

**ANALISIS DAN PENERAPAN SISTEM INFORMASI MANAJEMEN
SEKOLAH JIBAS (STUDI KASUS : SMAN 4 JEMBER)**

SKRIPSI

KONSENTRASI REKAYASA PERANGKAT LUNAK

Diajukan untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer



Disusun Oleh:

A.YAZID BUSTOMY

NIM. 0910681015

**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
PROGRAM STUDI INFORMATIKA
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA**

MALANG

2013

LEMBAR PERSETUJUAN

**ANALISIS DAN PENERAPAN SISTEM INFORMASI MANAJEMEN
SEKOLAH JIBAS (STUDI KASUS : SMAN 4 JEMBER)**

SKRIPSI

KONSENTRASI REKAYASA PERANGKAT LUNAK

Diajukan untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer



Disusun oleh :

A.YAZID BUSTOMY

NIM. 0910681015

Telah diperiksa dan disetujui pada

Dosen Pembimbing I

Dosen Pembimbing II

Ismiarta Aknuranda, ST.M.Sc, Phd

NIK. 74071906110079

Novanto Yudistira S.Kom, M.Sc

NIK. 831110 16 1 1 0425

LEMBAR PENGESAHAN
ANALISIS DAN PENERAPAN SISTEM INFORMASI MANAJEMEN
SEKOLAH JIBAS (STUDI KASUS : SMAN 4 JEMBER)

SKRIPSI
KONSENTRASI REKAYASA PERANGKAT LUNAK
Diajukan untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer

Disusun oleh :
A.YAZID BUSTOMY
NIM. 0910681015

Skripsi ini telah diuji dan dinyatakan lulus pada
tanggal 18 Juli 2013

Penguji I

Penguji II

Denny Sagita Rusdianto, S.Kom., M.Kom.

Fajar Pradana, S.ST.,M.Eng

NIK. 85112406110250

NIK. 831110 16 1 1 0425

Penguji III

Aryo Pinandito, S.T., M.MT.

NIK. 820404 06 1 1 0091

Mengetahui
Ketua Program Studi Informatika

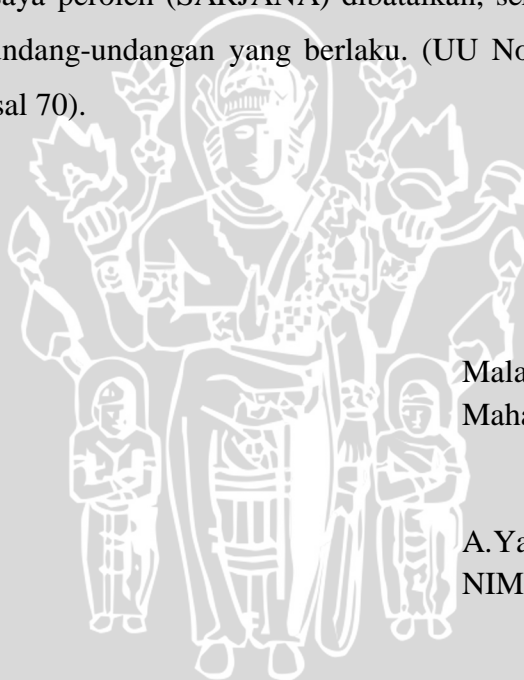
Drs. Marji, M.Si.

NIP. 19670801 199203 1 001

PERNYATAAN ORISINALITAS SKRIPSI

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 dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan 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, 22 Juli 2012
Mahasiswa,

A.Yazid Bustomy
NIM 0910681015

KATA PENGANTAR

Dengan nama Allah SWT Yang Maha Pengasih dan Penyayang. Segala puji bagi Allah SWT karena atas rahmat dan hidayahNya-lah penulis dapat menyelesaikan Tugas Akhir yang berjudul “**Analisis dan Penerapan Sistem Informasi Manajemen Sekolah Jibas (Studi Kasus : SMAN 4 Jember)**”. Shalawat dan salam atas junjungan besar kita Nabi Muhammad S.A.W. beserta keluarga dan para sahabat sekalian. Tugas Akhir ini disusun untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer di Program Studi Informatika Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya Malang.

Melalui kesempatan ini, penulis ingin menyampaikan rasa hormat dan terima kasih penulis yang sebesar-besarnya kepada semua pihak yang telah memberikan bantuan baik lahir maupun batin selama penulisan tugas akhir ini. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan rasa hormat dan terima kasih penulis kepada :

1. Kedua Orang Tua penulis (M. Zainal Abidin dan Nurifatus Sholeha) dan seluruh keluarga yang senantiasa tiada henti hentinya memberikan do'a demi terselesainya tugas akhir ini.
2. Bapak Ir. Sutrisno, M.T, Bapak Ir. Heru Nurwasito, M.Kom, Bapak Himawat Aryadita, S.T, M.Sc, dan Bapak Eddy Santoso, S.Kom selaku Ketua, Wakil Ketua 1, Wakil Ketua 2 dan Wakil Ketua 3 Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
3. Bapak Drs. Marji, MT dan Bapak Issa Arwani, S.Kom., M.Sc selaku Ketua dan Sekretaris Program Studi Informatika serta segenap Bapak/Ibu Dosen, Staff Administrasi dan Perpustakaan Program Studi Informatika Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
4. Bapak Ismiarta Aknuranda, ST. M.Sc, Phd dan Bapak Novanto Yudistira S.Kom, M.Sc selaku dosen pembimbing tugas akhir penulis.
5. Seluruh Dosen Program Studi Informatika atas kesediaan membagi ilmunya kepada penulis.

6. Semua Asisten, Ka. Lab serta Laboran dari Laboratorium Program Studi Informatika yang telah memberikan banyak bantuan dan dukungan dalam menyelesaikan tugas akhir ini.
7. Seluruh pihak yang telah membantu kelancaran penulisan tugas akhir yang tidak dapat kami sebutkan satu persatu.

Penulis menyadari bahwa tugas akhir ini masih banyak kekurangan dan masih jauh dari sempurna. Untuk itu, saran dan kritik yang membangun sangat penulis harapkan. Semoga tugas akhir ini membawa manfaat bagi penyusun maupun pihak lain yang menggunakannya.

Malang, Maret 2013

Penulis



ABSTRAK

Ahmad Yazid Bustomy. 2013. Analisis dan Penerapan Sistem Informasi Manajemen Sekolah Jibas (Studi Kasus : SMAN 4 Jember). Skripsi Program Studi Informatika, Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya. Dosen Pembimbing : Ismiarta Aknuranda, ST.M.Sc, Phd dan Novanto Yudistira S.Kom, M.Sc

Dalam dunia bisnis, salah satu sistem informasi terintegrasi adalah sistem ERP (*Enterprise Resource Planning*). ERP dapat diimplementasikan pada organisasi lain, misalnya pada organisasi pendidikan yang disebut dengan ERP pendidikan. Teknologi ERP pendidikan dapat dimanfaatkan lebih lanjut di dunia pendidikan.

Skripsi ini bertujuan untuk menganalisis dan menerapkan sistem ERP pendidikan di SMAN 4 Jember. Pengelolaan administrasi dan manajemen sekolah di SMAN 4 Jember masih manual dan kurang begitu baik, sehingga diharapkan pada saat diterapkannya sistem ERP pendidikan, pengelolaan administrasi dan manajemen di SMAN 4 Jember akan menjadi lebih efektif dan efisien. Salah satu sistem ERP pendidikan yang dapat diterapkan di SMAN 4 Jember yaitu Sistem Informasi Manajemen Sekolah (SIMS) Jibas. SIMS Jibas bersifat *open source* sehingga dapat dikustomisasi sesuai dengan kebutuhan yang ada di organisasi pendidikan. Namun tidak semua kebutuhan di SMAN 4 Jember dapat terpenuhi oleh SIMS Jibas, karena masih dibutuhkan suatu analisa kebutuhan yang hasilnya akan diperlukan untuk desain kustomisasi SIMS Jibas, agar SIMS Jibas dapat berjalan dengan optimal pada saat penerapan di SMAN 4 Jember. Pengujian SIMS Jibas ini meliputi pengujian *white-box testing*, *black-box testing*, pengujian *compatibility*, dan pengujian *usability*. Pengujian *compatibility* sistem yang dibuat dapat berjalan pada 2 sistem operasi yaitu sistem operasi windows dan sistem operasi linux. Hasil pengujian *usability* menunjukkan bahwa 85% pengguna merasa puas dengan kinerja sistem dan 100% pengguna memerlukan pelatihan untuk mengoperasikan sistem.

Kata kunci : ERP, SIMS Jibas, *open source*, kustomisasi

ABSTRACT

Ahmad Yazid Bustomy. 2013. *Analyze And Apply The Information System Of educational in SMAN 4 Jember*

Advisor: Ismiarta Aknuranda, ST.M.Sc, Phd dan Novanto Yudistira S.Kom, M.Sc

In the business world, *one of the* integrated information system is an ERP (*Enterprise Resource Planning*) system. ERP can be implemented in other organization, for example in the educational organization *which called* with educational ERP. *Technology of educational ERP* can be used further *in the educational world*.

This thesis aims to analyze and *apply* the ERP system of educational in SMAN 4 Jember. *The supervision of school administration and management* in SMAN 4 Jember is manual and deficient. When the ERP system of educational is applied, then be expected that *the supervision of school administration and management* in SMAN 4 Jember will be more effective and efficient. One of the ERP system of educational that can be applied in SMAN 4 Jember is the *information system of school management (SIMS) Jibas*. SIMS Jibas also open source, so it can be customized according to the requirements that exist in educational organization. But not all requirements in SMAN 4 Jember can fulfilled by SIMS Jibas, because it still *be required an analysis of the requirement* that *the result* will be required to design customization of SIMS Jibas *in order to run optimally* when applied in SMAN 4. *The SIMS Jibas testing include white-box testing, black-box testing, compatibility testing, and usability testing. The compatibility testing of system that constructed can run at two operation system, that is operation system of windows and operation system of linux. The result of usability testing indicated that 85% of users were satisfied with the performance of the system and 100% of users require training to operate the system.*

Keywords : ERP, SIMS, Jibas, open source, customization

DAFTAR ISI

KATA PENGANTAR	i
ABSTRAK	iii
ABSTRACT	iv
DAFTAR ISI	v
DAFTAR TABEL	viii
DAFTAR GAMBAR	x
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Skripsi	3
1.5 Manfaat Skripsi	3
1.6 Sistematika Penulisan	3
BAB II DASAR TEORI	5
2.1 Rekayasa Perangkat Lunak	5
2.2 ERP Perusahaan	6
2.2.1 ERP Pendidikan	7
2.3 Reuse Oriented Model	7
2.4 Sistem Informasi Terintegrasi Manajemen Sekolah Jibas	9
2.5 Unified Modelling Language	10
2.5.1 Use Case Diagrams	11
2.5.2 Activity Diagrams	11
2.6 MySQL – Xamp	12
2.7 HTML/PHP	14
2.8 Pengujian Perangkat Lunak	14
2.8.1 Teknik Pengujian	14
2.8.1.1 White-box Testing	15
2.8.1.2 Black-box Testing	16
2.8.2 Strategi Pengujian	17

2.8.2.1 Pengujian Unit	17
2.8.2.2 Pengujian Integrasi	18
2.8.2.3 Pengujian Validasi	20
BAB III METODE PENELITIAN	22
3.1 Studi Literatur Dan Penyusunan Dasar Teori	23
3.2 Analisis Dan Perancangan Perangkat Lunak	23
3.2.1 Analisis Kebutuhan	23
3.2.2 Perancangan Perangkat Lunak	24
3.2.3 Implementasi	24
3.2.4 Pengujian Dan Evaluasi Sistem	25
3.3 Pengambilan Kesimpulan Dan Saran	25
BAB IV PERANCANGAN PERANGKAT LUNAK	26
4.1 Analisis Kebutuhan	26
4.1.1 Pemodelan Proses Bisnis (as-is)	26
4.1.2 Pemodelan Proses Sistem (to-be)	37
4.2 Rekapitulasi Kebutuhan Sistem Dan Perancangan Perangkat Lunak	49
4.3 Perbandingan Sistem Jibas Dengan Sistem yang Dibuat	50
4.4 Arsitektur Aplikasi SIMS Jibas	52
4.5 Arsitektur Penerapan SIMS Jibas	53
BAB V IMPLEMENTASI	54
5.1 Spesifikasi Lingkungan Sistem	54
5.1.1 Spesifikasi Lingkungan Perangkat Keras	54
5.1.2 Spesifikasi Lingkungan Perangkat Lunak	54
5.2 Implementasi Antarmuka	55
5.2.1 Antarmuka Aplikasi	55
5.2.2 Antarmuka Akademik	56
5.2.3 Antarmuka Kesiswaan	56
5.2.4 Antarmuka Perpustakaan	57
5.2.1 Antarmuka Kepegawaian	57
5.3 Implementasi Modul	58
5.4 Modul Akademik	58
5.5 Modul Kesiswaan	58



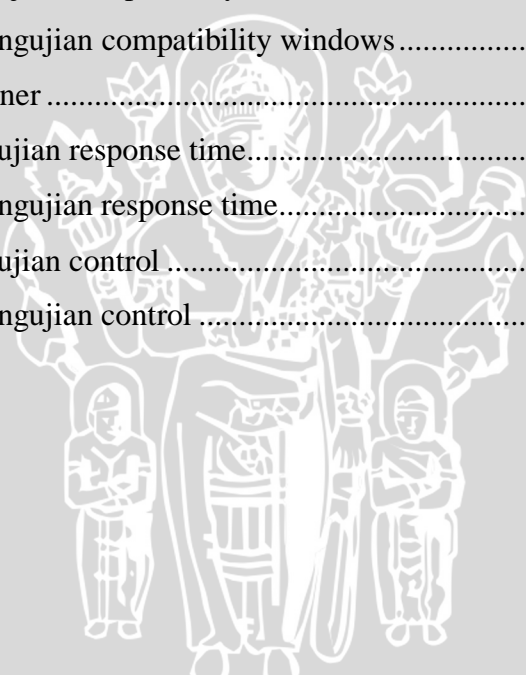
BAB VI PENGUJIAN DAN ANALISIS SISTEM	60
6.1 Pengujian.....	60
6.1.1 Pengujian Unit	60
6.1.1.1 Pengujian Unit untuk algoritma login.....	60
6.1.1.2 Pengujian Unit untuk algoritma index	62
6.1.2 Pengujian Integrasi.....	64
6.1.2.1 Pengujian integrasi untuk redirect	64
6.1.3 Pengujian validasi	66
6.1.3.1 Kasus Uji Pengolahan Data Penjadwalan.....	67
6.1.3.2 Kasus Uji Pengolahan Data Penilaian	67
6.1.3.3 Kasus Uji Pengolahan Data Penerimaan Siswa Baru	68
6.1.3.4 Kasus Uji Pengolahan Data Pengembalian Buku	69
6.1.3.5 Kasus Uji Pengolahan Data Pengaturan Pengguna.....	69
6.1.4 Pengujian non-fungsional	70
6.1.4.1 Pengujian availability	70
6.1.4.2 Pengujian compatibility	71
6.1.4.3 Pengujian usability.....	73
6.1.4.4 Pengujian response time	75
6.1.4.5 Pengujian Control	76
6.1.5 Analisis	78
6.1.5.1 Analisis pengujian unit	78
6.1.5.2 Analisis pengujian integrasi	78
6.1.5.3 Analisis pengujian validasi	79
6.1.5.4 Analisis pengujian availability	79
6.1.5.5 Analisis pengujian compatibility	79
6.1.5.6 Analisis pengujian usability.....	79
6.1.5.7 Analisis pengujian response time	79
6.1.5.8 Analisis pengujian control	80
BAB VII PENUTUP.....	81
7.1 Kesimpulan	81
7.2 Saran	81
DAFTAR PUSTAKA	82



DAFTAR TABEL

Tabel 2.1 Tabel perbandingan modul erp bisnis dengan erp pendidikan.....	7
Tabel 4.1 Tabel skenario model bisnis layanan akademik penjadwalan	28
Tabel 4.2 Tabel skenario model bisnis layanan akademik penilaian	29
Tabel 4.3 Tabel identifikasi aktor bisnis	30
Tabel 4.4 Tabel skenario model bisnis kesiswaan penerimaan siswa baru.....	31
Tabel 4.5 Tabel skenario model bisnis kesiswaan input presensi	33
Tabel 4.6 Tabel skenario model bisnis kesiswaan penempatan siswa	35
Tabel 4.7 Tabel skenario model bisnis kesiswaan input mutasi siswa	36
Tabel 4.8 Tabel identifikasi aktor sistem	38
Tabel 4.9 Tabel skenario sistem layanan akademik penjadwalan.....	39
Tabel 4.10 Tabel skenario sistem layanan akademik penilaian	40
Tabel 4.11 Tabel skenario sistem akademik pendataan guru dan pelajaran	42
Tabel 4.12 Tabel skenario sistem ubah pengguna	43
Tabel 4.13 Tabel skenario sistem pengaturan pengguna	44
Tabel 4.14 Tabel skenario sistem penerimaan siswa baru	46
Tabel 4.15 Tabel skenario sistem penempatan siswa.....	48
Tabel 4.16 Tabel daftar kebutuhan fungsional.....	49
Tabel 4.17 Tabel daftar kebutuhan non-fungsional	49
Tabel 4.18 Tabel perbandingan modul.....	51
Tabel 4.19 Tabel daftar fitur	51
Tabel 5.1 Spesifikasi perangkat keras komputer.....	54
Tabel 5.2 Spesifikasi perangkat lunak komputer	55
Tabel 5.3 Source code tampilan modul akademik	58
Tabel 5.4 Source code tampilan modul kesiswaa	59
Tabel 6.1 Pengujian unit untuk algoritma login.....	61
Tabel 6.2 Kasus uji pengujian unit login	62
Tabel 6.3 Pengujian unit untuk algoritma index	63
Tabel 6.4 Kasus uji pengujian index	64
Tabel 6.5 Pengujian integrasi untuk algoritma redirect	65

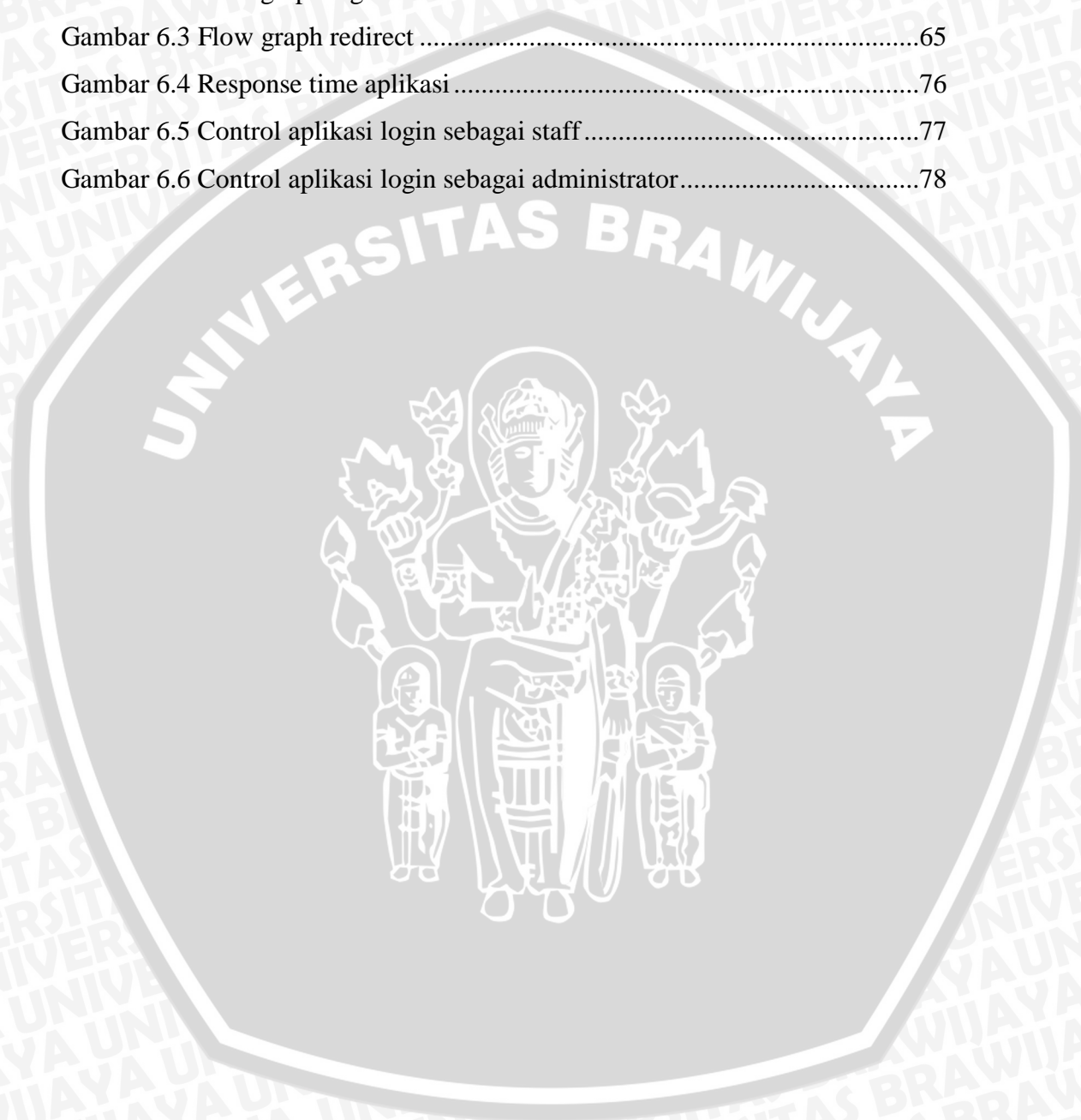
Tabel 6.6 Tabel kasus uji pengujian redirect	66
Tabel 6.7 Tabel kasus uji pengolahan data penjadwalan	67
Tabel 6.8 Tabel kasus uji pengolahan data penilaian.....	67
Tabel 6.9 Tabel kasus uji pengolahan data penerimaan siswa baru.....	68
Tabel 6.10 Tabel kasus uji pengolahan data pengembalian buku.....	69
Tabel 6.11 Tabel kasus uji pengaturan pengguna	69
Tabel 6.12 Daftar pengujian availability.....	71
Tabel 6.13 Skenario pengujian availability.....	71
Tabel 6.14 Daftar pengujian compatibility linux.....	71
Tabel 6.15 Skenario pengujian compatibility linux	72
Tabel 6.16 Daftar pengujian compatibility windows.....	72
Tabel 6.17 Skenario pengujian compatibility windows.....	72
Tabel 6.18 Hasil kuisisioner	73
Tabel 6.19 Daftar pengujian response time.....	75
Tabel 6.20 Skenario pengujian response time.....	76
Tabel 6.21 Daftar pengujian control	77
Tabel 6.22 Skenario pengujian control	77



DAFTAR GAMBAR

Gambar 2.1 Reuse-Oriented Model	8
Gambar 2.2 Pola implementasi jibas.....	10
Gambar 2.3 Transformasi flow chart ke flow graph	15
Gambar 2.4 Pengujian unit.....	18
Gambar 2.5 Pengujian integrasi top-down.....	19
Gambar 2.9 Pengujian integrasi bottom-up	20
Gambar 3.1 Diagram alur penelitian.....	22
Gambar 4.1 Use case model bisnis akademik (as-is).....	26
Gambar 4.2 Model bisnis layanan akademik penjadwalan.....	27
Gambar 4.3 Model bisnis layanan akademik penilaian	28
Gambar 4.4 Use case model bisnis kesiswaan (as-is).....	29
Gambar 4.5 Model bisnis layanan kesiswaan penerimaan siswa baru.....	31
Gambar 4.6 Model bisnis layanan kesiswaan input presensi.....	33
Gambar 4.7 Model bisnis layanan kesiswaan input penempatan siswa.....	34
Gambar 4.8 Model bisnis layanan kesiswaan input mutasi	36
Gambar 4.9 Use case layanan akademik (to-be).....	37
Gambar 4.10 Model sistem layanan akademik penjadwalan.....	38
Gambar 4.11 Model sistem layanan akademik penilaian.....	40
Gambar 4.12 Model sistem layanan akademik pendataan pelajaran dan guru ...	41
Gambar 4.13 Model sistem ubah pengguna.....	42
Gambar 4.14 Model sistem pengaturan pengguna.....	43
Gambar 4.15 Use case layanan kesiswaan (to - be).....	44
Gambar 4.16 Model sistem penerimaan siswa baru.....	46
Gambar 4.17 Model sistem penempatan siswa	47
Gambar 4.18 Arsitektur SIMS jibas.....	52
Gambar 4.19 Arsitektur Penerapan SIMS Jibas.....	53
Gambar 5.1 Antarmuka Aplikasi	55
Gambar 5.2 Antarmuka Akademik	56
Gambar 5.3 Antarmuka Kesiswaan.....	56

Gambar 5.4 Antarmuka Perpustakaan.....	57
Gambar 5.5 Antarmuka Kepegawaian.....	57
Gambar 6.1 Flow graph algoritma login.....	61
Gambar 6.2 Flow graph algoritma index.....	63
Gambar 6.3 Flow graph redirect.....	65
Gambar 6.4 Response time aplikasi.....	76
Gambar 6.5 Control aplikasi login sebagai staff.....	77
Gambar 6.6 Control aplikasi login sebagai administrator.....	78



BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi yang sangat cepat didorong oleh meningkatnya kebutuhan informasi yang semakin kompleks di berbagai aspek kehidupan. Sistem informasi sebagai infrastruktur yang mampu menangani pengelolaan data dan informasi menjadi sangat penting dan keharusan di era globalisasi saat ini. Untuk menjawab berbagai tantangan dan kebutuhan sistem yang ada, sistem informasi terintegrasi terus dikembangkan beberapa tahun belakangan ini.

Sistem informasi terintegrasi adalah sistem yang memungkinkan organisasi mengintegrasikan dan mengkoordinasikan proses bisnis yang mereka miliki. Ciri dari sistem informasi terintegrasi adalah tingkat keterpaduan (integrasi) yang tinggi untuk mengakomodasi kebutuhan data dan informasi yang terpadu. Perbandingan sistem informasi terintegrasi dan sistem informasi tidak terintegrasi adalah sistem informasi terintegrasi cepat dan akurat dalam memperoleh informasi, karena di sistem informasi terintegrasi integritas data antar modul yang ditekankan. [MJO-13].

Dalam dunia bisnis, salah satu sistem informasi terintegrasi adalah sistem ERP (*Enterprise Resource Planning*). Sistem ERP berkembang pesat di perusahaan produksi barang, namun seiring dengan perkembangan zaman ERP dapat diimplementasikan pada organisasi bisnis lain, misalnya pada organisasi pendidikan yang disebut dengan ERP pendidikan. Sistem ERP pada organisasi bisnis mempunyai beberapa modul, misalnya modul operasi, modul keuangan dan akuntansi, serta modul sumber daya manusia. Modul yang terdapat di ERP bisnis dapat dipetakan menjadi modul ERP pendidikan [AGS-09]. Teknologi ERP pendidikan dapat dimanfaatkan lebih lanjut di dunia pendidikan.

Pengelolaan administrasi dan manajemen di SMAN 4 Jember masih manual dan kurang begitu baik, sehingga dibutuhkan suatu sistem informasi yang dapat mengotomatisasi administrasi dan manajemen di SMAN 4 Jember. Terdapat beberapa unit di SMAN 4 Jember antarlain unit akademik, unit kesiswaan, unit

kepegawaian, dan unit perpustakaan. Keempat unit tersebut harus mempunyai integritas data yang baik agar proses bisnis yang ada dapat berjalan secara efektif dan efisien, hal ini tidak didapatkan dalam sistem informasi tradisional yang hanya mempunyai satu aplikasi. Dibutuhkan beberapa aplikasi yang saling terintegrasi dalam satu aplikasi besar, ini merupakan salah satu konsep dari sistem ERP. Oleh karena itu, skripsi ini bertujuan untuk menganalisis dan menerapkan sistem ERP pendidikan di SMAN 4 Jember.

Salah satu sistem ERP pendidikan yang dapat diterapkan di SMAN 4 Jember yaitu Sistem Informasi Manajemen Sekolah (SIMS) Jibas. SIMS Jibas memiliki beberapa aplikasi yang terintegrasi, seperti aplikasi akademik, keuangan, perpustakaan, kepegawaian, dan pelaporan. SIMS Jibas bersifat *open source* sehingga dapat dikustomisasi sesuai dengan kebutuhan yang ada di organisasi pendidikan. Namun tidak semua kebutuhan di SMAN 4 Jember dapat terpenuhi oleh SIMS Jibas, oleh karena itu penulis melakukan suatu analisa kebutuhan di SMAN 4 Jember yang hasilnya akan diperlukan untuk analisis komponen dan desain kustomisasi SIMS Jibas, agar SIMS Jibas dapat berjalan dengan optimal pada saat penerapan di SMAN 4 Jember.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, maka skripsi ini mempunyai rumusan masalah, yaitu bagaimana menerapkan konsep SIMS Jibas di SMAN 4 Jember yang meliputi aktivitas sebagai berikut :

- a. Analisis kebutuhan SMAN 4 Jember.
- b. Perancangan dan Implementasi SIMS Jibas di SMAN 4 Jember.
- c. Pengujian terhadap perangkat lunak SIMS Jibas yang diterapkan di SMAN 4 Jember.

