RANCANG BANGUN SISTEM MANAJEMEN ARSIP ELEKTRONIK YANG MENGACU PADA STANDAR PENCATATAN ISO 15489 (STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG)

SKRIPSI

Laboratorium Rekayasa Perangkat Lunak

Diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer

Disusun Oleh:

ERISZA PUSPITASARI

NIM. 125150209111015

KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI

PROGRAM STUDI INFORMATIKA / ILMU KOMPUTER

PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER

UNIVERSITAS BRAWIJAYA

MALANG

2015

LEMBAR PERSETUJUAN

RANCANG BANGUN SISTEM MANAJEMEN ARSIP ELEKTRONIK YANG MENGACU PADA STANDAR PENCATATAN ISO 15489 (STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG)

SKRIPSI

Laboratorium Rekayasa Perangkat Lunak

Diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer

Disusun Oleh:

ERISZA PUSPITASARI

NIM. 125150209111015

Telah diperiksa dan disetujui oleh :

Dosen Pembimbing I,	**Dosen Pembimbing II,**
Denny Sagita R., S.Kom., M.Kom	**Yusi Tyroni Mursityo, S.Kom., MS**
NIK. 851124 06 1 1 0250	**NIP. 19800228 20060 001**

LEMBAR PENGESAHAN

RANCANG BANGUN SISTEM MANAJEMEN ARSIP ELEKTRONIK YANG MENGACU PADA STANDAR PENCATATAN ISO 15489 (STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG)

SKRIPSI

KONSENTRASI REKAYASA PERANGKAT LUNAK

Untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer

Disusun Oleh:

ERISZA PUSPITASARI
NIM. 125150209111015

Skripsi ini telah diuji dan dinyatakan lulus pada tanggal 29 Juni 2015

Penguji I,	**Penguji II,**
Fajar Pradana, S.ST,.M.Eng	**Adharul Muttaqin, ST.,MT.**
NIK. 87112116110371	**NIP. 197601212005011001**

Penguji III,

Aswin Suharsono, ST.,MT.

NIK. 840919 06 1 1 0251

Mengetahui,

Ketua Program Studi Informatika / Ilmu Komputer

Drs. Marji,M.T.

NIP. 19670801 199203 1 001

PERNYATAAN

ORISINALITAS SKRIPSI

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

Apabila ternyata di dalam naskah SKRIPSI ini dapat dibuktikan terdapat unsur-unsur PLAGIASI, saya bersedia SKRIPSI ini digugurkan dan gelarakademik 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, Juni 2015

Mahasiswa

Erisza Puspitasari

NIM. 125150209111015

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah SWT karena atas segala rahmat, taufik serta hidayah-Nya, sehingga skripsi dengan judul**:** ***Rancang Bangun Sistem Manajemen Arsip Elektronik Yang Mengacu Pada Standar Pencatatan ISO 15489 Studi Kasus UPT Laboratorium Kesehatan Hewan Malang,*** dapat diselesaikan dengan baik.

Dalam penyusunan tugas akhir ini penulis banyak sekali mendapatkan doa, bimbingan, arahan, bantuan, masukan dari berbagai pihak. Atas berbagai apresiasi tersebut, pada kesempatan ini penulis mengucapkan ucapan terima kasih yang sebesar-besarnya kepada :

