

**PENGEMBANGAN APLIKASI WEB TERINTEGRASI PADA
PERPUSTAKAAN UNIVERSITAS BRAWIJAYA DENGAN
MODEL *PROTOTYPING***

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh :

FAISAL PRAMPUDINANTAKA

NIM : 115090600111037



PROGRAM STUDI INFORMATIKA / ILMU KOMPUTER

FAKULTAS ILMU KOMPUTER

UNIVERSITAS BRAWIJAYA

MALANG

2015

PENGESAHAN

PENGEMBANGAN APLIKASI WEB TERINTEGRASI PADA PERPUSTAKAAN
UNIVERSITAS BRAWIJAYA DENGAN MODEL *PROTOTYPING*

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
FAISAL PRAMPUDINANTAKA
NIM : 115090600111037

Skripsi ini telah diuji dan dinyatakan lulus pada
31 Desember 2015

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Fajar Pradana, S.ST., M.Eng
NIP : 198711212015041004

Ismiarta Aknuranda, ST., M.Sc., Ph.D
NIK : 74071906110079

Mengetahui
Ketua Program Studi Informatika

Drs. Marji, M.T
NIP: 196708011992031001

PERNYATAAN ORISINALITAS

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

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

Malang, 7 Desember 2015

Faisal Prampudinantaka

NIM: 115090600111037



KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadirat Allah SWT, karena dengan rahmat, karunia dan hidayah-Nya penulis dapat menyelesaikan skripsi dengan judul **“Pengembangan Web Aplikasi Terintegrasi pada Perpustakaan Universitas Brawijaya dengan Model Prototyping”**. Shalawat serta salam atas junjungan besar Nabi kita Muhammad S.A.W. beserta keluarga dan para sahabat sekalian.

Skripsi ini disusun sebagai salah satu persyaratan untuk memperoleh gelar Sarjana Komputer di Program Studi Informatika /Ilmu Komputer, Fakultas Ilmu Komputer Universitas Brawijaya. Melalui pengantar ini penulis mengucapkan banyak terima kasih karena dalam penulisan dan penyusunan laporan skripsi ini penulis mendapat bantuan dan dorongan baik secara langsung maupun tidak langsung dari berbagai pihak diantaranya:

1. Bapak Drs. Marji, M.T. selaku ketua Program Studi Informatika/Ilmu Komputer Fakultas Ilmu Komputer (FILKOM) Universitas Brawijaya.
2. Bapak Fajar Pradana, S.ST.,M.Eng., selaku dosen pembimbing I dan Bapak Ismiarta Aknuranda, S.T., M.Sc., Ph.D., selaku dosen pembimbing II yang telah sabar memberikan bimbingan, arahan, ilmu dan saran untuk laporan skripsi ini.
3. Ibunda Cicik Sri Ekoyekti, Ayahanda Puji Pritianto dan istri tercinta Anggun Dafilla Widyasari atas segala dukungan, kasih sayang, perhatian dan kesabarannya, serta yang senantiasa selalu memberikan doa dan semangat demi terselesaikannya skripsi ini.
4. Bapak dan Ibu dosen serta seluruh staf Fakultas Ilmu Komputer (FILKOM) Universitas Brawijaya yang telah memberikan ilmunya serta arahan selama masa perkuliahan.
5. Bapak Ibu Staf dan teman-teman *student employee* Perpustakaan Universitas Brawijaya yang selalu memberikan semangat dan bantuan dalam menyelesaikan skripsi ini.

6. Teman-teman seperjuangan ILKOM angkatan 2011 yang tidak bisa disebutkan satu persatu yang selalu mendukung penulis.
7. Seluruh Keluarga Besar Fakultas Ilmu Komputer yang telah memberikan tempat untuk kuliah dan mencari pengalaman di Fakultas Ilmu Komputer Universitas Brawijaya.
8. Semua pihak yang tidak dapat penulis sebutkan satu per satu yang terlibat baik secara langsung maupun tidak langsung demi terselesaikannya skripsi ini.

Semoga jasa dan amal baik mendapatkan balasan dari Allah SWT. Penulis menyadari bahwa skripsi ini masih jauh dari sempurna dan tidak terlepas dari kekurangan dan kesalahan karena keterbatasan materi dan pengetahuan yang dimiliki penulis. Oleh karena itu, segala kritik dan saran yang bersifat membangun penulis harapkan untuk penyempurnaan skripsi ini. Penulis juga berharap mudah-mudahan skripsi ini dapat bermanfaat bagi semua pihak.

Malang, 7 Desember 2015

Faisal Prampudinantaka

Faisal.Prampudinantaka@gmail.com



ABSTRAK

Perpustakaan Universitas Brawijaya merupakan salah satu program cakupan universitas yang berfungsi mendukung program akademik universitas yang tertuang dalam "Tri Darma Perguruan Tinggi" mencakup pendidikan, penelitian dan pengabdian kepada masyarakat. Perpustakaan Universitas Brawijaya memiliki sistem informasi manajemen perpustakaan, namun sistem informasi tersebut belum mampu mengintegrasikan data-data pustaka maupun karya ilmiah yang ada di setiap ruang baca dengan perpustakaan pusat. Akibat data yang tersebar, proses pencarian suatu bahan pustaka pada OPAC (*Online Public Access Catalog*) hanya mencakup perpustakaan pusat saja. Oleh karena itu untuk mengatasi permasalahan diatas diperlukan suatu sistem yang dapat mengintegrasikan data bahan pustaka yang ada di setiap ruang baca. Disisi lain Perpustakaan Universitas Brawijaya juga menyediakan fasilitas untuk pengajuan bahan pustaka baru oleh mahasiswa. Namun proses yang dilakukan masih manual, yaitu dengan cara menuliskan permintaan bahan pustaka baru pada selembur form. Hal ini membuat prosesnya masih memerlukan waktu yang cukup lama. Berdasarkan permasalahan tersebut, diperlukan sistem yang lebih efisien waktu agar proses bisnis yang ada di perpustakaan lebih cepat dari pada sebelumnya. Sistem dibangun menggunakan integrasi basis data dengan *web service*, sehingga mahasiswa dapat langsung masuk kedalam sistem dengan menggunakan akun SIAKAD UB tanpa harus registrasi terlebih dahulu.

Daftar kebutuhan perangkat lunak yang akan dibangun belum spesifik, sehingga pembuatan sistem ini menggunakan model *evolutionary prototyping* dengan tiga kali iterasi. Pada setiap iterasi yang dilakukan akan mengalami perubahan atau penambahan kebutuhan fungsional sesuai dengan kebutuhan pengguna. Hasil akhir dari iterasi *prototyping* yang dilakukan yaitu menghasilkan sistem versi 3.0 atau versi rilis. Pengujian perangkat lunak dilakukan pada sistem yang dikembangkan. Pengujian dilakukan menggunakan metode *black box* untuk pengujian validasi dan metode *white box* untuk pengujian unit. Selain itu juga dilakukan pengujian performansi dan pengujian pada *web service*. Berdasarkan hasil pengujian pada sistem yang telah dilakukan diperoleh kesimpulan bahwa kompleksitas dari sistem ini tergolong rendah sehingga kerumitan perbaikan *coding* rendah. Secara fungsionalitas dari sistem ini telah memenuhi kebutuhan dan secara keseluruhan berjalan dengan baik. Hasil dari pengujian performansi dapat disimpulkan bahwa rata-rata dari 10 kali pengujian waktu yang diperlukan untuk proses eksekusi *web service* selama 7,07 detik. Hasil pengujian *web service*, sistem telah dapat mengembalikan nilai sesuai dengan parameter yang diminta.

Kata Kunci : Perpustakaan, Aplikasi web, *web service*, *prototyping*, *evolutionary prototyping*

ABSTRACT

The Library of Brawijaya University is one university programs that support the academic of the university like in "Tri Dharma Perguruan Tinggi" includes education, research and community service. The Library of Brawijaya University has a library management information system, but the system has not been able to integrate information data library and scientific works available in each center with a library reading room. Due to scattered data, the process of finding a substance library on the OPAC (Online Public Access Catalog) only covers the central library only. Therefore, to overcome the thus problems, it need a system that can integrate data library materials available in each library. On the other hand The Library of Brawijaya University also provides the submission of new library materials by students. However, the process is still done manually, by writing a new demand for library materials in sheet form. This makes the process requires quite a long time. Based on these problems, we need a more efficient fine system, so that the business processes that exist in the library is done more quickly than before. The system is built using database integration with web services, so that students can directly enter the system by using an account of SIAKAD UB without prior registration.

List of software requirements to be developed has not been specific, thus making this system uses evolutionary prototyping model using three times the iterations. Every iteration will undergo changes or adding the functions that are needed according to user demand. The final result of prototyping iterations that are performed is system version 3.0 or release version. Software testing is done on the system that has been developed. Testing was done using black box method for validation testing and white box method for unit testing. The performance testing was also conducted with testing web service. Based on the test results in a system that has been developed, the conclusion is that the complexity of the system is relatively low, it means that the complexity of the maintenance process is low. According to the functionality, the system has met the needs and overall running well. The results of performance testing can be concluded that an average of 10 times the time required for the execution of a web service for 7.07 seconds. The results of testing for web service, the system has been able to reverse the value in accordance with the parameters requested.

Keywords : *Library, web application, web service, prototyping, evolutionary prototyping*

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT	vii
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR.....	xviii
DAFTAR LAMPIRAN	xx
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan	3
1.5 Manfaat	3
1.6 Sistematika Penulisan.....	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Pustaka.....	5
2.2 Dasar Teori.....	5
2.2.1 Pengertian Perpustakaan.....	5
2.2.2. Pengertian Rekayasa Perangkat Lunak	7
2.2.3 Pengertian Prototipe.....	7
2.2.4 Model <i>Prototyping</i>	8
2.2.5 Fase Evolutionary prototyping.....	10
2.2.6 <i>Web Service</i>	11
2.2.7 <i>Unified Modeling Language (UML)</i>	12
2.2.8 Pengertian Basis Data	14

2.2.9 Model Data Relasional	15
2.2.10 <i>Framework</i> YII	15
2.2.11 Konsep Pengujian Perangkat Lunak	17
BAB 3 METODOLOGI PENELITIAN	18
3.1 Studi Literatur	19
3.2 Analisis Kebutuhan	19
3.3 Perancangan Desain Sistem	20
3.4 Implementasi Sistem	20
3.5 Evaluasi Pengguna	20
3.6 Pengujian dan Analisis	20
3.7 Kesimpulan dan saran	21
BAB 4 ANALISIS DAN PERANCANGAN	22
4.1 Analisis Kebutuhan	22
4.1.1 Elisitasi Kebutuhan	23
4.1.2 Identifikasi Aktor	25
4.1.3 Spesifikasi Kebutuhan	26
4.1.4 Diagram <i>Use Case</i>	35
4.1.5 Skenario <i>Use Case</i>	37
4.2 Perancangan Perangkat Lunak	82
4.2.1 Perancangan Arsitektural	82
4.2.2 Perancangan Diagram <i>Sequence</i>	83
4.2.3 Perancangan Diagram <i>Class</i>	88
4.2.4 Model Data Relasional	96
4.2.5 Perancangan Prototipe	96
BAB 5 IMPLEMENTASI	105
5.1 Implementasi	105
5.1.1 Spesifikasi Sistem	105
5.1.2 Batasan Implementasi	106
5.1.3 Implementasi Kelas	107
5.1.4 Implementasi Algoritma	108
5.1.5 Implementasi Prototipe	109



5.2 Evaluasi	117
5.2.1 Evaluasi Prototipe Versi 0.0	117
5.2.2 Evaluasi Prototipe Versi 1.0	117
5.2.3 Evaluasi Prototipe Versi 2.0	118
BAB 6 PENGUJIAN DAN ANALISIS.....	119
6.1 Pengujian Unit	119
6.1.1 Pengujian Unit Proses <i>Login</i>	119
6.1.2 Pengujian Unit Proses Tambah Permintaan Bahan pustaka.....	122
6.1.3 Pengujian Unit Proses Cari Buku Label Merah.....	125
6.2 Pengujian Integrasi	129
6.2.1 Kasus Uji Pengujian Integrasi	129
6.3 Pengujian Validasi.....	132
6.3.1 Kasus Uji Pengujian Validasi.....	132
6.4 Pengujian Web Service	176
6.5 Pengujian Performansi	177
6.6 Analisis Hasil Pengujian	177
6.6.1 Analisis Hasil Pengujian Unit	177
6.6.2 Analisis Hasil Pengujian Integrasi.....	178
6.6.3 Analisis Hasil Pengujian Validasi	178
6.6.4 Analisis Hasil Pengujian <i>Web Service</i>	178
6.6.5 Analisis Hasil Pengujian Performansi	178
BAB 7 PENUTUP	179
7.1 Kesimpulan	179
7.2 Saran.....	179
DAFTAR PUSTAKA.....	180
LAMPIRAN A PERTANYAAN WAWANCARA.....	182
LAMPIRAN B PROSES ITERASI <i>PROTOTYPING</i>	183

DAFTAR TABEL

Tabel 2.1 <i>Use case</i> Diagram.....	12
Tabel 4.1 Kebutuhan perangkat lunak secara umum.....	22
Tabel 4.2 Identifikasi aktor iterasi ke-3.....	25
Tabel 4.3 Spesifikasi kebutuhan fungsional iterasi ke-3.....	26
Tabel 4.4 Spesifikasi kebutuhan non-fungsional.....	35
Tabel 4.5 Skenario <i>use case login</i>	37
Tabel 4.6 Skenario <i>use case</i> lihat detail biodata.....	38
Tabel 4.7 Skenario <i>use case</i> ubah biodata.....	38
Tabel 4.8 Skenario <i>use case</i> tambah staf.....	39
Tabel 4.9 Skenario <i>use case</i> ubah staf.....	40
Tabel 4.10 Skenario <i>use case</i> daftar staf.....	40
Tabel 4.11 Skenario <i>use case</i> tambah permintaan buku.....	41
Tabel 4.12 Skenario <i>use case</i> unduh <i>form</i> permintaan buku.....	42
Tabel 4.13 Skenario <i>use case</i> unggah <i>form</i> permintaan buku.....	42
Tabel 4.14 Skenario <i>use case</i> lihat daftar permintaan buku.....	43
Tabel 4.15 Skenario <i>use case</i> lihat daftar permintaan buku disetujui.....	44
Tabel 4.16 Skenario <i>use case</i> lihat daftar permintaan buku belum disetujui.....	45
Tabel 4.17 Skenario <i>use case</i> unduh daftar permintaan buku disetujui.....	45
Tabel 4.18 Skenario <i>use case</i> unduh daftar permintaan buku belum disetujui....	46
Tabel 4.19 Skenario <i>use case</i> lihat detail permintaan buku.....	47
Tabel 4.20 Skenario <i>use case</i> ubah permintaan buku.....	48
Tabel 4.21 Skenario <i>use case</i> hapus permintaan buku.....	48
Tabel 4.22 Skenario <i>use case</i> cari permintaan buku belum disetujui.....	49
Tabel 4.23 Skenario <i>use case</i> tambah buku dari permintaan.....	50
Tabel 4.24 Skenario <i>use case</i> tambah permintaan jurnal.....	50
Tabel 4.25 Skenario <i>use case</i> unduh form permintaan jurnal.....	51
Tabel 4.26 Skenario <i>use case</i> unggah form permintaan jurnal.....	52
Tabel 4.27 Skenario <i>use case</i> lihat daftar permintaan jurnal.....	53
Tabel 4.28 Skenario <i>use case</i> lihat daftar permintaan jurnal disetujui.....	54
Tabel 4.29 Skenario <i>use case</i> lihat daftar permintaan jurnal belum disetujui.....	54

Tabel 4.30 Skenario <i>use case</i> unduh daftar permintaan jurnal disetujui.....	55
Tabel 4.31 Skenario <i>use case</i> unduh daftar permintaan jurnal belum disetujui..	56
Tabel 4.32 Skenario <i>use case</i> lihat detail permintaan jurnal.....	57
Tabel 4.33 Skenario <i>use case</i> ubah permintaan jurnal.....	57
Tabel 4.34 Skenario <i>use case</i> hapus permintaan jurnal.....	58
Tabel 4.35 Skenario <i>use case</i> cari permintaan jurnal belum disetujui.....	59
Tabel 4.36 Skenario <i>use case</i> tambah jurnal dari permintaan.....	59
Tabel 4.37 Skenario <i>use case</i> tambah buku baru.....	60
Tabel 4.38 Skenario <i>use case</i> unduh <i>form</i> tambah buku.....	61
Tabel 4.39 Skenario <i>use case</i> unggah <i>form</i> tambah buku baru.....	61
Tabel 4.40 Skenario <i>use case</i> lihan daftar buku label merah.....	62
Tabel 4.41 Skenario <i>use case</i> cari buku label merah.....	63
Tabel 4.42 Skenario <i>use case</i> lihat detail buku label merah.....	63
Tabel 4.43 Skenario <i>use case</i> ubah buku label merah.....	64
Tabel 4.44 Skenario <i>use case</i> tambah jurnal baru.....	65
Tabel 4.45 Skenario <i>use case</i> unduh <i>form</i> tambah jurnal.....	66
Tabel 4.46 Skenario <i>use case</i> unggah <i>form</i> tambah jurnal.....	66
Tabel 4.47 Skenario <i>use case</i> lihan daftar jurnal label merah.....	67
Tabel 4.48 Skenario <i>use case</i> cari jurnal label merah.....	67
Tabel 4.49 Skenario <i>use case</i> lihat detail jurnal label merah.....	68
Tabel 4.50 Skenario <i>use case</i> ubah jurnal label merah.....	69
Tabel 4.51 Skenario <i>use case</i> tambah buku label putih.....	70
Tabel 4.52 Skenario <i>use case</i> lihan daftar buku label putih.....	70
Tabel 4.53 Skenario <i>use case</i> cari buku label putih.....	71
Tabel 4.54 Skenario <i>use case</i> ubah buku label putih.....	72
Tabel 4.55 Skenario <i>use case</i> lihat detail buku label putih.....	72
Tabel 4.56 Skenario <i>use case</i> tambah jurnal jurnal label putih.....	73
Tabel 4.57 Skenario <i>use case</i> lihan daftar jurnal label putih.....	73
Tabel 4.58 Skenario <i>use case</i> cari jurnal label putih.....	74
Tabel 4.59 Skenario <i>use case</i> ubah jurnal label putih.....	75
Tabel 4.60 Skenario <i>use case</i> lihat detail jurnal label putih.....	75
Tabel 4.61 Skenario <i>use case</i> cetak barcode dan label buku label merah.....	76



Tabel 4.62 Skenario <i>use case</i> cetak barcode dan label buku label putih.....	77
Tabel 4.63 Skenario <i>use case</i> cetak barcode dan label jurnal label merah.....	78
Tabel 4.64 Skenario <i>use case</i> cetak barcode dan label jurnal label putih.....	79
Tabel 4.65 Skenario <i>use case</i> buku.....	80
Tabel 4.66 Skenario <i>use case</i> jurnal.....	80
Tabel 4.67 Skenario <i>logout</i>	81
Tabel 4.68 Penambahan kelas dan method pada iterasi 1.....	88
Tabel 4.69 penambahan kelas dan method pada iterasi 3.....	89
Tabel 5.1 Spesifikasi Perangkat Keras.....	106
Tabel 5.2 Spesifikasi Perangkat Lunak.....	106
Tabel 5.3 Implementasi <i>Class Controller</i> pada kode program.....	107
Tabel 5.4 Implementasi <i>Class Model</i> pada kode program.....	107
Tabel 5.5 Implementasi algoritma <i>login</i>	108
Tabel 6.1 Pengujian unit <i>login</i>	120
Tabel 6.2 Kasus Uji Login.....	122
Tabel 6.3 Pengujian unit proses tambah permintaan bahan putaka.....	123
Tabel 6.4 Kasus uji tambah permintaan bahan putaka.....	125
Tabel 6.5 Pengujian unit proses cari buku label merah.....	125
Tabel 6.6 Kasus uji cari buku label merah.....	128
Tabel 6.1 Kasus Uji Integrasi <i>actionLogin</i> dan <i>cekDataLogin</i>	129
Tabel 6.8 Kasus Uji Integrasi <i>actionPermintaanIndividu</i> dan <i>inputBukuIndividu</i>	129
Tabel 6.9 Kasus Uji Integrasi <i>actionPermintaanIndividu</i> dan <i>inputJurnalIndividu</i>	130
Tabel 6.10 Kasus Uji Integrasi <i>actionSearchMerah</i> dan <i>cariJudul</i>	131
Tabel 6.11 Kasus uji Login.....	132
Tabel 6.12 Kasus uji login alternatif 1 :	
Jika <i>username</i> atau <i>password</i> kosong.....	133
Tabel 6.13 Kasus uji login alternatif 2 :	
Jika kombinasi <i>username</i> atau <i>password</i> salah.....	133
Tabel 6.14 Kasus uji lihat detail biodata.....	134
Tabel 6.15 Kasus uji ubah biodata.....	134
Tabel 6.16 Kasus uji ubah biodata alternatif 1 :	
Jika field yang wajib diisi kosong.....	134

Tabel 6.17 Kasus uji tambah staf.....	135
Tabel 6.18 Kasus uji tambah staf alternatif 1 :	
Jika field yang wajib diisi kosong.....	135
Tabel 6.19 Kasus uji ubah staf.....	136
Tabel 6.20 Kasus uji ubah staf alternatif 1 :	
Jika field yang wajib diisi kosong.....	136
Tabel 6.21 Kasus uji daftar staf.....	137
Tabel 6.22 Kasus uji tambah permintaan buku.....	137
Tabel 6.23 Kasus uji tambah permintaan buku alternatif 1 :	
Jika field yang wajib diisi kosong.....	137
Tabel 6.24 Kasus uji unduh form permintaan buku.....	138
Tabel 6.25 Kasus uji unggah form permintaan buku.....	138
Tabel 6.26 Kasus uji unggah form permintaan buku alternatif 1 :	
Jika yang diunggah tidak sesuai.....	139
Tabel 6.27 Kasus uji lihat daftar permintaan buku.....	140
Tabel 6.28 Kasus uji lihat daftar permintaan buku disetujui.....	140
Tabel 6.29 Kasus uji lihat daftar permintaan buku belum disetujui.....	141
Tabel 6.30 Kasus uji unduh daftar permintaan buku disetujui.....	141
Tabel 6.31 Kasus uji unduh daftar permintaan buku belum disetujui.....	142
Tabel 6.32 Kasus uji lihat detail permintaan buku.....	142
Tabel 6.33 Kasus uji ubah permintaan buku.....	143
Tabel 6.34 Kasus uji ubah permintaan buku alternatif 1 :	
Jika field yang wajib diisi kosong.....	143
Tabel 6.35 Kasus uji hapus permintaan buku.....	144
Tabel 6.36 Kasus uji cari permintaan buku belum disetujui.....	144
Tabel 6.37 Kasus uji tambah buku dari permintaan.....	145
Tabel 6.38 Kasus uji tambah permintaan jurnal.....	145
Tabel 6.39 Kasus uji tambah permintaan jurnal alternatif 1 :	
Jika field yang wajib diisi kosong.....	146
Tabel 6.40 Kasus uji unduh form permintaan jurnal.....	146
Tabel 6.41 Kasus uji unggah form permintaan jurnal.....	147
Tabel 6.42 Kasus uji unggah form permintaan jurnal alternatif 1 :	



Jika file yang diunggah tidak sesuai.....	147
Tabel 6.43 Kasus uji lihat daftar permintaan jurnal.....	148
Tabel 6.44 Kasus uji lihat daftar permintaan jurnal disetujui.....	148
Tabel 6.45 Kasus uji lihat daftar permintaan jurnal belum disetujui.....	149
Tabel 6.46 Kasus uji unduh daftar permintaan jurnal disetujui.....	149
Tabel 6.47 Kasus uji unduh daftar permintaan jurnal belum disetujui.....	150
Tabel 6.48 Kasus uji lihat detail permintaan jurnal.....	150
Tabel 6.49 Kasus uji ubah permintaan jurnal.....	151
Tabel 6.50 Kasus uji ubah permintaan jurnal alternatif 1 :	
Jika <i>field</i> yang wajib diisi kosong.....	151
Tabel 6.51 Kasus uji hapus permintaan jurnal.....	152
Tabel 6.52 Kasus uji cari permintaan jurnal belum disetujui.....	152
Tabel 6.53 Kasus uji tambah jurnal dari permintaan.....	153
Tabel 6.54 Kasus uji tambah buku baru.....	153
Tabel 6.55 Kasus uji tambah buku baru alternatif 1 :	
Jika <i>field field</i> yang wajib diisi kosong.....	154
Tabel 6.56 Kasus uji tambah buku baru alternatif 2 :	
Jika karakter yang dimasukkan tidak sesuai.....	154
Tabel 6.57 Kasus uji unduh form tambah buku.....	155
Tabel 6.58 Kasus uji unggah form tambah buku baru.....	155
Tabel 6.59 Kasus uji unggah form tambah buku baru alternatif 1:	
jika file yang diunggah tidak sesuai dengan jenis bahan pustaka.....	155
Tabel 6.60 Kasus uji lihat daftar buku label merah.....	156
Tabel 6.61 Kasus uji cari buku label merah.....	156
Tabel 6.62 Kasus uji cari buku label merah alternatif 1 :	
Jika tidak memilih jenis pencarian.....	157
Tabel 6.63 Kasus uji lihat detail buku label merah.....	157
Tabel 6.64 Kasus uji ubah buku label merah.....	157
Tabel 6.65 Kasus uji ubah buku label merah alternatif 1 :	
Jika <i>field</i> yang wajib diisi kosong.....	158
Tabel 6.66 Kasus uji tambah jurnal baru.....	156
Tabel 6.67 Kasus uji tambah jurnal baru alternatif 1 :	



Jika <i>field</i> yang wajib diisi kosong.....	159
Tabel 6.68 Kasus uji tambah jurnal baru alternatif 2 :	
Jika karakter yang diinputkan tidak sesuai.....	159
Tabel 6.69 Kasus uji unduh form tambah jurnal.....	160
Tabel 6.70 Kasus uji unggah form tambah jurnal.....	160
Tabel 6.71 Kasus uji unggah form tambah jurnal alternatif 1 :	
Jika file excel yang di unggah tidak sesuai dengan jenis bahan pustaka.....	161
Tabel 6.72 Kasus uji lihat darfat jurnal label merah.....	161
Tabel 6.73 Kasus uji cari jurnal label merah.....	161
Tabel 6.74 Kasus uji cari jurnal label merah alternatif 1 :	
Jika <i>field</i> pencarian kosong.....	162
Tabel 6.75 Kasus uji lihat detail jurnal label merah.....	162
Tabel 6.76 Kasus uji ubah jurnal label merah.....	163
Tabel 6.77 Kasus uji ubah jurnal label merah alternatif 1 :	
Jika <i>field</i> yang wajib diisi kosong.....	163
Tabel 6.78 Kasus uji tambah buku label putih.....	164
Tabel 6.79 Kasus uji lihat daftar buku label putih.....	164
Tabel 6.80 Kasus uji cari buku label putih.....	165
Tabel 6.81 Kasus uji cari buku label putih alternatif 1 :	
Jika tidak memilih jenis pencarian.....	165
Tabel 6.82 Kasus uji ubah buku label putih.....	166
Tabel 6.83 Kasus uji lihat detail buku buku label putih.....	166
Tabel 6.84 Kasus uji tambah jurnal label putih.....	167
Tabel 6.85 Kasus uji lihat daftar jurnal label putih.....	167
Tabel 6.86 Kasus uji cari jurnal label putih.....	167
Tabel 6.87 Kasus uji cari jurnal label putih alternatif 1 :	
Jika tidak memilih jenis pencarian.....	168
Tabel 6.88 Kasus uji ubah jurnal label putih.....	168
Tabel 6.89 Kasus uji lihat detail jurnal label putih.....	169
Tabel 6.90 Kasus uji cetak barcode dan label buku label merah.....	169
Tabel 6.91 Kasus uji cetak barcode dan label buku label merah alternatif 1 :	
Jika tidak melakukan <i>ceklist</i> pada <i>checkbox</i>	170

Tabel 6.92 Kasus uji cetak barcode dan label buku label putih.....	170
Tabel 6.93 Kasus uji cetak barcode dan label buku label putih alternatif 1 :	
Jika tidak melakukan <i>ceklist</i> pada <i>cekbox</i>	171
Tabel 6.94 Kasus uji cetak barcode dan label jurnal label merah.....	171
Tabel 6.95 Kasus uji cetak barcode dan label jurnal label merah alternatif 1 :	
Jika tidak melakukan <i>ceklist</i> pada <i>cekbox</i>	172
Tabel 6.96 Kasus uji cetak barcode dan label jurnal label putih.....	172
Tabel 6.97 Kasus uji cetak barcode dan label jurnal label putih alternatif 1 :	
Jika tidak melakukan <i>ceklist</i> pada <i>cekbox</i>	173
Tabel 6.98 Kasus uji validasi buku.....	173
Tabel 6.99 Kasus uji validasi buku alternatif 1 : Jika field register kosong.....	173
Tabel 6.100 Kasus uji validasi buku alternatif 2 :	
jika field register tidak diisi angka.....	174
Tabel 6.101 Kasus uji validasi jurnal.....	174
Tabel 6.102 Kasus uji validasi jurnal alternatif 1 : Jika <i>field</i> register kosong.....	175
Tabel 6.103 Kasus uji validasi jurnal alternatif 2 :	
Jika field register tidak diisi angka.....	175
Tabel 6.104 Kasus uji logout.....	175
Tabel 6.105 Hasil pengujian <i>web service</i> pada fungsi login.....	176
Tabel 6.106 Hasil pengujian performansi fungsi login.....	177
Tabel 6.107 Analisis Hasil Pengujian <i>White Box</i>	177

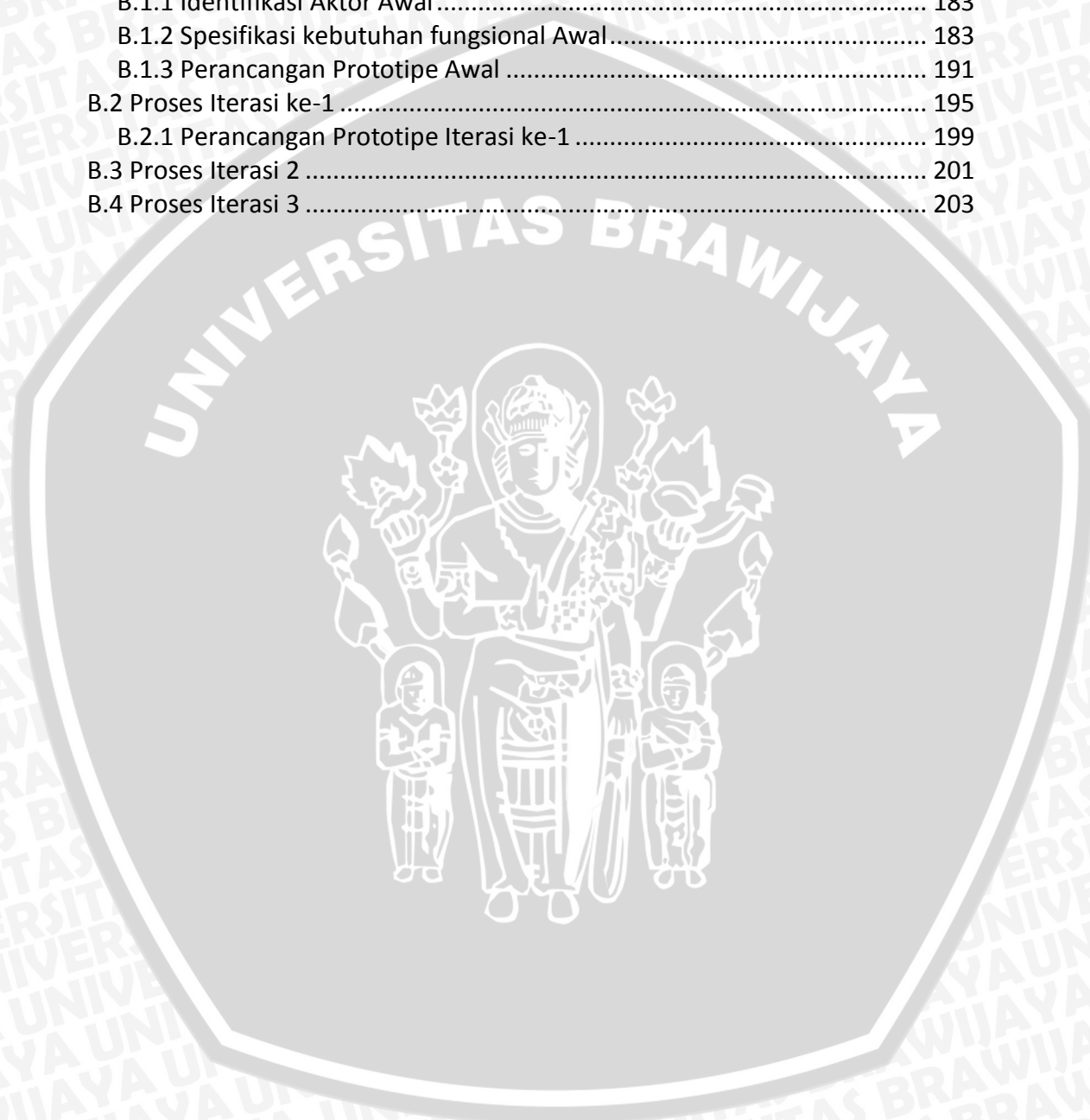
DAFTAR GAMBAR

Gambar 2.1 Struktur Organisasi Perpustakaan Universitas Brawijaya.....	6
Gambar 2.2 Konsep <i>Prototyping Model</i>	9
Gambar 2.3 Fase pada model <i>Evolutionary prototyping</i>	10
Gambar 2.4 Struktur statis aplikasi Yii.....	15
Gambar 2.5 Alur kerja umum aplikasi Yii.....	16
Gambar 3.1 Diagram Alir Metodologi Penelitian.....	18
Gambar 4.1 Diagram Blok Bab Analisis dan Perancangan.....	22
Gambar 4.2 Diagram <i>use case</i> diagram iterasi ke-3.....	36
Gambar 4.3 Arsitektur Umum Sistem.....	82
Gambar 4.4 Diagram <i>sequence login</i>	84
Gambar 4.5 Diagram <i>sequence</i> tambah staf.....	84
Gambar 4.6 Diagram <i>sequence</i> tambah permintaan buku dan jurnal.....	85
Gambar 4.7 Diagram <i>sequence</i> tambah bahan pustaka buku baru.....	86
Gambar 4.8 Diagram <i>sequence</i> tambah buku label putih (<i>copy</i>).....	87
Gambar 4.9 Diagram <i>sequence</i> cetak barcode dan label.....	88
Gambar 4.10 Class diagram login.....	90
Gambar 4.11 <i>Class diagram</i> anggota.....	91
Gambar 4.12 <i>Class diagram</i> staf.....	91
Gambar 4.13 <i>Class diagram</i> permintaan bahan pustaka.....	92
Gambar 4.14 <i>Class diagram</i> pengolahan permintaan bahan pustaka.....	93
Gambar 4.15 <i>Class diagram</i> bahan pustaka buku.....	94
Gambar 4.16 <i>Class diagram</i> bahan pustaka jurnal.....	95
Gambar 4.17 Relational Model.....	96
Gambar 4.18 Rancangan prototipe halaman <i>login</i>	97
Gambar 4.19 Rancangan prototipe halaman utama.....	97
Gambar 4.20 Rancangan prototipe halaman tambah staf.....	98
Gambar 4.21 Rancangan prototipe halaman daftar staf.....	98
Gambar 4.22 Rancangan prototipe halaman biodata anggota.....	99
Gambar 4.23 Rancangan prototipe halaman permintaan individu.....	99
Gambar 4.24 Rancangan prototipe halaman unduh form permintaan kolektif... ..	100

Gambar 4.25 Rancangan prototipe halaman unggah form permintaan.....	100
Gambar 4.26 Rancangan prototipe halaman pengolahan permintaan.....	101
Gambar 4.27 Rancangan prototype halaman registrasi buku.....	101
Gambar 4.28 Rancangan prototipe halaman registrasi jurnal.....	102
Gambar 4.29 Rancangan prototipe halaman registrasi buku secara kolektif.....	102
Gambar 4.30 Rancangan prototipe halaman registrasi jurnal secara kolektif.....	103
Gambar 4.31 Rancangan prototipe halaman daftar buku.....	103
Gambar 4.32 Rancangan prototipe halaman daftar jurnal.....	104
Gambar 4.33 Rancangan prototipe halaman validasi buku.....	104
Gambar 4.33 Rancangan prototipe halaman validasi jurnal.....	104
Gambar 5.1 Diagram Blok Bab Implementasi.....	105
Gambar 5.2 Halaman Login.....	109
Gambar 5.3 Halaman biodata anggota.....	110
Gambar 5.4 Halaman biodata staf.....	110
Gambar 5.5 Halaman tambah staf.....	111
Gambar 5.6 Halaman daftar staf.....	111
Gambar 5.7 Halaman Permintaan bahan pustaka.....	112
Gambar 5.8 Halaman permintaan bahan pustaka secara kolektif.....	112
Gambar 5.9 Halaman pengolahan permintaan bahan pustaka buku.....	113
Gambar 5.10 Halaman pengolahan permintaan bahan pustaka jurnal.....	113
Gambar 5.11 Halaman tambah bahan pustaka buku baru.....	114
Gambar 5.12 Halaman pengolahan bahan pustaka buku secara kolektif.....	114
Gambar 5.13 Halaman daftar bahan pustaka buku label merah.....	114
Gambar 5.14 Halaman validasi bahan pustaka buku.....	115
Gambar 5.15 Halaman pengolahan bahan pustaka jurnal baru.....	115
Gambar 5.16 Halaman pengolahan bahan pustaka jurnal secara kolektif.....	116
Gambar 5.17 Halaman daftar bahan pustaka jurnal label merah.....	116
Gambar 5.18 Halaman validasi bahan pustaka jurnal.....	117
Gambar 6.1 <i>Flow Graph Login</i>	121
Gambar 6.2 <i>Flow graph</i> tambah permintaan bahan pustaka.....	124
Gambar 6.3 <i>Flow graph</i> cari buku label merah.....	127

DAFTAR LAMPIRAN

LAMPIRAN A PERTANYAAN WAWANCARA.....	182
LAMPIRAN B PROSES ITERASI <i>PROTOTYPING</i>	183
B.1 Analisis Kebutuhan Awal	183
B.1.1 Identifikasi Aktor Awal	183
B.1.2 Spesifikasi kebutuhan fungsional Awal.....	183
B.1.3 Perancangan Prototipe Awal	191
B.2 Proses Iterasi ke-1	195
B.2.1 Perancangan Prototipe Iterasi ke-1	199
B.3 Proses Iterasi 2	201
B.4 Proses Iterasi 3	203



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Perpustakaan merupakan suatu institusi pengelola koleksi karya tulis, karya cetak, dan karya rekam secara profesional dengan sistem yang baku untuk memenuhi kebutuhan pendidikan, penelitian, pelestarian, dan kebutuhan akan informasi serta rekreasi para pemustaka menurut Undang-Undang Republik Indonesia nomor 43 tahun 2007 tentang Perpustakaan. Perpustakaan Universitas Brawijaya merupakan salah satu sistem pendukung universitas yang berfungsi mendukung program akademik universitas yang tertuang dalam "Tri Darma Perguruan Tinggi" mencakup pendidikan, penelitian dan pengabdian kepada masyarakat. Peran perpustakaan untuk mendukung proses pendidikan yaitu memberikan informasi, mengkoordinasikan dan menggabungkan semua bentuk layanan untuk meningkatkan proses belajar mengajar, penelitian dan layanan umum. Selain memiliki perpustakaan utama yang menjadi pusat dari koleksi bahan pustaka, Universitas Brawijaya juga memiliki perpustakaan di setiap fakultas maupun jurusan. Perpustakaan yang terletak pada fakultas atau jurusan disebut ruang baca.

Perpustakaan Universitas Brawijaya dalam meningkatkan pelayanannya, sudah memiliki sistem informasi manajemen perpustakaan. Namun sistem informasi tersebut belum mampu mengintegrasikan data-data bahan pustaka maupun karya ilmiah yang ada di setiap ruang baca dengan perpustakaan pusat. Akibat dari data yang masih tersebar, proses pencarian suatu bahan pustaka hanya mencakup perpustakaan pusat saja. Proses pencarian dilakukan menggunakan OPAC (*Online Public Access Catalog*) yang tersedia pada perpustakaan pusat. OPAC merupakan sistem katalog *online* yang dapat diakses secara umum dan dapat dipakai pengguna untuk menelusuri data katalog untuk memastikan apakah perpustakaan menyimpan karya tertentu untuk mendapatkan informasi tentang keberadaan bahan pustaka. Bahan pustaka yang ada pada setiap ruang baca belum terdeteksi oleh OPAC. Oleh karena itu untuk mengatasi permasalahan diatas diperlukan suatu sistem yang dapat mengintegrasikan data bahan pustaka yang ada di setiap ruang baca.

Perpustakaan Universitas Brawijaya terus menambah koleksi bahan pustakanya guna memenuhi kebutuhan mahasiswa. Berdasarkan data yang diperoleh dari perpustakaan pusat Universitas Brawijaya melalui wawancara bahwa saat ini bahan pustaka khususnya buku yang dimiliki sebanyak 156.776 buku. Koleksi-koleksi tersebut tidak semua pernah dipinjam oleh anggota perpustakaan, yaitu sekitar 84.075 buku atau dapat dipresentasikan 53,6%. Hal ini mungkin disebabkan bahan pustaka yang diadakan masih kurang tepat sasaran. Oleh karena itu pihak perpustakaan membuat kebijakan bahwa setiap mahasiswa maupun ruang baca yang ada di Universitas Brawijaya dapat mengajukan bahan pustaka baru.

Pengajuan bahan pustaka baru yang dilakukan oleh mahasiswa, masih dilakukan secara manual, yaitu dengan cara menuliskan permintaan bahan pustaka baru pada selebar form. Hal ini yang membuat prosesnya masih memerlukan waktu yang cukup lama. Dengan demikian, diperlukan sistem yang lebih efisien waktu agar proses bisnis yang ada di perpustakaan lebih cepat dari pada sebelumnya. Sistem dibangun menggunakan integrasi basis data dengan *web service*, sehingga mahasiswa dapat langsung masuk kedalam sistem dengan menggunakan akun SIAKAD UB tanpa harus registrasi terlebih dahulu.

Dalam pengembangan perangkat lunak seorang pengembang dituntut untuk memenuhi kebutuhan pengguna. Seorang pengguna membayangkan kumpulan kebutuhan yang diinginkan tapi tidak terspesifikasikan secara detail dari segi masukan (*input*), proses, maupun keluaran (*output*). Disisi lain seorang pengembang perangkat lunak harus menspesifikasikan kebutuhan secara detail dari segi teknis, dimana pengguna sering kurang mengerti mengenai hal teknis ini. Dalam kasus ini daftar kebutuhan akan sistem belum spesifik. Untuk itu diperlukan suatu model proses pembuatan perangkat lunak yang dapat menggambarkan atau mendemonstrasikan sebagian besar fungsi perangkat lunak yang akan dibangun, memperjelas spesifikasi kebutuhan yang diinginkan pengguna, serta dapat memberikan umpan balik kepada pengembang sebagai bahan evaluasi (Sukamto dan Shalahuddin, 2013). Adapun model proses yang digunakan pada penelitian ini yaitu *prototyping*. Model *prototyping* merupakan salah satu model proses dalam rekayasa perangkat lunak. Pada model *prototyping* pengembang dan pengguna dapat saling berinteraksi selama proses pengembangan sistem, sehingga perangkat lunak yang dihasilkan nantinya sesuai dengan keinginan dan kebutuhan pengguna (Pressman, 2012).

Berdasarkan hal-hal yang menjadi latar belakang diatas, maka penulis melakukan penelitian dengan judul “**Pengembangan Aplikasi Web Terintegrasi pada Perpustakaan Universitas Brawijaya dengan Model *Prototyping*”**.”

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijabarkan maka dapat dibuat rumusan masalah yaitu, bagaimanakah membangun aplikasi web yang dapat :

1. memanfaatkan *web service* SIAKAD UB agar dapat mengintegrasikan data yang sesuai dengan kebutuhan sistem,
2. mengintegrasikan data bahan pustaka yang masih tersebar dan
3. menampung dan mengolah permintaan bahan pustaka baru?

1.3 Batasan Masalah

Batasan penelitian yang dilakukan pada penulisan skripsi ini mempunyai beberapa batasan masalah berikut:

1. Studi Kasus dilakukan di perpustakaan Universitas Brawijaya.
2. Sistem dan basis data akan dipusatkan pada satu tempat yaitu perpustakaan pusat Universitas Brawijaya.

3. Sistem difokuskan pada bagian permintaan dan pengolahan bahan pustaka.
4. Tahap evaluasi *prototyping* dilakukan sebanyak 3 kali.

1.4 Tujuan

Tujuan yang ingin dicapai dalam penelitian ini yaitu membangun aplikasi web yang dapat mengintegrasikan data bahan pustaka yang tersebar. Membangun aplikasi web yang dapat menampung dan mengolah permintaan bahan pustaka baru yang terdapat pada Perpustakaan Universitas Brawijaya dengan menggunakan model *prototyping*. Dengan demikian dapat mengatasi permasalahan atau kekurangan pada sistem yang sudah pernah ada sebelumnya.

1.5 Manfaat

Adapun manfaat yang diharapkan dari penelitian ini adalah:

1. Adanya aplikasi ini dapat mempermudah proses bisnis di Perpustakaan Universitas Brawijaya.
2. Bagi Perpustakaan Universitas Brawijaya dapat memberikan sistem informasi yang saling terintegrasi.
3. Bagi penulis adalah sebagai pengembangan dan penerapan terhadap pembelajaran yang didapat selama masa perkuliahan.

1.6 Sistematika Penulisan

Dalam pembuatan skripsi ini digunakan sistematika penulisan sebagai berikut:

1. BAB I PENDAHULUAN

Bab ini berisi latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian dan sistematika penulisan skripsi.

2. BAB II KAJIAN PUSTAKA DAN DASAR TEORI

Bab ini membahas tentang dasar teori yang terkait dengan topik penulisan skripsi yang diangkat, yang menjadi acuan dasar dalam pembuatan sistem ini.

3. BAB III MODEL PENELITIAN

Bab ini membahas mengenai model yang dipakai oleh penulis dalam melakukan penelitian mengenai analisis kebutuhan pengguna, perancangan sistem, implementasi sistem, serta pengujian dan analisis.

4. BAB IV PERANCANGAN SISTEM

Membahas tentang analisis kebutuhan dan perancangan sistem yang kemudian menjadi landasan implementasi.

5. BAB V IMPLEMENTASI SISTEM

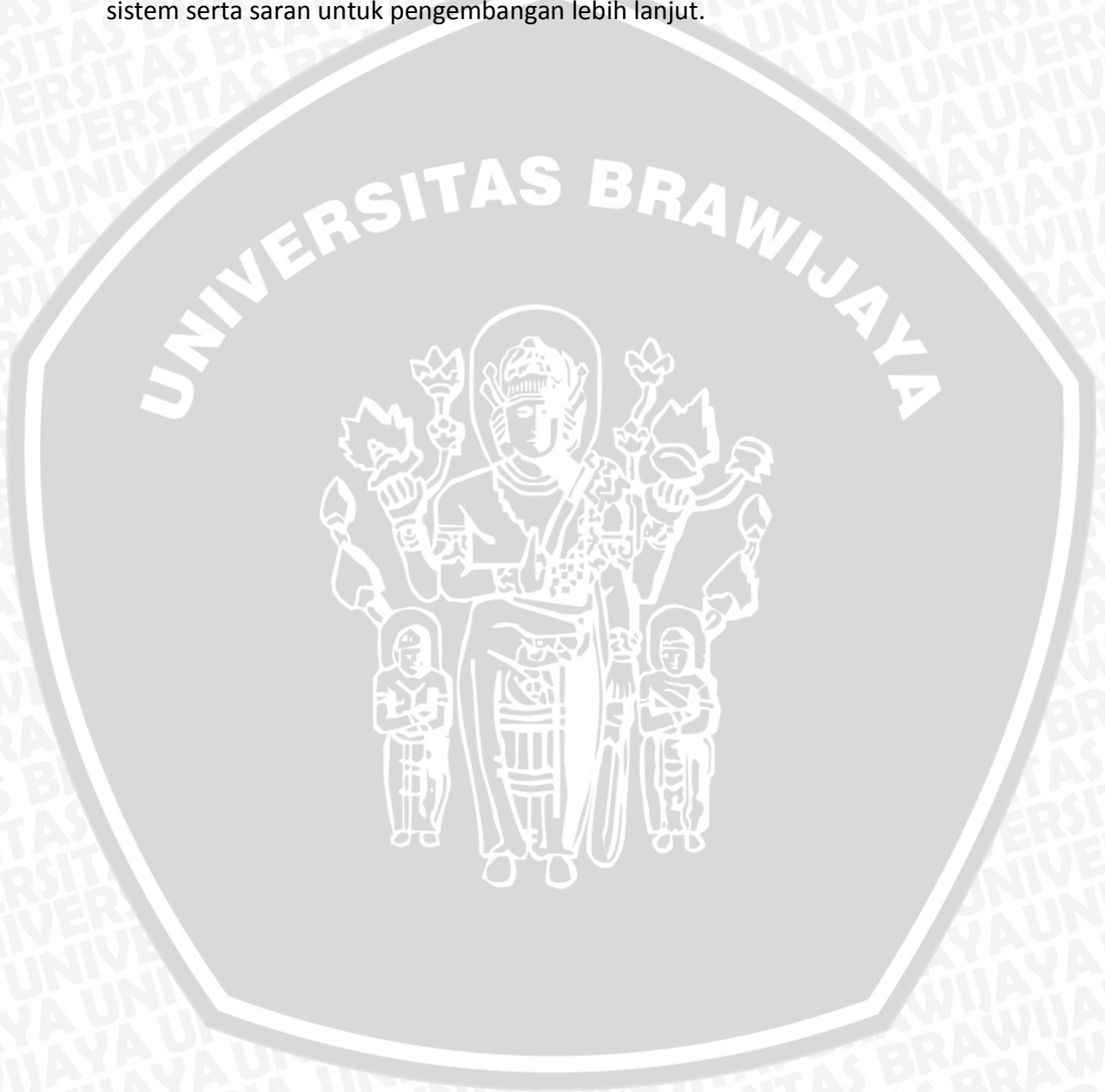
Membahas implementasi perangkat lunak sesuai dengan perancangan sistem yang telah dibuat.

6. BAB VI PENGUJIAN DAN ANALISIS

Membahas proses dan hasil pengujian terhadap sistem yang telah diimplementasi dan melakukan analisis terhadap hasil pengujian.

7. BAB VII PENUTUP

Bab ini berisi kesimpulan yang diperoleh dari pembuatan dan pengujian sistem serta saran untuk pengembangan lebih lanjut.



BAB 2 LANDASAN KEPUSTAKAAN

Bab ini terdiri dari kajian pustaka dan dasar teori. Kajian pustaka merupakan pembahasan penelitian yang sudah ada untuk mendukung penelitian yang akan diusulkan. Dasar teori membahas teori yang diperlukan untuk menyusun penelitian yang diusulkan.

2.1 Kajian Pustaka

Penelitian ini membahas tentang “Pengembangan Aplikasi Web Terintegrasi pada Perpustakaan Universitas Brawijaya dengan Model *Prototyping*”. Penelitian tentang penerapan model *prototyping* dalam perangkat lunak pernah dilakukan sebelumnya oleh oleh Xiaoshuan Zhang yang berjudul “*Applying evolutionary prototyping model for eliciting system requirement of meat traceability at agribusiness level*”. Penelitian ini membahas tentang sistem traceability berbasis TI pada tingkat agribisnis. Sistem ini menggunakan model *evolutioner prototyping* untuk memperoleh kebutuhan sistem karena persyaratan sistem dapat dikategorikan sebagai tidak spesifik. Tujuan dari penelitian ini adalah untuk membangun sistem traceability yang dapat memvalidasi persyaratan sistem pada tingkat agribisnis. Penelitian ini menghasilkan proses evolusi dari version 0.0, version 1.0 ke version 2.0 pada tahap evaluasi (Zhang, et al., 2010).

Penelitian serupa juga dilakukan oleh Xiaoshuan Zhang yang berjudul “*Applying evolutionary prototyping model in developing FIDSS: An intelligent decision support system for fish disease/health management*”. Penelitian ini membahas tentang membangun sebuah sistem pendukung keputusan penyakit ikan atau manajemen kesehatan. Sistem ini menggunakan dua jenis model *prototyping*, diantaranya *throwaway prototyping* dan *evolutionary prototyping*. *Throwaway prototyping* digunakan untuk menentukan kebutuhan aktual sistem dan pembuktian konsep sedangkan *evolutionary prototyping* untuk mengembangkan sistem (Zhang, et al., 2009).

Berdasarkan kedua penelitian tersebut, penulis menjadikan kedua penelitian tersebut sebagai salah satu acuan referensi. Pada penelitian diatas menjadikan *evolutionary prototyping* model sebagai acuan development proses. Penulis mereferensi cara penamaan version pada implementasi sistem dan *feedback* pengguna, dimana prototipe evolusioner terus menerus disempurnakan sampai memiliki seluruh fungsionalitas yang dibutuhkan pengguna.

2.2 Dasar Teori

2.2.1 Pengertian Perpustakaan

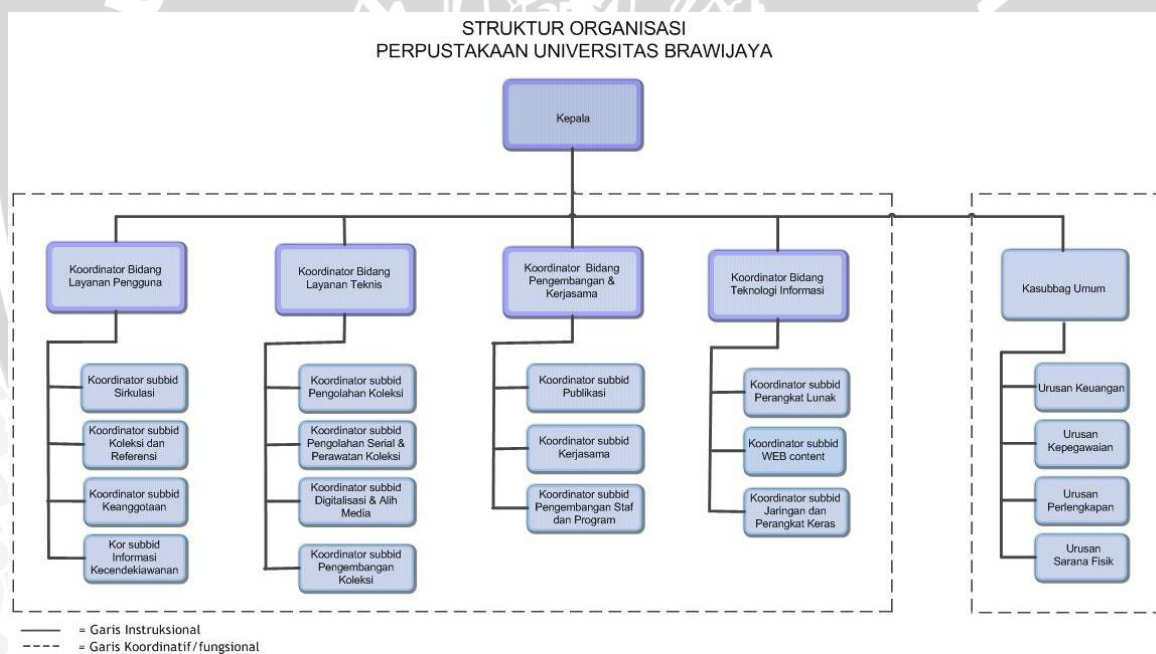
Perpustakaan merupakan suatu institusi pengelola koleksi karya tulis, karya cetak, dan karya rekam secara profesional dengan sistem yang baku untuk memenuhi kebutuhan pendidikan, penelitian, pelestarian, dan kebutuhan akan informasi serta rekreasi para pemustaka menurut Undang-Undang Republik Indonesia nomor 43 tahun 2007 tentang Perpustakaan.

Perpustakaan menurut Basuki (1991) adalah sebuah ruangan yang didalamnya terdapat buku dan terbitan lainya yang disimpan berdasarkan tata susunan tertentu yang digunakan pembaca dan tidak untuk perjuan belikan.

Dapat disimpulkan bahwa perpustakaan adalah suatu institusi unit kerja yang menyimpan koleksi bahan pustaka secara sistematis dan mengelolanya dengan cara khusus sebagai sumber informasi dan dapat digunakan oleh pemakainya.

2.2.1.1 Perpustakaan Universitas Brawijaya

Perpustakaan Universitas Brawijaya merupakan salah satu sistem pendukung universitas, berfungsi mendukung program akademik universitas yang tertuang dalam “Tri Darma Perguruan Tinggi” yang mencakup pendidikan penelitian dan pengabdian kepada masyarakat. Beberapa peran perpustakaan dalam mendukung proses pendidikan antara lain adalah memberikan informasi, mengkoordinasikan dan menggabungkan semua bentuk layanan untuk meningkatkan proses belajar mengajar, penelitian dan layanan umum. Pada akhirnya tujuannya adalah tercapainya proses peningkatan kualitas lulusan dalam hal pengembangan wawasan dan penguasaan keilmuannya.



Gambar 2.1 Struktur Organisasi Perpustakaan Universitas Brawijaya

Sumber : Perpustakaan Universitas Brawijaya (2014)

Stuktur organisasi yang terdapat pada **Gambar 2.1** memiliki beberapa bidang yang di kepalai oleh kepala perpustakaan. Adapun beberapa bidang di antaranya yaitu, bidang layanan pengguna, bidang layanan teknis, bidang pengembangan dan kerjasama, bidang teknologi informasi dan sub bagian umum. Masing-masing bidang memiliki sub bidang. Bidang layanan pengguna memiliki

sub bidang sirkulasi, koleksi dan referensi, keanggotaan dan informasi kecendekiawanan. Bidang layanan teknis memiliki sub bidang pengolahan koleksi, pengolahan serial dan perawatan koleksi, digitalisasi dan alih media, dan pengembangan koleksi. Bidang pengembangan dan kerjasama memiliki sub bidang publikasi, kerjasama, dan pengembangan staf dan program. Bidang teknologi informasi memiliki sub bidang perangkat lunak, web content, dan jaringan dan perangkat keras. Sub bagian umum meliputi beberapa bagian yaitu, urusan keuangan, urusan kepegawaian, urusan perlengkapan dan urusan sarana fisik.