1.3 Batasan Masalah

Berdasarkan latar belakang dan rumusan masalah yang telah dipaparkan maka skripsi ini mempunyai batasan – batasan masalah sebagai berikut:

1. Pembahasan difokuskan pada analisis kebutuhan SMAN4 Jember dengan menggunakan Jibas sebagai sistem informasinya.

2. Pengembangan perangkat lunak dilakukan dengan menggunakan bahasa pemrograman PHP.
3. DBMS (*Data Base Management System*) yang digunakan dalam pengembangan perangkat lunak ini adalah MySQL.
4. Implementasi yang dihasilkan dari sistem berupa sistem informasi yang terintegrasi dari 4 bagian di SMAN 4 Jember, yakni akademik, perpustakaan, kepegawaian, dan kesiswaan.
5. Modul yang digunakan antara lain modul akademik, modul perpustakaan, modul kepegawaian, dan modul kesiswaan (sesuai kebutuhan SMAN 4 Jember).

1.4 Tujuan Skripsi

Sesuai dengan latar belakang dan rumusan masalah tersebut, maka tujuan penulisan tugas akhir ini yaitu menganalisis dan menerapkan sistem informasi manajemen yang terintegrasi di organisasi pendidikan yakni SIMS Jibas dengan menyesuaikan kebutuhan di SMAN 4 Jember.

1.5 Manfaat Skripsi

Pengajuan skripsi ini diharapkan mempunyai manfaat yang baik dan berguna bagi pembaca dan penulis, adapun manfaat yang diharapkan adalah sebagai berikut :

- a. Bagi pengguna
 1. Dengan fasilitas website yang dibangun dapat memudahkan pihak SMAN 4 Jember untuk melaksanakan kegiatan – kegiatan secara efektif dan efisien.
 2. Menyediakan sarana informasi dan proses bisnis yang kompleks bagi semua pengguna.

1.6 Sistematika Penulisan

Sistematika penulisan ditunjukkan untuk memberikan gambaran dan uraian dari penulisan skripsi ini secara garis besar yang meliputi beberapa bab, sebagai berikut :

BAB I : Pendahuluan

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

BAB II : Kajian Pustaka dan Dasar Teori

Menguraikan tentang dasar teori dan referensi rekayasa perangkat lunak, ERP perusahaan, ERP pendidikan, *Reuse-Oriented Model*, Sistem Informasi Manajemen Sekolah Jibas, analisis dan perancangan menggunakan UML, basis data MySQL, bahasa pemrograman HTML/PHP dan pengujian perangkat lunak.

BAB III : Metodologi Penelitian

Membahas metode yang digunakan dalam penulisan yang terdiri dari studi literatur dan penyusunan dasar teori, analisis dan perancangan perangkat lunak, implementasi, pengujian dan evaluasi sistem, serta pengambilan kesimpulan dan saran.

BAB IV : Perancangan Perangkat Lunak

Membahas analisis kebutuhan dan perancangan yang sesuai dengan teori yang ada.

BAB V : Implementasi

Membahas tentang implementasi dari sistem aplikasi.

BAB VI : Pengujian dan Analisis

Memuat hasil pengujian dan analisis terhadap sistem yang telah direalisasikan.

BAB VII : Penutup

Memuat kesimpulan yang diperoleh dari pembuatan dan pengujian program, serta saran-saran untuk pengembangan lebih lanjut.

BAB II

DASAR TEORI

Pada bab ini dibahas mengenai dasar teori yang digunakan untuk menunjang penulisan skripsi mengenai analisis dan penerapan SIMS Jibas di SMAN 4 Jember dengan beberapa dasar teori yang dimaksud diantaranya yaitu rekayasa perangkat lunak, ERP perusahaan, ERP pendidikan, Reuse-Oriented Model, sistem informasi terintegrasi Manajemen Sekolah Jibas, analisis dan perancangan menggunakan UML, basis data MySql, bahasa pemrograman HTML/PHP dan pengujian perangkat lunak.

2.1 Rekayasa Perangkat Lunak

Rekayasa perangkat lunak adalah disiplin ilmu yang membahas semua aspek produksi perangkat lunak, mulai dari tahap awal spesifikasi sistem sampai pemeliharaan sistem setelah digunakan [SOI- 11:23].

Secara fungsional perangkat lunak yang baik adalah perangkat lunak yang bisa menjalankan semua perintah yang dilakukan oleh pengguna namun, dalam menjaga persaingan kesuksesan suatu perangkat lunak perlu untuk diperhatikan aspek – aspek nonfungsional perangkat lunak. Berikut adalah beberapa aspek nonfungsional yang perlu diperhatikan menurut Sommerville antara lain [SOI- 11] :

1. Dapat terus dipelihara setelah perangkat lunak selesai dibuat seiring berkembangnya teknologi dan lingkungan (*maintainability*)
2. Dapat diandalkan dengan proses bisnis yang dijalankan dan perubahan yang terjadi (*dependability and robust*)
3. Efisien dari segi sumber daya dan penggunaan
4. Kemampuan untuk dipakai sesuai dengan kebutuhan (*usability*)

Kesimpulan dari beberapa aspek di atas, perangkat lunak yang baik adalah perangkat lunak yang mampu memenuhi kebutuhan dari *customer* atau pemakai dari perangkat lunak (*user*) baik secara fungsional maupun nonfungsional sehingga perangkat lunak harus berorientasi pada pelanggan atau pemakai

perangkat lunak, bukan berorientasi pada pembuat atau pengembang perangkat lunak.

2.2 ERP Perusahaan

ERP merupakan sebuah kerangka kerja transaksi berskala perusahaan yang menghubungkan proses pemesanan barang, manajemen inventarisasi dan kontrol, perencanaan distribusi dan produksi, dan keuangan. ERP bekerja sebagai kekuatan lintas fungsional perusahaan yang mengintegrasikan dan mengotomatisasi berbagai proses bisnis internal dan sistem informasi termasuk manufaktur, logistik, distribusi, penggunaan, keuangan, dan sumber daya manusia dari sebuah perusahaan.

Fokus utama dari ERP adalah integrasi sistem, integrasi sistem yang dimaksud menyangkut hal – hal yang sebagai berikut antara lain [DE0-02:02]:

1. Penghubungan antar berbagai aliran proses bisnis.
2. Teknik komunikasi.
3. Sinkronisasi operasi bisnis.
4. Koordinasi operasi bisnis.

Adapun beberapa keuntungan dari penggunaan sistem informasi terpadu dalam konsep ERP ini antara lain dapat disebutkan sebagai berikut [DE0-00:02]:

- a. ERP menawarkan sistem terintegrasi di dalam perusahaan, sehingga proses dan pengambilan keputusan dapat dilakukan secara lebih efektif dan efisien.
- b. ERP juga memungkinkan melakukan integrasi secara global. Halangan yang tadinya berupa perbedaan valuta, perbedaan bahasa, dan perbedaan budaya, dapat dijumpai secara otomatis, sehingga data dapat diintegrasikan.
- c. ERP tidak hanya memadukan data dan orang, tetapi juga menghilangkan kebutuhan pemutakhiran dan pembetulan banyak sistem komputer yang terpisah.

2.2.1 ERP Pendidikan

ERP pendidikan merupakan terobosan terbaru yang dibangun untuk mendorong berkembangnya sistem informasi yang berkembang pesat. ERP biasanya digunakan di perusahaan-perusahaan *manufacturing* dan jasa, namun seiring dengan berkembangnya zaman ERP mulai merambah ke industri – industri kecil seperti pendidikan.

Sistem *ERP* perusahaan mempunyai beberapa modul yang pokok sebagai wilayah bisnis diantaranya *Human Resource, Sales and Marketing, Inventory, Costing, Manufacturing, Sales and Distribution, Finance*, sedangkan sistem ERP pendidikan mempunyai beberapa modul antarlain *Administration, University Correspondence, Library, Academic Schedule, Curriculum, On-Campus and Off-Campus, Finance*. Setiap modul saling terintegrasi karena kekuatan dari ERP adalah integrasi. Tabel 2.1 adalah perbandingan modul antara ERP perusahaan dan ERP pendidikan.

Tabel 2. 1 Perbandingan Modul ERP bisnis dengan ERP Pendidikan

Subject of Comparison	ERP	Univ./Institution
<i>Beneficiary</i>	Customers	Students & Staff
<i>Workers</i>	Employee	Staff & Administration
<i>Modules</i>	HR	Administration
	Sales & Marketing	University Correspondence
	Inventory	Library
	Costing	Academic Schedule
	Manufacturing	Curriculum
	Sales & Distribution	On-Campus & Off-Campus
	Finance	Finance

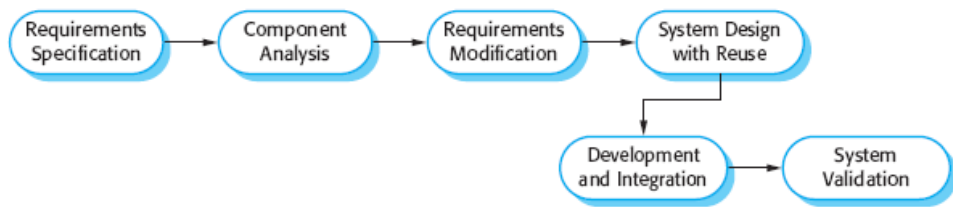
Sumber :[AGS-09]

2.3 Reuse-Oriented Model

Model proses rekayasa perangkat lunak dibentuk dengan menyesuaikan sifat dari aplikasi yang akan dibuat. Salah satu dari model proses yang digunakan adalah *Reuse-oriented*. *Reuse-oriented* merupakan rekayasa perangkat lunak yang

menggunakan pendekatan berbasis tersedianya jumlah komponen *reusable* yang signifikan. Proses pengembangan sistem berfokus pada integrasi komponen-komponen ini ke dalam sistem daripada mengembangkan sistem dari dasar [SOI-11:20].

Model proses secara umum dari pengembangan berbasis *reuse* dapat dilihat pada gambar 2.1 berikut :



Gambar 2.1 *Reuse-Oriented Model*

Sumber : [SOI – 11:35]

Secara sepintas model di atas sama dengan model perancangan perangkat lunak biasa, namun sebenarnya terdapat perbedaan pada pertengahan proses, yaitu dengan adanya proses *reuse-oriented*. Menurut Sommerville [SOI - 11] proses ini antara lain:

1. Analisis komponen. Dari spesifikasi kebutuhan yang telah ditentukan, dicari komponen yang dapat mengimplementasi spesifikasi tersebut. Biasanya tidak ada komponen yang benar-benar mampu memenuhi spesifikasi dan mungkin hanya beberapa fungsi yang digunakan dari komponen tersebut.
2. Modifikasi kebutuhan. Pada proses ini, kebutuhan dianalisis menggunakan informasi tentang komponen yang akan digunakan. Kemudian, kebutuhan ini dimodifikasi untuk merefleksikan komponen yang tersedia. Jika terdapat modifikasi yang tidak mungkin untuk diterapkan, proses analisis komponen dapat dilakukan kembali untuk mencari solusi alternatif.
3. Desain sistem dengan *reuse*. Pada proses ini, sistem yang telah ada digunakan kembali (*reused*) sesuai dengan kebutuhan yang ada sekarang.
4. Pengembangan dan integrasi.

Sommerville [SOI - 11] juga menyebutkan tiga tipe komponen perangkat lunak yang dapat digunakan dalam proses *reuse-oriented*, yaitu :

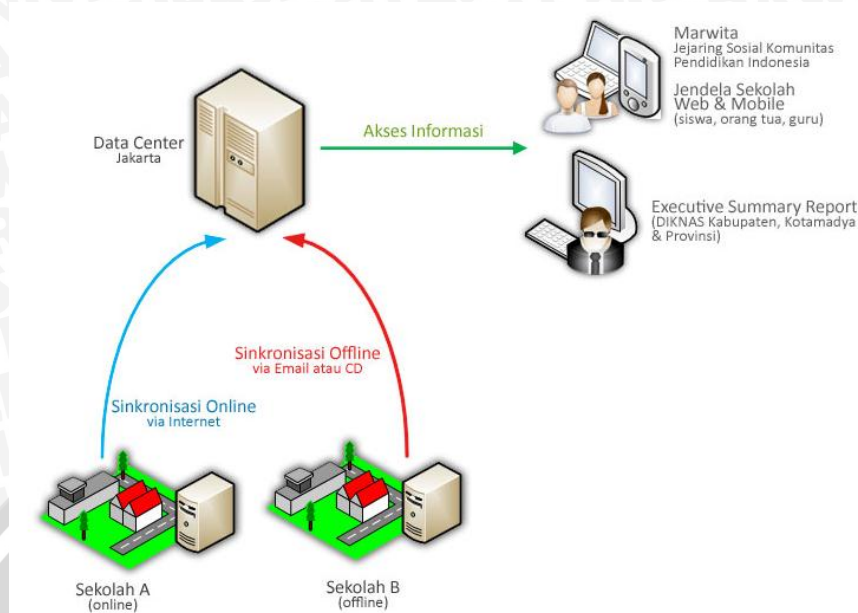
1. *Web service* yang dikembangkan berdasarkan *service standard* dan tersedia untuk digunakan secara *remote*.
2. Koleksi *object* yang dikembangkan sebagai *package* untuk diintegrasikan dengan sebuah *framework* komponen seperti *.NET* dan *J2EE*.
3. Sistem perangkat lunak yang berdiri sendiri yang dapat dikonfigurasi untuk digunakan dalam lingkungan tertentu.

Rekayasa perangkat lunak yang *reuse-oriented* memiliki keuntungan dalam mengurangi jumlah perangkat lunak yang dikembangkan dan mengurangi biaya serta resiko. *Reuseable* yang digunakan nantinya adalah bisnis proses sistem informasi Jibas yang dikembangkan untuk merancang aplikasi agar dapat digunakan kembali pada tahap pengembangan.

2.4 Sistem Informasi Terintegrasi Manajemen Sekolah Jibas

Jibas merupakan Jaringan Bersama Antar Sekolah yang mempunyai salah satu bentuk implementasi yakni sistem informasi sekolah terpadu yang mengintegrasikan seluruh bagian fungsionalitas yang ada di organisasi pendidikan. SIMS Jibas mempunyai bentuk aplikasi *Client-Server* yaitu satu komputer menyimpan data dan komputer lain mengaksesnya, dapat dijadikan internet web based bagi sekolah yang memiliki web server sendiri. SIMS Jibas bersifat *open source* sehingga kode SIMS Jibas dapat dikostumisasi sesuai dengan kebutuhan [JBS-13].

Dalam rangka menyusun pola implementasi Jibas di daerah, Jibas berpegang pada 3 prinsip yaitu masyarakat memperoleh manfaat nyata dari Jibas, meminimalkan anggaran yang diperlukan untuk implementasi Jibas dan penyelenggaraan Jibas menjadi program berkelanjutan yang melibatkan peran serta aktif sumber daya daerah. Gambar 2.2 merupakan pola implementasi Jibas



Gambar 2.2 Pola implementasi jibas

Sumber : [JBS-13]

Server untuk pengelolaan Jibas ditempatkan di setiap sekolah. Server ini cukup menggunakan PC (*Personal Computer*) desktop biasa. Hal ini untuk mempermudah penggunaan dan perawatan. Untuk menghemat anggaran pengiriman data pengelolaan sekolah ke data center Jibas dapat menggunakan jaringan internet yang tersedia. Jadi tidak diperlukan tambahan perangkat keras atau jasa lainnya. Bagi sekolah yang belum terjangkau internet dapat mengirimkan data pengelolaannya secara *offline* melalui CD/USB disk atau email [JBS-13].

2.5 Unified Modelling Language

Unified Modelling Language (UML) adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung yang memudahkan untuk dipahami [IBM-01:6-9]. *UML* ada untuk menspesifikasikan, menggambarkan, membangun, dan mendokumentasi sistem perangkat lunak yang dikembangkan. *UML* hanya berfungsi untuk melakukan pemodelan, sehingga penggunaan *UML* tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya *UML* paling banyak digunakan pada metodologi berorientasi objek [ROS-11].

Kategori *UML* dibedakan menjadi 3 kategori besar yakni:

1. *Structure diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
2. *Behavior diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
3. *Interaction diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antarsubsystem pada suatu sistem.

2.5.1 Use Case Diagram

Use case merupakan pemodelan untuk perilaku (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat [ROS-11].

Use case dapat dibagi menjadi 2 yakni *use case* bisnis, dan *use case* sistem. *Use case* bisnis menggambarkan aktifitas bisnis yang terjadi di organisasi dan *use case* sistem menggambarkan aktifitas yang dilakukan di sistem. Syarat penamaan pada *use case* adalah nama didefinisikan sesederhana mungkin dan mudah dipahami. Dua hal utama *use case* adalah pendefinisian apa yang disebut aktor dan *use case* [ROS-11].

1. Aktor dibagi menjadi dua macam yaitu aktor bisnis dan aktor sistem. Aktor bisnis merepresentasikan orang atau proses yang berada di luar bisnis, namun tetap berinteraksi dengan bisnis, sedangkan aktor sistem merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat.
2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai *unit-unit* yang saling bertukar pesan antar unit atau aktor.

2.5.2 Activity Diagram

Activity diagram menggambarkan aliran aktivitas dalam sistem yang sedang dirancang. Hal – hal yang perlu diperhatikan dalam perancangan *activity diagrams* adalah bagaimana masing-masing aliran berawal, *decision* yang

mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagrams* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi alasannya karena *activity diagrams* merupakan *state* diagram khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan *behaviour* internal sebuah sistem secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Gambaran proses bisnis dan urutan aktivitas dalam sebuah proses yang digunakan pada *business modeling* dapat memperlihatkan urutan aktifitas proses bisnis. Struktur diagram ini mirip *Flowchart* atau *Data Flow Diagram* pada perancangan terstruktur sehingga sangat bermanfaat apabila kita membuat diagram ini terlebih dahulu dalam memodelkan sebuah proses untuk membantu memahami proses secara keseluruhan [MEI -13].

2.6 MySQL - Xampp

MySQL- Xampp adalah sebuah perangkat lunak sistem manajemen basis data *SQL* atau *DBMS* yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. Aplikasi ini menggunakan basis data *MySQL* karena *MySQL* tersedia sebagai perangkat lunak gratis dibawah lisensi *GNU General Public License* (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus yang penggunaannya tidak cocok dengan penggunaan GPL [WKI-13]. Menurut [WKI – 13] *MySQL-Xampp* memiliki beberapa ke istimewa, antara lain:

- a. *Kompatibilitas*. *MySQL* dapat berjalan stabil pada berbagai sistem operasi seperti *Windows*, *Linux*, *FreeBSD*, *Mac Os X Server*, *Solaris*, *Amiga*, dan masih banyak lagi.
- b. *Open Source*. *MySQL-Xampp* di distribusikan secara *open source*, dibawah lisensi GPL sehingga dapat digunakan secara cuma-cuma.
- c. *'Multiuser'*. *MySQL-Xampp* dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

- d. *'Performance tuning'*. *MySQL-Xampp* memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak *SQL* per satuan waktu.
- e. Tipe Data. *MySQL* memiliki tipe data yang sangat kompleks, seperti *signed* atau *unsigned integer*, *float*, *double*, *char*, *text*, *date*, *timestamp*, dan lain-lain.
- f. Perintah dan Fungsi. *MySQL-Xampp* memiliki operator dan fungsi secara penuh yang mendukung perintah *Select* dan *Where* dalam perintah (*query*).
- g. Keamanan. *MySQL-Xampp* memiliki beberapa lapisan keamanan seperti *level subnetmask*, nama *host*, dan izin akses *user* dengan sistem perizinan yang mendetail serta sandi terenkripsi.
- h. Skalabilitas dan Pembatasan. *MySQL* mampu menangani basis data dalam skala besar, dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas *indeks* yang dapat di tampung mencapai 32 *indeks* pada tiap tabelnya.
- i. Konektivitas. *MySQL* dapat melakukan koneksi dengan klien menggunakan protokol *TCP/IP*, *Unix socket (UNIX)*, atau *Named Pipes(NT)*.
- j. Antar Muka. *MySQL-Xampp* memiliki antarmuka terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi *Application Programming Antarmuka*.
- k. Klien dan Peralatan. *MySQL-Xampp* dilengkapi dengan berbagai peralatan (*tools*) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk *online*.
- l. Struktur tabel. *MySQL* memiliki struktur tabel yang lebih fleksibel dalam menangani *ALTER TABLE*, dibandingkan basis data lainnya semacam *Postgre SQL* ataupun *Oracle* [WKI-13].

Oleh karena itu keutamaan dari *MySQL-Xampp* di atas inilah yang menjadikan skripsi ini menggunakan *MySQL- Xampp*.

2.7 HTML/PHP

HyperText Markup Language (HTML) adalah sebuah bahasa yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi di dalam sebuah penjelajah web internet dan pemformatan hiperteks sederhana yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi. Dengan kata lain, berkas yang dibuat dalam perangkat lunak pengolah kata dan disimpan dalam format ASCII normal sehingga menjadi halaman web dengan perintah-perintah HTML [WIK-13].

PHP merupakan singkatan rekursif (akronim berulang) dari PHP Hypertext Preprocessor. PHP adalah bahasa pemrograman script yang paling banyak dipakai saat ini atau dalam kata lain bisa diartikan sebuah bahasa pemrograman web yang bekerja di sisi server (server side scripting) yang dapat melakukan konektivitas pada basis data dimana hal itu tidak dapat dilakukan hanya dengan menggunakan sintaks-sintaks HTML biasa. PHP banyak dipakai untuk memprogram situsweb dinamis, walaupun tidak tertutup kemungkinan digunakan untuk pemakaian lain [NUR-13].

2.8 Pengujian Perangkat Lunak

Pengujian (*testing*) arsitektur dari perangkat lunak prosedural menghasilkan sekumpulan fungsi yang digunakan untuk menyelesaikan suatu permasalahan. Setiap elemen sistem (subsistem dan modul) melakukan fungsi yang membantu untuk mencapai kebutuhan sistem. Hal ini sangat penting untuk menguji sebuah sistem prosedur pada berbagai macam level yang berbeda dalam sebuah usaha untuk menemukan kesalahan-kesalahan yang mungkin terjadi dari kolaborasi kelas-kelas dan komunikasi subsistem melewati arsitektural layer [PRE-01:631].

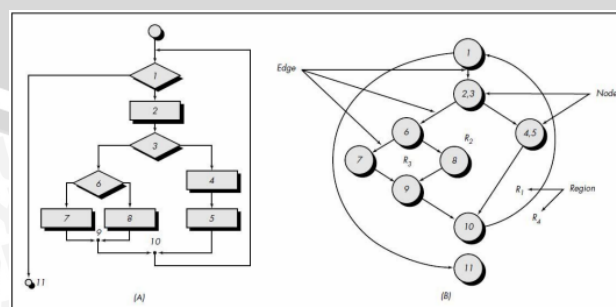
2.8.1 Teknik Pengujian

Pengujian perangkat lunak memerlukan perancangan kasus uji (test case) agar dapat menemukan kesalahan dalam waktu singkat dan usaha minimum. Berbagai macam metode perancangan kasus uji telah berevolusi. Metode-metode ini menyediakan developer pendekatan sistematis untuk pengujian. Terlebih lagi

metode-metode ini menyediakan mekanisme yang dapat membantu memastikan kelengkapan dari pengujian dan menyediakan kemungkinan tertinggi untuk menemukan kesalahan-kesalahan dalam perangkat lunak [PRE-01:443]. Teknik atau metode perancangan kasus uji yang digunakan adalah *white-box testing* dan *black-box testing*.

2.8.1.1. White-box Testing

White-box testing atau *glass-box testing* merupakan sebuah metode perancangan kasus uji yang menggunakan struktur kontrol dari perancangan prosedural untuk memperoleh kasus uji [PRE-01:444]. Ada dua jenis pengujian yang termasuk *white-box testing* yaitu *basis path testing* dan *control structure testing*. Pada skripsi ini menggunakan *basis path testing* yang diusulkan pertama kali oleh Tom McCabe [PRE-01:445]. *Basis path testing* ini memungkinkan perancang kasus uji memperoleh ukuran kompleksitas logis dari sebuah perancangan prosedural dan menggunakan pengukuran ini sebagai pedoman untuk mendefinisikan *basis set* dari jalur eksekusi (*execution path*). *Test case* yang dilakukan untuk menggunakan *basis set* tersebut dijamin untuk menggunakan setiap statement di dalam program paling tidak sekali selama pengujian. Sebelum metode *basis path* dapat diperkenalkan, notasi sederhana untuk representasi aliran kontrol yang disebut diagram alir (*flow graph*) harus diperkenalkan. Setiap representasi desain prosedural yang berupa *flow chart* dapat diterjemahkan ke dalam *flow graph*. Gambar 2.3 menunjukkan transformasi *flow chart* ke *flow graph*. Setelah *flow graph* didefinisikan maka harus ditentukan ukuran kompleksitas (*cyclomatic complexity*).



Gambar 2.3 Transformasi *flow chart* ke *flow graph*

Sumber : [PRE-01:447]

Cyclomatic complexity adalah metrik perangkat lunak yang memberikan pengukuran kuantitatif terhadap kompleksitas logis suatu program. Bila metrik ini digunakan dalam konteks metode pengujian *basis path*, maka nilai yang terhitung untuk *cyclomatic complexity* menentukan jumlah jalur independen (*independent path*) dalam *basis set* suatu program dan memberi batas atas bagi jumlah pengujian yang harus dilakukan untuk memastikan bahwa semua statement telah dieksekusi sedikitnya satu kali. Jalur independen adalah jalur yang melalui program yang mengenalkan sedikitnya satu rangkaian statement proses baru atau suatu kondisi baru. Untuk menentukan *cyclomatic complexity* bisa dilakukan dengan beberapa cara, diantaranya [PRE-01:448]:

1. Jumlah region pada *flow graph* sesuai dengan *cyclomatic complexity*.
2. *Cyclomatic complexity* $V(G)$, untuk grafik G adalah $V(G) = E - N + 2$, dimana E adalah jumlah edge, dan N adalah jumlah node.
3. $V(G) = P + 1$, dimana P adalah jumlah predicate node yaitu node yang merupakan kondisi (ada 2 atau lebih edge akan keluar node ini).

2.8.1.2 Black-box Testing

Black-box testing atau *behavioral testing* berfokus pada persyaratan fungsional perangkat lunak [PRE-01:459]. Dengan demikian, pengujian *black-box* memungkinkan perekrut perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk semua program. Pengujian *black-box* bukan merupakan alternatif dari teknik *white-box*, tetapi merupakan pendekatan komplementer yang kemungkinan besar mampu mengungkap kelas kesalahan daripada metode *white-box*.

Pengujian *black-box* berusaha menemukan kesalahan dalam kategori berikut :

1. Fungsi-fungsi yang tidak benar atau hilang.
2. Kesalahan interface.
3. Kesalahan dalam struktur data atau akses basis data eksternal.
4. Kesalahan kinerja.
5. Inisialisasi dan kesalahan terminasi.

Tidak seperti pengujian *white-box*, yang dilakukan pada saat awal proses pengujian, pengujian *black-box* cenderung diaplikasikan selama tahap akhir pengujian. Karena pengujian *black-box* tidak memperhatikan struktur kontrol, maka perhatian berfokus pada domain informasi. Pengujian didesain untuk menjawab pertanyaan-pertanyaan berikut :

- a. Bagaimana validitas fungsional diuji ?
- b. Kelas input apa yang akan membuat test case menjadi baik ?
- c. Apakah sistem sangat sensitif terhadap harga input tertentu ?
- d. Bagaimana batasan dari suatu data diisolasi ?
- e. Kecepatan dan volume data apa yang dapat ditolerir oleh sistem ?
- f. Apa pengaruh kombinasi tertentu dari data terhadap operasi sistem ?