1. Denny Sagita R., S.Kom, M.Kom dan Yusi Tyroni Mursityo, S.Kom., MS selaku pembimbing I dan II dalam penyusunan skripsi ini.
2. Drs. Marji, MT. dan Issa Arwani, ST., MT., selaku Ketua dan Sekretaris Program Studi Informatika
3. Ibunda dan Ayahanda yang selalu memberikan dukungan do'a dan kasih sayang untuk kelangsungan dan kesuksesan penulis. Semoga Allah SWT memberikan balasan terbaik untuk kedua orang tuaku tercinta.
4. Mashudi dan Ibu Novia selaku pembimbing di UPT Laboratorium Kesehatan Hewan Malang.
5. Dian Ariya Puji Pratama dan Galang Yuniar Mahendra yang telah turut membantu dalam penyelesaian skripsi ini.
6. Muhammad Taslim, yang telah memberikan dukungan dan motivasi untuk menyelesaikan skripsi ini.
7. Sahabat-sahabat SAP 2012 yang banyak memberikan informasi dan bantuan dalam penyusunan skripsi ini.
8. Serta semua pihak yang tidak dapat disebutkan satu persatu yang telah turut membantu baik secara langsung maupun tidak langsung dalam penyelesaian skripsi ini.

Penulis menyadari bahwa dalam skripsi ini masih sangat jauh dari kesempurnaan, untuk itu penyusun menerima kritik dan saran yang bersifat membangun dari semua pihak demi penyempurnaan selanjutnya. Akhir kata penulis berharap semoga skripsi ini dapat memberikan manfaat bagi *civitas* akademik Teknik Informatika khususnya dan Universitas Brawijaya pada umumnya.

Malang, Juni 2015

Penulis

repository.ub.ac.id

ABSTRAK

Erisza Puspitasari. 2015. Rancang Bangun Sistem Manajemen Arsip Elektronik Yang Mengacu Pada Standar Pencatatan ISO 15489 (Studi Kasus UPT Laboratorium Kesehatan Hewan Malang). Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya, Malang.
Dosen Pembimbing: Bapak Denny Sagita R., S.Kom, M.Kom dan Bapak Yusi Tyroni Mursityo, S.Kom., MS.

Sebagai sumber informasi, arsip mempunyai peranan yang sangat penting dalam pelaksanaan kegiatan administrasi yang menyangkut perencanaan, penganalisaan, pengambilan keputusan, dan pertanggungjawaban dalam menjalankan proses bisnis instansi seperti UPT Laboratorium Kesehatan Hewan Malang. Proses pengelolaan arsip UPT Laboratorium Kesehatan Hewan Malang saat ini dilakukan dengan pencatatan pada buku agenda, sehingga kurang efektif dalam proses pencarian dan temu kembali saat arsip tersebut dibutuhkan.

Sistem Manajemen Arsip Elektronik Yang Mengacu Pada Standar Pencatatan ISO 15489 dibangun untuk mempermudah proses pengelolaan arsip UPT Laboratorium Kesehatan Hewan Malang dalam bentuk digital. Sistem ini mengelola arsip mulai dari pencatatan pengaturan retensi arsip, peminjaman, pemusnahan sampai proses temu kembali arsip dalam bentuk digital. Sistem ini juga menyajikan laporan arsip dan mencetak berita acara pemusnahan arsip. Dalam pengembangannya, sistem ini dibangun dengan *framework* CodeIgniter, *template* bootstrap untuk mempermudah pengaturan tampilan dan MySql sebagai *database*. Hasil pengujian berbasis path menunjukkan bahwa unit modul dari program sudah memenuhi kebutuhan fungsional, dan pengujian *black box* dengan metode pengujian validasi menujukkan sistem 95% valid. Sedangkan hasil pengujian terhadap kesesuaian standar pencatatan ISO 15489 menunjukkan bahwa sistem memenuhi karakteristik standar sistem pencatatan yaitu *reliability*, *integrity*, *compliance*, *comprehensiveness*, dan *systematic*.

Kata kunci: Manajemen Arsip Elektronik, ISO 15489, CodeIgniter

UNIVERSITAS BRAWIJAYA

ABSTRACT

Erisza Puspitasari. 2015. ***Design of Electronic Archives Management System Based on The Recording Standard ISO 15489 (Case Study UPT Laboratorium Kesehatan Hewan Malang). Information Technology and Computer Science Program*****, Brawijaya** ***University*****, Malang.**
Supervisor : Mr. Denny Sagita R., S.Kom, M.Kom and Mr. Yusi Tyroni Mursityo, S.Kom., MS.