2.2.2. Pengertian 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 (Sommerville, 2003). Perekayasa perangkat lunak memakai pendekatan yang sistematis dan terorganisir terhadap pekerjaan mereka karena cara ini seringkali paling efektif masalah pemilihan metode yang paling efektif untuk menghasilkan perangkat lunak berkualitas tinggi. Rekayasa ini mencakup masalah pemilihan metode yang paling sesuai untuk satu set keadaan dan pendekatan yang lebih kreatif, informasi terhadap pengembangan yang mungkin efektif pada beberapa keadaan (Sommerville, 2003).

Seorang software engineer harus menggabungkan strategi pengembangan perangkat lunak yang meliputi proses, metode, dan alat bantu yang digunakan dalam proses pengembangan. Strategi ini disebut sebagai model proses atau paradigma rekayasa perangkat lunak dipilih sesuai dengan sifat dari proyek dan aplikasi yang akan dibuat. Terdapat beberapa model proses untuk rekayasa perangkat lunak antara lain linear sequential model (waterfall), Prototyping, Rapid Application Development (RAD), incremental model dan spiral model (Pressman, 2010). Pada penelitian ini menggunakan model *prototyping*.

2.2.3 Pengertian Prototipe

Prototipe merupakan versi awal dari sistem perangkat lunak yang digunakan untuk mendemonstrasikan atau menggambarkan konsep dari perangkat lunak yang akan dikembangkan. Setain itu, prototipe digunakan untuk menentukan kebutuhan-kebutuhan perangkat lunak (Sommerville, 2003).

Keuntungan menggunakan prototipe pada proses pengembangan perangkat lunak, diantaranya.

1. Kesalahpahaman antara pengembang perangkat lunak dengan pengguna dapat diidentifikasi ketika fungsi sistem didemonstrasikan.
2. Prototipe dapat digunakan sebagai dasar penulisan spesifikasi untuk sistem kualitas produksi.

2.2.4 Model *Prototyping*

Model *prototyping* merupakan salah satu metode dalam proses pengembangan perangkat lunak. Dalam metode ini pengembang dan pengguna dapat saling berinteraksi selama proses pengembangan sistem sehingga nantinya diharapkan dapat tercipta suatu sistem yang sesuai dengan harapan dan kebutuhan dari pengguna. Pengembang dapat secara langsung mendemonstrasikan bagaimana sebuah perangkat lunak atau komponen-komponen perangkat lunak akan bekerja dalam lingkungannya sebelum tahapan konstruksi aktual dilakukan (Pressman, 2012).

Seorang pelanggan terkadang tidak dapat mendefinisikan secara rinci kebutuhan apa saja yang dibutuhkan pada sistem, disinilah seorang pengembang bertugas untuk memberi gambaran kepada pengguna mengenai rancangan perangkat lunak yang akan dibuat tanpa mengesampingkan tujuan awal dari pembuatan perangkat lunak. Oleh sebab itu, dibutuhkan sebuah interaksi yang baik antara pengembang dan pengguna selama proses pembuatan perangkat lunak.

Berikut adalah tahapan-tahapan proses pengembangan dalam model *prototyping*.

1. Analisis Kebutuhan

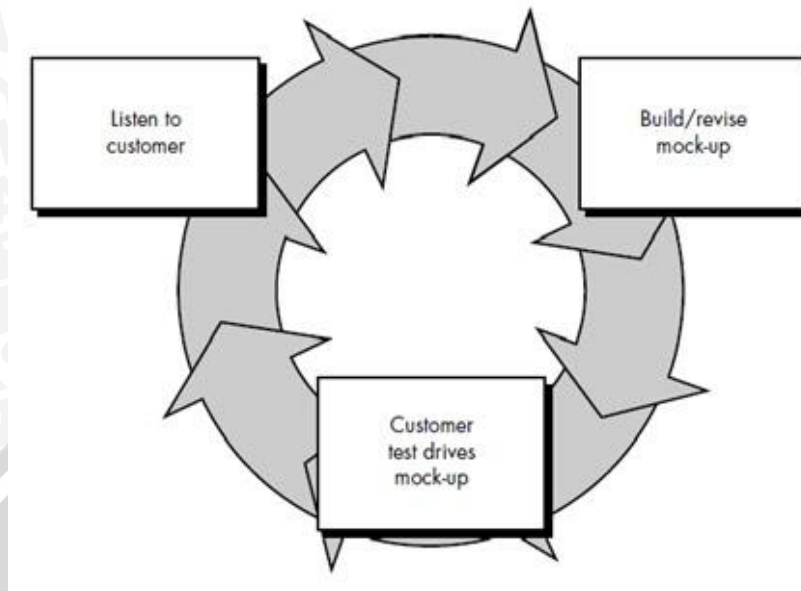
Tahapan ini merupakan tahapan yang pertama kali harus dilakukan oleh pengembang sebelum melakukan perancangan dan pembuatan perangkat lunak. Pada tahapan ini dilakukan pengumpulan kebutuhan dari sistem dengan cara berkomunikasi antara pengembang dan pengguna / pelanggan. Proses ini dapat dilakukan melalui wawancara, kuesioner, atau pun survey secara langsung untuk dapat menyimpulkan fitur apa saja yang sekiranya dibutuhkan pada perangkat lunak nanti.

2. Merancang dan membuat prototipe

Pada tahap ini, dilakukan perancangan dan pembuatan prototipe perangkat lunak. Prototipe yang dibuat disesuaikan dengan kebutuhan perangkat lunak yang telah didefinisikan pada tahap sebelumnya.

3. Evaluasi prototipe

Proses evaluasi prototipe dilakukan oleh pengguna. Pada tahapan ini pengguna dapat menilai apakah rancangan awal yang dibangun pengembang telah sesuai dengan yang diharapkan pengguna. Jika tidak, maka pelanggan dapat melakukan revisi prototipe dan menyesuaikan dengan kebutuhan. Maka dengan adanya proses ini diharapkan perangkat lunak nantinya akan sesuai dengan kebutuhan pengguna.



Gambar 2.2 Konsep *Prototyping Model*

Sumber : Pressman (2010)

Perulangan proses ini terus berlangsung hingga semua kebutuhan terpenuhi. Segala perubahan dapat terjadi pada saat prototipe dibuat untuk memenuhi kebutuhan pengguna. Setelah rancangan perangkat lunak disetujui oleh pengguna, maka langkah selanjutnya dilakukan proses pengkodean sistem dan pengujian sistem. Selanjutnya langkah terakhir yang dilakukan setelah perangkat lunak selesai adalah penyerahan kepada pelanggan untuk siap digunakan (Pressman, 2010).

Prototyping model memiliki 2 jenis pemodelan, yaitu throw-away prototyping dan evolutionary prototyping.

1. *Throw-away Prototyping*

Throw-away prototyping adalah model lifecycle perangkat lunak yang digunakan untuk mengembangkan sistem dimana pengembangan tidak memiliki pemahaman tentang yang jelas tentang perangkat lunak yang akan dikembangkan. Model ini digunakan untuk menyempurnakan dan memperjelas spesifikasi kebutuhan perangkat lunak. Prototipe yang dibangun akan terus di evaluasi sampai pelanggan merasa puas dan kebutuhan yang diharapkan terpenuhi. Prototipe dievaluasi dan dimodifikasi untuk mendapatkan spesifikasi kebutuhan yang terinci. Setelah proses evaluasi dilakukan prototipe akan dibuang dan tidak dipakai sebagai dasar pengembangan perangkat lunak lebih lanjut (Sommerville, 2003).

2. *Evolutionary prototyping*

Evolutionary prototyping adalah model *lifecycle* perangkat lunak di mana sistem dikembangkan secara bertahap sehingga mudah untuk dimodifikasi dalam menanggapi kebutuhan pengguna dan umpan balik dari pengguna.

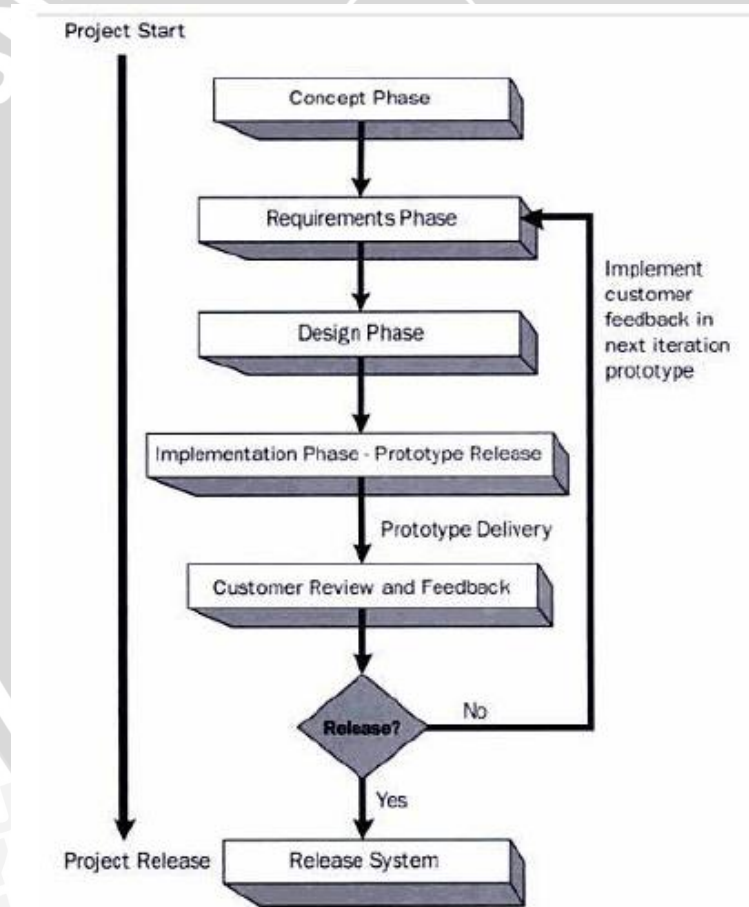
Pengembangan *evolutionary prototyping* dimulai dengan prototipe antarmuka dan kemudian mengembangkan sistem dari kebutuhan pengguna.

Evolutionary prototyping tidak membuang kode prototyping yang telah dibuat tetapi mengembangkan menjadi kode yang dibutuhkan user. Prototipe evolusioner terus menerus disempurnakan sampai memiliki seluruh fungsionalitas yang dibutuhkan pengguna dari sistem yang baru. Prototipe ini kemudian dilanjutkan dengan produksi. Jadi satu prototipe evolusioner akan menjadi sistem aktual (McConnell, 1996).

Pada penelitian ini akan menggunakan model *evolutionary prototyping*, sistem akan dikembangkan sampai memenuhi kebutuhan pengguna. Namun pada kasus ini perulangan evaluasi dibatasi sebanyak tiga kali iterasi.

2.2.5 Fase Evolutionary prototyping

Berikut merupakan fase-fase dalam model *evolutionary prototyping* yang dilakukan, dapat dilihat pada Gambar 2.3.



Gambar 2.3 Fase pada model *Evolutionary prototyping*

Sumber : Gunther dan Moeller (2003)

Berikut ini merupakan penjelasan dari fase pendekatan *evolutionary prototyping*:

1. Fase Konsep:

Pada tahap ini sebuah konsep perangkat lunak yang pertama kali dilakukan. Dokumen ini dapat terdiri dari tujuan bisnis untuk proyek atau ide untuk produk baru yang inovatif.

2. Fase Analisis Kebutuhan

Pada tahap ini dilakukan analisis kebutuhan dari sistem dengan cara bertemu dengan pelanggan. Untuk membuat suatu sistem yang sesuai kebutuhan perangkat lunak yang akan direpresentasikan kedalam bentuk prototipe.

3. Fase Perancangan

Dari tahap analisis kebutuhan, kemudian kebutuhan dispesifikasikan kedalam bentuk fungsional sistem. Selanjutnya dilakukan perancangan sesuai dengan kebutuhan customer. Jika customer tidak puas maka dilakukan iterasi hingga customer setuju.

4. Implementasi – Fase Rilis Prototipe

Pada tahap ini, melakukan implementasi prototipe dari hasil spesifikasi kebutuhan dan perancangan sistem. Selanjutnya prototipe diberikan kepada pengguna atau pelanggan untuk dievaluasi.

5. Customer Review dan Feedback

Pada tahap ini dilakukan *review* oleh pengguna atau pelanggan untuk memberikan masukan tentang sistem yang telah dibuat, serta memberikan perubahan dan fitur baru untuk dibuat pada iterasi selanjutnya, atau prototipe disetujui oleh pelanggan dan menjadikan prototipe sebagai produk final.

2.2.6 Web Service

Web service adalah aplikasi basis data (*database*), perangkat lunak (*software*) atau bagian dari perangkat lunak yang dapat diakses secara *remote* oleh berbagai perangkat dengan sebuah perantara tertentu. Secara umum, *web service* dapat diidentifikasi dengan menggunakan URL seperti *website* pada umumnya. Namun yang membedakan *web service* dengan *website* pada umumnya adalah interaksi yang diberikan oleh *web service*. Berbeda dengan URL *website*, URL *web service* hanya mengandung kumpulan informasi, perintah, konfigurasi atau sintaks yang berfungsi untuk membangun sebuah fungsi-fungsi tertentu aplikasi. *Web service* dapat diartikan juga sebuah metode pertukaran data, tanpa memperhatikan dimana sebuah *database* disimpan, dibuat dalam bahasa apa sebuah aplikasi yang mengkonsumsi data, dan di *platform* apa sebuah

data tersebut dikonsumsi. *Web service* mampu menjadi sebuah jembatan penghubung antara berbagai sistem yang ada (Nolan, et al., 2013).

Website pada umumnya digunakan untuk melakukan *request* dan *response* yang dilakukan antara *client* dengan *server*. Sebagai contoh, seorang pengguna layanan *web* tertentu mengetikkan alamat url *website* untuk membentuk sebuah *request*. Maka *request* akan sampai pada *server*, kemudian diolah dan disajikan dalam bentuk sebuah *response*. Dengan singkat kata terjadilah hubungan *client-server* secara sederhana. Sedangkan pada *web service* hubungan antara *client* dan *server* tidak terjadi secara langsung. Hubungan antara *client* dan *server* dijumpai oleh data *web service* dalam format tertentu. Sehingga akses terhadap *database* tidak akan ditangani secara langsung oleh *server*, melainkan melalui perantara yang disebut *web service*. Peran dari *web service* ini akan mempermudah distribusi sekaligus integrasi *database* yang tersebar di beberapa *server* sekaligus (Nolan, et al., 2013).

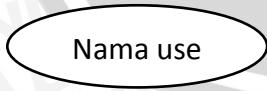

2.2.7 Unified Modeling Language (UML)

UML (*Unified Modeling Language*) merupakan sebuah bahasa grafik standar yang digunakan untuk memodelkan perangkat lunak berbasis objek. Pemodelan digunakan untuk menyederhanakan permasalahan yang kompleks sehingga lebih mudah di pahami dan di pelajari (Nugroho, 2010). UML pertama kali dikembangkan pada pertengahan 1990-an dengan kerjasama antara James Rumbaugh, Grady Booch dan Ivar Jacobson, yang masing-masingnya telah mengembangkan notasi mereka sendiri diawal 1990-an (Lethbridg & Laganiere 2002).


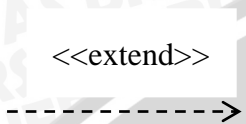

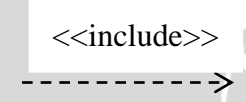
2.2.7.1 Diagram Use Case

Use case merupakan suatu pemodelan, yaitu apa yang akan dilakukan aktor terhadap sistem. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Adapun simbol yang digunakan dalam diagram *use case* sebagai berikut (Fazri, 2013).

Tabel 2.1 *Use case Diagram*

No.	Simbol	Deskripsi
1.		Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama use case.
2.		Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar dari orang, tapi aktor belum tentu merupakan orang.

Tabel 2.1 (lanjutan)

No.	Simbol	Deskripsi
3.	Asosiasi / <i>Association</i> 	Komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor.
4.	Ekstensi / <i>Extend</i> 	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu; mirip dengan prinsip inheritance pada pemrograman berorientasi objek; biasanya use case tambahan memiliki nama depan yang sama dengan use case yang ditambahkan.
5.	Generalisasi / <i>Generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6.	Menggunakan / <i>include</i> 	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini.

Sumber : Diadaptasi dari Fazri (2013)

2.2.7.2 Diagram Sequence

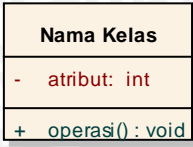




Diagram *sequence* adalah suatu diagram yang menggambarkan interaksi antar obyek dan mengindikasikan komunikasi diantara obyek-obyek selama eksekusi suatu pekerjaan berlangsung. Diagram ini juga menunjukkan serangkaian pesan yang dipertukarkan oleh obyek-obyek yang melakukan suatu tugas atau aksi tertentu. Obyek-obyek tersebut kemudian diurutkan dari kiri ke kanan, aktor yang menginisiasi interaksi biasanya ditaruh di paling kiri dari diagram. *Sequence diagram* juga dapat digunakan untuk menunjukkan interaksi-interaksi pada suatu *use case* atau pada suatu skenario dari suatu sistem perangkat lunak (Pressman, 2010).

2.2.7.3 Diagram Kelas

Diagram kelas digunakan saat mengembangkan suatu model sistem yang berorientasi objek untuk menunjukkan kelas dalam sistem dan hubungan antara setiap kelas yang ada. Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem sehingga pengembang dapat membuat kelas-kelas di dalam program perangkat lunak sesuai dengan perancangan diagram kelas. Setiap kelas memiliki informasi dengan kelas yang

terkait. Diagram kelas menggambarkan hubungan apa yang terjadi di sistem bukan apa yang terjadi jika mereka berhubungan satu sama lain. Hubungan yang ada pada diagram kelas ada beberapa macam misalnya seperti asosiasi, agregasi, dan pewarisa (Kurniawati, 2014).

Tabel 2.2 Keterangan Simbol - Simbol Class Diagram

No	Gambar	Nama	Keterangan
1		Class / Kelas	Kelas yang terdapat dalam sistem
2		Association / Asosiasi	Hubungan antar kelas secara umum
3		Dependency / Kbergantungan	Hubungan antar kelas dimana suatu kelas bergantung pada kelas yang lain.
4		Generalization / Generalisasi	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
5		Agregation / Agregasi	Hubungan yang menyatakan bagian ("terdiri atas..") atau biasa disebut relasi mempunyai sebuah.

Sumber : Diadaptasi dari Rosa & Shalahudin (2011)

2.2.8 Pengertian Basis Data

Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat.

Sistem informasi tidak dapat dipisahkan dengan kebutuhan akan basis data apapun bentuknya, entah berupa *file* teks maupun *Database Management System* (DBMS). Kebutuhan basis data dalam sistem informasi meliputi (Sukamto dan Shalahuddin, 2013) :

- Memasukkan, menyimpan, dan mengambil data.
- Membuat laporan berdasarkan data yang telah disimpan.

Tujuan dari dibuatnya tabel adalah untuk menyimpan data ke dalam tabel agar mudah diakses. Oleh karena itu dalam merancang tabel diperlukan pemikiran yang dalam karena data yang akan disimpan nantinya dalam bentuk baris data (*record*).



2.2.9 Model Data Relasional

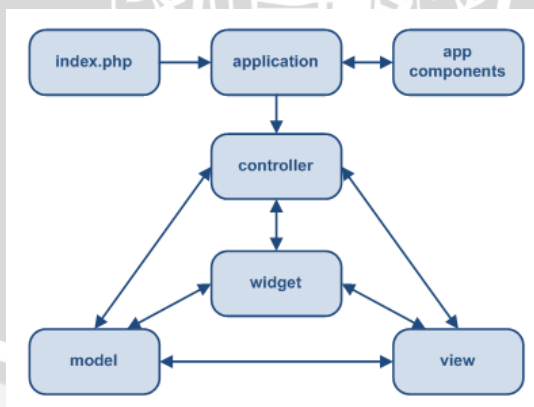
Model data relasional sering juga disebut sebagai model relasional atau basis data relasional merupakan hubungan antar data dalam basis data dengan merepresentasikannya ke dalam bentuk relasi-relasi berupa tabel mendatar yang terdiri atas sejumlah baris yang menunjukkan *record* dan kolom yang menunjukkan atribut tertentu. Relasi dirancang sedemikian rupa sehingga dapat menghilangkan kerangkapan data yang tidak berguna. Dalam sebuah basis data, hubungan antar relasi satu dengan yang lainnya ditunjukkan menggunakan *foreign key*, sering disingkat FK atau relasi bertipe transaksi (Sutanta, 2011).

2.2.10 Framework Yii

Yii adalah *framework* PHP berbasis komponen yang digunakan untuk pengembangan aplikasi web. Yii menyediakan *reusability* dalam pemrograman web dan mampu meningkatkan kecepatan pengembangan secara signifikan.

Framework Yii menerapkan pola desain *model-view-controller* (MVC). MVC bertujuan untuk memisahkan logika bisnis dari pertimbangan antar muka pengguna agar para pengembang bisa lebih mudah mengubah setiap bagian tanpa mempengaruhi yang lain. Dalam MVC, *model* menggambarkan informasi atau data dan aturan bisnis. *View* berisi elemen antar muka pengguna seperti teks, input form. Sedangkan *controller* mengatur komunikasi antar *model* dan *view*.

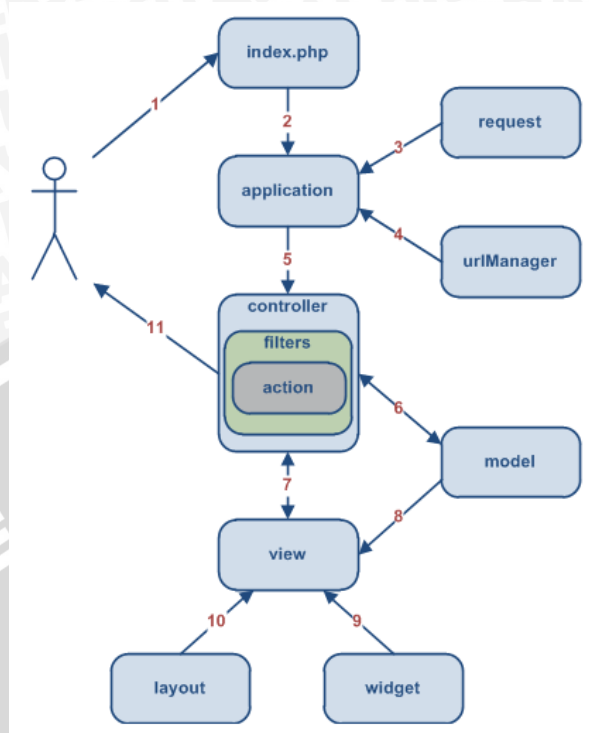
Selain implementasi MVC, Yii juga memperkenalkan *front-controller*, yang disebut *aplikasi*, yang mengenkapsulasi konteks eksekusi untuk memproses sebuah *request*. Aplikasi mengumpulkan beberapa informasi mengenai *request* pengguna dan kemudian mengirimnya ke *controller* yang sesuai untuk penanganan selanjutnya (Sharive, 2013). **Gambar 2.3** memperlihatkan struktur statis sebuah aplikasi Yii:



Gambar 2.3 Struktur statis aplikasi Yii

Sumber : Shariv (2013).

Diagram berikut memperlihatkan alur kerja umum sebuah aplikasi Yii saat menangani permintaan pengguna (Sharive, 2013):



Gambar 2.4 Alur kerja umum aplikasi Yii

Sumber : Sharive (2013)

1. Pengguna membuat permintaan dengan URL `http://www.example.com/index.php?r=post/show&id=1` dan server web menangani permintaan dengan menjalankan skrip bootstrap `index.php`.
2. Skrip bootstrap membuat sebuah instance aplikasi dan menjalankannya.
3. Aplikasi mendapatkan rincian informasi permintaan pengguna dari komponen aplikasi bernama *request*.
4. Aplikasi menentukan *controller* dan aksi yang diminta dengan bantuan komponen aplikasi bernama *urlManager*. Dalam contoh ini, *controller* adalah *post* yang merujuk pada kelas *PostController* dan aksi adalah *show* yang arti sebenarnya ditentukan oleh *controller*.
5. Aplikasi membuat *instance controller* yang diminta untuk selanjutnya menangani permintaan pengguna. *Controller* menentukan aksi *show* merujuk pada sebuah metode bernama *actionShow* dalam kelas *controller*. Kemudian membuat dan menjalankan *filter* (contoh kontrol akses, pengukuran) terkait dengan aksi ini. Aksi dijalankan jika diijinkan oleh filter.
6. Aksi membaca *post model* di mana ID adalah 1 dari database.
7. Aksi menyiapkan *view* bernama *show* dengan *model Post*.
8. *View* membaca dan menampilkan atribut *model Post*.
9. *View* menjalankan beberapa *widget*.

10. *View* menyiapkan hasil yang dipasangkan dalam *layout*.

11. Aksi mengakhiri pembuatan *view* dan menampilkan hasil akhir kepada pengguna.

2.2.11 Konsep Pengujian Perangkat Lunak

Pengujian perangkat lunak membutuhkan perancangan kasus uji dengan tujuan untuk mendapatkan kesalahan dengan waktu yang relative singkat dan usaha yang minimum. Ada banyak metode – metode perancangan kasus uji yang telah berkembang guna menyediakan pendekatan pengujian secara sistematis bagi para pengembang. Metode perancangan kasus uji ini juga menyediakan mekanisme yang dapat membantu kelengkapan dari pengujian dan memberikan kemungkinan tertinggi untuk dapat menemukan berbagai kesalahan dalam sebuah sistem atau perangkat lunak (Pressman, 2010).

2.2.11.1 Pengujian *White Box*

Pengujian *white box* adalah pengujian yang didasarkan pada pengecekan terhadap detail perancangan, menggunakan struktur kontrol dari desain program secara procedural untuk membagi pengujian ke dalam beberapa kasus pengujian (Graham, 2009). Penentuan kasus uji disesuaikan dengan struktur sistem, pengetahuan mengenai program digunakan untuk mengidentifikasi kasus uji tambahan.

2.2.11.2 Pengujian *Black Box*

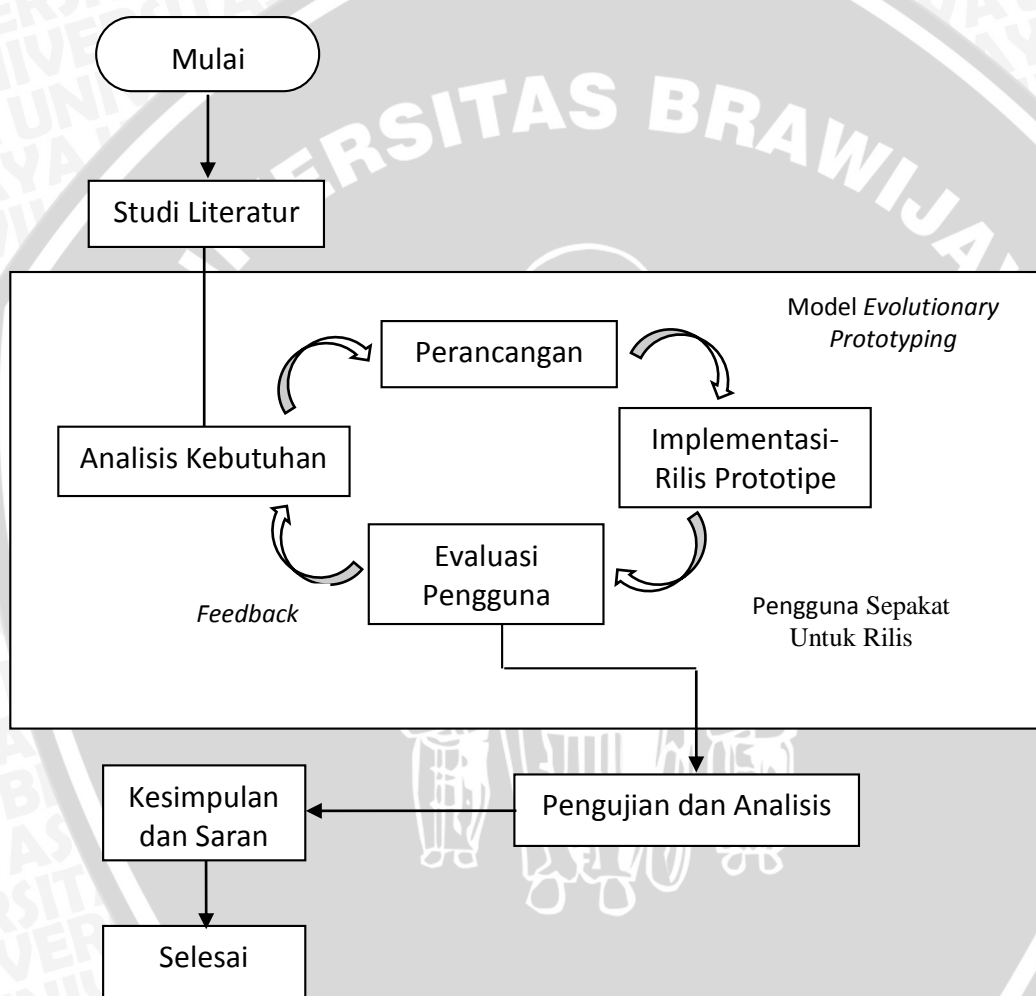
Pengujian *black box* digunakan untuk menguji fungsi-fungsi khusus dari perangkat lunak yang dirancang. Pada teknik ini, kebenaran perangkat lunak yang diuji hanya dilihat berdasarkan keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan untuk fungsi yang ada tanpa melihat bagaimana proses untuk mendapatkan keluaran tersebut. Dari keluaran yang dihasilkan, kemampuan program dalam memenuhi kebutuhan pemakai dapat diukur sekaligus dapat diketahui kesalahan-kesalahannya. Beberapa jenis kesalahan yang dapat diidentifikasi :

1. Fungsi tidak benar atau hilang
2. Kesalahan antar muka
3. Kesalahan pada struktur data (pengaksesan basis data)
4. Kesalahan inisialisasi dan akhir program
5. Kesalahan performansi

Walaupun sulit untuk menelusuri kesalahan yang mungkin didapat, teknik pengujian *black box* lebih sering dipilih untuk menguji perangkat lunak karena kemudahan dalam pelaksanaannya (Simarmata, 2010).

BAB 3 METODOLOGI PENELITIAN

Bab ini akan menjelaskan tentang langkah-langkah yang dilakukan dalam melakukan penelitian. Penelitian ini menerapkan model *evolutionary prototyping* pada perangkat lunak yang akan dibangun. Langkah-langkah yang dilakukan dalam penelitian ini dimulai dari studi literatur, analisis kebutuhan, perancangan sistem, implementasi-rilis *prototype*, evaluasi *prototype* oleh pengguna, pengujian serta pengambilan kesimpulan dan saran. Diagram alir langkah-langkah dalam penelitian ini ditunjukkan pada **Gambar 3.1**.



Gambar 3.1 Diagram Alir Metodologi Penelitian

Pelaksanaan penelitian ini dibagi menjadi tujuh tahap. Tahap pertama yaitu studi literatur, kemudian dilanjutkan tahapan yang merupakan metode dari *evolutionary prototyping*. Yang merupakan tahap *evolutionary prototyping* yaitu analisis kebutuhan, perancangan, implementasi-rilis *prototype* dan evaluasi pengguna. Tahapan pada metode *evolutionary prototyping* pada kasus ini dilakukan perulangan sebanyak 3 kali. Kemudian dilanjutkan dengan tahap pengujian, dan diakhiri dengan tahap kesimpulan dan saran dari hasil penelitian.

3.1 Studi Literatur

Studi literatur menjelaskan kajian pustaka dan dasar teori yang digunakan dalam proses pembangunan sistem dan penyusunan skripsi. Dalam studi literatur dikumpulkan data-data dari sumber yang terpercaya yang kemudian akan diimplementasikan lebih lanjut pada aplikasi yang dibuat. Dalam hal ini, studi literatur yang digunakan adalah:

- Pengertian Perpustakaan
- Perpustakaan Universitas Brawijaya
- Pengertian Rekayasa Perangkat Lunak
- Pengertian Prototipe
- Model *Prototyping*
- Fase *Evolutionary prototyping*
- *Web Service*
- *Unified Modelling Language (UML)*
 - *Diagram Use Case*
 - *Diagram Sequence*
 - *Diagram Class*
- Pengertian basis data
- Model Data Relasional
- Framework
 - Framework YII
- Konsep Pengujian Perangkat Lunak
 - Pengujian *white box*
 - Pengujian *black box*

3.2 Analisis Kebutuhan

Analisis kebutuhan atau tahapan mendengarkan pelanggan merupakan tahap yang paling penting dalam suatu pengembangan perangkat lunak, sebelum melakukan perancangan dan pembuatan perangkat lunak. Langkah-langkah yang dilakukan mulai dari wawancara secara langsung kepada pihak perpustakaan Universitas Brawijaya, dilanjutkan dengan melakukan survey dan analisis terhadap sistem yang sudah ada sebelumnya. Kemudian menentukan gambaran umum sistem, identifikasi aktor-aktor yang terlibat dalam sistem, analisis data yang akan disimpan, penjabaran tentang daftar kebutuhan dan kemudian memodelkannya kedalam diagram *use case*. Analisis kebutuhan ini bertujuan untuk

menggambarkan kebutuhan-kebutuhan yang harus disediakan oleh sistem agar dapat memenuhi kebutuhan pengguna.

3.3 Perancangan Desain Sistem

Pada tahap ini akan melakukan perancangan berdasarkan kebutuhan yang sudah didefinisikan sebelumnya. Pertama kali pada tahap ini akan menghasilkan perancangan awal, kemudian perancangan iterasi 1 dan seterusnya. Pada kasus ini akan terus diulang kembali sampai iterasi ketiga. Perancangan yang dibuat dapat dimanfaatkan kembali untuk membangun software lebih cepat. Tahap perancangan ini merupakan proses menterjemahkan kebutuhan ke dalam sebuah model perangkat lunak seperti perancangan diagram berdasarkan *Object Oriented Analysis* dan *Object Oriented Design* yaitu menggunakan pemodelan UML (*Unified Modeling Language*) seperti *sequence diagram*, *class diagram*, serta perancangan *user interface* dan model data relasional yang akan dibuat sesuai dengan kebutuhan pengguna.

3.4 Implementasi Sistem

Setelah rancangan perangkat lunak maka langkah selanjutnya dilakukan adalah implementasi *prototype*. Pada tahap ini, melakukan kegiatan pengkodean atau yang biasa dikenal dengan proses *coding*, yaitu proses penterjemahan sistem kedalam kode program dengan bahasa pemrograman. Implementasi yang akan dibuat berupa aplikasi web terintegrasi pada perpustakaan Universitas Brawijaya. Implementasi perangkat lunak dilakukan dengan menggunakan implementasi *basis data* SQL Server 2008 dan bahasa pemrograman PHP dengan framework Yii serta diintegrasikan dengan *web service* SIAKAD UB yang telah disediakan Pengkajian dan Pengembangan Teknologi Informasi Universitas Brawijaya. Pada tahap ini akan menghasilkan *prototype* versi 0.0 untuk perancangan pertama kali, setelah dilakukan iterasi atau evaluasi maka menghasilkan implementasi *prototype* versi 1.0 hingga menghasilkan implementasi rilis sistem.

3.5 Evaluasi Pengguna

Pada tahap ini proses evaluasi dilakukan oleh pengguna. Pengguna dapat menilai apakah rancangan awal yang dibangun telah sesuai dengan yang diharapkan. Jika tidak, maka dilakukan revisi dan *feedback* dengan menyesuaikan kebutuhan yang diinginkan pengguna. Maka dengan adanya proses ini diharapkan perangkat lunak nantinya akan sesuai dengan kebutuhan. Kemudian dilakukan review kekurangan-kekurangan dari kebutuhan pelanggan. Pengembang kemudian kembali mendengarkan keluhan dari pelanggan untuk memperbaiki sistem yang ada. Setelah proses ini dilakukan maka akan dilakukan redesain perancangan sistem sesuai dengan hasil evaluasi.

3.6 Pengujian dan Analisis

Tahapan ini berfungsi untuk melakukan pengujian kualitas perangkat lunak yang telah jadi. Pengujian dilakukan untuk mengetahui apakah sistem yang

dibangun telah sesuai dengan spesifikasi dari kebutuhan sistem. Pengujian yang dilakukan terdiri pengujian unit, pengujian integrasi, pengujian validasi serta pengujian *web service* dan pengujian performansi.

Pengujian unit dilakukan untuk memastikan bahwa unit-unit memiliki hasil implementasi yang sesuai dengan harapan diawal analisis kebutuhan. Teknik pengujian yang digunakan adalah teknik pengujian *white box* dengan jenis pengujian *basis path*. Teknik pengujian ini dilakukan dengan menggunakan rancangan atau kode program sebagai dasar untuk membuat *flow graph*. Setelah itu, berdasarkan *flow graph* yang telah dibuat akan ditentukan kompleksitas siklomatik serta himpunan basis dari jalur-jalur independen secara linier.

Pengujian integrasi dilakukan dengan menggunakan teknik pengujian *Black Box* dengan pendekatan *Big-Bang*. Proses pengujian dilakukan dengan menguji *class* yang saling berinteraksi. Diawali dengan membuat kasus uji yang terdiri dari rincian masukan dan hasil yang diharapkan. Kemudian menjalankan sistem sesuai kasus uji dan membandingkan antara hasil sebenarnya dengan hasil yang diharapkan.

Pengujian validasi dilakukan dengan menggunakan teknik pengujian *black-box*. Pendekatan yang dilakukan adalah dengan menguji sistem hasil implementasi berdasarkan spesifikasi kebutuhan sistem. Skenario pada pengujian ini dilakukan dengan menjalankan sistem sesuai dengan fungsi-fungsi internal dengan mencoba fitur pada sistem minimal 3 kali percobaan. Jika dari minimal 3 kali percobaan sistem berjalan tanpa ada kesalahan maka pengujian dianggap valid.

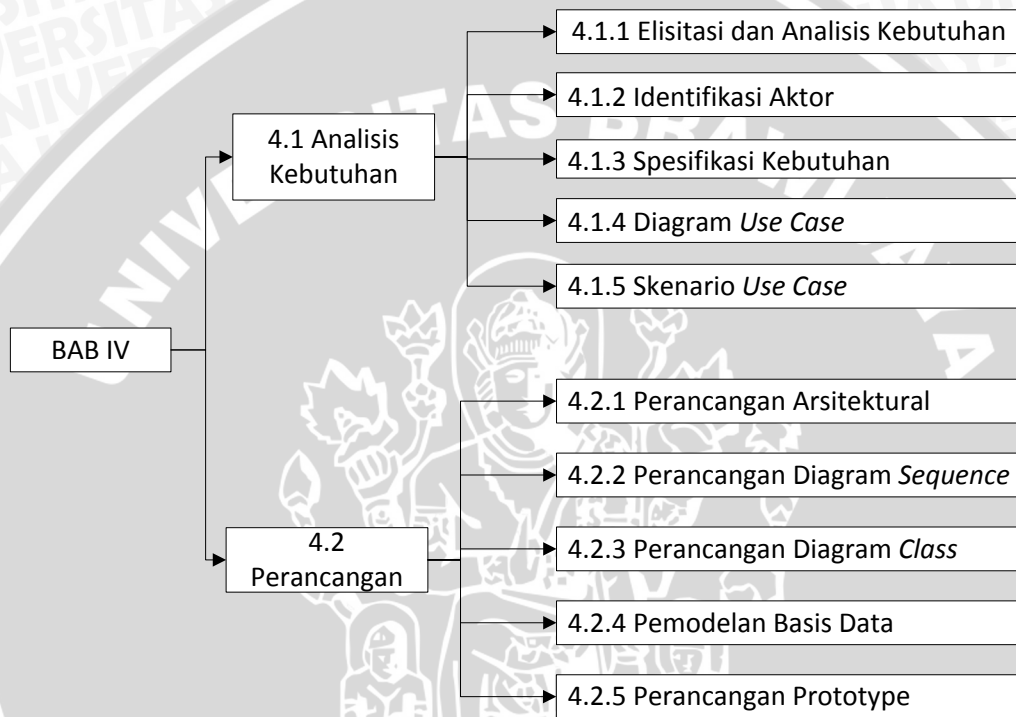
Pengujian *web service* dilakukan menggunakan metode *black box*. Sedangkan pengujian performansi dilakukan dengan menghitung waktu proses yang dibutuhkan untuk melakukan permintaan data ke *web service*.

3.7 Kesimpulan dan saran

Pengambilan kesimpulan dilakukan setelah semua tahapan perancangan, implementasi dan pengujian sistem aplikasi telah selesai dilakukan dan didasarkan pada kesesuaian antara teori dan praktik. Kesimpulan diambil dari hasil pengujian dan analisis terhadap sistem yang dibangun. Tahap terakhir dari penulisan ini adalah saran yang dimaksudkan untuk memperbaiki kesalahan-kesalahan yang terjadi dan menyempurnakan penulisan serta untuk memberikan pertimbangan atas pengembangan sistem informasi selanjutnya.

BAB 4 ANALISIS DAN PERANCANGAN

Bab ini akan membahas mengenai analisis kebutuhan dan perancangan *Aplikasi Web Terintegrasi Perpustakaan UB*. Pada tahap pertama yang akan dilakukan yaitu proses analisis kebutuhan dan tahap kedua proses perancangan perangkat lunak. Perancangan perangkat lunak pada skripsi ini menggunakan pendekatan desain berorientasi objek yang direpresentasikan dengan menggunakan UML (*Unified Modelling Language*). Untuk diagram blok bab analisis & perancangan dijelaskan pada **Gambar 4.1**.



Gambar 4.1 Diagram Blok Bab Analisis dan Perancangan

Pada skripsi ini menggunakan model *evolutionary prototyping* sehingga analisis kebutuhan dan perancangan dilakukan lebih dari satu kali iterasi. Interaksi yang akan dilakukan pada pembuatan sistem ini sebanyak 3 kali iterasi, pada bab ini akan dituliskan analisis kebutuhan dan perancangan iterasi yang terakhir yaitu iterasi ke-3. Sedangkan iterasi ke-0 sampai iterasi ke-2 dijelaskan pada lampiran bab proses iterasi *prototyping*.

4.1 Analisis Kebutuhan

Analisis kebutuhan merupakan tahapan yang pertama kali harus dilakukan sebelum melakukan perancangan dan pembuatan perangkat lunak. Tahap analisis kebutuhan pada penelitian ini diawali dengan elisitasi kebutuhan, identifikasi aktor yang terlibat dalam sistem, pendefinisian daftar spesifikasi kebutuhan yang kemudian ditranslasikan ke dalam diagram *use case*. Tahapan ini dilakukan untuk menggambarkan kebutuhan-kebutuhan yang dimiliki oleh sistem.

4.1.1 Elisitasi Kebutuhan

Tahap ini dilakukan untuk mengetahui masalah yang perlu dipecahkan dan mengenali pemangku kepentingan dari sistem. Langkah-langkah yang dilakukan yaitu, mulai dari wawancara secara langsung kepada staf pengolahan dan staf IT perpustakaan Universitas Brawijaya, kemudian dilanjutkan dengan melakukan pengamatan terhadap sistem yang sudah ada sebelumnya.

Berdasarkan hasil dari wawancara (Lampiran A) dan pengamatan terhadap sistem yang sudah pernah ada sebelumnya maka disimpulkan bahwa perangkat lunak yang akan dibangun meliputi:

1. Permintaan bahan pustaka

Sistem dapat menampung permintaan bahan pustaka baru yang meliputi buku dan jurnal. Permintaan bahan pustaka baru dapat dilakukan oleh mahasiswa selaku anggota perpustakaan serta bagian ruang baca dan pengolahan. Mahasiswa dapat mengajukan permintaan bahan pustaka baru dengan cara masuk kedalam sistem dengan menggunakan akun SIAKAD UB. Sehingga tidak perlu melakukan registrasi kedalam sistem karena sistem terintegrasi dengan SIAKAD UB. Integrasi sistem dilakukan dengan menggunakan *web service* dengan format XML. Sedangkan bagian ruang baca dan pengolahan ketika mengajukan permintaan bahan pustaka sama halnya dengan mahasiswa harus masuk kedalam sistem, namun disini bagian ruang baca melakukan registrasi untuk mendapatkan akun agar dapat bisa masuk kedalam sistem. Registrasi dilakukan dengan cara mengajukan akun baru kepada pihak perpustakaan pusat, kemudian bagian staf IT perpustakaan pusat akan membuatkan akun baru untuk ruang baca yang mengajukan.

2. Pengolahan Data Staf

Sistem dapat mengolah data staf yang terlibat dalam sistem. Pengolahan staf meliputi daftar staf, tambah staf dan edit staf. Sistem dapat menampilkan daftar staf yang sudah didaftarkan pada sistem. Penambahan staf dilakukan ketika ada staf baru. Edit staf dilakukan ketika data staf mengalami kekeliruan atau memang akan dilakukan perubahan pada data staf. Pengolahan data staf dilakukan oleh staf IT. Staf IT merupakan staf yang dapat mengakses langsung sistem tanpa harus didaftarkan, karena secara langsung sudah terdaftar pada sistem.

3. Pengolahan Permintaan Bahan Pustaka

Sistem dapat mengolah data permintaan bahan pustaka untuk dijadikan dasar melakukan pengadaan bahan pustaka. Dari keseluruhan data pengajuan bahan pustaka baru yang sudah tertampung, akan dilakukan seleksi oleh pihak pengolahan perpustakaan. Hasil seleksi akan masuk pada bagian pengadaan perpustakaan dan akan diproses lebih lanjut. Sehingga dengan adanya pendataan bahan pustaka baru ini akan menambah koleksi perpustakaan yang sesuai dengan kebutuhan mahasiswa. Bagian pengolahan perpustakaan pusat dapat masuk

kedalam sistem dengan akun yang telah didaftarkan oleh staf IT. Jadi antara bagian permintaan bahan pustaka dengan pengolahan permintaan bahan pustaka saling terintegrasi.

4. Pengolahan Bahan Pustaka

Sistem dapat melakukan pengolahan bahan pustaka. Proses pengolahan dimulai dari *entry* data baru dan data *copy*. Data bahan pustaka yang baru dimasukkan akan diberi label berah sedangkan data *copy* merupakan data bahan pustaka yang sudah pernah dimasukkan. Bahan pustaka *copy* akan diberi label putih. Setelah data bahan pustaka dimasukkan kedalam sistem, kemudian mencetak label dan barcode untuk bahan pustaka. Setelah diproses register dan pelabelan, akan dilakukan validasi bahan pustaka. Bahan pustaka yang sudah divalidasi sudah dapat digunakan. Proses pengolahan bahan pustaka disetiap ruang baca akan sama dan semua data akan dipusatkan pada perpustakaan pusat. Pemusatan data dilakukan untuk mempermudah proses pencarian bahan pustaka. Dengan data yang terpusat semua bisa diakses secara langsung, sedangkan data yang diintegrasikan dengan *web service* disetiap ruang baca akan lebih membutuhkan waktu dan proses pengerjaan yang lebih lama.

Dari penjabaran sistem diatas, maka dapat ditemukan kebutuhan yang harus ada dalam sistem. Kebutuhan yang harus ada dapat dilihat pada **Tabel 4.1**.

Tabel 4.1 Kebutuhan perangkat lunak secara umum

Tipe Kebutuhan	Kebutuhan
Fungsional	Perangkat lunak harus mampu menyediakan fasilitas login menggunakan akun SIAKAD (Sistem Informasi Akademik) bagi mahasiswa UB dan akun yang terdaftar bagi staf perpustakaan.
Inversi	Sistem tidak boleh dioperasikan oleh pengguna yang tidak memiliki otoritas.
Fungsional	Perangkat lunak harus mampu menyediakan fasilitas untuk mengelola biodata anggota.
Inversi	Sistem tidak boleh menghapus data anggota.
Fungsional	Perangkat lunak harus mampu menyediakan fasilitas untuk mengelola operator / staf.
Fungsional	Perangkat lunak harus mampu menyediakan fasilitas untuk mengelola permintaan bahan pustaka yang meliputi buku dan jurnal.
Fungsional	Perangkat lunak harus mampu menyediakan fasilitas untuk mengelola bahan pustaka yang meliputi buku dan jurnal.

Tabel 4.1 (Lanjutan)

Tipe Kebutuhan	Kebutuhan
Fungsional	Perangkat lunak harus mampu menambahkan bahan pustaka yang sudah pernah ada sebelumnya.
Fungsional	Perangkat lunak harus mampu menyediakan fasilitas mencetak barcode dan label.
Fungsional	Perangkat lunak harus mampu menyediakan fasilitas keluar dari sistem (<i>logout</i>)
<i>Implementation constraints</i>	Barcode yang digunakan harus menggunakan tipe <i>code39</i> .
<i>Implementation constraints</i>	Sistem harus diimplementasikan dengan PHP dan Framework Yii.
Non Fungsional	Perangkat lunak dapat berjalan dengan waktu tidak melebihi 30 detik.

4.1.2 Identifikasi Aktor

Pada tahap ini dilakukan proses identifikasi terhadap aktor-aktor yang berinteraksi dengan perangkat lunak yang dibangun. Pada identifikasi awal, aktor yang berinteraksi dengan sistem terdapat 4 aktor yaitu user tamu, *administrator*, staf dan anggota perpustakaan. Kemudian setelah dilakukan evaluasi atau iterasi ke-1, terjadi perubahan aktor, yaitu aktor administrator menjadi staf IT serta staf menjadi staf pengolahan dan staf ruang baca. Sedangkan pada evaluasi selanjutnya tidak mengalami perubahan pada aktor. **Tabel 4.2** memperlihatkan aktor-aktor yang terlibat serta penjelasan peran masing-masing aktor dalam sistem setelah dilakukan evaluasi sampai iterasi terakhir.

Tabel 4.2 Identifikasi aktor iterasi ke-3

Aktor	Deskripsi
User Tamu	User Tamu merupakan user yang dapat melihat halaman awal sistem dan melakukan login sistem.
Staf IT	Staf IT merupakan staf bagian IT di perpustakaan pusat. Staf IT berperan sebagai pengguna aplikasi yang memiliki hak atas fitur pengolahan staf.
Staf Pengolahan	Staf Pengolahan merupakan staf yang berada pada perpustakaan pusat. Staf pengolahan berperan sebagai pengguna aplikasi yang memiliki hak atas fitur pengolahan permintaan bahan pustakan dan pengolahan bahan pustaka.

Tabel 4.2 (Lanjutan)

Staf Ruang Baca	Staf Ruang Baca merupakan staf yang berada di ruang baca setiap fakultas. Staf ruang baca berperan sebagai pengguna aplikasi yang memiliki hak atas fitur pengolahan bahan pustaka.
Anggota Perpustakaan	Anggota perpustakaan merupakan mahasiswa yang berperan sebagai pengguna aplikasi yang dapat mengakses fitur permintaan bahan pustaka.

4.1.3 Spesifikasi Kebutuhan

Spesifikasi kebutuhan yang disediakan pada sistem ini mengacu kepada elisitasi kebutuhan pada bagian sebelumnya. Penentuan prioritas pada daftar spesifikasi kebutuhan fungsional didasarkan atas tingkat keterlibatannya terhadap proses bisnis utama yang telah diusulkan pada tahap elisitasi kebutuhan. Pada penentuan spesifikasi kebutuhan diawal memiliki spesifikasi kebutuhan sebanyak 44 spesifikasi kebutuhan fungsional dan 2 spesifikasi kebutuhan non-fungsional. Kemudian setelah dilakukan evaluasi atau iterasi ke-1, spesifikasi kebutuhan fungsional bertambah 12 kebutuhan fungsional, dan tidak terjadi perubahan pada spesifikasi kebutuhan non-fungsional. Pada evaluasi ke-2 tidak mengalami perubahan kebutuhan fungsional, hanya mengalami perubahan hak akses aktor terhadap fungsionalnya. Sedangkan pada evaluasi terakhir atau iterasi ke-3 terjadi penambahan spesifikasi fungsional sebanyak 6 fungsional dan spesifikasi kebutuhan non fungsional tidak mengalami perubahan. **Tabel 4.3** menjelaskan spesifikasi kebutuhan fungsional setelah dilakukan evaluasi atau iterasi ke-1 sampai iterasi ke-3. **Tabel 4.4** menjelaskan spesifikasi kebutuhan non fungsional dari sistem setelah dilakukan evaluasi atau iterasi ke-1 sampai iterasi ke-3.