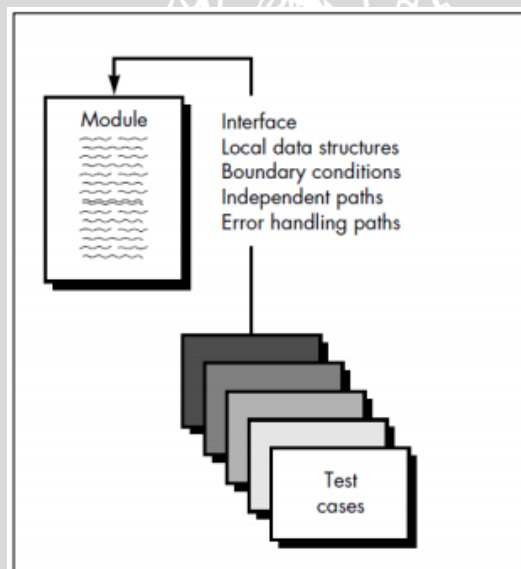
2.8.2 Strategi Pengujian

Strategi untuk pengujian perangkat lunak mengintegrasikan metode desain test case perangkat lunak ke dalam sederetan langkah yang direncanakan dengan baik, dan hasilnya adalah konstruksi perangkat lunak yang berhasil [PRE-01:477]. Sejumlah strategi pengujian perangkat lunak telah diusulkan di dalam literatur. Strategi pengujian harus mengakomodasi pengujian tingkat rendah yang diperlukan untuk membuktikan bahwa segmen kode sumber yang kecil telah diimplementasikan dengan tepat, demikian juga pengujian tingkat tinggi yang memvalidasi fungsi-fungsi sistem mayor yang berlawanan dengan kebutuhan pelanggan. Proses pengujian dimulai dengan pengujian yang berfokus pada setiap modul secara individual (*unit testing*), dilanjutkan dengan pengujian integrasi (*integration testing*) dan berakhir pada pengujian validasi (*validation testing*) [PRE-01:481].

2.8.2.1 Pengujian Unit

Pengujian unit berfokus pada usaha verifikasi pada inti terkecil dari desain perangkat lunak, yakni modul atau fungsi. Dengan menggunakan gambaran desain prosedural sebagai panduan, jalur kontrol yang penting diuji untuk mengungkap kesalahan di dalam batas modul tersebut. Kompleksitas relatif dari pengujian dan kesalahan yang diungkap dibatasi oleh ruang lingkup batasan yang

dibangun untuk pengujian unit. Pengujian unit biasanya berorientasi pada *white-box*, dan langkahnya dapat dilakukan secara paralel untuk model bertingkat [PRE-01:485]. Pengujian yang terjadi sebagai bagian dari inti digambarkan secara skematis pada gambar 2.4. Interface modul diuji untuk memastikan bahwa informasi secara tepat mengalir masuk dan keluar dari inti program yang diuji. Struktur data lokal diuji untuk memastikan bahwa data yang tersimpan secara temporal dapat tetap menjaga integritasnya selama semua langkah di dalam suatu algoritma dieksekusi. Kondisi batas diuji untuk memastikan bahwa modul beroperasi dengan tepat pada batas yang ditentukan untuk membatasi pemrosesan. Semua jalur independen (jalur dasar) yang melalui struktur kontrol dipakai sedikitnya satu kali. Dan akhirnya, penanganan kesalahan uji [PRE-01:485].



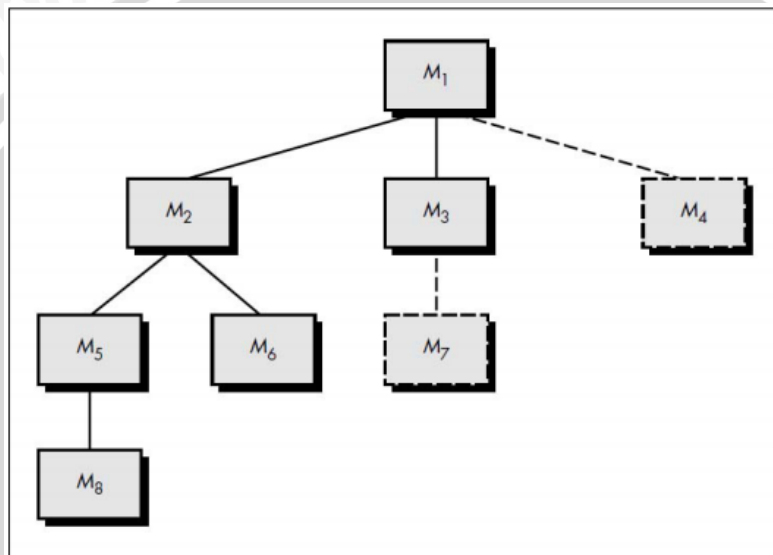
Gambar 2.4 Pengujian unit

Sumber : [PRE-01:487]

2.8.2.2 Pengujian Integrasi

Pengujian integrasi adalah teknik sistematis untuk mengkonstruksi struktur program sambil melakukan pengujian untuk mengungkap kesalahan sehubungan dengan interfacing. Sasarannya adalah untuk mengambil modul yang dikenai pengujian unit dan membangun struktur program yang telah ditentukan oleh desain. *integration testing* berorientasi *black box* dan mempunyai dua pola

pengujian yaitu integrasi *top-down* (*top-down integration*) dan integrasi *bottom-up* (*bottom-up integration*) [PRE-01:488]. Integrasi *top-down* adalah pendekatan inkremental terhadap struktur program. Modul diintegrasikan dengan menggerakkan ke bawah melalui hirarki kontrol, dimulai dengan modul kontrol utama (program utama). Subordinat program terhadap modul kontrol utama digabungkan ke dalam struktur dengan cara *depth-first* atau *breadth-first* [PRE-01:489]. Gambar 2.5 menunjukkan pola pengujian integrasi *top-down*.



Gambar 2.5 Pengujian integrasi *top-down*

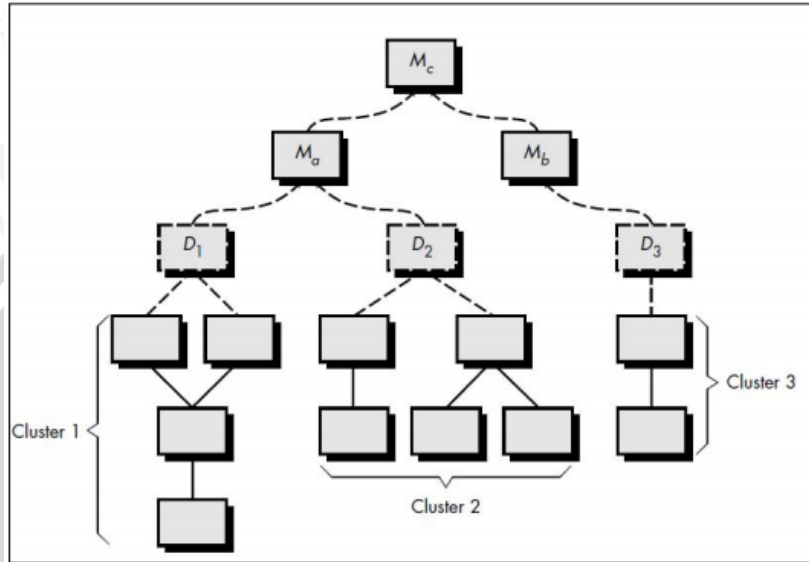
Sumber : [PRE-01:489]

Proses integrasi *top-down* dilakukan dalam lima langkah [PRE-01:489]:

1. Modul kontrol utama digunakan sebagai test driver dan stub digunakan untuk menggantikan semua komponen dibawahnya.
2. Pemilihan pendekatan integrasi yang diinginkan (*depth* atau *breadth first*).
3. Pengujian dikerjakan untuk setiap komponen yang diintegrasikan.
4. Stub digantikan dengan komponen yang sebenarnya setelah menyelesaikan serangkaian pengujian.
5. Proses akan terus dilakukan sampai membentuk sebuah perangkat lunak yang utuh.

Pengujian integrasi *bottom-up* memulai konstruksi dan pengujian dengan modul atomik (modul pada tingkat paling rendah pada struktur program). Hal ini

terlihat pada gambar 2.9. Karena modul diintegrasikan dari bawah ke atas, maka pemrosesan yang diperlukan untuk modul subordinat ke suatu tingkat yang diberikan akan selalu tersedia dan kebutuhan akan stub dapat dieliminasi [PRE-01:490].



Gambar 2.9 Pengujian integrasi *bottom-up*

Sumber : [PRE-01:491]

Integrasi *bottom-up* dapat diimplementasi dengan langkah-langkah berikut [PRE-01:490] :

1. Komponen pada level terendah digabung ke dalam sebuah sub fungsi (cluster).
2. Driver akan dibuat untuk menguji setiap cluster.
3. Driver akan diganti dengan modul sesungguhnya setelah sub fungsi teruji.
4. Proses akan terus dilakukan sampai membentuk sebuah perangkat lunak yang utuh.

2.8.2.3 Pengujian Validasi

Pada pengujian terintegrasi, perangkat lunak secara lengkap dirakit sebagai suatu paket, kesalahan interfacing telah diungkap dan dikoreksi, dan seri akhir dari pengujian perangkat lunak, yaitu pengujian validasi dapat dimulai. Validasi dapat ditentukan dengan berbagai cara, tetapi definisi yang sederhana adalah

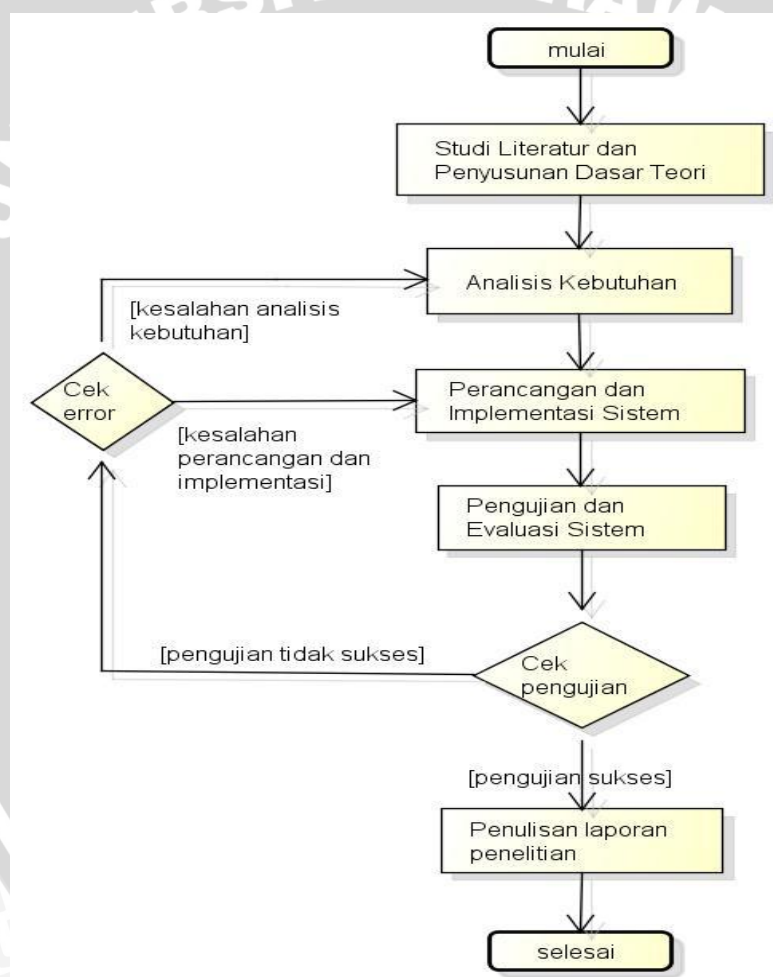
bahwa validasi berhasil bila perangkat lunak berfungsi dengan cara yang dapat diharapkan secara bertanggung jawab oleh pelanggan. Validasi perangkat lunak dicapai melalui sederetan pengujian *black-box* yang memperlihatkan konformitas dengan persyaratan. Rencana pengujian menguraikan kelas-kelas pengujian yang akan dilakukan, dan prosedur pengujian menentukan test case spesifik yang akan digunakan untuk mengungkap kesalahan dalam konformitas dengan persyaratan. Baik rencana dan prosedur didesain untuk memastikan apakah semua persyaratan fungsional dipenuhi, semua persyaratan kinerja dicapai, dokumentasi benar dan direkayasa oleh manusia, dan persyaratan lainnya dipenuhi (transportabilitas, kompatibilitas, pembetulan kesalahan, maintainabilitas) [PRE-01:495].



BAB III

METODE PENELITIAN

Bagian ini menjelaskan langkah-langkah yang akan ditempuh dalam penyusunan skripsi, yaitu studi literatur dan penyusunan dasar teori, analisis dan perancangan perangkat lunak, implementasi, pengujian dan evaluasi sistem, serta pengambilan kesimpulan dan saran. Untuk lebih memudahkan dalam pengerjaan aplikasi maka dibuat sebuah alur kerja yang dapat dipresentasikan oleh *flowchart* berikut ini :



Gambar 3.1. Diagram alur penelitian

3.1 Studi Literatur dan Penyusunan Dasar Teori

Studi literatur menjelaskan dasar teori yang digunakan untuk menunjang penulisan skripsi. Teori-teori pendukung tersebut meliputi:

1. ERP Perusahaan
2. ERP Pendidikan
3. *Reuse-Oriented Model*
4. *Unified Modelling Language*
 - a. *Use Case Diagram*
 - b. *Activity Diagram*
5. Pengujian dan Analisis Perangkat Lunak
 - a. Pengujian
 - b. Pengujian dan analisis unit
 - c. Pengujian dan analisis integrasi
 - d. Pengujian dan analisis validasi
 - e. Pengujian dan analisis *availability*
 - f. Pengujian dan analisis *compatibility*
 - g. Pengujian dan analisis *usability*
 - h. Pengujian dan analisis *response time*
 - i. Pengujian dan analisis *control*

3.2 Analisis dan Perancangan Perangkat Lunak

3.2.1 Analisis Kebutuhan

Analisis kebutuhan dilakukan dengan menentukan kebutuhan apa saja yang dibutuhkan untuk menganalisis dan melakukan implementasi SIMS Jibas.

Proses analisis ini meliputi :

1. Pemodelan *use case* untuk proses bisnis dan proses sistem.
2. Pemodelan aktivitas bisnis.
3. Pemodelan aktivitas sistem.

Metode analisis yang digunakan adalah *Object Oriented Analysis* dengan menggunakan bahasa pemodelan *UML (Unified Modelling Language)*. Fungsi-fungsi yang dapat dilakukan oleh aplikasi ini antara lain:

1. CRUD (*Create, Update, Delete*) pengguna, pelajaran, siswa dan pegawai serta semua entity yang diperlukan oleh sistem sesuai dengan kebutuhan di SMAN 4 Jember.
2. Melakukan pelaporan mengenai data-data siswa, data-data perpustakaan, data-data kepegawaian dan data-data akademik.

Kebutuhan yang digunakan dalam pembuatan aplikasi ini meliputi:

1. Kebutuhan *Hardware*, meliputi:
 - a. Komputer PC.
 - b. *Hub* atau *Switch*.
 - c. Kabel UTP.
2. Kebutuhan *Software*, meliputi:
 - a. Microsoft Windows 7 dan Linux sebagai sistem operasi.
 - b. Sublime sebagai *platform* pengembangan.
 - c. *XAMPP* dan *LAMPP* sebagai *tool* pembuatan basis data.
 - d. *AstahUML* sebagai *tool* pembuatan diagram pemodelan sistem.

3.2.2 Perancangan perangkat lunak

Setelah menentukan jenis kebutuhan yang diperlukan tahap selanjutnya adalah desain atau perancangan sistem. Perancangan terdiri dari beberapa tahap yang sesuai dengan model perangkat lunak *Reuse-Oriented*, antara lain :

1. Analisis komponen
Tahap ini bertujuan untuk menentukan data – data komponen yang berkaitan dalam pembuatan aplikasi.
2. Adopsi Komponen
Tahap ini bertujuan untuk menyesuaikan atau memodifikasi setiap komponen pada aplikasi agar sesuai dengan data - data kebutuhan yang didapatkan.

Pada perancangan skripsi ini SIMS Jibas digunakan sebagai dasar pengembangan aplikasi yang nantinya akan mengalami penambahan modul dan kustomisasi modul. Setiap proses bisnis yang ada dibuat dalam suatu *use case* umum untuk kemudian diperjelas dengan *activity diagram*.

3.2.3 Implementasi

Implementasi aplikasi dilakukan dengan mengacu pada perancangan perangkat lunak yang sudah didesain pada sub bab analisis kebutuhan dan perancangan perangkat lunak. Implementasi perangkat lunak dilakukan dengan menggunakan bahasa sublime untuk platform pengembangan dan *MySQL-Xampp* untuk basis data

3.2.4 Pengujian dan Evaluasi Sistem

Pengujian perangkat lunak dilakukan untuk mengetahui apakah kinerja sistem perangkat lunak Jibas telah memenuhi spesifikasi kebutuhan yang melandasinya. Strategi pengujian perangkat lunak yang akan digunakan yaitu pengujian unit (*unit testing*), pengujian integrasi (*integration testing*), pengujian validasi (*validation testing*), pengujian *compatibility*, dan pengujian *usability*. Metode pengujian yang akan digunakan adalah *white-box testing*, *black-box testing*, melakukan instalasi di sistem operasi linux dan memberikan kuisisioner kepada pengguna. Proses pengujian perangkat lunak dimulai dari pengujian unit, kemudian dilanjutkan dengan pengujian integrasi, dan berakhir pada pengujian validasi. Pada tahap pengujian unit dan pengujian integrasi digunakan metode *white-box testing* dengan teknik *basis path*. Kemudian pada tahap pengujian validasi digunakan metode *black-box testing*. Pengujian *compatibility* dilakukan untuk mengetahui apakah SIMS Jibas dapat berjalan secara normal sesuai dengan kebutuhan di sistem operasi windows dan linux, pengujian *usability* dilakukan untuk mengetahui tingkat kemudahan penggunaan sistem dengan pemberian kuisisioner kepada pengguna atau staff yang ada di SMAN 4 Jember. Analisis juga dilakukan untuk mengetahui hasil dari pengujian perangkat lunak sehingga dapat didapatkan kesimpulan dari pengembangan perangkat lunak yang telah dilakukan.

3.3 Pengambilan Kesimpulan dan Saran

Pengambilan kesimpulan dilakukan setelah semua tahapan perancangan perangkat lunak, implementasi perangkat lunak, dan pengujian perangkat lunak telah selesai dilakukan. Kesimpulan diambil dari hasil pengujian dan analisis terhadap sistem yang dibangun. Tahap terakhir dari penulisan adalah saran yang

dimaksudkan untuk memperbaiki kesalahan – kesalahan yang terjadi dan menyempurnakan penulisan serta untuk memberikan pertimbangan atas pengembangan perangkat lunak lebih lanjut.



BAB IV

PERANCANGAN PERANGKAT LUNAK

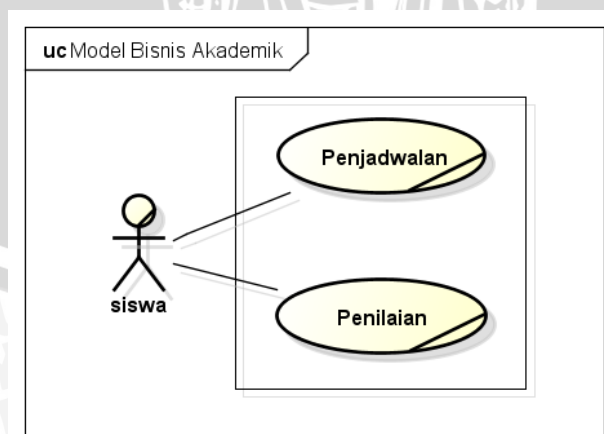
Perancangan perangkat lunak digunakan sebagai tahap awal pengimplementasian desain sistem dan perancangan antarmuka yang digunakan. Perancangan perangkat lunak dibangun berdasarkan hasil studi literatur dan analisis kebutuhan sistem. Tahap perancangan sistem meliputi pemodelan proses bisnis (as-is), pemodelan sistem (to-be), rekapitulasi kebutuhan sistem dan perancangan, perbandingan SIMS Jibas dengan sistem yang dibuat dan arsitektur aplikasi SIMS Jibas.

4.1 Analisis Kebutuhan

Pada tahap ini proses analisis kebutuhan dan perancangan sistem dibagi menjadi 2 yakni, pemodelan untuk proses bisnis (as-is) dan pemodelan proses sistem (to-be).

4.1.1 Pemodelan Proses Bisnis (as –is)

Pada bagian ini akan dibahas mengenai proses – proses bisnis yang sekarang terjadi di SMAN 4 Jember, sebelum menggunakan aplikasi SIMS Jibas. Satu proses bisnis di SMAN 4 Jember digambarkan dengan satu *use case*, dan satu *use case* digambarkan dengan sekumpulan aktivitas. Proses bisnis akademik SMAN 4 Jember dapat dijelaskan oleh gambar 4.1 berikut :

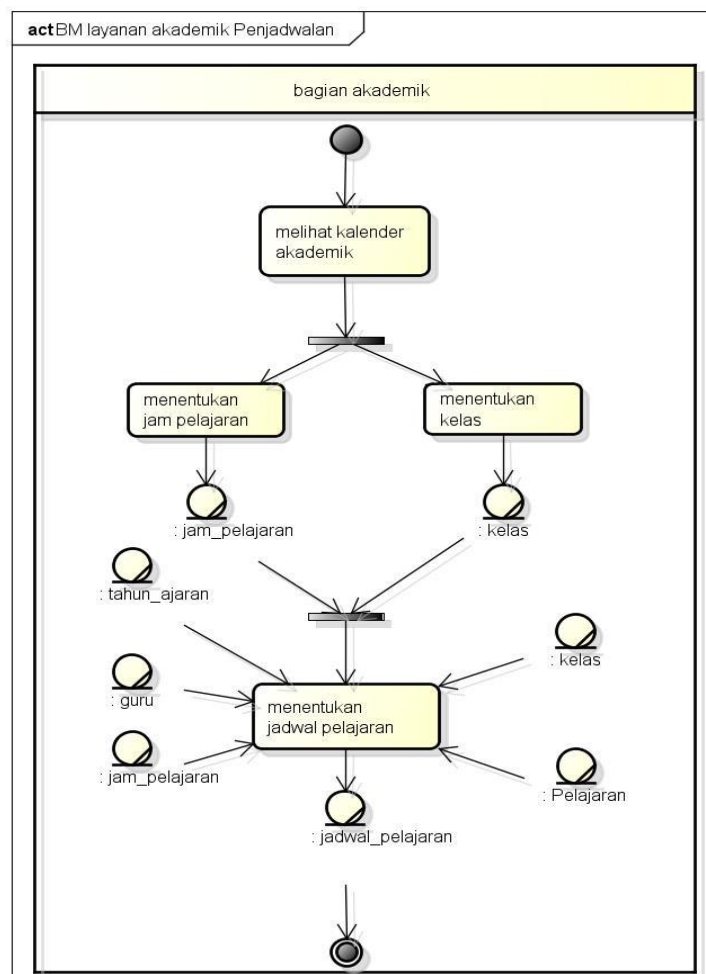


Gambar 4.1 Use case model bisnis akademik (as –is)

Setiap *use case* bisnis dan aktor bisnis pada gambar 4.1 dapat dijelaskan sebagai berikut:

1. Aktor bisnis yang berperan dalam model bisnis di bagian akademik adalah siswa.
2. *Use case* bisnis penjadwalan adalah *use case* bisnis yang digunakan untuk proses manajemen penjadwalan.
3. *Use case* bisnis penilaian adalah *use case* bisnis yang digunakan untuk proses penilaian siswa.

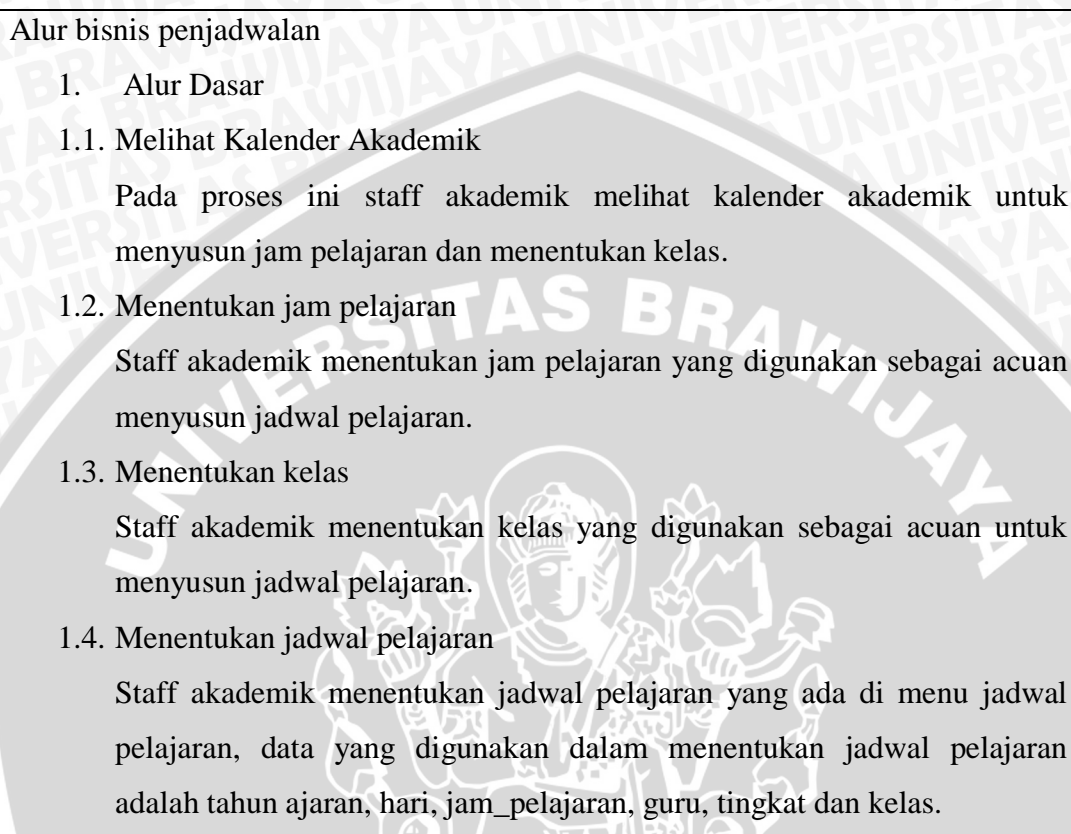
Setiap *use case* bisnis memiliki beberapa aktivitas yang menjelaskan proses apa yang mungkin dilakukan oleh aktor maupun oleh sistem. Gambar 4.2 adalah *activity diagram* yang menjelaskan detail *use case* bisnis penjadwalan



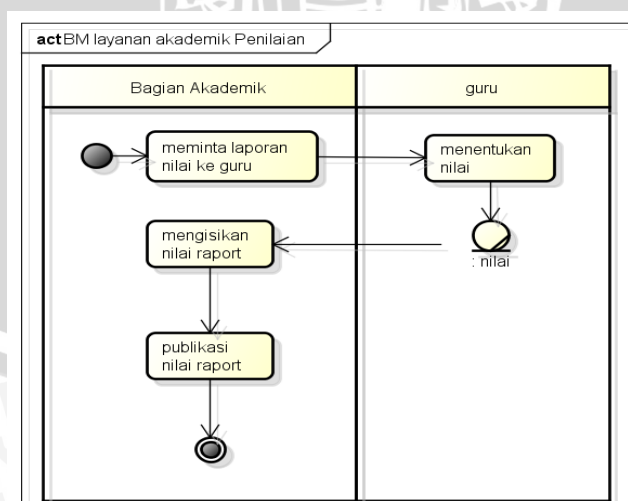
Gambar 4.2 Model bisnis layanan akademik penjadwalan

Model bisnis pejadwalan pada gambar 4.2 dapat diperjelas dengan runtutan kegiatan seperti pada tabel 4.1 berikut :

Tabel 4.1 Tabel skenario model bisnis layanan akademik penjadwalan



Gambar 4.3 adalah *activity diagram* yang menjelaskan detail *use case* bisnis penilaian:



powered by Astah

Gambar 4.3 Model bisnis layanan akademik penilaian

Model bisnis penilaian pada gambar 4.3 dapat diperjelas dengan runtutan kegiatan seperti pada tabel 4.2 berikut :

Tabel 4.2 Tabel skenario model bisnis layanan akademik penilaian

Alur Bisnis Layani Penilaian

1. Alur Dasar

1.1. Meminta Laporan Penilaian ke Guru

Pada proses ini staff akademik meminta laporan penilaian kepada guru.

1.2. Menentukan Nilai

Guru menentukan nilai pelajaran dan nilai raport untuk diinputkan oleh staff akademik, data yang dioperasikan adalah data nilai.

