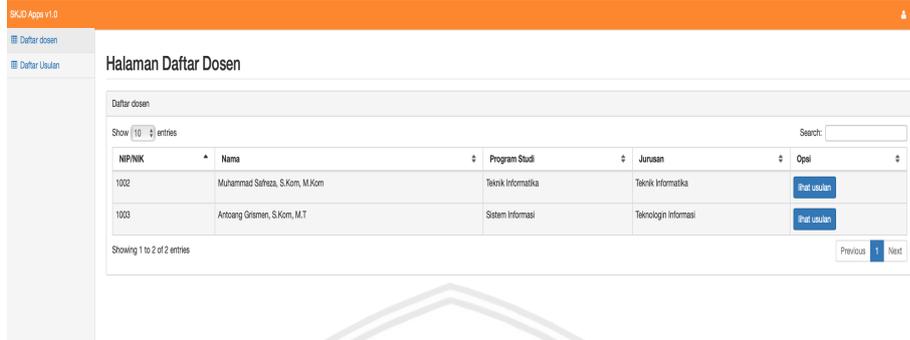
Implementasi riwayat usulan digunakan pada fungsionalitas lihat usulan. Implementasi riwayat usulan digambarkan pada gambar 5.34 yang digunakan untuk melihat daftar usulan.



Gambar 5.34 Implementasi antarmuka riwayat usulan

5.9.10 Implementasi Daftar dosen

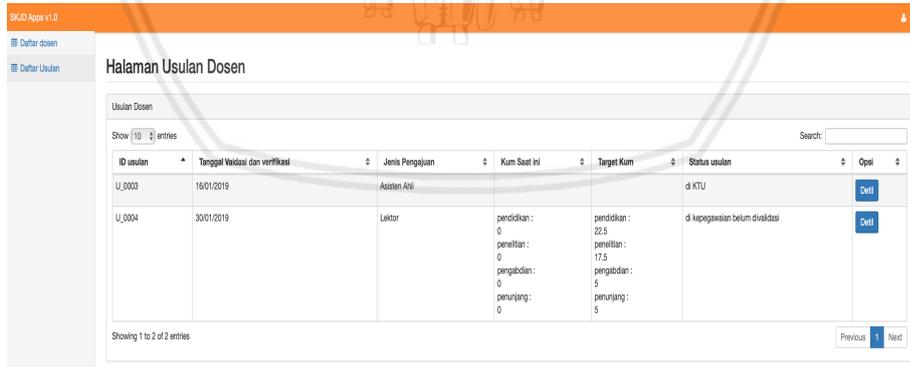
Implementasi daftar dosen digambarkan pada gambar 5.35 adalah representasi dari fungsionalitas lihat daftar dosen. daftar dosen adalah menampilkan seluruh data dosen yang diambil dari *webservice*.



Gambar 5.35 Implementasi antarmuka daftar dosen

5.9.11 Implementasi antarmuka daftar usulan dosen

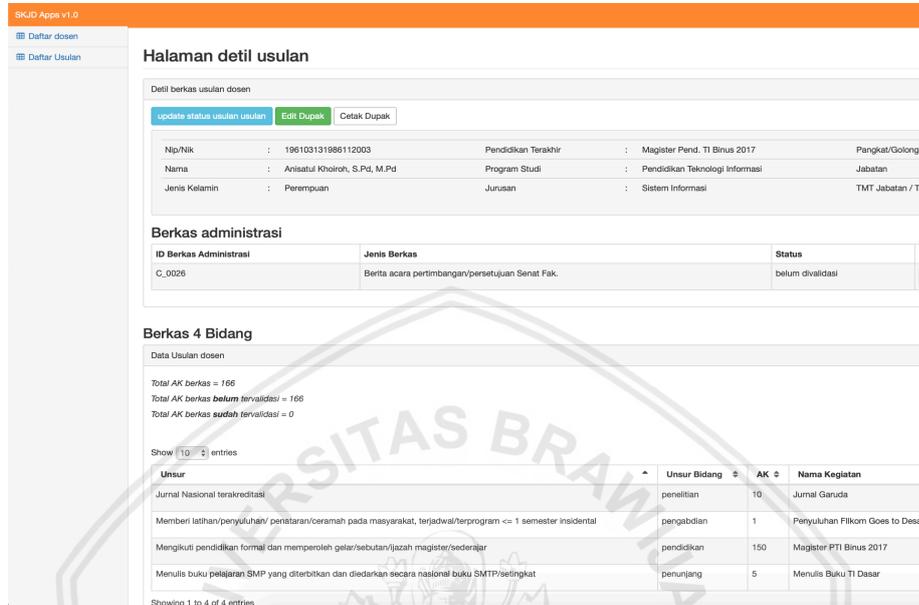
Implementasi daftar usulan dosen digunakan pada fungsionalitas lihat usulan dosen. Implementasi daftar usulan dosen digambarkan pada gambar 5.36 yang digunakan untuk melihat daftar usulan dosen.



Gambar 5.36 Implementasi antarmuka daftar usulan dosen

5.9.12 Implementasi antarmuka Detil berkas usulan dosen

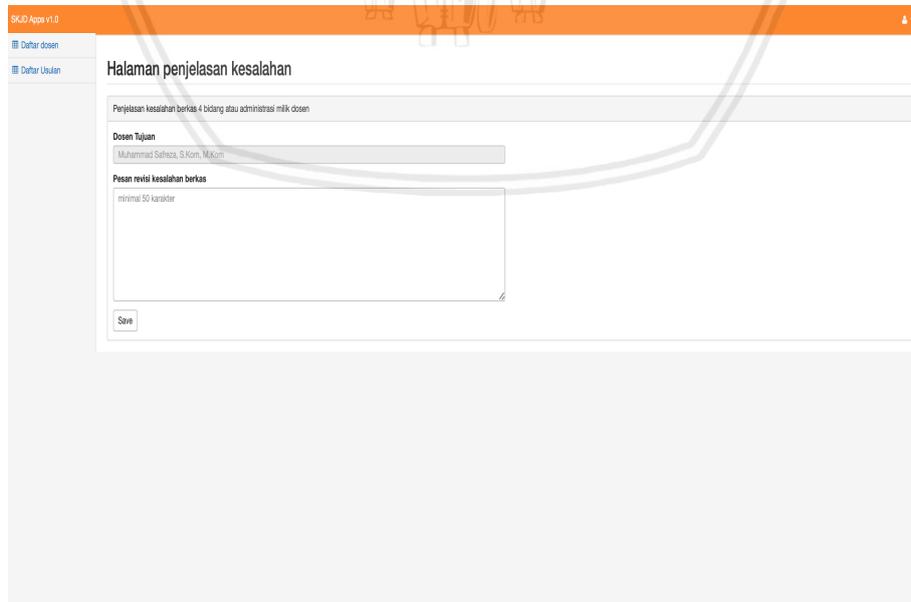
Implementasi detil berkas usulan dosen digambarkan pada gambar 5.37 adalah representasi dari fungsionalitas lihat detil berkas usulan. Detil berkas usulan dosen adalah menampilkan data diri, data administrasi dan berkas usulan.