Tabel 4.3 Spesifikasi kebutuhan fungsional iterasi ke-3

Kode SKPL	Deskripsi	Use Case	Aktor	Prioritas
SKPL-01-01	Perangkat lunak harus mampu menyediakan fasilitas autentifikasi menggunakan akun SIAKAD (Sistem Informasi Akademik) dan menggunakan username dan password yang sudah terdaftar.	Login	User Tamu	Tinggi
SKPL-02-01	Perangkat lunak mampu menyediakan fasilitas untuk melihat biodata user yang melakukan login.	Lihat detail biodata	Staf Pengolahan, staf ruang baca dan Anggota	Rendah

Tabel 4.3 (Lanjutan)

Kode SKPL	Deskripsi	Use Case	Aktor	Prioritas
SKPL-02-02	Perangkat lunak mampu menyediakan fasilitas ubah biodata agar user yang melakukan login dapat merubah biodata jika terjadi kesalahan.	Ubah biodata	Staf Pengolahan, Staf ruang baca dan Anggota	Sedang
SKPL-03-01	Perangkat lunak mampu menyediakan fasilitas untuk menambah staf baru.	Tambah staf	Staf IT	Tinggi
SKPL-03-02	Perangkat lunak dapat menyediakan fasilitas ubah data staf ruang baca sehingga data dapat dirubah jika terjadi kesalahan.	Ubah staf	Staf IT	Sedang
SKPL-03-03	Perangkat lunak dapat menyediakan fasilitas untuk melihaf daftar staf yang terdaftar.	Daftar staf	Staf IT	Sedang
SKPL-04-01	Perangkat lunak dapat menyediakan fasilitas tambah permintaan bahan pustaka buku.	Tambah permintaan buku	Anggota, staf ruang baca dan staf pengolahan	Tinggi
SKPL-04-02	Perangkat lunak dapat menyediakan fasilitas unduh form permintaan bahan pustaka buku.	Unduh form permintaan buku	Staf ruang baca dan staf pengolahan	Rendah
SKPL-04-03	Perangkat lunak dapat menyediakan fasilitas unggah form untuk menambah permintaan bahan pustaka buku secara kolektif.	Unggah form permintaan buku	Staf ruang baca dan staf pengolahan	Rendah
SKPL-04-04	Perangkat lunak dapat menyediakan fasilitas lihat daftar permintaan bahan pustaka buku yang diajukan.	Lihat daftar permintaan buku	Staf pengolahan, staf ruang baca dan anggota	Sedang

Tabel 4.3 (Lanjutan)

Kode SKPL	Deskripsi	Use Case	Aktor	Prioritas
SKPL-04-05	Perangkat lunak dapat menyediakan fasilitas lihat daftar permintaan bahan pustaka buku yang sudah disetujui.	Lihat daftar permintaan buku disetujui	Staf pengolahan	Sedang
SKPL-04-06	Perangkat lunak dapat menyediakan fasilitas lihat daftar permintaan bahan pustaka buku yang belum disetujui.	Lihat daftar permintaan buku belum disetujui	Staf pengolahan	Sedang
SKPL-04-07	Perangkat lunak dapat menyediakan fasilitas yang dapat menunduh data permintaan bahan pustaka buku yang sudah disetujui.	Unduh daftar permintaan buku disetujui	Staf pengolahan	Rendah
SKPL-04-08	Perangkat lunak dapat menyediakan fasilitas yang dapat menunduh data permintaan bahan pustaka buku yang belum disetujui.	Unduh daftar permintaan buku belum disetujui	Staf pengolahan	Rendah
SKPL-04-09	Perangkat lunak dapat menyediakan fasilitas yang dapat melihat detail data permintaan bahan pustaka buku.	Lihat detail permintaan buku	Staf pengolahan	Rendah
SKPL-04-10	Perangkat lunak dapat menyediakan fasilitas yang dapat mengubah data permintaan bahan pustaka buku.	Ubah permintaan buku	Staf pengolahan	Sedang
SKPL-04-11	Perangkat lunak dapat menyediakan fasilitas yang dapat menghapus data permintaan bahan pustaka buku.	Hapus permintaan buku	Staf pengolahan	Rendah

Tabel 4.3 (Lanjutan)

Kode SKPL	Deskripsi	Use Case	Aktor	Prioritas
SKPL-04-12	Perangkat lunak dapat menyediakan fasilitas yang dapat mencari data permintaan bahan pustaka buku yang belum disetujui.	Cari permintaan buku belum disetujui	Staf pengolahan	Rendah
SKPL-04-13	Perangkat lunak dapat menyediakan fasilitas untuk melakukan tambah bahan pustakan buku baru dari list permintaan buku.	Tambah buku dari permintaan	Staf pengolahan	Tinggi
SKPL-04-14	Perangkat lunak dapat menyediakan fasilitas tambah permintaan bahan pustaka jurnal.	Tambah permintaan jurnal	Anggota dan staf ruang baca	Tinggi
SKPL-04-15	Perangkat lunak dapat menyediakan fasilitas unduh form permintaan bahan pustaka jurnal.	Unduh form permintaan jurnal	Staf ruang baca dan Staf pengolahan	Rendah
SKPL-04-16	Perangkat lunak dapat menyediakan fasilitas unggah form untuk menambah permintaan bahan pustaka jurnal secara kolektif.	Unggah form permintaan jurnal	Staf ruang baca dan Staf pengolahan	Rendah
SKPL-04-17	Perangkat lunak dapat menyediakan fasilitas lihat daftar permintaan bahan pustaka jurnal yang diajukan.	Lihat daftar permintaan jurnal	Anggota dan staf ruang baca	Sedang
SKPL-04-18	Perangkat lunak dapat menyediakan fasilitas lihat daftar permintaan bahan pustaka jurnal.	Lihat daftar permintaan jurnal disetujui	Staf pengolahan	Sedang
SKPL-04-19	Perangkat lunak dapat menyediakan fasilitas lihat daftar permintaan bahan pustaka jurnal.	Lihat daftar permintaan jurnal belum disetujui	Staf pengolahan	Sedang

Tabel 4.3 (Lanjutan)

Kode SKPL	Deskripsi	Use Case	Aktor	Prioritas
SKPL-04-20	Perangkat lunak dapat menyediakan fasilitas yang dapat menunduh data permintaan bahan pustaka jurnal yang sudah disetujui.	Unduh daftar permintaan jurnal disetujui	Staf pengolahan	Rendah
SKPL-04-21	Perangkat lunak dapat menyediakan fasilitas yang dapat menunduh data permintaan bahan pustaka jurnal yang belum disetujui.	Unduh daftar permintaan jurnal belum disetujui	Staf pengolahan	Rendah
SKPL-04-22	Perangkat lunak dapat menyediakan fasilitas yang dapat melihat detail data permintaan bahan pustaka jurnal.	Lihat detail permintaan jurnal	Staf pengolahan	Rendah
SKPL-04-23	Perangkat lunak dapat menyediakan fasilitas yang dapat mengubah data permintaan bahan pustaka jurnal.	Ubah permintaan jurnal	Staf pengolahan	Sedang
SKPL-04-24	Perangkat lunak dapat menyediakan fasilitas yang dapat menghapus data permintaan bahan pustaka jurnal.	Hapus permintaan jurnal	Staf pengolahan	Rendah
SKPL-04-25	Perangkat lunak dapat menyediakan fasilitas yang dapat mencari data permintaan bahan pustaka jurnal yang belum disetujui.	Cari permintaan jurnal belum disetujui	Staf pengolahan	Rendah
SKPL-04-26	Perangkat lunak dapat menyediakan fasilitas untuk melakukan tambah bahan pustakan jurnal baru dari list permintaan buku.	Tambah jurnal dari permintaan	Staf pengolahan	Tinggi

Tabel 4.3 (lanjutan)

Kode SKPL	Deskripsi	Use Case	Aktor	Prioritas
SKPL-05-01	Perangkat lunak dapat menyediakan fasilitas untuk melakukan tambah bahan pustakan buku baru / buku label merah.	Tambah buku baru	Staf ruang baca dan Staf pengolahan	Tinggi
SKPL-05-02	Perangkat lunak dapat menyediakan fasilitas unduh form tambah koleksi bahan pustaka buku.	Unduh form tambah buku	Staf ruang baca dan Staf pengolahan	Rendah
SKPL-05-03	Perangkat lunak dapat menyediakan fasilitas unggah form tambah buku untuk menambah bahan pustaka buku secara kolektif.	Unggah form tambah buku	Staf ruang baca dan Staf pengolahan	Rendah
SKPL-05-04	Perangkat lunak dapat menyediakan fasilitas untuk melihat daftar permintaan bahan pustaka buku yang berlabel merah.	Lihan daftar buku label merah	Staf ruang baca dan staf pengolahan	Sedang
SKPL-05-05	Perangkat lunak dapat menyediakan fasilitas yang dapat melakukan pencarian bahan pustaka buku label merah berdasarkan judul, register, isbn dan penerbit.	Cari buku label merah	Staf ruang baca dan staf pengolahan	Rendah
SKPL-05-06	Perangkat lunak dapat menyediakan fasilitas yang dapat melihat detail bahan pustaka buku yang berlabel merah.	Lihat detail buku label merah	Staf ruang baca dan staf pengolahan	Rendah
SKPL-05-07	Perangkat lunak dapat menyediakan fasilitas yang dapat mengubah data bahan pustaka buku berlabel merah.	Ubah buku label merah	Staf ruang baca dan staf pengolahan	Sedang
SKPL-05-08	Perangkat lunak dapat menyediakan fasilitas untuk melakukan tambah bahan pustakan jurnal baru / jurnal label merah.	Tambah jurnal baru	Staf ruang baca dan Staf pengolahan	Tinggi

Tabel 4.3 (lanjutan)

Kode SKPL	Deskripsi	Use Case	Aktor	Prioritas
SKPL-05-09	Perangkat lunak dapat menyediakan fasilitas unduh form tambah koleksi bahan pustaka jurnal.	Unduh form tambah jurnal	Staf ruang baca dan Staf pengolahan	Rendah
SKPL-05-10	Perangkat lunak dapat menyediakan fasilitas unggah form tambah jurnal untuk menambah bahan pustaka jurnal secara kolektif.	Unggah form tambah jurnal	Staf ruang baca dan Staf pengolahan	Rendah
SKPL-05-11	Perangkat lunak dapat menyediakan fasilitas untuk melihat daftar permintaan bahan pustaka jurnal yang berlabel merah.	Lihan daftar jurnal label merah	Staf ruang baca dan staf pengolahan	Sedang
SKPL-05-12	Perangkat lunak dapat menyediakan fasilitas yang dapat melakukan pencarian bahan pustaka jurnal label merah berdasarkan judul, register, isbn dan penerbit.	Cari jurnal label merah	Staf ruang baca dan staf pengolahan	Rendah
SKPL-05-13	Perangkat lunak dapat menyediakan fasilitas yang dapat melihat detail bahan pustaka jurnal yang berlabel merah.	Lihat detail jurnal label merah	Staf ruang baca dan staf pengolahan	Rendah
SKPL-05-14	Perangkat lunak dapat menyediakan fasilitas yang dapat mengubah data bahan pustaka jurnal berlabel merah.	Ubah jurnal label merah	Staf ruang baca dan staf pengolahan	Sedang
SKPL-06-01	Perangkat lunak dapat menyediakan fasilitas untuk melakukan tambah bahan pustakan buku yang sudah pernah ada sebelumnya / buku label putih.	Tambah buku label putih	Staf ruang baca dan staf pengolahan	Tinggi

Tabel 4.3 (lanjutan)

Kode SKPL	Deskripsi	Use Case	Aktor	Prioritas
SKPL-06-02	Perangkat lunak dapat menyediakan fasilitas untuk melihat daftar permintaan bahan pustaka buku yang berlabel putih.	Lihan daftar buku label putih	Staf ruang baca dan staf pengolahan	Sedang
SKPL-06-03	Perangkat lunak dapat menyediakan fasilitas yang dapat melakukan pencarian bahan pustaka buku label putih berdasarkan judul, register, isbn dan penerbit.	Cari buku label putih	Staf ruang baca dan staf pengolahan	Rendah
SKPL-06-04	Perangkat lunak dapat menyediakan fasilitas yang dapat mengubah data bahan pustaka jurnal berlabel merah.	Ubah buku label putih	Staf ruang baca dan staf pengolahan	Sedang
SKPL-06-05	Perangkat lunak dapat menyediakan fasilitas yang dapat melihat detail bahan pustaka buku yang berlabel putih.	Lihat detail buku label putih	Staf ruang baca dan staf pengolahan	Rendah
SKPL-06-06	Perangkat lunak dapat menyediakan fasilitas untuk melakukan tambah bahan pustakan jurnal yang sudah pernah ada sebelumnya / jurnal label putih..	Tambah jurnal label putih	Staf ruang baca dan staf pengolahan	Tinggi
SKPL-06-07	Perangkat lunak dapat menyediakan fasilitas untuk melihat daftar permintaan bahan pustaka jurnal yang berlabel putih.	Lihan daftar jurnal label putih	Staf ruang baca dan staf pengolahan	Sedang
SKPL-06-08	Perangkat lunak dapat menyediakan fasilitas yang dapat melakukan pencarian bahan pustaka jurnal label putih berdasarkan judul, register, isbn dan penerbit.	Cari jurnal label putih	Staf ruang baca dan staf pengolahan	Rendah

Tabel 4.3 (Lanjutan)

Kode SKPL	Deskripsi	Use Case	Aktor	Prioritas
SKPL-06-09	Perangkat lunak dapat menyediakan fasilitas yang dapat mengubah data bahan pustaka jurnal berlabel putih.	Ubah jurnal label putih	Staf ruang baca dan staf pengolahan	Sedang
SKPL-06-10	Perangkat lunak dapat menyediakan fasilitas yang dapat melihat detail bahan pustaka jurnal yang berlabel putih.	Lihat detail jurnal label putih	Staf ruang baca dan staf pengolahan	Rendah
SKPL-07-01	Perangkat lunak dapat menyediakan fasilitas yang dapat mencetak barcode dan label bahan pustaka buku label merah.	Cetak barcode dan label buku label merah	Staf ruang baca dan staf pengolahan	Tinggi
SKPL-07-02	Perangkat lunak dapat menyediakan fasilitas yang dapat mencetak barcode dan label bahan pustaka buku label putih.	Cetak barcode dan label buku label putih	Staf ruang baca dan staf pengolahan	Tinggi
SKPL-07-03	Perangkat lunak dapat menyediakan fasilitas yang dapat mencetak barcode dan label bahan pustaka jurnal label merah.	Cetak barcode dan label jurnal label merah	Staf ruang baca dan staf pengolahan	Tinggi
SKPL-07-04	Perangkat lunak dapat menyediakan fasilitas yang dapat mencetak barcode dan label bahan pustaka jurnal label putih.	Cetak barcode dan label jurnal label putih	Staf ruang baca dan staf pengolahan	Tinggi
SKPL-08-01	Perangkat lunak dapat menyediakan fasilitas yang dapat melakukan validasi bahan pustaka buku.	Validasi buku	Staf ruang baca dan staf pengolahan	Tinggi
SKPL-08-02	Perangkat lunak dapat menyediakan fasilitas yang dapat melakukan validasi bahan pustaka jurnal.	Validasi jurnal	Staf ruang baca dan staf pengolahan	Tinggi
SKPL-09-01	Perangkat lunak dapat menyediakan fasilitas <i>logout</i> setelah <i>user</i> masuk (<i>login</i>) kedalam sistem.	<i>Logout</i>	Anggota, staf IT, staf pengolahan dan staf ruang baca	Tinggi

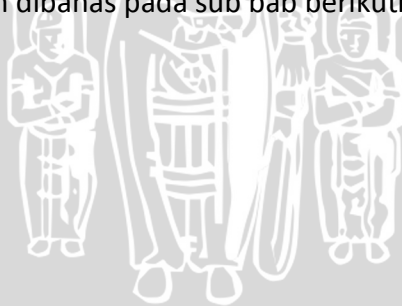
Tabel 4.4 Spesifikasi kebutuhan non-fungsional

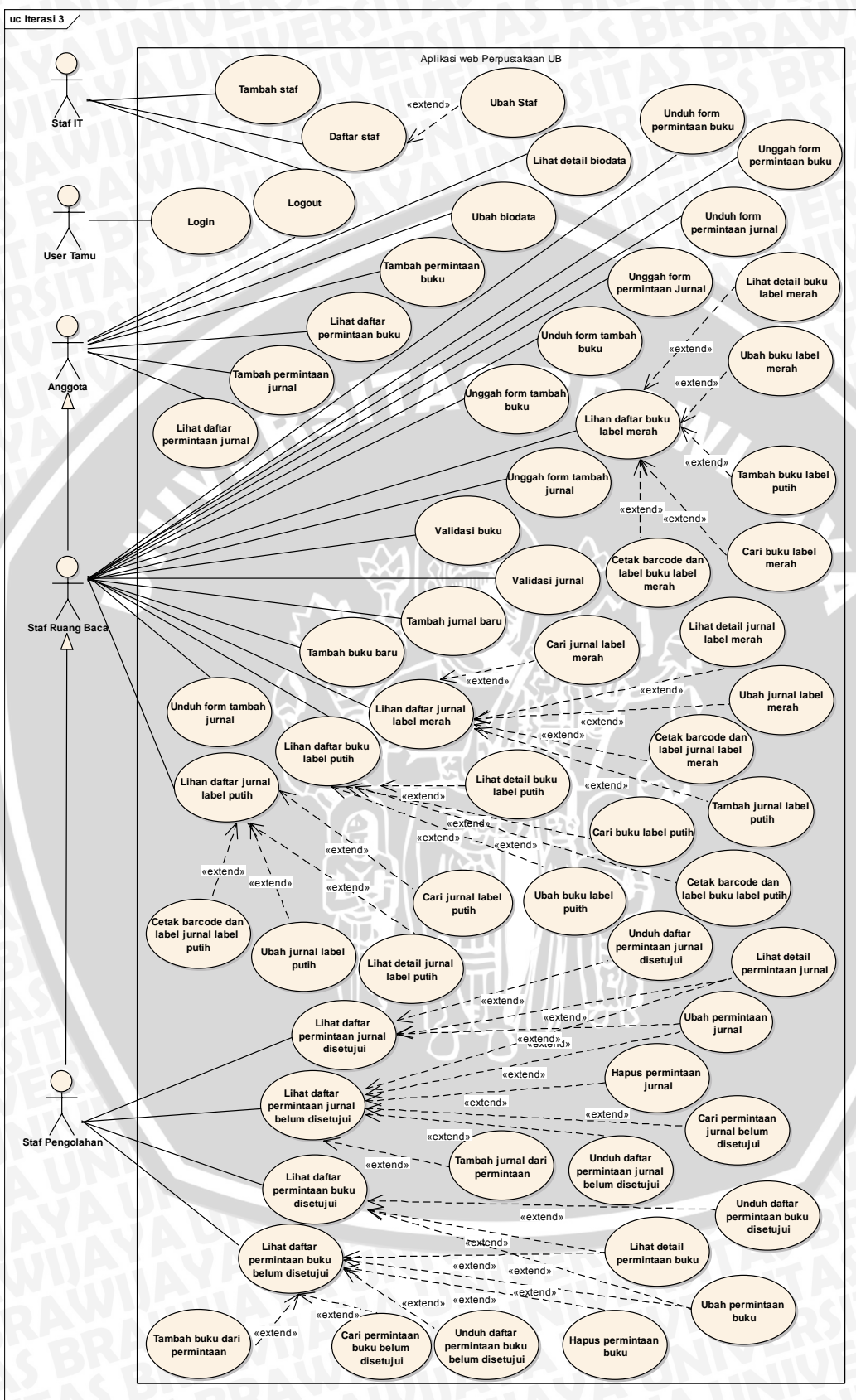
Kode SKPL	Parameter	Deskripsi Kebutuhan
SKPL-10-01	<i>Performance</i>	Proses login menggunakan akun SIAKAD UB tidak boleh lebih dari 30 detik

4.1.4 Diagram Use Case

Diagram *use case* adalah diagram yang menggambarkan fitur yang dimiliki oleh sistem. Diagram *use case* merupakan diagram yang terdiri dari *use case* dan *actor*. *Use case* merupakan fungsionalitas dari sistem yang bisa dilakukan oleh *user*, sedangkan *actor* merupakan *user* yang menjalankan sistem. Diagram *use case* perangkat lunak dirancang berdasarkan kegiatan yang dapat dikerjakan oleh setiap *actor* terhadap perangkat lunak.

Berdasarkan fungsionalitas yang sudah ditentukan sebelumnya pada tahap awal memiliki 43 *use case* dan 4 aktor. Kemudian setelah dilakukan evaluasi tahap awal atau iterasi ke-1, diagram *use case* mengalami perubahan. Perubahan yang terjadi yaitu bertambah 12 *use case* dan 1 aktor. Sedangkan pada evaluasi kedua hanya mengalami perubahan hak akses aktor terhadap *use case*. Selanjutnya pada evaluasi terakhir atau iterasi ke-3, *use case* bertambah 6 *use case*. **Gambar 4.2** menjelaskan diagram *use case* yang telah dilakukan evaluasi tahap akhir atau iterasi ke-3. Terdapat generalisasi pada beberapa aktor yaitu, Aktor anggota merupakan generalisasi dari staf ruang baca, dan staf ruang baca generalisasi dari staf pengolahan. Generalisasi yang dimaksud yaitu tentang hak akses dari fitur yang ada. Agar lebih jelas, maka masing-masing *use case* akan dijabarkan pada skenario *use case* yang akan dibahas pada sub bab berikutnya.





Gambar 4.2 Diagram use case diagram iterasi ke-3



4.1.5 Skenario Use Case

1. Skenario use case login

Tabel 4.5 Skenario use case login

Kode Use Case	SKPL-01-01
Nama Use Case	Login
Aktor	User Tamu
Tujuan	Untuk masuk kedalam sistem sesuai dengan hak akses aktor
Deskripsi	Use case ini menjelaskan bagaimana aktor melakukan login kedalam perangkat lunak dengan akun yang terdaftar didalam database maupun akun SIAKAD
Skenario Utama	
Kondisi Awal	Sistem menampilkan halaman utama yang terdapat form loginnya
Aksi Aktor	Tanggapan dari sistem
<ol style="list-style-type: none"> Aktor memasukkan data login (<i>username</i> dan <i>password</i>) Aktor menekan tombol login 	<ol style="list-style-type: none"> Sistem akan menerima data login, kemudian sistem melakukan pengecekan terhadap data yang dimasukkan oleh aktor. Jika data benar maka aktor dapat masuk kedalam sistem aplikasi sesuai hak akses yang diberikan
Skenario Alternatif 1: Jika <i>username</i> atau <i>password</i> kosong	
<ol style="list-style-type: none"> Aktor mengkosongkan <i>username</i> atau <i>password</i> Aktor menekan tombol login 	<ol style="list-style-type: none"> Sistem akan menampilkan pesan <i>error</i> bahwa <i>username</i> atau <i>password</i> kosong
Skenario Alternatif 2 : Jika kombinasi <i>username</i> dan <i>password</i> salah	
<ol style="list-style-type: none"> Aktor memasukkan kombinasi <i>username</i> dan <i>password</i> salah Aktor menekan tombol login 	<ol style="list-style-type: none"> Sistem akan menampilkan pesan peringatan bahwa <i>username</i> atau <i>password</i> salah
Konsisi Akhir	Sistem akan menampilkan halaman utama sistem sesuai dengan halaman masing-masing aktor

2. Skenario *use case* lihat detail biodata

Tabel 4.6 Skenario *use case* lihat detail biodata

Kode Use Case	SKPL-02-01
Nama Use Case	Lihat detail biodata
Aktor	Staf pengolahan, staf ruang baca dan anggota
Tujuan	Untuk melihat detail biodata aktor yang login
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana melihat detail biodata berdasarkan hak akses aktor yang login
Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu akun 3. Aktor memilih menu <i>profil</i>	2. Sistem akan menampilkan menu <i>dropdown profile</i> dan <i>logout</i> 4. Sistem akan menampilkan biodata aktor
Konsisi Akhir	Sistem akan menampilkan halaman biodata aktor yang melakukan login

3. Skenario *use case* ubah biodata

Tabel 4.7 Skenario *use case* ubah biodata

Kode Use Case	SKPL-02-02
Nama Use Case	Ubah biodata
Aktor	Staf pengolahan, staf ruang baca dan anggota
Tujuan	Untuk melakukan ubah biodata aktor yang login
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan ubah biodata berdasarkan aktor yang <i>login</i>
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman profil
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih ikon ubah yang ada pada halaman profil 3. Aktor mengubah data pada form biodata 4. Aktor menekan tombol submit	2. Sistem akan menampilkan form ubah biodata 5. Sistem akan menyimpan biodata yang telah diubah



Tabel 4.7 (lanjutan)

Skenario Alternatif 1 : Jika <i>field</i> yang wajib diisi kosong	
1. Aktor memilih ikon ubah yang ada pada halaman profil	2. Sistem akan menampilkan form ubah biodata
3. Aktor mengkosongkan <i>field</i> yang wajib	5. Sistem akan menampilkan pesan <i>error</i> bahwa <i>field</i> tidak boleh kosong
4. Aktor menekan tombol submit	
Konsisi Akhir	Sistem akan menampilkan halaman detail profil yang telah di ubah

4. Skenario *use case* tambah staf

Tabel 4.8 Skenario *use case* tambah staf

Kode Use Case	SKPL-03-01
Nama Use Case	Tambah staf
Aktor	Staf IT
Tujuan	Untuk menambah staf baru.
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor staf IT melakukan penambahan staf baru
Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu staf 3. Aktor memilih sub menu tambah staf 5. Aktor mengisi form tambah staf 6. Aktor menekan tombol submit	2. Sistem akan menampilkan menu <i>dropdown</i> 4. Sistem akan menampilkan form tambah data staf 7. Sistem akan menyimpan data staf baru
Skenario Alternatif 1 : Jika <i>field</i> yang wajib diisi kosong	
1. Aktor memilih menu staf 3. Aktor memilih sub menu tambah staf 5. Aktor mengkosongkan <i>field</i> yang wajib diisi 6. Aktor menekan tombol submit	2. Sistem akan menampilkan menu <i>dropdown</i> 4. Sistem akan menampilkan form tambah staf 7. Sistem akan menampilkan pesan <i>error</i> bahwa <i>field</i> tidak boleh kosong

Tabel 4.8 (lanjutan)

Konsisi Akhir	Sistem akan menampilkan halaman daftar staf
----------------------	---

5. Skenario *use case* ubah staf

Tabel 4.9 Skenario *use case* ubah staf

Kode Use Case	SKPL-03-02
Nama Use Case	Ubah staf
Aktor	Staf IT
Tujuan	Untuk melakukan ubah data staf
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor staf IT melakukan ubah data staf
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman daftar staf
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih aksi ubah pada daftar aktor yang akan diubah 3. Aktor mengubah data staf 4. Aktor menekan tombol submit	2. Sistem akan menampilkan form ubah data staf 5. Sistem akan menyimpan data yang telah diubah
Skenario Alternatif 1 : Jika <i>field</i> yang wajib diisi kosong	
1. Aktor memilih aksi ubah 2. Aktor mengkosongkan <i>field</i> yang wajib diisi 3. Sistem menekan tombol submit	4. Sistem akan menampilkan pesan <i>error</i> bahwa <i>field</i> tidak boleh kosong
Konsisi Akhir	Sistem akan menampilkan halaman daftar staf

6. Skenario *use case* daftar staf

Tabel 4.10 Skenario *use case* daftar staf

Kode Use Case	SKPL-03-03
Nama Use Case	Daftar staf
Aktor	Staf IT
Tujuan	Untuk menampilkan daftar staf.
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor staf IT menampilkan daftar staf
Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>



Tabel 4.10 (lanjutan)

Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu staf 3. Aktor memilih sub menu daftar staf	2. Sistem akan menampilkan menu <i>dropdown</i> staf 4. Sistem akan menampilkan daftar staf
Konsisi Akhir	Sistem akan menampilkan halaman daftar staf

7. Skenario *use case* tambah permintaan buku

Tabel 4.11 Skenario *use case* tambah permintaan buku

Kode Use Case	SKPL-04-01
Nama Use Case	Tambah permintaan buku
Aktor	Staf pengolahan, staf ruang baca dan anggota
Tujuan	Untuk menambahkan permintaan buku
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor menambahkan permintaan bahan pustaka buku
Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu permintaan 3. Aktor memilih sub menu individu 5. Aktor memilih jenis permintaan bahan pustaka buku 7. Aktor menekan tombol submit	2. Sistem akan menampilkan menu <i>dropdown</i> permintaan 4. Sistem akan menampilkan <i>combobox</i> jenis permintaan 6. Sistem akan menampilkan form permintaan buku 8. Sistem akan menyimpan data permintaan buku
Skenario Alternatif 1 : Jika <i>field</i> yang wajib diisi kosong	
1. Aktor memilih menu permintaan 2. Aktor memilih sub menu individu 3. Aktor memilih jenis permintaan bahan pustaka buku 4. Aktor menekan tombol submit	5. Sistem akan menampilkan pesan kesalahan <i>field cannot blank</i>

Tabel 4.11 (lanjutan)

Konsisi Akhir	Sistem akan menampilkan pesan bahwa pesan <i>permintaan buku berhasil ditambahkan!</i>
----------------------	--

8. Skenario use case unduh form permintaan buku

Tabel 4.12 Skenario use case unduh form permintaan buku

Kode Use Case	SKPL-04-02
Nama Use Case	Unduh form permintaan bahan pustaka
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk unduh form permintaan buku
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor mengunduh <i>form</i> permintaan buku
Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu permintaan 3. Aktor memilih sub menu kolektif 5. Aktor menekan link unduh	2. Sistem akan menampilkan menu <i>dropdown</i> permintaan 4. Sistem akan menampilkan form permintaan secara kolektif 6. Sistem akan melakukan proses unduh <i>form</i> permintaan
Konsisi Akhir	Sistem akan menampilkan halaman permintaan kolektif

9. Skenario use case unggah form permintaan buku

Tabel 4.13 Skenario use case unggah form permintaan buku

Kode Use Case	SKPL-04-03
Nama Use Case	Unggah form permintaan buku
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk menambahkan permintaan bahan pustaka secara kolektif dengan cara unggah form permintaan buku
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor menambahkan permintaan bahan pustaka secara kolektif dengan cara unggah form permintaan bahan pustaka buku



Tabel 4.13 (lanjutan)

Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu permintaan 3. Aktor memilih sub menu kolektif 5. Aktor memilih tab menu unggah permintaan 7. Aktor memilih kategori bahan pustaka buku dan unggah file excel yang disediakan 9. Aktor menekan tombol submit	2. Sistem akan menampilkan menu <i>dropdown</i> permintaan 4. Sistem akan menampilkan halaman permintaan 6. Sistem akan menampilkan halaman unggah form permintaan 8. Sistem akan menyimpan data permintaan buku 10. Sistem akan menampilkan pesan sukses unggah
Skenario Alternatif 1 : Jika file yang diunggah tidak sesuai	
1. Aktor memilih menu permintaan 3. Aktor memilih sub menu kolektif 5. Aktor memilih tab menu unggah permintaan 7. Aktor memilih kategori bahan pustaka buku dan unggah file excel yang tidak sesuai 9. Aktor menekan tombol submit	2. Sistem akan menampilkan menu <i>dropdown</i> permintaan 4. Sistem akan menampilkan halaman permintaan 6. Sistem akan menampilkan halaman unggah form permintaan 8. Sistem akan menyimpan data permintaan buku 10. Sistem akan menampilkan pesan kesalahan
Konsisi Akhir	Sistem akan menampilkan pesan <i>data permintaan buku berhasil ditambahkan!</i> atau <i>data gagal dimasukkan. Pastikan file anda bernama FormReqBuku.xls!</i>

10. Skenario *use case* lihat daftar permintaan buku

Tabel 4.14 Skenario *use case* lihat daftar permintaan buku

Kode Use Case	SKPL-04-04
Nama Use Case	Lihat daftar permintaan buku
Aktor	Staf pengolahan, staf ruang baca dan anggota



Tabel 4.14 (lanjutan)

Tujuan	Untuk menampilkan daftar permintaan buku	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihat daftar permintaan buku	
Skenario Utama		
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>	
Aksi Aktor		Tanggapan dari sistem
1. Aktor memilih menu permintaan	3. Aktor memilih sub menu kolektif/individu	5. Aktor memilih tab menu daftar permintaan
2. Sistem akan menampilkan menu <i>dropdown</i> permintaan	4. Sistem akan menampilkan halaman permintaan	6. Sistem akan menampilkan daftar permintaan
7. Aktor memilih sub tab menu buku	8. Sistem akan menampilkan daftar permintaan buku	
Konsisi Akhir	Sistem akan menampilkan daftar permintaan buku	

11. Skenario *use case* lihat daftar permintaan buku disetujui

Tabel 4.15 Skenario *use case* lihat daftar permintaan buku disetujui

Kode Use Case	SKPL-04-05	
Nama Use Case	Lihat daftar permintaan buku disetujui	
Aktor	Staf pengolahan	
Tujuan	Untuk menampilkan daftar permintaan buku disetujui	
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihat daftar permintaan buku yang sudah disetujui	
Skenario Utama		
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>	
Aksi Aktor		Tanggapan dari sistem
1. Aktor memilih menu pengolahan	3. Aktor memilih sub menu permintaan	5. Aktor memilih tab menu permintaan buku
2. Sistem akan menampilkan menu <i>dropdown</i> pengolahan	4. Sistem menampilkan menu permintaan	6. Sistem menampilkan daftar permintaan buku
7. Aktor memilih sub tab menu disetujui	8. Sistem menampilkan daftar permintaan buku disetujui	



Tabel 4.15 (lanjutan)

Konsisi Akhir	Sistem akan menampilkan daftar permintaan buku yang sudah disetujui
----------------------	---

12. Skenario *use case* lihat daftar permintaan buku belum disetujui

Tabel 4.16 Skenario *use case* lihat daftar permintaan buku belum disetujui

Kode Use Case	SKPL-04-06
Nama Use Case	Lihat daftar permintaan buku
Aktor	Staf pengolahan
Tujuan	Untuk menampilkan daftar permintaan buku yang belum disetujui
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihat daftar permintaan buku yang belum disetujui
Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu pengolahan 3. Aktor memilih sub menu permintaan 5. Aktor memilih tab menu permintaan buku 7. Aktor memilih sub tab menu belum disetujui	2. Sistem akan menampilkan menu <i>dropdown</i> pengolahan 4. Sistem menampilkan menu permintaan 6. Sistem menampilkan daftar permintaan buku 8. Sistem menampilkan daftar permintaan buku belum disetujui
Konsisi Akhir	Sistem akan menampilkan daftar permintaan buku yang belum disetujui

13. Skenario *use case* unduh daftar permintaan buku disetujui

Tabel 4.17 Skenario *use case* unduh daftar permintaan buku disetujui

Kode Use Case	SKPL-04-07
Nama Use Case	Unduh daftar permintaan buku disetujui
Aktor	Staf pengolahan
Tujuan	Untuk melakukan unduh permintaan bahan pustaka buku yang sudah disetujui
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan unduh permintaan bahan pustaka buku yang sudah disetujui

Tabel 4.17 (lanjutan)

Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu pengolahan 3. Aktor memilih sub menu permintaan 5. Aktor memilih tab menu permintaan buku kemudian sub tab menu disetujui 7. Aktor mengisi tanggal dan memilih <i>combobox</i> bahasa permintaan buku 8. Aktor menekan link unduh	2. Sistem akan menampilkan menu <i>dropdown</i> pengolahan 4. Sistem akan menampilkan halaman pengolahan permintaan 6. Sistem akan menampilkan daftar permintaan buku disetujui 9. Sistem akan menunduh daftar permintaan yang sudah ditentukan
Konsisi Akhir	Sistem akan menampilkan daftar permintaan bahan pustaka yang akan di unduh

14. Skenario *use case* unduh daftar permintaan buku belum disetujui

Tabel 4.18 Skenario *use case* unduh daftar permintaan buku belum disetujui

Kode Use Case	SKPL-04-08
Nama Use Case	Unduh daftar permintaan buku belum disetujui
Aktor	Staf pengolahan
Tujuan	Untuk melakukan unduh permintaan bahan pustaka buku yang sudah belum disetujui
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan unduh permintaan bahan pustaka buku yang belum sudah disetujui
Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu pengolahan 3. Aktor memilih sub menu permintaan	2. Sistem akan menampilkan menu <i>dropdown</i> pengolahan 4. Sistem akan menampilkan halaman pengolahan permintaan

Tabel 4.18 (lanjutan)

Aksi Aktor	Tanggapan dari sistem
5. Aktor memilih tab menu permintaan buku kemudian sub tab menu belum disetujui 7. Aktor memilih combobox bahasa permintaan buku 8. Aktor menekan link unduh	6. Sistem akan menampilkan daftar permintaan buku belum disetujui 9. Sistem akan mengunduh daftar permintaan yang sudah ditentukan
Konsisi Akhir	Sistem akan menampilkan daftar permintaan bahan pustaka yang akan di unduh

15. Skenario *use case* lihat detail permintaan buku

Tabel 4.19 Skenario *use case* lihat detail permintaan buku

Kode Use Case	SKPL-04-09
Nama Use Case	Lihat detail permintaan buku
Aktor	Staf pengolahan
Tujuan	Untuk melihat detail permintaan bahan pustaka buku
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihat detail daftar permintaan bahan pustaka buku
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman pengolahan permintaan bahan pustaka buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih tab menu permintaan buku 2. Aktor memilih sub tab permintaan buku disetujui / belum disetujui 3. Aktor memilih ikon detail pada daftar permintaan buku	4. Sistem akan menampilkan detail permintaan buku yang dipilih
Konsisi Akhir	Sistem akan menampilkan detail permintaan bahan pustaka buku yang dipilih

16. Skenario *use case* ubah permintaan buku

Tabel 4.20 Skenario *use case* ubah permintaan buku

Kode Use Case	SKPL-04-10
Nama Use Case	Ubah permintaan buku
Aktor	Staf pengolahan
Tujuan	Untuk mengubah permintaan bahan pustaka buku
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan ubah permintaan bahan pustaka buku
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman daftar permintaan buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih ikon edit pada daftar permintaan buku disetujui / belum disetujui 3. Aktor merubah data buku 4. Aktor menekan tombol submit	2. Sistem akan menampilkan form ubah permintaan bahan pustaka buku yang ingin dirubah 5. Sistem akan menyimpan data yang telah diubah
Skenario Alternatif 1 : Jika <i>field</i> yang wajib di isi kosong	
1. Aktor memilih ikon edit pada daftar permintaan buku disetujui / belum disetujui 3. Aktor merubah data buku 4. Aktor menekan tombol submit	2. Sistem akan menampilkan form ubah permintaan bahan pustaka buku yang ingin dirubah 5. Sistem akan menampilkan pesan kesalahan <i>field</i> tidak boleh kosong
Konsisi Akhir	Sistem akan menampilkan pesan sukses mengubah data dan menampilkan detail permintaan buku

17. Skenario *use case* hapus permintaan buku

Tabel 4.21 Skenario *use case* hapus permintaan buku

Kode Use Case	SKPL-04-11
Nama Use Case	Hapus permintaan buku
Aktor	Staf pengolahan
Tujuan	Untuk menghapus permintaan bahan pustaka buku

Tabel 4.21 (lanjutan)

Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan hapus permintaan bahan pustaka buku	
Skenario Utama		
Kondisi Awal	Aktor sudah masuk halaman daftar permintaan buku	
Aksi Aktor	Tanggapan dari sistem	
1. Aktor memilih ikon hapus pada daftar permintaan bahan pustaka buku 3. Aktor memilih tombol ya	2. Sistem akan menampilkan notifikasi apakah data ingin dihapus 4. Sistem akan menghapus data permintaan bahan pustaka buku	
Konsisi Akhir	Sistem akan menampilkan halaman daftar permintaan bahan pustaka buku	

18. Skenario *use case* cari permintaan buku belum disetujui

Tabel 4.22 Skenario *use case* cari permintaan buku belum disetujui

Kode Use Case	SKPL-04-12
Nama Use Case	Cari permintaan buku belum disetujui
Aktor	Staf pengolahan
Tujuan	Untuk mencari data permintaan bahan pustaka buku yang belum disetujui
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan cari data permintaan bahan pustaka buku yang belum disetujui
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman daftar permintaan buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih sub tab belum disetujui 3. Aktor mengisi kata kunci pencarian berdasarkan ISBN 4. Aktor menekan tombol cari	2. Sistem menampilkan daftar permintaan buku belum disetujui 5. Sistem akan menampilkan daftar pencarian
Konsisi Akhir	Sistem akan menampilkan halaman daftar permintaan buku berdasarkan data yang dicari



19. Skenario *use case* tambah buku dari permintaan

Tabel 4.23 Skenario *use case* tambah buku dari permintaan

Kode Use Case	SKPL-04-13
Nama Use Case	Tambah buku dari permintaan
Aktor	Staf pengolahan
Tujuan	Untuk menambah data permintaan bahan pustaka buku berdasarkan daftar permintaan buku yang diajukan
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan tambah data permintaan bahan pustaka buku berdasarkan daftar permintaan buku yang diajukan
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman daftar permintaan buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih ikon tambah pada daftar permintaan buku belum disetujui 3. Aktor melengkapi informasi data buku 4. Aktor menekan tombol submit	2. Sistem akan menampilkan form informasi data bahan pustaka buku 5. Sistem akan menampilkan pesan data buku berhasil dimasukkan dan mengirim pesan kepada anggota yang mengajukan permintaan melalui email
Konsisi Akhir	Sistem akan menampilkan halaman daftar permintaan bahan pustaka buku

20. Skenario *use case* tambah permintaan jurnal

Tabel 4.24 Skenario *use case* tambah permintaan jurnal

Kode Use Case	SKPL-04-14
Nama Use Case	Tambah permintaan jurnal
Aktor	Staf pengolahan, staf ruang baca dan anggota
Tujuan	Untuk menambahkan permintaan jurnal
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor menambahkan permintaan bahan pustaka jurnal
Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>



Tabel 4.24 (lanjutan)

Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu permintaan 3. Aktor memilih sub menu individu 5. Aktor memilih jenis permintaan bahan pustaka jurnal 7. Aktor mengnekan tombol submit	2. Sistem akan menampilkan menu <i>dropdown</i> permintaan 4. Sistem akan menampilkan <i>combobox</i> jenis permintaan 6. Sistem akan menampilkan form permintaan jurnal 8. Sistem akan menyimpan data permintaan jurnal
Skenario Alternatif 1 : Jika <i>field</i> yang wajib diisi kosong	
1. Aktor memilih menu permintaan 3. Aktor memilih sub menu individu 5. Aktor memilih jenis permintaan bahan pustaka jurnal 7. Aktor mengnekan tombol submit	2. Sistem akan menampilkan menu <i>dropdown</i> permintaan 4. Sistem akan menampilkan <i>combobox</i> jenis permintaan 6. Sistem akan menampilkan form permintaan jurnal 8. Sistem akan menampilkan pesan kesalahan field tidak boleh kosong
Konsisi Akhir	Sistem akan menampilkan pesan bahwa data sukses dimasukkan atau pesan kesalahan

21. Skenario *use case* unduh form permintaan jurnal

Tabel 4.25 Skenario *use case* unduh form permintaan jurnal

Kode Use Case	SKPL-04-15
Nama Use Case	Unduh form permintaan jurnal
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk unduh form permintaan jurnal
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor unduh form permintaan jurnal
Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>

Tabel 4.25 (lanjutan)

Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu permintaan	2. Sistem akan menampilkan menu <i>dropdown</i> permintaan
3. Aktor memilih sub menu kolektif	4. Sistem akan menampilkan form permintaan secara kolektif
5. Aktor menekan link unduh	6. Sistem akan melakukan proses unduh form permintaan
Konsisi Akhir	Sistem akan menampilkan halaman permintaan kolektif

22. Skenario *use case* unggah form permintaan jurnal

Tabel 4.26 Skenario *use case* unggah form permintaan jurnal

Kode Use Case	SKPL-04-16
Nama Use Case	Unggah form permintaan jurnal
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk menambahkan permintaan bahan pustaka secara kolektif dengan cara unggah form permintaan jurnal.
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor menambahkan permintaan bahan pustaka secara kolektif dengan cara unggah form permintaan bahan pustaka jurnal.
Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu permintaan	2. Sistem akan menampilkan menu <i>dropdown</i> permintaan.
3. Aktor memilih sub menu kolektif	4. Sistem akan menampilkan halaman permintaan
5. Aktor memilih tab menu unggah permintaan	6. Sistem akan menampilkan halaman unggah permintaan jurnal
7. Aktor memilih kategori bahan pustaka jurnal dan unggah file excel	8. Sistem akan menyimpan data permintaan jurnal
9. Aktor menekan tombol submit	10. Sistem akan menampilkan pesan sukses unggah

Tabel 4.26 (lanjutan)

Skenario Alternatif 1 : Jika file yang diunggah tidak sesuai	
1. Aktor memilih menu permintaan	2. Sistem akan menampilkan menu <i>dropdown</i> permintaan.
3. Aktor memilih sub menu kolektif	4. Sistem akan menampilkan halaman permintaan
5. Aktor memilih tab menu unggah permintaan	6. Sistem akan menampilkan halaman unggah permintaan jurnal
7. Aktor memilih kategori bahan pustaka jurnal dan unggah file excel yang tidak sesuai	8. Sistem akan menyimpan data permintaan jurnal
9. Aktor menekan tombol submit	10. Sistem akan menampilkan pesan kesalahan unggah file
Konsisi Akhir	Sistem akan menampilkan pesan sukses / gagal unggah

23. Skenario *use case* lihat daftar permintaan jurnal

Tabel 4.27 Skenario *use case* lihat daftar permintaan jurnal

Kode Use Case	SKPL-04-17
Nama Use Case	Lihat daftar permintaan jurnal
Aktor	Staf pengolahan, staf ruang baca dan anggota
Tujuan	Untuk menampilkan daftar permintaan jurnal
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihat daftar permintaan jurnal
Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu permintaan	2. Sistem akan menampilkan menu <i>dropdown</i> permintaan
3. Aktor memilih sub menu kolektif/individu	5. Sistem akan menampilkan halaman permintaan
6. Aktor memilih tab menu daftar permintaan	9. Sistem akan menampilkan daftar permintaan jurnal
7. Aktor memilih sub tab menu jurnal	

Tabel 4.27 (lanjutan)

Konsisi Akhir	Sistem akan menampilkan daftar permintaan bahan pustaka.
----------------------	--

24. Skenario *use case* lihat daftar permintaan jurnal disetujui

Tabel 4.28 Skenario *use case* lihat daftar permintaan jurnal disetujui

Kode Use Case	SKPL-04-18
Nama Use Case	Lihat daftar permintaan jurnal disetujui
Aktor	Staf pengolahan
Tujuan	Untuk menampilkan daftar permintaan jurnal disetujui
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihat daftar permintaan jurnal yang sudah disetujui
Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu pengolahan 3. Aktor memilih sub menu permintaan 5. Aktor memilih tab menu permintaan jurnal 7. Aktor memilih sub tab menu disetujui	2. Sistem akan menampilkan menu <i>dropdown</i> pengolahan 4. Sistem menampilkan menu permintaan 6. Sistem menampilkan daftar permintaan jurnal 8. Sistem menampilkan daftar permintaan jurnal disetujui
Konsisi Akhir	Sistem akan menampilkan daftar permintaan bahan pustaka jurnal yang sudah disetujui

25. Skenario *use case* lihat daftar permintaan jurnal belum disetujui

Tabel 4.29 Skenario *use case* lihat daftar permintaan jurnal belum disetujui

Kode Use Case	SKPL-04-19
Nama Use Case	Lihat daftar permintaan jurnal
Aktor	Staf pengolahan, staf ruang baca dan anggota
Tujuan	Untuk menampilkan daftar permintaan jurnal yang belum disetujui
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihat daftar permintaan jurnal yang belum disetujui

Tabel 4.29 (lanjutan)

Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu pengolahan 3. Aktor memilih sub menu permintaan 5. Aktor memilih tab menu permintaan buku 7. Aktor memilih sub tab menu belum disetujui	2. Sistem akan menampilkan menu <i>dropdown</i> pengolahan 4. Sistem menampilkan menu permintaan 6. Sistem menampilkan daftar permintaan buku 8. Sistem menampilkan daftar permintaan buku belum disetujui
Konsisi Akhir	Sistem akan menampilkan daftar permintaan bahan pustaka.

26. Skenario *use case* unduh daftar permintaan jurnal disetujui

Tabel 4.30 Skenario *use case* unduh daftar permintaan jurnal disetujui

Kode Use Case	SKPL-04-20
Nama Use Case	Unduh daftar permintaan jurnal disetujui
Aktor	Staf pengolahan
Tujuan	Untuk melakukan unduh permintaan bahan pustaka jurnal yang sudah disetujui
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan unduh permintaan bahan pustaka jurnal yang sudah disetujui
Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu pengolahan 3. Aktor memilih sub menu permintaan 5. Aktor memilih tab menu permintaan jurnal kemudian sub tab menu disetujui	2. Sistem akan menampilkan menu <i>dropdown</i> pengolahan 4. Sistem akan menampilkan halaman pengolahan permintaan 6. Sistem akan menampilkan daftar permintaan jurnal disetujui



Tabel 4.30 (lanjutan)

Aksi Aktor	Tanggapan dari sistem
7. Aktor mengisi tanggal dan memilih <i>combobox</i> bahasa permintaan jurnal 8. Aktor menekan link unduh	9. Sistem akan menunduh daftar permintaan yang sudah ditentukan
Konsisi Akhir	Sistem akan menampilkan daftar permintaan bahan pustaka yang akan di unduh.

27. Skenario *use case* unduh daftar permintaan jurnal belum disetujui

Tabel 4.31 Skenario *use case* unduh daftar permintaan jurnal belum disetujui

Kode Use Case	SKPL-04-21
Nama Use Case	Unduh daftar permintaan jurnal belum disetujui
Aktor	Staf pengolahan
Tujuan	Untuk melakukan unduh permintaan bahan pustaka jurnal yang sudah belum disetujui
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan unduh permintaan bahan pustaka jurnal yang belum disetujui
Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih menu pengolahan 3. Aktor memilih sub menu permintaan 5. Aktor memilih tab menu permintaan jurnal kemudian sub tab menu belum disetujui 7. Aktor memilih <i>combobox</i> bahasa permintaan jurnal 8. Aktor menekan link unduh	2. Sistem akan menampilkan menu <i>dropdown</i> pengolahan 4. Sistem akan menampilkan halaman pengolahan permintaan 6. Sistem akan menampilkan daftar permintaan jurnal belum disetujui 9. Sistem akan menunduh daftar permintaan yang sudah ditentukan
Konsisi Akhir	Sistem akan menampilkan daftar permintaan bahan pustaka yang akan di unduh.

28. Skenario *use case* lihat detail permintaan jurnal

Tabel 4.32 Skenario *use case* lihat detail permintaan jurnal

Kode Use Case	SKPL-04-22
Nama Use Case	Lihat detail permintaan jurnal
Aktor	Staf pengolahan
Tujuan	Untuk melihat detail permintaan bahan pustaka jurnal
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihat detail daftar permintaan bahan pustaka jurnal
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman permintaan jurnal
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih sub tab menu disetujui / belum disetujui 3. Aktor memilih ikon detail pada daftar permintaan jurnal	2. Sistem akan menampilkan daftar permintaan jurnal 4. Sistem akan menampilkan detail permintaan jurnal yang dipilih
Konsisi Akhir	Sistem akan menampilkan detail permintaan bahan pustaka yang dipilih

29. Skenario *use case* ubah permintaan jurnal

Tabel 4.33 Skenario *use case* ubah permintaan jurnal

Kode Use Case	SKPL-04-23
Nama Use Case	Ubah permintaan jurnal
Aktor	Staf pengolahan
Tujuan	Untuk mengubah permintaan bahan pustaka jurnal
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan ubah permintaan bahan pustaka jurnal
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman pengolahan permintaan jurnal
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih sub tab menu disetujui / belum disetujui 3. Aktor memilih tombol ubah pada daftar permintaan jurnal	2. Sistem akan menampilkan daftar permintaan jurnal 4. Sistem akan menampilkan form ubah permintaan jurnal yang ingin dirubah



Tabel 4.33 (lanjutan)

Aksi Aktor	Tanggapan dari sistem
5. Aktor merubah data jurnal 6. Aktor menekan tombol submit	7. Sistem akan menyimpan data yang telah diubah
Skenario Alternatif 1 : Jika field yang wajib di isi kosong	
1. Aktor memilih sub tab menu disetujui / belum disetujui 3. Aktor memilih tombol ubah pada daftar permintaan jurnal 5. Aktor merubah data jurnal 6. Aktor menekan tombol submit	2. Sistem akan menampilkan daftar permintaan jurnal 4. Sistem akan menampilkan form ubah permintaan jurnal yang ingin dirubah 7. Sistem akan menampilkan pesan kesalahan bahwa field tidak boleh kosong
Konsisi Akhir	Sistem akan menampilkan pesan sukses / gagal mengubah data

30. Skenario use case hapus permintaan jurnal

Tabel 4.34 Skenario use case hapus permintaan jurnal

Kode Use Case	SKPL-04-24
Nama Use Case	Hapus permintaan jurnal
Aktor	Staf pengolahan
Tujuan	Untuk menghapus permintaan bahan pustaka jurnal
Deskripsi	Use case ini menjelaskan bagaimana aktor melakukan hapus permintaan bahan pustaka jurnal
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman pengolahan permintaan buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih sub tab menu disetujui / belum disetujui 3. Aktor memilih tombol hapus pada daftar permintaan jurnal 5. Aktor menekan tombol ya	2. Sistem akan menampilkan daftar permintaan jurnal 4. Sistem akan menampilkan pesan apakah data ingin dihapus 6. Sistem akan menghapus data yang dipilih
Konsisi Akhir	Sistem akan menampilkan halaman daftar permintaan bahan pustaka jurnal

3.1 Skenario *use case* cari permintaan jurnal belum disetujui

Tabel 4.35 Skenario *use case* cari permintaan jurnal belum disetujui

Kode Use Case	SKPL-04-25
Nama Use Case	Cari permintaan jurnal belum disetujui
Aktor	Staf pengolahan
Tujuan	Untuk mencari data permintaan bahan pustaka jurnal yang belum disetujui
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan cari data permintaan bahan pustaka jurnal yang belum disetujui
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman permintaan jurnal
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih sub tab belum disetujui 3. Aktor mengisi kata kunci pencarian berdasarkan ISSN 4. Aktor menekan tombol cari	2. Sistem menampilkan daftar permintaan jurnal belum disetujui 5. Sistem akan menampilkan daftar pencarian jurnal
Konsisi Akhir	Sistem akan menampilkan halaman daftar permintaan buku berdasarkan data yang dicari

3.2 Skenario *use case* tambah jurnal dari permintaan

Tabel 4.36 Skenario *use case* tambah jurnal dari permintaan

Kode Use Case	SKPL-04-26
Nama Use Case	Tambah jurnal dari permintaan
Aktor	Staf pengolahan
Tujuan	Untuk menambah data permintaan bahan pustaka jurnal berdasarkan daftar permintaan jurnal yang diajukan
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan tambah data permintaan bahan pustaka jurnal berdasarkan daftar permintaan buku yang diajukan
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman pengolahan permintaan jurnal

Tabel 4.36 (lanjutan)

Aksi Aktor	Tanggapan dari sistem
2. Aktor memilih ikon tambah pada daftar permintaan jurnal belum disetujui	3. Sistem akan menampilkan form informasi data bahan pustaka jurnal
5. Aktor melengkapi informasi data jurnal	6. Sistem akan menampilkan pesan data jurnal berhasil dimasukkan dan mengirim pesan kepada anggota yang mengajukan permintaan melalui email
6. Aktor menekan tombol submit	
Konsisi Akhir	Sistem akan menampilkan halaman daftar permintaan bahan pustaka jurnal

3.3 Skenario *use case* tambah buku baru

Tabel 4.37 Skenario *use case* tambah buku baru

Kode Use Case	SKPL-05-01
Nama Use Case	Tambah buku baru
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan tambah buku baru.
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan penambahan buku baru / buku label merah
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman pengolahan buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih sub menu registrasi buku	2. Sistem akan menampilkan form entri buku
3. Aktor memasukkan data informasi buku	5. Sistem akan menyimpan data informasi buku
4. Aktor menekan tombol submit informasi	6. Sistem akan menampilkan pesan data berhasil dimasukkan
Skenario Alternatif 1 : Jika <i>field</i> yang wajib di isi kosong	
1. Aktor memilih sub menu registrasi buku	4. Sistem akan menampilkan pesan kesalahan bahwa proses input gagal
2. Aktor mengkonkan field yang wajib diisi	
3. Aktor menekan tombol submit informasi	

Tabel 4.37 (lanjutan)

Skenario Alternatif 2 : Jika karekter yang dimasukkan tidak sesuai	
1. Aktor memasukkan karekter yang tidak sesuai	3. Sistem akan menampilkan pesan error bahwa inputan tidak sesuai
2. Aktor menekan tombol submit	
Konsisi Akhir	Sistem akan menampilkan pesan berhasil atau gagal

34. Skenario *use case* unduh *form* tambah buku

Tabel 4.38 Skenario *use case* unduh *form* tambah buku

Kode Use Case	SKPL-05-02
Nama Use Case	Unduh <i>form</i> tambah buku
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan unduh <i>form</i> tambah buku
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan unduh <i>form</i> tambah buku
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman pengolahan buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu kolektif	2. Sistem akan menampilkan halaman registrasi bahan pustaka secara kolektif
3. Aktor menekan <i>link</i> unduh	4. Sistem akan melakukan proses unduh <i>form</i> dalam bentuk file excel
Konsisi Akhir	Sistem akan menampilkan halaman unduh <i>form</i> buku kolektif

3.5 Skenario *use case* unggah *form* tambah buku baru

Tabel 4.39 Skenario *use case* unggah *form* tambah buku baru

Kode Use Case	SKPL-05-03
Nama Use Case	Unggah <i>form</i> tambah buku baru
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan unggah <i>form</i> tambah buku
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan unggah <i>form</i> tambah buku

Tabel 4.39 (lanjutan)

Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu kolektif 3. Aktor menekan <i>button browse</i> dan memilih <i>file excel</i> yang akan diunggah 4. Aktor menekan tombol submit	2. Sistem akan menampilkan halaman registrasi bahan pustaka secara kolektif 5. Sistem akan melakukan proses unggah <i>form</i> dalam bentuk file excel
Skenario Alternatif 1 : Jika file yang diunggah tidak sesuai dengan jenis bahan pustaka	
1. Aktor menekan <i>tab</i> menu kolektif 3. Aktor menekan <i>button browse</i> dan memilih <i>file</i> yang tidak sesuai 4. Aktor menekan tombol submit	2. Sistem akan menampilkan halaman registrasi bahan pustaka buku secara kolektif 5. Sistem akan menampilkan pesan kesalahan bawa <i>form</i> yang diunggah gagal
Konsisi Akhir	Sistem akan menampilkan pesan bahwa <i>file</i> telah berhasil diunggah atau gagal

3.6 Skenario *use case* lihan daftar buku label merah

Tabel 4.40 Skenario *use case* lihan daftar buku label merah

Kode Use Case	SKPL-05-04
Nama Use Case	Lihan daftar buku label merah
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan lihan daftar buku label merah
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihan daftar buku label merah
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label merah	2. Sistem akan menampilkan daftar buku yang berlabel merah
Konsisi Akhir	Sistem akan menampilkan daftar buku yang berlabel merah

37. Skenario *use case* cari buku label merah

Tabel 4.41 Skenario *use case* cari buku label merah

Kode Use Case	SKPL-05-05
Nama Use Case	Cari buku label merah
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan pencarian buku label merah
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan pencarian buku label merah
Skenario Utama	
Kondisi Awal	Aktor sudah masuk pada halaman registrasi buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label merah 3. Aktor memilih jenis pencarian dan mengisi field pencarian 4. Aktor menekan button cari	2. Sistem akan menampilkan daftar buku 5. Sistem akan melakukan pencarian bahan pustaka berdasarkan jenis pencarian
Skenario Alternatif 1 : Jika tidak memilih jenis pencarian	
1. Aktor menekan <i>tab</i> menu label merah 3. Aktor tidak memilih jenis pencarian 4. Aktor menekan button cari	2. Sistem akan menampilkan daftar buku 5. Sistem akan menampilkan pesan kesalahan
Konsisi Akhir	Sistem akan menampilkan daftar bahan pustaka yang dicari

38. Skenario *use case* lihat detail buku label merah

Tabel 4.42 Skenario *use case* lihat detail buku label merah

Kode Use Case	SKPL-05-06
Nama Use Case	Lihat detail buku label merah
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan lihat detail buku label merah
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan lihat detail buku label merah
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi buku



Tabel 4.42 (lanjutan)

Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label merah 3. Aktor menekan ikon detail yang terdapat pada daftar buku	2. Sistem akan menampilkan daftar buku 4. Sistem akan menampilkan detail buku yang dipilih
Konsisi Akhir	Sistem akan menampilkan detail informasi bahan pustaka yang dipilih.