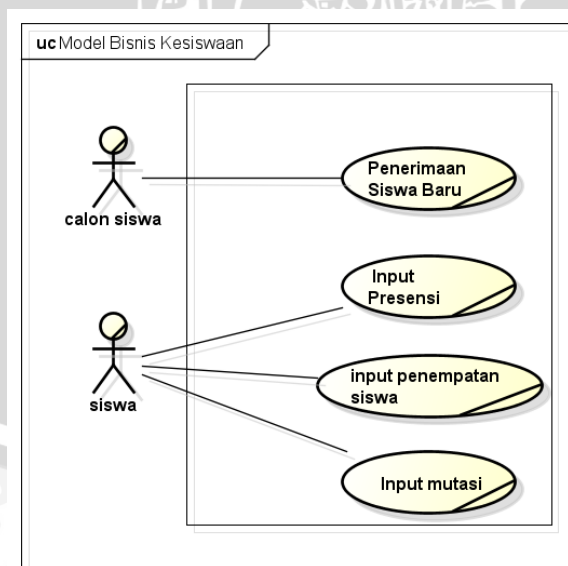
1.3. Mengisi Nilai Raport

Staff akadmik mengisi nilai raport yang diberikan oleh guru, nilai raport diperoleh dari nilai pelajaran sehari – hari.

1.4. Publikasi Nilai Raport

Staff akademik melakukan publikasi nilai raport dengan mencetak nilai raport, keseluruhan nilai raport tiap kelas dijadikan satu menjadi legger.

Proses bisnis kesiswaan yang sekarang terjadi di SMAN 4 Jember dapat dijelaskan pada gambar 4.4 berikut:



powered by Astah

Gambar 4.4 Use case model bisnis kesiswaan (as-is)

Adapun setiap *use case* bisnis dan aktor bisnis gambar 4.4 dapat dijelaskan sebagai berikut:

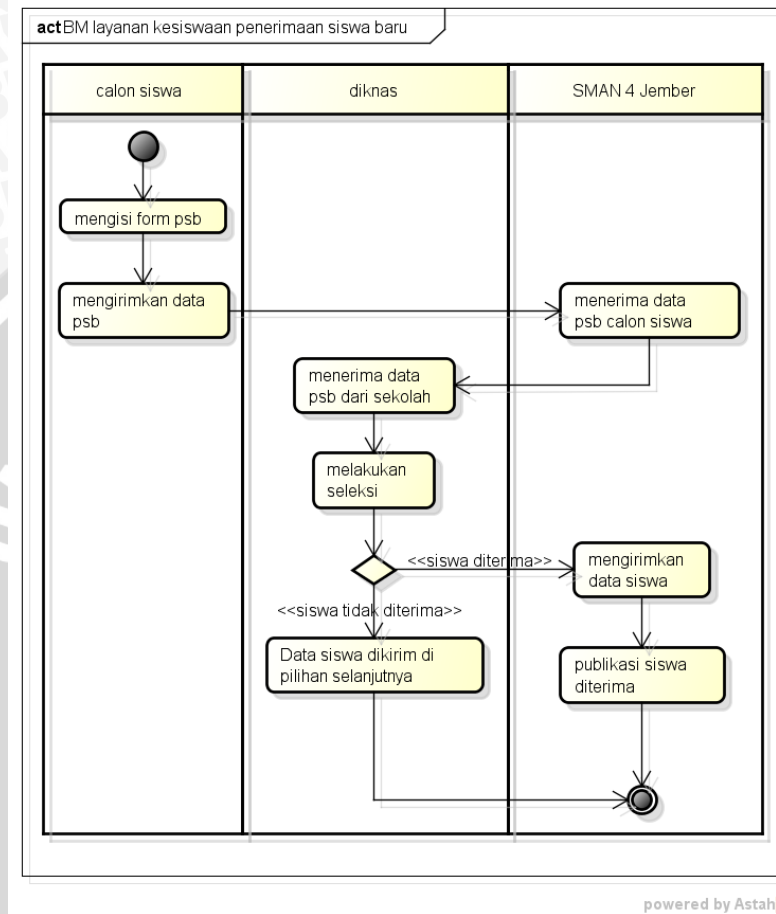
1. Aktor bisnis yang berperan dalam model bisnis bagian kesiswaan adalah calonsiswa dan siswa.
2. *Use case* bisnis penerimaan siswa baru adalah aktivitas yang digunakan untuk melakukan proses penerimaan siswa baru, *use case* bisnis ini memberikan nilai tambah untuk calon siswa yang ada di sman 4 jember.
3. *Use case* bisnis input presensi adalah aktivitas yang digunakan untuk melakukan presensi siswa sehari – hari, *use case* bisnis ini memberikan nilai tambah bagi siswa.
4. *Use case* bisnis input penempatan siswa adalah aktivitas yang digunakan untuk menentukan penempatan siswa naik ke tingkat atau departemen yang lebih tinggi, *use case* bisnis ini memberikan nilai tambah bagi siswa.
5. *Use case bisnis* input mutasi adalah aktivitas yang digunakan untuk menentukan siswa yang mengalami mutasi.

Dari pemodelan bisnis pada gambar 4.4 dapat diidentifikasi aktor bisnis yang berperan dalam bisnis di SMAN 4 Jember sebagai berikut:

Tabel 4.3 Tabel identifikasi aktor bisnis

Aktor	Deskripsi Aktor Bisnis
Siswa	Peserta didik yang tercatat secara administrasi di SMAN 4
Calon Siswa	Seseorang yang mempunyai minat untuk menjadi peserta didik di SMAN 4 Jember.
Pegawai	Seseorang yang menggunakan tenaga dan kemampuannya untuk SMAN 4 Jember dan tercatat secara administrasi di SMAN 4 Jember
Calon Pegawai	Seseorang yang mempunyai minat untuk menjadi pegawai di SMAN 4 Jember

Setiap *use case* bisnis pada gambar 4.4 memiliki beberapa aktivitas yang menjelaskan proses apa yang mungkin dilakukan oleh aktor maupun oleh sistem. Gambar 4.5 merupakan *activity diagram* input penerimaan siswa baru:



Gambar 4.5 Model bisnis layanan kesiswaan penerimaan siswa baru

Model bisnis layanan kesiswaan penerimaan siswa baru pada gambar 4.5 dapat diperjelas dengan runtutan kegiatan seperti pada tabel 4.4 berikut :

Tabel 4.4 Tabel skenario bisnis layanan kesiswaan penerimaan siswa baru

<p>Alur Bisnis layanan kesiswaan penerimaan siswa baru</p> <ol style="list-style-type: none"> 1. Alur Dasar <ol style="list-style-type: none"> 1.1. Mengisi Form PSB <p>Calon siswa mengisi form psb proses secara lengkap.</p> 1.2. Mengirimkan Data PSB <p>Calon siswa mengirimkan data pribadi siswa sesuai form psb ke SMAN 4 Jember.</p>

1.3.Menerima Data PSB calon siswa

Staff kesiswaan menerima data pribadi calon siswa untuk dikirimkan ke diknas pusat kabupaten.

1.4.Mengirimkan Data PSB calon siswa ke diknas

Staff kesiswaan mengirimkan data pribadi siswa ke diknas pusat kabupaten.

1.5.Menerima Data PSB calon siswa dari sman 4 Jember

Bagian diknas menerima data psb calon siswa dari SMAN 4 Jember kemudian melakukan proses seleksi.

1.6.Publikasi Siswa Diterima

Staff kesiswaan melakukan publikasi siswa yang diterima di departemen SMAN 4 Jember.

2. *Alternative Flow*

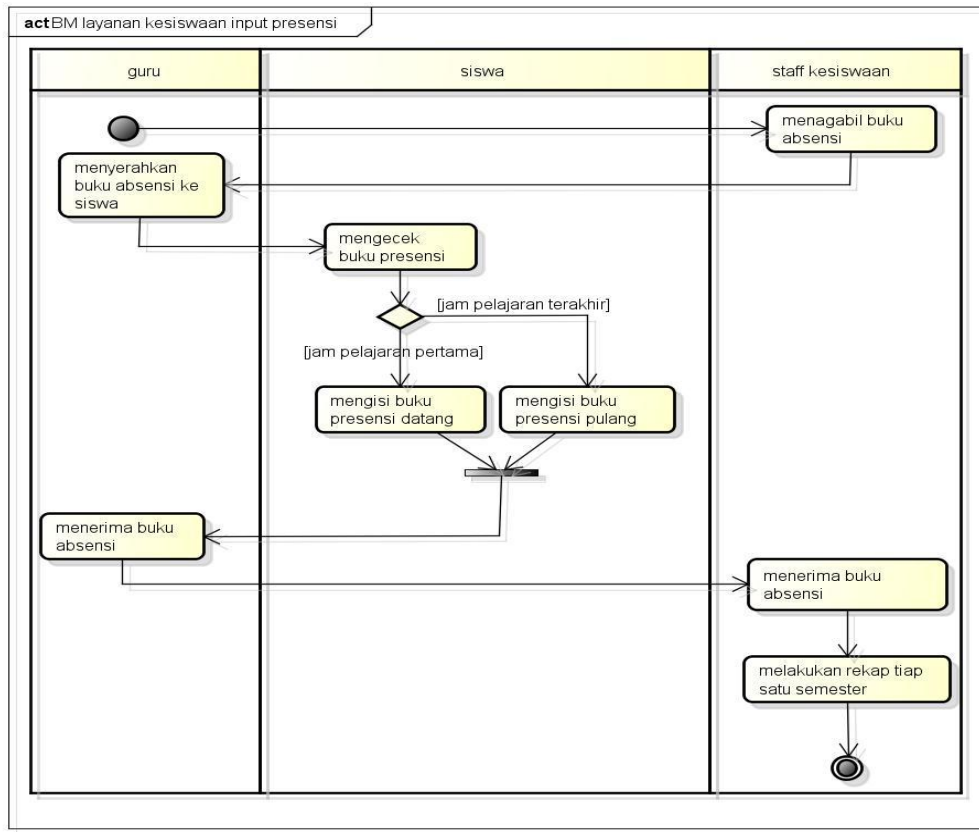
2.1.Siswa Diterima

Data hasil seleksi siswa yang diterima dikirimkan oleh diknas ke SMAN 4 Jember.

2.2. Siswa Tidak Diterima

Data siswa yang tidak diterima di SMAN 4 jember dikirim ke pilihan SMAN selanjutnya sesuai pilihan siswa.

Gambar 4.6 merupakan *activity diagram* yang menjelaskan detail *use case* bisnis input presensi siswa:



Gambar 4.6 Model bisnis layanan kesiswaan input presensi

Model bisnis layanan kesiswaan input penerimaan siswa baru pada gambar 4.6 dapat diperjelas dengan runtutan kegiatan seperti pada tabel 4.5 berikut :

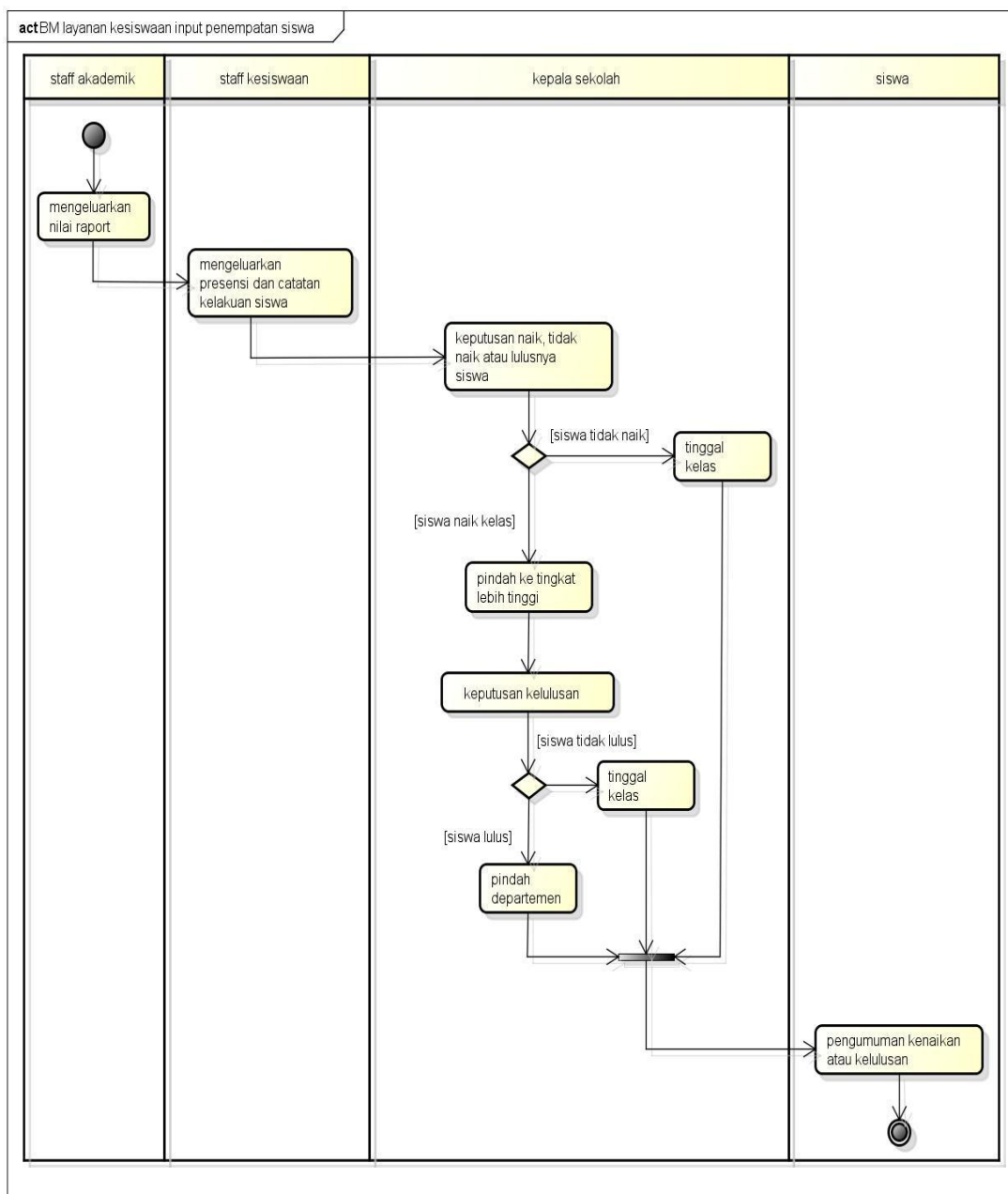
Tabel 4.5 Tabel skenario bisnis layanan kesiswaan input presensi

Alur Bisnis Layanan Kesiswaan Input Presensi	
1.	Alur Dasar
1.1.	Mengambil Buku Absensi
	Guru Pelajaran Mengambil Absensi ke bagian kesiswaan
1.2.	Melakukan Presensi
	Siswa melakukan presensi 2 kali dalam sehari yakni datang dan pulang
1.3.	Menerima Buku Presensi
	Guru menerima buku presensi dari siswa kemudian di berikan ke bagian staff kesiswaan

1.4.Melakukan rekapitulasi

Staff kesiswaan melakukan rekapitulasi presensi setiap satu semester untuk dituliskan ke nilai hasil belajar siswa.

Gambar 4.7 merupakan *activity diagram* yang menjelaskan detail *use case* bisnis input penempatan siswa:



Gambar 4.7 Model bisnis layanan kesiswaan input penempatan siswa

Model bisnis layanan kesiswaan input penempatan siswa pada gambar 4.7 dapat diperjelas dengan runtutan kegiatan seperti pada tabel 4.6 berikut :

Tabel 4.6 Tabel skenario bisnis layanan kesiswaan input penempatan siswa

Alur bisnis layanan kesiswaan input penempatan siswa

1. Alur Dasar

1.1. Mengeluarkan Nilai Raport

Staff akademik mengeluarkan nilai raport akhir siswa.

1.2. Mengeluarkan Presensi dan Catatan Kelakuan Siswa

Staff kesiswaan mengeluarkan presensi dan catatan kelakuan siswa.

1.3. Rapat keputusan kenaikan dan kelulusan siswa

Rapat keputusan kenaikan dan kelulusan siswa dipimpin oleh kepala sekolah dengan dua alternative keputusan yakni naik atau tidak naiknya dan lulus atau tidak lulus.

1.4. Pengumuman Kenaikan dan Kelulusan siswa

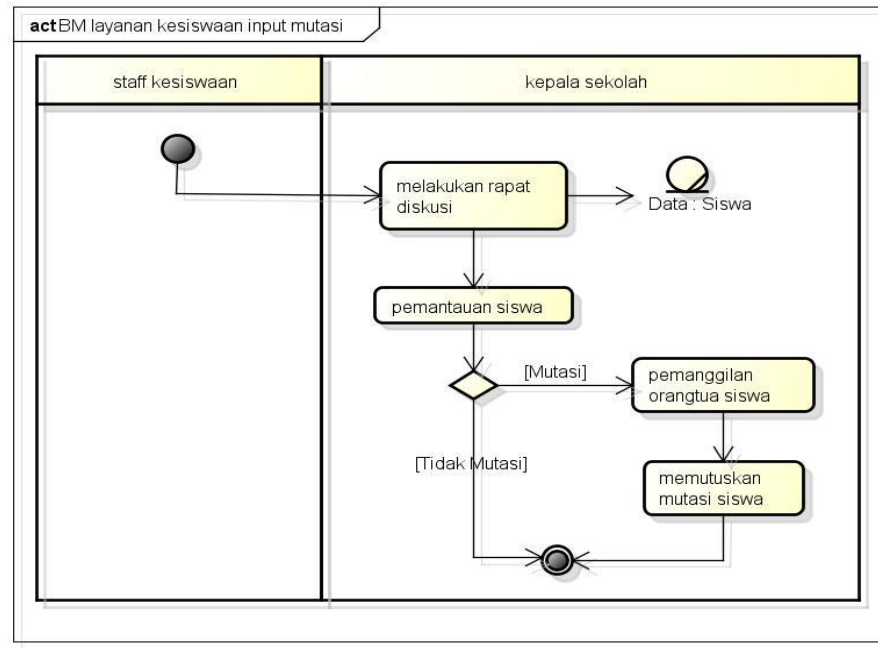
Siswa menerima publikasi kenaikan dan kelulusan siswa kemudian melakukan penempatan siswa ke tingkat ataupun departemen yang lebih tinggi.

2. *Alternative Flow*

2.1. Siswa Tidak Naik Kelas. Siswa akan tetap tinggal kelas

2.2. Siswa Naik Kelas. Siswa akan ke departemen atau tingkat yang lebih tinggi

Gambar 4.8 merupakan *activity diagram* yang menjelaskan detail *use case* bisnis input mutasi siswa:



Gambar 4.8 Model Bisnis Layanan Kesiswaan Input Mutasi Siswa

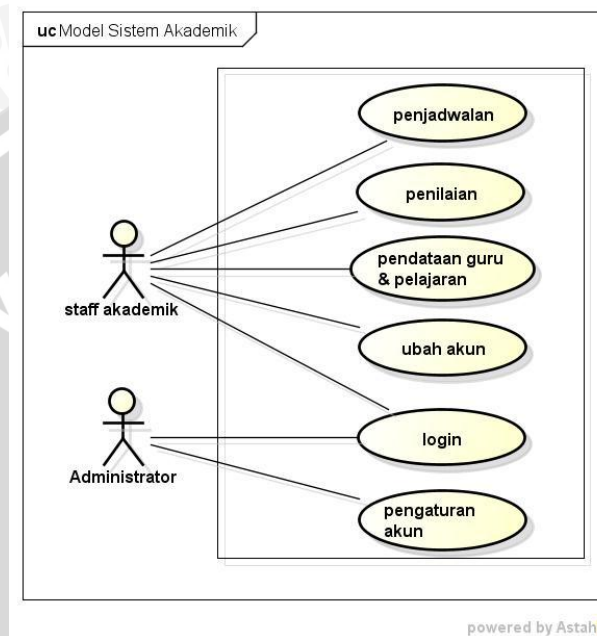
Model bisnis layanan kesiswaan input mutasi siswa pada gambar 4.8 dapat diperjelas dengan runtutan kegiatan seperti pada tabel 4.7 berikut :

Tabel 4.7 Tabel Skenario Bisnis Layanan Kesiswaan Input mutasi siswa

<p>Alur bisnis layanan kesiswaan input mutasi siswa</p> <ol style="list-style-type: none"> 1. Alur Dasar <ol style="list-style-type: none"> 1.1.Melakukan Rapat Diskusi Kepala Sekolah beserta guru melakukan rapat diskusi sebelum melakukan mutasi siswa 1.2.Pemantauan Siswa Melakukan Pemantauan siswa untuk siswa yang bermasalah 2. <i>Alternative Flow</i> <ol style="list-style-type: none"> 2.1.Siswa Mutasi Melakukan pemanggilan terhadap orang tua siswa yang mutasi 2.2.Tidak Memutasi Siswa Siswa kembali normal tanpa dilakukan pemantauan secara intensif

4.1.2 Pemodelan Sistem (to-be)

Dari penjelasan proses bisnis pada sub bab 4.1.1 maka digambarkan pemodelan sistem untuk SIMS Jibas agar dapat mengurangi aktivitas manual yang terjadi di SMAN 4 Jember. Gambar 4.9 merupakan pemodelan sistem untuk bagian akademik



Gambar 4.9 Use case layanan akademik (to-be)

Adapun setiap *use case* sistem model pada gambar 4.9 dapat dijelaskan sebagai berikut:

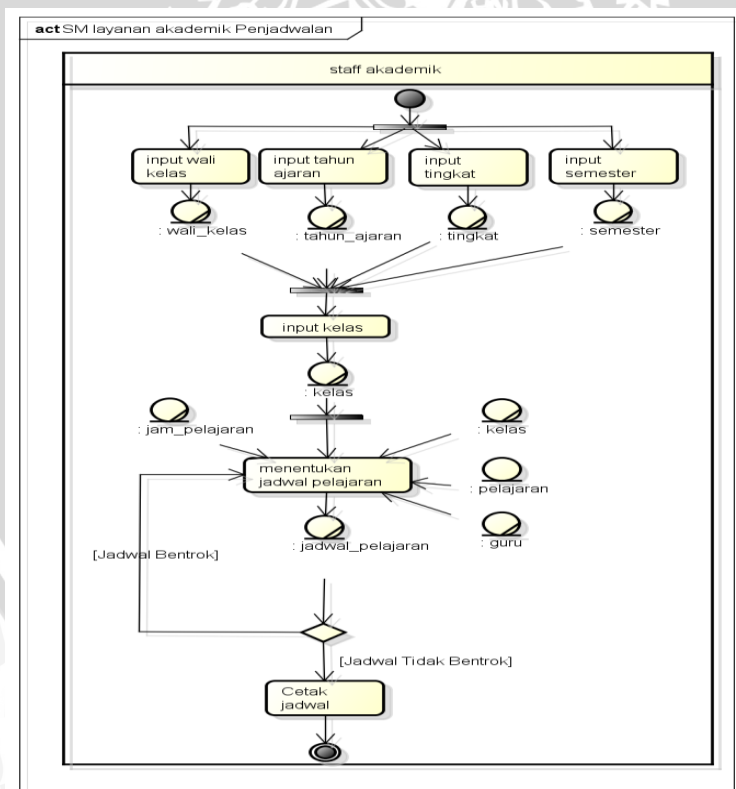
1. *Use case* penjadwalan dapat menangani proses penyusunan penjadwalan.
2. *Use case* penilaian dapat melayani penilaian satu tahun akademik siswa.
3. *Use case* pendataan guru dan pelajaran digunakan untuk melakukan aktivitas pendataan guru dan pelajaran.
4. *Use case* ubah pengguna merupakan aktivitas yang digunakan untuk merubah pengguna staff akademik yakni *username* dan password.
5. *Use case* pengaturan pengguna digunakan untuk menambah, menghapus, dan mengubah pengguna SMAN 4 Jember, use case ini dilakukan oleh administrator.

Dari pemodelan sistem gambar 4.9 dapat diidentifikasi aktor yang berperan dalam sistem SIMS Jibas sebagai berikut:

Tabel 4.8 Tabel identifikasi aktor

Aktor	Deskripsi Aktor
Administrator	Administrator merupakan aktor yang mengoperasikan penambahan pengguna sistem dan dapat mengakses seluruh modul.
Staff akademik	Staff akademik merupakan aktor pengguna sistem yang berperan dalam proses yang ada di bagian akademik.
Staff kesiswaan	Staff kesiswaan merupakan aktor pengguna sistem yang berperan dalam proses yang ada di bagian kesiswaan.
Staff kepegawaian	Staff kepegawaian merupakan aktor pengguna sistem yang berperan dalam proses yang ada di bagian kepegawaian.
Staff perpustakaan	Staff perpustakaan merupakan aktor pengguna sistem yang berperan dalam proses yang ada di bagian perpustakaan.

Setiap *use case* sistem memiliki beberapa aktivitas yang menjelaskan proses apa yang mungkin dilakukan oleh aktor maupun oleh sistem. Gambar 4.10 adalah aktivitas yang menjelaskan detail *use case* sistem penjadwalan:



Gambar 4.10 Model sistem layanan akademik penjadwalan

Secara garis besar proses bisnis yang dikerjakan oleh sistem layanan akademik penjadwalan sama dengan model bisnis layanan akademik penjadwalan namun, ada beberapa kegiatan yang mungkin hilang atau menjadi otomatis. Hal ini dapat dijelaskan sebagai berikut :

Tabel 4.9 Tabel skenario sistem layanan akademik penjadwalan

Alur sistem layanan akademik penjadwalan

1. Alur Dasar

1.1. *Login*

Staff akademik dan administrator diharuskan untuk *login* pada sistem. Pada proses ini diasumsikan *login* berhasil.

1.2. Input tahun ajaran

Pada proses ini staff akademik atau administrator menginputkan tahun akademik.

1.3. Input semester

Pada proses ini staff akademik menginputkan semester.

1.4. Input Kelas

Pada proses ini staff akademik atau administrator menginputkan kelas bersamaan dengan menginputkan wali kelas.

1.5. Menentukan Jam Pelajaran

Pada proses ini staff akademik atau administrator menentukan jam pelajaran yang digunakan untuk referensi penyusunan jadwal.

1.6. Menentukan Jadwal Pelajaran

Pada proses ini staff akademik atau administrator menentukan jadwal pelajaran dengan entity yang diperlukan: kelas, jam pelajaran, pelajaran, guru.

2. *Alternative Flow*

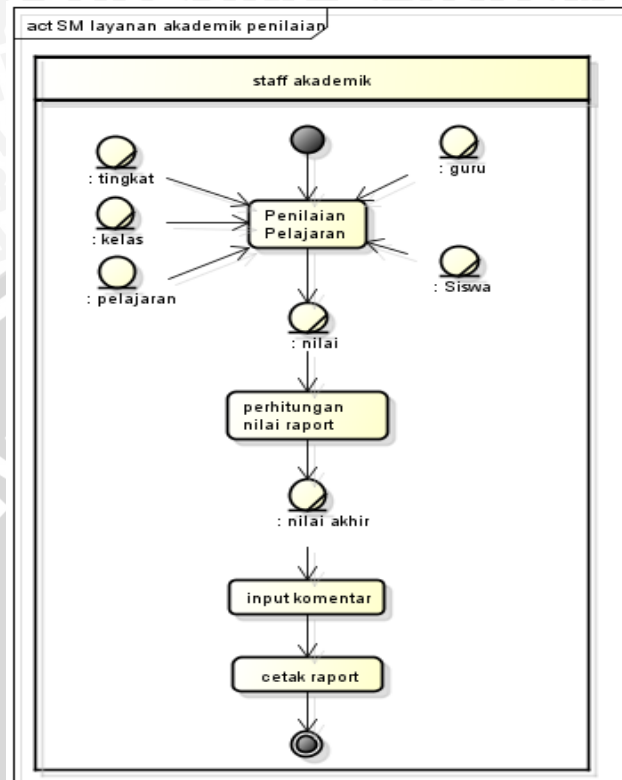
2.1. Jadwal Bentrok

Apabila jadwal bentrok maka sistem akan memunculkan peringatan, oleh karena itu staff akademik atau administrator harus mengganti jadwal.

2.2. Jadwal Tidak Bentrok

Jadwal satu tahun akademik siap untuk di cetak.

Gambar 4.11 merupakan *activity diagram* yang menjelaskan detail *use case* penilaian:



Gambar 4.11 Model sistem layanan akademik penilaian

Secara garis besar proses bisnis yang dikerjakan sistem layanan akademik penilaian sama dengan model bisnis layanan akademik penilaian namun, ada beberapa kegiatan yang mungkin hilang atau menjadi otomatis. Hal ini dapat dijelaskan sebagai berikut :

Tabel 4.10 Tabel skenario sistem layanan akademik penilaian

Alur Sistem Layanan Akademik Penilaian	
1.	Alur Dasar
1.1.	<i>Login</i> Staff akademik dan administrator diharuskan untuk <i>login</i> pada sistem. Pada proses ini diasumsikan <i>login</i> berhasil.
1.2.	Penilaian Pelajaran Pada proses ini staff akademik atau administrator dapat menginputkan nilai siswa berdasarkan pelajaran.