Gambar 5.37 Implementasi antarmuka detil berkas usulan dosen

5.9.13 Implementasi antarmuka Penjelasan kesalahan

Implementasi penjelasan kesalahan digunakan pada fungsionalitas kirim penjelasan kesalahan. Implementasi penjelasan kesalahan digambarkan pada gambar 5.38 yang digunakan untuk mengirim penjelasan kesalahan.



Gambar 5.38 Implementasi antarmuka penjelasan kesalahan

5.9.14 Implementasi antarmuka Update status usulan

Implementasi *update* status usulan digambarkan pada gambar 5.39 adalah representasi dari fungsionalitas *update* usulan. *Update* status usulan adalah mengirim informasi terbaru terkait usulan kepada dosen.

SKJD Apps v1.0

Daftar dosen
Daftar Usulan

Halaman Update Status Usulan

status proses usulan dosen

ID Usulan
LU_0003

Nama Dosen
Muhammad Saheza, S.Kom, M.Kom

Jenis Pengajaran
Asisten Aji

Status usulan
selesai

TMT Pangkat
Masukan TMT Pangkat

TMT Jabatan
Masukan TMT Jabatan

February 2019

Su	Mo	Tu	We	Th	Fr	Sa
						1 2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28		

Gambar 5.39 Implementasi antarmuka update status usulan

5.9.15 Implementasi antarmuka Edit dupak

Implementasi *edit* dupak digambarkan pada gambar 5.40 adalah representasi dari fungsionalitas *edit* dupak. *Edit* dupak adalah menampilkan menambah atau mengubah data terkait dupak dosen.

SKJD Apps v1.0

Daftar dosen
Daftar Usulan

Halaman edit dupak

Halaman

Dekan/Rektor/Kajur Data Pendukung 4 Bidang Dari Tim Penilai

Ketua Jurusan Nama

Ketua Jurusan NIP

Ketua Jurusan Pangkat

Ketua Jurusan Jabatan

Ketua Jurusan Unit Kerja

Rektor Nama

Rektor NIP

Save

Gambar 5.40 Implementasi antarmuka edit dupak

BAB 6 PENGUJIAN

Pada bab pengujian ini akan membahas pengujian sistem yang sudah diimplementasikan pada bab perancangan dan implementasi. Tahap pertama adalah pengujian *white box*, *black box*, dan pengujian integrasi.

6.1 Pengujian *White Box*

Pengujian *white box* adalah pengujian yang menguraikan secara detail proses pada suatu fungsi yang akan diuji.

6.1.1 Pengujian *white box update status usulan*

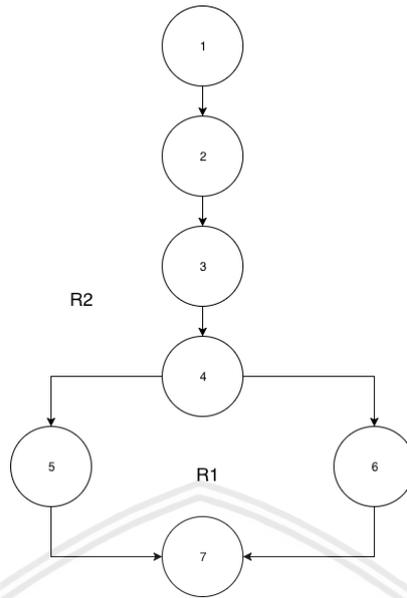
Dilakukan pengujian fungsi *update* status usulan berdasarkan algoritme *update* status usulan. Tabel 6.1 menjelaskan *pseudocode* dari algoritme *update* status usulan yang nantinya dicari jalur independen, dan *cyclomatic complexity*.

Tabel 6.1 Tabel *pseudocode white box update status usulan*

Algoritme 1: Fungsi Update status usulan	
1	void update_status_usulan () {
2	initialize nip/nik
3	initialize id usulan
4	initialize status usulan
5	initialize jenis pengajuan
6	initialize email tujuan
7	initialize tmt pangkat
8	initialize tmt jabatan
9	initialize AK
10	
11	data dosen = validasi_usulan (tmt jabatan, tmt pangkat, AK,
12	jenis pengajuan)
13	data dosen add nip/nik
14	
15	data usulan add nip/nik, id usulan, status usulan
16	
17	if status usulan == "selesai" then
18	set data usulan to database
19	set data dosen to database
20	set status usulan and email tujuan to send email
21	else
22	set data usulan to database
23	set status usulan and email tujuan to send email
24	end if
25	}

Flowgraph didapat dari *pseudocode update* status usulan. Setelah ditentukan *flowgraph* maka dicari *cyclomatic complexity*, jalur independen, dan kasus uji.

Pada gambar 6.1 menjelaskan *flowgraph* dari fungsi *update* status usulan dan pada tabel 6.2 menjelaskan tabel kasus uji dari fungsi *update* status usulan.



Gambar 6.1 Flowgraph white box update status usulan

Cyclometric complexity $V(G)$

$V(G)$ = Jumlah region = 2.

$V(G)$ = Edge – Node + 2 = 7 – 7 + 2 = 2.

Independent Path

Jalur 1: 1-2-3-4-5-7.

Jalur 2: 1-2-3-4-6-7.

Tabel 6.2 Tabel kasus uji white box update status usulan

ID	Path	Test Case	Expected Result	Actual Result	Valid/Invalid
1.	1-2-3-4-5-7	Kepegawaian memilih status usulan "selesai"	Menyimpan data tmt jabatan dan/atau tmt pangkat, AK, mengubah status usulan pada <i>database</i> menjadi selesai dan mengirim <i>email</i> pemberitahuan.	Menyimpan data tmt jabatan dan/atau tmt pangkat, AK, mengubah status usulan pada <i>database</i> menjadi selesai dan mengirim <i>email</i> pemberitahuan.	<i>valid</i>
2.	1-2-3-4-6-7	Kepegawaian memilih status usulan selain "selesai"	Mengubah status usulan pada <i>database</i> dan mengirim <i>email</i> pemberitahuan.	Mengubah status usulan pada <i>database</i> dan mengirim <i>email</i> pemberitahuan.	<i>valid</i>