39. Skenario *use case* ubah buku label merah

Tabel 4.43 Skenario *use case* ubah buku label merah

Kode Use Case	SKPL-05-07
Nama Use Case	Ubah buku label merah
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan ubah buku label merah
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan ubah buku label merah
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label merah 3. Aktor menekan ikon ubah yang terdapat pada daftar buku 5. Aktor mengubah data yang ingin dirubah dan menekan button submit	2. Sistem akan menampilkan daftar buku 4. Sistem akan menampilkan form ubah daftar buku yang dipilih 6. Sistem akan menampilkan detail informasi buku yang telah diubah
Skenario Alternatif 1 : Jika <i>field</i> yang wajib diisi kosong	
1. Aktor menekan <i>tab</i> menu label merah 3. Aktor menekan ikon ubah yang terdapat pada daftar buku 5. Aktor mengubah data yang ingin dirubah dan menekan button submit	2. Sistem akan menampilkan daftar buku 4. Sistem akan menampilkan form ubah daftar buku yang dipilih 6. Sistem akan menampilkan detail informasi buku yang telah diubah

Tabel 4.43 (lanjutan)

Konsisi Akhir	Sistem akan menampilkan detail informasi bahan pustaka yang telah diubah
----------------------	--

40. Skenario *use case* tambah jurnal baru

Tabel 4.44 Skenario *use case* tambah jurnal baru

Kode Use Case	SKPL-05-08
Nama Use Case	Tambah jurnal baru
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan tambah jurnal baru
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan penambahan jurnal baru / jurnal label merah
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman register jurnal
Aksi Aktor	Tanggapan dari sistem
1. Aktor memilih tab menu individu 3. Aktor memasukkan data informasi jurnal dan menekan tombol submit informasi	2. Sistem akan menampilkan form entri jurnal 5. Sistem akan menyimpan data informasi jurnal 6. Sistem akan menampilkan pesan proses input berhasil
Skenario Alternatif 1 : Jika <i>field</i> yang wajib di isi kosong	
1. Aktor memilih tab menu individu 3. Aktor mengosongkan field yang wajib diisi 4. Aktor menekan tombol submit	2. Sistem akan menampilkan form entri jurnal 5. Sistem akan menampilkan pesan <i>error</i> bahwa <i>field</i> tidak boleh kosong
Skenario Alternatif 2 : Jika karakter yang diinputkan tidak sesuai	
1. Aktor memilih tab menu individu 3. Aktor memasukkan karakter yang tidak sesuai 35. Aktor menekan tombol submit	2. Sistem akan menampilkan form entri jurnal 5. Sistem akan menampilkan pesan <i>error</i> bahwa inputan tidak sesuai
Konsisi Akhir	Sistem akan menampilkan pesan data berhasil ditambahkan atau pesan kesalahan



41. Skenario *use case* unduh *form* tambah jurnal

Tabel 4.45 Skenario *use case* unduh *form* tambah jurnal

Kode Use Case	SKPL-05-09
Nama Use Case	Unduh <i>form</i> tambah jurnal
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan unduh <i>form</i> tambah jurnal
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan unduh <i>form</i> tambah jurnal
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi jurnal
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu kolektif 3. Aktor menekan <i>link</i> unduh <i>form</i> tambah jurnal	2. Sistem akan menampilkan halaman registrasi jurnal secara kolektif 4. Sistem akan melakukan proses unduh <i>form</i> dalam bentuk file excel
Konsisi Akhir	Sistem akan menampilkan halaman registrasi serta proses unduh <i>form</i>

42. Skenario *use case* unggah *form* tambah jurnal

Tabel 4.46 Skenario *use case* unggah *form* tambah jurnal

Kode Use Case	SKPL-05-10
Nama Use Case	Unggah <i>form</i> tambah jurnal
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan unggah <i>form</i> tambah jurnal
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan unggah <i>form</i> tambah jurnal
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi jurnal
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu kolektif 3. Aktor menekan <i>button browse</i> dan memilih <i>file excel</i> yang akan diunggah 4. Aktor menekan tombol sumit	2. Sistem akan menampilkan halaman registrasi jurnal secara kolektif 5. Sistem akan melakukan proses unggah <i>form</i> dalam bentuk file excel

Tabel 4.46 (lanjutan)

Skenario Alternatif 1 : Jika file <i>excel</i> yang diunggah tidak sesuai dengan jenis bahan pustaka	
1. Aktor menekan <i>tab</i> menu kolektif	2. Sistem akan menampilkan halaman registrasi jurnal secara kolektif
3. Akor mengunggah file <i>excel</i> yang tidak sesuai	5. Sistem akan menampilkan pesan <i>error</i> bahwa <i>file</i> yang di unggah tidak sesuai
4. Aktor menekan tombol unggah	
Konsisi Akhir	Sistem akan menampilkan pesan bahwa <i>file</i> telah berhasil diunggah atau file gagal diunggah

43. Skenario *use case* lihan daftar jurnal label merah

Tabel 4.47 Skenario *use case* lihan daftar jurnal label merah

Kode Use Case	SKPL-05-11
Nama Use Case	Lihan daftar jurnal label merah
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan lihan daftar jurnal label merah
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melihan daftar jurnal label merah
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi jurnal
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label merah	2. Sistem akan menampilkan daftar jurnal yang berlabel merah
Konsisi Akhir	Sistem akan menampilkan daftar jurnal label merah

44. Skenario *use case* cari jurnal label merah

Tabel 4.48 Skenario *use case* cari jurnal label merah

Kode Use Case	SKPL-05-12
Nama Use Case	Cari jurnal label merah
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan pencarian jurnal label merah

Tabel 4.47 (lanjutan)

Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan pencarian jurnal label merah	
Skenario Utama		
Kondisi Awal	Aktor sudah masuk pada halaman registrasi jurnal	
Aksi Aktor	Tanggapan dari sistem	
<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 3. Aktor memilih kategori pencarian dan mengisi field pencarian 4. Aktor menekan button cari 	<ol style="list-style-type: none"> 2. Sistem akan menampilkan daftar jurnal label merah 5. Sistem akan melakukan pencarian bahan pustaka berdasarkan kategori pencarian 	
Skenario Alternatif 1 : Jika field pencarian kosong		
<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 3. Aktor memilih kategori pencarian dan mengisi field pencarian 4. Aktor menekan button cari 	<ol style="list-style-type: none"> 2. Sistem akan menampilkan daftar jurnal label merah 5. Sistem akan menampilkan pesan kesalahan field tidak boleh kosong 	
Konsisi Akhir	Sistem akan menampilkan daftar jurnal label merah yang dicari atau pesan kesalahan <i>pilih jenis pencarian!</i>	

45. Skenario *use case* lihat detail jurna label merah

Tabel 4.49 Skenario *use case* lihat detail jurnal label merah

Kode <i>Use Case</i>	SKPL-05-13
Nama <i>Use Case</i>	Lihat detail jurnal label merah
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan lihat detail jurnal label merah
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan lihat detail jurnal label merah
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi jurnal

Tabel 4.49 (lanjutan)

Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label merah	2. Sistem akan menampilkan daftar bajurnalhan pustaka
3. Aktor menekan tombol detail yang terdapat pada daftar jurnal	4. Sistem akan menampilkan detail jurnal yang dipilih
Konsisi Akhir	Sistem akan menampilkan detail informasi jurnal yang dipilih

46. Skenario *use case* ubah jurnal label merah

Tabel 4.50 Skenario *use case* ubah jurnal label merah

Kode Use Case	SKPL-05-14
Nama Use Case	Ubah jurnal label merah
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan ubah jurnal label merah
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan ubah jurnal label merah
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi jurnal
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label merah	2. Sistem akan menampilkan daftar jurnal
3. Aktor menekan ikon ubah yang terdapat pada daftar bajurnal	4. Sistem akan menampilkan form ubah data jurnal yang dipilih
5. Aktor mengubah data yang ingin dirubah dan menekan button submit	6. Sistem akan menampilkan detail informasi jurnal yang telah di ubah
Skenario Alternatif 1 : Jika <i>field</i> yang wajib diisi kosong	
1. Aktor menekan <i>tab</i> menu label merah	2. Sistem akan menampilkan daftar jurnal
3. Aktor menekan ikon ubah yang terdapat pada daftar jurnal	4. Sistem akan menampilkan form ubah data jurnal yang dipilih
5. Aktor mengosongkan field yang wajib diisi dan menekan button submit	6. Sistem akan menampilkan pesan kesalahan field tidak boleh kosong

Tabel 4.50 (lanjutan)

Konsisi Akhir	Sistem akan menampilkan detail informasi bahan pustaka yang telah di ubah.
----------------------	--

47. Skenario use case tambah buku label putih

Tabel 4.51 Skenario use case tambah buku label putih

Kode Use Case	SKPL-06-01
Nama Use Case	Tambah buku label putih
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan tambah buku label putih
Deskripsi	Use case ini menjelaskan bagaimana aktor melakukan penambahan buku label putih
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label merah / label putih 3. Aktor menekan ikon tambah yang terdapat pada daftar buku 5. Aktor melengkapi data buku dan menekan button submit	2. Sistem akan menampilkan daftar buku 4. Sistem akan menampilkan form tambah buku label putih / tambah copy 6. Sistem akan menyimpan data buku
Konsisi Akhir	Sistem akan menampilkan pesan data berhasil ditambahkan

48. Skenario use case lihan daftar buku label putih

Tabel 4.52 Skenario use case lihan daftar buku label putih

Kode Use Case	SKPL-06-02
Nama Use Case	Lihan daftar buku label putih
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan lihan daftar buku label putih
Deskripsi	Use case ini menjelaskan bagaimana aktor melihan daftar buku label putih
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi buku



Tabel 4.52 (lanjutan)

Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label putih	2. Sistem akan menampilkan daftar buku yang berlabel putih
Konsisi Akhir	Sistem akan menampilkan daftar buku yang berlabel putih

49. Skenario *use case* cari buku label putih

Tabel 4.53 Skenario *use case* cari buku label putih

Kode Use Case	SKPL-06-03
Nama Use Case	Cari buku label putih
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan pencarian buku yang berlabel putih
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan pencarian buku yang berlabel putih
Skenario Utama	
Kondisi Awal	Aktor sudah masuk pada halaman registrasi buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label putih	2. Sistem akan menampilkan daftar buku label putih
3. Aktor memilih jenis pencarian dan mengisi field pencarian	5. Sistem akan melakukan pencarian buku berdasarkan jenis pencarian
4. Aktor menekan button cari	
Skenario Alternatif 1 : Jika tidak memilih jenis pencarian	
1. Aktor menekan <i>tab</i> menu label putih	2. Sistem akan menampilkan daftar buku label putih
3. Aktor tidak memilih jenis pencarian dan mengisi field pencarian	5. Sistem akan menampilkan pesan kesalahan <i>pilih jenis pencarian!</i>
4. Aktor menekan button cari	
Konsisi Akhir	Sistem akan menampilkan daftar bahan buku label putih yang dicari atau menampilkan pesan kesalahan <i>pilih jenis pencarian!</i>

50. Skenario *use case* ubah buku label putih

Tabel 4.54 Skenario *use case* ubah buku label putih

Kode Use Case	SKPL-06-04
Nama Use Case	Ubah buku label putih
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan ubah buku label putih
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan ubah buku label putih
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label putih	2. Sistem akan menampilkan daftar buku label putih
3. Aktor menekan ikon ubah yang terdapat pada daftar buku	4. Sistem akan menampilkan form ubah buku
5. Aktor mengubah data yang ingin dirubah dan menekan button simpan	6. Sistem akan menampilkan detail informasi buku yang telah di ubah
Konsisi Akhir	Sistem akan menampilkan detail informasi bahan pustaka buku yang telah di ubah

51. Skenario *use case* lihat detail buku label putih

Tabel 4.55 Skenario *use case* lihat detail buku label putih

Kode Use Case	SKPL-06-05
Nama Use Case	Lihat detail buku label putih
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan lihat detail buku label putih
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan lihat detail buku label putih
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi buku

Tabel 4.55 (lanjutan)

Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label putih	2. Sistem akan menampilkan daftar buku label putih
3. Aktor menekan tombol detail yang terdapat pada daftar buku label putih	4. Sistem akan menampilkan detail data buku label putih yang dipilih
Konsisi Akhir	Sistem akan menampilkan detail informasi buku label putih yang dipilih

52. Skenario *use case* tambah jurnal label putih

Tabel 4.56 Skenario *use case* tambah jurnal jurnal label putih

Kode Use Case	SKPL-06-07
Nama Use Case	Tambah jurnal label putih
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan tambah label putih
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan penambahan jurnal label putih
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi jurnal
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label merah / label putih	2. Sistem akan menampilkan daftar jurnal
3. Aktor menekan ikon tambah yang terdapat pada daftar jurnal	4. Sistem akan menampilkan form tambah jurnal label putih
5. Aktor melengkapi data buku dan menekan button submit	6. Sistem akan menyimpan data buku
Konsisi Akhir	Sistem akan menampilkan pesan data berhasil ditambahkan

53. Skenario *use case* lihan daftar jurnal label putih

Tabel 4.57 Skenario *use case* lihan daftar jurnal label putih

Kode Use Case	SKPL-06-08
Nama Use Case	Lihan daftar jurnal label putih
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan lihan daftar jurnal label putih

Tabel 4.57 (lanjutan)

Deskripsi	Use case ini menjelaskan bagaimana aktor melihan daftar jurnal label putih	
Skenario Utama		
Kondisi Awal	Aktor sudah masuk halaman registrasi jurnal	
Aksi Aktor		Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label putih		2. Sistem akan menampilkan daftar buku yang berlabel putih
Konsisi Akhir	Sistem akan menampilkan daftar buku yang berlabel putih	

54. Skenario use case cari jurnal label putih

Tabel 4.58 Skenario use case cari jurnal label putih

Kode Use Case	SKPL-06-09	
Nama Use Case	Cari jurnal label putih	
Aktor	Staf pengolahan dan staf ruang baca	
Tujuan	Untuk melakukan pencarian jurnal label putih	
Deskripsi	Use case ini menjelaskan bagaimana aktor melakukan perncarian jurnal label putih	
Skenario Utama		
Kondisi Awal	Aktor sudah masuk pada halaman registrasi jurnal	
Aksi Aktor		Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label putih		2. Sistem akan menampilkan daftar jurnal label putih
3. Aktor memilih jenis pencarian dan mengisi field pencarian		5. Sistem akan melakukan pencarian jurnal berdasarkan jenis pencarian
4. Aktor menekan button cari		
Skenario Alternatif 1 : Jika tidak memilih jenis pencarian		
1. Aktor menekan <i>tab</i> menu label putih		2. Sistem akan menampilkan daftar jurnal label putih
3. Aktor tidak memilih jenis pencarian dan mengisi field pencarian		5. Sistem akan menampilkan pesan kesalahan <i>pilih jenis pencarian!</i>
4. Aktor menekan button cari		



Tabel 4.58 (lanjutan)

Konsisi Akhir	Sistem akan menampilkan daftar bahan jurnal label putih yang dicari atau menampilkan pesan kesalahan <i>pilih jenis pencarian!</i>
----------------------	--

55. Skenario use case ubah jurnal label putih

Tabel 4.59 Skenario use case ubah jurnal label putih

Kode Use Case	SKPL-06-10
Nama Use Case	Ubah jurnal label putih
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan ubah jurnal label putih
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan ubah jurnal label putih
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi jurnal
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label putih	2. Sistem akan menampilkan daftar jurnal label putih
3. Aktor menekan ikon ubah yang terdapat pada daftar jurnal	4. Sistem akan menampilkan form ubah jurnal
5. Aktor mengubah data yang ingin dirubah dan menekan button simpan	6. Sistem akan menampilkan detail informasi jurnal yang telah di ubah
Konsisi Akhir	Sistem akan menampilkan detail informasi bahan pustaka jurnal yang telah di ubah

56. Skenario use case lihat detail jurna label putih

Tabel 4.60 Skenario use case lihat detail jurnal label putih

Kode Use Case	SKPL-06-11
Nama Use Case	Lihat detail jurnal label putih
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan lihat detail jurnal label putih
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan lihat detail jurnal label putih

Tabel 4.60 (lanjutan)

Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi jurnal
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label putih 3. Aktor menekan tombol detail yang terdapat pada daftar buku label putih	2. Sistem akan menampilkan daftar jurnal label putih 4. Sistem akan menampilkan detail data buku label putih yang dipilih
Konsisi Akhir	Sistem akan menampilkan detail informasi bahan pustaka jurnal yang dipilih

57. Skenario *use case* cetak barcode dan label buku label merah

Tabel 4.61 Skenario *use case* cetak barcode dan label buku label merah

Kode Use Case	SKPL-07-01
Nama Use Case	Cetak barcode dan label buku label merah
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan cetak barcode dan label buku label merah
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan cetak barcode dan label buku label merah
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label merah 3. Aktor melakukan <i>ceklist</i> pada <i>checkbox</i> yang terdapat pada daftar buku, kemudian menekan <i>button</i> cetak barcode dan label 5. Aktor menekan tombol Ctrl+p	2. Sistem akan menampilkan daftar buku 4. Sistem akan menampilkan barcode dan label buku yang dipilih 6. Sistem akan melakukan proses cetak <i>barcode</i> dan label

Tabel 4.56 (lanjutan)

Skenario Alternatif 1 : Jika tidak melakukan <i>ceklist</i> pada <i>cekbox</i>	
1. Aktor menekan <i>tab</i> menu label merah	2. Sistem akan menampilkan daftar buku
3. Aktor tidak melakukan <i>ceklist</i> pada <i>cekbox</i> yang terdapat pada daftar buku, kemudian menekan <i>button</i> cetak barcode dan label	4. Sistem akan menampilkan pesan kesalahan <i>Harap pilih jurnal yang akan dicetak register dan labelnya!</i>
Konsisi Akhir	Sistem akan menampilkan <i>barcode</i> dan label yang akan dicetak atau menampilkan pesan kesalahan

58. Skenario *use case* cetak barcode dan label buku label putih

Tabel 4.62 Skenario *use case* cetak barcode dan label buku label putih

Kode Use Case	SKPL-07-02
Nama Use Case	Cetak barcode dan label buku label putih
Aktor	taf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan cetak barcode dan label buku label putih
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan cetak barcode dan label buku label putih
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label putih	2. Sistem akan menampilkan daftar buku
3. Aktor melakukan <i>ceklist</i> pada <i>cekbox</i> yang terdapat pada daftar buku, kemudian menekan <i>button</i> cetak barcode dan label	4. Sistem akan menampilkan barcode dan label buku yang dipilih
5. Aktor menekan tombol Ctrl+p	6. Sistem akan melakukan proses cetak <i>barcode</i> dan label

Tabel 4.62 (lanjutan)

Skenario Alternatif 1 : Jika tidak melakukan <i>ceklist</i> pada <i>cekbox</i>	
1. Aktor menekan <i>tab</i> menu label putih	2. Sistem akan menampilkan daftar buku
3. Aktor tidak melakukan <i>ceklist</i> pada <i>cekbox</i> yang terdapat pada daftar buku, kemudian menekan <i>button</i> cetak barcode dan label	4. Sistem akan menampilkan pesan kesalahan <i>Harap pilih buku yang akan dicetak register dan labelnya!</i>
Konsisi Akhir	Sistem akan menampilkan <i>barcode</i> dan label yang akan dicetak atau menampilkan pesan kesalahan

59. Skenario *use case* cetak barcode dan label jurnal label merah

Tabel 4.63 Skenario *use case* cetak barcode dan label jurnal label merah

Kode Use Case	SKPL-07-03
Nama Use Case	Cetak barcode dan label jurnal label merah
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan cetak barcode dan label jurnal label merah
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan cetak barcode dan label jurnal label merah
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi jurnal
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label merah	2. Sistem akan menampilkan daftar kernal
3. Aktor melakukan <i>ceklist</i> pada <i>cekbox</i> yang terdapat pada daftar jurnal, kemudian menekan <i>button</i> cetak barcode dan label	4. Sistem akan menampilkan barcode dan label jurnal yang dipilih
5. Aktor menekan tombol Ctrl+p	6. Sistem akan melakukan proses cetak <i>barcode</i> dan label

Tabel 4.63 (lanjutan)

Skenario Alternatif 1 : Jika tidak melakukan <i>ceklist</i> pada <i>cekbox</i>	
1. Aktor menekan <i>tab</i> menu label merah	2. Sistem akan menampilkan daftar jurnal
3. Aktor tidak melakukan <i>ceklist</i> pada <i>cekbox</i> yang terdapat pada daftar jurnal, kemudian menekan <i>button</i> cetak barcode dan label	4. Sistem akan menampilkan pesan kesalahan <i>Harap pilih jurnal yang akan dicetak register dan labelnya!</i>
Konsisi Akhir	Sistem akan menampilkan <i>barcode</i> dan label yang akan di cetak

60. Skenario *use case* cetak barcode dan label jurnal label putih

Tabel 4.64 Skenario *use case* cetak barcode dan label jurnal label putih

Kode Use Case	SKPL-07-04
Nama Use Case	Cetak barcode dan label jurnal label putih
Aktor	taf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan cetak barcode dan label jurnal label putih
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan cetak barcode dan label jurnal label putih
Skenario Utama	
Kondisi Awal	Aktor sudah masuk halaman registrasi dan pelabelan.
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan <i>tab</i> menu label merah / label putih.	2. Sistem akan menampilkan daftar bahan pustaka.
3. Aktor melakukan <i>ceklist</i> pada <i>cekbox</i> yang terdapat pada daftar bahan pustaka, kemudian menekan <i>button</i> cetak barcode dan label.	4. Sistem akan menampilkan barcode dan label yang sudah dipilih.
5. Aktor menekan tombol Ctrl+p.	6. Sistem akan melakukan proses cetak <i>barcode</i> dan label.
Skenario Alternatif 1	
1. Jika aktor tidak melakukan <i>ceklist</i> pada <i>cekbox</i> salah satu bahan pustaka.	2. Sistem akan menampilkan pesan harap pilih buku yang akan dicetak register dan labelnya/ tambah copy/ ingin dihapus.
Konsisi Akhir	Sistem akan menampilkan <i>barcode</i> dan label yang akan di cetak.

61. Skenario *use case* validasi buku

Tabel 4.65 Skenario *use case* buku

Kode Use Case	SKPL-08-01
Nama Use Case	Vali Tabel 4.65 (lanjutan)
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan validasi buku
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan validasi buku
Skenario Utama	
Kondisi Awal	Aktor sudah masuk dalam halaman validasi buku
Aksi Aktor	Tanggapan dari sistem
1. Aktor memasukkan kode register buku 2. Aktor menekan tombol validasi	3. Sistem akan melakukan validasi buku 4. Sistem akan menampilkan informasi data buku yang divalidasi
Skenario Alternatif 1 : Jika <i>field</i> register kosong	
1. Aktor tidak memasukkan data pada <i>field</i> register 2. Aktor menekan tombol validasi	3. Sistem akan menampilkan pesan <i>field</i> register tidak boleh kosong
Skenario Alternatif 2 : Jika <i>field</i> register tidak diisi angka	
1. Aktor mengisi huruf pada <i>field</i> register 2. Aktor menekan tombol validasi	3. Sistem akan menampilkan pesan masukkan nomor
Konsisi Akhir	Sistem akan menampilkan data buku yang divalidasi atau pesan kesalahan

62. Skenario *use case* validasi jurnal

Tabel 4.66 Skenario *use case* jurnal

Kode Use Case	SKPL-08-02
Nama Use Case	Validasi jurnal
Aktor	Staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan validasi jurnal
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor melakukan validasi jurnal



Tabel 4.66 (lanjutan)

Skenario Utama	
Kondisi Awal	Aktor sudah masuk dalam halaman validasi jurnal
Aksi Aktor	Tanggapan dari sistem
1. Aktor memasukkan kode register jurnal 2. Aktor menekan tombol validasi	3. Sistem akan melakukan validasi jurnal 4. Sistem akan menampilkan informasi data jurnal yang divalidasi
Skenario Alternatif 1 : Jika <i>field</i> register kosong	
1. Aktor tidak memasukkan data pada <i>field</i> register 2. Aktor menekan tombol validasi	3. Sistem akan menampilkan pesan field register tidak boleh kosong
Skenario Alternatif 2 : Jika <i>field</i> register tidak diisi angka	
1. Aktor mengisi huruf pada field register 2. Aktor menekan tombol validasi	3. Sistem akan menampilkan pesan masukkan nomor
Konsisi Akhir	Sistem akan menampilkan data jurnal yang divalidasi atau pesan kesalahan

63. Skenario *use case* *logout*

Tabel 4.67 Skenario *logout*

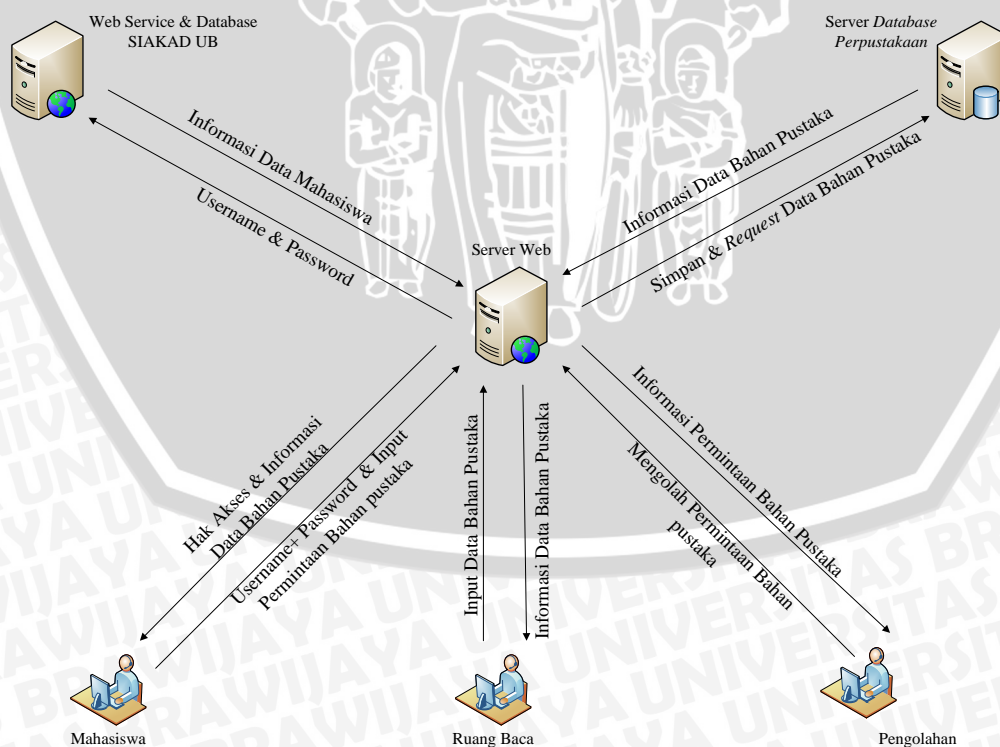
Kode Use Case	SKPL-09-01
Nama Use Case	<i>Logout</i>
Aktor	Anggota, staf IT, staf pengolahan dan staf ruang baca
Tujuan	Untuk melakukan <i>logout</i> .
Deskripsi	<i>Use case</i> ini menjelaskan bagaimana aktor keluar dari sistem atau <i>logout</i>
Skenario Utama	
Kondisi Awal	Aktor sudah dalam keadaan <i>login</i> dan sistem menampilkan halaman awal setelah <i>login</i>
Aksi Aktor	Tanggapan dari sistem
1. Aktor menekan ikon akun 2. Aktor menekan tombol <i>logout</i>	3. Sistem akan menampilkan menu <i>dropdown</i> 4. Sistem akan keluar dan menampilkan halaman awal sistem
Konsisi Akhir	Sistem akan menampilkan halaman <i>login</i> sistem

4.2 Perancangan Perangkat Lunak

Tahap perancangan sistem pada penelitian ini dilakukan dengan lima tahap pengerjaan yaitu merancang arsitektur umum dari sistem yang dibangun, melakukan pemodelan diagram *sequence* berdasarkan *usecase*, kemudian melakukan pemodelan diagram kelas, lalu dilanjutkan dengan pemodelan relasional data model dan pemodelan antarmuka sistem. Pada tahapan perancangan penelitian ini menggunakan pendekatan desain berorientasi objek yang direpresentasikan menggunakan UML (*Unified Modelling Language*).

4.2.1 Perancangan Arsitektural

Sistem perpustakaan terintegrasi ini menggunakan arsitektur *client-server* yang bertujuan untuk dapat diakses dari berbagai area. Terdapat lima bagian yaitu area mahasiswa, area ruang baca, area pengolahan dan 3 server. Mahasiswa sebagai anggota perpustakaan, ketika mahasiswa akan mengakses sistem perpustakaan maka akan dilakukan pengecekan *username* dan *password* pada server SIAKAD UB melalui *web service*. Ruang baca sebagai tempat untuk memasukkan data bahan pustaka, data yang dimasukkan oleh bagian ruang baca akan tersimpan pada server yang dipusatkan diperpustakaan. Pengolahan sebagai tempat pengolahan permintaan bahan pustaka maupun bahan pustaka baru. Kemudian terdapat 3 server yaitu server pertama sebagai server tempat aplikasi perpustakaan, server kedua sebagai tempat menyimpan *database* perpustakaan dan server ketiga yaitu server SIAKAD UB. Desain arsitektural sistem yang dikembangkan dapat dilihat pada **Gambar 4.3**.



Gambar 4.3 Arsitektur Umum Sistem

Berdasarkan desain arsitektural sistem pada **Gambar 4.3** proses integrasi yang terjadi diantaranya yaitu.

a. Integrasi Data Mahasiswa (Anggota)

Mahasiswa merupakan anggota perpustakaan, ketika mahasiswa mengakses sistem perpustakaan maka akan dilakukan pengecekan *username* dan *password* pada server SIAKAD UB melalui *web service*. Jika *username* dan *password* yang dimasukkan sesuai, maka sistem akan melakukan pengecekan data mahasiswa pada basis data perpustakaan pusat. Jika data mahasiswa belum ada pada basis data perpustakaan pusat maka data mahasiswa yang didapatkan melalui *web service* akan disimpan kedalam basis data perpustakaan.

b. Integrasi Data Permintaan Bahan Pustaka

Permintaan bahan pustaka dapat dilakukan oleh mahasiswa atau anggota, staf ruang baca dan staf pengolahan. Setiap permintaan bahan pustaka yang dilakukan, semua data akan disimpan pada basis data perpustakaan pusat. Data permintaan bahan pustaka akan diolah atau dikelola oleh staf pengolahan. Staf pengolahan akan memproses data permintaan sehingga menghasilkan bahan pustaka yang siap digunakan.

c. Integrasi Data Bahan Pustaka

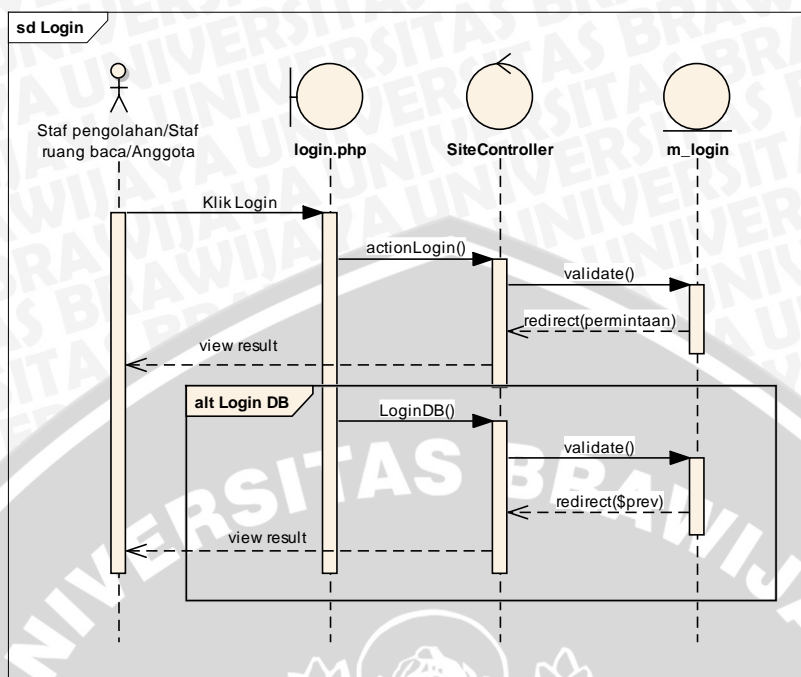
Data bahan pustaka yang berdapa disetiap ruang baca akan dimasukkan kedalam basis data perpustakaan pusta. Proses ini dilakukan agar data-data bahan pustaka tidak tersebar dan terpusat pada perpustakaan pusat. Data bahan pustaka yang sudah tersimpan pada basis data perpustakaan pusat dapat dengan mudah dalam proses pencariannya. Dengan demikian ketika kita melakukan pencarian bahan pustaka dengan menggunakan OPAC makan data bahan pustaka yang berada disetiap ruang baca dapat terdeteksi.

4.2.2 Perancangan Diagram *Sequence*

Sequence diagram menggambarkan interaksi antar masing-masing objek pada setiap usecase dalam urutan waktu. *Sequence diagram* digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai *respons* dari sebuah *event* untuk menghasilkan *output* tertentu. Pada tahap perancangan diagram *sequence* tidak semua kebutuhan fungsional akan dituliskan hanya menuliskan enam diagram *sequence* pada kasus ini.

1. Diagram *Sequence Login*

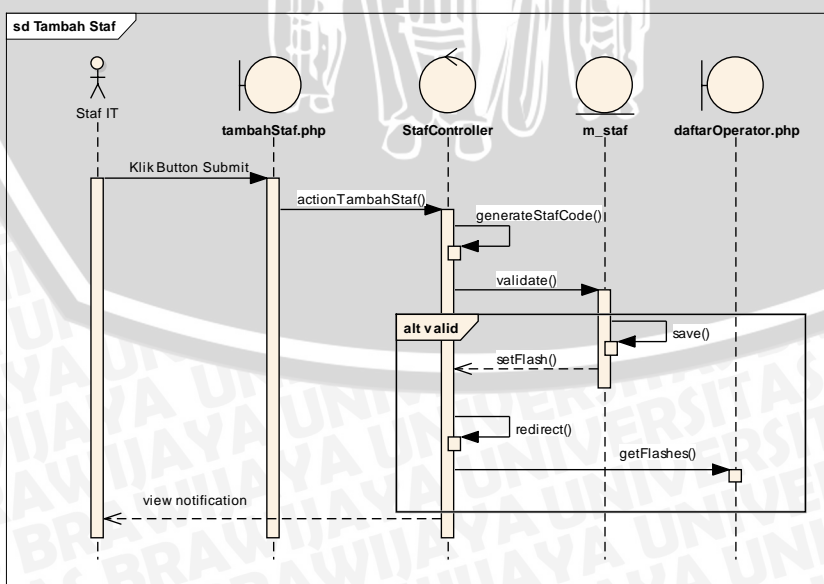
Diagram *sequence login* menjelaskan mengenai proses login yang dilakukan oleh aktor yaitu *staf IT*, staf pengolahan, staf ruang baca dan anggota. Proses *login* dilakukan dengan cara aktor memasukkan *username* dan *password* pada *userinterface* login, kemudian akan dilakukan pengecekan pada *controller SiteController* dan dilakukan validasi data pada *model m_login*. Perancangan diagram *sequence login* dapat dilihat pada **Gambar 4.4**.



Gambar 4.4 Diagram *sequence login*

2. Diagram *Sequence* Tambah Staf

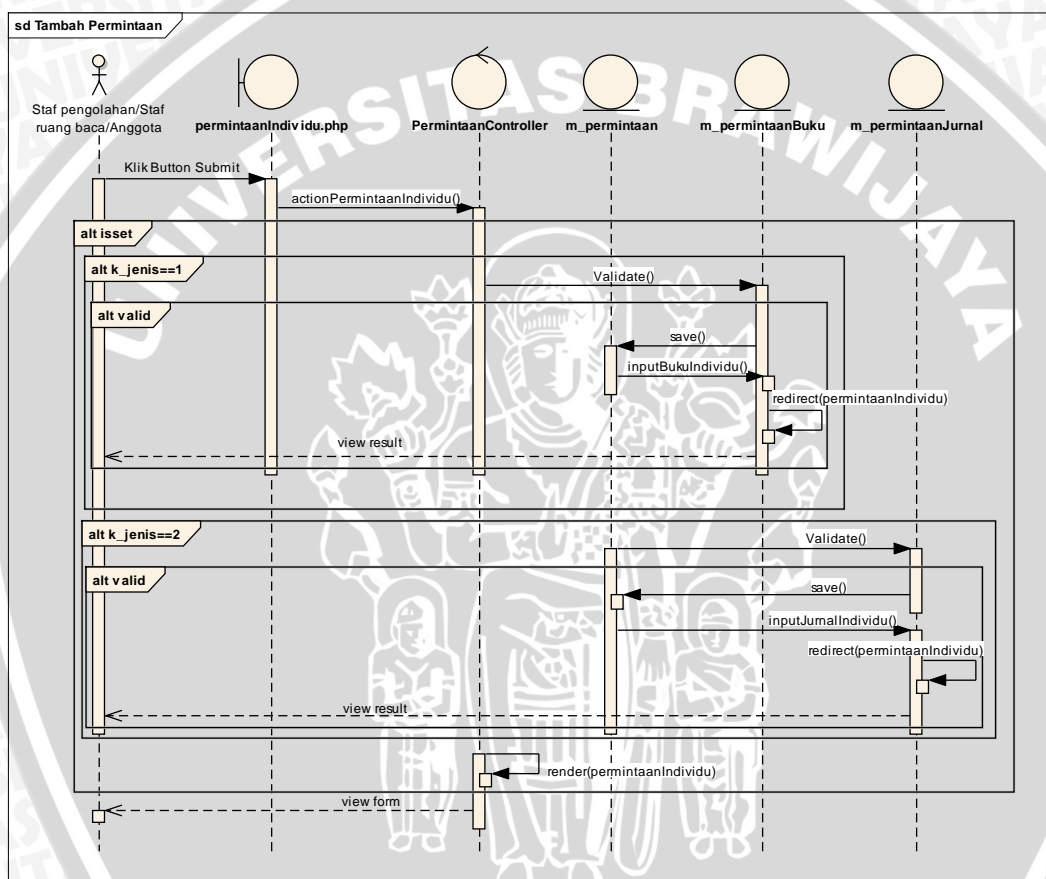
Diagram *sequence* tambah staf menjelaskan mengenai proses menambahkan staf. Proses yang dilakukan untuk menambahkan staf yaitu aktor staf IT mengisi data staf pada form, kemudian data akan diproses pada *StafController* dan akan dilakukan validasi pada model *m_staf*, jika data yang dimasukkan valid maka data disimpan. Setelah data tersimpan maka akan muncul pesan sukses dan *redirect* ke halaman daftar staf. Jika data yang dimasukkan tidak valid maka akan menampilkan notifikasi pada view tambah staf. Perancangan diagram *sequence* tambah staf dapat dilihat pada **Gambar 4.5**.



Gambar 4.5 Diagram *sequence* tambah staf

3. Diagram Sequence Tambah Permintaan Buku dan jurnal

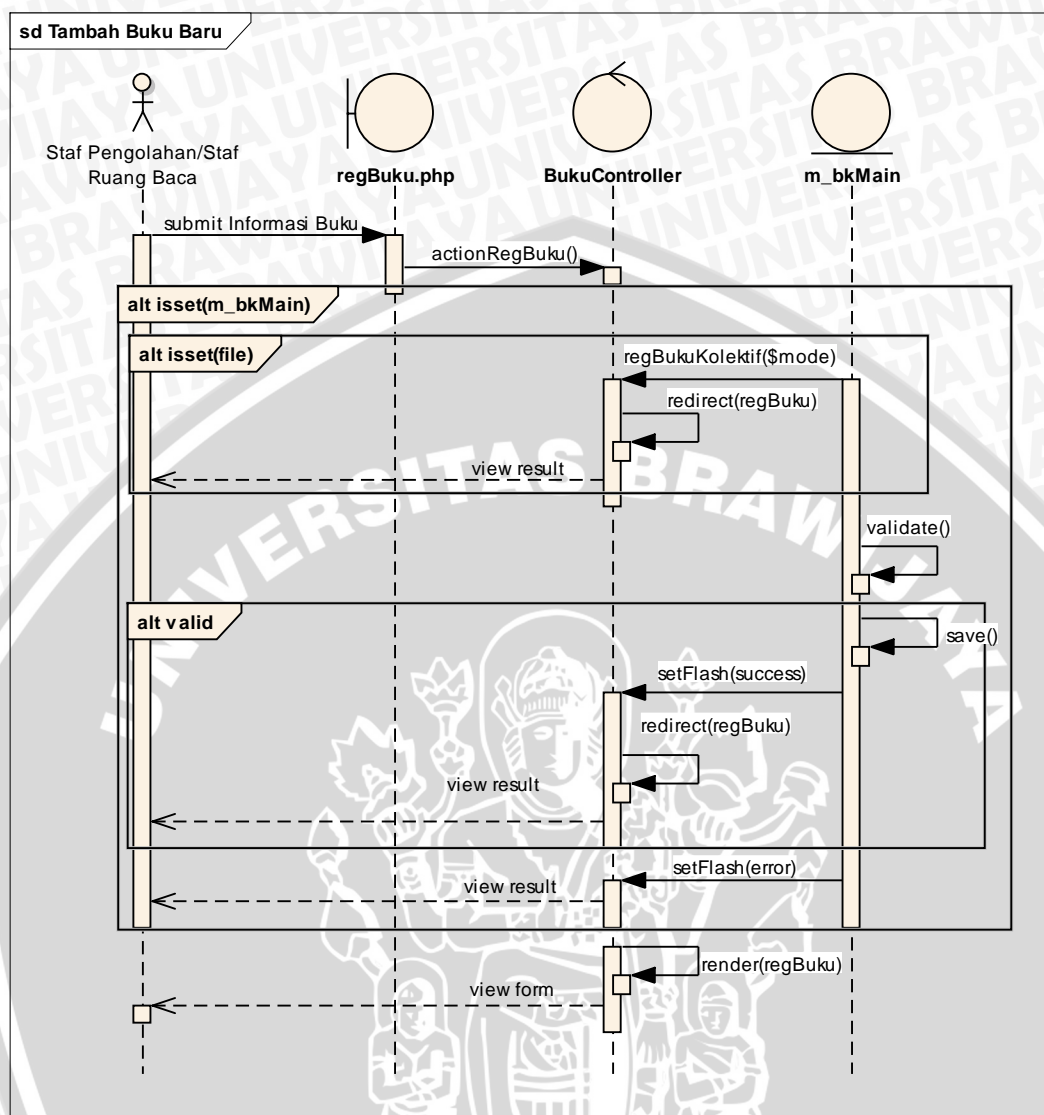
Diagram *sequence* tambah permintaan buku dan jurnal menjelaskan mengenai proses menambah data permintaan buku dan jurnal. Proses yang dilakukan yaitu aktor mengisi form permintaan buku atau jurnal, kemudian menekan *submit*. Selanjutnya data akan diproses pada *PermintaanController* untuk menyimpan data permintaan. Jika yang diisi form buku maka akan menjalankan proses penyimpanan buku. Jika yang diisi form jurnal maka akan menjalankan proses penyimpanan jurnal. Perancangan diagram *sequence* tambah permintaan buku dan jurnal dapat dilihat pada **Gambar 4.6**.



Gambar 4.6 Diagram *sequence* tambah permintaan buku dan jurnal

4. Diagram Sequence Tambah Buku Baru

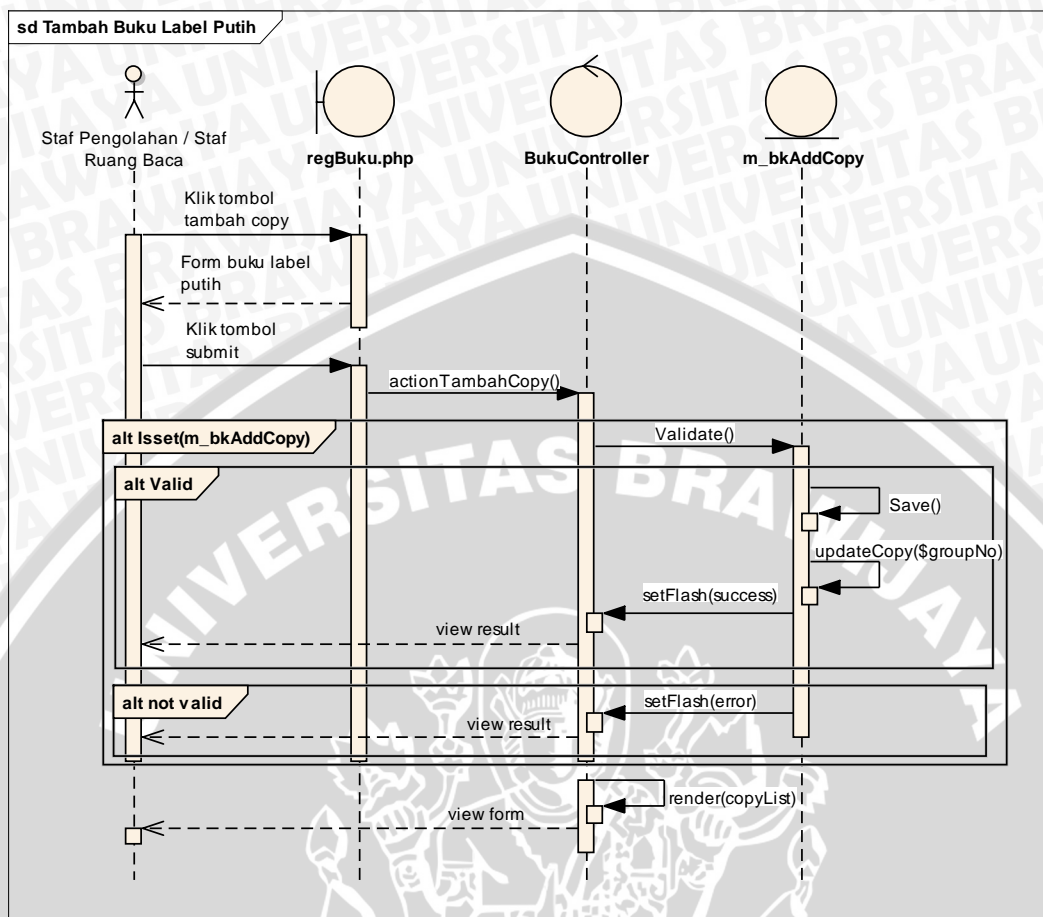
Diagram *sequence* tambah buku baru menjelaskan mengenai proses menambah data buku secara individu maupun kolektif. Proses yang dilakukan yaitu aktor mengisi form atau mengupload file excel, kemudian dilanjutkan dengan menekan *submit*. Data akan diproses pada *bukuController*. Setelah data tersimpan maka akan ditampilkan pesan informasi data tersimpan atau gagal tersimpan. Perancangan diagram *sequence* tambah buku dapat dilihat pada **Gambar 4.7**.



Gambar 4.7 Diagram *sequence* tambah bahan pustaka buku baru

5. Diagram *Sequence* Tambah Buku Label Putih (*Copy*)

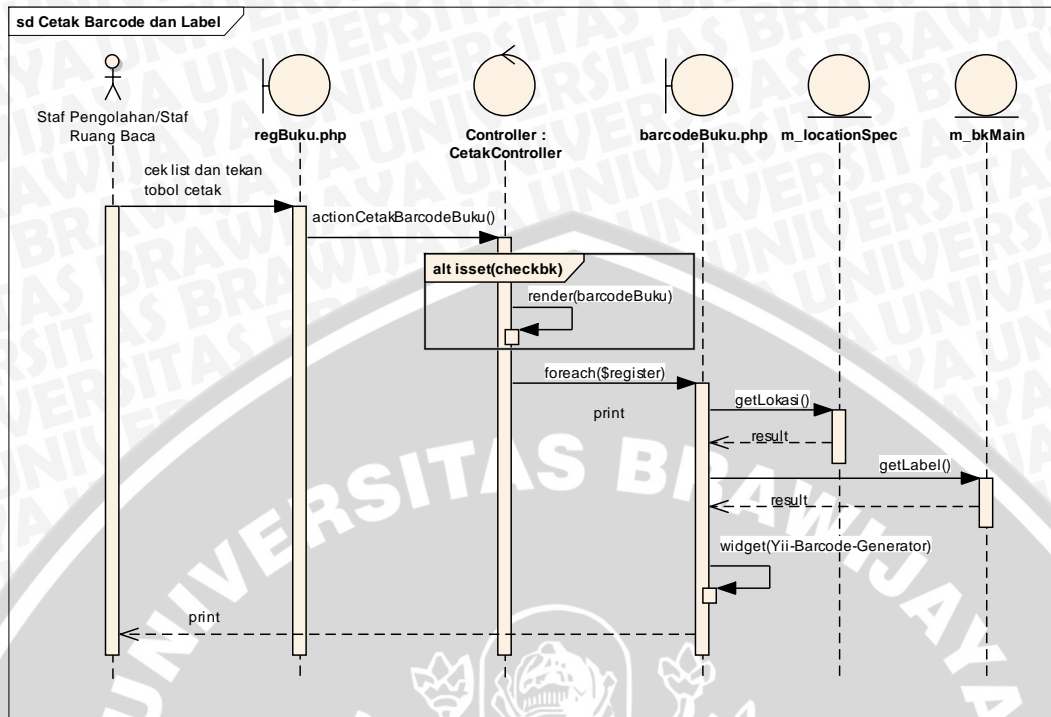
Diagram *sequence* tambah buku label putih atau *copy* bahan pustaka menjelaskan mengenai proses menambah data bahan pustaka yang sudah pernah dimasukkan kedalam sistem. Proses yang dilakukan yaitu aktor mengisi form bahan pustaka, kemudian menekan tombol *submit*. Selanjutnya data akan diproses pada *bukuController* untuk menyimpan data tersebut dengan cara memanggil fungsi *save()* yang dimiliki *model m_bkAddCopy*. Dilanjutkan dengan *update* jumlah *copy* berdasarkan kode group bahan pustaka. Setelah data tersimpan maka akan di tampilkan pesan informasi keberhasilan menyimpan data. Perancangan diagram *sequence* tambah bahan pustaka label putih dapat dilihat pada **Gambar 4.8**.



Gambar 4.8 Diagram *sequence* tambah buku label putih (*copy*)

6. Diagram *Sequence* Cetak Barcode dan Label Buku Label Merah

Diagram *sequence* cetak barcode dan label buku label merah menjelaskan mengenai proses mencetak barcode dan label bahan pustaka buku yang berlabel merah. Proses yang dilakukan yaitu aktor membuka halaman daftar bahan pustaka, kemudian *cetakController* melakukan suatu proses untuk melakukan *render* ke barcode, kemudian kemudian mengambil data lokasi dan label untuk ditampilkan pada *view*, ditampilkan dalam bentuk barcode dengan menggunakan *widget(Yii-Barcode-Generator)*. Perancangan diagram *sequence* cetak barcode dan label bahan pustaka dapat dilihat pada **Gambar 4.9**.



Gambar 4.9 Diagram *sequence* cetak barcode dan label

4.2.3 Perancangan Diagram *Class*

Diagram *class* memberikan gambaran pemodelan elemen-elemen *class* yang membentuk sebuah sistem. *Class* bisa didapatkan dengan menganalisis secara detail terhadap *use case* yang dimodelkan. Perancangan diagram *class* ini dilakukan sebanyak 3 kali iterasi sesuai dengan perubahan pada kebutuhan sistem. Penambahan kelas dan method pada iterasi pertama dapat dilihat pada **Tabel 4.68**, iterasi kedua tidak mengalami perubahan diagram *class*. Sedangkan iterasi ketiga dapat dilihat pada **Tabel 4.69**. Perancangan keseluruhan diagram *class* dijelaskan pada sub bab dibawah ini.

Tabel 4.68 Penambahan kelas dan method pada iterasi 1

Nama Kelas	Nama Method
PermintaanController	+ BacaPermintaanBuku() + BacaPermintaanJurnal() +actionPermintaanKolektif()
m_permintaanBuku	+ tambahBuku() + bukuSemuaBahasa() + bukuBerdasarkanBahasa(\$bahasa)
m_permintaanJurnal	+ tambahJurnal() + jurnalSemuaBahasa() + jurnalBerdasarkanBahasa(\$bahasa)

Tabel 4.68 (Lanjutan)

Nama Kelas	Nama Method
BukuController	+ regBukuKolektif()
JurnalController	+ regJurnalKolektif()
CetakController	+ actionCetakBuku() + actionCetakJurnal()

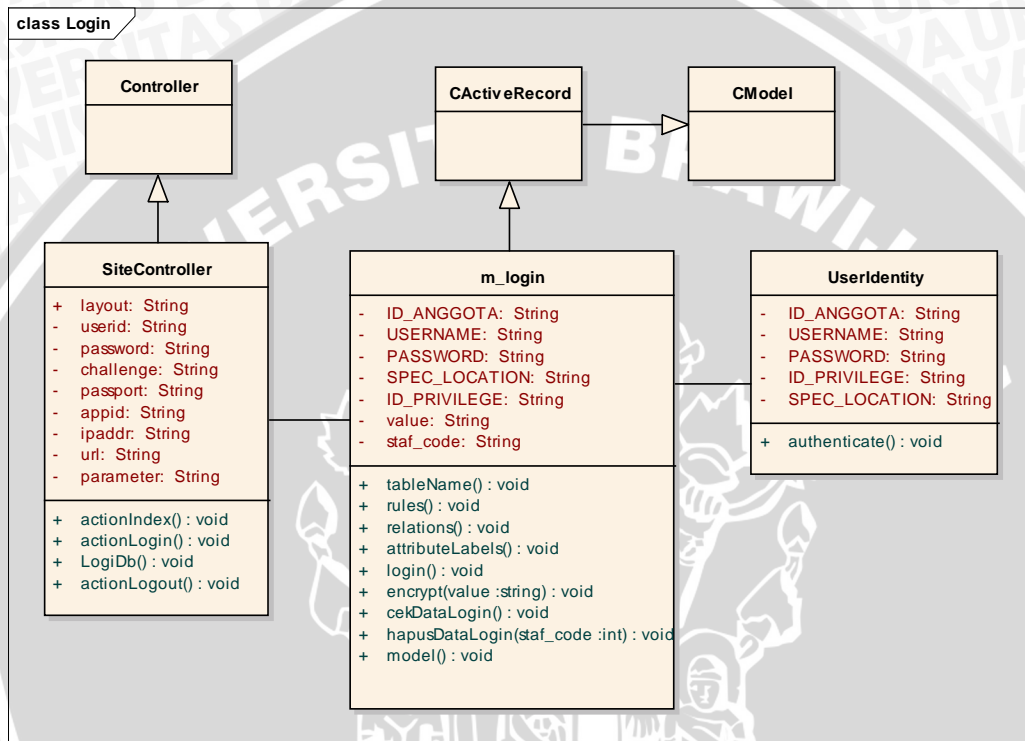
Tabel 4.69 penambahan kelas dan method pada iterasi 3

Nama Kelas	Nama Method
PengolahanController	+ actionCarilsbnBuku() + actionCarilssnJurnal()
m_permintaanBuku	+ cariBuku(\$isbn)
m_permintaanJurnal	+ cariJurnal(\$issn)
BukuController	+ actionSearchMerah() + actionSearchPutih()
JurnalController	+ actionSearchMerah() + actionSearchPutih()
m_bkMain	+ cariJudul(\$cari,\$location) + cariRegister(\$cari,\$location) + carilsbn(\$cari,\$location) + cariPenerbit(\$cari,\$location) + bkMainAll(\$location)
m_bkAddCopy	+ cariJudul(\$cari,\$location) + cariRegister(\$cari,\$location) + carilsbn(\$cari,\$location) + cariPenerbit(\$cari,\$location) + bkCopyAll(\$location)
m_rrAddCopy	+ cariJudul(\$cari,\$reg,\$location) + cariRegister(\$cari,\$location) + carilssn(\$cari,\$location) + cariPenerbit(\$cari,\$location) + dataCopy(\$reg,\$location)

4.2.3.1 Perancangan Diagram Class Keseluruhan

1. Class Diagram Login

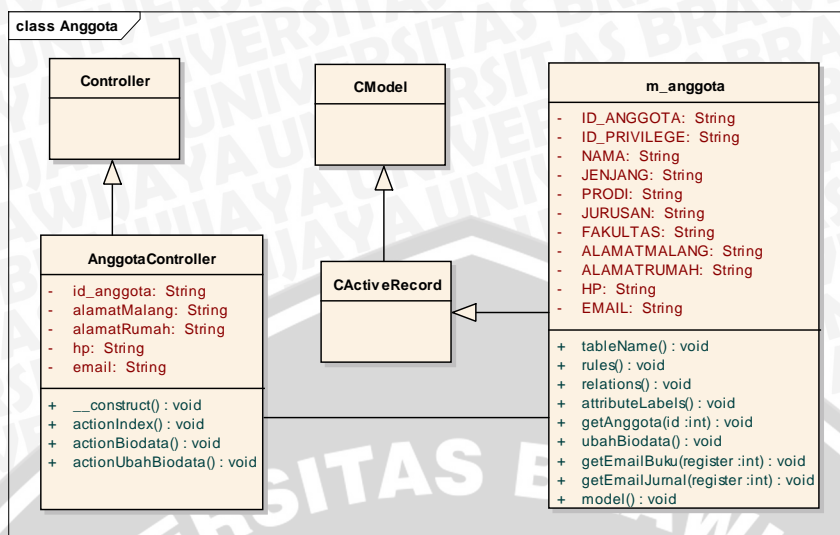
Class diagram login merupakan rancangan kelas-kelas pada menu *login* yang nantinya akan diimplementasikan pada saat pembuatan aplikasi. Terdapat 2 kelas *controller* yaitu kelas *controller* YII sebagai generasilasi dari *SiteController*. Kemudian terdapat kelas model diantaranya, *Cmodel*, *CactiveRecord*, *m_login*. Adapun rancangan *class diagram login* dapat dilihat pada **Gambar 4.10**.



Gambar 4.10 Class diagram login

2. Class Diagram Anggota

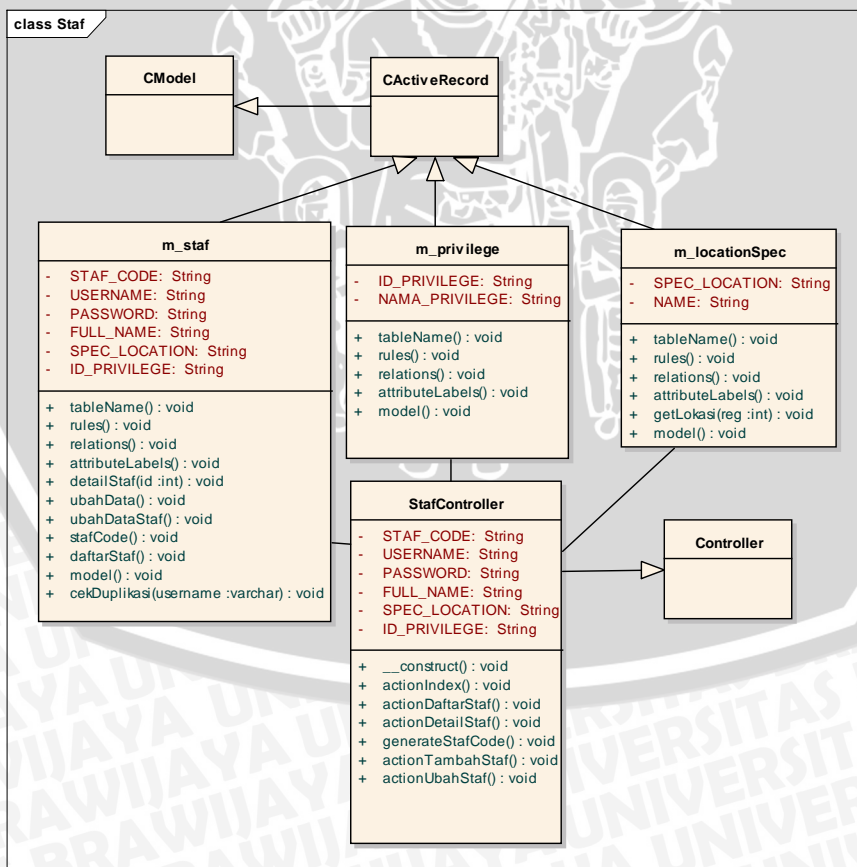
Class diagram anggota merupakan rancangan kelas-kelas pada fitur anggota yang nantinya akan diimplementasikan pada saat pembuatan aplikasi. Pada kelas diagram ini terdapat beberapa kelas yaitu kelas *Controller* yang merupakan generalisasi dari kelas *AnggotaController*, kemudian terdapat kelas *CModel* yang merupakan generalisasi dari kelas *CactiveRecord* dan *m_anggota*. Adapun rancangan *class diagram* anggota dapat dilihat pada **Gambar 4.11**.