As a source of information, archives has a very important role in implementation of administrative activities related to planning, analysis, decision making, and accountability in carrying out business processes institutions such as UPT Laboratorium Kesehatan Hewan Malang. Records management process in UPT Laboratorium Kesehatan Hewan Malang currently done by recording on date book, so it less effective in the process of researching and retrieval when archives is required.

Electronic Archives Management System Based on The Recording Standard ISO 15489 built to assist archives management process of UPT Laboratorium Kesehatan Hewan Malang in digital form. This system manages archives from recording, setting archives retention, borrowing, disposal and archives retrieval process in digital form. This system also presents archives reports and print disposal event news of archives. In its development, the system built with CodeIgniter framework, bootstrap template to ease the display control and MySql as database. Path-based testing results shows that the unit modules of the program is already fulfill the functional requirements, and black box testing with validation testing method shows 95% validity in system function. Meanwhile the results of compliance testing based on record standard ISO 15489 shows that the system has fulfilled the characteristic of the standard recording system that is reliability, integrity, compliance, comprehensiveness, and systematic.

Keywords: Electronic Archives Management, ISO 15489, CodeIgniter

DAFTAR ISI

LEMBAR PERSETUJUAN ii
LEMBAR PENGESAHAN iii
ORISINALITAS SKRIPSI iv
KATA PENGANTAR v
ABSTRAK vii
ABSTRACT viii
DAFTAR ISI ix
DAFTAR GAMBAR xii
DAFTAR TABEL xiv
DAFTAR LAMPIRAN xvi
BAB I PENDAHULUAN 1
1.1. Latar Belakang 1
1.2. Rumusan Masalah 3
1.3. Batasan Masalah 3
1.4. Tujuan 3
1.5. Manfaat 4
1.6. Sistematika Penulisan 4
BAB II KAJIAN PUSTAKA DAN LANDASAN TEORI 6
2.1. Kajian Pustaka 6
2.2. UPT Laboratorium Kesehatan Hewan Malang 8
2.2.1. Tugas Lab Keswan Malang *8*
2.3. Arsip 9
2.3.1. Arsip Elektronik 9
2.3.2. Manajemen Arsip Elektronik 10
2.3.3. Dasar Pengelolaan Arsip Elektronik 11
2.4. Sistem Manajemen Arsip di Kantor UPT Laboratorium Kesehatan Hewan Malang 12
2.4.1 Alur Surat Masuk 13
2.4.2 Alur Surat Keluar 14
2.4.3 Alur Pencarian Arsip 15
2.5. ISO 15489 17
2.5.1. Records Management Processes (Proses Pengelolaan Catatan) 17
2.6. Perangkat Lunak 21
2.6.1. Definisi Perangkat Lunak 21
2.6.2. Karakteristik Perangkat Lunak 22
2.7. Rekayasa Perangkat Lunak 23
2.8 *System Development Life-Cycle* (Waterfall Model) 23
2.9 *Unified Modellling Language* (UML) 25
2.9.1. *Use Case Diagram* 26
2.9.2. *Activity Diagram* 26
2.9.3. *Sequence Diagram* 27