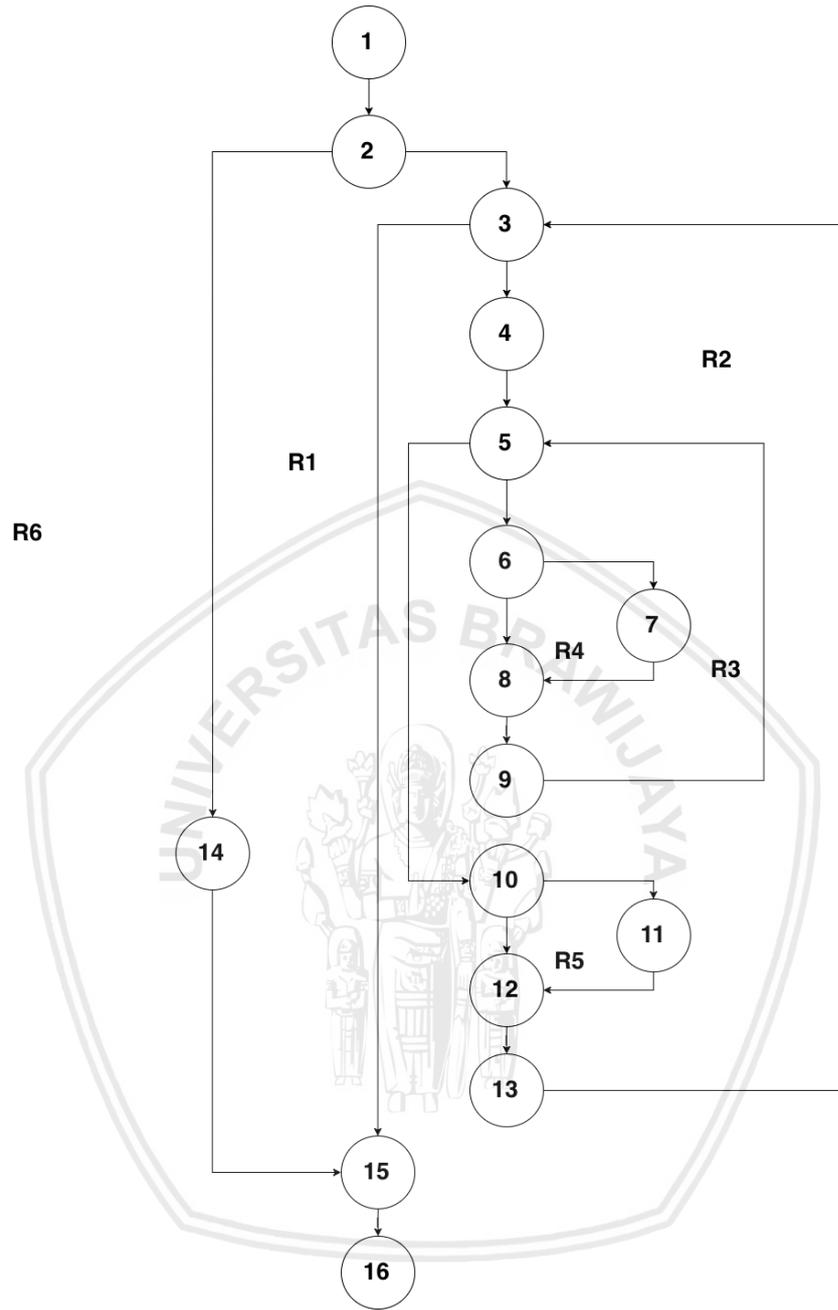
6.1.2 Pengujian *white box* lihat berkas bkd

Dilakukan pengujian fungsi lihat usulan berdasarkan algoritme lihat berkas bkd. Tabel 6.3 menjelaskan *pseudocode* dari algoritme lihat berkas bkd yang nantinya dicari jalur independen, dan *cyclomatic complexity*.

Tabel 6.3 Tabel *pseudocode white box* lihat berkas bkd

Algoritme 2: Fungsi lihat berkas bkd	
1	void lihat_berkas_bkd () {
2	
3	inisialize nip/nik
4	berkasFromApi = get from api with nip/nik
5	initialize arrayFilter
6	berkasFromApp = get from database berkas usulan with nip/nik
7	
8	if berkasFromApi > 0 then
9	
10	loop x from 0 until end of index berkasFromApi
11	
12	temporary = false
13	
14	loop y from 0 until end of index berkasFromApp
15	
16	if berkasAppApi at x == berkasAppApi at y then
17	
18	temporary = true break
19	
20	endIf
21	
22	endLoop
23	
24	if temporary == false then
25	
26	push berkasFromApi at x to arrayFilter
27	
28	endIf
29	
30	endLoop
31	
32	else
33	
34	arrayFilter = berkasFromApi
35	
36	endIf
37	
38	show view
39	}

Pada gambar 6.2 menjelaskan *flowgraph* dari fungsi lihat berkas bkd dan pada tabel 6.4 menjelaskan tabel kasus uji dari fungsi lihat berkas bkd.



Gambar 6.2 Flowgraph white box lihat berkas bkd

Cyclometric complexity $V(G)$

$V(G)$ = Jumlah region = 6.

$V(G)$ = Edge – Node + 2 = 20 – 16 + 2 = 6.

Independent Path

Jalur 1: 1-2-3-4-5-6-7-8-9-5-10-11-12-13-3-15-16.

Jalur 2: 1-2-3-4-5-10-11-12-13-3-15-16.

Jalur 3: 1-2-3-4-5-10-12-13-3-15-16.



Jalur 4: 1-2-3-4-5-6-8-9-5-10-11-12-13-3-15-16.

Jalur 5: 1-2-3-4-5-6-8-9-5-10-12-13-3-15-16.

Jalur 6: 1-2-14-15-16.

Tabel 6.4 Tabel kasus uji *white box* lihat berkas bkd

ID	Path	Test Case	Expected Result	Actual Result	Valid/Invalid
1.	1-2-3-4-5-6-7-8-9-5-10-11-12-13-3-15-16	Ada data di <i>webservice</i> BKD yang sama. Atau sudah diusulkan.	Berkas yang sudah diusulkan tidak tampil pada tabel berkas BKD.	Berkas yang sudah diusulkan tidak tampil pada tabel berkas BKD.	<i>valid</i>
2.	1-2-3-4-5-10-11-12-13-3-15-16	Tidak ada data yang sama antara <i>webservice</i> BKD dengan usulan. Lalu di masukan ke berkas baru baru	Menampilkan semua berkas BKD dari <i>Websercive</i> .	Menampilkan semua berkas BKD dari <i>Websercive</i> .	<i>valid</i>
3.	1-2-3-4-5-10-12-13-3-15-16	Tidak ada data yang sama antara <i>webservice</i> BKD dengan usulan. Lalu tidak di masukan ke berkas baru	Menampilkan kosong semua berkas BKD dari <i>Websercive</i> .	Menampilkan kosong semua berkas BKD dari <i>Websercive</i> .	<i>valid</i>
4.	1-2-3-4-5-6-8-9-5-10-11-12-13-3-15-16	Menampilkan data berdasarkan <i>webservice</i> berkas BKD sama dengan usulan.	Menampilkan semua berkas BKD dari <i>Websercive</i> .	Menampilkan semua berkas BKD dari <i>Websercive</i> .	<i>valid</i>
5.	1-2-3-4-5-6-8-9-5-10-12-13-3-15-16	Tidak Menampilkan data berdasarkan <i>webservice</i> berkas BKD sama dengan usulan.	Menampilkan kosong semua berkas BKD dari <i>Websercive</i> .	Menampilkan kosong semua berkas BKD dari <i>Websercive</i> .	<i>valid</i>
6.	1-2-14-15-16	Tidak ada berkas BKD yang diusulkan.	Menampilkan semua berkas BKD dari <i>Websercive</i> .	Menampilkan semua berkas BKD dari <i>Websercive</i> .	<i>valid</i>