1.3.Perhitungan Nilai Raport

Pada proses ini staff akademik atau administrator akan menginputkan nilai raport yang diambil dari perhitungan nilai pelajaran.

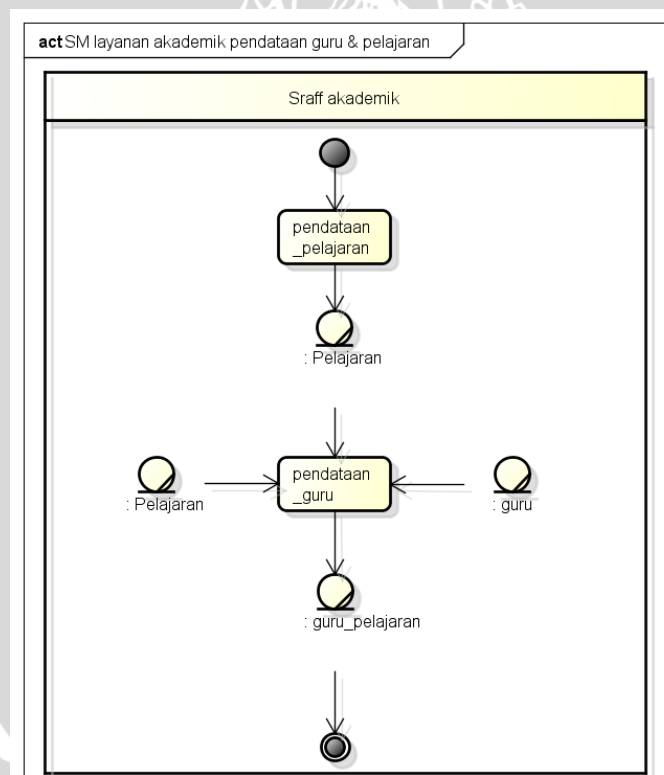
1.4.Input Komentar

Pada proses ini staff akademik atau administrator akan menginputkan komentar untuk masing – masing pelajaran.

1.5.Cetak raport

Pada proses ini staff akademik atau administrator akan mencetak nilai raport untuk dipublikasikan di siswa.

Gambar 4.12 merupakan *activity diagram* yang menjelaskan detail *use case* pendataan guru dan pelajaran:



powered by Astah

Gambar 4.12 Model sistem layanan akademik pendataan pelajaran dan guru

Secara garis besar sistem bisnis yang dikerjakan dalam layanan pendataan pelajaran dan guru akan dijelaskan sebagai berikut :

Tabel 4.11 Tabel skenario sistem layanan akademik pendataan guru dan pelajaran
Alur Sistem Layanan Akademik Pendataan guru dan pelajaran

1. Alur Dasar

1.1.Login

Staff akademik dan administrator diharuskan untuk *login* pada sistem.
Pada proses ini diasumsikan *login* berhasil.

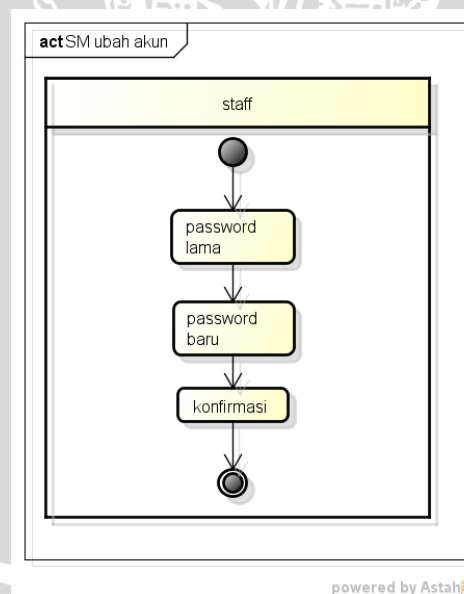
1.2.Pendataan Pelajaran

Pada proses ini staff akademik atau administrator dapat menginputkan data pelajaran untuk kemudian digunakan dalam penentuan guru pelajaran.
Pendataan pelajaran menghasilkan entity pelajaran.

1.3.Pendataan Guru

Pada proses ini staff akademik atau administrator akan menginputkan data guru diambil dari pegawai yang bertipe akademik, kemudian mengambil data pelajaran untuk dijadikan entity guru pelajaran.

Gambar 4.13 merupakan *activity diagram* yang menjelaskan detail *use case* ubah pengguna:



Gambar 4.13 Model sistem ubah pengguna

Secara garis besar sistem bisnis yang dikerjakan dalam ubah pengguna akan dijelaskan sebagai berikut :

Tabel 4.12 Tabel skenario sistem ubah pengguna

Alur Sistem Pengaturan Pengguna

1. Alur Dasar

1.1. Login

Administrator diharuskan *login* pada sistem. Diasumsikan *login* berhasil.

1.2. Password Lama

Pada proses ini staff akademik memasukkan password lama.

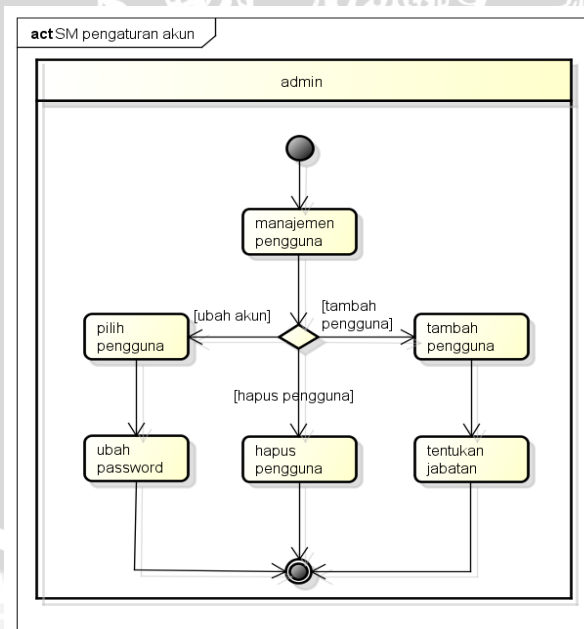
1.3. Password Baru

Pada proses ini staff akademik memasukkan password baru.

1.4. Konfirmasi

Pada proses ini staff akademik diharuskan melakukan konfirmasi password baru.

Untuk menangani penggunaan aplikasi maka dibuatlah suatu sistem yang hanya dapat digunakan oleh administrator dalam mengatur pengguna aplikasi. Hal ini dapat dijelaskan oleh *activity diagram* sebagai berikut :



powered by Astah

Gambar 4.14 Model sistem pengaturan pengguna

Secara garis besar sistem bisnis yang dikerjakan dalam ubah pengguna akan dijelaskan sebagai berikut :

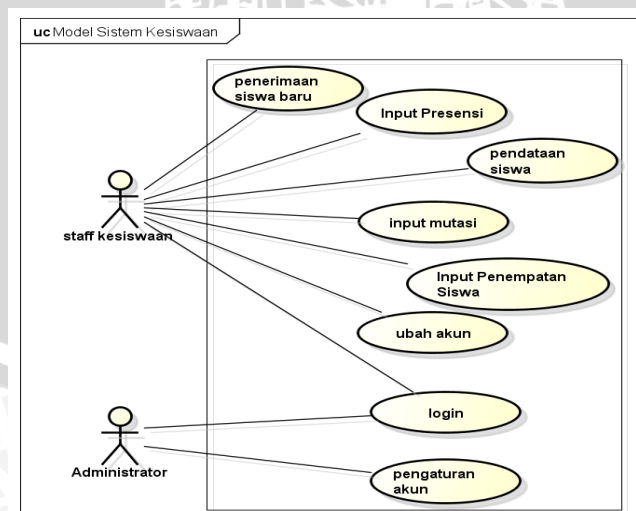
Tabel 4.13 Tabel skenario sistem pengaturan pengguna

Alur sistem pengaturan pengguna

1. Alur Dasar
 - 1.1. Login

Administrator diharuskan *login* pada sistem. Diasumsikan *login* berhasil.
 - 1.2. Manajemen Pengguna

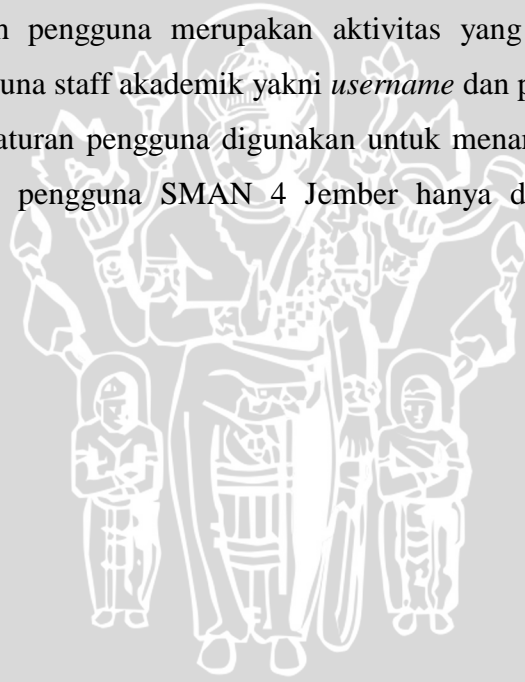
Pada proses ini administrator dapat melakukan pengaturan semua pengguna yang menggunakan sistem. Hal ini memiliki beberapa kemungkinan yakni tambah pengguna, hapus pengguna dan mengubah password pengguna.
2. Alternative Flow
 - 2.1. Menambah pengguna. Administrator menambah pengguna sistem karena ada pegawai baru.
 - 2.2. Menghapus pengguna. Administrator menghapus pengguna sistem karena ada pegawai yang keluar.
 - 2.3. Mengubah password pengguna. Administrator mengubah password pengguna sistem.



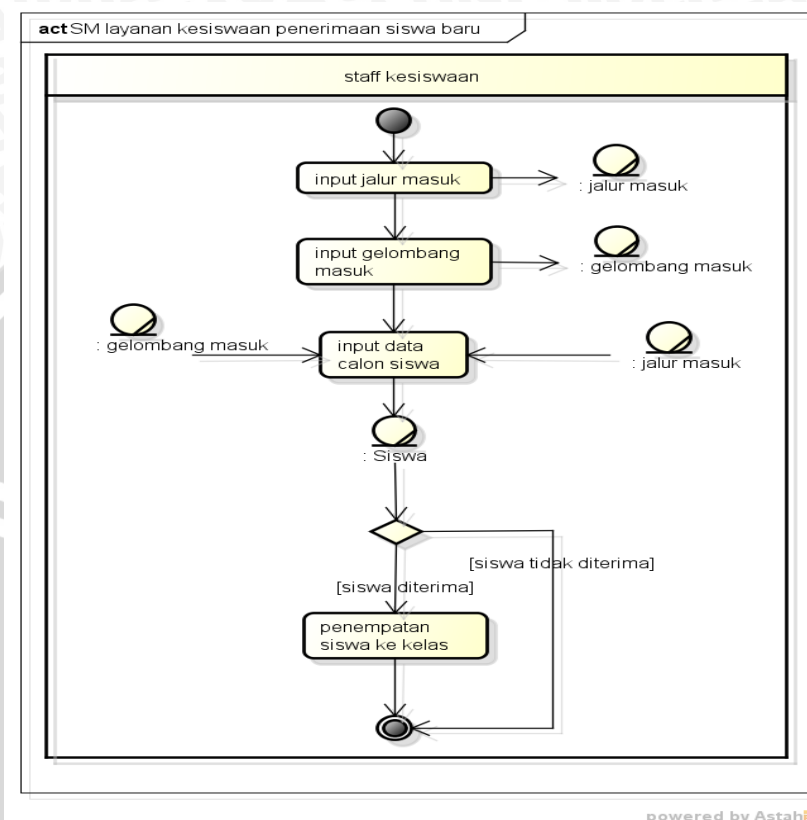
Gambar 4.15 Use case layanan kesiswaan (to - be)

Adapun setiap *use case* sistem model tersebut dapat dijelaskan sebagai berikut:

1. *Use case* penerimaan siswa baru dapat menangani proses penerimaan siswa baru.
2. *Use case* input presensi merupakan *use case* yang digunakan untuk melakukan presensi ke sistem.
3. *Use case* pendataan siswa digunakan untuk melakukan aktivitas pendataan siswa.
4. *Use case* input mutasi digunakan untuk melakukan penginputan mutasi siswa.
5. *Use case* input penempatan siswa merupakan *use case* yang digunakan untuk penempatan siswa ke tingkat maupun jenjang yang lebih tinggi.
6. *Use case* ubah pengguna merupakan aktivitas yang digunakan untuk merubah pengguna staff akademik yakni *username* dan password.
7. *Use case* pengaturan pengguna digunakan untuk menambah, menghapus, dan mengubah pengguna SMAN 4 Jember hanya dapat diakses oleh administrator.



Dari *Use case* sistem gambar 4.15 dapat digambarkan *activity diagram* penempatan siswa baru. *activity diagram* penempatan siswa baru adalah sebagai berikut :



Gambar 4.16 Model sistem penerimaan siswa baru

Secara garis besar proses bisnis yang dikerjakan oleh sistem layanan kesiswaan penerimaan siswa baru sama dengan model bisnis layanan kesiswaan penerimaan siswa baru, namun ada beberapa kegiatan yang mungkin hilang atau menjadi otomatis. Hal ini dapat dijelaskan sebagai berikut :

Tabel 4.14 Tabel skenario sistem penerimaan siswa baru

<p>Alur sistem pengaturan penerimaan siswa baru</p> <ol style="list-style-type: none"> 1. Alur Dasar <ol style="list-style-type: none"> 1.1. Login <p>Administrator dan staff kesiswaan diharuskan <i>login</i> pada sistem. Diasumsikan <i>login</i> berhasil.</p>

1.2. Input jalur masuk

Administrator atau staff kesiswaan melakukan penginputan jalur masuk, entity jalur masuk digunakan untuk menentukan gelombang.

1.3. Input gelombang

Administrator atau staff kesiswaan melakukan penginputan gelombang.

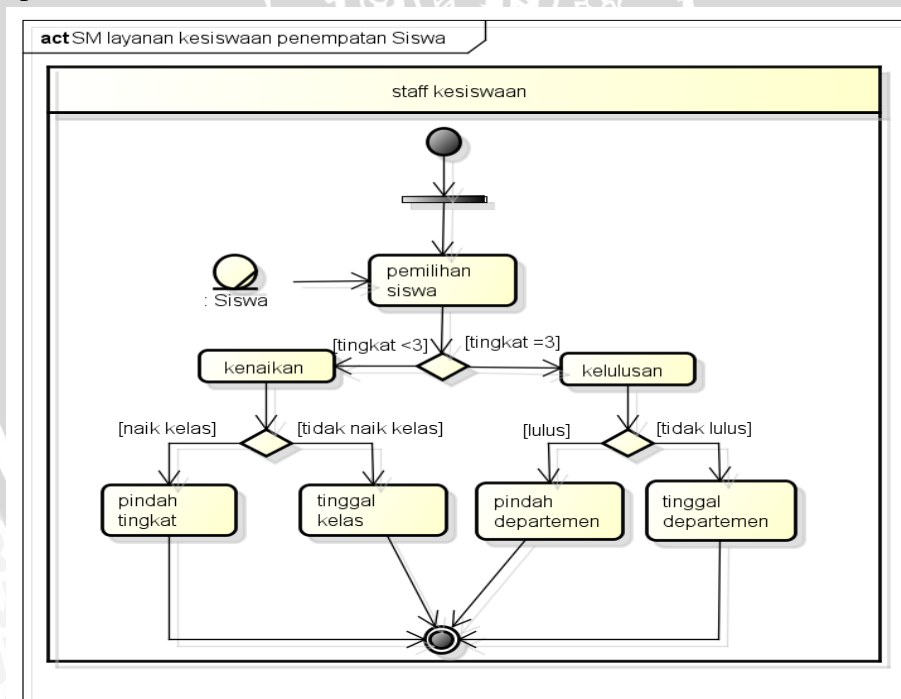
1.4. Input data calon siswa

Administrator atau staff kesiswaan melakukan penginputan data calon siswa.

1.5. Input penempatan siswa

Siswa yang telah lolos seleksi ditempatkan di kelas yang sudah diatur kapasitas siswanya.

Gambar 4.17 merupakan *activity diagram* yang menjelaskan detail *use case* penempatan siswa:



Gambar 4.17 Model sistem penempatan siswa

Secara garis besar proses bisnis yang dikerjakan dalam model sistem layanan kesiswaan penempatan siswa sama dengan model bisnis layanan

kesiswaan penempatan siswa namun, ada beberapa kegiatan yang mungkin hilang atau menjadi otomatis. Hal ini dapat dijelaskan pada tabel 4.15 berikut:

Tabel 4.15 Tabel skenario sistem penempatan siswa

Alur sistem pengaturan penerimaan siswa baru

1. Alur Dasar

1.1. *Login*

Administrator dan staff kesiswaan diharuskan *login* pada sistem. Diasumsikan *login* berhasil.

1.2. *Pemilihan Siswa*

Administrator atau staff kesiswaan memilih entity siswa yang akan diolah.

2. *Alternative Flow*

2.1. *Kenaikan*

Staff kesiswaan menentukan siswa, apabila dia berada di tingkat < 3 maka aktivitas kenaikan yang akan dipilih oleh staff kesiswaan.

2.2. *Kelulusan*

Staff kesiswaan menentukan siswa, apabila dia berada di tingkat $= 3$ maka aktivitas kelulusan yang akan dipilih oleh staff kesiswaan.

2.3. *Pindah tingkat*

Staff kesiswaan menentukan siswa, apabila dia naik kelas maka staff kesiswaan melakukan pindah tingkat ke siswa tersebut.

2.4. *Tinggal kelas*

Staff kesiswaan menentukan siswa, apabila dia tidak naik kelas maka staff kesiswaan melakukan tidak melakukan pindah tingkat ke siswa tersebut.

2.5. *Pindah departemen*

Staff kesiswaan menentukan siswa, apabila dia lulus maka staff kesiswaan melakukan pindah departemen ke siswa tersebut.

2.6. *Tinggal departemen*

Staff kesiswaan menentukan siswa, apabila dia tidak lulus maka staff kesiswaan tidak melakukan pindah departemen ke siswa tersebut.

4.2. Rekapitulasi Kebutuhan Sistem dan Perancangan Perangkat Lunak

Daftar rekapitulasi sistem terdiri dari kebutuhan fungsional dan kebutuhan non-fungsional. Tabel 4.16 merupakan rekapitulasi kebutuhan fungsional sistem:

Tabel 4.16 Daftar kebutuhan fungsional

ID	Requirements	Aktor	Use case
SRS_01	Sistem dapat mengolah proses penjadwalan	Staff akademik	Penjadwalan
SRS_02	Sistem dapat mengolah proses penilaian	Staff akademik	Penilaian
SRS_03	Sistem dapat menerima data siswa	Staff kesiswaan	Penerimaan siswa baru
SRS_04	Sistem dapat mengolah data presensi siswa	Staff kesiswaan	Input presensi
SRS_05	Sistem dapat menerima data mutasi	Staff kesiswaan	Input mutasi
SRS_06	Sistem dapat mengolah penempatan siswa ke tingkat maupun departemen yang lebih tinggi	Staff kesiswaan	Input penempatan siswa
SRS_07	Sistem dapat menerima data pegawai	Staff pegawai	Perekrutan
SRS_08	Sistem dapat mengolah agenda	Staff akademik	Menyusun agenda
SRS_09	Sistem dapat menerima data gaji pegawai	Staff akademik	Penggajian
SRS_10	Sistem dapat menerima data referensi	Staff Perpustakaan	Input pustaka
SRS_11	Sistem dapat mengolah data peminjaman	Staff Perpustakaan	Peminjaman
SRS_12	Sistem dapat mengolah data pengembalian	Staff Perpustakaan	Pengembalian
SRS_13	Sistem dapat mengelola pengguna / user yang berperan pada sistem.	Administrator	Pengaturan Pengguna

Daftar kebutuhan non-fungsional ditunjukkan pada tabel 4.17

Tabel 4.17 Daftar kebutuhan non-fungsional

Parameter	Deskripsi Kebutuhan
<i>Availability</i>	Perangkat lunak harus dapat beroperasi minimal 8 jam kerja.
<i>Compatibility</i>	Perangkat lunak harus dapat digunakan di sistem operasi <i>Windows dan Linux</i>
<i>Usability</i>	Perangkat lunak harus dapat digunakan dengan mudah oleh pengguna.

Response Time	Perangkat lunak harus dapat berjalan dengan waktu proses 3 detik.
Control	Membatasi akses penggunaan perangkat lunak yaitu staff dan administrator.

4.3 Perbandingan sistem Jibas dengan sistem yang dibuat

Setelah melakukan spesifikasi kebutuhan, tahap berikutnya adalah menganalisis komponen – komponen yang sesuai dengan kebutuhan awal. Pada tahap analisis kebutuhan prinsip kerja yang dilakukan bukanlah mengubah kebutuhan untuk disesuaikan dengan aplikasi, namun memodifikasi komponen agar sesuai dengan kebutuhan awal. Desain *reuse* Jibas digunakan sebagai dasar pembuatan sistem administrasi dan manajemen sekolah di SMAN 4 Jember. Sistem yang dikembangkan ini memodifikasi kerangka kerja Jibas dengan menambahkan modul baru pada sistem. Pada proses adopsi komponen yang ada di Jibas, seluruh komponen yang sesuai dengan kebutuhan di SMAN 4 Jember digunakan dan untuk proses modifikasi kebutuhan harus dipisahkan beberapa komponen yang ada di jibas sesuai analisis kebutuhan yang ada di SMAN4 Jember, sehingga menjadi Jibas modifikasi.

Modul dalam Jibas secara umum dapat dikategorikan dalam 3 modul besar yakni modul akademik yang mencakup semua data akademik, modul kepegawaian yang mencakup semua data kepegawian serta modul perpustakaan yang mengatur semua data perpustakaan. Dalam pengembangan sistem informasi sekolah SMAN 4 Jember ada 4 modul besar yakni modul akademik yang mencakup semua data akademik, modul kesiswaan yang mencakup semua data kesiswaan, modul kepegawaian yang mencakup semua data kepegawaian, modul perpustakaan yang mencakup semua data perpustakaan. Terjadi penambahan modul yakni modul kesiswaan dengan komponennya mengambil dari modul akademik Jibas, karena memang diperlukan pemisahan antara modul kesiswaan dan modul akademik di SMAN 4 Jember, sedangkan di Jibas komponen-komponen untuk modul kesiswaan terdapat di modul akademik. Melakukan modifikasi fitur dalam modul akademik, dengan menyesuaikan analisis kebutuhan sistem informasi manajemen sekolah SMAN 4 Jember. Tabel 4.18 merupakan daftar modul – modul yang sesuai dengan *use cases* yang dibutuhkan pada sistem

Tabel 4.18 Tabel perbandingan modul

Use Case	ERP modul (Jibas)	ERP modul pengembangan)
penjadwalan	Akademik	Akademik
	Jadwal	Jadwal
	Kalender Akdemik	Kalender Akademik
Penilaian	Akademik	Akademik
	Penilaian raport	Penilaian raport
Penerimaan Siswa Baru	Akademik	Kesiswaan
Input Presensi	Akademik	Kesiswaan
Pendataan Siswa	Akademik	Keiswaan
Input Mutasi	Akademik	Kesiswaan
Input Penempatan Siswa	-	Kesiswaan
Pengaturan pengguna	Manajemen user	Manajemen user
	User	User

Setiap modul tersebut memiliki beberapa fitur utama yang penting dalam perancangan sistem yang akan dikembangkan. Beberapa fitur dalam Jibas dimodifikasi sesuai dengan analisis kebutuhan dan fitur –fitur tersebut dapat dijelaskan pada tabel 4.19 berikut :

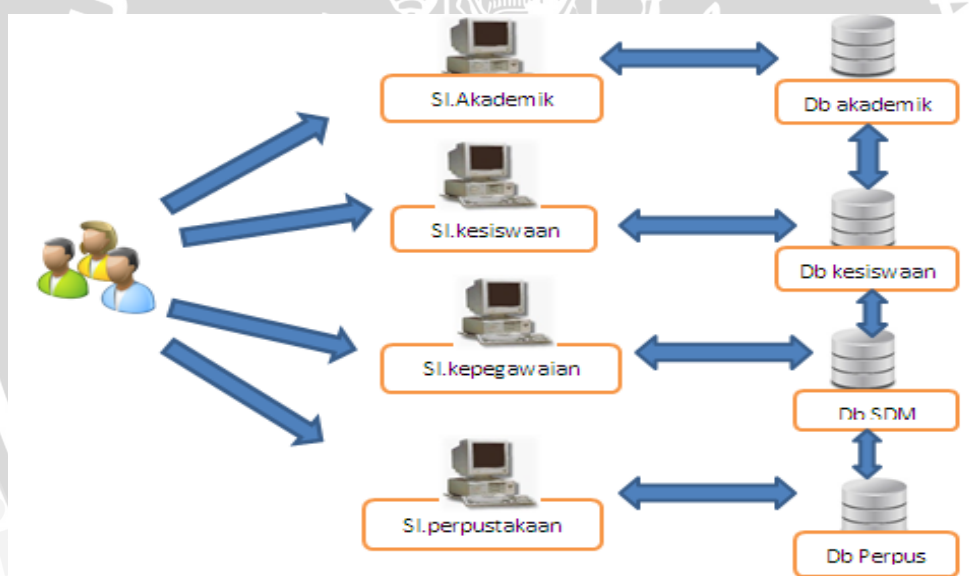
Tabel 4.19 Tabel daftar fitur

Modul	Fitur
1. Akademik	- Penghapusan Tambah Departemen
	- Penghapusan Hapus Departemen
	- Pendataan guru dan pelajaran
	- Penghapusan fitur PSB
	- Penghapusan Kesiswaan
	- Penghapusan presensi siswa
	- Penghapusan Kenaikan dan Kelulusan Siswa
	- Penghapusan mutasi
2. Kesiswaan	- Penambahan fitur PSB
	- Penambahan Input Siswa
	- Penambahan input presensi
	- Penempatan siswa dan kelulusan

	- Penambahan input mutasi
3. Kepegawaian	- CRUD (Create, update, delete) data pegawai
	- CRUD (Create, update, delete) gaji pegawai
	- CRUD (Create, update, delete) agenda pegawai
4. Perpustakaan	- CRUD (Create, update, delete) anggota perpustakaan
	- CRUD (Create, update, delete) referensi
	- CRUD (Create, update, delete) peminjaman
	- CRUD (Create, update, delete) pengembalian

4.4 Arsitektur Aplikasi SIMS Jibas

Aplikasi SIMS Jibas berbasis ERP dapat melakukan kegiatan layanan akademik, layanan kesiswaan, layanan perpustakaan, dan layanan kepegawaian. Gambar 4.18 berikut menunjukkan arsitektur aplikasi sistem secara umum.

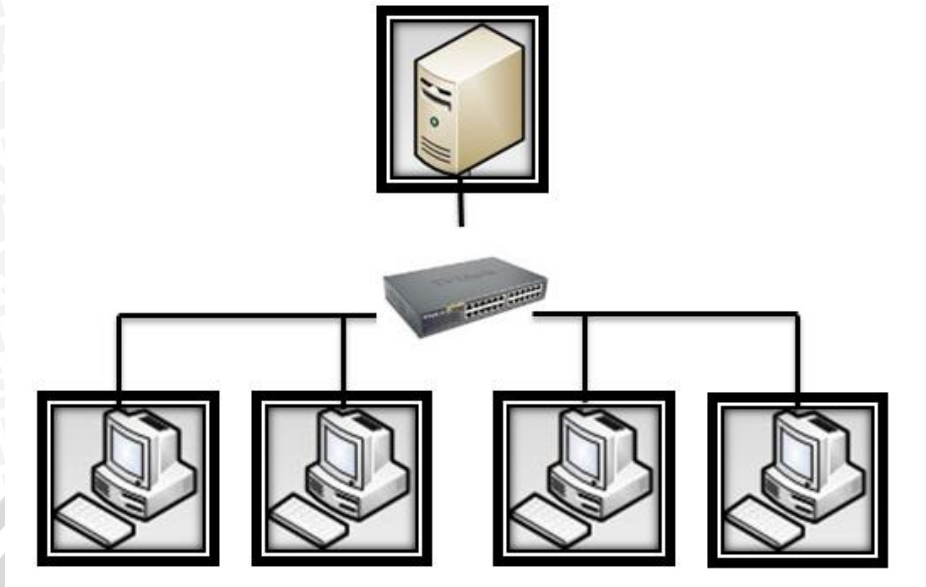


Gambar 4.18 Arsitektur aplikasi SIMS Jibas

Arsitektur SIMS jibas mempunyai beberapa basis data, yang semua basis data tersebut saling terhubung sehingga komunikasi untuk pertukaran data dapat dilakukan, Hal ini menjadikan integritas data semakin baik

4.5 Arsitektur Penerapan SIMS Jibas

Setelah melakukan analisis kebutuhan dan modifikasi sistem, langkah selanjutnya adalah melakukan penerapan SIMS Jibas di SMAN 4 Jember. Gambar 4.19 merupakan arsitektur penerapan SIMS Jibas



Gambar 4.19 Arsitektur Penerapan SIMS Jibas

Arsitektur Penerapan SIMS Jibas terdiri dari *server*, *hub* atau *switch*, kabel UTP (*Unshielded Twisted Pair*) dan *client*. *Server* dan *client* menggunakan komputer. Instalasi di *client* hanya membutuhkan web browser karena menerapkan konsep *thin client*, semua proses pengolahan data dilakukan di *server*. Kabel UTP dan hub atau switch digunakan untuk penghubung komunikasi data antara *client* dan *server*.