2.9.4. *Class Diagram* 29
2.10 Teori Normalisasi 32
2.11 MVC (Model View Controller) 33
2.12 Pengujian Perangkat Lunak 35
2.12.1. Functional Testing 35
2.12.2. White-Box Testing (pengujian kotak putih) 35
2.12.3. Black-Box Testing 37
2.12.4. Pengujian Kesesuaian Sistem Terhadap Standar ISO 37
BAB III METODOLOGI PENELITIAN 42
3.1. Identifikasi Permasalahan 42
3.2. Identifikasi Solusi 43
3.3. Realisasi Solusi 48
3.3.1 Pendefinisian Kebutuhan 49
3.3.2 Perancangan Sistem 52
3.3.3 Implementasi dan Pengujian Unit 54
3.3.4 Pengujian Integrasi Sistem 55
3.4. Analisa Hasil Realisasi Solusi 55
BAB IV ANALISIS DAN PERANCANGAN 56
4.1. Gambaran Umum 56
4.2. Elisitasi 56
4.3. Swimlane Diagram Proses Bisnis Sistem 58
4.4. Identifikasi Aktor dan Fungsionalitas Sistem 64
4.5. Daftar Kebutuhan Perangkat Lunak 66
4.6. Analisis Kebutuhan Sistem 68
4.6.1. Use Case Diagram Sistem Manajemen Arsip Elektronik 69
4.6.2. Skenario Use Case 70
4.6.3. Perancangan Activity Diagram 86
4.6.4. Perancangan Sequence Diagram 92
4.6.5. Perancangan Class diagram 96
4.6.6. Perancangan Entity Relationship Diagram (ERD) 96
4.6.7. Data Type Definition 100
4.6.8. Perancangan Antar Muka 108
BAB V IMPLEMENTASI DAN PENGUJIAN 111
5.1 Spesifikasi Perangkat Lunak 111
5.2 Batasan-Batasan Implementasi 111
5.3 Implementasi Basis data 112
5.4 Implementasi Antarmuka (User Interface) 112
5.4.1. Implementasi HTML dan CSS pada Antarmuka 112
5.4.2. Antarmuka (User Interface) 116
5.5 PENGUJIAN 128
5.5.1. Pengujian Unit 128
5.5.2. Pengujian Validasi 147
5.5.3. Pengujian Checklist ISO 15489 dan Kemampuan Sistem Terhadap Kesesuaian Standar Sistem Pencatatan ISO 15489 153

5.6 ANALISIS 164
5.6.1. Analisis Hasil Pengujian Unit 164
5.6.2. Analisis Hasil Pengujian Validasi 164
5.6.3. Analisis Hasil Pengujian Sistem Terhadap Kesesuaian ISO 15489 164
BAB VI PENUTUP 167
DAFTAR PUSTAKA 168
LAMPIRAN 171

DAFTAR GAMBAR

Gambar 2.1 Alur Pengelolaan Surat Masuk 13
Gambar 2.2 Alur Surat Keluar 14
Gambar 2.3 Alur Pencarian Arsip 15
Gambar 2.4 Pengelolaan Arsip Elektronik Di UPT Laboratorium Kesehatan Hewan Malang 16
Gambar 2.5 Desain dan Implementasi Sistem Perekaman (*Design and Implementation of Records System*(DIRS)) 19
Gambar 2.6 Waterfall Model 24
Gambar 2.7 Gambar *Use Case* Diagram 26
Gambar 2.8 Gambar *Activity Diagram* 27
Gambar 2.9 Gambar *Sequence Diagram* 28
Gambar 2.10 Gambar Asosiasi Kelas Sederhana 30
Gambar 2.11 Gambar *Class* dan Asosiasi 30
Gambar 2.12 Gambar Consultation *Class* 31
Gambar 2.13 Gambar Konsep MVC 34
Gambar 2.14 Flowchart dan Flow Graph Jalur Independen 36
Gambar 3.1 Diagram Blok Metode Penelitian 42
Gambar 3.2 Metode DIRS ISO 15489 45
Gambar 4.1 Alur Kelola Data User 58
Gambar 4.2 Alur Kelola Data Surat Masuk 59
Gambar 4.3 Alur Kelola Konsep Surat Keluar 60
Gambar 4.5 Alur Kelola Data Revisi Arsip 62
Gambar 4.6 Alur Kelola Data Peminjaman dan Pengembalian Arsip 63
Gambar 4.8 Usecase Diagram Sistem Manajemen Arsip Elektronik 69
Gambar 4.9 *Activity Diagram* Login 87
Gambar 4.10 *Activity Diagram* Tambah Data Arsip 88
Gambar 4.11 *Activity Diagram* Ubah Data Arsip 89
Gambar 4.12 *Activity Diagram* Hapus Data Arsip 90
Gambar 4.13 *Activity Diagram* Validasi Revisi Arsip 91
Gambar 4.14 Kelola Daftar Usul Musnah Arsip 92
Gambar 4.15 *Sequence Diagram* Kelola Data User 93
Gambar 4.16 *Sequence Diagram* Kelola Data Arsip 94
Gambar 4.17 Sequence Diagam Validasi Revisi Arsip 95
Gambar 4.18 *Sequence Diagram* Kelola Data Konsep Surat Keluar 95
Gambar 4.19 *Sequence Diagram* Kelola Data Surat Masuk 96
Gambar 4.20 Class Diagram Sistem Manajemen Arsip Elektronik 97
Gambar 4.21 Entity Relationship Diagram Sistem Manajemen Arsip Elektronik 98
Gambar 4.22 Rancangan Database Sistem Manajemen Arsip Elektronik 99
Gambar 4.23 Desain Tampilan halaman Login 108
Gambar 4.24 Desain Tampilan Halaman Data Tabel 109