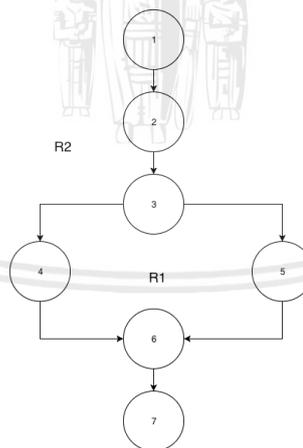
6.1.3 Pengujian *white box* ajukan berkas BKD

Dilakukan pengujian fungsi lihat usulan berdasarkan algoritme ajukan berkas BKD. Tabel 6.5 menjelaskan *pseudocode* dari algoritme ajukan berkas BKD yang nantinya dicari jalur independen, dan *cyclomatic complexity*.

Tabel 6.5 Tabel *pseudocode white box* ajukan berkas BKD

Algoritme 3: Fungsi ajukan berkas bkd	
1	void riwayat_usulan () {
2	
3	inialize nip/nik, id berkas, nama berkas, file, status
4	berkas, id unsur, tanggal, keterangan, jml vol, jml ak 1
5	inialize array data berkas BKD
6	
7	add nip/nik, id berkas, nama berkas, file, status berkas, id
8	unsur, tanggal, keterangan, jml vol, jml ak to data berkas
9	BKD 2
10	
11	if id unsur not empty and jml vol not empty and jml ak not
12	empty then 3
13	set data berkas BKD to store database 4
14	else
15	alert "Pilih unsur bidang dan isi jumlah volume kegiatan
16	dan jumlah AK" 5
17	end if 6
18	
19	redirect to lihat berkas bkd 7
	}

Pada gambar 6.3 menjelaskan *flowgraph* dari fungsi ajukan berkas bkd dan pada tabel 6.6 menjelaskan tabel kasus uji dari fungsi ajukan berkas bkd.



Gambar 6.3 *Flowgraph white box* ajukan berkas BKD

Cyclometric complexity $V(G)$

$V(G)$ = Jumlah region = 2.

$V(G)$ = Edge – Node + 2 = 7 – 7 + 2 = 2.

Independent Path

Jalur 1: 1-2-3-4-6-7.

Jalur 2: 1-2-3-5-6-7.

Tabel 6.6 Tabel kasus uji *white box* ajukan berkas BKD

ID	Path	Test Case	Expected Result	Actual Result	Valid/Invalid
1.	1-2-3-4-6-7	Dosen mengisi jenis kegiatan, jumlah volume, jumlah AK dan menekan tombol centang.	Berkas bkd menjadi berkas usulan dan tampil di tabel berkas usulan.	Berkas bkd menjadi berkas usulan dan tampil di tabel berkas usulan.	valid
2.	1-2-3-5-6-7	Dosen menekan tombol centang saja.	Menampilkan peringatan "Pilih unsur bidang dan isi jumlah volume kegiatan dan jumlah AK".	Menampilkan peringatan "Pilih unsur bidang dan isi jumlah volume kegiatan dan jumlah AK".	valid

6.2 Pengujian *Black Box*

Pengujian *black box* adalah pengujian yang fungsionalitas bertumpu pada hasil eksekusi fungsi yang dibuat. Pengujian *black box* di buat untuk semua fungsionalitas yang berdasarkan *usecase* yang ada di *usecase diagram*. Pada tabel 6.2 menjelaskan tentang pengujian fungsionalitas dengan pengujian *black box*.

Tabel 6.7 Tabel pengujian *black box*

ID	Fungsi	Test Case	Expected Result	Actual Result	Valid/Invalid
1.	Login	Pengguna memasukan username dan password dengan benar.	Diarahkan ke dashboard dosen.	Diarahkan ke dashboard dosen.	Valid
2.	Tambah usulan	Dosen menambah usulan dengan lengkap dan benar.	Usulan tampil pada halaman riwayat usulan.	Usulan tampil pada halaman riwayat usulan.	Valid
3.	Hapus usulan	Dosen menghapus salah satu usulan dan menekan tombol ok.	Usulan terhapus dan tidak tampil pada halaman riwayat usulan.	Usulan terhapus dan tidak tampil pada halaman riwayat usulan.	Valid
4.	Hapus berkas administrasi	Dosen menghapus salah satu	Berkas administrasi terhapus dan	Berkas administrasi terhapus dan	Valid

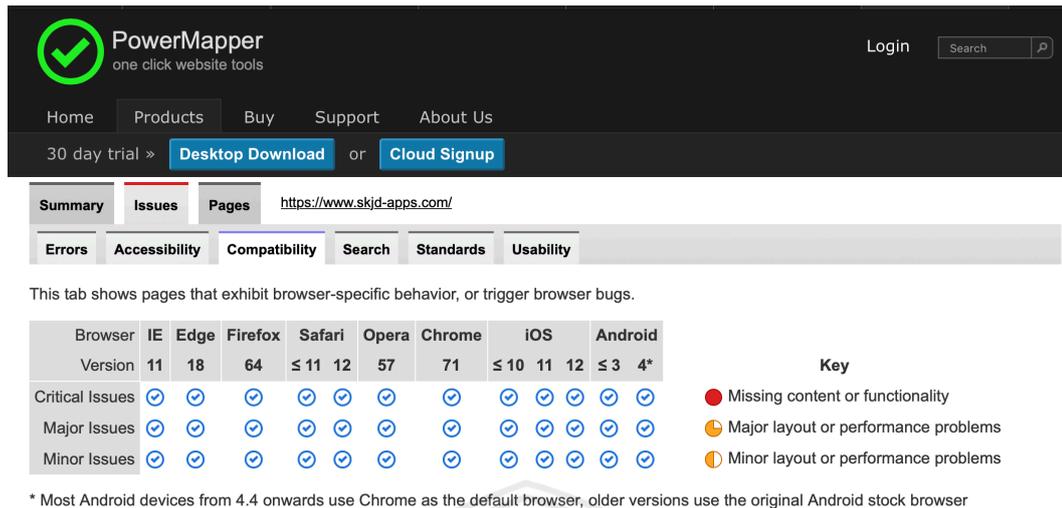
		berkas administrasi dan menekan tombol ok.	tidak tampil pada halaman data diri.	tidak tampil pada halaman data diri.	
5.	Cetak dupak	Dosen atau kepegawaian memilih salah satu usulan dan menekan tombol cetak.	Tampil berkas pdf pada halaman baru.	Tampil berkas pdf pada halaman baru.	<i>Valid</i>
6.	Tambah berkas	Dosen atau kepegawaian menambah berkas usulan, berkas administrasi dan berkas administrasi tambahan.	Berkas usulan yang ditambah tampil di halaman berkas usulan dan berkas administrasi tampil di halaman data diri.	Berkas usulan yang ditambah tampil di halaman berkas usulan dan berkas administrasi tampil di halaman data diri.	<i>Valid</i>
7.	Lihat data diri	Dosen menekan menu lihat data diri.	Tampil data diri dosen secara lengkap.	Tampil data diri dosen secara lengkap.	<i>Valid</i>
8.	Lihat berkas BKD	Dosen menekan tombol lihat berkas BKD.	Tampil berkas BKD dosen berbentuk tabel.	Tampil berkas BKD dosen berbentuk tabel.	<i>valid</i>
9.	Edit usulan	Dosen mengubah semua data usulan dengan lengkap.	Data usulan berubah dan tampil pada halaman riwayat usulan.	Data usulan berubah dan tampil pada halaman riwayat usulan.	<i>Valid</i>
10.	Lihat berkas usulan	Dosen menekan menu lihat berkas usulan.	Tampil berkas usulan berbentuk tabel.	Tampil berkas usulan berbentuk tabel.	<i>Valid</i>
11.	Lihat grafik AK	Dosen menekan menu dashboard.	Tampil grafik keurangan AK jabatan dan pangkat.	Tampil grafik keurangan AK jabatan dan pangkat.	<i>Valid</i>
12.	Ajukan berkas BKD	Dosen tidak menekan tombol centang	Tampil peringatan harus diisi semua.	Tampil peringatan harus diisi semua.	<i>Valid</i>
13.	Edit berkas	Dosen upload data berkas	Berkas berhasil	Berkas berhasil	<i>Valid</i>