Gambar 4.11 Class diagram anggota

3. Class Diagram Staf

Class diagram staf merupakan rancangan kelas-kelas pada fitur operator atau staf ruang baca yang nantinya akan diimplementasikan pada saat pembuatan aplikasi. Adapun rancangan class diagram staf dapat dilihat pada Gambar 4.12.

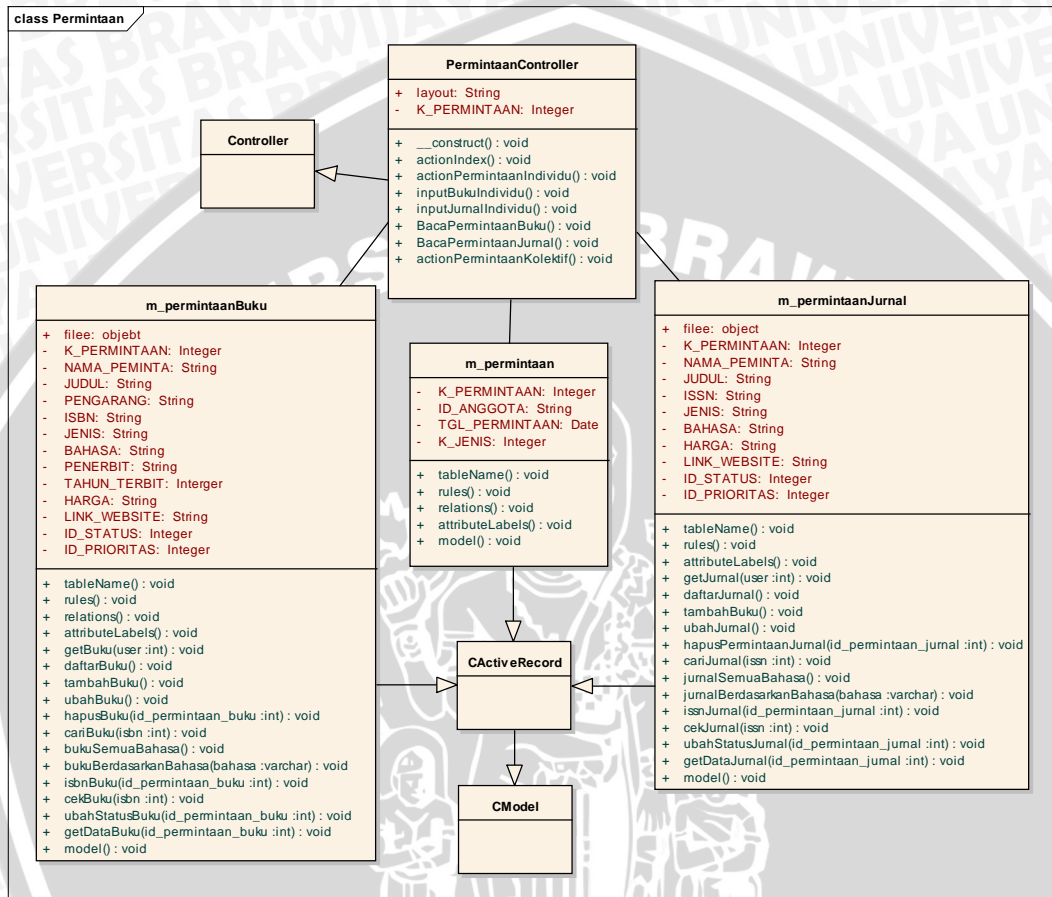


Gambar 4.12 Class diagram staf



4. Class Diagram Permintaan Bahan Pustaka

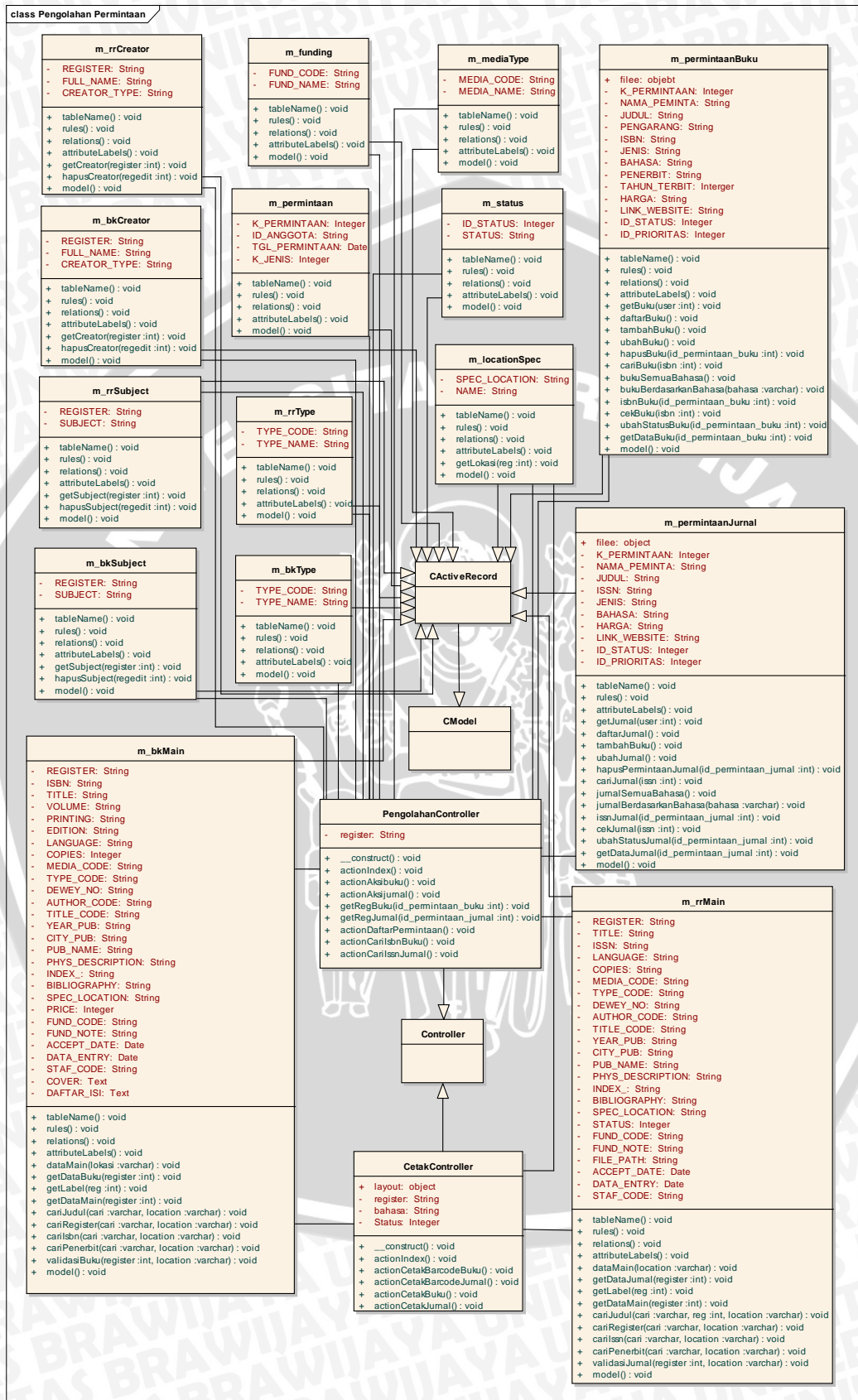
Class diagram permintaan bahan pustaka merupakan rancangan kelas-kelas pada fitur permintaan bahan pustaka yang nantinya akan diimplementasikan pada saat pembuatan aplikasi. Adapun rancangan class diagram permintaan bahan pustaka dapat dilihat pada Gambar 4.13.



Gambar 4.13 Class diagram permintaan bahan pustaka

5. Class Diagram Pengolahan Permintaan Bahan Pustaka

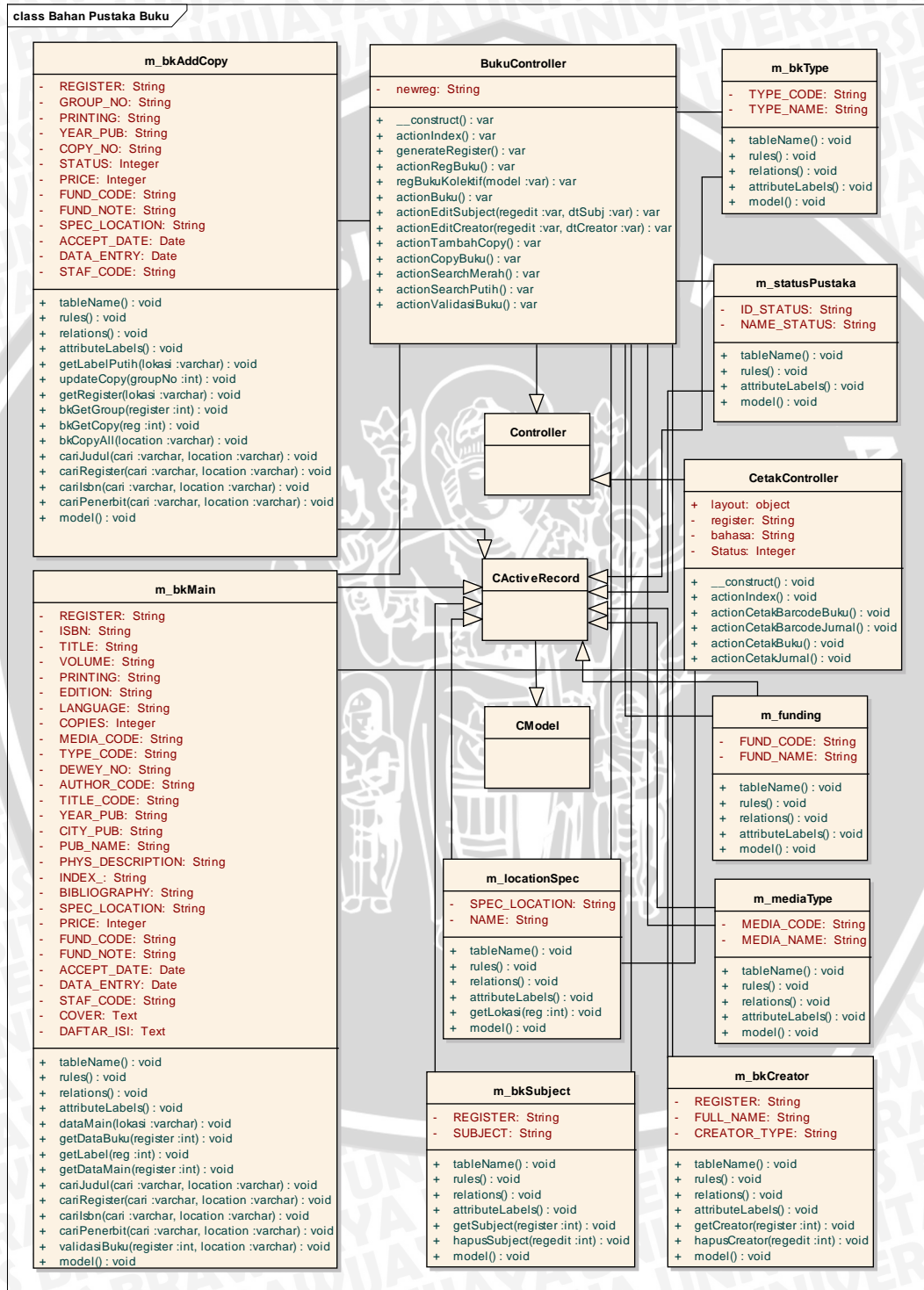
Class diagram pengolahan permintaan bahan pustaka merupakan rancangan kelas-kelas yang nantinya akan diimplementasikan pada saat pembuatan aplikasi. Adapun rancangan class diagram pengolahan permintaan bahan pustaka dapat dilihat pada Gambar 4.14.



Gambar 4.14 Class diagram pengolahan permintaan bahan pustaka

6. Class Diagram Bahan Pustaka Buku

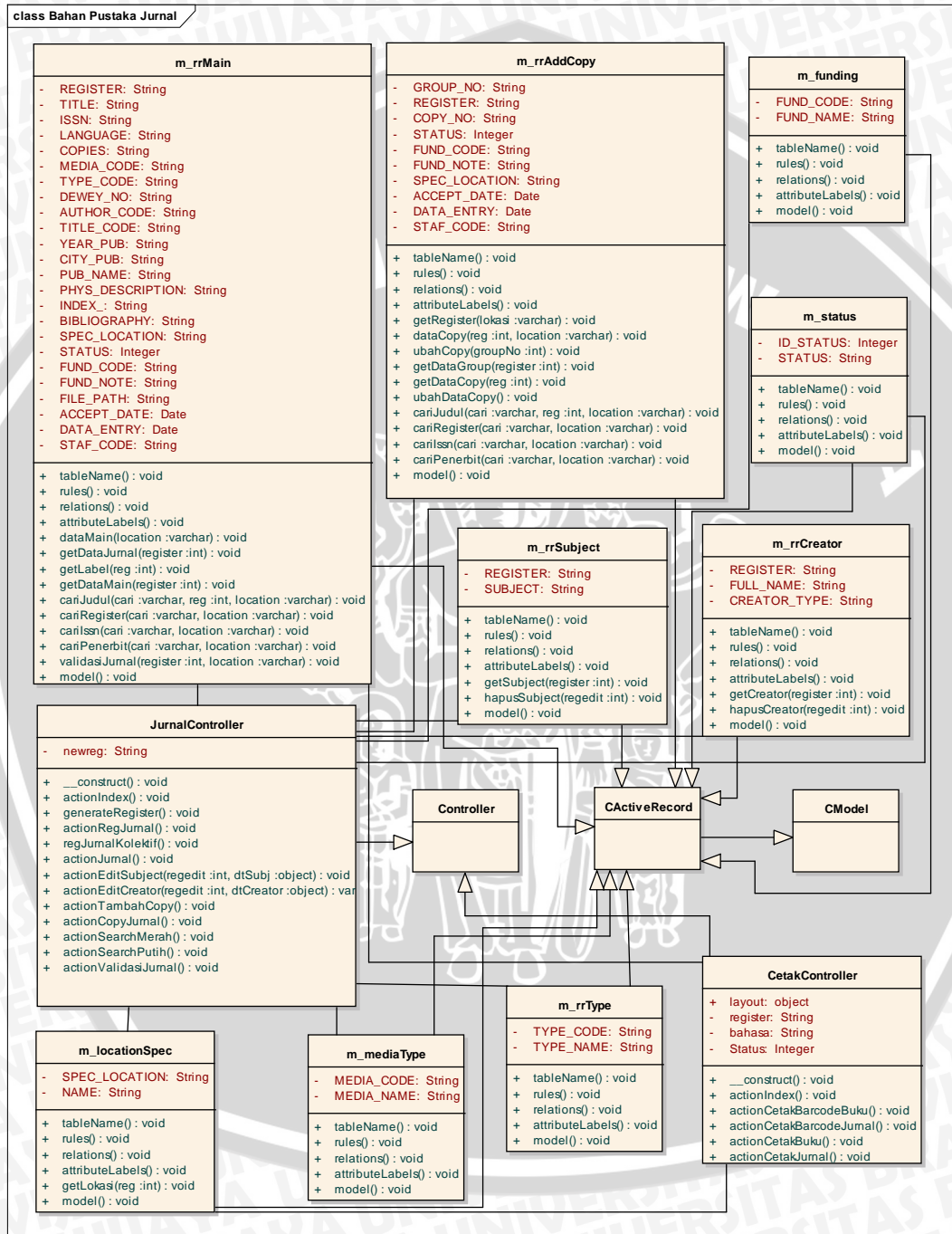
Class diagram bahan pustaka buku merupakan rancangan kelas-kelas yang nantinya akan diimplementasikan pada saat pembuatan aplikasi. Adapun rancangan class diagram bahan pustaka buku dapat dilihat pada **Gambar 4.15**.



Gambar 4.15 Class diagram bahan pustaka buku

7. Class Diagram Bahan Pustaka Jurnal

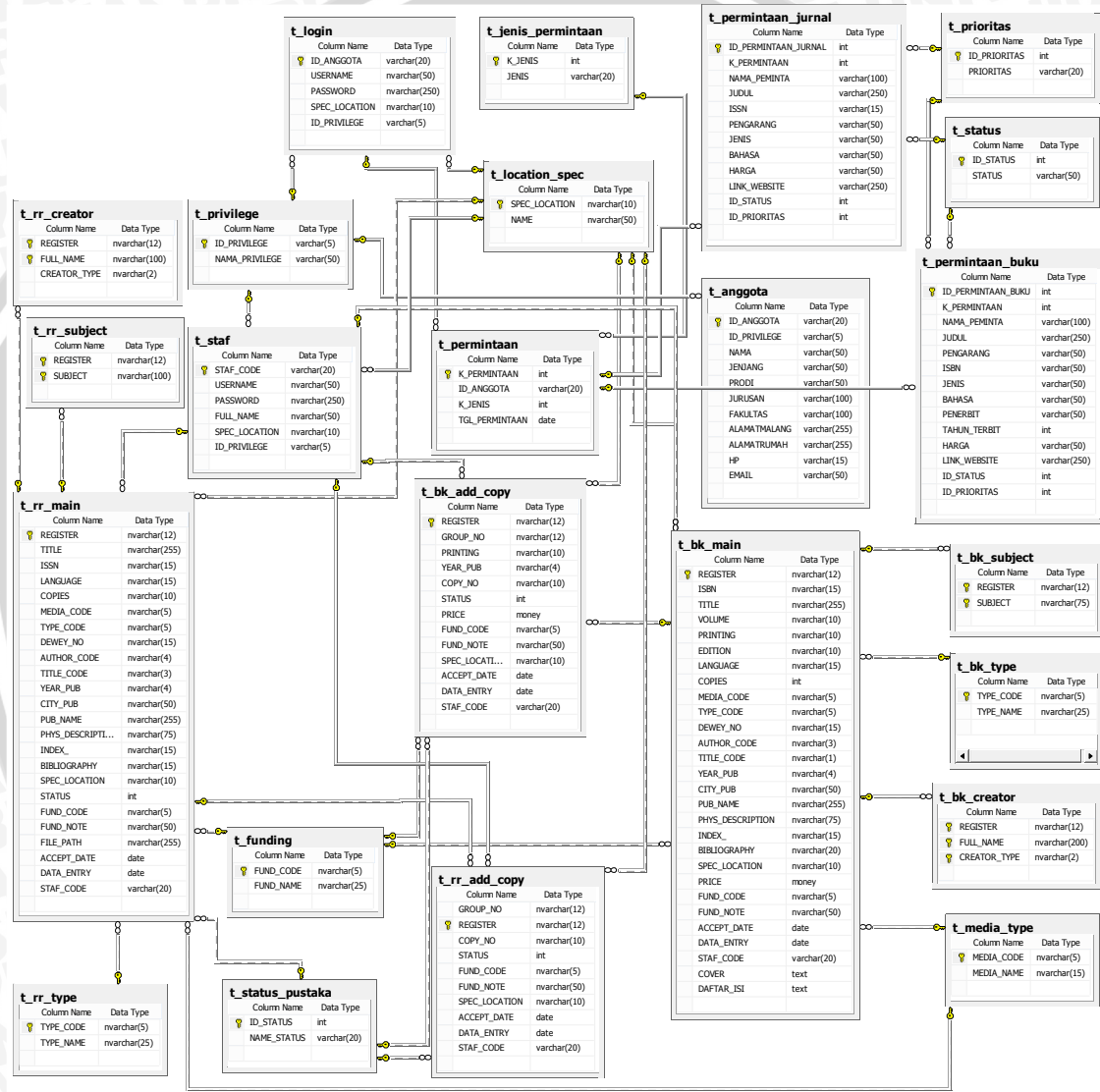
Class diagram bahan pustaka jurnal merupakan rancangan kelas-kelas yang nantinya akan diimplementasikan pada saat pembuatan aplikasi. Adapun rancangan class diagram bahan pustaka jurnal dapat dilihat pada Gambar 4.16.



Gambar 4.16 Class diagram bahan pustaka jurnal

4.2.4 Model Data Relasional

Model data relasional sering juga disebut sebagai model relasional atau basis data relasional digunakan untuk memodelkan data sesuai dengan implementasinya pada *database*. Model relasional berupa tabel-tabel yang saling berelasi. Setiap tabel mempunyai sejumlah kolom dimana setiap kolom memiliki nama yang unik. Model relasional pada aplikasi yang akan dibangun dapat dilihat pada **Gambar 4.17**.



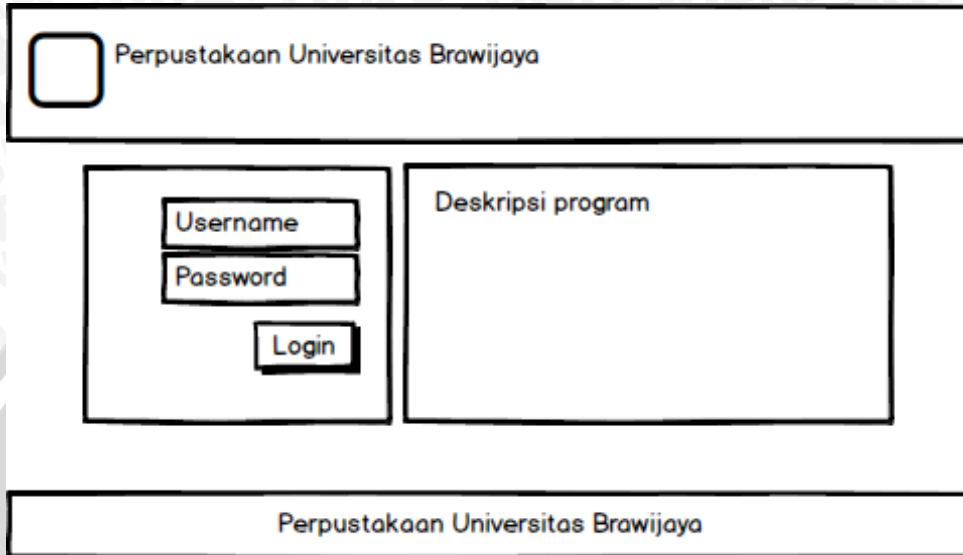
Gambar 4.17 Relational Model

4.2.5 Perancangan Prototipe

Pada bagian ini akan dijelaskan tentang perancangan prototipe perangkat lunak yang akan dibangun. Berikut merupakan rancangan prototipe dari perangkat lunak yang akan dibangun.

1. Rancangan Prototipe Halaman *Login*

Halaman login merupakan halaman awal dari sistem yang berisi form login dan deskripsi sistem. Berikut ini merupakan rancangan prototipe halaman login pada **Gambar 4.18**.

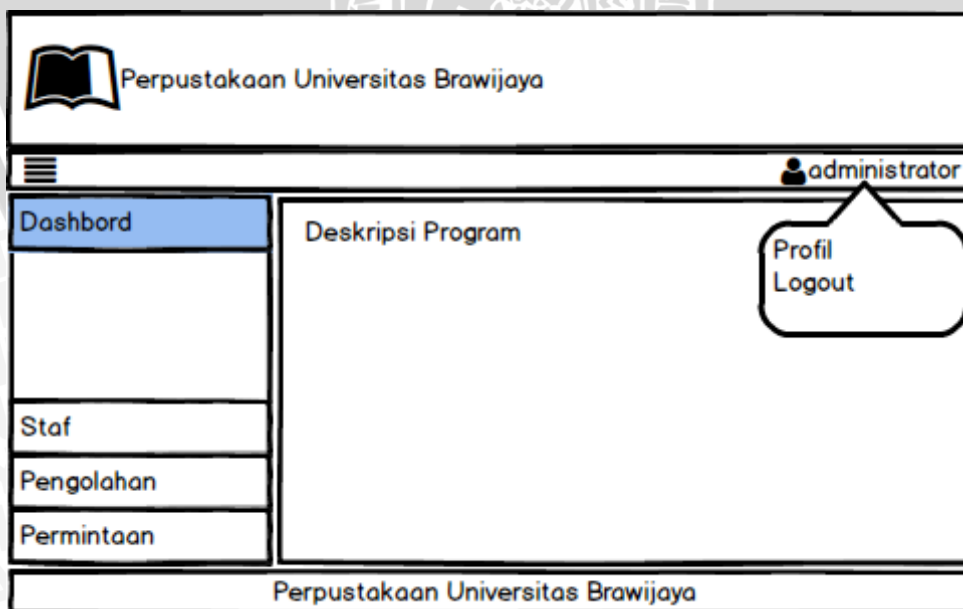


The image shows a wireframe for a login page. At the top, there is a header bar with a square logo on the left and the text "Perpustakaan Universitas Brawijaya" on the right. Below the header, the main content area is divided into two columns. The left column contains three stacked input fields labeled "Username", "Password", and a "Login" button. The right column is a large empty box labeled "Deskripsi program". At the bottom, there is a footer bar with the text "Perpustakaan Universitas Brawijaya".

Gambar 4.18 Rancangan prototipe halaman *login*

2. Rancangan Prototype Halaman Utama

Halaman ini merupakan halaman utama dari sistem yang berisi menu dan deskripsi sistem. Berikut ini merupakan rancangan prototipe halaman utama pada **Gambar 4.19**.



The image shows a wireframe for a main dashboard page. At the top, there is a header bar with a book icon on the left and the text "Perpustakaan Universitas Brawijaya" on the right. Below the header, there is a navigation bar with a hamburger menu icon on the left and a user profile icon labeled "administrator" on the right. The main content area is divided into two columns. The left column contains a vertical menu with four items: "Dashbord" (highlighted in blue), "Staf", "Pengolahan", and "Permintaan". The right column is a large empty box labeled "Deskripsi Program". In the top right corner of the main content area, there is a callout box containing the text "Profil" and "Logout". At the bottom, there is a footer bar with the text "Perpustakaan Universitas Brawijaya".

Gambar 4.19 Rancangan prototipe halaman utama

3. Rancangan Prototype Halaman Staf

Halaman ini merupakan halaman utama dari sistem yang berisi menu *dropdown* dan form tambah staf dan daftar staf. Berikut ini merupakan rancangan prototipe halaman tambah staf pada **Gambar 4.20** dan daftar staf **Gambar 4.21**

Gambar 4.20 Rancangan prototipe halaman tambah staf

Kode Staf	Username	Nama	Lokasi	Bagian	aksi
OP00	staf00	Fasial	Perpus Pusat	Pengolahan	<input type="checkbox"/>
OP01	staf01	Dinan	Ruang Baca PTIIC	Riang Baca	<input type="checkbox"/>

Gambar 4.21 Rancangan prototipe halaman daftar staf

4. Rancangan Prototype Biodata Anggota

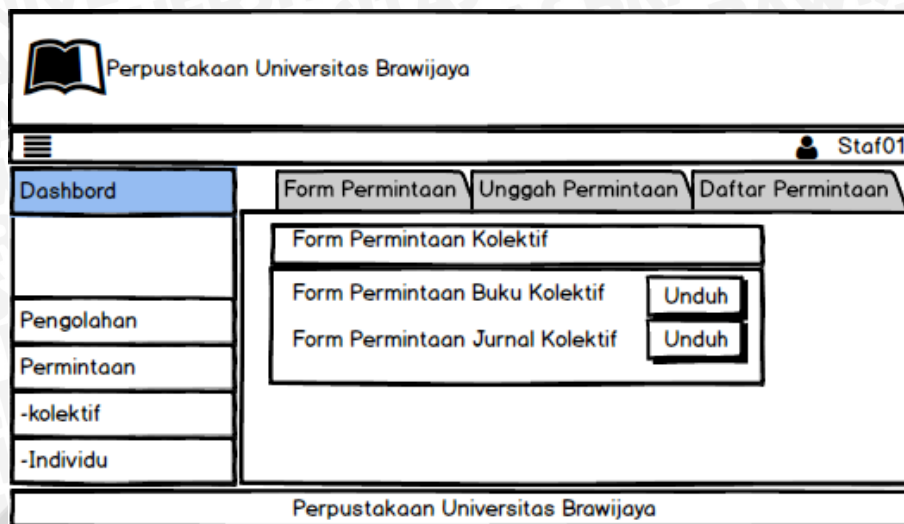
Halaman ini merupakan halaman biodata anggota perpustakaan. Berikut ini rancangan prototipe halaman biodata anggota pada **Gambar 4.22**.

Gambar 4.22 Rancangan prototipe halaman biodata anggota

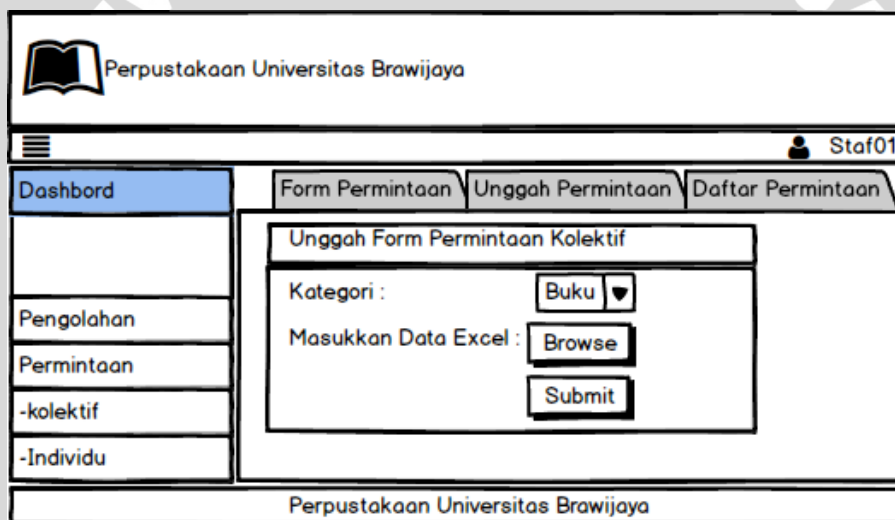
5. Rancangan Prototipe Halaman Permintaan Bahan Pustaka

Halaman ini merupakan halaman permintaan bahan pustaka yang meliputi buku dan jurnal. Berikut ini merupakan rancangan prototipe halaman permintaan bahan pustaka pada **Gambar 4.23**. Rancangan prototipe halaman unduh form permintaan kolektif pada **Gambar 4.24**. Rancangan prototipe halaman unggah form permintaan kolektif pada **Gambar 4.25**.

Gambar 4.23 Rancangan prototipe halaman permintaan individu



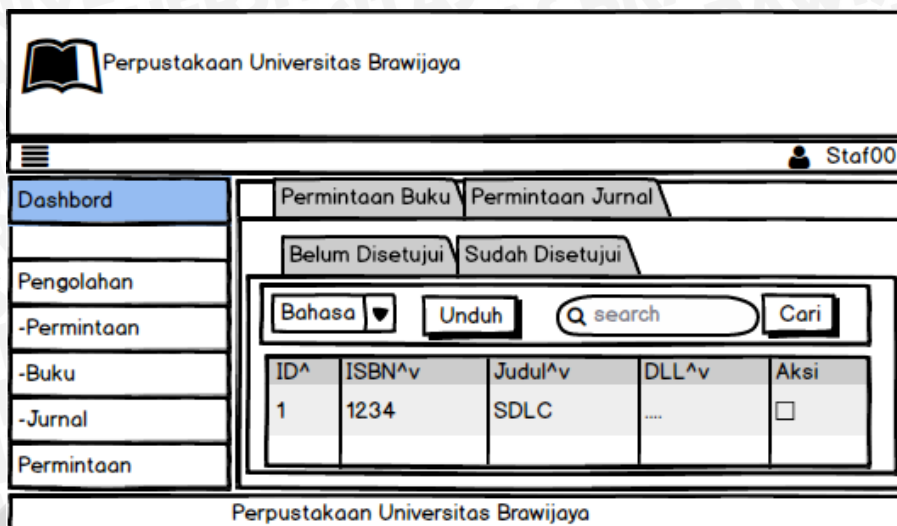
Gambar 4.24 Rancangan prototipe halaman unduh form permintaan kolektif



Gambar 4.25 Rancangan prototipe halaman unggah form permintaan

6. Rancangan Prototipe Halaman Pengolahan Permintaan Bahan Pustaka

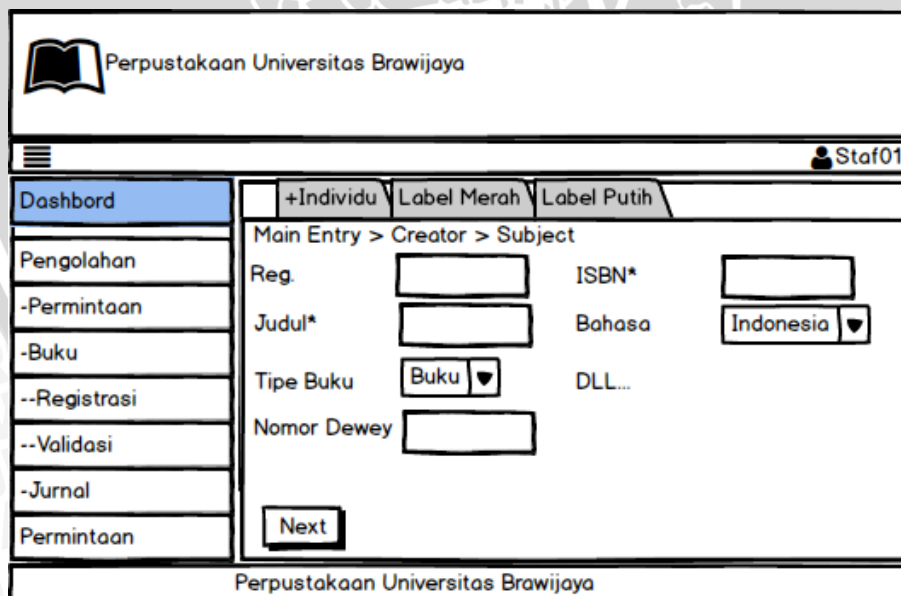
Halaman ini merupakan halaman pengolahan permintaan bahan pustaka, halaman ini berisi menu *dropdown*, *tab* menu, dan daftar permintaan bahan pustaka. Berikut ini merupakan rancangan prototipe halaman pengolahan permintaan bahan pustaka pada **Gambar 4.26**.



Gambar 4.26 Rancangan prototipe halaman pengolahan permintaan

7. Rancangan Prototipe Halaman Pengolahan Bahan Pustaka

Halaman ini merupakan halaman pengolahan bahan pustaka yang meliputi buku dan jurnal. Berikut ini merupakan rancangan prototipe halaman registrasi buku pada **Gambar 4.27**. Rancangan prototipe halaman registrasi jurnal pada **Gambar 4.28**. Rancangan prototipe halaman registrasi buku dan jurnal secara kolektif pada **Gambar 4.29** dan **Gambar 4.30**. Rancangan prototipe halaman daftar bukudan jurnal pada **Gambar 4.31** dan **Gambar 4.32**. Rancangan prototipe halaman validasi buku dan jurnal pada **Gambar 4.33** dan **Gambar 4.34**.



Gambar 4.27 Rancangan prototipe halaman registrasi buku

Perpustakaan Universitas Brawijaya

Staf01

Dashboard +Individu Label Merah Label Putih

Main Entry > Creator > Subject

Reg. ISSN*

Judul* Bahasa

Tipe Jurnal DLL...

Nomor Dewey

Next

Perpustakaan Universitas Brawijaya

Gambar 4.28 Rancangan prototipe halaman registrasi jurnal

Perpustakaan Universitas Brawijaya

Staf01

Dashboard +Individu +kolektif Label Merah Label Putih

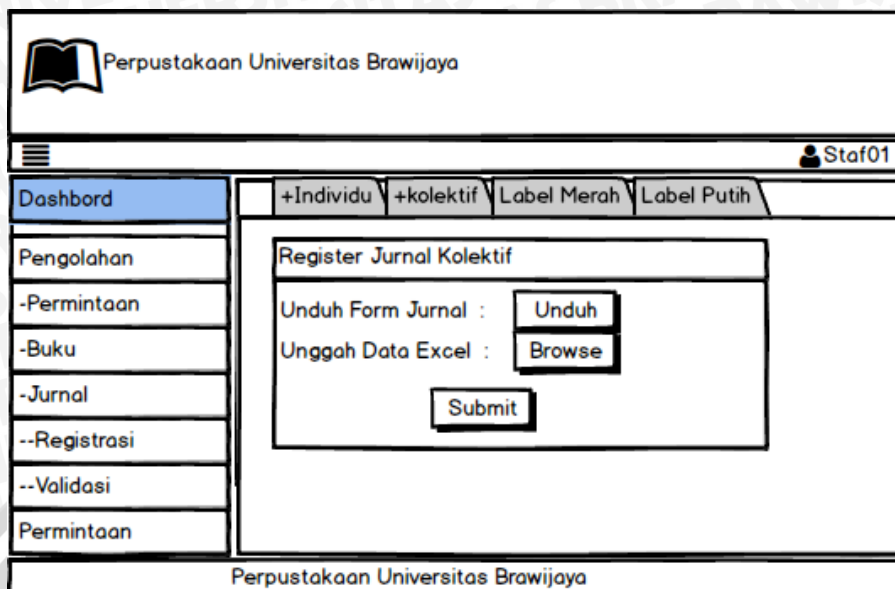
Register Buku Kolektif

Unduh Form Buku :

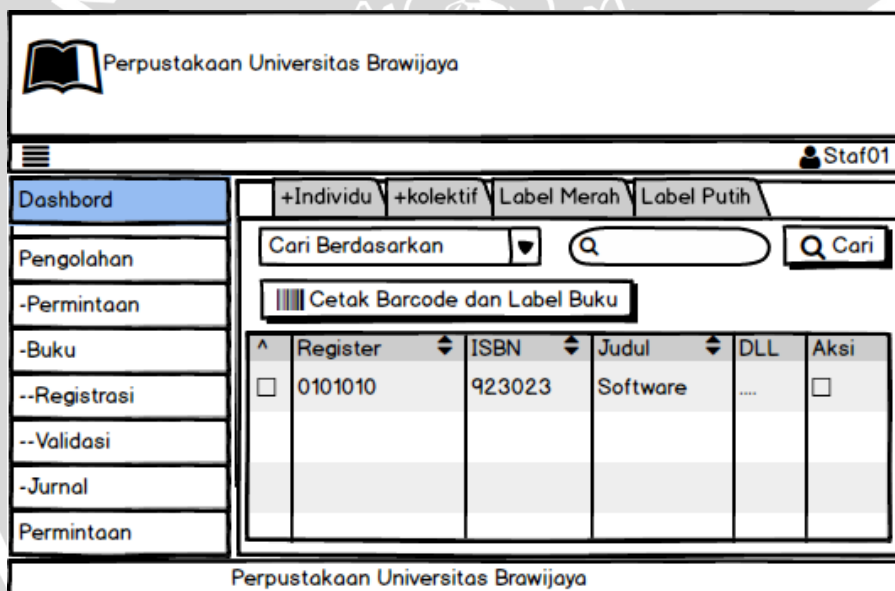
Unggah Data Excel :

Perpustakaan Universitas Brawijaya

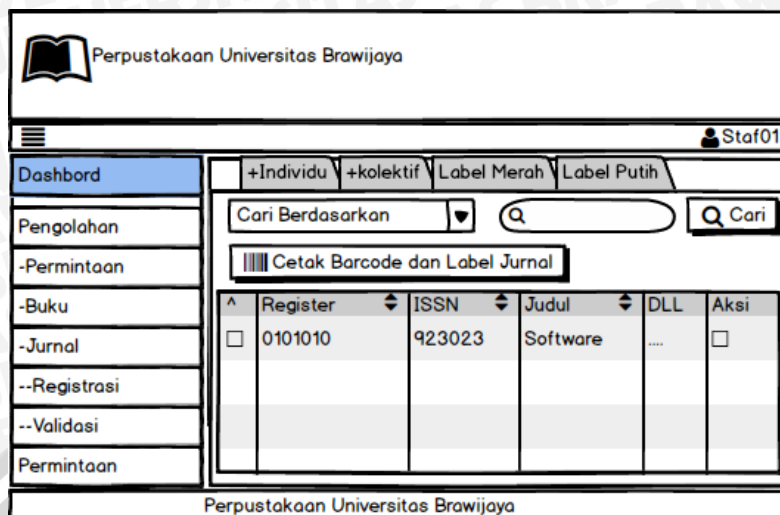
Gambar 4.29 Rancangan prototipe halaman registrasi buku secara kolektif



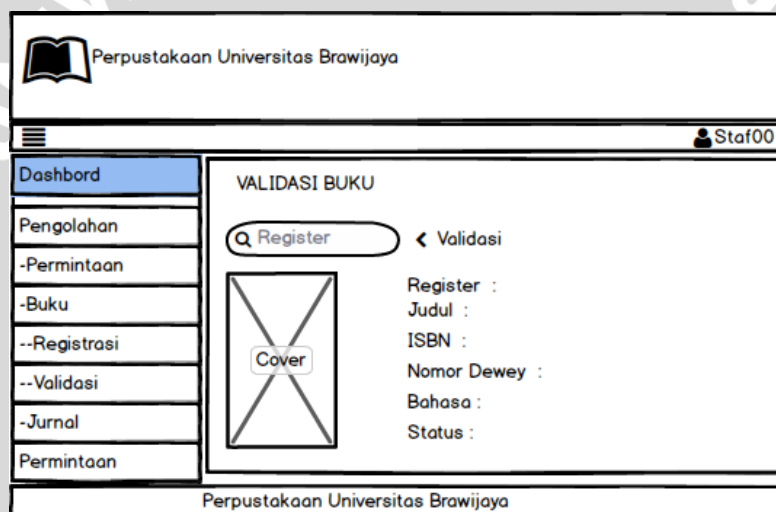
Gambar 4.30 Rancangan prototipe halaman registrasi jurnal secara kolektif



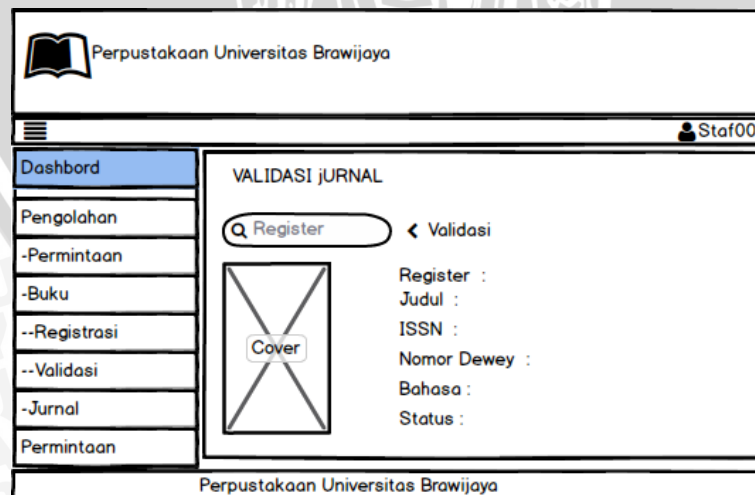
Gambar 4.31 Rancangan prototipe halaman daftar buku



Gambar 4.32 Rancangan prototipe halaman daftar jurnal



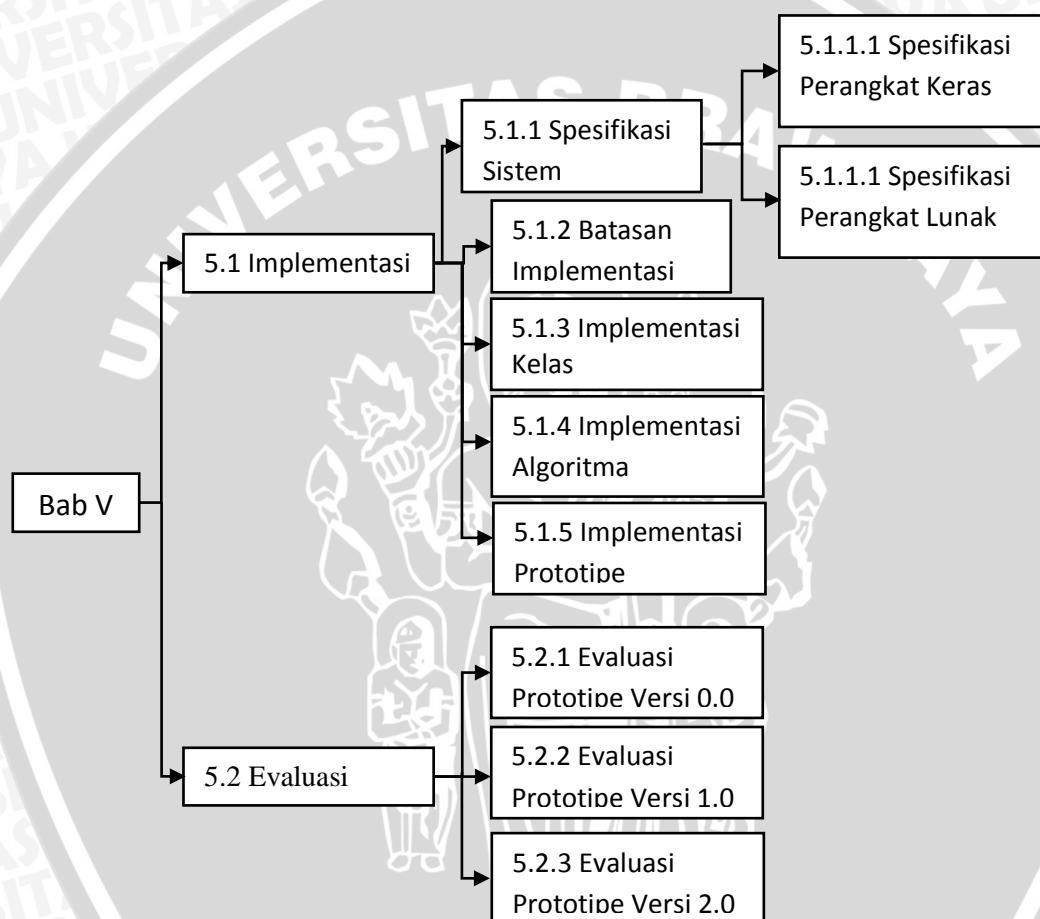
Gambar 4.33 Rancangan prototipe halaman validasi buku



Gambar 4.33 Rancangan prototipe halaman validasi jurnal

BAB 5 IMPLEMENTASI

Bab ini akan menjelaskan tentang tahapan implementasi dan evaluasi prototipe perangkat lunak berdasarkan hasil dari analisis kebutuhan. Tahapan yang dilakukan pada implementasi ini terdiri dari spesifikasi sistem, batasan implementasi, implementasi basis data, implementasi kelas, implementasi algoritma dan implementasi prototyping. Dilanjutkan dengan evaluasi prototyping berdasarkan iterasi yang dilakukan. Berikut ini diagram blok bab implementasi yang ditunjukkan oleh **Gambar 5.1**.



Gambar 5.1 Diagram Blok Bab Implementasi

5.1 Implementasi

5.1.1 Spesifikasi Sistem

Spesifikasi sistem merupakan lingkungan implementasi dalam pembuatan sistem aplikasi ini yang terdiri dari spesifikasi perangkat keras dan perangkat lunak.

5.1.1.1 Spesifikasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan dalam proses implementasi aplikasi web terintegrasi pada perpustakaan Universitas Brawijaya dapat dilihat pada **Tabel 5.1**.

Tabel 5.1 Spesifikasi Perangkat Keras

Nama Komponen	Spesifikasi
<i>System Model</i>	Lenovo Idepad Z400
<i>Processor</i>	Intel® Core™ i5-3230M
<i>Memory</i>	1TB
<i>Memory RAM</i>	4GB
<i>Display</i>	14" LED HD 1366 x 728 pixel 16:9 aspect ratio

5.1.1.2 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam proses implementasi aplikasi web terintegrasi pada perpustakaan Universitas Brawijaya dapat dilihat pada **Tabel 5.2**.

Tabel 5.2 Spesifikasi Perangkat Lunak

Nama Komponen	Spesifikasi
<i>Operating System</i>	Windows 8
Bahasa Pemrograman	HTML, PHP, CSS, JavaScript
<i>Tools</i>	XAMPP 1.8.8 SQLServer 2008 Navicat 11.1.8 Enterprise Architect 8.0 Brackets Balsamiq Mockups 3.1.7 Microsoft Word 2010

5.1.2 Batasan Implementasi

Berikut adalah batasan-batasan dalam implementasi aplikasi web terintegrasi pada perpustakaan Universitas Brawijaya diantaranya yaitu :

1. *Database Management System* yang digunakan adalah SQL Server 2008.
2. *Framework YII* yang digunakan adalah versi 1.1.
3. Sistem yang dibangun khusus menangani permintaan bahan pustaka dan pengolahan bahan pustaka.

5.1.3 Implementasi Kelas

Setiap diagram kelas yang telah dirancang pada tahap perancangan sistem direalisasikan pada sebuah file dengan ekstrasi .php. Penjelasan tentang *class controller* yang digunakan pada sistem akan dijelaskan pada **Tabel 5.3** sedangkan implementasi *class model* dijelaskan pada **Tabel 5.4**.

Tabel 5.3 Implementasi *Class Controller* pada kode program

Nama package	Nama class	Nama file
<i>Controller</i>	SiteController	SiteController.php
<i>Controller</i>	PermintaanController	PermintaanController.php
<i>Controller</i>	PengolahanController	PengolahanController.php
<i>Controller</i>	BukuController	BukuController.php
<i>Controller</i>	JurnalController	JurnalController.php
<i>Controller</i>	CetakController	CetakController.php
<i>Controller</i>	StafController	StafController.php
<i>Controller</i>	AnggotaController	MemberController.php

Tabel 5.4 Implementasi *Class Model* pada kode program

Nama package	Nama class	Nama file
<i>Model</i>	m_login	m_login.php
<i>Model</i>	m_privilege	m_privilege.php
<i>Model</i>	m_permintaan	m_permintaan.php
<i>Model</i>	m_permintaanBuku	m_permintaanBuku.php
<i>Model</i>	m_permintaanJurnal	m_permintaanJurnal.php
<i>Model</i>	m_status	m_status.php
<i>Model</i>	m_bkMain	m_bkMain.php
<i>Model</i>	m_bkCreator	m_bkCreator.php
<i>Model</i>	m_bkSubject	m_bkSubject.php
<i>Model</i>	m_bkType	m_bkType.php
<i>Model</i>	m_mediaType	m_mediaType.php
<i>Model</i>	m_funding	m_funding.php
<i>Model</i>	m_locationSpec	m_locationSpec.php
<i>Model</i>	m_statusPustaka	m_statusPustaka.php

Tabel 5.4 (Lanjutan)

Nama <i>package</i>	Nama <i>class</i>	Nama file
Model	m_bkAddCopy	m_bkAddCopy.php
Model	m_rrMain	m_rrMain.php
Model	m_rrCreator	m_rrCreator.php
Model	m_rrSubject	m_rrSubject.php
Model	m_rrType	m_rrType.php
Model	m_rrAddCopy	m_rrAddCopy.php
Model	m_staf	m_staf.php
Model	m_anggota	m_anggota.php

5.1.4 Implementasi Algoritma

Didalam penulisan skripsi hanya mencantumkan *method* proses pemanggilan *web service* sehingga tidak semua *method* dicantumkan.. Pengimplementasian algoritma ini akan direpresentasikan dalam bentuk *pseudocode*.

5.1.4.1 Algoritma Login

Proses login merupakan proses pertama ketika menjalankan program. Pada proses ini terdapat proses penerapan *web service* SIAKAD UB yang dipergunakan khusus untuk perpustakaan Universitas Brawijaya. Implementasi algoritma login dapat dilihat pada **Tabel 5.5**.

Tabel 5.5 Implementasi algoritma login

<p>Instansiasi object m_login</p> <p>Jika m_login telah diisi maka</p> <p style="padding-left: 20px;">Memasukkan nilai input kedalam model</p> <p>Jika model telah valid maka</p> <p style="padding-left: 20px;">Menjalankan web service mengecek username dan password serta mengambil data</p> <p>Jika username dikenali maka</p> <p style="padding-left: 20px;">Jika password cocok maka</p> <p style="padding-left: 40px;">Session untuk user akan diisi dan menjalankan method cekDataLogin()</p> <p>Jika data login kosong maka</p> <p style="padding-left: 20px;">Simpan data login</p> <p style="padding-left: 20px;">Menuju halaman ubah biodata</p> <p>Jika data login tidak kosong maka</p>
--

Tabel 5.5 (Lanjutan)

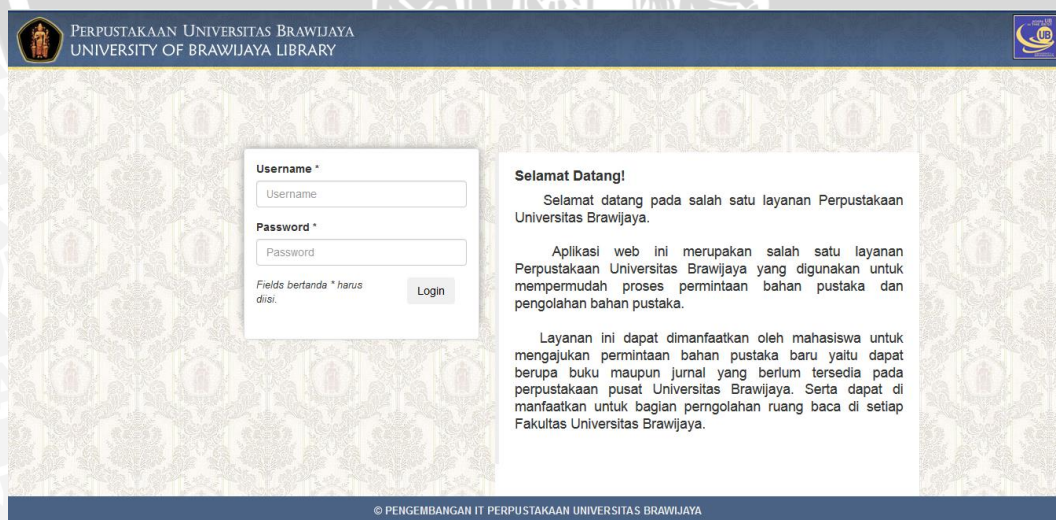
<p>Masuk halaman permintaan</p> <p>Jika password tidak cocok maka</p> <p>Mengisi variabel data dengan "Username atau password tidak sesuai!" serta menjalankan method logiDb()</p> <p>Jika username tidak dikenali maka</p> <p>Mengisi variabel data dengan "Username atau password tidak sesuai!" serta menjalankan method logiDb()</p> <p>Jika model tidak valid maka</p> <p>Menjalankan method logiDb()</p> <p>Jika m_login tidak terisi maka</p> <p>Kembali ke halaman login</p> <p>Selesai</p>

5.1.5 Implementasi Prototipe

Implementasi perangkat lunak yang dibangun berdasarkan perancangan menghasilkan 4 versi rilis berdasarkan hasil 3 kali evaluasi sistem oleh pihak pengguna. Hasil akhir rilis prototipe yang menjadi sitem aktual adalah versi 3.0 yang dapat dilihat pada bagian berikut ini.

1. Halaman Login

Halaman login merupakan halaman yang digunakan untuk masuk kedalam sistem. Semua aktor yang terlibat dapat melakukan login melalui halaman ini.



Gambar 5.2 Halaman Login

2. Halaman biodata anggota

Halaman biodata anggota yaitu halaman yang menampilkan detail biodata anggota dan ubah biodata anggota.

The screenshot shows the 'BIODATA' page for a member named FAISAL PRAMPUDINANTAKA. The page includes a header with the library logo and name, a navigation menu on the left, and a main content area with a table of personal details. A profile card on the right shows the member's name and a 'Logout' button.

NIM	115090600111037
Nama	FAISAL PRAMPUDINANTAKA
Jenjang	Sarjana S1
Program Studi	Ilmu Komputer
Fakultas	Program Teknologi Informasi dan Ilmu Komputer
Alamat Di Malang	Klayatan gang 1 no. 30
No Handphone (HP)	085755145486
Email	faisal.prampudinantaka@gmail.com

Gambar 5.3 Halaman biodata anggota

3. Halaman biodata staf

Halaman biodata anggota yaitu halaman yang menampilkan detail biodata staf dan edit biodata staf.

The screenshot shows the 'BIODATA STAF' page for a staff member named staf00. The page includes a header with the library logo and name, a navigation menu on the left, and a main content area with a table of staff details. A profile card on the right shows the staff member's name and a 'Logout' button.

Kode Staf	OP00
Username	staf00
Nama Lengkap	staf00
Lokasi	Perpustakaan Pusat
Unit	pengolahan

Gambar 5.4 Halaman biodata staf

4. Halaman Kelola Staf

Halaman kelola staf terdiri dari tambah staf, edit staf dan daftar staf. Halaman ini tidak dapat diakses oleh semua aktor, yang dapat mengakses halaman ini yaitu staf IT.