BAB V

IMPLEMENTASI

Dalam bab ini akan dibahas implementasi sistem berdasarkan analisis kebutuhan dan perancangan sistem yang sudah dijelaskan dalam bab III dan bab IV. Pembahasan terdiri dari spesifikasi sistem yang digunakan, implementasi antarmuka pengguna dan implementasi modul untuk beberapa pengguna yang berperan dalam sistem.

5.1 Spesifikasi Lingkungan Sistem

Spesifikasi lingkungan sistem dibagi menjadi 2 bagian yakni spesifikasi lingkungan perangkat keras dan spesifikasi lingkungan perangkat lunak.

5.1.1 Spesifikasi Lingkungan Perangkat Keras

Spesifikasi lingkungan perangkat keras yang digunakan dalam pengembangan sistem informasi manajemen SMAN 4 Jember berbasis ERP dijelaskan pada tabel 5.1 berikut.

Tabel 5.1 Spesifikasi Perangkat Keras Komputer

Notebook Accer 4732	
<i>Processor</i>	Intel(R) Core (TM)2 Duo CPU T6600 @2.20 GHz 2.20 GHz
<i>Memory (RAM)</i>	3 GB
<i>Hardisk</i>	Hitachi HTS547550A9E384 ATA Device / 500 GB
<i>Motherboard</i>	Accer Notebook Intel Motherboard
<i>Graphic Card</i>	Mobile Intel(R) 4 Series Express Chipset Family
<i>Monitor</i>	Generic PnP Monitor

5.1.2 Spesifikasi Lingkungan Perangkat Lunak

Spesifikasi lingkungan perangkat lunak yang digunakan dalam pengembangan sistem informasi manajemen SMAN 4 Jember berbasis ERP dijelaskan pada tabel 5.2 berikut.

Tabel 5.2 Spesifikasi Perangkat Lunak Komputer

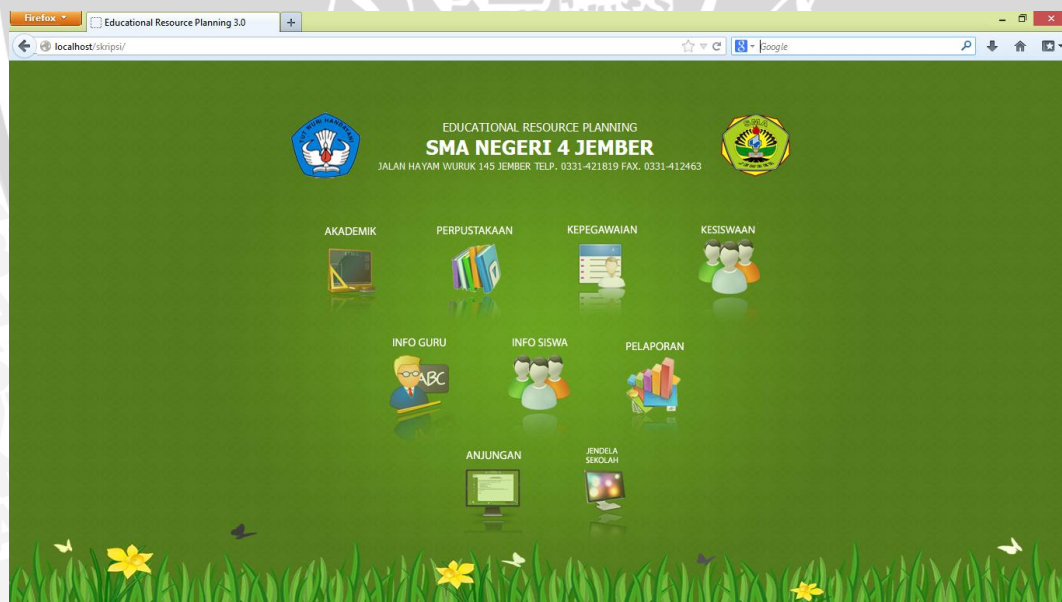
Notebook Accer 4732	
<i>Operaring System</i>	Microsoft Windows 7 Ultimate 32-bit
<i>DiretcX Version</i>	DirectX 11
<i>Programing Language</i>	PHP/HTML
<i>Application Programmable Interface</i>	Mozilla Firefox, Sublime, Xampp
<i>Database Management System</i>	Xampp win32-1.7.7

5.2 Implementasi Antarmuka

Implementasi antarmuka dalam sistem ini dibagi menjadi 5, yakni antarmuka aplikasi antarmuka aplikasi, antarmuka modul akademik ,antamuka modul kesiswaan, antarmuka modul perpustakaan, dan antarmuka modul kepegawaian.

5.2.1 Antarmuka aplikasi

Gambar 5.1 merupakan gambar dari antarmuka aplikasi SIMS Jibas yang diterapkan di SMAN 4 Jember

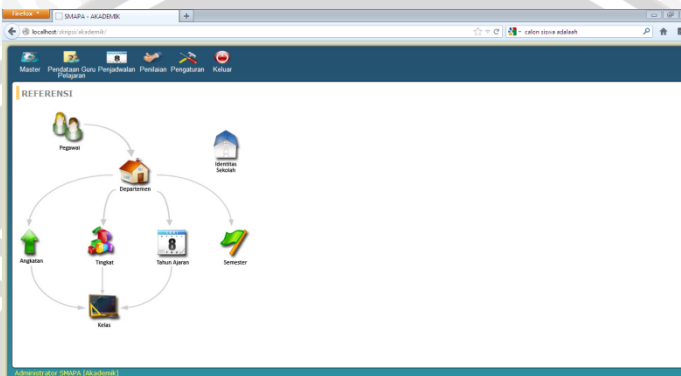


Gambar 5.1 Antarmuka aplikasi

Gambar 5.1 menjelaskan bagaimana bentuk tampilan dari halaman awal aplikasi. Antarmuka aplikasi SIMS Jibas ini memiliki modul – modul akademik, perpustakaan, kepegawaian dan kesiswaan.

5.2.2 Antarmuka Akademik

Gambar 5.2 merupakan gambar dari antarmuka akademik aplikasi SIMS Jibas yang diterapkan di SMAN 4 Jember

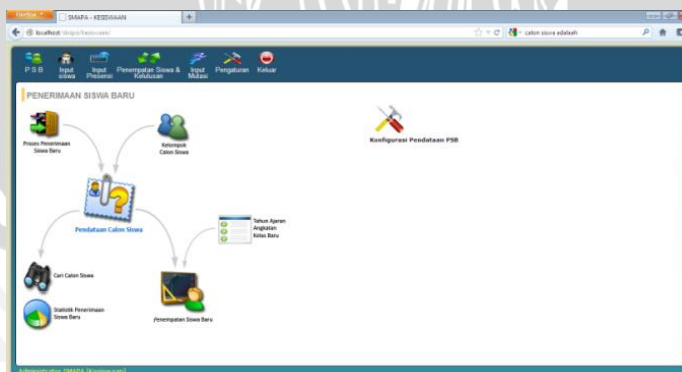


Gambar 5.2 Antarmuka Akademik

Antarmuka modul akademik memiliki beberapa pengurangan fitur pengembangan diantaranya tambah departemen, kesiswaan, presensi siswa, mutasi siswa. Untuk penjelasan *source code* dari isi modul akademik akan dibahas pada sub bab 5.3.

5.2.3 Antarmuka Kesiswaan

Gambar 5.3 merupakan gambar dari antarmuka kesiswaan aplikasi SIMS Jibas yang diterapkan di SMAN 4 Jember



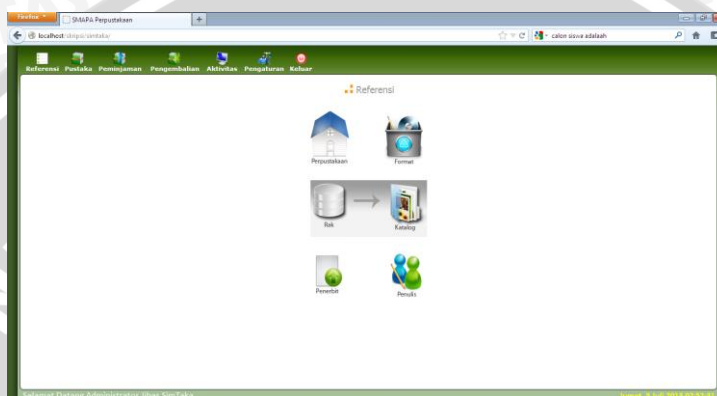
Gambar 5.3 Antarmuka Kesiswaan

Gambar 5.3 menjelaskan bagaimana bentuk tampilan dari modul kesiswaan. Antarmuka. Modul kesiswaan memiliki beberapa sub menu

diantaranya psb, input siswa, input presensi, penempatan siswa dan kelulusan, input mutasi Untuk penjelasan *source code* dari isi modul kesiswaan akan dibahas pada sub bab 5.3.

5.2.4 Antarmuka Perpustakaan

Gambar 5.4 merupakan gambar dari antarmuka kesiswaan aplikasi SIMS Jibas yang diterapkan di SMAN 4 Jember



Gambar 5.4 Antarmuka Perpustakaan

Gambar 5.4 menjelaskan bagaimana bentuk tampilan modul perpustakaan. Antarmuka modul perpustakaan memiliki beberapa menu diantaranya referensi, pustaka, peminjaman, pengembalian, aktifitas, pengaturan dan keluar.

5.2.5 Antarmuka Kepegawaian

Gambar 5.5 merupakan gambar dari antarmuka aplikasi kepegawaian SIMS Jibas yang diterapkan di SMAN 4 Jember



Gambar 5.5 Antarmuka Kepegawaian

Gambar 5.5 menjelaskan bagaimana bentuk tampilan modul kepegawaian. Antarmuka modul kepegawaian memiliki beberapa menu diantaranya referensi, kepegawaian, pengaturan dan keluar.

5.3 Implementasi Modul pada bagian sekolah

Pada sub bab ini akan dibahas detail dari pengimplementasian sourcecode pada modul akademik dan modul kesiswaan. Setiap modul bisa memiliki lebih dari 1 submenu dan banyak fitur penting dalam pengolahannya. Berikut adalah detailnya :

5.3.1 Modul Akademik

Modul akademik ini mengolah data – data yang berhubungan dengan kegiatan akademik baik dalam pengelolaan jadwal pelajaran, maupun pengelolaan penilaian. Adapun data yang diolah oleh modul ini sudah dijelaskan pada Tabel daftar fitur di Bab IV. *Source code* berikut menjelaskan halaman dari modul akademik.

Tabel 5.3 *Source Code* tampilan modul akademik

<i>Source code</i> modul_akademik
<pre><? //memanggil fungsi cek.php //mengeset tema akademik //mengeset halaman awal akademik <frame name="content" src="referensi.php"/> <frame name="frameright" src="frameright.php" scrolling="no" noresize="noresize" /> </frameset> <frame name="framebottom" src="framebottom.php" scrolling="no" noresize="noresize" /></pre>

5.3.2 Modul Kesiswaan

Modul kesiswaan mempunyai menu diantaranya psb, input siswa, input presensi, penempatan siswa dan kelulusan, input mutasi, pengaturan dan keluar. Diantara sub menu terdapat beberapa fitur yang digunakan untuk mengatur aktifitas yang ada di kesiswaan. Berikut adalah *source code* halaman awal modul kesiswaan:

Tabel 5.4 *Source Code* tampilan modul kesiswaan

Source code mod_kesiswaan

```
<?
require_once("cek.php");
if (isset($_REQUEST['theme']))
    $theme=$_REQUEST['theme'];
?>
<title>SMAPA - KESISWAAN</title>
<frameset border="0" frameborder="0" framespacing="0"
rows="87,*,41">
//Frame Kanan, kiri,atas dan bawah
<frame name="content" src="siswa_baru.php"/>
```



BAB VI

PENGUJIAN DAN ANALISIS SISTEM

Bab ini membahas mengenai tahapan pengujian dan analisis perangkat lunak Jibas yang telah dikembangkan. Proses pengujian dilakukan melalui lima tahapan pengujian, yaitu pengujian unit, pengujian integrasi, pengujian validasi, pengujian *compatibility*, pengujian *usability*. Pada pengujian unit dan integrasi, akan digunakan teknik pengujian *white-box testing*. Pada pengujian validasi akan digunakan teknik pengujian *black-box testing*. Untuk pengujian *availability* dilakukan proses operasional sistem minimal 8 jam. Untuk pengujian *compatibility* dilakukan installasi di *sistem operasi* Linux dan Windows, untuk strategi pengujian *usability* diberikan kuisioner kepada staff pengguna SISM Jibas di SMAN 4 Jember, untuk pengujian *response time* melihat *response* yang diberikan sistem dan untuk pengujian *control* dilakukan dengan melakukan pembatasan akses penggunaan perangkat lunak yaitu staff dan *administrator*.

6.1 Pengujian

Proses pengujian yang dilakukan melalui lima tahapan pengujian yaitu pengujian unit, pengujian integrasi, pengujian validasi, pengujian *availability*, pengujian *compatibility*, pengujian *usability*, pengujian *response time*, dan pengujian *control*.

6.1.1 Pengujian Unit

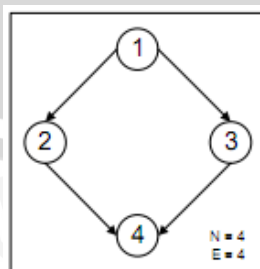
Pada pengujian unit SISM Jibas digunakan teknik *white-box testing* dengan teknik *basis path testing*. Pada teknik *basis path testing*, proses pengujian dilakukan dengan memodelkan algoritma pada suatu *flow graph*, menentukan jumlah kompleksitas siklomatis (*cyclomatic complexity*), menentukan sebuah basis set dari jalur independen dan memberikan kasus uji (*test case*) pada setiap basis set yang telah ditentukan.

6.1.1.1 Pengujian Unit untuk algoritma Login

Login merupakan persyaratan untuk memasuki modul yang lebih besar. *Login* berfungsi untuk melakukan seleksi kondisi pada SISM Jibas. Tabel 6.1 menunjukkan proses pengujian unit untuk algoritma *login*, dan Gambar 6.1 menunjukkan pemodelan *flow graph* pada algoritma *login*.

Tabel 6.1 Pengujian unit untuk algoritma login

Nama algoritma <i>login</i>	
	<p>Deklarasi</p> <ul style="list-style-type: none"> String → user, pass <p>Deskripsi</p> <ul style="list-style-type: none"> Masukan : user, pass Proses <ol style="list-style-type: none"> Memberikan nilai atribut user melalui set user pada variabel user Memberikan nilai atribut pass melalui set pass pada variable pass. Menjalankan seleksi kondisi pengecekan form, untuk mengecek apakah ada data user dan pass yang dimasukkan Jika tidak ada data yang dimasukkan maka variabel akan memunculkan alert dan mengembalikan nilai <i>false</i>. Jika data ada maka variabel akan mengembalikan nilai <i>true</i>. Melakukan pengembalian nilai dari variabel status. <ul style="list-style-type: none"> Keluaran : nilai variabel



Gambar 6.1 Flow Graph Algoritma login

Pemodelan ke dalam flow graph yang telah dilakukan terhadap algoritma *login* menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau edge (garis penghubung antar node) dan N merupakan jumlah simpul (node).

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 4 - 4 + 2 \\ &= 2 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan dua buah basis set dari jalur independent, yaitu :

Jalur 1 : 1 – 2 – 4

Jalur 2 : 1 – 3 – 4

Penentuan kasus uji untuk jalur independent tersebut dan hasil eksekusinya dijelaskan pada Tabel 6.2.

Tabel 6.2 Kasus Uji untuk Pengujian Unit Login

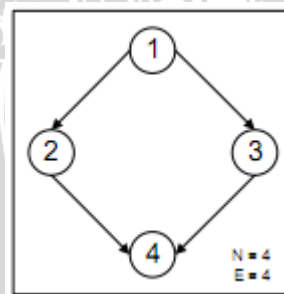
Jalur	Kasus Uji	Hasil yang Diharapkan	Hasil yang Didapatkan
1.	Memberi nilai atribut user dan pass dengan data yang sudah ada di basis data, kemudian mengeksekusi algoritma <i>login</i>	Fungsi memberikan nilai kembalian dengan nilai true.	Fungsi memberikan nilai kembalian dengan nilai true.
2.	Tidak Memberi nilai atribut user dan pass atau memberikan dengan data yang berbeda dengan yang ada di basis data, kemudian mengeksekusi algoritma <i>login</i>	Fungsi memberikan nilai kembalian dengan nilai false dan memunculkan alert.	Fungsi memberikan nilai kembalian dengan nilai false dan memunculkan alert.

6.1.1.2 Pengujian untuk algoritma index

Index merupakan algoritma awal untuk menentukan apakah sistem akan melakukan login atau langsung masuk ke dalam modul. Tabel 6.3 menunjukkan proses pengujian unit untuk algoritma index, dan Gambar 6.2 menunjukkan pemodelan *flow graph* pada algoritma index.

Tabel 6.3 Pengujian unit untuk algoritma index

Nama algoritma index	
Deskripsi	<ul style="list-style-type: none"> • Memasukkan konfigurasi sistem • Mengeset nama session • Memanggil session • Proses
1	<ul style="list-style-type: none"> 1. Menjalankan seleksi kondisi session 2. Jika masih terdapat session, maka akan memasuki halama awal secara langsung
3	<ul style="list-style-type: none"> 3. Jika sessionnya tidak ada maka akan masuk ke login 4. Melakukan Pengembalian session



Gambar 6.2 Flow graph algoritma index

Pemodelan ke dalam *flow graph* yang telah dilakukan terhadap algoritma index menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau edge (garis penghubung antar node) dan N merupakan jumlah simpul (node).

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 4 - 4 + 2 \\
 &= 2
 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan dua buah basis set dari jalur independent, yaitu :

Jalur 1 : 1 – 2 – 4

Jalur 2 : 1 – 3 – 4

Penentuan kasus uji untuk jalur independent tersebut dan hasil eksekusinya dijelaskan pada Tabel 6.4.

Tabel 6.4 Kasus Uji untuk Pengujian Unit index

Jalur	Kasus Uji	Hasil yang Diharapkan	Hasil yang Didapatkan
1.	Membiarkan session ada, dengan cara masuk ke modul tanpa harus melakukan <i>logout</i>	Sistem langsung masuk ke modul yang sessionnya ada.	Sistem langsung masuk ke modul yang sessionnya ada.
2.	Tidak memberikan session ke modul yang akan diuji	Sistem akan menampilkan form login	Sistem menampilkan form login.

6.1.2 Pengujian Integrasi

Pengujian integrasi diterapkan pada proses yang mengintegrasikan fungsionalitas dari beberapa fungsi untuk melakukan sebuah operasi tertentu. Pada pengujian integrasi yang dijadikan sebagai objek uji adalah inti program yang menggabungkan kinerja dari *login* dan antarmuka halaman awal. Pengujian integrasi menggunakan strategi *bottom-up*, modul - modul yang diintegrasikan masing - masing diuji terlebih dahulu dalam pengujian unit dan kemudian bergerak menuju ke pengujian modul - modul yang mengintegrasikannya.

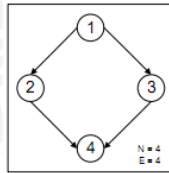
6.1.2.1 Pengujian Integrasi untuk *redirect*

Redierect merupakan inti program yang berfungsi untuk mengatur pengguna yang akan masuk ke dalam modul. Di dalam *redirect* terdapat set

username untuk admin, tabel 6.5 menunjukkan proses pengujian pada algoritma *redirect* dan gambar 6.3 menunjukkan pemodelan *flow graph* pada algoritma *redirect*

Tabel 6.5 Pengujian integrasi untuk algoritma *redirect*

Nama algoritma <i>redirect</i>		
	<p>Deklarasi</p> <ul style="list-style-type: none"> • String → username, password, tingkat, departemen • Integer → status • SetAdmin <p>Deskripsi</p> <ul style="list-style-type: none"> • Masukan : username, password, tingkat, departemen • SetAdmin • Proses 	
1	<p>1. Melakukan set password</p> <p>Melakukan pemanggilan method seleksi kondisi, jika nilai parameter username atau password kosong maka muncul alert untuk memasukkan username dan password</p> <p>2. Jika nilai parameter username dan password tidak kosong maka akan dilakukan langkah – langkah sebagai berikut :</p>	
	<p>a. Pengecekan query databse untuk proses penentuan <i>login</i> dengan masukan username dan password, variabel status diberi nilai sesuai dengan nilai kembalian dari method <i>login</i>.</p>	2
3	<p>b. Jika variabel status bernilai 1, maka variable tingkat diberi nilai “Username dan Password administrator”</p> <p>c. Jika variabel status tidak bernilai 1, maka ditampilkan untuk tingkat staff</p>	
	<ul style="list-style-type: none"> • Keluaran : Nilai kembalian variabel tingkat 	4



Gambar 6.3 Flow graph redirect

Pemodelan ke dalam *flow graph* yang telah dilakukan terhadap algoritma *redirect* menghasilkan jumlah kompleksitas siklomatis (cyclomatic complexity) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau edge (garis penghubung antar node) dan N merupakan jumlah simpul (node).

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 4 - 4 + 2 \\ &= 2 \end{aligned}$$

Berdasarkan dari nilai cyclomatic complexity yang telah didapatkan dari perhitungan maka ditentukan dua buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1 – 3 – 4

Jalur 2 : 1 – 2 – 4

Penentuan kasus uji untuk jalur *independent* tersebut dan hasil eksekusinya dijelaskan pada Tabel 6.6

Tabel 6.6 Kasus uji untuk pengujian integrasi *redirect*

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1.	Membiarkan data masuk kosong	Proses akan terhenti dan ditampilkan pesan	Proses terhenti dan ditampilkan pesan
2.	Memberikan nilai parameter username dan password, sesuai dengan nilai parameter yang ada di basis data kemudian mengeksekusi	Data masukan akan diolah dengan mengeksekusi query yang memberikan nilai kembalian. Modul akan terbuka sesuai dengan hak akses pengguna	Data masukan diolah dengan mengeksekusi query yang memberikan nilai kembalian. Modul akan terbuka sesuai dengan hak akses pengguna



6.1.3 Pengujian Validasi

Pengujian validasi digunakan untuk mengetahui apakah sistem yang dibangun sudah benar sesuai dengan yang dibutuhkan. Item - item yang telah dirumuskan dalam daftar kebutuhan yang merupakan hasil analisis kebutuhan akan menjadi acuan untuk melakukan pengujian validasi. Pengujian validasi menggunakan metode pengujian *black-box*, karena tidak diperlukan konsentrasi terhadap alur jalannya algoritma program dan lebih ditekankan untuk menemukan konformitas antara kinerja sistem dengan daftar kebutuhan. Pada skripsi ini dilakukan pengujian validasi terhadap perangkat SIMS Jibas.

6.1.3.1 Kasus uji pengolahan data penjadwalan

Tabel 6.7 Kasus uji pengolahan data penjadwalan

Nama kasus uji	Kasus uji pengolahan penjadwalan
Objek uji	Kebutuhan fungsional (SRS_01)
Tujuan pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional penjadwalan
Prosedur uji	<ol style="list-style-type: none"> 1. staff <i>login</i> ke aplikasi 2. Karena data pelajaran sudah ada maka langsung ke halaman penjadwalan 3. staff menekan tombol “tambah”
Hasil yang diharapkan	Aplikasi dapat menyimpan data penjadwalan dan jika aplikasi berjalan benar maka akan menampilkan jadwal di waktu, tingkat, kelas dan pelajaran yang diharapkan.
Hasil yang diperoleh	Aplikasi dapat menyimpan data penjadwalan dan jika aplikasi berjalan benar maka akan menampilkan jadwal di waktu, tingkat, kelas dan pelajaran yang diharapkan.
Status	Valid

6.1.3.2 Kasus uji pengolahan data penilaian

Tabel 6.8 Kasus uji pengolahan data penilaian

Nama kasus uji	Kasus uji pengolahan data penilaian
Objek uji	Kebutuhan fungsional (SRS_02)
Tujuan pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional penilaian
Prosedur uji	<ol style="list-style-type: none"> 1. staff <i>login</i> ke aplikasi 2. staff menerima nilai dari guru 3. staff memasukkan nilai dari guru ke sistem 4. staff menekan tombol “simpan” pada aplikasi
Hasil yang diharapkan	Aplikasi menampilkan nilai siswa secara tepat, jika aplikasi benar maka aplikasi menampilkan nilai siswa sesuai inputan staff
Hasil yang diperoleh	Aplikasi menampilkan nilai siswa sesuai dengan inputan staff
Status	Valid

6.1.3.3 Kasus uji pengolahan data penerimaan siswa baru

Tabel 6.9 Kasus uji pengolahan data penerimaan siswa baru

Nama kasus uji	Kasus uji pengolahan data penerimaan siswa baru
Objek uji	Kebutuhan fungsional (SRS_03)
Tujuan pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional modul kesiswaan penerimaan siswa baru
Prosedur uji	<ol style="list-style-type: none"> 1. staff <i>login</i> ke aplikasi 2. staff memilih menu penerimaan siswa baru 3. staff memasukkan data – data master penerimaan siswa baru 4. staff memasukkan data calon siswa ke form calon siswa dan menekan tombol “simpan”.
Hasil yang diharapkan	Aplikasi mampu menyimpan data calon siswa sesuai

	dengan jalur masuk penerimaan siswa baru.
Hasil yang diperoleh	Aplikasi mampu menyimpan data calon siswa sesuai dengan jalur masuk penerimaan siswa baru.
Status	Valid

6.1.3.4. Kasus uji pengolahan data pengembalian buku

Tabel 6.10 Kasus uji pengolahan data pengembalian buku

Nama kasus uji	Kasus uji pengolahan data pengembalian buku
Objek uji	Kebutuhan fungsional (SRS_12)
Tujuan pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional modul perpustakaan pengembalian buku
Prosedur uji	<ol style="list-style-type: none"> 1. staff <i>login</i> ke aplikasi 2. staff menekan tombol “pengembalian” kemudian memasukkan kode buku yang dipinjam oleh peminjam 3. Kode buku ditampilkan dan dipilih oleh staff. 4. Staff memasukkan denda apabila pengembalian melebihi batas peminjaman 5. Staff perpustakaan menekan tombol “simpan”
Hasil yang diharapkan	Aplikasi akan mampu mengelola dan menyimpan data pengembalian
Hasil yang diperoleh	Aplikasi mampu mengelola dan menyimpan data pengembalian
Status	Valid

6.1.3.5 Kasus uji pengolahan data pengaturan pengguna

Tabel 6.11 Kasus uji pengolahan data pengaturan pengguna

Nama kasus uji	Kasus uji pengolahan data pengaturan pengguna
Objek uji	Kebutuhan fungsional (SRS_13)
Tujuan pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional Administrator

	dalam mengolah pengguna yang akan menggunakan sistem
Prosedur uji	<ol style="list-style-type: none"> 1. Admin <i>login</i> ke aplikasi 2. Admin menekan tombol “pengaturan” untuk mengolah data pengguna yang akan menggunakan system
Hasil yang diharapkan	Aplikasi akan mampu mengolah data pengguna yang akan menggunakan sistem
Hasil yang diperoleh	Aplikasi mampu mengolah data pengguna yang akan menggunakan system
Status	Valid

6.1.4 Pengujian non fungsional

Pada pengujian non fungsional ini sistem akan diuji secara *availability*, *compatibility*, *usability*, *response time*, dan *control*. Pada pengujian *availability* sistem akan dibiarkan beroperasi minimal 8 jam kerja, pada pengujian *compatibility* sesuai daftar kebutuhan pada bab IV maka sistem akan diaplikasikan pada 2 sistem operasi yang berbeda yakni *windows* dan *linux*. Untuk pengujian *usability* sistem akan diuji dengan menggunakan kuisisioner tentang kemudahan penggunaan sistem kepada para pengguna atau staff di SMAN 4 Jember. Pengujian ini digunakan untuk melihat seberapa besar kepuasan pengguna dalam menggunakan aplikasi yang telah dikembangkan dan juga digunakan sebagai tolak ukur kesuksesan dalam mengembangkan suatu system. Untuk pengujian *response time* perangkat lunak harus dapat berjalan dengan waktu proses 3 detik, hal ini dilakukan untuk melihat *response* yang diberikan sistem. Untuk pengujian *control* sistem dapat membatasi akses penggunaan perangkat lunak yaitu staff dan administrator.