	usulan	baru.	diganti.	diganti.	
14.	Hapus berkas usulan	Dosen memilih berkas usulan dan menghapus berkas dengan menekan tombol ok.	Berkas usulan tidak tampil pada halaman berkas usulan.	Berkas usulan tidak tampil pada halaman berkas usulan.	<i>Valid</i>
15.	Lihat usulan	Dosen menekan menu riwayat usulan.	Tampil daftar usulan berupa tabel.	Tampil daftar usulan berupa tabel.	<i>Valid</i>
16.	logout	Dosen/kepegawaian menekan menu <i>logout</i> .	Berhasil mengarahkan ke halaman <i>login</i> .	Berhasil mengarahkan ke halaman <i>login</i> .	<i>Valid</i>
17.	Kirim Pemberitahuan kenaikan pangkat dan jabatan	Cronjob mengeksekusi data tanggal TMT dosen yang tepat 2 tahun jatuh pada hari ini.	Dosen mendapatkan email pemberitahuan dapat mengajukan kenaikan pangkat atau jabatan.	Dosen mendapatkan email pemberitahuan dapat mengajukan kenaikan pangkat atau jabatan.	<i>Valid</i>
18.	Lihat usulan dosen	Kepegawaian menekan menu daftar usulan.	Tampil daftar usulan berupa tabel.	Tampil daftar usulan berupa tabel.	<i>Valid</i>
19.	Batal validasi dan verifikasi berkas administrasi	Kepegawaian menekan tombol batal validasi dan verifikasi.	Tombol berubah menjadi "validasi dan verifikasi"	Tombol berubah menjadi "validasi dan verifikasi"	<i>Valid</i>
20.	Batal validasi dan verifikasi berkas usulan	Kepegawaian menekan tombol batal validasi dan verifikasi.	Tombol berubah menjadi "validasi dan verifikasi"	Tombol berubah menjadi "validasi dan verifikasi"	<i>Valid</i>
21.	Lihat detail usulan.	kepegawaian menekan tombol detail.	Tampil data diri dosen, berkas administrasi, dan berkas usulan dosen.	Tampil data diri dosen, berkas administrasi, dan berkas usulan dosen.	<i>Valid</i>
22.	Lihat daftar dosen	Kepegawaian menekan menu daftar dosen.	Tampil daftar dosen berupa tabel.	Tampil daftar dosen berupa tabel.	<i>Valid</i>
23.	Validasi dan verifikasi usulan	Kepegawaian menekan tombol validasi	halaman detail berkas usulan dosen menjadi	halaman detail berkas usulan dosen	<i>Valid</i>

		dan verifikasi usulan.	“update status usulan” dan <i>Email</i> terkirim pada dosen.	menjadi “update status usulan” dan <i>Email</i> terkirim pada dosen.	
24.	Kirim penjelasan kesalahan	Kepegawaian memasukan penjelasan kesalahan dan menekan tombol <i>save</i> .	<i>Email</i> penjelasan kesalahan terkirim pada dosen.	<i>Email</i> penjelasan kesalahan terkirim pada dosen.	<i>Valid</i>
25.	Validasi dan verifikasi berkas administrasi	Kepegawaian menekan tombol validasi dan verifikasi.	Status berkas menjadi sudah divalidasi.	Status berkas menjadi sudah divalidasi.	<i>Valid</i>
26.	Validasi dan verifikasi berkas usulan	Kepegawaian menekan tombol validasi dan verifikasi.	Status berkas menjadi belum divalidasi.	Status berkas menjadi belum divalidasi.	<i>Valid</i>
27.	Update status usulan	Kepegawaian memilih status usulan “di KTU”.	Status usulan menjadi “di KTU”.	Status usulan menjadi “di KTU”.	<i>Valid</i>
28.	Edit Dupak	Kepegawaian menambah data kepala jurusan.	Kepala jurusan tampil pada berkas lembar pdf dupak.	Kepala jurusan tampil pada berkas lembar pdf dupak.	<i>Valid</i>

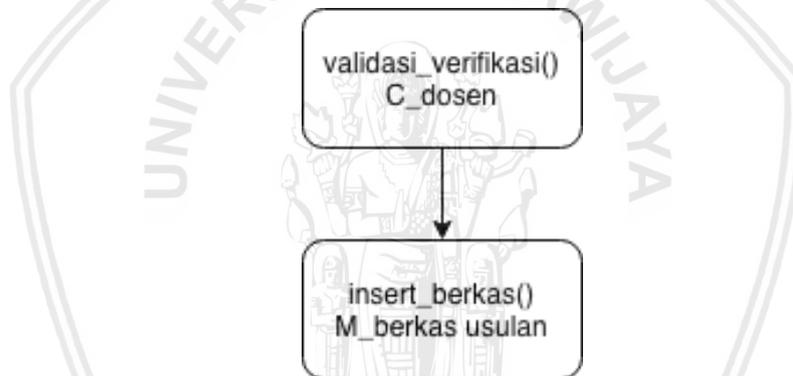
6.3 Pengujian *Compatibility*

Pengujian *compatibility* adalah pengujian sistem dapat beradaptasi pada lingkungan yang berbeda-beda. Dalam hal ini lingkungan tersebut adalah *browser* pengguna. Pada pengujian non-fungsional ini dilakukan dengan 5 *browser* uji yaitu *Safari 12*, *Microsoft Edge 18*, *Firefox 64*, *Chrome 71* dan *Opera 57*. Pada pengujian ini menggunakan tools online yaitu *SortSite* dengan memasukan alamat URL (*Uniform Resource Locator*) sistem. Dan hasilnya sistem kenaikan pangkat dan jabatan Fakultas Ilmu Komputer Universitas Brawijaya atau *SKJD Apps* ini memenuhi parameter di lingkungan 5 *browser* tersebut yang digambarkan pada gambar 6.4.