PERPUSTAKAAN UNIVERSITAS BRAWIJAYA
UNIVERSITY OF BRAWIJAYA LIBRARY

administrator

Dashboard

STAF

- Tambah Staf
- Daftar Staf

Tambah Staf

Fields bertanda * harus diisi:

Kode Staf: OP09

Username*: Username

Password*: Password

Nama Lengkap: Nama Lengkap

Bagian: Pengolahan

Submit

© PENGEMBANGAN IT PERPUSTAKAAN UNIVERSITAS BRAWIJAYA

Gambar 5.5 Halaman tambah staf

PERPUSTAKAAN UNIVERSITAS BRAWIJAYA
UNIVERSITY OF BRAWIJAYA LIBRARY

administrator

Dashboard

STAF

- Tambah Staf
- Daftar Staf

DAFTAR STAF

10 records per page

Kode Staf	Username	Nama Lengkap	Lokasi	Bagian	Aksi
OP00	staf00	staf00	Perpustakaan Pusat	pengolahan	✎
OP01	staf01	staf01	Ruang Baca PTIIK	ruang baca	✎
OP02	staf02	staf02	Ruang Baca Mesin	ruang baca	✎
OP03	staf03	staf03	Ruang Baca Matematika	ruang baca	✎
OP04	staf04	staf04	Ruang Baca Pertanian	ruang baca	✎
OP05	staf05	staf05	Ruang Baca THP	ruang baca	✎
OP06	staf06	staf06	Ruang Baca MIPA	ruang baca	✎
OP07	OP07	OP07	Ruang Baca Ilmu Administrasi Negara	ruang baca	✎
OP08	staf08	wisnu	Ruang Baca Ilmu Budaya	ruang baca	✎

© PENGEMBANGAN IT PERPUSTAKAAN UNIVERSITAS BRAWIJAYA

1

Gambar 5.6 Halaman daftar staf

5. Halaman permintaan bahan pustaka

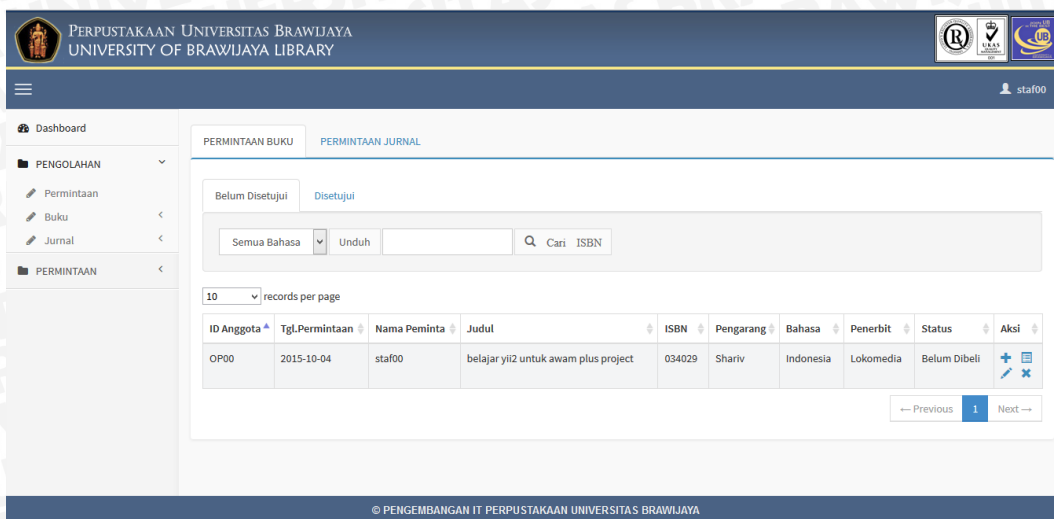
Halaman permintaan bahan pustaka terdiri dari permintaan bahan pustaka buku dan jurnal yang dapat dilakukan secara satuan maupun kolektif.

Gambar 5.7 Halaman Permintaan bahan pustaka

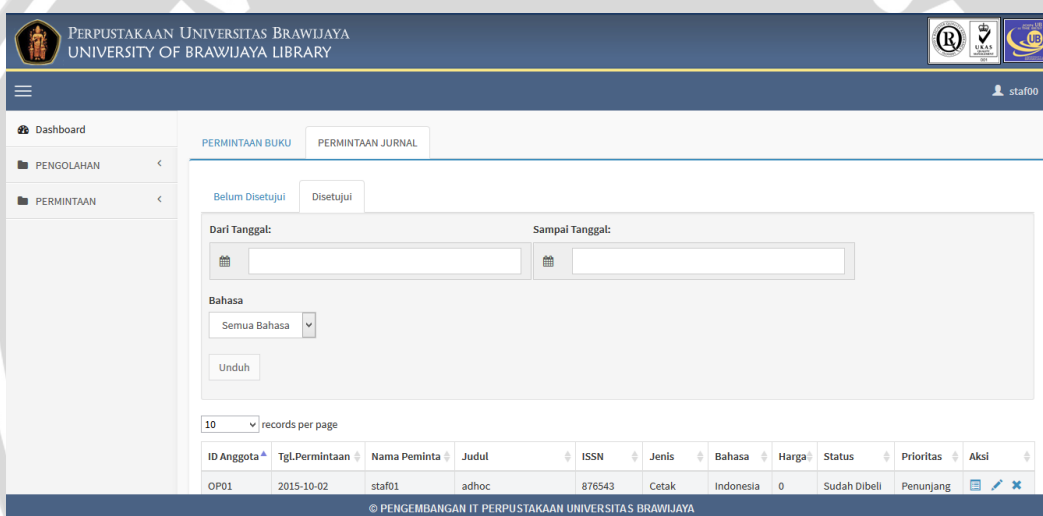
Gambar 5.8 Halaman permintaan bahan pustaka secara kolektif

6. Halaman pengolahan permintaan bahan pustaka

Pada halaman pengolahan permintaan bahan pustaka ini terdiri dari permintaan bahan pustaka buku dan jurnal.



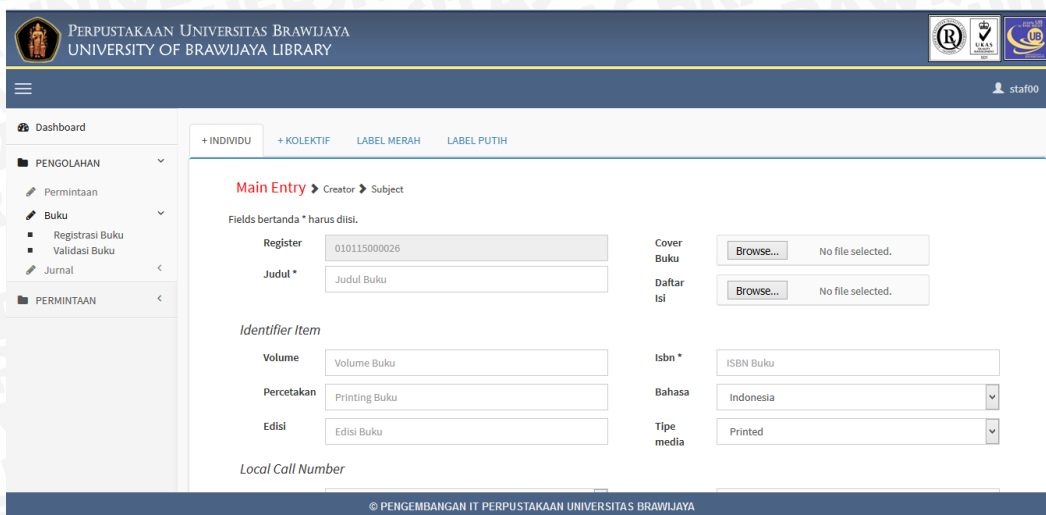
Gambar 5.9 Halaman pengolahan permintaan bahan pustaka buku



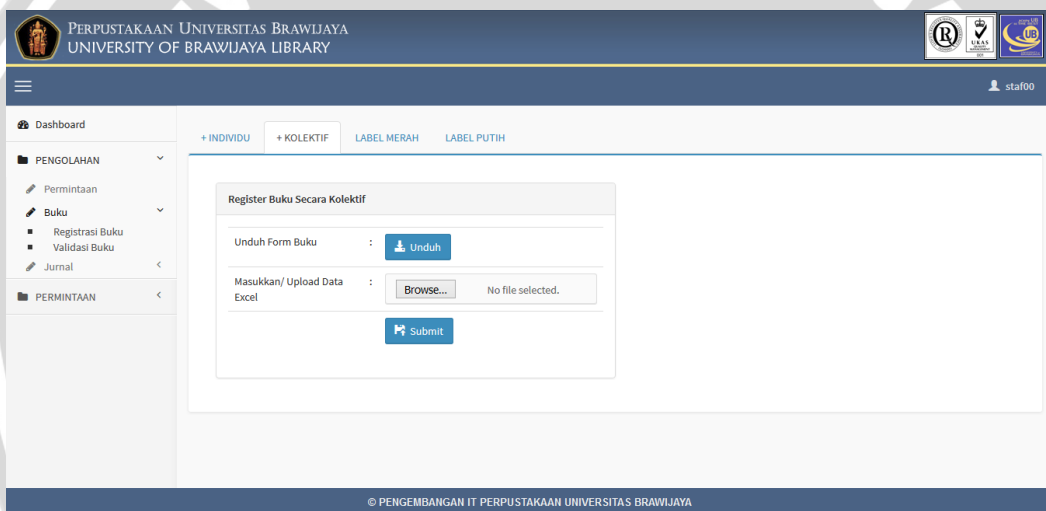
Gambar 5.10 Halaman pengolahan permintaan bahan pustaka jurnal

7. Halaman pengolahan bahan pustaka buku

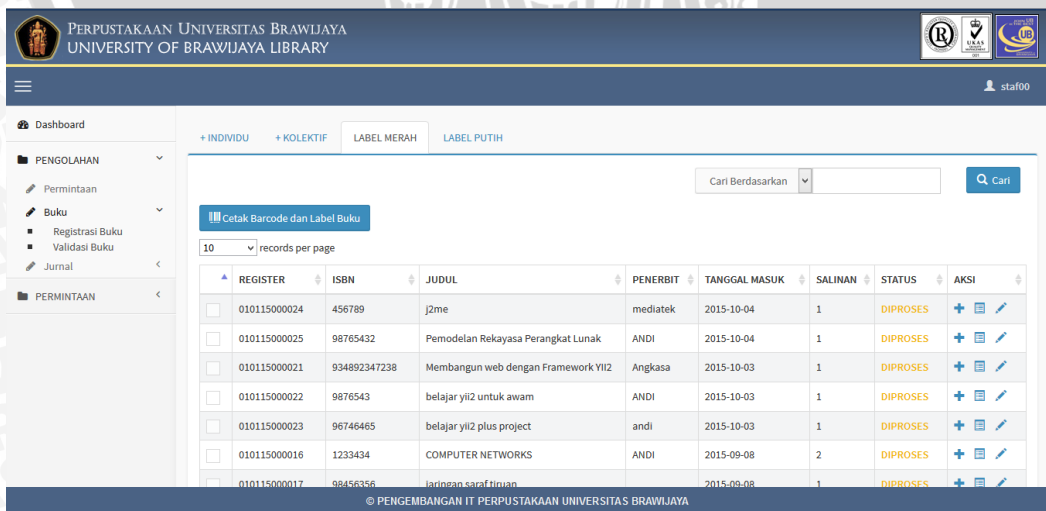
Pada halaman pengolahan bahan pustaka buku ini terdiri dari tambah buku baru, tambah buku secara kolektif, daftar buku label merah dan daftar buku label putih. Serta fungsi detail dan update data buku, cetak label dan validasi buku.



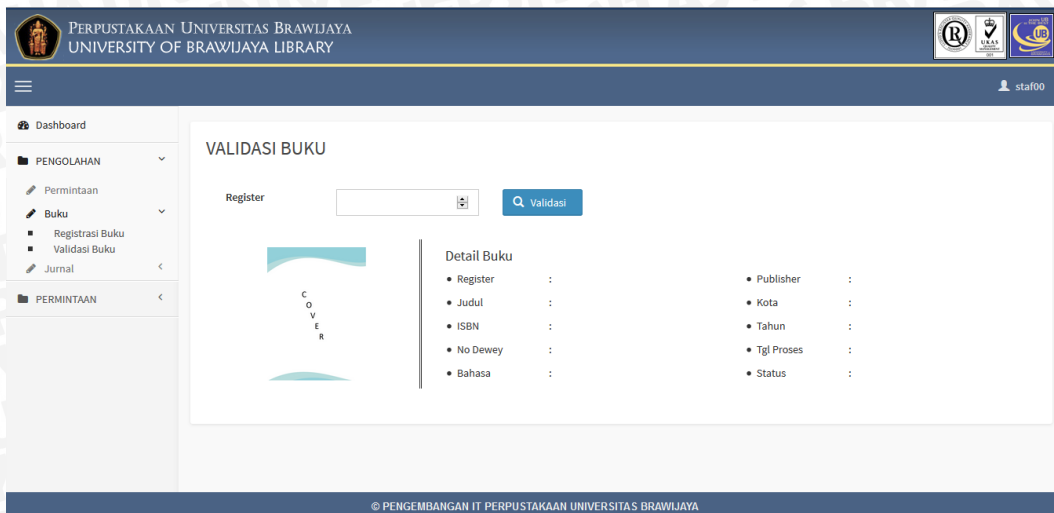
Gambar 5.11 Halaman tambah bahan pustaka buku baru



Gambar 5.12 Halaman pengolahan bahan pustaka buku secara kolektif



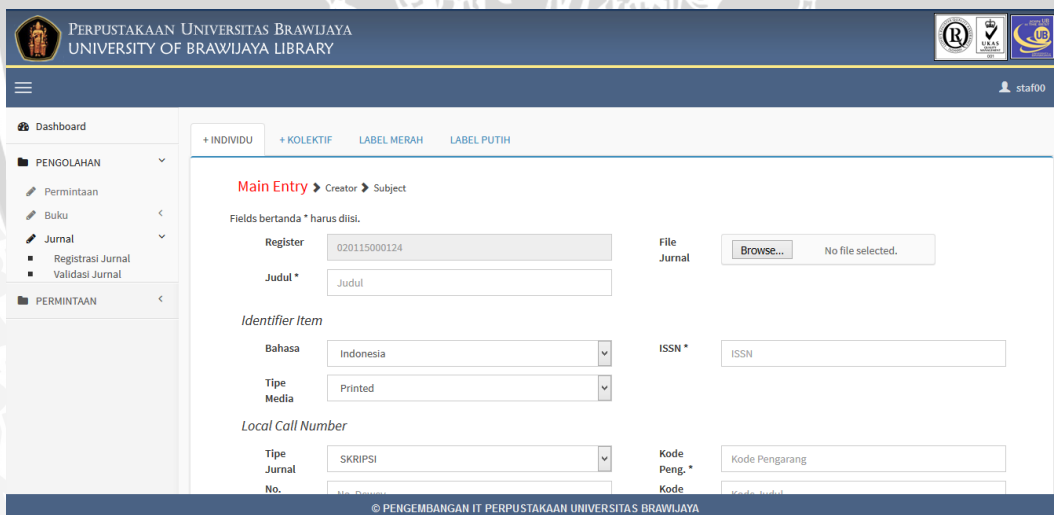
Gambar 5.13 Halaman daftar bahan pustaka buku label merah



Gambar 5.14 Halaman validasi bahan pustaka buku

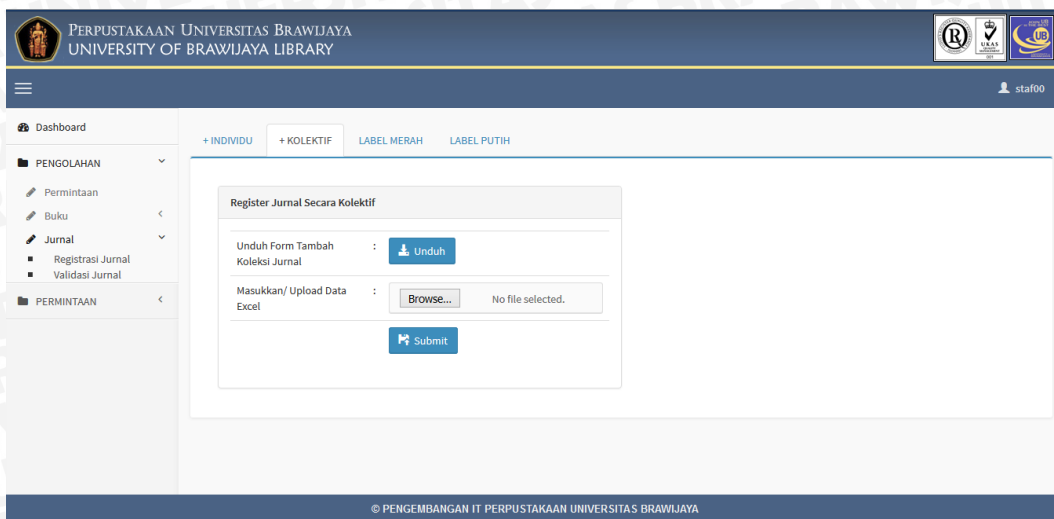
8. Halaman pengolahan bahan pustaka jurnal

Pada halaman pengolahan bahan pustaka buku ini terdiri dari tambah jurnal baru, tambah jurnal secara kolektif, daftar jurnal label merah dan daftar jurnal label putih. Serta fungsi detail dan update data jurnal, serta cetak label dan validasi jurnal.

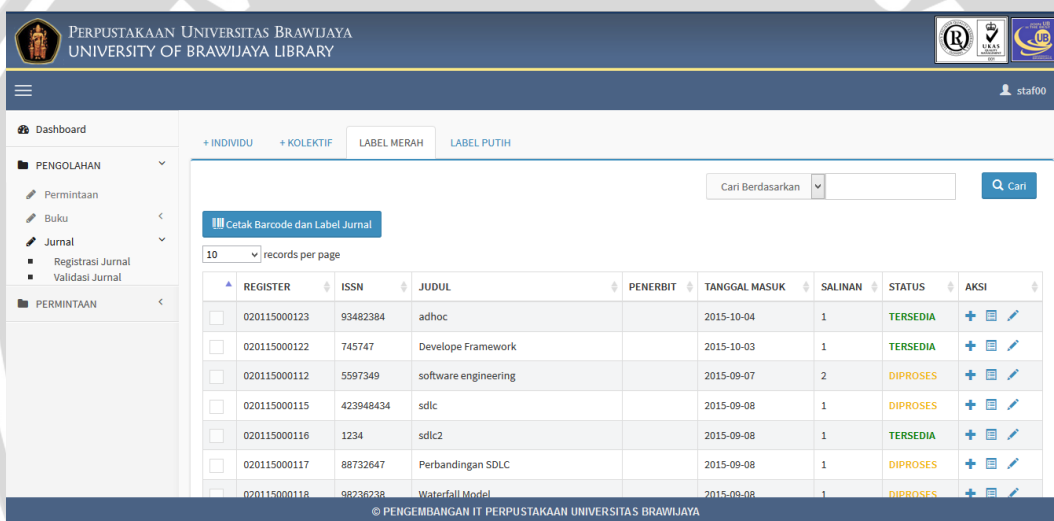


Gambar 5.15 Halaman pengolahan bahan pustaka jurnal baru

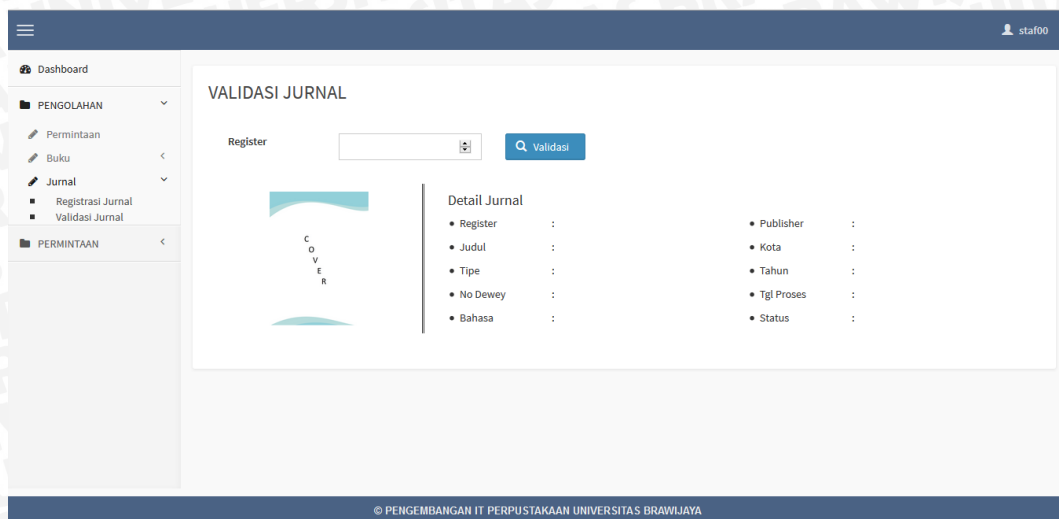




Gambar 5.16 Halaman pengolahan bahan pustaka jurnal secara kolektif



Gambar 5.17 Halaman daftar bahan pustaka jurnal label merah



Gambar 5.18 Halaman validasi bahan pustaka jurnal

5.2 Evaluasi

Pada tahap proses evaluasi *prototype* sistem ini dilakukan oleh pelanggan apakah *prototype* yang sudah dibangun sudah sesuai dengan keinginan pelanggan. Jika sudah sesuai maka implementasi sistem. Jika tidak, maka *prototype* direvisi dan memberi *feedback* dengan mengulang dari analisis kebutuhan pengguna. Dalam kasus ini dibatasi hanya dilakukan sebanyak 3 kali, sesuai dengan batasan yang telah ditetapkan.

5.2.1 Evaluasi Prototipe Versi 0.0

Prototipe versi 0.0 merupakan hasil dari analisis kebutuhan awal yang ditransformasikan dalam bentuk prototipe. Hasil dari evaluasi tahap awal adalah sebagai berikut.

1. Perubahan aktor *administrator* menjadi staf IT
2. Perubahan aktor staf menjadi staf ruang baca dan staf pengolahan serta hak aksesnya.
3. Perangkat lunak dapat melakukan permintaan bahan pustaka secara kolektif.
4. Perangkat lunak dapat menambah bahan pustaka secara kolektif
5. Perangkat lunak dapat menyediakan fasilitas unduh daftar permintaan bahan pustaka.

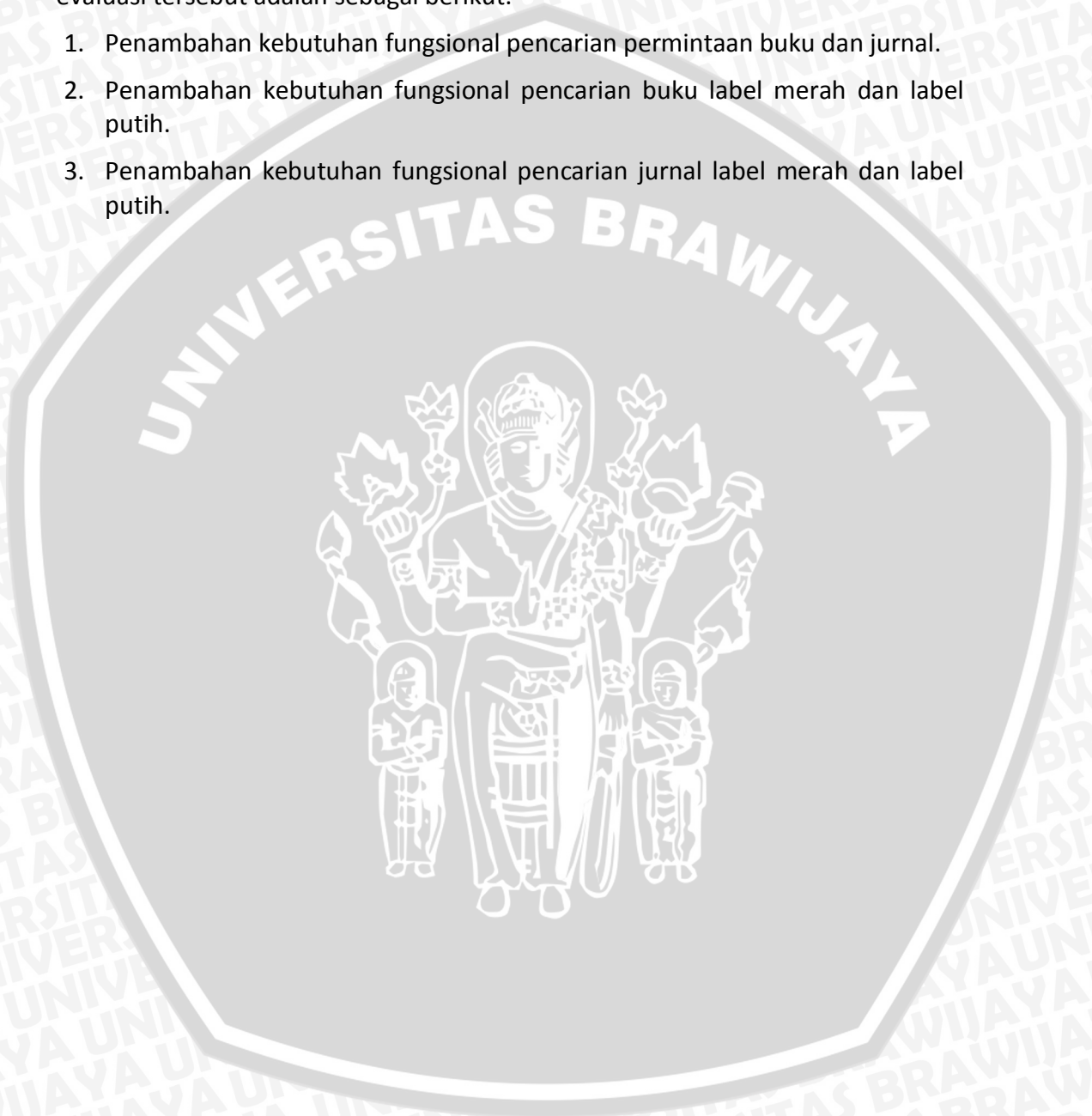
5.2.2 Evaluasi Prototipe Versi 1.0

Prototipe versi 1.0 dilakukan setelah dilakukan evaluasi dari protitipe versi 0.0 atau iterasi 2. Hasil dari evaluasi yaitu perubahan hak akses staf IT, awalnya staf IT dapat mengakses keseluruhan fungsional. Setelah dilakukan evaluasi staf IT hanya dapat mengakses pengolahan staf.

5.2.3 Evaluasi Prototipe Versi 2.0

Prototipe versi 2.0 dilakukan setelah dilakukan evaluasi dari protitipe versi 1.0 atau iterasi 3. Hasil dari evaluasi versi 2.0 akan menghasilkan versi rilis yaitu prortotipe versi 3.0 yang merupakan perangkat lunak yang sudah jadi. Hasil dari evaluasi tersebut adalah sebagai berikut.

1. Penambahan kebutuhan fungsional pencarian permintaan buku dan jurnal.
2. Penambahan kebutuhan fungsional pencarian buku label merah dan label putih.
3. Penambahan kebutuhan fungsional pencarian jurnal label merah dan label putih.



BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini akan dilakukan proses pengujian dan analisis hasil pengujian. Tujuan tahapan ini adalah untuk menemukan kesalahan-kesalahan yang bisa saja terjadi dalam sistem. Tahapan ini juga digunakan untuk memastikan bahwa sistem yang dibangun telah sesuai dengan apa yang direncanakan. Pengujian sistem dilakukan dengan menggunakan dua metode yaitu metode *white-box* dan metode *black-box*. Metode *white-box* digunakan untuk melakukan pengujian unit. Metode *black box* digunakan untuk melakukan pengujian integrasi, pengujian validasi dan untuk menguji *web service*. Pengujian performansi dilakukan dengan menghitung waktu proses yang dibutuhkan untuk melakukan permintaan data ke *web service*.

6.1 Pengujian Unit

Proses pengujian basis path merupakan proses pengujian unit yang dilakukan dengan menggunakan *pseudocode* yang menerapkan metode atau logika yang digunakan oleh sistem. *Pseudocode* tersebut akan dimodelkan ke dalam suatu *flow graph*. Proses ini dilakukan untuk menentukan jumlah *cyclomatic complexity* (kompleksitas siklomatis) dan menentukan jalur independen. Jumlah kompleksitas siklomatis diperoleh melalui tiga persamaan berikut:

$$V(G) = E - N + 2$$

$$V(G) = P + 1$$

$$V(G) = R$$

Keterangan,

$V(G)$: Jumlah kompleksitas siklomatis.

E : Sisi atau edge (garis penghubung antar node).

N : Jumlah simpul (node).

P : *Predicate node* pada grafik alir.

R : Jumlah region pada *flow graph*.

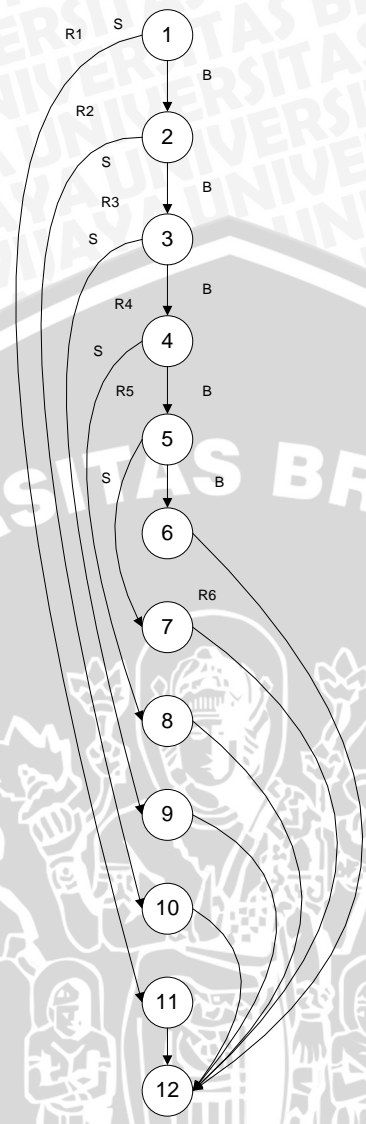
6.1.1 Pengujian Unit Proses Login

Berikut ini merupakan algoritma untuk melakukan proses *login* yang dijelaskan pada **Tabel 6.1**.

Tabel 6.1 Pengujian unit *login*

<p>Nama Algoritma: <i>login</i></p> <p>Deskripsi:</p> <ul style="list-style-type: none"> • Masukan: <i>Username</i> dan <i>password</i> • Proses 	
<ol style="list-style-type: none"> 1. 2. 3. 4. 5. 6. 7. 8. 9. 10. 11. 12. 	<p>Instansiasi object <code>m_login</code></p> <p>Jika <code>m_login</code> telah diisi maka Memasukkan nilai input kedalam model</p> <p>Jika model telah valid maka Menjalankan web service mengecek username dan password serta mengambil data</p> <p>Jika username dikenali maka</p> <p>Jika password cocok maka Session untuk user akan diisi dan menjalankan method <code>cekDataLogin()</code></p> <p>Jika data login kosong maka Simpan data login</p> <p>Menuju Halaman Ubah Biodata</p> <p>Jika data login tidak kosong maka Masuk halaman permintaan</p> <p>Jika password tidak cocok maka Mengisi variabel data dengan "Username atau password tidak sesuai!" serta menjalankan method <code>logiDb()</code></p> <p>Jika username tidak dikenali maka Mengisi variabel data dengan " Username atau password tidak sesuai!" serta menjalankan method <code>logiDb()</code></p> <p>Jika model tidak valid maka Menjalankan method <code>logiDb()</code></p> <p>Jika <code>m_login</code> tidak terisi maka Kembali ke halaman login</p> <p>Selesai</p>

Berdasarkan algoritma yang telah diperoleh seperti pada **Tabel 6.1** maka diperoleh *flow graph* seperti pada **Gambar 6.1**.



Gambar 6.2 Flow Graph Login

Berdasarkan *flow graph* yang diperoleh seperti pada **Gambar 6.1** maka diperoleh jumlah kompleksitas siklomatis (*cyclomatic complexity*) seperti berikut:

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 16 - 12 + 2 \\
 &= 6
 \end{aligned}$$

$$\begin{aligned}
 V(G) &= P + 1 \\
 &= 5 + 1 \\
 &= 6
 \end{aligned}$$

$$\begin{aligned}
 V(G) &= R \\
 &= 6
 \end{aligned}$$



Berdasarkan jumlah kompleksitas siklomatis yang telah didapatkan, maka akan ditentukan enam jalur independen, yaitu:

Jalur 1: 1 – 11 – 12

Jalur 2: 1 – 2 – 10 – 12

Jalur 3: 1 – 2 – 3 – 9 – 12

Jalur 4: 1 – 2 – 3 – 4 – 8 – 12

Jalur 5: 1 – 2 – 3 – 4 – 5 – 7 – 12

Jalur 6: 1 – 2 – 3 – 4 – 5 – 6 – 12

Berdasarkan jalur independen yang ditentukan, maka dapat diperoleh kasus uji login dijelaskan pada **Tabel 6.2**.

Tabel 6.2 Kasus Uji Login

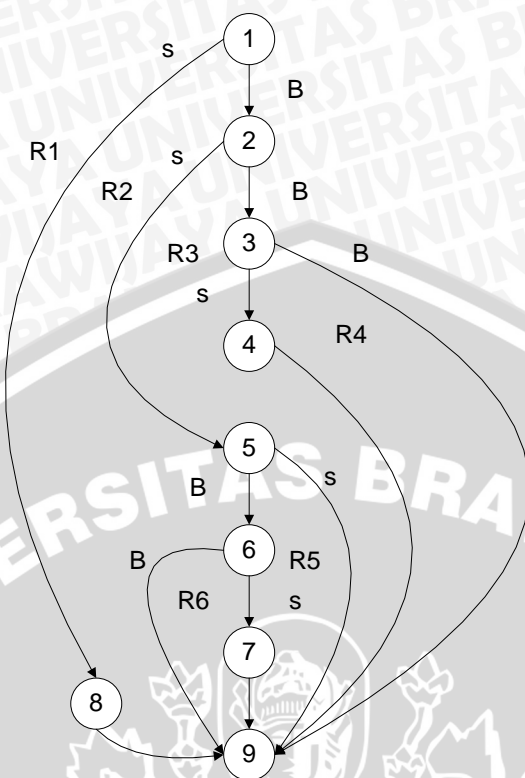
Jalur	Data Input	Hasil yang diharapkan	Hasil yang diperoleh	Status
1	Tidak ada data yang dimasukkan	Menampilkan halaman login	Menampilkan halaman login	valid
2	Hanya mengisi salah satu antara <i>username</i> dan <i>password</i>	Menampilkan halaman login dan pesan kesalahan	Menampilkan halaman login dan pesan kesalahan	valid
3	<i>Username</i> dan <i>password</i> tidak terdaftar	Menampilkan halaman login dan pesan kesalahan	Menampilkan halaman login dan pesan kesalahan	valid
4	<i>Username</i> benar dan <i>password</i> salah	Menampilkan halaman login dan pesan kesalahan	Menampilkan halaman login dan pesan kesalahan	valid
5	<i>Username</i> dan <i>password</i> benar	Menampilkan halaman permintaan	Menampilkan halaman permintaan	valid
6	<i>Username</i> dan <i>password</i> benar	Menampilkan halaman ubah biodata	Menampilkan halaman ubah biodata	valid

6.1.2 Pengujian Unit Proses Tambah Permintaan Bahan pustaka

Berikut ini merupakan algoritma untuk melakukan proses *login* yang dijelaskan pada **Tabel 6.3**.

Tabel 6.3 Pengujian unit proses tambah permintaan bahan putaka

Nama Algoritma: permintaan individu	
Deskripsi:	
<ul style="list-style-type: none"> • Masukan: data permintaan bahan pustaka • Proses 	
	<p>Instansiasi objek m_permintaan</p> <p>Instansiasi objek m_permintaanBuku</p> <p>Instansiasi objek m_permintaanJurnal</p> <p>Inisialisasi variable \$user dengan session id_anggota</p>
1.	<p>Jika data m_permintaan telah diisi maka</p> <p> Memasukkan nilai input kedalam model</p> <p> Mengisi variable \$k_jenis</p>
2.	<p>Jika variable \$k_jenis sama dengan 1 maka</p> <p> Memasukkan nilai input kedalam modelBk</p>
3.	<p>Jika modelBk valid maka</p> <p> Simpan data \$model</p> <p> Mengambil id_terakhir</p> <p> Menjalankan method inputBukuIndividu</p> <p> Menuju halaman permintaan</p>
4.	<p>Jika modelBk tidak valid maka</p> <p> Mengambil data buku</p> <p> Mengambil data Jurnal</p> <p> Render halaman permintaan</p>
5.	<p>Jika variable \$k_jenis sama dengan 2 maka</p> <p> Memasukkan nilai input kedalam modelJur</p>
6.	<p>Jika modelJur valid maka</p> <p> Simpan data \$model</p> <p> Mengambil id_terakhir</p> <p> Menjalankan method inputJurnalIndividu</p> <p> Menuju halaman permintaan</p>
7.	<p>Jika modelJur tidak valid maka</p> <p> Mengambil data buku</p> <p> Mengambil data Jurnal</p> <p> Render halaman permintaan</p>
8.	<p>Jika m_permintaan tidak terisi</p> <p> Mengisi variable \$dataBuku dengan hasil method getBuku</p> <p> Mengisi variable \$dataJur dengan hasil method getJurnal</p> <p> Render halaman permintaan</p>
9.	Selesai



Gambar 6.2 Flow graph tambah permintaan bahan pustaka

Berdasarkan *flow graph* yang diperoleh seperti pada **Gambar 6.2** maka diperoleh jumlah kompleksitas siklomatis (*cyclomatic complexity*) seperti berikut:

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 13 - 9 + 2 \\
 &= 6
 \end{aligned}$$

$$\begin{aligned}
 V(G) &= P + 1 \\
 &= 5 + 1 \\
 &= 6
 \end{aligned}$$

$$V(G) = R = 6$$

Berdasarkan jumlah kompleksitas siklomatis yang telah didapatkan, maka akan ditentukan enam jalur independen, yaitu:

Jalur 1: 1 – 8 – 9

Jalur 2: 1 – 2 – 5 – 9

Jalur 3: 1 – 2 – 5 – 6 – 9

Jalur 4: 1 – 2 – 5 – 6 – 7 – 9

Jalur 5: 1 – 2 – 3 – 9

Jalur 6: 1 – 2 – 3 – 4 – 9

Berdasarkan jalur independen yang ditentukan, maka dapat diperoleh kasus uji tambah permintaan bahan pustaka dijelaskan pada **Tabel 6.4**.

Tabel 6.4 Kasus uji tambah permintaan bahan putaka

Jalur	Data Input	Hasil yang diharapkan	Hasil yang diperoleh	Status
1	Tidak ada data yang dimasukkan	Menampilkan halaman form permintaan	Menampilkan halaman form permintaan	valid
2	Data permintaan bahan pustaka	Menampilkan halaman form permintaan	Menampilkan halaman form permintaan	valid
3	Data permintaan bahan pustaka	Menampilkan halaman daftar permintaan jurnal	Menampilkan halaman daftar permintaan jurnal	valid
4	Data permintaan bahan pustaka	Menampilkan halaman form permintaan dan pesan kesalahan	Menampilkan halaman form permintaan dan pesan kesalahan	valid
5	Data permintaan bahan pustaka	Menampilkan halaman daftar permintaan buku	Menampilkan halaman daftar permintaan buku	valid
6	Data permintaan bahan pustaka	Menampilkan halaman form permintaan dan pesan kesalahan	Menampilkan halaman form permintaan dan pesan kesalahan	valid

6.1.3 Pengujian Unit Proses Cari Buku Label Merah

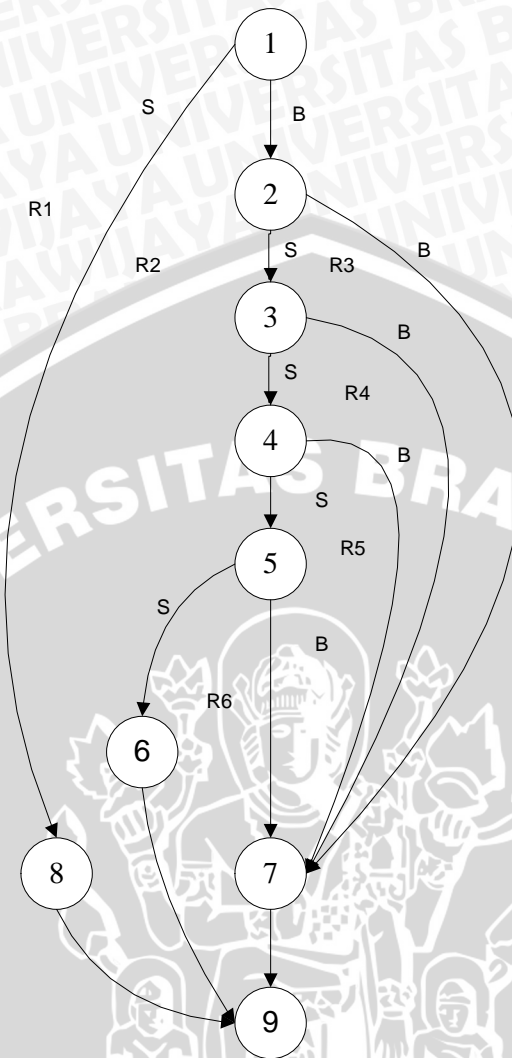
Berikut ini merupakan algoritma untuk melakukan proses cari buku label merah yang dijelaskan pada **Tabel 6.5**.

Tabel 6.5 Pengujian unit proses cari buku label merah

Nama Algoritma: Cari buku label merah	
Deskripsi:	
<ul style="list-style-type: none"> Masukan: data yang dicari Proses 	
	Instansiasi objek m_bkMain Instansiasi objek m_locationSpec Instansiasi objek m_funding Instansiasi objek m_bkType

Tabel 6.5 (Lanjutan)

	<p>Instansiasi objek m_mediaType</p> <p>Instansiasi objek m_bkCreator</p> <p>Instansiasi objek m_bkSubject</p> <p>Instansiasi objek m_bkAddCopy</p>
1.	<p>Jika \$_POST['searchMerah'] bernilai maka</p> <p style="padding-left: 20px;">Variable \$kriteria sama dengan \$_POST['searchMerah']</p> <p style="padding-left: 20px;">Inisialisasi \$location dengan session bagian</p> <p style="padding-left: 20px;">Inisialisasi \$cari dengan "%".\$_POST['cariMerah']."%</p>
2.	<p>Jika variable \$kriteria sama dengan judul maka</p> <p style="padding-left: 20px;">Menjalankan fungsi \$model->cariJudul(\$cari,\$location)</p>
3.	<p>Jika variable \$kriteria sama dengan register maka</p> <p style="padding-left: 20px;">Menjalankan fungsi \$model->cariRegister(\$cari,\$location)</p>
4.	<p>Jika variable \$kriteria sama dengan isbn maka</p> <p style="padding-left: 20px;">Menjalankan fungsi \$model->cariIsbn(\$cari,\$location)</p>
5.	<p>Jika variable \$kriteria sama dengan penerbit maka</p> <p style="padding-left: 20px;">Menjalankan fungsi \$model->cariPenerbit(\$cari,\$location)</p>
6.	<p>Jika variabel \$kriteria tidak terisi maka</p> <p style="padding-left: 20px;">Inisialisasi session tab dengan "merah"</p> <p style="padding-left: 20px;">Inisialisasi pesan error</p> <p style="padding-left: 20px;">Menuju halaman registrasi buku</p>
7.	<p>Inisialisasi \$data dengan fungsi \$modCopy->bkCopyAll(\$location)</p> <p>Inisialisasi \$newReg dengan fungsi \$this->generateRegister()</p> <p>Inisialisasi session tab dengan "merah"</p> <p>Render buku/regBuku</p>
8.	<p>Jika \$_POST['searchMerah'] tidak memiliki nilai maka</p> <p style="padding-left: 20px;">Inisialisasi session tab dengan "merah"</p> <p style="padding-left: 20px;">Inisialisasi pesan error</p> <p style="padding-left: 20px;">Menuju halaman registrasi buku</p>
9.	<p>Selesai</p>



Gambar 6.3 Flow graph cari buku label merah

Berdasarkan *flow graph* yang diperoleh seperti pada **Gambar 6.3** maka diperoleh jumlah kompleksitas siklomatis (*cyclomatic complexity*) seperti berikut:

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 13 - 9 + 2 \\ &= 6 \end{aligned}$$

$$\begin{aligned} V(G) &= P + 1 \\ &= 5 + 1 \\ &= 6 \end{aligned}$$

$$\begin{aligned} V(G) &= R \\ &= 6 \end{aligned}$$

Berdasarkan jumlah kompleksitas siklomatis yang telah didapatkan, maka akan ditentukan enam jalur independen, yaitu:

Jalur 1: 1 – 8 – 9

Jalur 2: 1 – 2 – 7 – 9

Jalur 3: 1 – 2 – 3 – 7 – 9

Jalur 4: 1 – 2 – 3 – 4 – 7 – 9

Jalur 5: 1 – 2 – 3 – 4 – 5 – 7 – 9

Jalur 6: 1 – 2 – 3 – 4 – 5 – 6 – 9

Berdasarkan jalur independen yang ditentukan, maka dapat diperoleh kasus uji cari buku label merah dijelaskan pada **Tabel 6.6**.

Tabel 6.6 Kasus uji cari buku label merah

Jalur	Data Input	Hasil yang diharapkan	Hasil yang diperoleh	Status
1	Tidak ada data yang dimasukkan	Menampilkan halaman registrasi buku tab daftar buku label merah	Menampilkan halaman registrasi buku tab daftar buku label merah	valid
2	Data pencarian	Menampilkan daftar buku label merah yang dicari berdasarkan judul	Menampilkan daftar buku label merah yang dicari berdasarkan judul	valid
3	Data pencarian	Menampilkan daftar buku label merah yang dicari berdasarkan register	Menampilkan daftar buku label merah yang dicari berdasarkan register	valid
4	Data pencarian	Menampilkan daftar buku label merah yang dicari berdasarkan isbn	Menampilkan daftar buku label merah yang dicari berdasarkan isbn	valid
5	Data pencarian	Menampilkan halaman registrasi buku tab daftar buku label merah dan menampilkan pesan kesalahan	Menampilkan halaman registrasi buku tab daftar buku label merah dan menampilkan pesan kesalahan	valid
6	Tidak memilih jenis pencarian	Menampilkan halaman form permintaan dan pesan kesalahan	Menampilkan halaman form permintaan dan pesan kesalahan	valid

6.2 Pengujian Integrasi

Pengujian integrasi dilakukan pada proses yang melibatkan interaksi antar kelas. Di mana teknik yang digunakan pada pengujian ini menggunakan *black box*. Sistem memenuhi aspek pengujian integrasi apabila seluruh hasil yang diharapkan (*expected result*) sesuai dengan hasil yang diperoleh (*actual result*). Namun jika terdapat hasil yang tidak sesuai yang diharapkan maka proses integrasi antar kelas harus diperbaiki.

6.2.1 Kasus Uji Pengujian Integrasi

Berikut merupakan kasus uji yang digunakan untuk melakukan pengujian integrasi.

Tabel 6.2 Kasus Uji Integrasi *actionLogin* dan *cekDataLogin*

Nomor Kasus Uji	INT_01
Nama Kasus Uji	Kasus Uji Integrasi unit <i>actionLogin</i> dan <i>cekDataLogin</i>
Unit yang Terlibat	Unit <i>actionLogin</i> dan <i>cekDataLogin</i>
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa proses yang melibatkan interaksi antara unit <i>actionLogin</i> dan <i>cekDataLogin</i> telah terintegrasi dengan benar
Prosedur Uji	<ol style="list-style-type: none"> 1. Penguji menyediakan kelas Tester untuk mengaktifkan unit <i>actionLogin</i> 2. Penguji menyediakan masukan yang diperlukan unit <i>actionLogin</i> dan <i>cekDataLogin</i> 3. Penguji mengeksekusi kelas Tester 4. Penguji mendokumentasikan hasil eksekusi bernilai <i>true</i> atau <i>false</i>
Hasil yang Diharapkan	Nilai kembalian bernilai <i>true</i> jika <i>cekDataLogin</i> berhasil mengembalikan nilai variabel dan nilai kembalian bernilai <i>false</i> jika data <i>cekDataLogin</i> tidak berhasil mengembalikan nilai variabel
Hasil Aktual	Nilai kembalian bernilai <i>true</i> jika <i>cekDataLogin</i> berhasil mengembalikan nilai variabel dan nilai kembalian bernilai <i>false</i> jika data <i>cekDataLogin</i> tidak berhasil mengembalikan nilai variabel

Tabel 6.8 Kasus Uji Integrasi *actionPermintaanIndividu* dan *inputBukuIndividu*

Nomor Kasus Uji	INT_02
Nama Kasus Uji	Kasus Uji Integrasi unit <i>actionPermintaanIndividu</i> dan <i>inputBukuIndividu</i>
Unit yang Terlibat	Unit <i>actionPermintaanIndividu</i> dan <i>inputBukuIndividu</i>

Tabel 6.8 (Lanjutan)

Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa proses yang melibatkan interaksi antara unit <i>actionPermintaanIndividu</i> dan <i>inputBukuIndividu</i> telah terintegrasi dengan benar
Prosedur Uji	<ol style="list-style-type: none"> 1. Penguji menyediakan kelas Tester untuk mengaktifkan unit <i>actionLogin</i> 2. Penguji menyediakan masukan yang diperlukan unit <i>actionPermintaanIndividu</i> dan <i>inputBukuIndividu</i> 3. Penguji mengeksekusi kelas Tester 4. Penguji mendokumentasikan hasil eksekusi bernilai <i>true</i> atau <i>false</i>
Hasil yang Diharapkan	Nilai kembalian bernilai <i>true</i> jika unit <i>inputBukuIndividu</i> berhasil memasukkan data buku dan nilai kembalian bernilai <i>false</i> jika unit <i>inputBukuIndividu</i> tidak berhasil memasukkan data buku
Hasil Aktual	Nilai kembalian bernilai <i>true</i> jika unit <i>inputBukuIndividu</i> berhasil memasukkan data buku dan nilai kembalian bernilai <i>false</i> jika unit <i>inputBukuIndividu</i> tidak berhasil memasukkan data buku
Status Validasi	Valid

Tabel 6.9 Kasus Uji Integrasi *actionPermintaanIndividu* dan *inputJurnalIndividu*

Nomor Kasus Uji	INT_03
Nama Kasus Uji	Kasus Uji Integrasi unit <i>actionPermintaanIndividu</i> dan <i>inputJurnalIndividu</i>
Unit yang Terlibat	Unit <i>actionPermintaanIndividu</i> dan <i>inputJurnalIndividu</i>
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa proses yang melibatkan interaksi antara unit <i>actionPermintaanIndividu</i> dan <i>inputJurnalIndividu</i> telah terintegrasi dengan benar

Tabel 6.9 (Lanjutan)

Prosedur Uji	<ol style="list-style-type: none"> 1. Penguji menyediakan kelas Tester untuk mengaktifkan unit <i>actionLogin</i> 2. Penguji menyediakan masukan yang diperlukan unit <i>actionPermintaanIndividu</i> dan <i>inputJurnalIndividu</i> 3. Penguji mengeksekusi kelas Tester 4. Penguji mendokumentasikan hasil eksekusi bernilai <i>true</i> atau <i>false</i>
Hasil yang Diharapkan	Nilai kembalian bernilai <i>true</i> jika unit <i>inputJurnalIndividu</i> berhasil memasukkan data jurnal dan nilai kembalian bernilai <i>false</i> jika unit <i>inputJurnalIndividu</i> tidak berhasil memasukkan data jurnal
Hasil Aktual	Nilai kembalian bernilai <i>true</i> jika unit <i>inputJurnalIndividu</i> berhasil memasukkan data jurnal dan nilai kembalian bernilai <i>false</i> jika unit <i>inputJurnalIndividu</i> tidak berhasil memasukkan data jurnal
Status Validasi	Valid

Tabel 6.10 Kasus Uji Integrasi *actionSearchMerah* dan *cariJudul*

Nomor Kasus Uji	INT_04
Nama Kasus Uji	Kasus Uji Integrasi unit <i>actionSearchMerah</i> dan <i>cariJudul</i>
Unit yang Terlibat	Unit <i>actionSearchMerah</i> dan <i>cariJudul</i>
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa proses yang melibatkan interaksi antara unit <i>actionSearchMerah</i> dan <i>cariJudul</i> telah terintegrasi dengan benar
Prosedur Uji	<ol style="list-style-type: none"> 1. Penguji menyediakan kelas Tester untuk mengaktifkan unit <i>actionSearchMerah</i> 2. Penguji menyediakan masukan yang diperlukan unit <i>actionSearchMerah</i> dan <i>cariJudul</i> 3. Penguji mengeksekusi kelas Tester 4. Penguji mendokumentasikan hasil eksekusi bernilai <i>true</i> atau <i>false</i>

Tabel 6.10 (Lanjutan)

Hasil yang Diharapkan	Nilai kembalian bernilai <i>true</i> jika unit <i>cariJudul</i> berhasil mengembalikan nilai variabel yang diinginkan dan nilai kembalian bernilai <i>false</i> jika unit <i>cariJudul</i> tidak berhasil mengembalikan nilai variabel yang diinginkan
Hasil Aktual	Nilai kembalian bernilai <i>true</i> jika unit <i>cariJudul</i> berhasil mengembalikan nilai variabel yang diinginkan dan nilai kembalian bernilai <i>false</i> jika unit <i>cariJudul</i> tidak berhasil mengembalikan nilai variabel yang diinginkan
Status Validasi	Valid

6.3 Pengujian Validasi

Pengujian validasi merupakan proses menguji seluruh skenario use case yang telah dibuat pada bab sebelumnya.

6.3.1 Kasus Uji Pengujian Validasi

Berikut ini merupakan kasus uji untuk melakukan pengujian validasi pada web aplikasi perpustakaan terintegrasi pada Universitas Brawijaya. Berikut ini kasus uji yang dibuat berdasarkan use case yang telah dibuat pada bab sebelumnya.