6.1.4.1 Pengujian *Availability*

Untuk melakukan pengujian *availability* langkah pertama adalah melakukan analisis kebutuhan yang diperlukan untuk menguji sistem. Tabel 6.12 adalah daftar kebutuhan pengujian *availability*:

Tabel 6.12 Daftar pengujian *availability*

Daftar kebutuhan pengujian <i>compatibility</i>
1. <i>Server</i> berupa komputer
2. <i>Client</i> berupa komputer
3. Switch atau Hub
4. Kabel UTP
5. Mozilla firefox

Setelah didapatkan daftar kebutuhan yang diperlukan untuk menguji sistem langkah selanjutnya adalah membuat skenario pengujian *availability* sistem yakni:

Tabel 6.13 Skenario pengujian *Availability*

Skenario pengujian <i>availability</i>	Hasil yang diperoleh
1. Melakukan installasi pada komputer yang akan dijadikan <i>server</i> .	Berdasarkan pengujian <i>availability</i> ini maka diperoleh suatu hasil yakni sistem dapat berjalan minimal 8 jam dengan melakukan proses operasional secara normal atau sistem tidak mengalami <i>error</i> selama proses pengujian. Sesuai dengan hasil yang didapat sistem dapat memenuhi kebutuhan non-fungsional <i>availability</i> .
2. Membuat arsitektur perapan SIMS Jibas sesuai dengan gambar 4.19	
3. Melakukan setting ip di <i>server</i> dan masing – masing <i>client</i> .	
4. Mengoperasikan sistem dari jam 07.00 - 15.00	

6.1.4.2 Pengujian *Compatibility*

Untuk melakukan pengujian *compatibility* langkah pertama adalah melakukan analisis kebutuhan yang diperlukan untuk menguji sistem. Tabel 6.14 adalah daftar kebutuhan pengujian *compatibility* pada sistem operasi *Linux* :

Tabel 6.14 Daftar pengujian *compatibility Linux*

Daftar kebutuhan pengujian <i>compatibility</i>
1. Linux Ubuntu 12.04
2. Lampp 1.8.1
3. Mozzila firefox 15

Setelah didapatkan kebutuhan perangkat lunak yang diperlukan untuk menguji sistem langkah selanjutnya adalah membuat skenario pengujian *compatibility* sistem yakni :

Tabel 6.15 Skenario pengujian *compatibility Linux*

Skenario pengujian <i>compatibility</i>	Hasil yang diperoleh
1. Instalasi Linux Ubuntu 12.04	Berdasarkan pengujian ini maka diperoleh suatu hasil yakni sistem dapat berjalan dengan lancar pada sistem operasi Linux Ubuntu 12.04
2. Instalasi Lampp 1.8.1	
3. Salin file projek ke dalam folder htdocs	
4. Import basis data ke phpmyadmin	
5. Buka localhost/skripsi dan <i>login</i> dengan masing – masing user yang ada	

Berikut untuk pengujian pada sistem operasi Windows :

Tabel 6.16 Daftar pengujian *compatibility Windows*

Daftar kebutuhan pengujian <i>compatibility</i>
1. Windows 7 ultimate
2. Xampp 1.8.1
3. Mozzila firefox 15

Setelah didapatkan perangkat lunak yang diperlukan untuk menguji sistem langkah selanjutnya adalah membuat skenario pengujian *compatibility* sistem yakni :

Tabel 6.17 Skenario pengujian *compatibility* Windows

Skenario pengujian <i>compatibility</i>	Hasil yang diperoleh
<ol style="list-style-type: none"> 1. Instalasi Windows 7 ultimate 2. Instalasi Xampp 1.8.1 3. Salin file projek ke dalam folder htdoc 4. Import basis data ke phpmyadmin 5. Buka localhost/skripsi dan <i>login</i> dengan masing – masing user yang ada 	Berdasarkan pengujian ini maka diperoleh suatu hasil yakni sistem dapat berjalan dengan lancar pada sistem operasi Windows 7 ultimate.

6.1.4.3 Pengujian *Usability*

Pengujian *usability* ini dilakukan dengan cara memberikan kuisioner kepada 8 orang dari 4 staff yang ada di SMAN 4 Jember, secara garis besar berisi pertanyaan – pertanyaan yang berkaitan dengan kepuasan pengguna terhadap sistem yang dibuat. Dari kuisioner yang telah diisi oleh para pengguna atau staff SMAN 4 Jember diperoleh hasil sebagai berikut :

Tabel 6.18 Tabel hasil kuisioner

Pertanyaan	Hasil
1. Bagaimanakah menurut Anda mengenai tampilan aplikasi yang dibuat?	75 % pengguna dari ke-4 jabatan yakni staff akademik, staff kesiswaan, staff kepegawaian, dan staff perpustakaan menyatakan bahwa aplikasi memiliki tampilan yang sederhana dan mudah untuk dipahami. Karena terdapat alur untuk menyelesaikan satu kegiatan, dan 25 % pengguna dari 4 jabatan menyatakan bahwa aplikasi memiliki tampilan yang tidak sederhana dan sulit untuk dipahami. Karena untuk mengatur 1 proses bisnis harus melalui

	beberapa tahapan.
2. Bagaimanakah menurut Anda mengenai menu – menu yang disediakan oleh aplikasi?	87,5 % pengguna dari ke-4 jabatan yakni staff akademik, staff kesiswaan, staff kepegawaian, dan staff perpustakaan menyatakan bahwa aplikasi memiliki menu – menu yang cukup untuk memenuhi proses bisnis sehari – hari dan menu – menu tersebut mudah untuk dimengerti secara langsung karena menggunakan bahasa yang sudah umum bagi mereka. Dan 12,5 % pengguna dari 4 jabatan tersebut menyatakan bahwa aplikasi memiliki menu-menu yang terlalu banyak sehingga untuk menyelesaikan proses bisnis sehari – hari harus mengatur menu lain terlebih dahulu.
3. Apakah aplikasi yang dikembangkan turut membantu dalam mempermudah kinerja pengguna?	87,5 % pengguna dari ke-4 jabatan yakni staff akademik, staff kesiswaan, staff kepegawaian, dan staff perpustakaan menyatakan bahwa aplikasi sangat membantu kinerja proses bisnis sehari – hari karena dengan adanya aplikasi mempercepat pelayanan. Dan 12,5% dari pengguna menyatakan bahwa aplikasi belum sepenuhnya membantu dalam mempermudah kinerja pengguna karena masih kesulitan untuk mengoperasikan aplikasi.

<p>4. Bagaimana pendapat anda jika aplikasi ini digunakan dan terus dikembangkan hingga menjadi aplikasi dalam sekolah yang lebih kompleks?</p>	<p>87,5% pengguna dari ke-4 jabatan staff akademik, staff kesiswaan, staff kepegawaian, dan staff perpustakaan menyatakan bahwa aplikasi dapat dikembangkan menjadi aplikasi sekolah yang lebih kompleks karena menu – menu yang disediakan oleh aplikasi cukup lengkap untuk diimplementasikan di sekolah. Dan 12,5% dari pengguna menyatakan bahwa aplikasi sudah cukup untuk aplikasi sekolah.</p>
<p>5. Apakah perlu dilakukan pelatihan untuk pengoperasional sistem?</p>	<p>100% pengguna menjawab perlu adanya pelatihan untuk pengoperasionalan sistem dalam jangkawaktu rata-rata 2 minggu.</p>

6.1.4.4 Pengujian *Response Time*

Untuk melakukan pengujian *response time* langkah pertama adalah melakukan analisis kebutuhan yang diperlukan untuk menguji sistem. Tabel 6.19 adalah daftar kebutuhan pengujian *response time* pada sistem

Tabel 6.19 Daftar Pengujian *Response Time*

Daftar kebutuhan pengujian <i>response time</i>
1. <i>Server</i> berupa komputer
2. <i>Client</i> berupa komputer
3. Switch atau Hub
4. Kabel UTP
5. Mozilla firefox

Setelah didapatkan daftar kebutuhan yang diperlukan untuk menguji sistem langkah selanjutnya adalah membuat skenario pengujian *response time* sistem yakni pada tabel 6.20

Tabel 6.20 Skenario pengujian *response time*

Skenario pengujian <i>response time</i>	
1.	Melakukan instalasi pada komputer yang akan dijadikan <i>server</i> .
2.	Membuat arsitektur perapan SIMS Jibas sesuai dengan gambar 4.19
3.	Melakukan setting ip di <i>server</i> dan masing – masing di <i>client</i> .
4.	Melakukan pengamatan <i>response time</i> yang diberikan oleh sistem.

Hasil yang diperoleh dari pengujian *response time* sesuai dengan skenario pada tabel 6.20 dibuktikan dengan tabel 6.21

Tabel 6.21 Hasil pengujian *response time* pada 10 halaman berbeda

No.	Pengujian 1	Modul	Response time
1.	Halaman awal	-	2.01 s
2.	Login	Semua Modul	1.01 s
3.	Penjadwalan	Akademik	2.96 s
4.	Penilaian	Akademik	3.06 s
5.	PSB	Kesiswaan	2.71 s
Rata – rata pengujian			2,35 s

Dari hasil pengujian *response time* pada tabel 6.21 sistem dapat memberikan *response* rata-rata 2,35 s terhadap 5 pengujian tersebut, dan hasil tersebut tidak selalu akurat karena masih dapat dipengaruhi oleh koneksi.

6.1.4.4 Pengujian *Control*

Untuk melakukan pengujian *control* langkah pertama adalah melakukan analisis kebutuhan yang diperlukan untuk menguji sistem. Tabel 6.22 adalah daftar kebutuhan pengujian *control*

Tabel 6.22 Daftar Pengujian *Control*

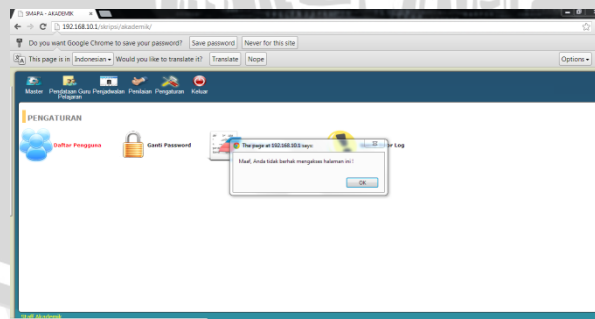
Daftar kebutuhan pengujian <i>response time</i>
1. <i>Server</i> berupa komputer
2. <i>Client</i> berupa komputer
3. Switch atau Hub
4. Kabel UTP
5. Mozilla firefox

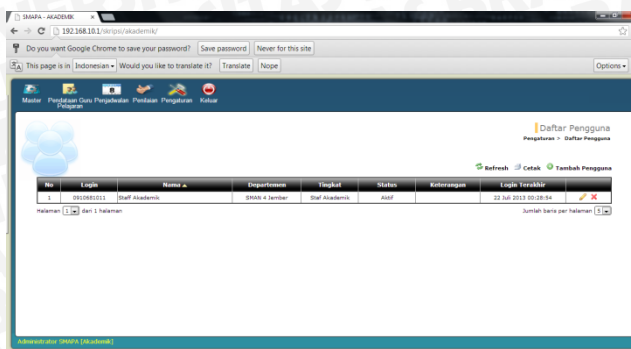
Setelah didapatkan daftar kebutuhan yang diperlukan untuk menguji sistem langkah selanjutnya adalah membuat skenario pengujian *control* sistem yakni pada tabel 6.23

Tabel 6.23 Skenario Pengujian *Control*

Skenario pengujian <i>response time</i>
1. Melakukan instalasi pada komputer yang akan dijadikan <i>server</i> .
2. Membuat arsitektur perapan SIMS Jibas
3. Melakukan setting ip di <i>server</i> dan masing – masing di <i>client</i> .
4. Menjalankan aplikasi sebagai administrator dan staff.

Hasil dari pengujian *control* pada skenario tabel 6.23 dibuktikan dengan gambar 6.5, 6.6, dan tabel 6.24

Gambar 6.5 *Control* aplikasi login sebagai staffGambar 6.5 *Control* aplikasi login sebagai administrator



Tabel 6.24 Hasil pengujian control terhadap 5 pengujian *use case*

No	Pengujian Usecase	Modul	Hasil	
			Administrator	Staff
1.	Penjadwalan	Akademik	✓	✓
2.	Penilaian	Akademik	✓	✓
3.	PSB	Kesiswaan	✓	✓
4.	Manajemen Pengguna	Seluruh Modul	✓	-
5.	Ubah Pengguna	Seluruh Modul	✓	✓

Berdasarkan pengujian *control* pada tabel 6.24 maka diperoleh suatu hasil yakni sistem dapat membatasi akses penggunaan administrator dan staff, administrator dapat melakukan manajemen pengguna sedangkan staff tidak dapat melakukan manajemen pengguna.

6.1.5 Analisis

Proses analisis bertujuan untuk mendapatkan kesimpulan dari hasil pengujian perangkat lunak SIMS Jibas yang telah dilakukan. Proses analisis mengacu pada dasar teori sesuai dengan hasil pengujian yang didapatkan. Analisis dilakukan terhadap hasil pengujian di setiap tahap pengujian. Proses analisis yang dilakukan meliputi analisis hasil pengujian unit, analisis hasil pengujian integrasi, analisis hasil pengujian validasi, analisis hasil pengujian *availability*, analisis hasil pengujian *compatibility*, dan analisis hasil pengujian *usability*, analisis hasil pengujian *response time*, analisis hasil pengujian *control*.

6.1.5.1 Analisa Pengujian Unit

Proses analisis terhadap hasil pengujian unit dilakukan dengan melihat kesesuaian fungsi dari implementasi unit fungsi yang diuji dengan hasil

perancangan perangkat lunak yang telah dirancang sebelumnya. Berdasarkan hal tersebut, maka dapat diambil kesimpulan sesuai dengan tabel 6.2 bahwa unit fungsi yang diuji dari program sudah memenuhi kebutuhan fungsional yang telah dirancang pada tahap perancangan.

6.1.5.2 Analisis Pengujian Integrasi

Proses analisis terhadap hasil pengujian integrasi dilakukan dengan melihat kesesuaian beberapa unit fungsi yang menyusun satu blok fungsi dalam SIMS Jibas. Berdasarkan hal tersebut, maka dapat diambil kesimpulan sesuai dengan tabel 6.4 bahwa integrasi dari beberapa unit modul program yang diuji sudah memenuhi kebutuhan fungsional yang telah dirancang pada tahap perancangan.

6.1.5.3 Analisis pengujian validasi

Proses analisis terhadap hasil pengujian validasi dilakukan dengan melihat konformitas antara hasil kinerja sistem dengan daftar kebutuhan. Berdasarkan hasil pengujian validasi dapat disimpulkan bahwa implementasi dan fungsionalitas perangkat lunak Jibas modifikasi telah memenuhi kebutuhan yang telah dijabarkan pada tahap analisis kebutuhan.

6.1.5.4 Analisis Pengujian *Availability*

Proses analisis terhadap hasil pengujian *availability* dilakukan dengan melihat hasil kinerja sistem yang bekerja minimal 8 jam. Berdasarkan hasil pengujian *availability* pada tabel 6.13 dapat disimpulkan bahwa SIMS Jibas dapat berjalan secara normal selama 8 jam sesuai dengan kebutuhan non-fungsional.

6.1.5.5 Analisis Pengujian *compatibility*

Proses analisis terhadap hasil pengujian *compatibility* dilakukan dengan melihat hasil kinerja sistem yang dilakukan terhadap 2 sistem operasi windows dan sistem operasi linux. Berdasarkan hasil pengujian *compatibility* pada tabel 6.15 dan 6.16 dapat disimpulkan bahwa SIMS Jibas dapat berjalan secara normal pada 2 sistem operasi tersebut.

6.1.5.6 Analisis Pengujian *Usability*

Proses analisis terhadap hasil pengujian *usability* dilakukan dengan melihat hasil kuisisioner yang diberikan ke staff SMAN 4 Jember. Berdasarkan hasil pengujian *usability* pada tabel 6.18 dapat disimpulkan bahwa SIMS Jibas dapat dioperasionalkan di SMAN 4 Jember dengan memberikan pelatihan terlebih dahulu kepada semua pengguna, karena pengguna masih belum terbiasa menggunakan aplikasi.

6.1.5.7 Analisis Pengujian *Response Time*

Proses analisis terhadap hasil pengujian *response time* dilakukan dengan melihat *response* yang diberikan oleh sistem. Berdasarkan hasil pengujian *response time* pada tabel 6.19 dapat disimpulkan bahwa SIMS Jibas dapat memberikan *response time* rata-rata kurang dari 3 detik dari lima percobaan pada saat melakukan operasional sistem.

6.1.5.8 Analisis Pengujian *Control*

Proses analisis terhadap hasil pengujian *control* dilakukan dengan melihat pembatasan akses pengguna perangkat lunak Berdasarkan hasil pengujian *control* sesuai skenario tabel 6.23 dapat disimpulkan bahwa SIMS Jibas dapat memberikan batasan terhadap pengguna sesuai dengan kebutuhan non-fungsional.

BAB VII PENUTUP

7.1 Kesimpulan

1. Untuk menganalisis kebutuhan sistem, dilakukan survey langsung pada SMAN 4 Jember dalam menyusun administrasi dan manajemen yang ada pada saat itu (as-is). Dari hasil analisis tersebut dibuat desain perancangan untuk proses kustomisasi SIMS Jibas
2. Implementasi SIMS Jibas dilakukan setelah mengalami kustomisasi SIMS Jibas sesuai dengan kebutuhan di SMAN 4 Jember.
3. Berdasarkan hasil pengujian unit, pengujian integrasi, dan pengujian validasi sistem telah memenuhi kebutuhan fungsional yang di spesifikasikan.
4. Berdasarkan pengujian *compatibility* sistem yang dibuat dapat berjalan pada 2 platform yaitu platform Windows dan platform Linux.
5. Berdasarkan hasil kuisiner didapatkan bahwa 85% pengguna berpendapat, bahwa sistem dapat membantu meringankan kinerja pengguna dalam proses administrasi dan manajemen sehari – hari dan 100% pengguna memerlukan pelatihan untuk pengoperasionalan sistem.

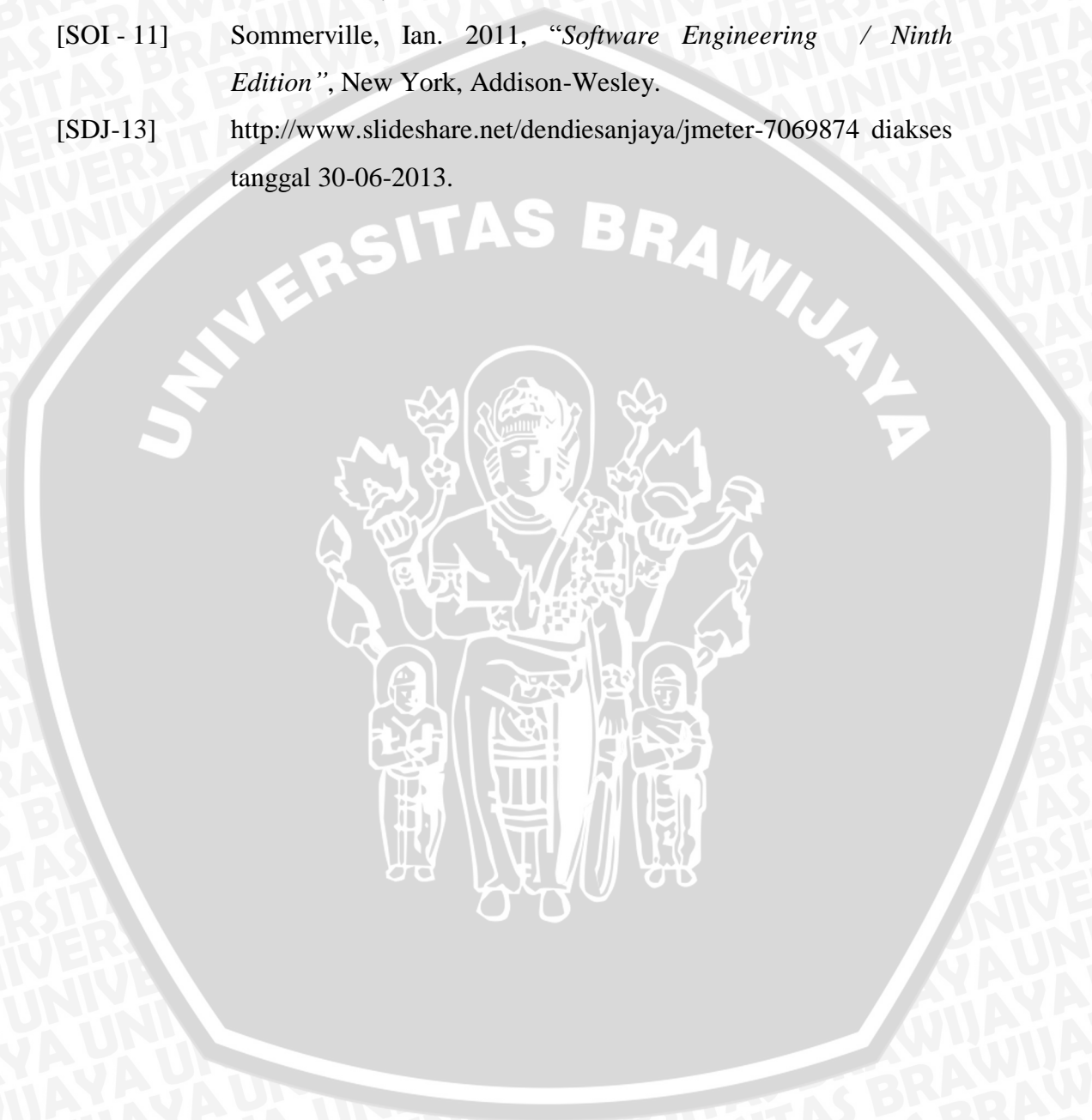
7.2 Saran

1. Untuk pengembang versi lebih lanjut sebaiknya source code yang digunakan berbasis OOP (Object Oriented Programming), karena OOP mempunyai nilai tambah dari pada prosedural untuk pengembangan software. namun dengan catatan tidak mengubah organisasi pendidikan yang terlanjur menggunakan SIMS Jibas.
2. Dalam pengembangan OOP disarankan dikembangkan juga kerangka kerja MVC (Model-View-Controller) untuk mempermudah arsitektur pengembangan dalam melakukan kustomisasi.
3. Untuk pengembangan versi lebih lanjut sebaiknya mengikutsertakan phpmyadmin pada saat instalasi program agar pengembang dapat memahami lebih dalam tentang basis data SISM Jibas atau memberikan keterangan tentang phpmyadmin di dokumentasi teknis.

DAFTAR PUSTAKA

- [AGS-09] Aarthi, Garde, Sharma. 2009, “*Educational Resource Planning – A Framework for Educational Institutions*”, Second International Conference on Emerging Trends in Engineering and Technology, ICETET.
- [DEO-00] O’Leary, Daniel, E.2000, “*Enterprise Resource Planning Systems, LifrCycle, Electronic Commerce, and Risk*”, Cambridge University.
- [ETE-13] [http://aryawiguna.wordpress.com/No. 4. ENTERPRISE RESOURCE PLANNING \(ERP\) DAN IMPLEMENTASI SISTEM INFORMASI BERBASIS ERP _ Sasmoyo.html](http://aryawiguna.wordpress.com/No.4. ENTERPRISE RESOURCE PLANNING (ERP) DAN IMPLEMENTASI SISTEM INFORMASI BERBASIS ERP _ Sasmoyo.html) diakses pada tanggal 6 maret 2013
- [JBS-13] <http://www.jibas.net/content/index/index.php> diakses tanggal 03 Februari 2013
- [IBM-01] Rational University, 2001, ” Business Modeling with UML Student Manual’, Rational Software Corporation.
- [MJO-13] Josua M. Sinambela, M. Eng, 2013, ”*Integrasi Sistem Informasi*”.
- [MEI-13] <http://meilidasari.blogspot.com/2012/04/apa-itu-use-case-activity-diagram-dan.html> diakses tanggal 15 april 2013
- [NUR-13] <http://www.nurahratu.comtutorialweb-design1-latest-news203-php-adalah.html> diakses pada tanggal 7-3-2013
- [PRE -10] Pressman, Roger. 2010. *Software Engineering: A Practioner’s Approach, 7th Edition*. Mc Graw-Hill.
- [RES - 12] Lianqiu Zhou, "Research on the Integration Application of Business Intelligence and ERP," icmecg, pp.269-271, 2012 International Conference on Management of e-Commerce and e-Government, 2012
- [ROS-11] A,S,Rosa.2011, “*Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*”,Modula,Bandung.

- [RSE-12] Xugang Tan, Xiaojuan Sun. 2012 "Research on Modeling of Earned Value Analysis Based on ERP System," cmcsm, pp.392-395, 2012 International Conference on Computing, Measurement, Control and Sensor Network.
- [SOI - 11] Sommerville, Ian. 2011, "*Software Engineering / Ninth Edition*", New York, Addison-Wesley.
- [SDJ-13] <http://www.slideshare.net/dendiesanjaya/jmeter-7069874> diakses tanggal 30-06-2013.



Lampiran A

Analisis Kebutuhan (*analysis requirement*)

SMAN 4 Jember

No	Bagian	Fitur	Kebutuhan	TTD
1.	HRD	<ul style="list-style-type: none"> • Perekrutan • Absensi Pegawai (Akademik) • Penggajian • Laporan per bulan (absensi jumlah pegawai) 	Data Pegawai, Tingkat Gaji, Mutasi Pegawai, Kenaikan Pangkat, Cara Absensi	
2.	KESISWAAN	<ul style="list-style-type: none"> • Siswa • Buku Induk Siswa • Absensi Siswa • Lulusan • Mutasi Siswa <ul style="list-style-type: none"> • Mutasi Masuk • Mutasi Keluar • Laporan (Buku Induk Siswa) 	Data Siswa, Tingkat, Jurusan, Absensi Siswa, Lulusan Siswa, Laporan	
3.	Akademik	<ul style="list-style-type: none"> • Penjadwalan • Raport • Laporan per semester (Legger) 	Data Pelajaran, Data Jadwal, Ruangan, Raport, Kalender akademik	
4.	Perpustakaan	<ul style="list-style-type: none"> • Anggota (Pegawai+siswa) • Input Referensi • Peminjaman • Pengembalian • Laporan (Jumlah Buku Sesuai Kategori, transaksi, pengunjung) 	Data Referensi, Data Anggota, Peminjaman, Pengembalian(Denda), Laporan	

Lampiran B

KUISIONER APLIKASI JIBAS MODIFIKASI

Nama :

Jabatan :

Berilah jawaban pada pertanyaan (yang paling sesuai dengan kondisi Anda).