Gambar 6.4 Hasil Pengujian *Compatibility*

6.4 Pengujian Integrasi



Gambar 6.5 Diagram hierarki pengujian integrasi

Pengujian Integrasi adalah pengujian yang digunakan menguji komunikasi antar modul. Pengujian ini menggunakan pendekatan integrasi *top-down*. Pengujian *top-down* dilakukan dengan menguji *lower* modul yang belum terintegrasi atau *stubs*. *Stubs* nantinya akan diuji luaranya dan diintegrasikan dengan fungsi yang memanggilnya.

Pada pengujian integrasi ini menggunakan uji sampel fungsi ajukan berkas bkd yang berada ada *controller* dengan nama fungsi *validasi_verifikasi()* dan *lower* model yang digunakan adalah fungsi *insert_berkas()* pada model *M_berkas_usulan()*.

Tabel 6.8 Tabel *pseudocode* stubAjukanBerkasBKD()

Pseudocode : stubAjukanBerkasBKD	
1	START
2	Get parameter data
3	Return variable data
4	END

Tabel 6.9 Tabel pseudocode pengujian fungsi stubAjukanBerkasBKD()

Pseudocode : pengujian stubAjukanBerkasBKD	
1	START
2	Init nip/nik = '196103131986112001'
3	init id berkas = 'A_0002'
4	init nama berkas = 'Mengajar ASD TIF'
5	init file = 'http://api.skjd-apps.com/assets/dokumen_2.pdf'
6	status berkas = 'belum divalidasi'
7	id unsur = 'P_0034'
8	tanggal = 'Semester Genap 2019-2019'
9	keterangan = '1. SK'
10	jml vol = '5'
11	jml ak = '2,5'
12	
13	init array data
14	
15	add data from nip/nik, id berkas, nama berkas, file, status
16	berkas, id unsur, tanggal, keterangan, jml vol, jml ak to
17	data berkas BKD
18	
19	if id unsur not empty and jml vol not empty and jml ak not
20	empty then
21	call stubAjukanBerkasBKD(data)
22	else
23	alert "Pilih unsur bidang dan isi jumlah volume kegiatan
24	dan jumlah AK"
25	end if
26	redirect to lihat berkas bkd
27	END

Pada pengujian hasil ujicoba stubAjukanBerkasBKD menghasilkan luaran data sesuai dengan masukan kasus uji masukan yang digambarkan pada gambar 6.5. Selanjutnya melakukan pengujian integrasi yang antara *controller fungsi validasi_verifikasi()* dengan fungsi *insert_berkas()* pada model *M_berkas_usulan()*. Pengujian dilakukan dengan *pseudocode* kedua fungsi yang di tuliskan pada tabel dan tabel 6.10 dan 6.11.



Gambar 6.6 Hasil Stub stubAjukanBerkasBKD()

Tabel 6.10 Tabel pseudocode fungsi validasi_verifikasi() di kelas C_dosen

Pseudocode : fungsi validasi verifikasi() controller C dosen	
1	START
2	Init nip/nik = '196103131986112001'
3	init id berkas = 'A_0002'
4	init nama berkas = 'Mengajar ASD TIF'
5	init file = 'http://api.skjd-apps.com/assets/dokumen_2.pdf'



```

6 status berkas = 'belum divalidasi'
7 id unsur = 'P_0034'
8 tanggal = 'Semester Genap 2019-2019'
9 keterangan = '1. SK'
10 jml vol = '5'
11 jml ak = '2,5'
12
13 init array data
14
15 add data from nip/nik, id berkas, nama berkas, file, status
16 berkas, id unsur, tanggal, keterangan, jml vol, jml ak to
17 data berkas BKD
18
19 if id unsur not empty and jml vol not empty and jml ak not
20 empty then
21 call M_berkas_usulan->insert_berkas(data)
22 else
23 alert "Pilih unsur bidang dan isi jumlah volume kegiatan
24 dan jumlah AK"
25 end if
26
27 redirect to lihat berkas bkd
28 END
    
```

Tabel 6.11 Tabel *pseudocode* pengujian fungsi insert_berkas() di kelas M_berkas_usulan

Pseudocode : fungsi insert berkas() model M berkas usulan	
1	START
2	Get parameter data
3	call function query insert data
4	END

Pada hasil pengujian integrasi diatas menghasilkan bahwa pengujian data yang diintegrasikan adalah valid dan data yang di masukan sesuai dengan data yang diperoleh dari stubAjukanBerkasBKD() yang digambarkan pada gambar 6.6.

Mengikuti pendidikan formal dan memperoleh gelar/sebutan/jazah doktor/ sederajar	pendidikan	200	U_0040	Doktor S3 ILKOM UII 2015	lihat	belum divalidasi	Edit Hapus
Menulis buku pelajaran SMA yang diterbitkan dan diedarkan secara nasional buku SMTA/ setingkat	penunjang	5	U_0040	Menulis buku IT Dasar	lihat	belum divalidasi	Edit Hapus
perkuliahan/tutorial/perkuliahan praktikum dan membimbing, menguji serta menyelenggarakan pendidikan di laboratorium, praktik kejuruan, bengkel/studio/kebum percobaan/teknologi pengajaran dan praktik lapangan (tenaga pengajar, asisten ahli)	pendidikan	2.5	belum diusulkan	Mengajar ASD TIF	lihat	belum divalidasi	Edit Hapus

Showing 1 to 5 of 5 entries

Previous 1 Next

Gambar 6.7 Hasil dari luaran integrasi fungsi insert_berkas dari M_berkas_usulan dan fungsi validasi_berifikasi _dari C_dosen



BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan hasil analisis, perancangan, implementasi dan pengujian yang telah dilakukan maka dapat disimpulkan sebagai berikut.

1. Berdasarkan hasil analisis kebutuhan didapatkan 28 kebutuhan fungsional dan 1 kebutuhan non-fungsional sistem. Dari hasil tersebut teridentifikasi 2 aktor yang berlaku sebagai pengguna.
2. Berdasarkan hasil perancangan arsitektur sistem, Sistem dapat mengakses berkas BKD melalui *webservice dummy* BKD. Selain itu juga dilakukan perancangan basis data, perancangan algoritme program, perancangan *class* diagram, perancangan purwarupa sistem dan perancangan *sequence* diagram.
3. Berdasarkan hasil implementasi sistem menerapkan Implementasi basis data, implementasi *class* diagram, implementasi kode program dan implementasi antar muka.
4. Berdasarkan hasil pengujian unit dan integrasi menghasilkan 100% hasil kasus uji bernilai valid. Dan yang terakhir hasil uji *compatibility* sistem pada lingkungan *browser* dengan alat bantu *online* SortSite menghasilkan sistem dapat berjalan dengan baik di berbagai *browser*.