Tabel 6.11 Kasus uji Login

Nama Kasus Uji	Login
Objek Uji	SKPL-01-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan login ke sistem
Prosedur Uji	<ol style="list-style-type: none"> 1. Membuka sistem 2. Memasukkan <i>username</i> dan <i>password</i> 3. Menekan tombol <i>login</i>
Hasil yang Diharapkan	Sistem dapat mengecek kebenaran input data aktor sehingga dapat masuk ke dalam sistem
Hasil Aktual	Seletelah dilakukan pengujian sistem dapat mengecek kebenaran input data sistem sehingga dapat masuk ke dalam system
Status Validasi	Valid

Tabel 6.12 Kasus uji login alternatif 1 : Jika *username* atau *password* kosong

Nama Kasus Uji	Login
Objek Uji	SKPL-01-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan <i>error</i> ketika salah satu atau kedua field kosong
Prosedur Uji	<ol style="list-style-type: none"> 1. Membuka sistem 2. Mengosongkan <i>field username</i> dan <i>password</i> 3. Menekan tombol <i>login</i>
Hasil yang Diharapkan	Sistem dapat menampilkan pesan <i>username cannot blank</i> Tabel 6.8 (Lanjutan) : <i>blank</i>
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan <i>username cannot blank</i> dan <i>password cannot blank</i>
Status Validasi	Valid

Tabel 6.13 Kasus uji login alternatif 2 : Jika kombinasi *username* atau *password* salah

Nama Kasus Uji	Login
Objek Uji	SKPL-01-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan <i>error</i> ketika kombinasi <i>username</i> dan <i>password</i> salah
Prosedur Uji	<ol style="list-style-type: none"> 1. Membuka sistem 2. Memasukkan <i>username</i> dan <i>password</i> dengan data yang tidak sesuai 3. Menekan tombol <i>login</i>
Hasil yang Diharapkan	Sistem dapat menampilkan pesan <i>user tidak dikenali!</i> atau <i>password salah!</i>
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan <i>username atau password tidak sesuai!</i>
Status Validasi	Valid

Tabel 6.14 Kasus uji lihat detail biodata

Nama Kasus Uji	Lihat detail biodata
Objek Uji	SKPL-02-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melihat detail biodata
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu akun 2. Aktor memilih menu <i>profil</i>
Hasil yang Diharapkan	Sistem akan menampilkan halaman biodata aktor yang melakukan login
Hasil Aktual	Sebelum dilakukan pengujian sistem dapat menampilkan halaman biodata aktor yang melakukan login ke dalam sistem
Status Validasi	Valid

Tabel 6.15 Kasus uji ubah biodata

Nama Kasus Uji	Ubah biodata
Objek Uji	SKPL-02-02
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat mengubah biodata
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih ikon ubah yang ada pada halaman profil 2. Aktor mengubah data pada form biodata 3. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem akan menampilkan pesan <i>data berhasil diubah!</i> dan menampilkan detail biodata
Hasil Aktual	Sebelum dilakukan pengujian sistem dapat menampilkan pesan <i>data berhasil diubah!</i> dan menampilkan detail biodata
Status Validasi	Valid

Tabel 6.16 Kasus uji ubah biodata alternatif 1 : Jika field yang wajib diisi kosong

Nama Kasus Uji	Ubah biodata
Objek Uji	SKPL-02-02
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan error jika <i>field</i> yang wajib diisi kosong

Tabel 6.16 (Lanjutan)

Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih ikon ubah yang ada pada halaman profil 2. Aktor mengokongkan <i>field</i> yang wajib diisi 3. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem akan menampilkan pesan <i>error</i> ketika <i>field</i> yang wajib diisi kosong
Hasil Aktual	Seletelah dilakukan pengujian sistem dapat menampilkan pesan <i>error</i> ketika <i>field</i> yang wajib diisi kosong
Status Validasi	Valid

Tabel 6.17 Kasus uji tambah staf

Nama Kasus Uji	Tambah staf
Objek Uji	SKPL-03-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menambah staf baru
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu staf 2. Aktor memilih menu tambah staf 3. Aktor mengisi form tambah staf 4. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem akan menampilkan pesan <i>staf berhasil ditambahkan!</i> dan menampilkan daftar staf
Hasil Aktual	Seletelah dilakukan pengujian sistem dapat menampilkan pesan <i>staf berhasil ditambahkan!</i> dan menampilkan daftar staf
Status Validasi	Valid

Tabel 6.18 Kasus uji tambah staf alternatif 1 : Jika field yang wajib diisi kosong

Nama Kasus Uji	Tambah staf
Objek Uji	SKPL-03-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan error jika <i>field</i> yang wajib diisi kosong
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu staf 2. Aktor memilih menu tambah staf 3. Aktor mengkosongkan <i>field</i> yang wajib diisi 4. Aktor menekan tombol submit

Tabel 6.18 (Lanjutan)

Hasil yang Diharapkan	Sistem akan menampilkan pesan <i>error</i> ketika <i>field</i> yang wajib diisi kosong
Hasil Aktual	Seletelah dilakukan pengujian sistem dapat menampilkan pesan <i>error</i> ketika <i>field</i> yang wajib diisi kosong
Status Validasi	Valid

Tabel 6.19 Kasus uji ubah staf

Nama Kasus Uji	Ubah staf
Objek Uji	SKPL-03-02
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat mengubah data staf
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih aksi ubah pada daftar staf yang akan diubah 2. Aktor ngeubah data staf 3. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem akan menampilkan pesan <i>data berhasil diubah!</i>
Hasil Aktual	Seletelah dilakukan pengujian sistem dapat menampilkan pesan <i>data berhasil diubah!</i>
Status Validasi	Valid

Tabel 6.20 Kasus uji ubah staf alternatif 1 : Jika *field* yang wajib diisi kosong

Nama Kasus Uji	Ubah staf
Objek Uji	SKPL-03-02
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan error jika <i>field</i> yang wajib diisi kosong
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilik aksi ubah 2. Aktor mengkosongkan <i>field</i> yang wajib diisi 3. Sistem menekan tombol submit
Hasil yang Diharapkan	Sistem akan menampilkan pesan <i>error</i> ketika <i>field</i> yang wajib diisi kosong
Hasil Aktual	Seletelah dilakukan pengujian sistem dapat menampilkan pesan <i>error</i> ketika <i>field</i> yang wajib diisi kosong
Status Validasi	Valid

Tabel 6.21 Kasus uji daftar staf

Nama Kasus Uji	Daftar staf
Objek Uji	SKPL-03-03
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan daftar staf
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu staf 2. Aktor memilih sub menu daftar staf
Hasil yang Diharapkan	Sistem akan menampilkan halaman daftar staf
Hasil Aktual	Setelah dilakukan pengujian sistem akan menampilkan halaman daftar staf
Status Validasi	Valid

Tabel 6.22 Kasus uji tambah permintaan buku

Nama Kasus Uji	Tambah permintaan buku
Objek Uji	SKPL-04-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menambah permintaan buku
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu permintaan 2. Aktor memilih sub menu individu 3. Aktor memilih jenis permintaan bahan pustaka 4. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem dapat menyimpan permintaan buku dan menampilkan pesan <i>permintaan buku berhasil ditambahkan!</i>
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menyimpan permintaan buku dan menampilkan pesan <i>permintaan buku berhasil ditambahkan!</i>
Status Validasi	Valid

Tabel 6.23 Kasus uji tambah permintaan buku alternatif 1 : Jika field yang wajib diisi kosong

Nama Kasus Uji	Ubah staf
Objek Uji	SKPL-04-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan <i>field cannot blank</i>

Tabel 6.23 (Lanjutan)

Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu permintaan 2. Aktor memilih sub menu individu 3. Aktor memilih jenis permintaan bahan pustaka buku 4. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan <i>field cannot blank</i> ketika <i>field</i> yang wajib diisi kosong
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan <i>field cannot blank</i> ketika <i>field</i> yang wajib diisi kosong
Status Validasi	Valid

Tabel 6.24 Kasus uji unduh form permintaan buku

Nama Kasus Uji	Unduh form permintaan buku
Objek Uji	SKPL-04-02
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melangukan unduh <i>form</i> permintaan buku
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu permintaan 2. Aktor memilih sub menu kolektif 3. Aktor menekan link unduh
Hasil yang Diharapkan	Sistem dapat melakukan unduh form permintaan buku dalam bentuk file <i>excel</i>
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan unduh form permintaan buku dalam bentuk file <i>excel</i>
Status Validasi	Valid

Tabel 6.25 Kasus uji unggah form permintaan buku

Nama Kasus Uji	Unggah form permintaan buku
Objek Uji	SKPL-04-03
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melangukan unggah <i>form</i> permintaan buku

Tabel 6.25 (Lanjutan)

Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu permintaan 2. Aktor memilih sub menu kolektif 3. Aktor memilih tab menu unggah permintaan 4. Aktor memilih kategori bahan pustaka buku dan unggah file excel 5. Aktor menekan tombol submit
Hasil yang Diharapkan	sistem dapat mengunggah form permintaan buku dan sistem dapat menampilkan pesan <i>data permintaan buku berhasil ditambahkan!</i>
Hasil Aktual	Setelah dilakukan pengujian sistem dapat mengunggah form permintaan buku dan sistem dapat menampilkan pesan <i>data permintaan buku berhasil ditambahkan!</i>
Status Validasi	Valid

Tabel 6.26 Kasus uji unggah form permintaan buku alternatif 1 : Jika yang diunggah tidak sesuai

Nama Kasus Uji	Unggah form permintaan buku
Objek Uji	SKPL-04-03
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan unggah form permintaan
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu permintaan 2. Aktor memilih sub menu kolektif 3. Aktor memilih tab menu unggah permintaan 4. Aktor memilih kategori bahan pustaka buku dan unggah file excel yang bukan disediakan 5. Aktor menekan tombol submit
Hasil yang Diharapkan	sistem dapat mengunggah form permintaan buku dan sistem dapat menampilkan pesan <i>data gagal dimasukkan. Pastikan file anda bernama FormReqBuku.xls!</i>
Hasil Aktual	Setelah dilakukan pengujian sistem dapat mengunggah form permintaan buku dan sistem dapat menampilkan pesan <i>data gagal dimasukkan. Pastikan file anda bernama FormReqBuku.xls!</i>
Status Validasi	Valid

Tabel 6.27 Kasus uji lihat daftar permintaan buku

Nama Kasus Uji	Lihat daftar permintaan buku
Objek Uji	SKPL-04-04
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan daftar permintaan buku
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu permintaan 2. Aktor memilih sub menu kolektif/individu 3. Aktor memilih tab menu daftar permintaan 4. Aktor memilih sub tab menu buku
Hasil yang Diharapkan	Sistem dapat menampilkan daftar permintaan buku
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan daftar permintaan buku
Status Validasi	Valid

Tabel 6.28 Kasus uji lihat daftar permintaan buku disetujui

Nama Kasus Uji	Lihat daftar permintaan buku disetujui
Objek Uji	SKPL-04-05
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan daftar permintaan buku yang sudah disetujui
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu pengolahan 2. Aktor memilih sub menu permintaan 3. Aktor memilih tab menu permintaan buku 4. Aktor memilih sub tab menu disetujui
Hasil yang Diharapkan	Sistem dapat menampilkan daftar permintaan buku yang sudah disetujui
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan daftar permintaan buku yang sudah disetujui
Status Validasi	Valid

Tabel 6.29 Kasus uji lihat daftar permintaan buku belum disetujui

Nama Kasus Uji	Lihat daftar permintaan buku belum disetujui
Objek Uji	SKPL-04-06
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem akan menampilkan daftar permintaan buku yang belum disetujui
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu pengolahan 2. Aktor memilih sub menu permintaan 3. Aktor memilih tab menu permintaan buku 4. Aktor memilih sub tab menu belum disetujui
Hasil yang Diharapkan	Sistem akan menampilkan daftar permintaan buku yang belum disetujui
Hasil Aktual	Setelah dilakukan pengujian sistem akan menampilkan daftar permintaan buku yang belum disetujui
Status Validasi	Valid

Tabel 6.30 Kasus uji unduh daftar permintaan buku disetujui

Nama Kasus Uji	Unduh daftar permintaan buku disetujui
Objek Uji	SKPL-04-07
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan unduh daftar permintaan buku disetujui
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu pengolahan 2. Aktor memilih sub menu permintaan 3. Aktor memilih tab menu permintaan buku kemudian sub tab menu disetujui 4. Aktor menekan link unduh
Hasil yang Diharapkan	sistem dapat melakukan unduh daftar permintaan buku disetujui
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan unduh daftar permintaan buku disetujui
Status Validasi	Valid

Tabel 6.31 Kasus uji unduh daftar permintaan buku belum disetujui

Nama Kasus Uji	Unduh daftar permintaan buku belum disetujui
Objek Uji	SKPL-04-08
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan unduh daftar permintaan buku belum disetujui
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu pengolahan 2. Aktor memilih sub menu permintaan 3. Aktor memilih tab menu permintaan buku kemudian sub tab menu belum disetujui 4. Aktor menekan link unduh
Hasil yang Diharapkan	sistem dapat melakukan unduh daftar permintaan buku belum disetujui
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan unduh daftar permintaan buku belum disetujui
Status Validasi	Valid

Tabel 6.32 Kasus uji lihat detail permintaan buku

Nama Kasus Uji	Lihat detail permintaan buku
Objek Uji	SKPL-04-09
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan detail permintaan bahan pustaka buku
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih tab menu permintaan buku 2. Aktor memilih sub tab permintaan buku disetujui / belum disetujui 3. Aktor memilih ikon detail pada daftar permintaan buku
Hasil yang Diharapkan	Sistem dapat menampilkan detail permintaan bahan pustaka buku
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan detail permintaan bahan pustaka buku
Status Validasi	Valid

Tabel 6.33 Kasus uji ubah permintaan buku

Nama Kasus Uji	Ubah permintaan buku
Objek Uji	SKPL-04-10
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan ubah permintaan bahan pustaka buku
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih ikon edit pada daftar permintaan buku disetujui / belum disetujui 2. Aktor merubah data buku 3. Aktor menekan tombol submit
Hasil yang Diharapkan	sistem dapat melakukan ubah permintaan bahan pustaka buku
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan ubah permintaan bahan pustaka buku
Status Validasi	Valid

Tabel 6.34 Kasus uji ubah permintaan buku alternatif 1 : Jika field yang wajib diisi kosong

Nama Kasus Uji	Ubah permintaan buku
Objek Uji	SKPL-04-10
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan ketika field yang wajib diisi kosong
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih ikon edit pada daftar permintaan buku disetujui / belum disetujui 2. Aktor merubah data buku 3. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan ketika field yang wajib diisi kosong
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan ketika field yang wajib diisi kosong
Status Validasi	Valid

Tabel 6.35 Kasus uji hapus permintaan buku

Nama Kasus Uji	Hapus permintaan buku
Objek Uji	SKPL-04-11
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menghapus permintaan bahan pustaka
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih ikon hapus pada daftar permintaan bahan pustaka buku 2. Aktor memilih tombol ya
Hasil yang Diharapkan	Sistem dapat menghapus permintaan bahan pustaka
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menghapus permintaan bahan pustaka
Status Validasi	Valid

Tabel 6.36 Kasus uji cari permintaan buku belum disetujui

Nama Kasus Uji	Cari permintaan buku belum disetujui
Objek Uji	SKPL-04-12
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan pencarian permintaan bahan pustaka buku belum disetujui berdasarkan ISBN
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih sub tab belum disetujui 2. Aktor mengisi kata kunci pencarian berdasarkan ISBN 3. Aktor menekan tombol cari
Hasil yang Diharapkan	Sistem dapat menampilkan daftar permintaan bahan pustaka buku yang dicari
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan daftar permintaan bahan pustaka buku yang dicari
Status Validasi	Valid

Tabel 6.37 Kasus uji tambah buku dari permintaan

Nama Kasus Uji	Tambah buku dari permintaan
Objek Uji	SKPL-04-13
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan tambah buku dari daftar permintaan
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih ikon tambah pada daftar permintaan buku belum disetujui 2. Aktor melengkapi informasi data buku 3. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem dapat melakukan tambah buku dari daftar permintaan dan menampilkan pesan data buku berhasil dimasukkan serta mengirim pesan kepada anggota yang mengajukan permintaan melalui email
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan tambah buku dari daftar permintaan dan menampilkan pesan data buku berhasil dimasukkan serta mengirim pesan kepada anggota yang mengajukan permintaan melalui email
Status Validasi	Valid

Tabel 6.38 Kasus uji tambah permintaan jurnal

Nama Kasus Uji	Tambah permintaan jurnal
Objek Uji	SKPL-04-14
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan tambah permintaan jurnal
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu permintaan 2. Aktor memilih sub menu individu 3. Aktor memilih jenis permintaan bahan pustaka jurnal 4. Aktor mengnekan tombol submit
Hasil yang Diharapkan	Sistem dapat menambah permintaan jurnal dan menampilkan pesan <i>permintaan jurnal berhasil ditambahkan!</i>
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menambah permintaan jurnal dan menampilkan pesan <i>permintaan jurnal berhasil ditambahkan!</i>
Status Validasi	Valid

Tabel 6.39 Kasus uji tambah permintaan jurnal alternatif 1 : Jika field yang wajib diisi kosong

Nama Kasus Uji	Tambah permintaan jurnal
Objek Uji	SKPL-04-14
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan ketika filed yang wajib diisi kosong
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu permintaan 2. Aktor memilih sub menu individu 3. Aktor memilih jenis permintaan bahan pustaka jurnal 4. Aktor mengnekan tombol submit
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan ketika filed yang wajib diisi kosong
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan ketika filed yang wajib diisi kosong
Status Validasi	Valid

Tabel 6.40 Kasus uji unduh form permintaan jurnal

Nama Kasus Uji	Unduh form permintaan jurnal
Objek Uji	SKPL-04-15
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan unduh form permintaan jurnal
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu permintaan 2. Aktor memilih sub menu kolektif 3. Aktor menekan link unduh
Hasil yang Diharapkan	Sistem dapat melakukan unduh form permintaan jurnal
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan unduh form permintaan jurnal
Status Validasi	Valid

Tabel 6.41 Kasus uji unggah form permintaan jurnal

Nama Kasus Uji	Unggah form permintaan jurnal
Objek Uji	SKPL-04-16
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan unggah form permintaan jurnal
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu permintaan 2. Aktor memilih sub menu kolektif 3. Aktor memilih tab menu unggah permintaan 4. Aktor memilih kategori bahan pustaka jurnal dan unggah file excel 5. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem dapat melakukan unggah form permintaan jurnal dan menampilkan pesan data berhasil dimasukkan
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan unggah form permintaan jurnal dan menampilkan pesan data berhasil dimasukkan
Status Validasi	Valid

Tabel 6.42 Kasus uji unggah form permintaan jurnal alternatif 1 : Jika file yang diunggah tidak sesuai

Nama Kasus Uji	Unggah form permintaan jurnal
Objek Uji	SKPL-04-16
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan file yang diunggah tidak sesuai
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu permintaan 2. Aktor memilih sub menu kolektif 3. Aktor memilih tab menu unggah permintaan 4. Aktor memilih kategori bahan pustaka jurnal dan unggah file excel yang tidak sesuai 5. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan file yang diunggah tidak sesuai

Tabel 6.42 (Lanjutan)

Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan file yang diunggah tidak sesuai
Status Validasi	Valid

Tabel 6.43 Kasus uji lihat daftar permintaan jurnal

Nama Kasus Uji	Lihat daftar permintaan jurnal
Objek Uji	SKPL-04-17
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan daftar permintaan jurnal
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu permintaan 2. Aktor memilih sub menu kolektif/individu 3. Aktor memilih tab menu daftar permintaan 4. Aktor memilih sub tab menu jurnal
Hasil yang Diharapkan	Sistem dapat menampilkan daftar permintaan jurnal
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan daftar permintaan jurnal
Status Validasi	Valid

Tabel 6.44 Kasus uji lihat daftar permintaan jurnal disetujui

Nama Kasus Uji	Lihat daftar permintaan jurnal disetujui
Objek Uji	SKPL-04-18
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan daftar permintaan jurnal yang disetujui
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu pengolahan 2. Aktor memilih sub menu permintaan 3. Aktor memilih tab menu permintaan jurnal 4. Aktor memilih sub tab menu disetujui
Hasil yang Diharapkan	Sistem dapat menampilkan daftar permintaan jurnal yang disetujui
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan daftar permintaan jurnal yang disetujui
Status Validasi	Valid

Tabel 6.45 Kasus uji lihat daftar permintaan jurnal belum disetujui

Nama Kasus Uji	Lihat daftar permintaan jurnal belum disetujui
Objek Uji	SKPL-04-19
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan daftar permintaan jurnal yang belum disetujui
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu pengolahan 2. Aktor memilih sub menu permintaan 3. Aktor memilih tab menu permintaan buku 4. Aktor memilih sub tab menu belum disetujui
Hasil yang Diharapkan	Sistem dapat menampilkan daftar permintaan jurnal yang belum disetujui
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan daftar permintaan jurnal yang belum disetujui
Status Validasi	Valid

Tabel 6.46 Kasus uji unduh daftar permintaan jurnal disetujui

Nama Kasus Uji	Unduh daftar permintaan jurnal disetujui
Objek Uji	SKPL-04-20
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan melakukan unduh daftar permintaan jurnal yang disetujui
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu pengolahan 2. Aktor memilih sub menu permintaan 3. Aktor memilih tab menu permintaan jurnal kemudian sub tab menu disetujui 4. Aktor mengisi tanggal dan memilih <i>combobox</i> bahasa permintaan jurnal 5. Aktor menekan link unduh
Hasil yang Diharapkan	Sistem dapat menampilkan melakukan unduh daftar permintaan jurnal yang disetujui
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan melakukan unduh daftar permintaan jurnal yang disetujui
Status Validasi	Valid

Tabel 6.47 Kasus uji unduh daftar permintaan jurnal belum disetujui

Nama Kasus Uji	Lihat daftar permintaan jurnal belum disetujui
Objek Uji	SKPL-04-21
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan unduh daftar permintaan jurnal yang belum disetujui
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu pengolahan 2. Aktor memilih sub menu permintaan 3. Aktor memilih tab menu permintaan jurnal kemudian sub tab menu belum disetujui 4. Aktor memilih <i>combox</i> bahasa permintaan jurnal 5. Aktor menekan link unduh
Hasil yang Diharapkan	Sistem dapat menampilkanmelakukan unduh daftar permntaan jurnal yang belum disetujui
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkanmelakukan unduh daftar permntaan jurnal yang belum disetujui
Status Validasi	Valid

Tabel 6.48 Kasus uji lihat detail permintaan jurnal

Nama Kasus Uji	Lihat detail permintaan jurnal
Objek Uji	SKPL-04-22
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan detail permintaan jurnal
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih sub tab menu disetujui / belum disetujui 2. Aktor memilih ikon detail pada daftar permintaan jurnal
Hasil yang Diharapkan	Sistem dapat menampilkan detail permintaan jurnal
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan detail permintaan jurnal
Status Validasi	Valid

Tabel 6.49 Kasus uji ubah permintaan jurnal

Nama Kasus Uji	Ubah permintaan jurnal
Objek Uji	SKPL-04-23
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan ubah permintaan jurnal
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih sub tab menu disetujui / belum disetujui 2. Aktor memilih tombol ubah pada daftar permintaan jurnal 3. Aktor merubah data jurnal 4. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem dapat melakukan ubah permintaan jurnal dan menampilkan pesan data telah berhasil diubah
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan ubah permintaan jurnal dan menampilkan pesan data telah berhasil diubah
Status Validasi	Valid

Tabel 6.50 Kasus uji ubah permintaan jurnal alternatif 1 : Jika *field* yang wajib diisi kosong

Nama Kasus Uji	Ubah permintaan jurnal
Objek Uji	SKPL-04-13
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan field tidak boleh kosong pada form ubah permintaan jurnal
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih sub tab menu disetujui / belum disetujui 2. Aktor memilih tombol ubah pada daftar permintaan jurnal 3. Aktor merubah data jurnal 4. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan bahwa field tidak boleh kosong pada form ubah permintaan jurnal

Tabel 6.50 (Lanjutan)

Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan bahwa field tidak boleh kosong pada form ubah permintaan jurnal
Status Validasi	Valid

Tabel 6.51 Kasus uji hapus permintaan jurnal

Nama Kasus Uji	Hapus permintaan jurnal
Objek Uji	SKPL-04-24
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menghapus permintaan jurnal
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih sub tab menu disetujui / belum disetujui 2. Aktor memilih tombol hapus pada daftar permintaan jurnal 3. Aktor menekan tombol ya
Hasil yang Diharapkan	Sistem dapat menghapus permintaan jurnal
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menghapus permintaan jurnal
Status Validasi	Valid

Tabel 6.52 Kasus uji cari permintaan jurnal belum disetujui

Nama Kasus Uji	Hapus permintaan jurnal
Objek Uji	SKPL-04-25
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan pencarian permintaan jurnal yang belum disetujui
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih sub tab belum disetujui 2. Aktor mengisi kata kunci pencarian berdasarkan ISSN 3. Aktor menekan tombol cari
Hasil yang Diharapkan	Sistem dapat melakukan pencarian permintaan jurnal yang belum disetujui
Hasil Aktual	Setelah dilakukan pencarian sistem dapat melakukan pencarian permintaan jurnal yang belum disetujui
Status Validasi	Valid

Tabel 6.53 Kasus uji tambah jurnal dari permintaan

Nama Kasus Uji	Tambah jurnal dari permintaan
Objek Uji	SKPL-04-26
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menambah jurnal dari daftar permintaan jurnal
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih ikon tambah pada daftar permintaan jurnal belum disetujui 2. Aktor melengkapi informasi data jurnal 3. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem dapat menambah jurnal dari daftar permintaan jurnal dan mengirim pesan kepada anggota yang mengajukan permintaan melalui email
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menambah jurnal dari daftar permintaan jurnal dan mengirim pesan kepada anggota yang mengajukan permintaan melalui email
Status Validasi	Valid

Tabel 6.54 Kasus uji tambah buku baru

Nama Kasus Uji	Tambah buku baru
Objek Uji	SKPL-05-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menambah buku baru / buku label merah
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih sub menu registrasi buku 2. Aktor memasukkan data informasi buku 3. Aktor menekan tombol submit informasi
Hasil yang Diharapkan	Sistem dapat menambah buku baru / buku label merah dan menampilkan pesan data berhasil ditambahkan
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menambah buku baru / buku label merah dan menampilkan pesan data berhasil ditambahkan
Status Validasi	Valid

Tabel 6.55 Kasus uji tambah buku baru alternatif 1 : Jika *field field* yang wajib diisi kosong

Nama Kasus Uji	Tambah buku baru
Objek Uji	SKPL-05-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan ketika field yang wajib diisi kosong
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih sub menu registrasi buku 2. Aktor mengkononkan field yang wajib diisi 3. Aktor menekan tombol submit informasi
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan ketika field yang wajib diisi kosong
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan ketika field yang wajib diisi kosong
Status Validasi	Valid

Tabel 6.56 Kasus uji tambah buku baru alternatif 2 : Jika karakter yang dimasukkan tidak sesuai

Nama Kasus Uji	Tambah buku baru
Objek Uji	SKPL-05-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan ketika karakter yang dimasukkan tidak sesuai
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih sub menu registrasi buku 2. Aktor memasukkan karakter yang tidak sesuai 3. Aktor menekan tombol submit informasi
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan ketika karakter yang dimasukkan tidak sesuai
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan ketika karakter yang dimasukkan tidak sesuai
Status Validasi	Valid

Tabel 6.57 Kasus uji unduh form tambah buku

Nama Kasus Uji	Unduh form tambah buku
Objek Uji	SKPL-05-02
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan unduh form tambah buku
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu kolektif 2. Aktor menekan <i>link</i> unduh
Hasil yang Diharapkan	Sistem dapat melakukan unduh form tambah buku
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan unduh form tambah buku
Status Validasi	Valid

Tabel 6.58 Kasus uji unggah form tambah buku baru

Nama Kasus Uji	Unggah <i>form</i> tambah buku baru
Objek Uji	SKPL-05-03
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan unggah <i>form</i> tambah buku
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu kolektif 2. Aktor menekan <i>button browse</i> dan memilih <i>file excel</i> yang akan diunggah 3. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem dapat melakukan unggah <i>form</i> tambah buku dan menampilkan pesan <i>Data permintaan buku berhasil ditambahkan!</i>
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan unggah <i>form</i> tambah buku dan menampilkan pesan <i>Data permintaan buku berhasil ditambahkan!</i>
Status Validasi	Valid

Tabel 6.59 Kasus uji unggah form tambah buku baru alternatif 1: jika file yang diunggah tidak sesuai dengan jenis bahan pustaka

Nama Kasus Uji	Unggah <i>form</i> tambah buku baru
Objek Uji	SKPL-05-03
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan ketika file yang diupload tidak sesuai

Tabel 6.59 (Lanjutan)

Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu kolektif 2. Aktor menekan <i>button browse</i> dan memilih <i>file excel</i> yang akan diunggah 3. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan ketika file yang diupload tidak sesuai
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan ketika file yang diupload tidak sesuai
Status Validasi	Valid

Tabel 6.60 Kasus uji lihat daftar buku label merah

Nama Kasus Uji	Lihat daftar buku label merah
Objek Uji	SKPL-05-04
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan daftar buku yang berlabel merah
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah
Hasil yang Diharapkan	Sistem dapat menampilkan daftar buku yang berlabel merah
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan daftar buku yang berlabel merah
Status Validasi	Valid

Tabel 6.61 Kasus uji cari buku label merah

Nama Kasus Uji	Cari buku label merah
Objek Uji	SKPL-05-05
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan pencarian buku label merah
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 2. Aktor memilih jenis pencarian dan mengisi field pencarian 3. Aktor menekan <i>button cari</i>
Hasil yang Diharapkan	Sistem dapat melakukan pencarian buku label merah
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan pencarian buku label merah
Status Validasi	Valid

Tabel 6.62 Kasus uji cari buku label merah alternatif 1 : Jika tidak memilih jenis pencarian

Nama Kasus Uji	Cari buku label merah
Objek Uji	SKPL-05-05
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan ketika tidak dipilih jenis pencarian
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 2. Aktor tidak memilih jenis pencarian 3. Aktor menekan button cari
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan ketika tidak dipilih jenis pencarian
Hasil Aktual	Sistem dapat menampilkan pesan kesalahan ketika tidak dipilih jenis pencarian
Status Validasi	Valid

Tabel 6.63 Kasus uji lihat detail buku label merah

Nama Kasus Uji	Lihat detail buku label merah
Objek Uji	SKPL-05-06
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem melihat detail buku label merah
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 2. Aktor menekan ikon detail yang terdapat pada daftar buku
Hasil yang Diharapkan	Sistem melihat detail buku label merah
Hasil Aktual	Setelah dilakukan pengujian sistem melihat detail buku label merah
Status Validasi	Valid

Tabel 6.64 Kasus uji ubah buku label merah

Nama Kasus Uji	Ubah buku label merah
Objek Uji	SKPL-05-07
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sitem dapat melakukan ubah buku label merah

Tabel 6.64 (Lanjutan)

Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 2. Aktor menekan ikon ubah yang terdapat pada daftar buku 3. Aktor mengubah data yang ingin dirubah dan menekan button submit
Hasil yang Diharapkan	Sitem dapat melakukan ubah buku label merah dan menampilkan pesan data berhasil diubah
Hasil Aktual	Setelah dilakukan pengujian sitem dapat melakukan ubah buku label merah dan menampilkan pesan data berhasil diubah
Status Validasi	Valid

Tabel 6.65 Kasus uji ubah buku label merah alternatif 1 : jika *field* yang wajib diisi kosong

Nama Kasus Uji	Ubah buku label merah
Objek Uji	SKPL-05-07
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan ketika field yang wajib diisi pada ubah buku kosong
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 2. Aktor menekan ikon ubah yang terdapat pada daftar buku 3. Aktor mengubah data yang ingin dirubah dan menekan button submit
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan ketika field yang wajib diisi pada ubah buku kosong
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan ketika field yang wajib diisi pada ubah buku kosong
Status Validasi	Valid

Tabel 6.66 Kasus uji tambah jurnal baru

Nama Kasus Uji	Tambah jurnal baru
Objek Uji	SKPL-05-08
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan tambah jurnal

Tabel 6.66 (Lanjutan)

Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih tab menu individu 2. Aktor memasukkan data informasi jurnal dan menekan tombol submit informasi
Hasil yang Diharapkan	Sistem dapat melakukan tambah jurnal dan menampilkan pesan berhasil menambah data jurnal
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan tambah jurnal dan menampilkan pesan berhasil menambah data jurnal
Status Validasi	Valid

Tabel 6.67 Kasus uji tambah jurnal baru alternatif 1 : jika *field* yang wajib diisi kosong

Nama Kasus Uji	Tambah jurnal baru
Objek Uji	SKPL-05-08
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan ketika <i>field</i> yang wajib diisi pada <i>form</i> tambah jurnal kosong
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih tab menu individu 2. Aktor mengosongkan field yang wajib diisi 3. Aktor menekan tombol submit
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan ketika <i>field</i> yang wajib diisi pada <i>form</i> tambah jurnal kosong
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan ketika <i>field</i> yang wajib diisi pada <i>form</i> tambah jurnal kosong
Status Validasi	Valid

Tabel 6.68 Kasus uji tambah jurnal baru alternatif 2 : jika karakter yang diinputkan tidak sesuai

Nama Kasus Uji	Tambah jurnal baru
Objek Uji	SKPL-05-08
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan ketika karakter yang dimasukkan tidak sesuai
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih tab menu individu 2. Aktor memasukkan karakter yang tidak sesuai 3. Aktor menekan tombol submit

Tabel 6.68 (Lanjutan)

Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan ketika karakter yang dimasukkan tidak sesuai
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan ketika karakter yang dimasukkan tidak sesuai
Status Validasi	Valid

Tabel 6.69 Kasus uji unduh form tambah jurnal

Nama Kasus Uji	Unduh form tambah jurnal
Objek Uji	SKPL-05-09
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan funduh form tambah jurnal
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu kolektif 2. Aktor menekan <i>link</i> unduh <i>form</i> tambah jurnal
Hasil yang Diharapkan	Sistem dapat melakukan funduh form tambah jurnal
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan funduh form tambah jurnal
Status Validasi	Valid

Tabel 6.70 Kasus uji unggah form tambah jurnal

Nama Kasus Uji	Unggah form tambah jurnal
Objek Uji	SKPL-05-10
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan uanggah form tambah jurnal
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu kolektif 2. Aktor menekan <i>button browse</i> dan memilih <i>file excel</i> yang akan diunggah 3. Aktor menekan tombol sumit
Hasil yang Diharapkan	Sistem dapat melakukan uanggah form tambah jurnal dan pesan berhasilunggah file
Hasil Aktual	Sistem dapat melakukan uanggah form tambah jurnal dan pesan berhasil unggah file
Status Validasi	Valid

Tabel 6.71 Kasus uji unggah form tambah jurnal alternatif 1 : Jika file excel yang di unggah tidak sesuai dengan jenis bahan pustaka

Nama Kasus Uji	Unggah form tambah jurnal
Objek Uji	SKPL-05-10
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan ketika salah mengunggah file
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu kolektif 2. Aktor menekan <i>button browse</i> dan memilih <i>file excel</i> yang tidak sesuai 3. Aktor menekan tombol sumit
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan ketika salah mengunggah file
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan ketika salah mengunggah file
Status Validasi	Valid

Tabel 6.72 Kasus uji lihat daftar jurnal label merah

Nama Kasus Uji	Lihat daftar jurnal label merah
Objek Uji	SKPL-05-11
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan daftar jurnal label merah
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah
Hasil yang Diharapkan	Sistem dapat menampilkan daftar jurnal label merah
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan daftar jurnal label merah
Status Validasi	Valid

Tabel 6.73 Kasus uji cari jurnal label merah

Nama Kasus Uji	Cari jurnal label merah
Objek Uji	SKPL-05-12
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan pencarian berdasarkan jenis pencarian

Tabel 6.73 (Lanjutan)

Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 2. Aktor memilih kategori pencarian dan mengisi field pencarian 3. Aktor menekan button cari
Hasil yang Diharapkan	Sistem akan menampilkan daftar jurnal label merah berdasarkan pencarian
Hasil Aktual	Setelah dilakukan pengujian sistem akan menampilkan daftar jurnal label merah berdasarkan pencarian
Status Validasi	Valid

Tabel 6.74 Kasus uji cari jurnal label merah alternatif 1 : jika *field* pencarian kosong

Nama Kasus Uji	Cari jurnal label merah
Objek Uji	SKPL-05-12
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan daftar jurnal label merah yang dicari atau pesan kesalahan <i>pilih jenis pencarian!</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 2. Aktor memilih kategori pencarian dan mengisi field pencarian 3. Aktor menekan button cari
Hasil yang Diharapkan	Sistem akan menampilkan daftar jurnal label merah yang dicari atau pesan kesalahan <i>pilih jenis pencarian!</i>
Hasil Aktual	Setelah dilakukan pengujian sistem akan menampilkan daftar jurnal label merah yang dicari atau pesan kesalahan <i>pilih jenis pencarian!</i>
Status Validasi	Valid

Tabel 6.75 Kasus uji lihat detail jurnal label merah

Nama Kasus Uji	Lihat daftar jurnal label merah
Objek Uji	SKPL-05-13
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan detail jurnal

Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 2. Aktor menekan tombol detail yang terdapat pada daftar jurnal
Hasil yang Diharapkan	Sistem dapat menampilkan detail jurnal
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan detail jurnal
Status Validasi	Valid

Tabel 6.76 Kasus uji ubah jurnal label merah

Nama Kasus Uji	Ubah jurnal label merah
Objek Uji	SKPL-05-14
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan ubah data jurnal label merah
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 2. Aktor menekan ikon ubah yang terdapat pada daftar bajurnal 3. Aktor mengubah data yang ingin dirubah dan menekan button submit
Hasil yang Diharapkan	Sistem dapat melakukan ubah data jurnal label merah dan menampilkan detal jurnal setelah di ubah
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan ubah data jurnal label merah dan menampilkan detal jurnal setelah di ubah
Status Validasi	Valid

Tabel 6.77 Kasus uji ubah jurnal label merah alternatif 1 : jika *field* yang wajib diisi kosong

Nama Kasus Uji	Ubah jurnal label merah
Objek Uji	SKPL-05-14
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan field yang wajib diisi tidak boleh kosong
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 2. Aktor menekan ikon ubah yang terdapat pada daftar bajurnal 3. Aktor mengosongkan field yang wajib diisi dan menekan button submit

Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan field yang wajib diisi tidak boleh kosong
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan field yang wajib diisi tidak boleh kosong
Status Validasi	Valid

Tabel 6.78 Kasus uji tambah buku label putih

Nama Kasus Uji	Tambah buku label putih
Objek Uji	SKPL-06-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan tambah buku label putih / salinan buku
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah / label putih 2. Aktor menekan ikon tambah yang terdapat pada daftar buku 3. Aktor melengkapi data buku dan menekan button submit
Hasil yang Diharapkan	Sistem dapat melakukan tambah buku label putih / buku salinan
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan tambah buku label putih / buku salinan
Status Validasi	Valid

Tabel 6.79 Kasus uji lihat daftar buku label putih

Nama Kasus Uji	Lihat daftar buku label putih
Objek Uji	SKPL-06-02
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan daftar buku label putih
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label putih
Hasil yang Diharapkan	Sistem dapat menampilkan daftar buku label putih
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan daftar buku label putih
Status Validasi	Valid

Tabel 6.80 Kasus uji cari buku label putih

Nama Kasus Uji	Cari buku label putih
Objek Uji	SKPL-06-03
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan pencarian buku label putih
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label putih 2. Aktor memilih jenis pencarian dan mengisi field pencarian 3. Aktor menekan button cari
Hasil yang Diharapkan	Sistem dapat melakukan pencarian buku label putih dan menampilkan daftar buku label putih yang dicari
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan pencarian buku label putih dan menampilkan daftar buku label putih yang dicari
Status Validasi	Valid

Tabel 6.81 Kasus uji cari buku label putih alternatif 1 : jika tidak memilih jenis pencarian

Nama Kasus Uji	Cari buku label putih
Objek Uji	SKPL-06-03
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan jika tidak memilih jenis pencarian
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label putih 2. Aktor tidak memilih jenis pencarian dan mengisi field pencarian 3. Aktor menekan button cari
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan <i>pilih jenis pencarian!</i>
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan <i>pilih jenis pencarian!</i>
Status Validasi	Valid

Tabel 6.82 Kasus uji ubah buku label putih

Nama Kasus Uji	Ubah buku label putih
Objek Uji	SKPL-06-04
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan ubah data buku label putih
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label putih 2. Aktor menekan ikon ubah yang terdapat pada daftar buku 3. Aktor mengubah data yang ingin dirubah dan menekan button simpan
Hasil yang Diharapkan	Sistem dapat melakukan ubah data buku label putih dan menampilkan pesan <i>Ubah Data Berhasil!</i>
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan ubah data buku label putih dan menampilkan pesan <i>Ubah Data Berhasil!</i>
Status Validasi	Valid

Tabel 6.83 Kasus uji lihat detail buku buku label putih

Nama Kasus Uji	Lihat detail buku label putih
Objek Uji	SKPL-06-05
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan detail buku yang berlabel putih
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label putih 2. Aktor menekan tombol detail yang terdapat pada daftar buku label putih
Hasil yang Diharapkan	Sistem dapat menampilkan detail buku yang berlabel putih
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan detail buku yang berlabel putih
Status Validasi	Valid

Tabel 6.84 Kasus uji tambah jurnal label putih

Nama Kasus Uji	Tambah jurnal label putih
Objek Uji	SKPL-06-07
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan tambah jurnal yang berlabel putih
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah / label putih 2. Aktor menekan ikon tambah yang terdapat pada daftar jurnal 3. Aktor melengkapi data buku dan menekan button submit
Hasil yang Diharapkan	Sistem dapat melakukan tambah jurnal yang berlabel putih
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan tambah jurnal yang berlabel putih
Status Validasi	Valid

Tabel 6.85 Kasus uji lihat daftar jurnal label putih

Nama Kasus Uji	Lihat daftar jurnal label putih
Objek Uji	SKPL-06-08
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan daftar jurnal yang berlabel putih
Prosedur Uji	1. Aktor menekan <i>tab</i> menu label putih
Hasil yang Diharapkan	Sistem dapat menampilkan daftar jurnal yang berlabel putih
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan daftar jurnal yang berlabel putih
Status Validasi	Valid

Tabel 6.86 Kasus uji cari jurnal label putih

Nama Kasus Uji	Cari jurnal label putih
Objek Uji	SKPL-06-09
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan pencarian jurnal yang belabel putih dan menampilkan daftar pencarian

Tabel 6.86 (Lanjutan)

Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label putih 2. Aktor memilih jenis pencarian dan mengisi field pencarian 3. Aktor menekan button cari
Hasil yang Diharapkan	Sistem dapat melakukan pencarian jurnal yang belabel putih dan menampilkan daftar pencarian
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan pencarian jurnal yang belabel putih dan menampilkan daftar pencarian
Status Validasi	Valid

Tabel 6.87 Kasus uji cari jurnal label putih alternatif 1 : Jika tidak memilih jenis pencarian

Nama Kasus Uji	Cari jurnal label putih
Objek Uji	SKPL-06-09
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem menampilkan pesan kesalahan <i>pilih jenis pencarian!</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label putih 2. Aktor tidak memilih jenis pencarian dan mengisi field pencarian 3. Aktor menekan button cari
Hasil yang Diharapkan	Sistem menampilkan pesan kesalahan <i>pilih jenis pencarian!</i>
Hasil Aktual	Setelah dilakukan pengujian sistem menampilkan pesan kesalahan <i>pilih jenis pencarian!</i>
Status Validasi	Valid

Tabel 6.88 Kasus uji ubah jurnal label putih

Nama Kasus Uji	Ubah jurnal label putih
Objek Uji	SKPL-06-10
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan ubah data jurnal label putih
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label putih 2. Aktor menekan ikon ubah yang terdapat pada daftar jurnal 3. Aktor mengubah data yang ingin dirubah dan menekan button simpan

Tabel 6.88 (Lanjutan)

Hasil yang Diharapkan	Sistem dapat melakukan ubah data jurnal label putih dan menampilkan pesan <i>Ubah Data Berhasil!</i>
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan ubah data jurnal label putih dan menampilkan pesan <i>Ubah Data Berhasil!</i>
Status Validasi	Valid

Tabel 6.89 Kasus uji lihat detail jurnal label putih

Nama Kasus Uji	Lihat detail jurnal label putih
Objek Uji	SKPL-06-11
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan detail jurnal yang berlabel putih
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label putih 2. Aktor menekan tombol detail yang terdapat pada daftar jurnal label putih
Hasil yang Diharapkan	Sistem dapat menampilkan detail jurnal yang berlabel putih
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan detail jurnal yang berlabel putih
Status Validasi	Valid

Tabel 6.90 Kasus uji cetak barcode dan label buku label merah

Nama Kasus Uji	Cetak barcode dan label buku label merah
Objek Uji	SKPL-07-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan cetak barcode dan label buku label merah
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 2. Aktor melakukan <i>ceklis</i> pada <i>checkbox</i> yang terdapat pada daftar buku, kemudian menekan <i>button</i> cetak barcode dan label 3. Aktor menekan tombol Ctrl+p
Hasil yang Diharapkan	Sistem dapat melakukan cetak barcode dan label buku label merah

Tabel 6.90 (Lanjutan)

Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan cetak barcode dan label buku label merah
Status Validasi	Valid

Tabel 6.91 Kasus uji cetak barcode dan label buku label merah alternatif 1 : Jika tidak melakukan *ceklis* pada *checkbox*

Nama Kasus Uji	Cetak barcode dan label buku label merah
Objek Uji	SKPL-07-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan <i>Harap pilih buku yang akan dicetak register dan labelnya!</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 2. Aktor tidak melakukan <i>ceklis</i> pada <i>checkbox</i> yang terdapat pada daftar buku, kemudian menekan <i>button</i> cetak barcode dan label
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan <i>Harap pilih buku yang akan dicetak register dan labelnya!</i>
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan <i>Harap pilih buku yang akan dicetak register dan labelnya!</i>
Status Validasi	Valid

Tabel 6.92 Kasus uji cetak barcode dan label buku label putih

Nama Kasus Uji	Cetak barcode dan label buku label putih
Objek Uji	SKPL-07-02
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan cetak barcode dan label buku label putih
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label putih 2. Aktor melakukan <i>ceklis</i> pada <i>checkbox</i> yang terdapat pada daftar buku, kemudian menekan <i>button</i> cetak barcode dan label 3. Aktor menekan tombol Ctrl+p
Hasil yang Diharapkan	Sistem dapat melakukan cetak barcode dan label buku label putih
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan cetak barcode dan label buku label putih
Status Validasi	Valid

Tabel 6.93 Kasus uji cetak barcode dan label buku label putih alternatif 1 : Jika tidak melakukan *ceklis* pada *cekbox*

Nama Kasus Uji	Cetak barcode dan label buku label putih
Objek Uji	SKPL-07-02
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan <i>Harap pilih buku yang akan dicetak register dan labelnya!</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label putih 2. Aktor tidak melakukan <i>ceklis</i> pada <i>cekbox</i> yang terdapat pada daftar buku, kemudian menekan <i>button</i> cetak barcode dan label
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan <i>Harap pilih buku yang akan dicetak register dan labelnya!</i>
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan <i>Harap pilih buku yang akan dicetak register dan labelnya!</i>
Status Validasi	Valid

Tabel 6.94 Kasus uji cetak barcode dan label jurnal label merah

Nama Kasus Uji	Cetak barcode dan label jurnal label merah
Objek Uji	SKPL-07-03
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan cetak barcode dan label jurnal label merah
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 2. Aktor melakukan <i>ceklis</i> pada <i>cekbox</i> yang terdapat pada daftar jurnal, kemudian menekan <i>button</i> cetak barcode dan label 3. Aktor menekan tombol Ctrl+p
Hasil yang Diharapkan	Sistem dapat melakukan cetak barcode dan label jurnal label merah
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan cetak barcode dan label jurnal label merah
Status Validasi	Valid

**Tabel 6.95 Kasus uji cetak barcode dan label jurnal label merah alternatif 1 :
Jika tidak melakukan *ceklis* pada *cekbox***

Nama Kasus Uji	Cetak barcode dan label jurnal label merah
Objek Uji	SKPL-07-03
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan <i>Harap pilih jurnal yang akan dicetak register dan labelnya!</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label merah 2. Aktor tidak melakukan <i>ceklis</i> pada <i>cekbox</i> yang terdapat pada daftar jurnal, kemudian menekan <i>button</i> cetak barcode dan label
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan <i>Harap pilih jurnal yang akan dicetak register dan labelnya!</i>
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan <i>Harap pilih jurnal yang akan dicetak register dan labelnya!</i>
Status Validasi	Valid

Tabel 6.96 Kasus uji cetak barcode dan label jurnal label putih

Nama Kasus Uji	Cetak barcode dan label jurnal label putih
Objek Uji	SKPL-07-04
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan cetak barcode dan label jurnal label putih
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label putih 2. Aktor melakukan <i>ceklis</i> pada <i>cekbox</i> yang terdapat pada daftar jurnal, kemudian menekan <i>button</i> cetak barcode dan label 3. Aktor menekan tombol Ctrl+p
Hasil yang Diharapkan	Sistem dapat melakukan cetak barcode dan label jurnal label putih
Hasil Aktual	Setelah dilakukan pengujian sistem dapat melakukan cetak barcode dan label jurnal label putih
Status Validasi	Valid

Tabel 6.97 Kasus uji cetak barcode dan label jurnal label putih alternatif 1 : Jika tidak melakukan *ceklis* pada *cekbox*

Nama Kasus Uji	Cetak barcode dan label jurnal label putih
Objek Uji	SKPL-07-04
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan <i>Harap pilih jurnal yang akan dicetak register dan labelnya!</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan <i>tab</i> menu label putih 2. Aktor tidak melakukan <i>ceklis</i> pada <i>cekbox</i> yang terdapat pada daftar jurnal, kemudian menekan <i>button</i> cetak barcode dan label
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan <i>Harap pilih jurnal yang akan dicetak register dan labelnya!</i>
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan <i>Harap pilih jurnal yang akan dicetak register dan labelnya!</i>
Status Validasi	Valid

Tabel 6.98 Kasus uji validasi buku

Nama Kasus Uji	Validasi buku
Objek Uji	SKPL-08-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan validasi buku
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memasukkan kode register buku 2. Aktor menekan tombol validasi
Hasil yang Diharapkan	Sistem dapat melakukan validasi buku
Hasil Aktual	Setelah melakukan pengujian sistem dapat melakukan validasi buku
Status Validasi	Valid

Tabel 6.99 Kasus uji validasi buku alternatif 1 : Jika field register kosong

Nama Kasus Uji	Validasi buku
Objek Uji	SKPL-08-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan field tidak boleh kosong

Tabel 6.99 (Lanjutan)

Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor tidak memasukkan data pada <i>field</i> register 2. Aktor menekan tombol validasi
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan <i>field</i> tidak boleh kosong
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan <i>field</i> tidak boleh kosong
Status Validasi	Valid

Tabel 6.100 Kasus uji validasi buku alternatif 2 : jika field register tidak diisi angka

Nama Kasus Uji	Validasi buku
Objek Uji	SKPL-08-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan field harus diisi nomor
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor mengisi huruf pada field register 2. Aktor menekan tombol validasi
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan field harus diisi nomor
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan field harus diisi nomor
Status Validasi	Valid

Tabel 6.101 Kasus uji validasi jurnal

Nama Kasus Uji	Validasi jurnal
Objek Uji	SKPL-08-02
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat melakukan validasi jurnal
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memasukkan kode register jurnal 2. Aktor menekan tombol validasi
Hasil yang Diharapkan	Sistem dapat melakukan validasi jurnal
Hasil Aktual	Setelah melakukan pengujian sistem dapat melakukan validasi jurnal
Status Validasi	Valid

Tabel 6.102 Kasus uji validasi jurnal alternatif 1 : Jika *field* register kosong

Nama Kasus Uji	Validasi jurnal
Objek Uji	SKPL-08-02
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan <i>field</i> tidak boleh kosong ketika <i>field</i> register tidak diisi
Prosedur Uji	1. Aktor tidak memasukkan data pada <i>field</i> register 2. Aktor menekan tombol validasi
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan <i>field</i> tidak boleh kosong ketika <i>field</i> register tidak diisi
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan <i>field</i> tidak boleh kosong ketika <i>field</i> register tidak diisi
Status Validasi	Valid

Tabel 6.103 Kasus uji validasi jurnal alternatif 2 : Jika *field* register tidak diisi angka

Nama Kasus Uji	Validasi jurnal
Objek Uji	SKPL-08-02
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa sistem dapat menampilkan pesan kesalahan <i>field</i> harus diisi nomor
Prosedur Uji	1. Aktor mengisi huruf pada <i>field</i> register 2. Aktor menekan tombol validasi
Hasil yang Diharapkan	Sistem dapat menampilkan pesan kesalahan <i>field</i> harus diisi nomor
Hasil Aktual	Setelah dilakukan pengujian sistem dapat menampilkan pesan kesalahan <i>field</i> harus diisi nomor
Status Validasi	Valid

Tabel 6.104 Kasus uji logout

Nama Kasus Uji	<i>Logout</i>
Objek Uji	SKPL-09-01
Tujuan Pengujian	Pengujian ini dilakukan untuk memastikan bahwa aktor login dapat keluar dari sistem

Tabel 6.104 (Lanjutan)

Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor menekan ikon akun 2. Aktor menekan tombol <i>logout</i>
Hasil yang Diharapkan	Aktor keluar dari sistem dan menampilkan halaman <i>login</i>
Hasil Aktual	Setelah dilakukan pengujian Aktor keluar dari sistem dan menampilkan halaman <i>login</i>
Status Validasi	Valid

6.4 Pengujian Web Service

Pengujian *web service* dilakukan untuk mengetahui apakah dalam implementasinya baik dari sisi server maupun sisi client (yang *me-request*) sudah berjalan seperti yang diharapkan. Pengujian *web service* ini menggunakan strategi pengujian validasi.

1. Kasus Uji GET Biodata

Pada tahap ini dilakukan, pengujian *web service* dengan menggunakan metode blackbox pada fungsi login.

Tabel 6.105 Hasil pengujian *web service* pada fungsi login

Nama Kasus Uji	Kasus Uji <i>GET Biodata</i>
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa <i>web service</i> yang dibuat dapat memenuhi kebutuhan fungsional untuk dapat masuk kedalam sistem dan menyimpan biodata anggota (mahasiswa)
Data Masukan	<ol style="list-style-type: none"> 1. <i>userid</i> : berisi nim mahasiswa login 2. <i>challenge</i> : 5 karakter random alfanumerik 3. <i>passport</i> : md5(challenge.password).'_'.<i>userid</i> 4. <i>appid</i> : kode APP 5. <i>ipaddr</i> : IP Address pengakses
Prosedur Uji	<ol style="list-style-type: none"> 1. Menuliskan <i>service endpoint</i> dari <i>server</i> di <i>browser</i> (https://bais.ub.ac.id/api/login/xmlapi/) 2. Menambahkan data masukan yang telah ditentukan (https://bais.ub.ac.id/api/login/xmlapi/?userid=115090600111037&passport=704cbe4cddb2c6130fa18aa7b5402cc9_115090600111037&challenge=123ab&appid=SIPUS&ipaddr:::1) 3. Menjalankan <i>browser</i>
Hasil yang Diharapkan	Browser dapat menampilkan representasi dari biodata anggota (mahasiswa) dalam format XML.

6.5 Pengujian Performansi

Pengujian performansi dilakukan pada setiap proses jalannya sistem untuk mengecek apakah performa *web service* yang telah dibangun dapat menangani beban kerja yang ada. Pengujian ini dilakukan dengan cara menghitung waktu eksekusi yang dibutuhkan dalam menjalankan proses. Perhitungan pada pengujian ini dilakukan sebanyak 10 kali percobaan dan yang ditampilkan adalah nilai rata-rata waktu akses. **Tabel 6.106** menunjukkan pengujian dalam 10 kali percobaan pada fungsi login.

Tabel 6.106 Hasil pengujian performansi fungsi login

Halaman Pengujian	Rata-rata waktu yang dibutuhkan selama eksekusi (det)
Halaman <i>login</i>	7,07

6.6 Analisis Hasil Pengujian

Pada bagian ini akan dilakukan seluruh analisis hasil dari semua pengujian yang telah dilakukan, analisis bertujuan untuk menganalisis data hasil pengujian hingga menghasilkan kesimpulan-kesimpulan. Analisis ini dilakukan berdasarkan hasil implementasi hingga pengujian yang telah dilakukan terhadap sistem dan mengacu pada dasar teori. Proses analisis yang dilakukan diantaranya analisis hasil dari pengujian *white box* dan analisis hasil dari pengujian *black box*. Analisis dilakukan dengan menarik kesimpulan dari hasil setiap pengujian pada setiap kasus ujinya.

6.6.1 Analisis Hasil Pengujian Unit

Pada **Tabel 6.107** akan ditunjukkan hasil dari pengujian unit dengan metode *white box* pada semua kasus uji yang telah dilakukan.

Tabel 6.107 Analisis Hasil Pengujian *White Box*

Kasus Uji	Total Regions	Total Independent Path	V(G)
Kasus Uji Login	6	6	6
Kasus Uji Tambah Permintaan bahan pustaka	6	6	6
Kasus Uji Cari Buku Label Merah	6	6	6

Dari **Tabel 6.107** ditunjukkan bahwa pada setiap kasus uji yang dilakukan dari 3 fungsional didapatkan nilai *cyclomatic complexity* sebesar 6. Hal ini menunjukkan bahwa pada kasus uji ini memiliki prosedur yang terstruktur dan stabil serta memiliki resiko yang rendah sehingga kerumitan untuk melakukan perbaikan *coding* tergolong rendah.

Dari 3 kasus uji yang dilakukan didapatkan *cyclomatic complexity* sebesar 6, Hal ini menunjukkan bahwa pada kasus uji ini memiliki prosedur yang terstruktur dan stabil serta memiliki resiko yang rendah sehingga kerumitan untuk melakukan perbaikan *coding* tergolong rendah.

6.6.2 Analisis Hasil Pengujian Integrasi

Berdasarkan hasil pengujian integrasi, proses yang melibatkan interaksi antar kelas pada sistem telah terintegrasi dengan baik. Maka dari itu tidak diperlukan adanya perbaikan proses integrasi antar kelas sebab seluruh kasus uji integrasi telah valid.

6.6.3 Analisis Hasil Pengujian Validasi

Proses analisis pengujian validasi mengacu pada kesesuaian kinerja sistem dengan kebutuhan fungsionalitas sistem. Berdasarkan hasil pengujian validasi dapat disimpulkan bahwa implementasi fungsionalitas sistem perpustakaan terintegrasi pada Universitas Brawijaya telah memenuhi kebutuhan yang diuraikan pada tahap analisis kebutuhan sebab hasil dari pengujian validasi ini menghasilkan status valid. Hasil pengujian menunjukkan dari 63 pengujian yang diuji minimal 2 kali menunjukkan bahwa 100% kebutuhan fungsional dari sistem telah valid ketika diuji dengan kasus tertentu.

6.6.4 Analisis Hasil Pengujian *Web Service*

Mengacu pada hasil pengujian *web service*, sistem telah dapat mengembalikan nilai sesuai dengan parameter yang diminta. Namun, banyak faktor yang dapat mempengaruhi bahkan memperburuk jalannya proses integrasi ini, seperti kualitas koneksi internet dan padatnnya aktivitas pengakses server SIAKAD UB.

6.6.5 Analisis Hasil Pengujian Performansi

Dari hasil pengujian performansi, dapat disimpulkan bahwa rata-rata dari 10 kali pengujian waktu yang diperlukan untuk proses eksekusi *web service* selama 7,07 detik. Hal ini menunjukkan kebutuhan non fungsional bahwa perangkat lunak dapat berjalan dengan waktu tidak melebihi 30 detik pada saat login menggunakan akun SIAKAD terpenuhi.

BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan maka dapat diambil kesimpulan sebagai berikut:

1. Aplikasi web perpustakaan Universitas Brawijaya yang dibangun dapat memanfaatkan *web service* SIAKAD UB dalam mengintegrasikan data mahasiswa sebagai anggota perpustakaan Universitas Brawijaya.
2. Aplikasi web perpustakaan Universitas Brawijaya yang dibangun menggunakan pendekatan *evolutionary prototyping* telah sesuai dengan perancangan kebutuhan yang diharapkan pengguna, yaitu dapat menampung permintaan bahan pustaka baru dan kemudian dikelola menjadi bahan pustaka yang siap digunakan. Semua data bahan pustaka dapat dipusatkan pada perpustakaan pusat Universitas Brawijaya sehingga data tidak tersebar lagi.

Implementasi semua fungsi yang terdapat pada perancangan web aplikasi terintegrasi perpustakaan Universitas Brawijaya berjalan dengan baik dan semua kebutuhan terpenuhi. Hasil pengujian yang dilakukan menunjukkan sistem berjalan dengan baik ditunjukkan dengan semua hasil pengujian yang dilakukan menunjukkan status valid.

7.2 Saran

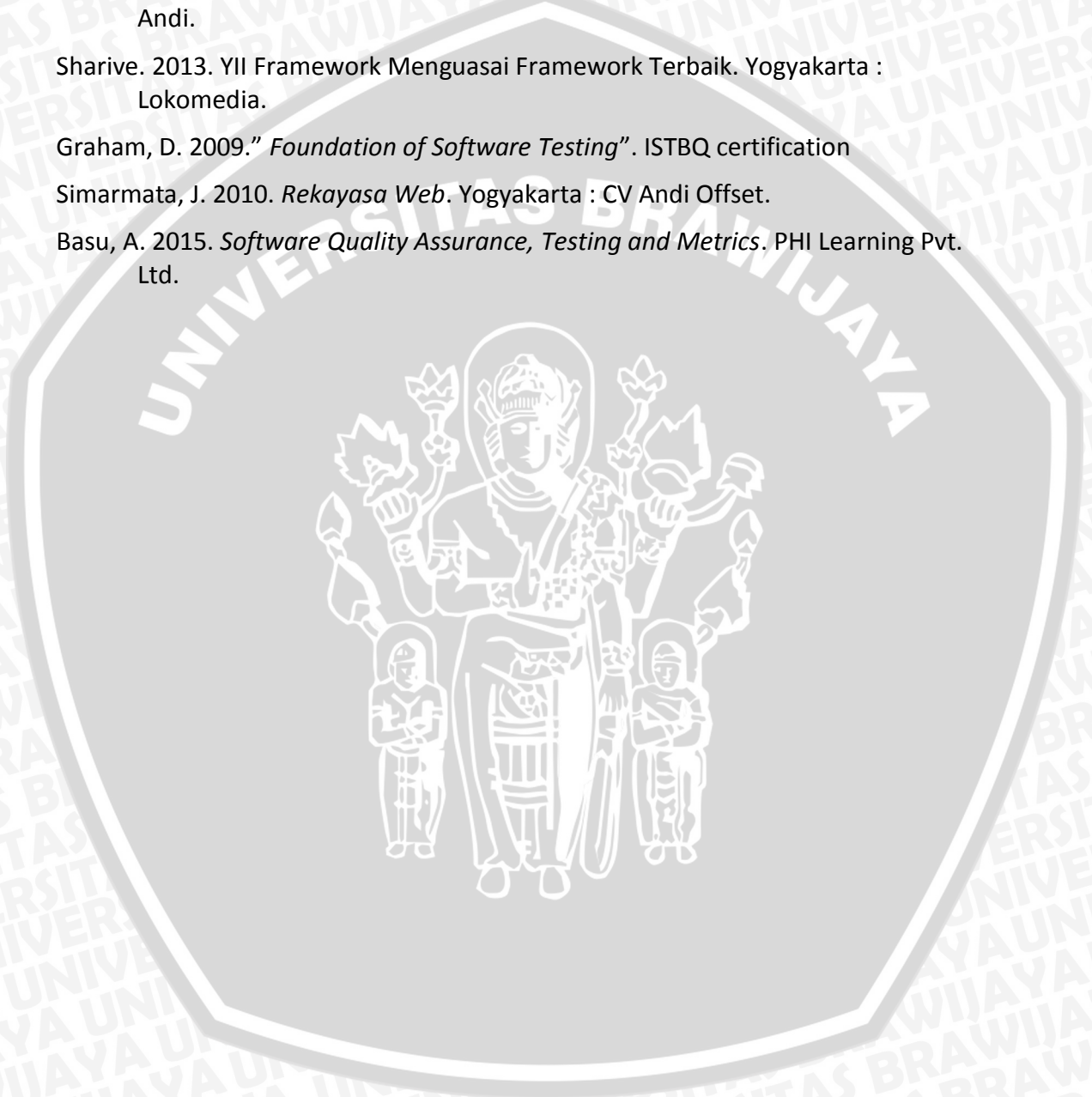
Saran yang dapat diberikan penulis untuk pengembangan Penelitian antara lain.