Gambar 4.25 Desain Tampilan Halaman Data Tabel 109
Gambar 4.26 Desain Tampilan Halaman Profil 110
Gambar 4.27 Desain Halaman Laporan 110
Gambar 5.1. Implementasi Basis Data 112
Gambar 5.2 Halaman Login 116
Gambar 5.3. Halaman Menu Data User 117
Gambar 5.4. Halaman Form Validasi Data User 117
Gambar 5.5. Halaman Admin Menu Data Arsip 118
Gambar 5.6. Halaman Admin Data Revisi Arsip 118
Gambar 5.7 Halaman Admin Masa Simpan Arsip Telah Habis 119
Gambar 5.8 Halaman Admin Perpanjang Masa Simpan Arsip 119
Gambar 5.9 Halaman Admin Form Peminjaman Arsip 119
Gambar 5.10 Halaman Data Klasifikasi Arsip 120
Gambar 5.11 Halaman Form Tambah Data Klasifikasi Arsip 120
Gambar 5.12 Halaman Admin Notifikasi Validasi Arsip Baru 121
Gambar 5.13 Halaman *Preview* Arsip 121
Gambar 5.14 Halaman Notifikasi Revisi Arsip 121
Gambar 5.15 Halaman Admin Kepegawaian Data Karyawan 122
Gambar 5.16 Halaman Data Jabatan 122
Gambar 5.17 Halaman Admin Kepegawaian Data Jabatan Fungsional 123
Gambar 5.18 Halaman Persuratan Data Disposisi. 123
Gambar 5.19 Halaman Data Surat Masuk. 124
Gambar 5.20 Halaman Form Tambah Data Surat Masuk. 124
Gambar 5.21 Halaman Notifikasi Disposisi Surat Belum Dibaca 125
Gambar 5.22 Halaman Form Tambah Konsep Surat Keluar 125
Gambar 5.23 Halaman Notifikasi Disposisi Surat Masuk Baru. 126
Gambar 5.24 Halaman Form Disposisi Surat Masuk. 126
Gambar 5.25 Halaman Pilih Laporan 127
Gambar 5.26 Halaman Lihat Laporan 127
Gambar 5.27 Halaman Cetak Laporan 128
Gambar 5.28 *Flow Graph Function* input_data_dokumen dari Class Controller dokumen 130
Gambar 5.29 *Flowgraph Class function* input_surat_masuk() 134
Gambar 5.30 *Flowgraph Class function* edit_surat_masuk() 138
Gambar 5.31 *Flowgraph Class function* edit_surat_keluar() 141
Gambar 5.32 *Flowgraph Class function* input_berita_acara() 145
Gambar 5.33 Grafik Nilai Kesesuaian Sistem Dari Tiap Kategori Pertanyaan 166