7.2 Saran

Saran yang diberikan pada pembangunan sistem kenaikan pangkat dan jabatan FILKOM ini selanjutnya sebagai berikut.

1. Ranah sistem sebaiknya dapat diluaskan pada lingkup universitas. Karena proses yang dilakukan masih pada tahap kepegawaian fakultas.
2. Sistem sebaiknya diintegrasikan dengan data-data dosen yang ada di FILKOM. Data di FILKOM yang diintegrasikan seperti data ketua jurusan, data berkas kegiatan akademik dan penelitian dosen.

DAFTAR PUSTAKA

- Aditama, M.S.P., Agung Wicaksono, S. and Pradana, F., 2018. Pembangunan Sistem Informasi Kenaikan Jabatan Fungsional Dosen Universitas Brawijaya. 2(10), pp.3538–3544.
- Evi, A., Rohmah, N., Rusdianto, D.S. and Santoso, N., 2019. Pembangunan Sistem Informasi Manajemen Perizinan di Dinas KOMINFO Kota Pasuruan. 3(1), pp.665–672.
- Rasyid, A., Pradana, F. and Rusdianto, D.S., 2018. Pembangunan Sistem Informasi Pengajuan Kegiatan dan Pemberian Poin Bagi Mahasiswa Fakultas Ilmu Komputer Universitas Brawijaya. 2(9), pp.3409–3415.
- Sudrajat, T., 2016. *Materi Sosialisasi Pengusulan Jabatan Fungsional Dosen*. [Online]
Available at: <http://kepegawaian.ub.ac.id/layanan/materi-sosialisasi-pengusulan-jabatan-fungsional-dosen/>
[Accessed 14 November 2018].
- Larman, C., 2005. *Applying UML and patterns: an introduction to object-oriented analysis and design and iterative development*. 3rd Edition ed. New Jersey: Prentice Hall PTR.
- Pressman, R. S., 2010. *Software engineering: a practitioner's approach*. 7th Edition ed. New York: McGraw-Hill Higher Education.
- Kurniawan, T. A., 2018. Pemodelan Use Case (UML): Evaluasi Terhadap beberapa Kesalahan dalam Praktik. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 5(1), p. 77.
- Rahman, M. F., 2017. *Bagian Kepegawaian UB*. [Online]
Available at: <http://kepegawaian.ub.ac.id/wp-content/uploads/2017/07/JAD-UB-JULI-2017-ubud-hotel.pdf>
[Accessed 11 November 2018].
- CodeIgniter, 2018. *Welcome to CodeIgniter*. [Online]
Available at: https://www.codeigniter.com/user_guide/general/welcome.html
[Accessed 12 Juni 2018].
- Kemenristekdikti, 2016. *Sistem Penilaian Angka Kredit Dosen*. [Online]
Available at: <http://pak.ristekdikti.go.id/portal?download=322>
[Accessed 12 November 2018].
- PPTI UB, 2015. *Manual Sistem Informasi Dosen*. Malang: Universitas Brawijaya.
- PPTI UB, 2013. *Manual Sistem Informasi Pegawai*. Malang: Universitas Brawijaya.
- Masse, M., 2011. *REST API Design Rulebook*. Sebastopol: O'Reilly Media.
- Pitt, C., 2012. *Pro PHP MVC*. 1st Edition ed. Berlin: Apress.

Visual Paradigm, 2019. *What is Class Diagram?*. [Online] Available at: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-class-diagram/> [Accessed 15 Februari 2019].

Visual Paradigm, 2019. *What is Sequence Diagram?*. [Online] Available at: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/> [Accessed 15 Februari 2019].

Prosedur Kenaikan Pangkat Fungsional Dosen tahun 2018. Jakarta: Kemenristekdikti.



LAMPIRAN A HASIL WAWANCARA

A.1 Hasil Wawancara Beberapa Dosen FILKOM

Berikut ini adalah hasil wawancara dari beberapa dosen FILKOM yang sudah pernah melakukan kenaikan pangkat/jabatan sebelumnya.

Tanggal : 1 Oktober 2018 – 5 Oktober 2018

Tempat : Fakultas Ilmu Komputer Universitas Brawijaya

Narasumber :

1. Fitra Abdurrachman Bachtiar, Dr.Eng., S.T, M.Eng
2. Fajar Pradana, S.ST, M.Eng
3. Bayu Priyambadha, S.Kom, M.Kom
4. Ratih Kartika Dewi, S.T., M.Kom
5. Faizatul Amalia, S.Pd.,M.Pd

Media : Smartphone Recorder.

Tujuan dari Wawancara ini adalah untuk menggali beberapa permasalahan yang ada pada saat mengusulkan kenaikan pangkat atau jabatan pada bagian kepegawaian.

Pertanyaan:

1. Apakah bapak/ibu mengetahui waktu naik pangkat/syarat untuk melakukan kenaikan pangkat?
2. Apakah bapak/ibu mengetahui/melakukan penghitungan angka kumulatif sebelum mengajukan kenaikan pangkat?
3. Apakah diingatkan oleh kepegawaian?
4. Kendala nya apa saja?
5. Apa pernah bapak/ibu melakukan penghitungan sendiri dan penghitungan kepegawaian berbeda

Jawaban diurutkan berdasarkan nomor narasumber.

Jawaban bapak Fitra Abdurrachman Bachtiar, Dr.Eng., S.T, M.Eng

1. informasinya kurang, hanya disampaikan waktu sosialisasi saja, untuk info yg terintegrasi di suatu tempat, belum tau dimana. Waktu kenaikan pangkatnya tau. Persyaratannya tidak tau.
2. belum pernah.
3. belum pernah diingatkan, beri info di email filkom ada sosialisasi.

4. sibuk mengajar dsb, lupa mengurus kepangkatan, ketika ada sosialisasi baru ingat, buktinya susah didapatkan.

5. belum pernah.

Jawaban bapak Fajar Pradana, S.ST, M.Eng

1. tau.

2. pernah.

3. Tidak pernah.

4. Prosesnya lama waktu di kepegawaian.

5. pernah.

Jawaban bapak Bayu Priyambadha, S.Kom, M.Kom

1. Tidak tahu, karena itu yg paling paham bagian kepegawaian, tempo hari sempat di warning tidak boleh studi lanjut dulu sebelum naik pangkat.

2. pernah.

3. tidak dikasih tau kapan mengurus apa aja.

4. sosialisasi nya tidak mendetail, barang buktinya kadang hilang, salah menentukan berkas nya itu masuk dibidang mana.

5. belum pernah.

Jawaban ibu Ratih Kartika Dewi, S.T., M.Kom

1. Tau, ke tanya ke pegawaian.

2. iya.

3. Sana (kepegawaian) ngasih tahu, lewat obrolan.

4. Belum ada.

5. belum pernah.