1. Untuk penelitian lebih lanjut, diharapkan sistem ini dapat mencakup semua bagian perpustakaan seperti, bagian sirkulasi dan keanggotaan menjadi satu aplikasi atau sistem yang terintegrasi.
2. Pengembangan sistem dapat dilakukan dengan menambahkan fitur sms *gateway* untuk memberikan informasi bahan pustaka kepada anggota perpustakaan.

DAFTAR PUSTAKA

- Undang-undang Republik Indonesia nomor 43 tahun 2007 tentang Perpustakaan.
- Sukamto, R. A. dan Shalahuddin, M. 2013. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung : Informatika.
- Pressman, R.S. 2012. *Rekayasa Perangkat Lunak : Pendekatan Praktisi*. Diterjemahkan dari Bahasa Inggris oleh A. Nugroho (et.al). 2012. Yogyakarta: Andi.
- Zhang, X., Lv, S., Xu, M. & Mu, W. 2010. "Applying evolutionary prototyping model for eliciting system requirement of meat traceability at agribusiness level". ScienceDirect. Tersedia melalui Universitas Brawijaya < <http://www.sciencedirect.com> > [Diakses 21 juli 2015].
- Zhang, X., Fu, Z., Cai, W., Tian, D. & Zhang, J. 2009. "Applying evolutionary prototyping model in developing FIDSS: An intelligent decision support system for fish disease/health management". ScienceDirect. Tersedia melalui Universitas Brawijaya < <http://www.sciencedirect.com> > [Diakses 21 juli 2015].
- Basuki, S. 1991. Pengantar Ilmu Perpustakaan. Jakarta: Gramedia Pustaka Utama.
- Perpustakaan Universitas Brawijaya, 2014. *Manual Mutu ISO 9001 : 2008 Perpustakaan Universitas Brawijaya*. Malang : Perpustakaan Universitas Brawijaya.
- Sommerville, I. 2003. *Software Engineering : Rekayasa Perangkat Lunak, Edisi Enam, Jilid Satu*, (Alih Bahasa, Dra. Yuhilza Human M.Eng). Jakarta: Erlangga.
- Pressman, R. S. 2010. *Software Engineering: A Practitioner's Approach, 7th Edition*. New York : McGraw-Hill, Inc.
- McConnell, S.. 1996. "Rapid Development". Canada : Microsoft Press.
- Gunther, L. & Moeller, T. 2003. ".Net- A Complete Development Cycle". Boston : Pearson Education, Inc.
- Nolan, G., Truxall, D. & Cinar, O. 2013. *Android Best Practices*. Addison-Wesley.
- Nugroho, A. 2010. *Rekayasa Perangkat Lunak Berorientasi Objek dengan Metode USDP (Unified Software Development Proses)*. Yogyakarta : Penerbit Andi.
- Lethbridge, T. C. & Laganier, R. 2002. *Object-Oriented Software Engineering: Practical Software Development using UML and Java*. England : McGraw-Hill International (UK) Limited.
- Fazri, A. 2013. *Perancangan Perangkat Lunak Berbasis Wap untuk Manajemen Percetakan yang Terintegrasi dengan SMS Gateway pada Percetakan Mentari*. Palembang : Universitas Binadarma.

- Sukamto, R. A. dan Shalahuddin, M. 2011. "*Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*". Bandung: Modula.
- Kurniawati, S. 2014. *Rancang Bangun Aplikasi Tribinacita Berbasis Android*. Malang : Universitas Brawijaya
- Sutanta, E. 2011. *Basis Data dalam Tinjauan Konseptual*. Yogyakarta : Penerbit Andi.
- Sharive. 2013. *YII Framework Menguasai Framework Terbaik*. Yogyakarta : Lokomedia.
- Graham, D. 2009. "*Foundation of Software Testing*". ISTBQ certification
- Simarmata, J. 2010. *Rekayasa Web*. Yogyakarta : CV Andi Offset.
- Basu, A. 2015. *Software Quality Assurance, Testing and Metrics*. PHI Learning Pvt. Ltd.



LAMPIRAN A PERTANYAAN WAWANCARA

Lampiran ini berisi pertanyaan-pertanyaan yang diajukan ketika melakukan wawancara secara terstruktur. Wawancara dilakukan dengan kepada staf pengolahan dan staf IT perpustakaan Universitas Brawijaya. Berikut adalah beberapa pertanyaan yang penulis utarakan kepada narasumber.

1. Bagaimana struktur organisasi di Perpustakaan UB?
2. Apa hubungan perpustakaan pusat dengan ruang baca di setiap fakultas?
3. Apakah ada keterkaitan antara perpustakaan UB dengan PPTI UB?
4. Bahan pustaka apa saja yang dimiliki oleh perpustakaan UB? Dan berapa banyak jumlah bahan pustaka yang dimiliki?
5. Apakah bahan pustaka yang menjadi koleksi perpustakaan sudah sesuai dengan kebutuhan mahasiswa?
6. Kendala apa saja yang terjadi di perpustakaan UB?
7. Bagaimana penanganan kendala-kendala tersebut?
8. Bagaimana proses pengadaan bahan pustaka selama ini?
9. Bagaimana proses pengolahan bahan pustaka selama ini?
10. Staf apa saja yang terlibat dalam pengolahan bahan pustaka?
11. Solusi seperti apa yang diharapkan untuk mengatasi kendala-kendala tersebut?



LAMPIRAN B PROSES ITERASI *PROTOTYPING*

Proses pembangunan perangkat lunak yang dilakukan pada skripsi ini menggunakan model *evolutionary prototyping* sehingga analisis kebutuhan dan perancangan dilakukan lebih dari satu kali iterasi. Iterasi yang akan dilakukan pada pembuatan sistem ini sebanyak 3 kali iterasi, yaitu mulai dari iterasi ke-0 sampai iterasi ke-3.

B.1 Analisis Kebutuhan Awal

Tahap perancangan analisis ini dilakukan mulai dari wawancara secara langsung kepada pihak perpustakaan Universitas Brawijaya, kemudian dilanjutkan dengan melakukan analisis terhadap sistem yang sudah ada sebelumnya dan kemudian direpresentasikan dalam bentuk *use case*. Tujuan dari tahap ini adalah untuk mengumpulkan informasi tentang entitas yang terlibat dalam sistem dan kebutuhan-kebutuhan sistem yang menggambarkan bagaimana sistem Perpustakaan Terintegrasi pada Universitas Brawijaya didefinisikan.

B.1.1 Identifikasi Aktor Awal

Pada tahap ini dilakukan proses identifikasi terhadap aktor-aktor yang berinteraksi dengan perangkat lunak yang dibangun. **Tabel B.1** memperlihatkan aktor-aktor yang terlibat serta penjelasan peran masing-masing aktor dalam sistem pada awal perancangan atau iterasi ke-0.

Tabel B.1 Identifikasi aktor interaksi ke-0

Aktor	Deskripsi
User Tamu	User Tamu merupakan user yang dapat melihat halaman awal sistem dan melakukan login sistem.
<i>Administrator</i>	<i>Administrator</i> berperan sebagai pengguna dari aplikasi yang memiliki hak akses semua fitur yang terdapat pada aplikasi.
Staf	Staf berperan sebagai pengguna aplikasi yang memiliki hak atas fitur permintaan dan pengolahan bahan pustaka.
Anggota Perpustakaan	Anggota perpustakaan merupakan mahasiswa berperan sebagai pengguna aplikasi yang hanya dapat mengakses fitur permintaan bahan pustaka.

B.1.2 Spesifikasi kebutuhan fungsional Awal

Spesifikasi kebutuhan yang disediakan pada sistem ini merupakan penjabaran dari kebutuhan perangkat lunak secara umum. **Tabel B.2** menjelaskan spesifikasi kebutuhan fungsional awal atau iterasi ke-0, sedangkan **Tabel B.3** menjelaskan spesifikasi kebutuhan non fungsional dari sistem awal atau iterasi ke-0.

Tabel B.2 Spesifikasi kebutuhan fungsional iterasi ke-0

Kode SKPL	Deskripsi	Use Case	Aktor	Prioritas
SKPL-01-01	Perangkat lunak harus mampu menyediakan fasilitas autentifikasi menggunakan akun SIAKAD (Sistem Informasi Akademik) dan menggunakan <i>username</i> dan <i>password</i> yang sudah terdaftar.	Login	User Tamu	Tinggi
SKPL-02-01	Perangkat lunak mampu menyediakan fasilitas untuk melihat biodata <i>user</i> yang melakukan login.	Lihat detail biodata	Staf, anggota, dan <i>administrator</i>	Rendah
SKPL-02-02	Perangkat lunak mampu menyediakan fasilitas ubah biodata agar <i>user</i> yang melakukan login dapat merubah biodata jika terjadi kesalahan.	Ubah biodata	Staf, anggota, dan <i>administrator</i>	Sedang
SKPL-03-01	Perangkat lunak mampu menyediakan fasilitas untuk menambah staf baru.	Tambah staf	<i>administrator</i>	Tinggi
SKPL-03-02	Perangkat lunak dapat menyediakan fasilitas ubah data staf ruang baca sehingga data dapat dirubah jika terjadi kesalahan.	Ubah staf	<i>administrator</i>	Sedang
SKPL-03-03	Perangkat lunak dapat menyediakan fasilitas untuk melihaf daftar staf yang terdaftar.	Daftar staf	<i>administrator</i>	Sedang
SKPL-04-01	Perangkat lunak dapat menyediakan fasilitas tambah permintaan bahan pustaka buku.	Tambah permintaan buku	Anggota, staf dan <i>administrator</i>	Tinggi

Tabel B.2 (Lanjutan)

Kode SKPL	Deskripsi	Use Case	Aktor	Prioritas
SKPL-04-02	Perangkat lunak dapat menyediakan fasilitas lihat daftar permintaan bahan pustaka buku yang diajukan.	Lihat daftar permintaan buku	Staf, anggota dan <i>administrator</i>	Sedang
SKPL-04-03	Perangkat lunak dapat menyediakan fasilitas lihat daftar permintaan bahan pustaka buku yang sudah disetujui.	Lihat daftar permintaan buku disetujui	Staf dan <i>administrator</i>	Sedang
SKPL-04-04	Perangkat lunak dapat menyediakan fasilitas lihat daftar permintaan bahan pustaka buku yang belum disetujui.	Lihat daftar permintaan buku belum disetujui	Staf dan <i>administrator</i>	Sedang
SKPL-04-05	Perangkat lunak dapat menyediakan fasilitas yang dapat melihat detail data permintaan bahan pustaka buku.	Lihat detail permintaan buku	Staf dan <i>administrator</i>	Rendah
SKPL-04-6	Perangkat lunak dapat menyediakan fasilitas yang dapat mengubah data permintaan bahan pustaka buku.	Ubah permintaan buku	Staf dan <i>administrator</i>	Sedang
SKPL-04-7	Perangkat lunak dapat menyediakan fasilitas yang dapat menghapus data permintaan bahan pustaka buku.	Hapus permintaan buku	Staf dan <i>administrator</i>	Rendah
SKPL-04-8	Perangkat lunak dapat menyediakan fasilitas tambah permintaan bahan pustaka jurnal.	Tambah permintaan jurnal	Anggota, staf dan <i>administrator</i>	Tinggi
SKPL-04-9	Perangkat lunak dapat menyediakan fasilitas lihat daftar permintaan bahan pustaka jurnal yang diajukan.	Lihat daftar permintaan jurnal	Anggota, staf dan <i>administrator</i>	Sedang

Tabel B.2 (Lanjutan)

Kode SKPL	Deskripsi	Use Case	Aktor	Prioritas
SKPL-04-10	Perangkat lunak dapat menyediakan fasilitas lihat daftar permintaan bahan pustaka jurnal.	Lihat daftar permintaan jurnal disetujui	Staf dan <i>administrator</i>	Sedang
SKPL-04-11	Perangkat lunak dapat menyediakan fasilitas lihat daftar permintaan bahan pustaka jurnal.	Lihat daftar permintaan jurnal belum disetujui	Staf dan <i>administrator</i>	Sedang
SKPL-04-12	Perangkat lunak dapat menyediakan fasilitas yang dapat melihat detail data permintaan bahan pustaka jurnal.	Lihat detail permintaan jurnal	Staf dan <i>administrator</i>	Rendah
SKPL-04-13	Perangkat lunak dapat menyediakan fasilitas yang dapat mengubah data permintaan bahan pustaka jurnal.	Ubah permintaan jurnal	Staf dan <i>administrator</i>	Sedang
SKPL-04-14	Perangkat lunak dapat menyediakan fasilitas yang dapat menghapus data permintaan bahan pustaka jurnal.	Hapus permintaan jurnal	Staf dan <i>administrator</i>	Rendah
SKPL-05-01	Perangkat lunak dapat menyediakan fasilitas untuk melakukan tambah bahan pustakan buku baru / buku label merah.	Tambah buku baru	Staf dan <i>administrator</i>	Tinggi
SKPL-05-02	Perangkat lunak dapat menyediakan fasilitas untuk melihat daftar permintaan bahan pustaka buku yang berlabel merah.	Lihan daftar buku label merah	Staf dan <i>administrator</i>	Sedang
SKPL-05-03	Perangkat lunak dapat menyediakan fasilitas yang dapat melihat detail bahan pustaka buku yang berlabel merah.	Lihat detail buku label merah	Staf dan <i>administrator</i>	Rendah

Tabel B.2 (Lanjutan)

Kode SKPL	Deskripsi	Use Case	Aktor	Prioritas
SKPL-05-04	Perangkat lunak dapat menyediakan fasilitas yang dapat mengubah data bahan pustaka buku berlabel merah.	Ubah buku label merah	Staf dan <i>administrator</i>	Sedang
SKPL-05-05	Perangkat lunak dapat menyediakan fasilitas untuk melakukan tambah bahan pustakan jurnal baru / jurnal label merah.	Tambah jurnal baru	Staf dan <i>administrator</i>	Tinggi
SKPL-05-06	Perangkat lunak dapat menyediakan fasilitas untuk melihat daftar permintaan bahan pustaka jurnal yang berlabel merah.	Lihan daftar jurnal label merah	Staf dan <i>administrator</i>	Sedang
SKPL-05-07	Perangkat lunak dapat menyediakan fasilitas yang dapat melihat detail bahan pustaka jurnal yang berlabel merah.	Lihat detail jurnal label merah	Staf dan <i>administrator</i>	Rendah
SKPL-05-08	Perangkat lunak dapat menyediakan fasilitas yang dapat mengubah data bahan pustaka jurnal berlabel merah.	Ubah jurnal label merah	Staf dan <i>administrator</i>	Sedang
SKPL-06-01	Perangkat lunak dapat menyediakan fasilitas untuk melakukan tambah bahan pustakan buku yang sudah pernah ada sebelumnya / buku label putih.	Tambah buku label putih	Staf dan <i>administrator</i>	Tinggi
SKPL-06-02	Perangkat lunak dapat menyediakan fasilitas untuk melihat daftar permintaan bahan pustaka buku yang berlabel putih.	Lihan daftar buku label putih	Staf dan <i>administrator</i>	Sedang

Tabel B.2 (Lanjutan)

Kode SKPL	Deskripsi	Use Case	Aktor	Prioritas
SKPL-06-03	Perangkat lunak dapat menyediakan fasilitas yang dapat mengubah data bahan pustaka jurnal berlabel merah.	Ubah buku label putih	Staf dan <i>administrator</i>	Sedang
SKPL-06-04	Perangkat lunak dapat menyediakan fasilitas yang dapat melihat detail bahan pustaka buku yang berlabel putih.	Lihat detail buku label putih	Staf dan <i>administrator</i>	Rendah
SKPL-06-05	Perangkat lunak dapat menyediakan fasilitas untuk melakukan tambah bahan pustakan jurnal yang sudah pernah ada sebelumnya / jurnal label putih..	Tambah jurnal label putih	Staf dan <i>administrator</i>	Tinggi
SKPL-06-06	Perangkat lunak dapat menyediakan fasilitas untuk melihat daftar permintaan bahan pustaka jurnal yang berlabel putih.	Lihan daftar jurnal label putih	Staf dan <i>administrator</i>	Sedang
SKPL-06-07	Perangkat lunak dapat menyediakan fasilitas yang dapat mengubah data bahan pustaka jurnal berlabel putih.	Ubah jurnal label putih	Staf dan <i>administrator</i>	Sedang
SKPL-06-08	Perangkat lunak dapat menyediakan fasilitas yang dapat melihat detail bahan pustaka jurnal yang berlabel putih.	Lihat detail jurnal label putih	Staf dan <i>administrator</i>	Rendah
SKPL-07-01	Perangkat lunak dapat menyediakan fasilitas yang dapat mencetak barcode dan label bahan pustaka buku label merah.	Cetak barcode dan label buku label merah	Staf dan <i>administrator</i>	Tinggi

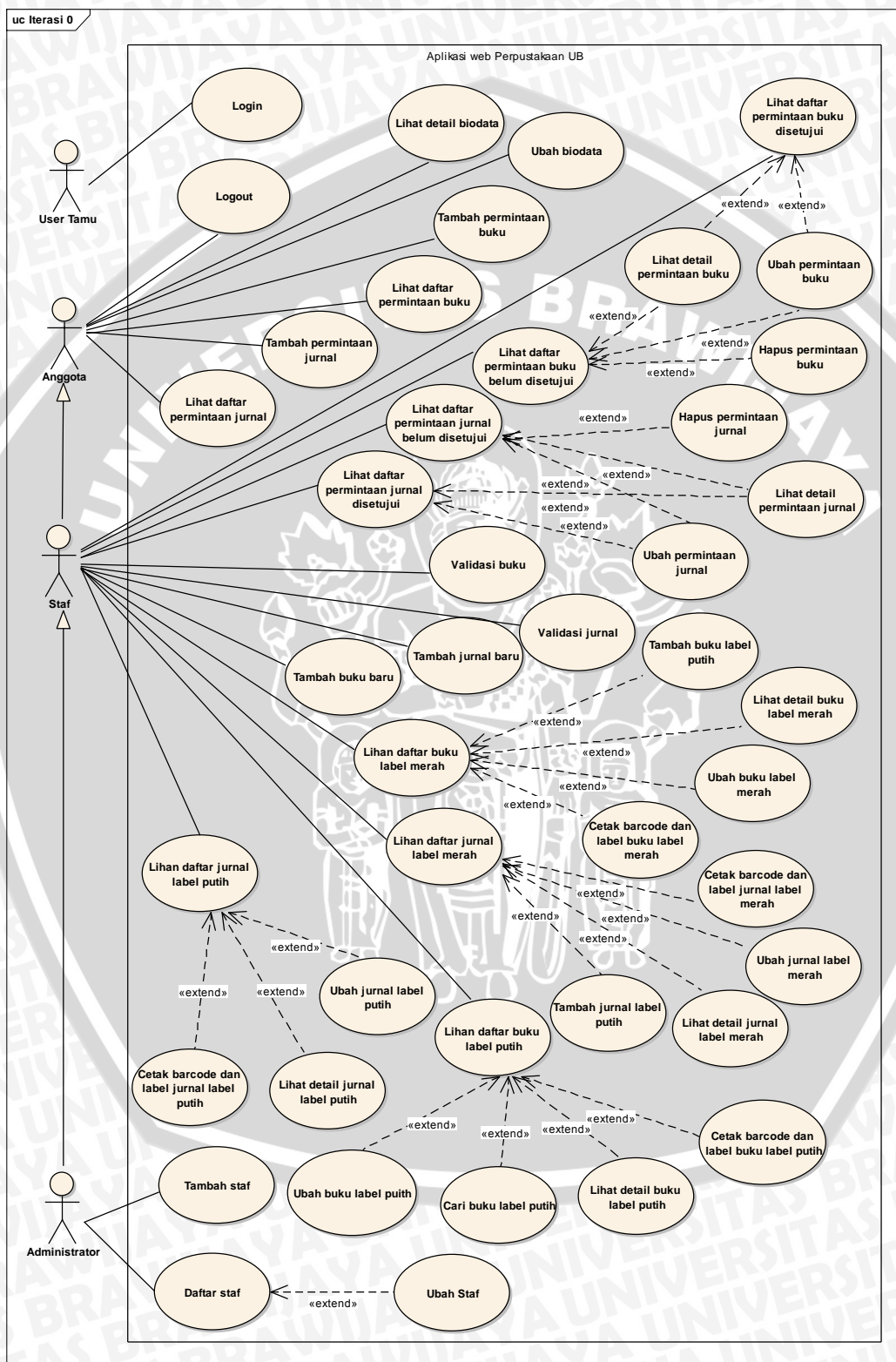
Tabel B.2 (Lanjutan)

Kode SKPL	Deskripsi	Use Case	Aktor	Prioritas
SKPL-07-02	Perangkat lunak dapat menyediakan fasilitas yang dapat mencetak barcode dan label bahan pustaka buku label putih.	Cetak barcode dan label buku label putih	Staf dan <i>administrator</i>	Tinggi
SKPL-07-03	Perangkat lunak dapat menyediakan fasilitas yang dapat mencetak barcode dan label bahan pustaka jurnal label merah.	Cetak barcode dan label jurnal label merah	Staf dan <i>administrator</i>	Tinggi
SKPL-07-04	Perangkat lunak dapat menyediakan fasilitas yang dapat mencetak barcode dan label bahan pustaka jurnal label putih.	Cetak barcode dan label jurnal label putih	Staf dan <i>administrator</i>	Tinggi
SKPL-08-01	Perangkat lunak dapat menyediakan fasilitas yang dapat melakukan validasi bahan pustaka buku.	Validasi buku	Staf dan <i>administrator</i>	Tinggi
SKPL-08-02	Perangkat lunak dapat menyediakan fasilitas yang dapat melakukan validasi bahan pustaka jurnal.	Validasi jurnal	Staf dan <i>administrator</i>	Tinggi
SKPL-09-01	Perangkat lunak dapat menyediakan fasilitas <i>logout</i> setelah <i>user</i> masuk (<i>login</i>) kedalam sistem.	<i>Logout</i>	Anggota dan <i>administrator</i>	Rendah

Tabel B.3 Spesifikasi kebutuhan non-fungsional iterasi ke-0

Nomor SKPL	Parameter	Deskripsi Kebutuhan
SKPL_22	<i>Response Time</i>	Perangkat lunak dapat berjalan dengan waktu tidak melebihi 30 detik pada saat login menggunakan akun SIAKAD.

Model diagram *use case* pada awal perancangan akan digambarkan pada **Gambar B.1.** *Use case* ini dirancang berdasarkan spesifikasi kebutuhan fungsional.



Gambar B.1 Diagram *use case* diagram iterasi ke-0

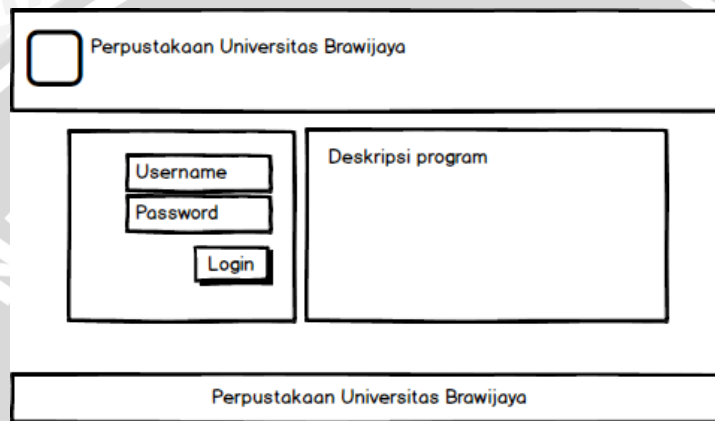


B.1.3 Perancangan Prototipe Awal

Pada bagian ini akan dijelaskan tentang perancangan prototipe perangkat lunak pada tahap awal. Berikut merupakan rancangan prototipe dari perangkat lunak tahap awal.

1. Rancangan Prototipe Halaman *Login*

Halaman login merupakan halaman awal dari sistem yang berisi form login dan deskripsi sistem. Berikut ini merupakan rancangan prototipe halaman login pada **Gambar B.2**.

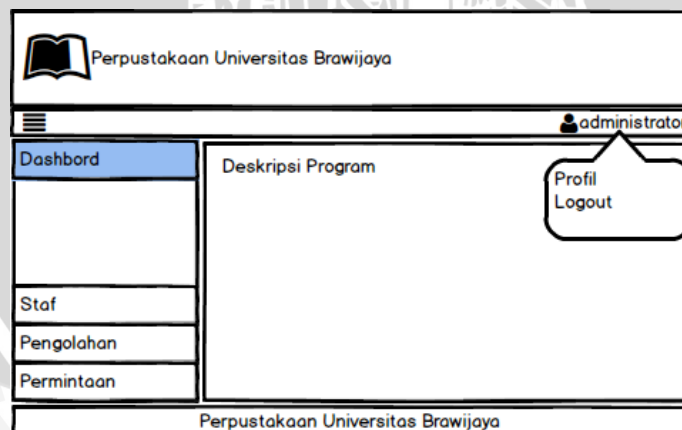


The image shows a wireframe for a login page. At the top left, there is a logo consisting of a square with a circle inside, followed by the text 'Perpustakaan Universitas Brawijaya'. Below this, there is a form with two columns. The left column contains three input fields: 'Username', 'Password', and a 'Login' button. The right column is a large empty box labeled 'Deskripsi program'. At the bottom of the page, there is a footer bar with the text 'Perpustakaan Universitas Brawijaya'.

Gambar B.2 Rancangan prototipe halaman *login*

2. Rancangan Prototipe Halaman Utama

Halaman ini merupakan halaman utama dari sistem yang berisi menu dan deskripsi sistem. Berikut ini merupakan rancangan prototipe halaman utama pada **Gambar B.3**.



The image shows a wireframe for a main dashboard page. At the top left, there is a logo with a book icon and the text 'Perpustakaan Universitas Brawijaya'. To the right of the logo, there is a user profile icon and the text 'administrator'. Below this, there is a navigation menu on the left side with four items: 'Dashbord', 'Staf', 'Pengolahan', and 'Permintaan'. The 'Dashbord' item is highlighted. The main content area is a large empty box labeled 'Deskripsi Program'. In the top right corner of this area, there is a dropdown menu with two options: 'Profil' and 'Logout'. At the bottom of the page, there is a footer bar with the text 'Perpustakaan Universitas Brawijaya'.

Gambar B.3 Rancangan prototipe halaman utama

4. Rancangan Prototipe Halaman Staf

Halaman ini merupakan halaman utama dari sistem yang berisi menu *dropdown* dan form tambah staf dan daftar staf. Berikut ini merupakan rancangan prototipe halaman tambah staf pada **Gambar A.4** dan daftar staf **Gambar A.5**

Perpustakaan Universitas Brawijaya

administrator

Dashbord

Staf

-Tambah Staf

-Daftar Staf

Kode Staf

Username

Password

Nama

Bagian

Pengolahan

Submit

Perpustakaan Universitas Brawijaya

Gambar B.4 Rancangan prototipe halaman tambah staf

Perpustakaan Universitas Brawijaya

administrator

Dashbord

Staf

-Tambah Staf

-Daftar Staf

Kode Staf	Username	Nama	Lokasi	Bagian	aksi
OP00	staf00	Fasial	Perpus Pusat	Pengolahan	<input type="checkbox"/>
OP01	staf01	Dinan	Ruang Baca PTIHK	Riang Baca	<input type="checkbox"/>

Perpustakaan Universitas Brawijaya

Gambar B.5 Rancangan prototipe halaman daftar staf

5. Rancangan Prototipe Biodata Anggota

Halaman ini merupakan halaman biodata anggota perpustakaan. Berikut ini rancangan prototipe halaman biodata anggota pada **Gambar B.6**.

Perpustakaan Universitas Brawijaya

Faisal

Dashbord

Permintaan

-Individu

BIODATA

NIM : 115090600111037

Nama : Faisal Prampudinantaka

Jenjang : Sarjana S1

Program Studi : Informatika

Fakultas : FILKOM

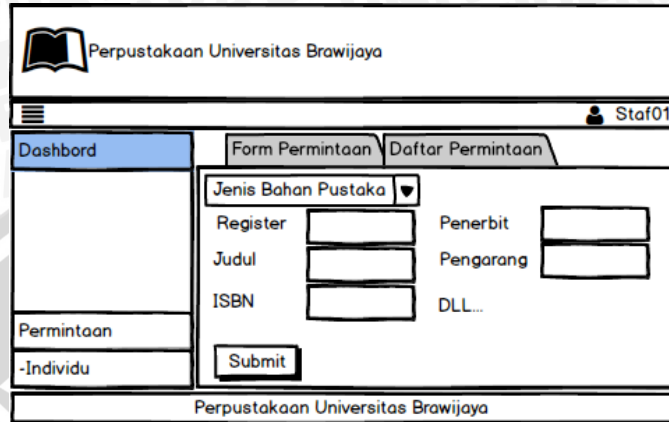
DLL...

Perpustakaan Universitas Brawijaya

Gambar B.6 Rancangan prototipe halaman biodata anggota

5. Rancangan Prototipe Halaman Permintaan Bahan Pustaka

Halaman ini merupakan halaman permintaan bahan pustaka yang meliputi buku dan jurnal. Berikut ini merupakan rancangan prototipe halaman permintaan bahan pustaka pada **Gambar B.7**.

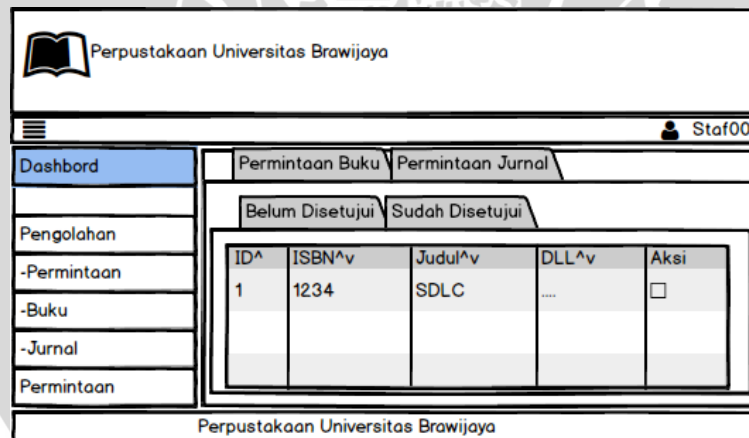


The image shows a web form for requesting books. At the top, it says 'Perpustakaan Universitas Brawijaya' with a book icon. Below that is a navigation bar with 'Dashbord', 'Form Permintaan', and 'Daftar Permintaan'. The 'Form Permintaan' section has a dropdown menu for 'Jenis Bahan Pustaka', and input fields for 'Register', 'Penerbit', 'Judul', 'Pengarang', 'ISBN', and 'DLL...'. There is a 'Submit' button at the bottom. The footer says 'Perpustakaan Universitas Brawijaya'.

Gambar B.7 Rancangan prototipe halaman permintaan individu

6. Rancangan Prototipe Halaman Pengolahan Permintaan Bahan Pustaka

Halaman ini merupakan halaman pengolahan permintaan bahan pustaka, halaman ini berisi menu *dropdown*, *tab* menu, dan daftar permintaan bahan pustaka. Berikut ini merupakan rancangan prototipe halaman pengolahan permintaan bahan pustaka pada **Gambar B.8**.



The image shows a web page for processing book requests. At the top, it says 'Perpustakaan Universitas Brawijaya' with a book icon. Below that is a navigation bar with 'Dashbord', 'Permintaan Buku', and 'Permintaan Jurnal'. The 'Permintaan Buku' section has two tabs: 'Belum Disetujui' and 'Sudah Disetujui'. Below the tabs is a table with columns: ID, ISBN, Judul, DLL, and Aksi. The table contains one row with the following data:

ID	ISBN	Judul	DLL	Aksi
1	1234	SDLC	<input type="checkbox"/>

The footer says 'Perpustakaan Universitas Brawijaya'.

Gambar B.8 Rancangan prototipe halaman pengolahan permintaan

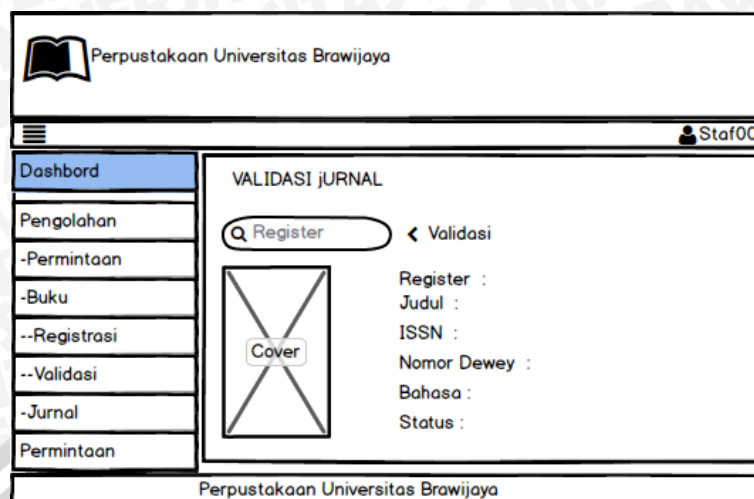
7. Rancangan Prototipe Halaman Pengolahan Bahan Pustaka

Halaman ini merupakan halaman pengolahan bahan pustaka yang meliputi buku dan jurnal. Berikut ini merupakan rancangan prototipe halaman registrasi buku pada **Gambar B.9**. Rancangan prototipe halaman registrasi jurnal pada **Gambar B.10**. Rancangan prototipe halaman validasi buku dan jurnal pada **Gambar B.11** dan **Gambar B.12**.

Gambar B.9 Rancangan prototype halaman registrasi buku

Gambar B.10 Rancangan prototype halaman registrasi jurnal

Gambar B.11 Rancangan prototype halaman validasi buku



Gambar B.12 Rancangan prototipe halaman validasi jurnal

B.2 Proses Iterasi ke-1

Pada iterasi ke-1 terjadi perubahan pada kebutuhan perangkat lunak setelah dilakukan evaluasi pada prototyping tahap pertama. Hasil dari evaluasi tersebut adalah sebagai berikut.

1. Perubahan aktor *administrator* menjadi staf IT
2. Perubahan aktor staf menjadi staf ruang baca dan staf pengolahan serta hak aksesnya.
3. Perangkat lunak dapat melakukan permintaan bahan pustaka secara kolektif.
4. Perangkat lunak dapat menambah bahan pustaka secara kolektif
5. Perangkat lunak dapat menyediakan fasilitas unduh daftar permintaan bahan pustaka.

Hasil evaluasi pada tahap awal mengakibatkan perubahan dan penambahan aktor dan kebutuhan perangkat lunak. Dari kebutuhan perangkat lunak akan diperjelas keladam spesifikasi kebutuhan perangkat lunak. Berikut ini penambahan aktor dan spesifikasi kebutuhan perangkat lunak pada iterasi ke-1.

Tabel B.4 Penambahan aktor interasi ke-1

Aktor	Deskripsi
Staf IT	Staf IT berperan sebagai pengguna dari aplikasi yang memiliki hak akses semua fitur yang terdapat pada aplikasi.
Staf Pengolahan	Staf Pengolahan berperan sebagai pengguna aplikasi yang memiliki hak atas fitur pengolahan permintaan bahan pustakan dan pengolahan bahan pustaka.
Staf Ruang Baca	Staf Ruang Baca berperan sebagai pengguna aplikasi yang memiliki hak atas fitur pengolahan bahan pustaka.

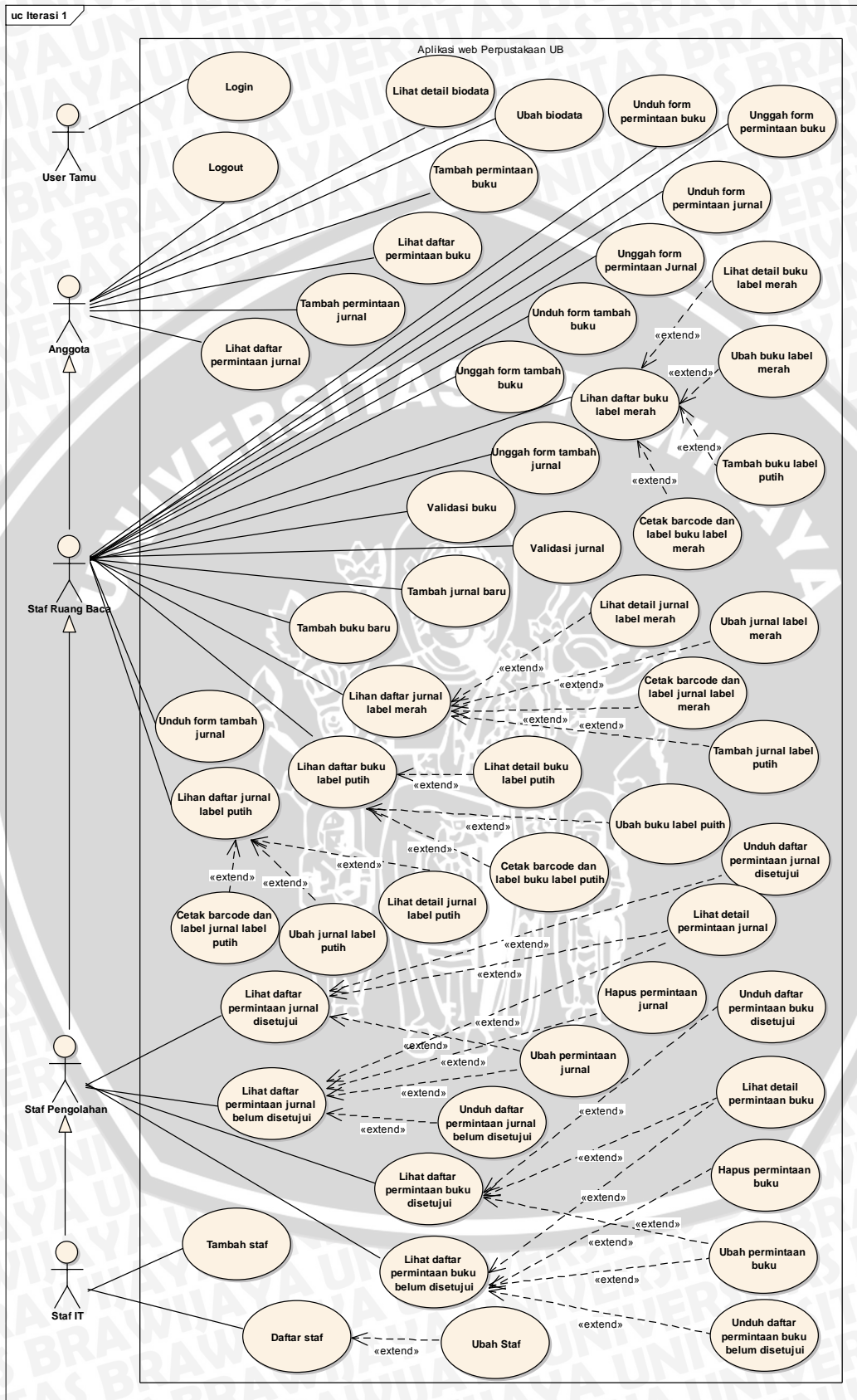
Tabel B.5 Spesifikasi kebutuhan fungsional

Fungsi	Deskripsi	Aktor	Prioritas
Unduh form permintaan buku	Perangkat lunak dapat menyediakan fasilitas unduh form permintaan bahan pustaka buku.	Staf IT, staf pengolahan dan staf ruang baca	Rendah
Unduh form permintaan jurnal	Perangkat lunak dapat menyediakan fasilitas unduh form permintaan bahan pustaka jurnal.	Staf IT, staf pengolahan dan staf ruang baca	Rendah
Unggah form permintaan buku	Perangkat lunak dapat menyediakan fasilitas unggah form untuk menambah permintaan bahan pustaka buku secara kolektif.	Staf IT, staf pengolahan dan staf ruang baca	Rendah
Unggah form permintaan buku	Perangkat lunak dapat menyediakan fasilitas unggah form untuk menambah permintaan bahan pustaka jurnal secara kolektif.	Staf IT, staf pengolahan dan staf ruang baca	Rendah
Unduh form tambah buku	Perangkat lunak dapat menyediakan fasilitas unduh form tambah koleksi bahan pustaka buku.	Staf IT dan staf pengolahan	Rendah
Unggah form tambah buku	Perangkat lunak dapat menyediakan fasilitas unggah form tambah buku untuk menambah bahan pustaka buku secara kolektif.	Staf IT dan staf pengolahan	Rendah
Unduh form tambah jurnal	Perangkat lunak dapat menyediakan fasilitas unduh form tambah koleksi bahan pustaka jurnal.	Staf IT dan staf pengolahan	Rendah
Unggah form tambah jurnal	Perangkat lunak dapat menyediakan fasilitas unggah form tambah jurnal untuk menambah bahan pustaka jurnal secara kolektif.	Staf IT dan staf pengolahan	Rendah

Tabel B.5 (Lanjutan)

Fungsi	Deskripsi	Aktor	Prioritas
Unduh daftar permintaan buku disetujui	Perangkat lunak dapat menyediakan fasilitas yang dapat menunduh data permintaan bahan pustaka buku yang sudah disetujui.	Staf IT dan staf pengolahan	Rendah
Unduh daftar permintaan buku belum disetujui	Perangkat lunak dapat menyediakan fasilitas yang dapat menunduh data permintaan bahan pustaka buku yang belum disetujui.	Staf IT dan staf pengolahan	Rendah
Unduh daftar permintaan jurnal disetujui	Perangkat lunak dapat menyediakan fasilitas yang dapat menunduh data permintaan bahan pustaka jurnal yang sudah disetujui.	Staf IT dan staf pengolahan	Rendah
Unduh daftar permintaan jurnal belum disetujui	Perangkat lunak dapat menyediakan fasilitas yang dapat menunduh data permintaan bahan pustaka jurnal yang belum disetujui.	Staf IT dan staf pengolahan	Rendah

Berikut model use case pada iterasi ke-1 setelah dilakukan evaluasi pada tahap awal.



Gambar B.13 Diagram use case diagram iterasi ke-1

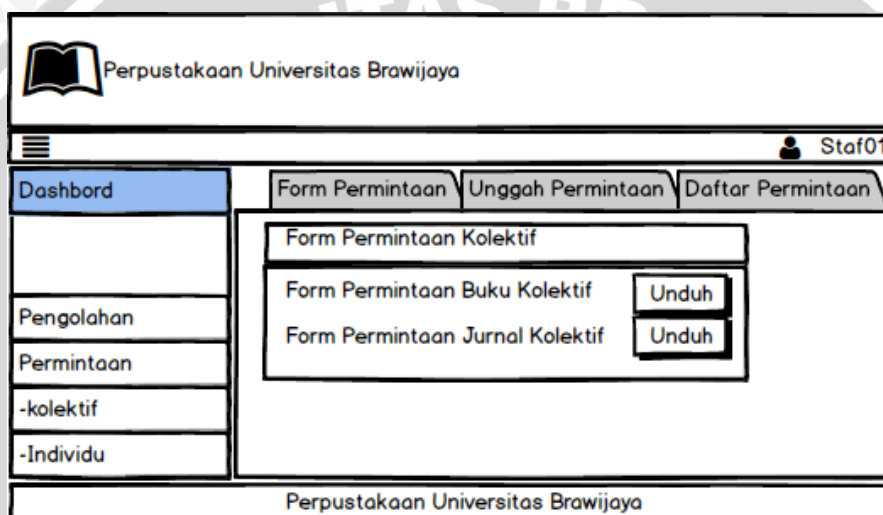


B.2.1 Perancangan Prototipe Iterasi ke-1

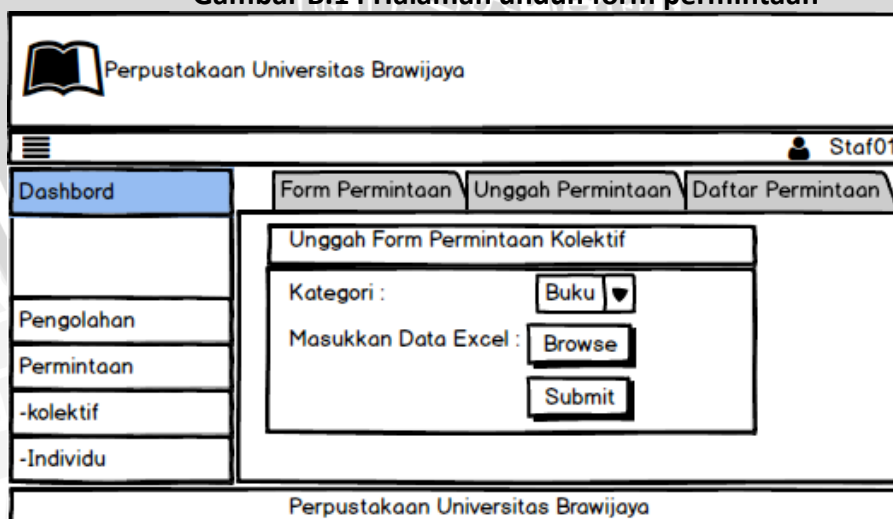
Pada bagian ini akan dijelaskan tentang perancangan prototipe perangkat lunak setelah dilakukan evaluasi pada tahap awal. Berikut merupakan perubahan rancangan prototipe setelah dilakukan evaluasi tahap awal.

1. Rancangan Prototipe Halaman Permintaan Bahan Pustaka Kolektif

Halaman ini merupakan halaman permintaan bahan pustaka buku dan jurnal yang dilakukan secara kolektif. Proses permintaan kolektif dilakukan dengan cara mengunduh dan mengunggah form permintaan. Rancangan prototipe halaman unduh form permintaan dapat dilihat pada **Gambar B.14**. Rancangan prototipe halaman unggah form permintaan dapat dilihat pada **Gambar B.15**.



Gambar B.14 Halaman unduh form permintaan



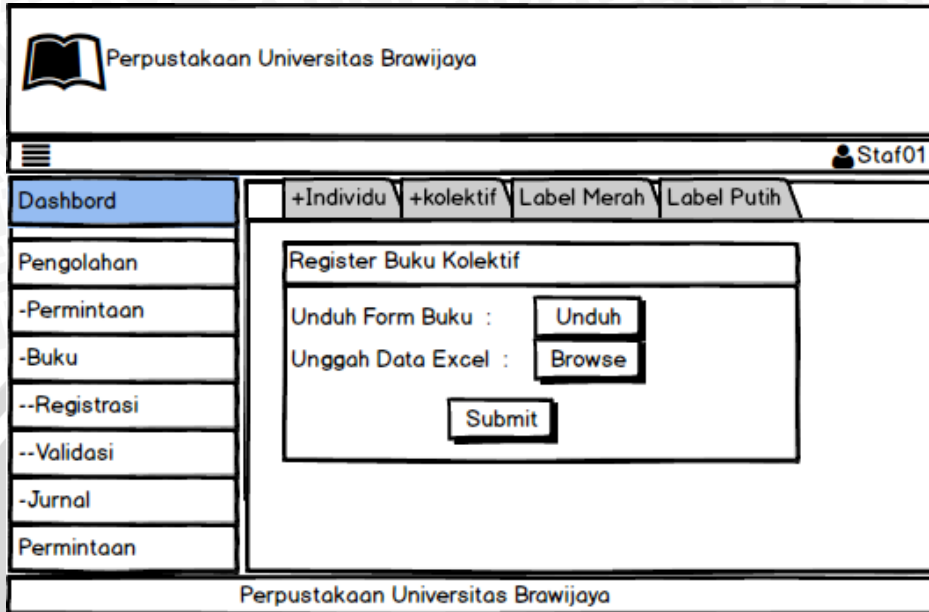
Gambar B.15 Halaman unggah form permintaan

2. Rancangan Prototipe Halaman Registrasi Bahan Pustaka Kolektif

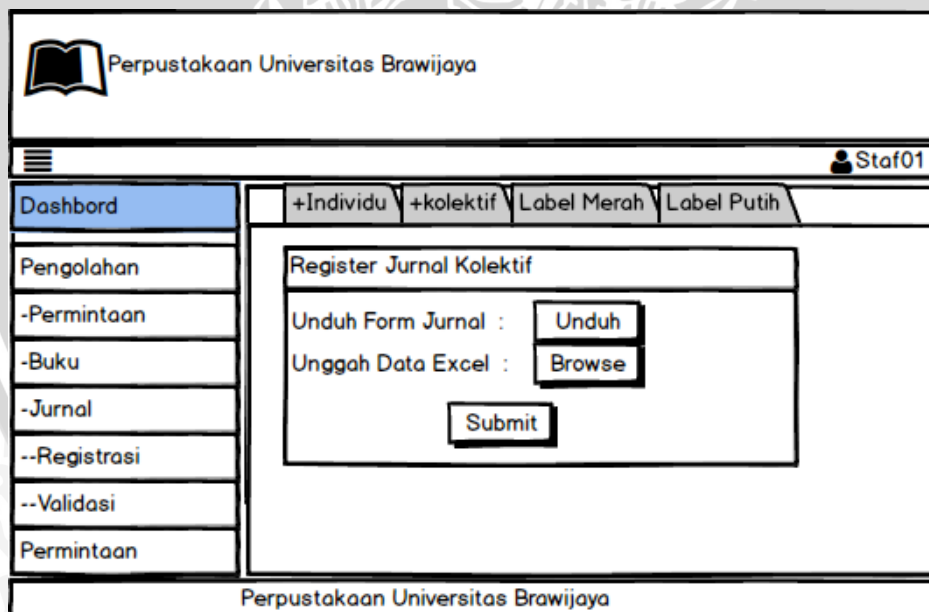
Halaman ini merupakan halaman registrasi bahan pustaka buku dan jurnal yang dilakukan secara kolektif. Proses permintaan kolektif dilakukan dengan cara

repository.ub.ac.id

mengunduh dan mengunggah form dalam bentuk file excel. Rancangan prototipe halaman unduh form buku dapat dilihat pada **Gambar B.16**. Rancangan prototipe halaman unggah form permintaan dapat dilihat pada **Gambar B.17**.



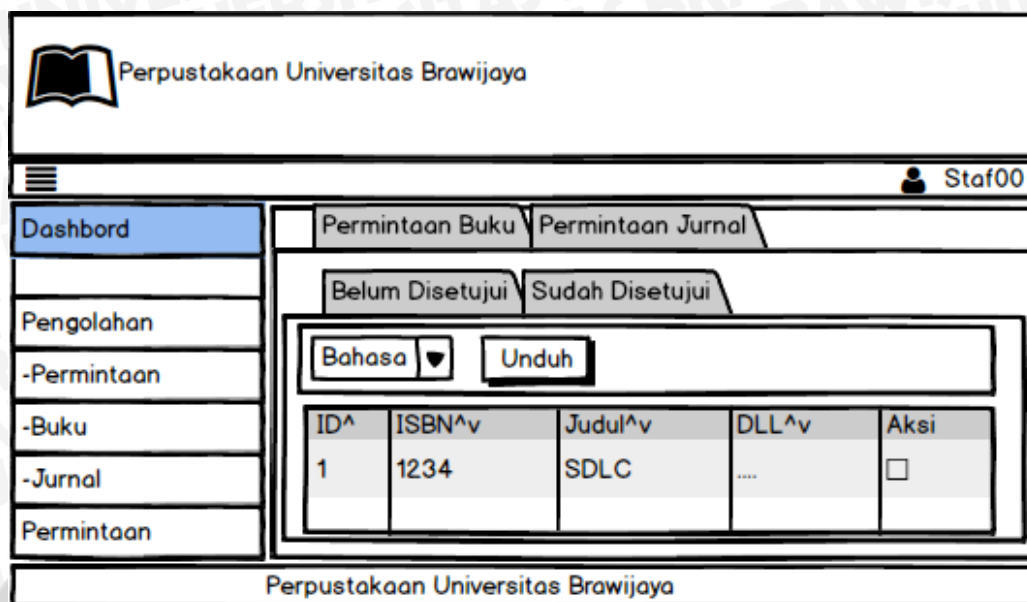
Gambar B.16 Halaman registrasi buku secara kolektif



Gambar B.17 Halaman registrasi jurnal secara kolektif

3. Rancangan Prototipe Halaman Pengolahan Permintaan Bahan Pustaka

Halaman ini merupakan halaman pengolahan permintaan bahan pustaka buku dan jurnal. Pada halaman ini akan ditambahkan fungsi unduh bahan pustaka dari rancangan yang dibuat sebelumnya. Rancangan prototipe halaman pengolahan permintaan bahan pustaka buku atau pun jurnal dapat dilihat pada **Gambar B.18**.

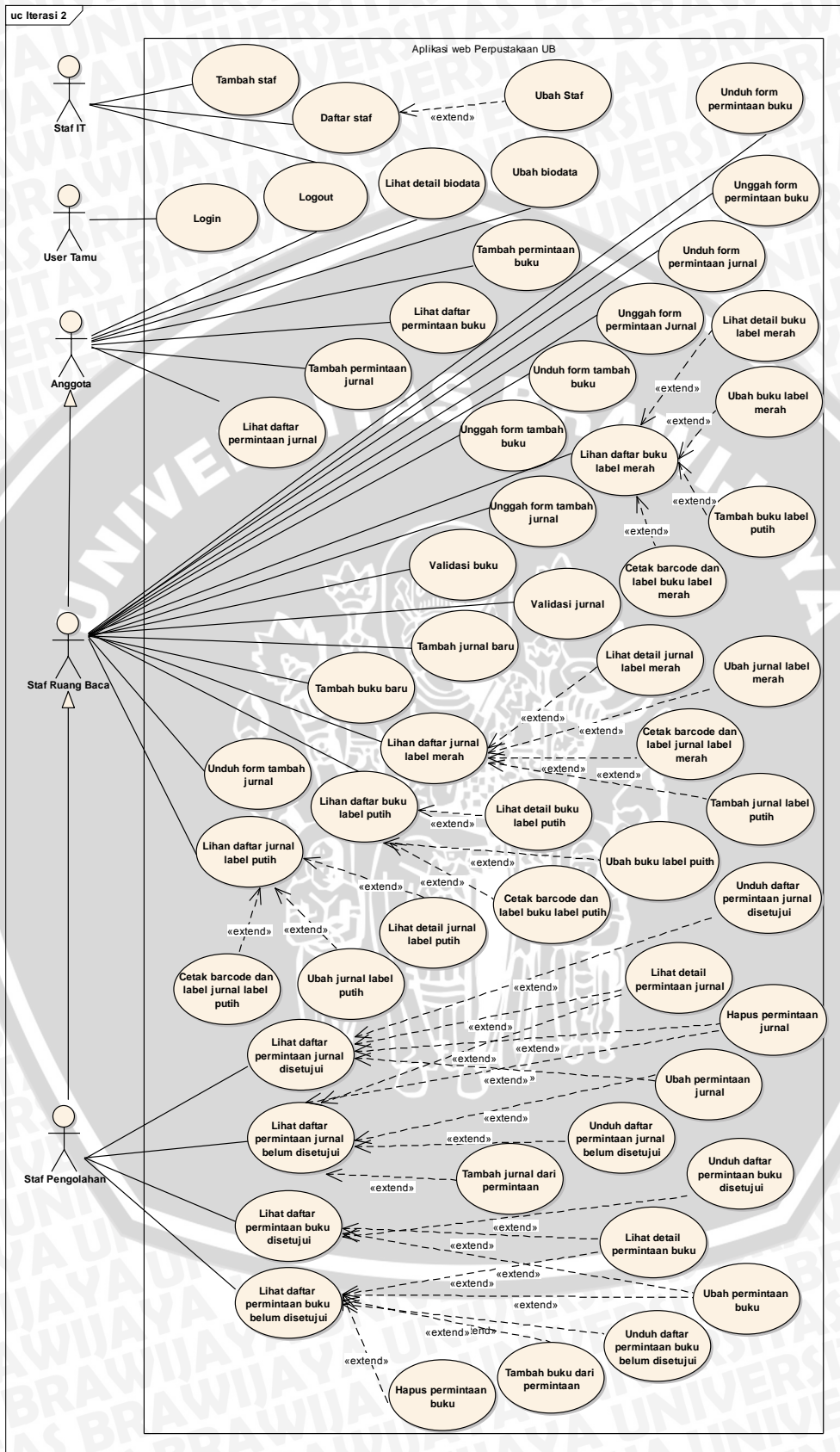


Gambar B.18 Halaman pengolahan permintaan bahan pustaka

B.3 Proses Iterasi 2

Iterasi ke-2 dilakukan karena terjadi perubahan pada kebutuhan perangkat lunak setelah dilakukan evaluasi pada prototyping tahap kedua. Hasil dari evaluasi yaitu perubahan hak akses staf IT, awalnya staf IT dapat mengakses keseluruhan fungsional. Setelah dilakukan evaluasi staf IT hanya dapat mengakses pengolahan staf. Tidak terjadi perubahan perancangan prototipe pada iterasi ini. Adapun perubahan yang terjadi dapat dilihat pada use case diagram pada **Gambar B.19**.





Gambar B.19 Diagram use case diagram iterasi ke-2

B.4 Proses Iterasi 3

Iterasi ke-3 dilakukan karena terjadi perubahan pada kebutuhan perangkat lunak setelah dilakukan evaluasi pada prototyping tahap ketiga. Hasil dari evaluasi tersebut adalah sebagai berikut.

1. Penambahan kebutuhan fungsional pencarian permintaan buku dan jurnal.
2. Penambahan kebutuhan fungsional pencarian buku label merah dan label putih.
3. Penambahan kebutuhan fungsional pencarian jurnal label merah dan label putih.

Adapun perubahan pada kebutuhan perangkat lunak dan perancangan dapat dilihat pada bab perancangan pada skripsi ini.