1. Bagaimanakah menurut Anda mengenai tampilan aplikasi yang dibuat?

- | | |
|---------------------|---------------------|
| a. Sangat memuaskan | d. Kurang memuaskan |
| b. Memuaskan | e. Tidak memuaskan |
| c. Cukup | |

Berikan alasan :

2. Bagaimanakah menurut Anda mengenai menu – menu yang disediakan oleh aplikasi?

- | | |
|---------------------|---------------------|
| a. Sangat memuaskan | d. Kurang memuaskan |
| b. Memuaskan | e. Tidak memuaskan |
| c. Cukup | |

Berikan alasan :

3. Apakah aplikasi yang dikembangkan turut membantu dalam mempermudah kinerja pengguna?

- | | |
|-----------------------|-----------------------|
| a. Sangat mempermudah | d. Kurang mempermudah |
| b. Mempermudah | e. Tidak mempermudah |
| c. Cukup | |

Berikan alasan :

4. Apakah Anda setuju kalau dilakukan pelatihan untuk operasional sistem?

- | |
|------------------|
| a. Sangat Setuju |
| b. Tidak Setuju |

Berikan alasan :

5. Kalau Anda Setuju, berapa lama pelatihan untuk operasional sistem ?

- | |
|-------------|
| a. 1 minggu |
| b. 2 minggu |

c. 3 minggu

d. 4 minggu

6. Bagaimana pendapat anda jika aplikasi ini digunakan dan terus dikembangkan hingga menjadi aplikasi dalam sekolah yang lebih kompleks?

a. Sangat sesuai

d. Kurang sesuai

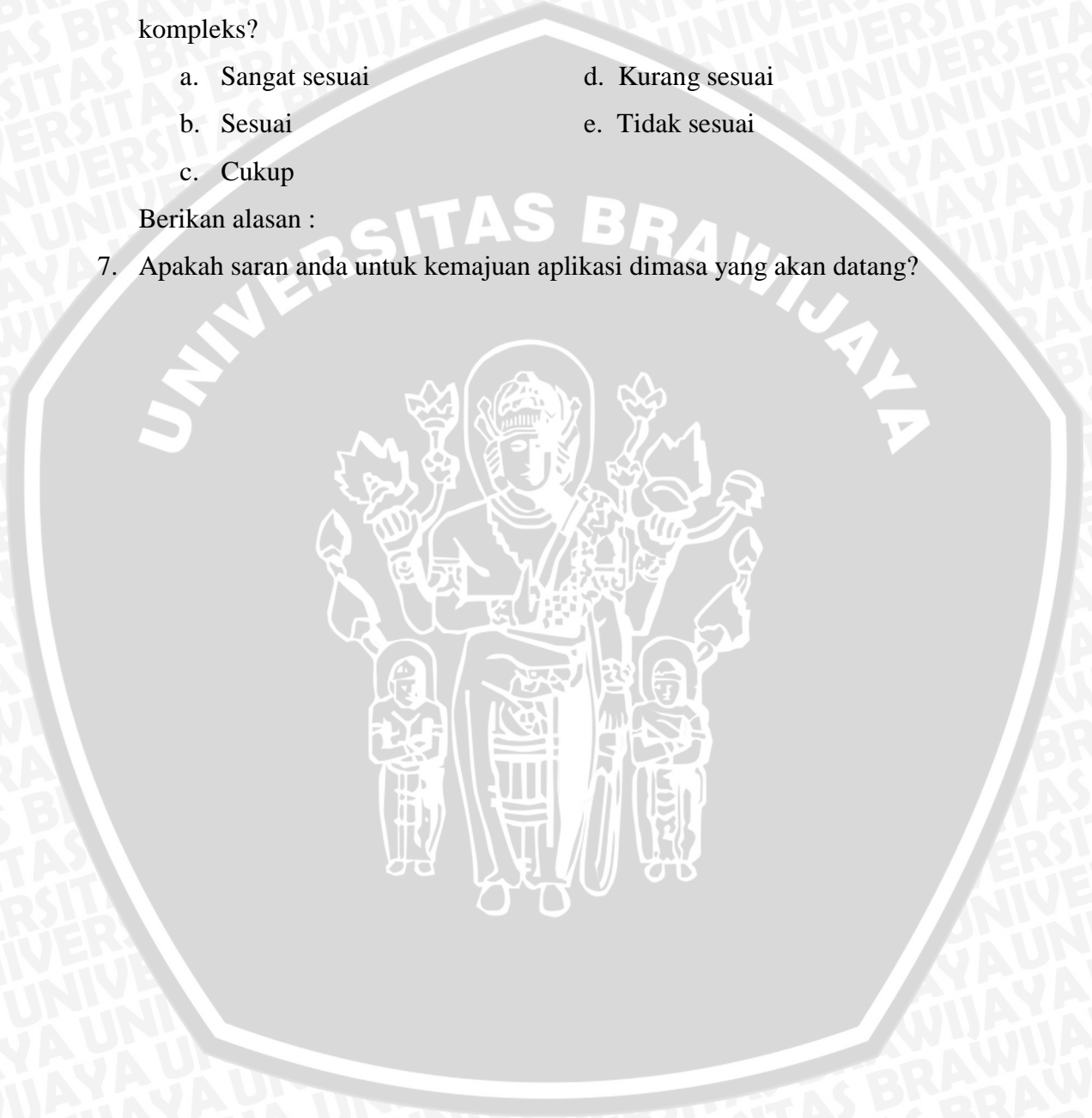
b. Sesuai

e. Tidak sesuai

c. Cukup

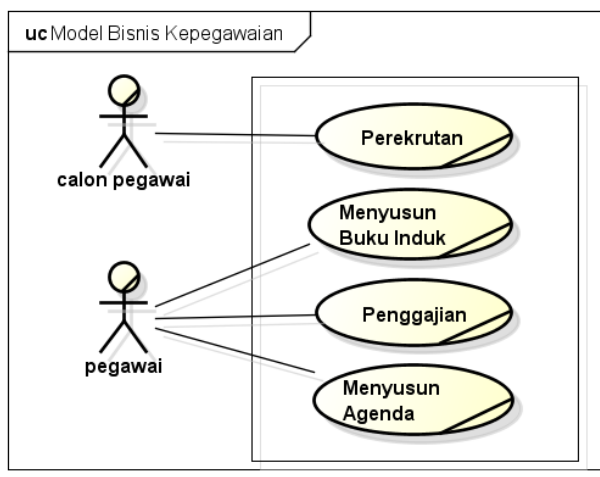
Berikan alasan :

7. Apakah saran anda untuk kemajuan aplikasi dimasa yang akan datang?

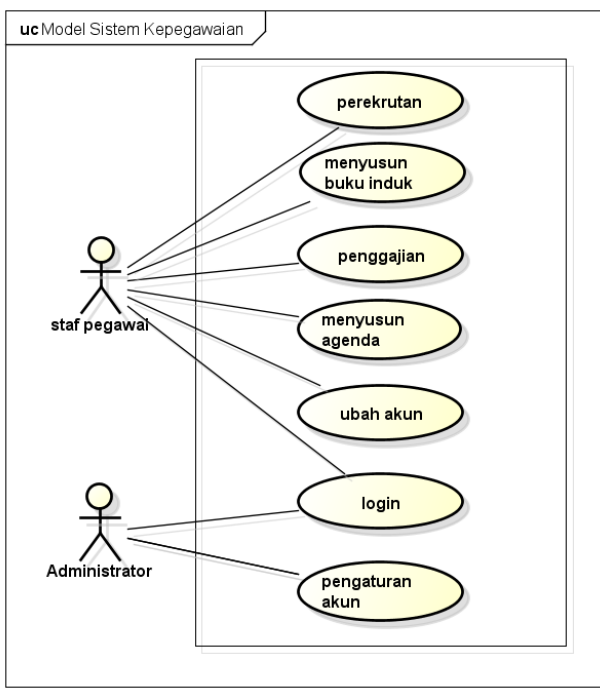


Lampiran C

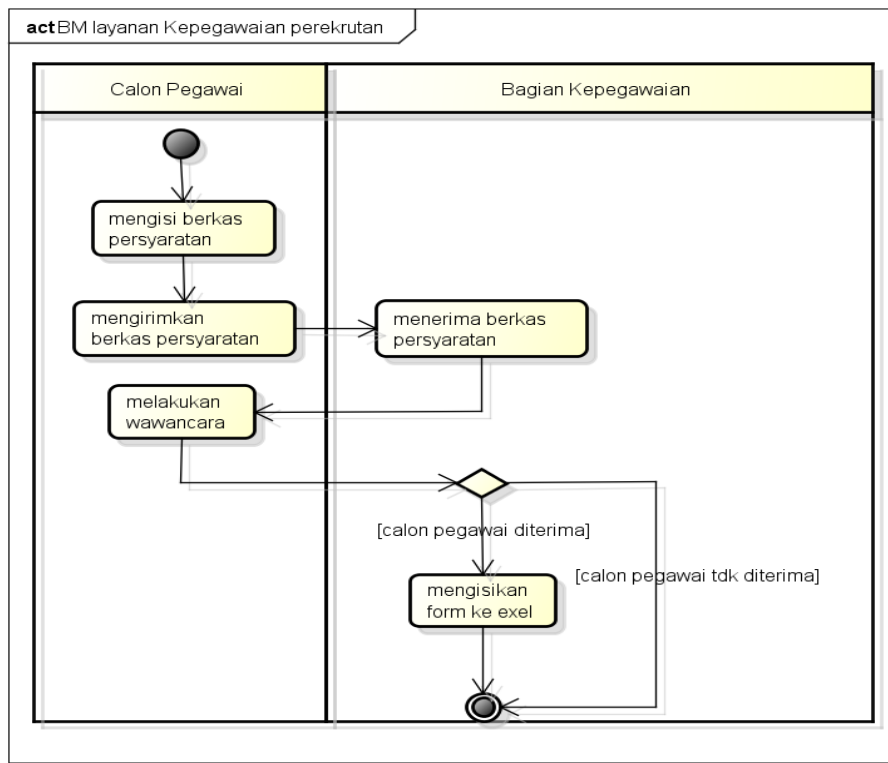
Pemodelan bisnis kepegawaian (as-is)



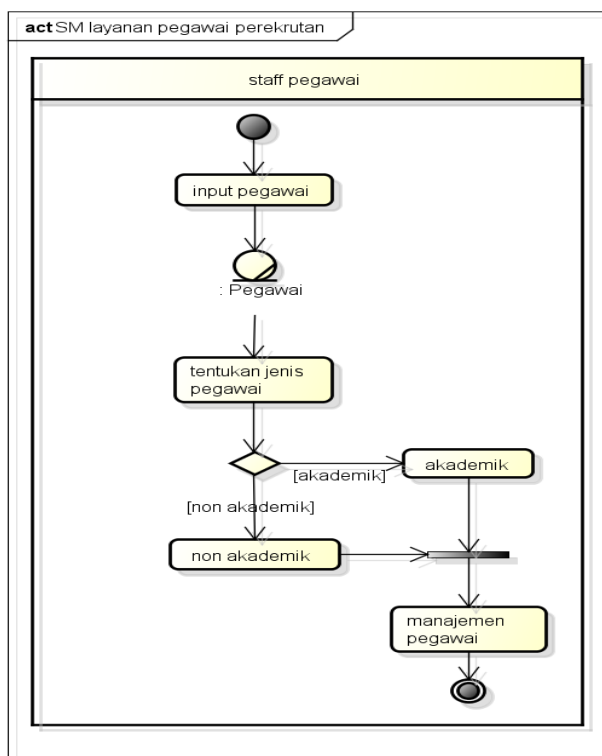
Pemodelan bisnis kepegawaian (to-be)



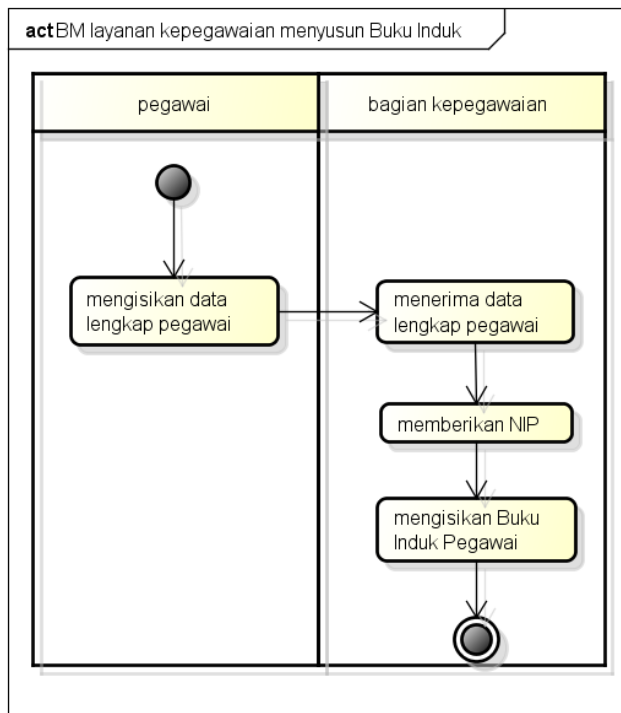
Aktivitas bisnis layanan perekrutan pegawai (as-is)



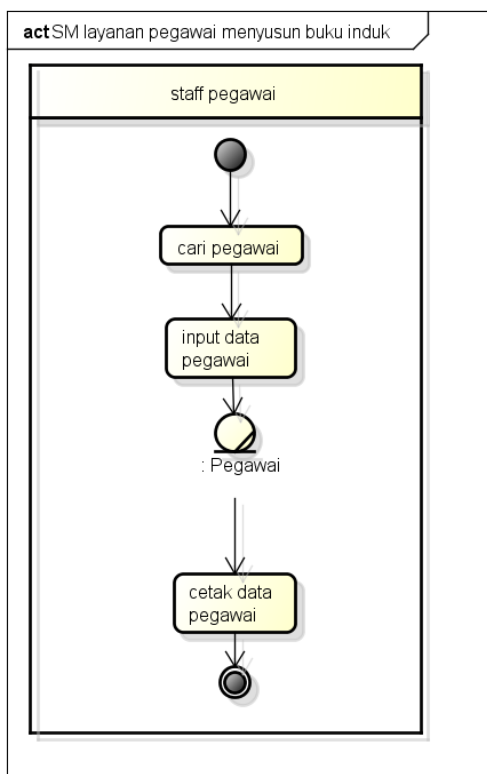
Aktivitas sistem layanan perekrutan pegawai (to-be)



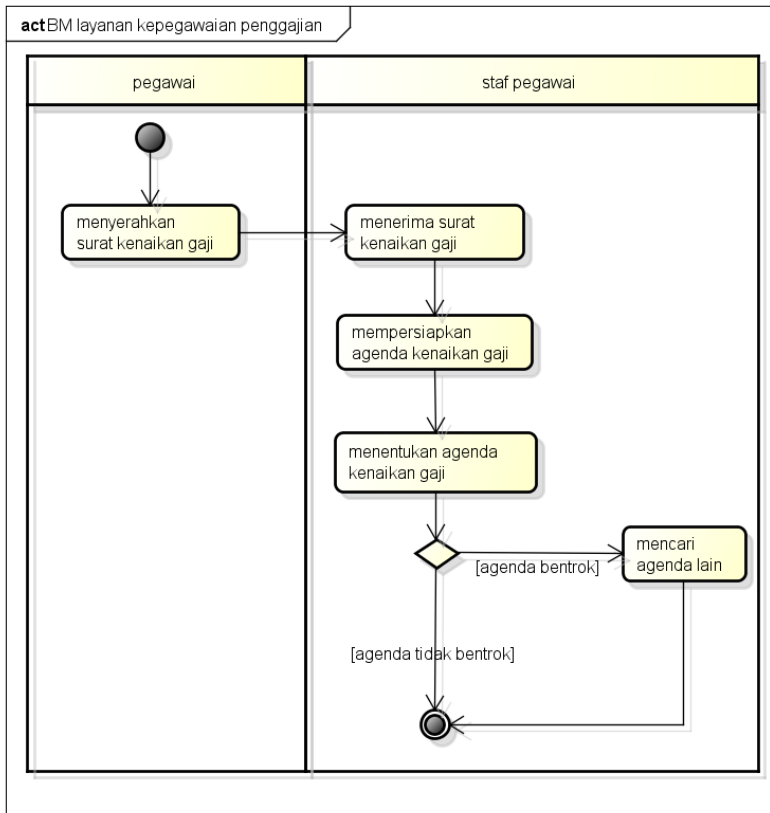
Aktivitas bisnis layanan kepegawaian menyusun buku induk (as-is)



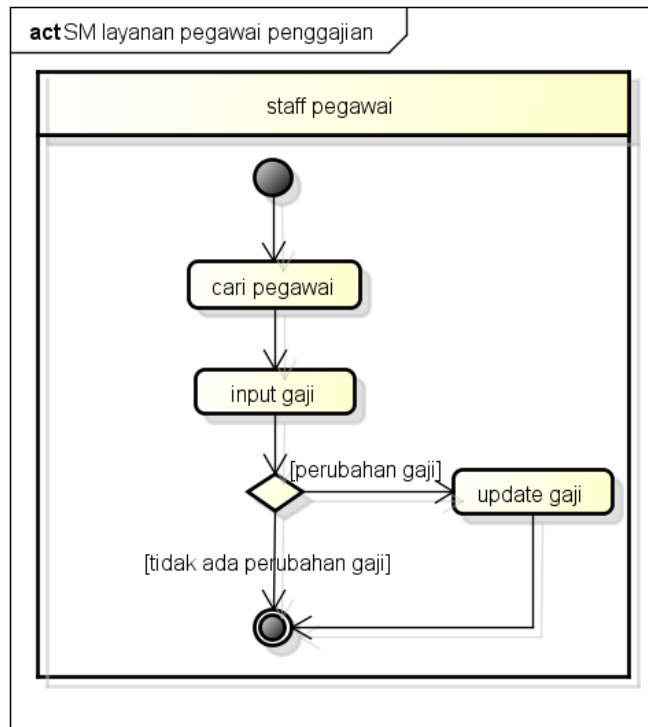
Aktivitas sistem layanan kepegawaian menyusun buku induk (to-be)



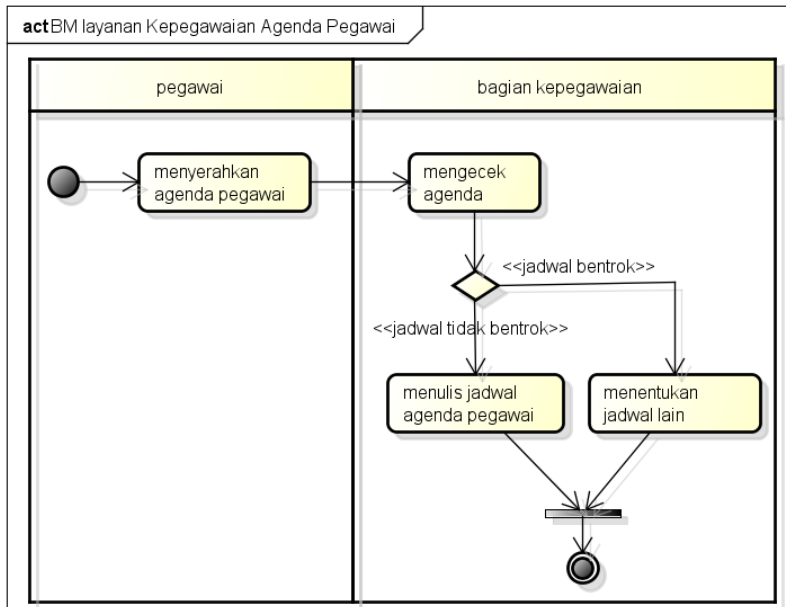
Aktivitas Bisnis Penggajian pegawai (as-is)



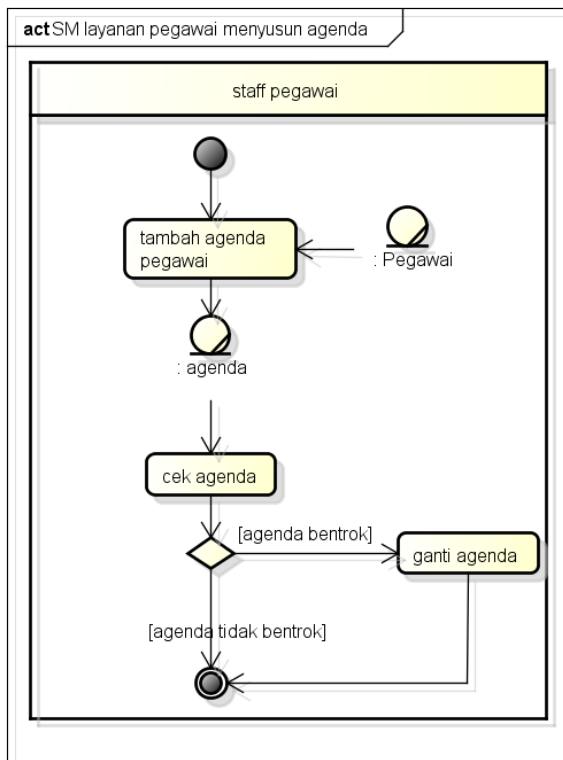
Aktivitas Sistem penggajian pegawai (to-be)



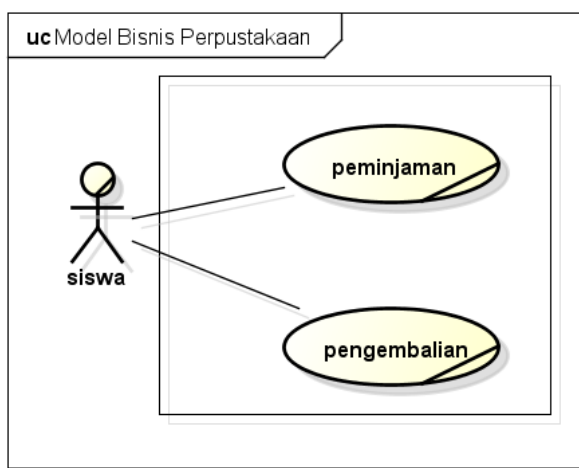
Aktivitas bisnis kepegawaian agenda pegawai (as-is)



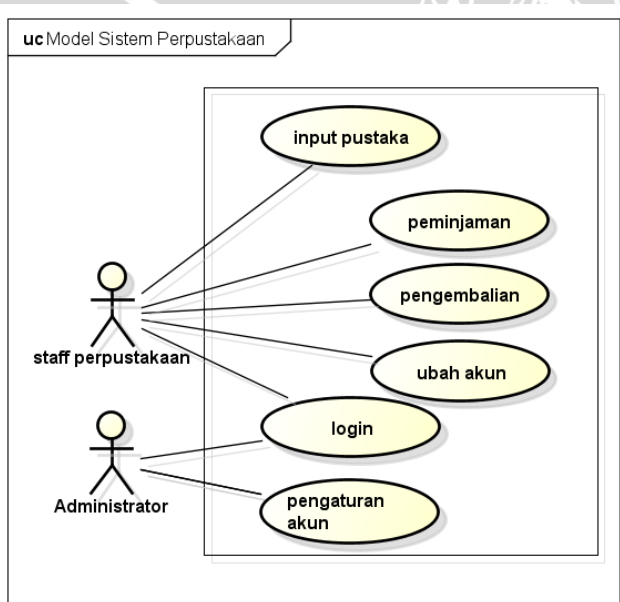
Aktivitas sistem kepegawaian agenda pegawai (to-be)



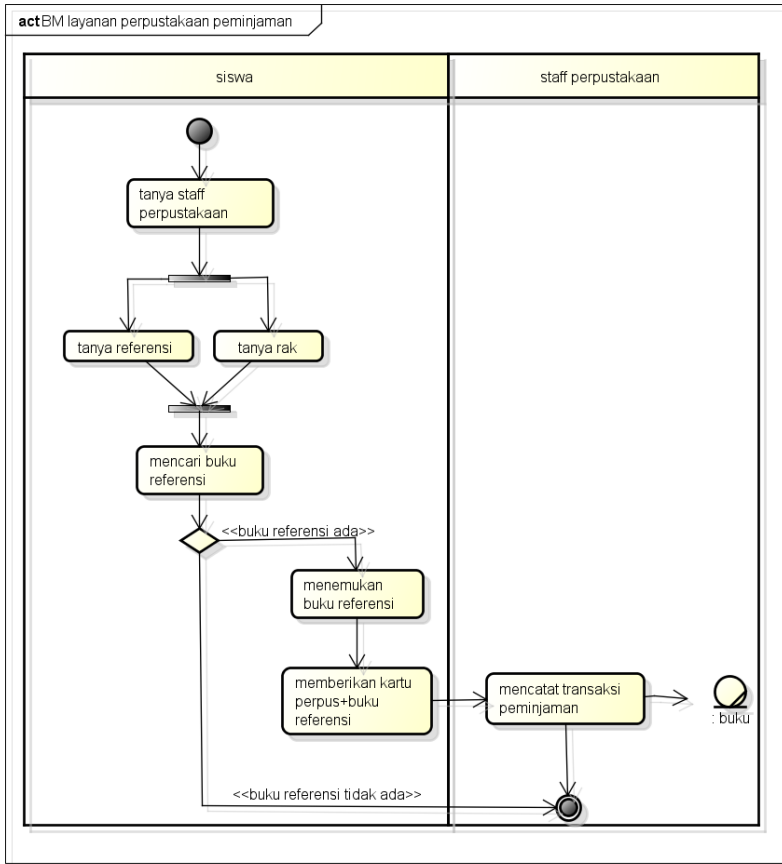
Pemodelan bisnis perpustakaan (as-is)



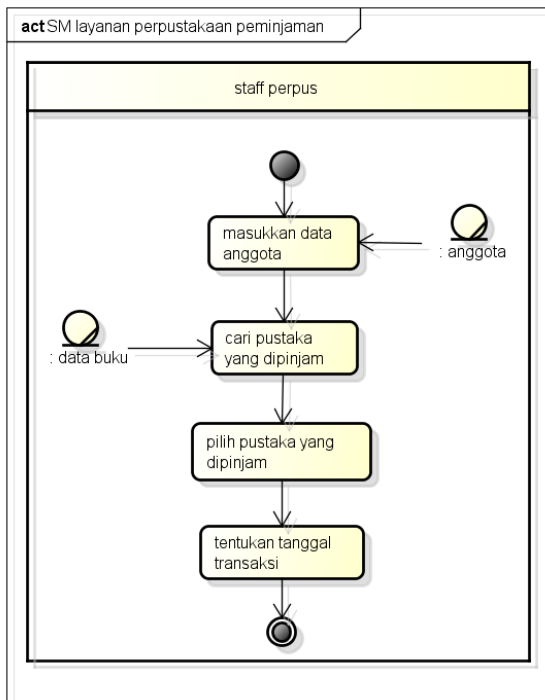
Pemodelan sistem perpustakaan (to-be)



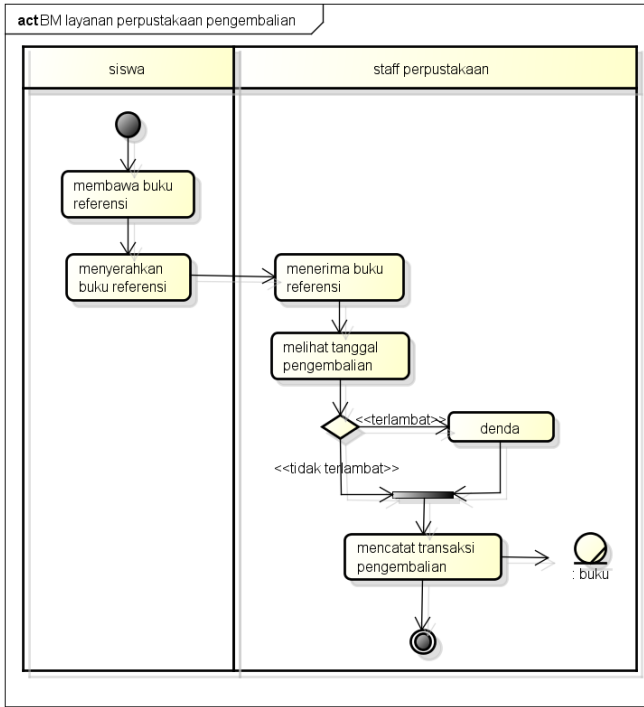
Aktivitas bisnis layanan perpustakaan peminjaman (as-is)



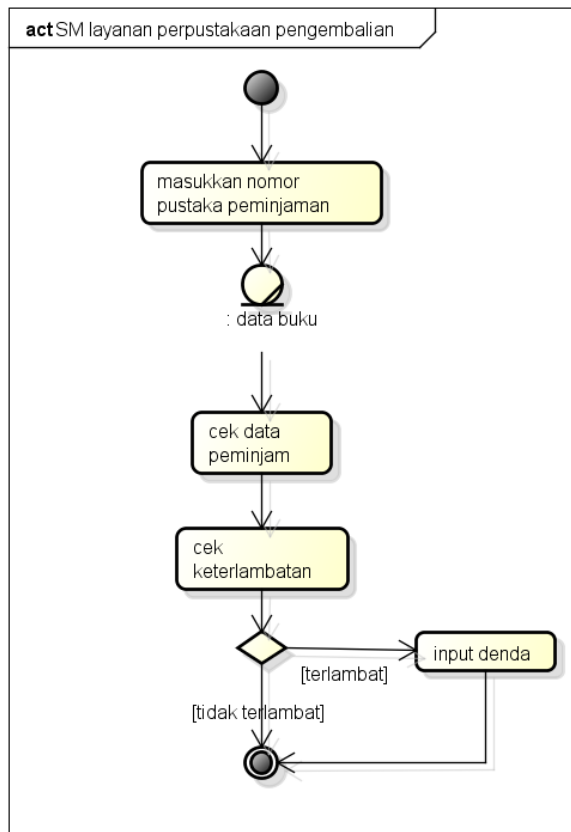
Aktivitas sistem layanan perpustakaan peminjaman (to-be)



Aktivitas Bisnis Model Pengembalian (as-is)



Aktivitas Sistem Model Pengembalian (to-be)



Aktivitas Sistem Input Pustaka (to-be)