Jawaban Ibu Faizatul Amalia, S.Pd.,M.Pd

1. tau sendiri, diberi tau dari teman, syaratnya belum, karena syaratnya belum jelas, maka tanya ke kepegawaian.

2. belum pernah.

3. belum pernah.

4. Info terlalu mepet.

5. belum pernah.

A.2 Hasil Wawancara Staf Kepegawaian FILKOM

Berikut ini adalah hasil wawancara dari staf kepegawaian FILKOM yang menangani pengurusan kenaikan pangkat/jabatan .

Tanggal	: 1 Oktober 2018 – 30 Oktober 2018
Tempat	: Fakultas Ilmu Komputer Universitas Brawijaya
Narasumber	: Winda Sastra Dewi, S.AB
Media	: Smartphone Recorder.

Tujuan dari Wawancara ini adalah untuk menggali beberapa pengetahuan tentang kenaikan pangkat dan jabatan disisi kepegawaian.

Pertanyaan 1 : Apa saja yang diisi dosen dana pa saja yang diisi oleh kepegawaian di DUPAK?

Jawaban 1 : Dosen hanya mengisi berkas kegiatan saja, yang lainnya diisi oleh kepegawaian.

Pertanyaan 2 : Apakah berkas BKD dapat diusulkan kenaikan pangkat/jabatan?

Jawaban 2 : Bisa dijadikan angka kredit.

Pertanyaan 3 : Apakah sosialisasi dilakukan tiap tahun oleh kepegawaian?

Jawaban 3 : Tidak tau, yang mengurus itu pusat.

Pertanyaan 4 : Apakah alur kenaikan jabatan dosen non pns sama dengan dosen pns?

Jawaban 4 : Aturannya sama dengan pns. Non pns tmt nya dari TMT CDT (Calon Dosen Tetap)

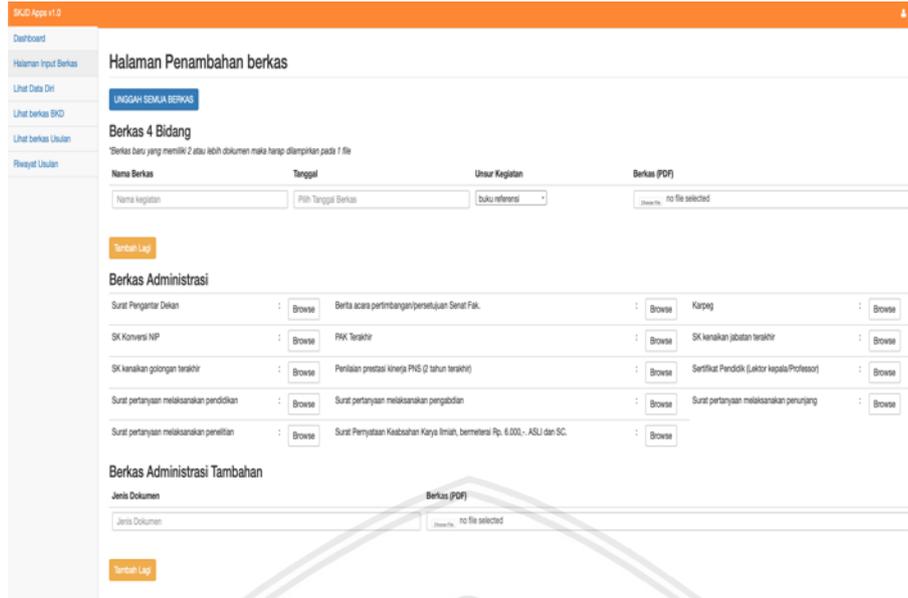
Pertanyaan 5 : Apakah ada tenggat waktu dalam pengumpulan berkas?

Jawaban 5 : untuk pengusulan kenaikan jabatan tidak ada waktu, untuk kenaikan pangkat minimal Januari itu TMT bulan April dan Juli untuk TMT bulan Oktober.

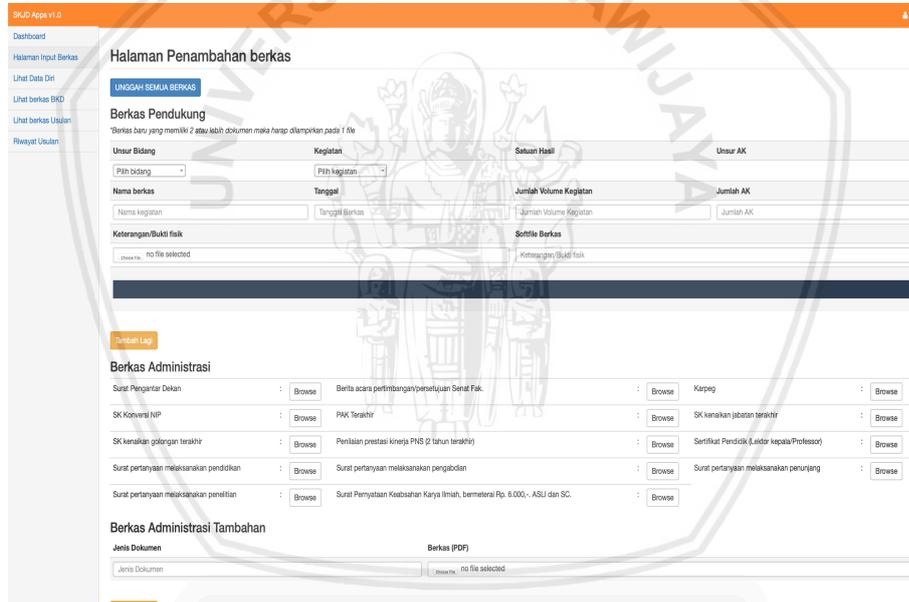
A.3 Iterasi Perubahan Antar Muka

Pada perancangan pertama yang dilakukan pada 21 Januari 2019 dengan narasumber pengguna dosen FILKOM ibu Ratih Kartika Dewi, S.T, M.Kom. telah dikoreksi pada iterasi pertama yang awal mula desain digambarkan pada gambar berikut.

1. Antar muka penambahan diawal



2. Antar muka setelah direvisi



3. Dokumen bahwa sistem telah direvisi oleh pengguna

KUESIONER PENGALAMAN PENGGUNA

Tujuan dari wawancara pengalaman pengguna pada Sistem kenaikan pangkat dan jabatan dosen Fakultas Ilmu Komputer Universitas Brawijaya. Lembar kuesioner pengguna ini digunakan untuk artefak dokumen penelitian skripsi atas nama Muhammad Safreza Teknik Informatika 2015 dan juga sebagai landasan proses pengembangan perangkat lunak.

Masukan pengguna..

Sudah baik, ada fitur point kum yang otomatis di generate sistem saat pengguna memasukkan komponen, misalnya perubahan /keterlal /praktikum mendapat nilai kum o.s.

Malang,


(Rafli Kartika Dewi, S.T, Nijom)