DAFTAR TABEL

Tabel 2.1 Perbandingan *Boloix and Robillard's Systems Evaluation Framework* dan persyaratan ISO 15489 7
Tabel 2.2 Contoh *Form Checklist* ISO 15489 38
Tabel 2.3 *Example of an Audit Outcomes Report* 39
Tabel 2.2 Kriteria Presentase Tanggapan Responden terhadap Skor Ideal 41
Tabel 3.1 Kebutuhan Fungsional 51
Tabel 3.2 Kebutuhan Pengembangan 52
Tabel 4.1 Identifikasi Aktor dan Fungsionalitas Sistem 64
Tabel 4.2 Spesifikasi Kebutuhan Fungsional 66
Tabel 4.3 Spesifikasi Kebutuhan Non-fungsional 68
Tabel 4.4 Skenario *Use Case* Kelola Data User 70
Tabel 4.5 Skenario *Use Case* Login 74
Tabel 4.6 Skenario *Use Case* Kelola Data Arsip 76
Tabel 4.7 Skenario *Use Case* Daftar Usul Musnah Arsip. 80
Tabel 4.8 Skenario *Use Case* Kelola Data Peminjaman Arsip 81
Tabel 4.9 Skenario Kelola Disposisi Surat Masuk 83
Tabel 4.10 Skenario *Use Case* Validasi Registrasi Arsip 84
Tabel 4.11 Skenario *Use Case* kelola Data Berita Acara 85
Tabel 4.12 Tabel *Data Type Definition* dari tabel bagian 100
Tabel 4.13 Tabel *Data Type Definition* dari tabel berita_acara 100
Tabel 4.14 Tabel *Data Type Definition* dari tabel detail_berita_acara 100
Tabel 4.15 Tabel *Data Type Definition* dari tabel detail_pinjam_dokumen ... 101
Tabel 4.16 Tabel *Data Type Definition* dari tabel detail_tempat 101
Tabel 4.17 Tabel *Data Type Definition* dari tabel disposisi 101
Tabel 4.18 Tabel *Data Type Definition* dari tabel dokumen 102
Tabel 4.19 Tabel *Data Type Definition* dari tabel jabatan 103
Tabel 4.20 Tabel *Data Type Definition* dari tabel jabatan_fungsional 103
Tabel 4.21 *Tabel Data Type Definition* dari tabel karyawan 104
Tabel 4.21 Tabel *Data Type Definition* dari tabel kategori_dokumen 104
Tabel 4.22 Tabel *Data Type Definition* dari tabel klasifikasi 104
Tabel 4.23 Tabel *Data Type Definition* dari tabel kode_surat 105
Tabel 4.24 Tabel *Data Type Definition* dari tabel konsep_surat 105
Tabel 4.25 *Tabel Data Type Definition* dari tabel level 106
Tabel 4.26 Tabel *Data Type Definition* dari tabel level 106
Tabel 4.27 Tabel *Data Type Definition* dari tabel perusahaan 107
Tabel 4.28 Tabel *Data Type Definition* dari tabel pinjam_dokumen 107
Tabel 4.29 T*abel Data Type Definition* dari tabel revisi 107
Tabel 4.30 Tabel *Data Type Definition* dari tabel revisi 108
Tabel 4.31 Tabel *Data Type Definition* dari tabel revisi 108
Tabel 5.1. Spesifikasi Perangkat Lunak 111
Tabel 5.2. Implementasi HTML pada view dokumen.php. 112

Tabel 5.3. Potongan Kode CSS neon.css pada *Class* dokumen. 114
Tabel 5.4. Potongan Kode CSS pada bootstrap-modal.css. 115
Tabel 5.5 Kode Program function_input_dokumen() 129
Tabel 5.6 Kasus Uji *function_input*_dokumen() 131
Tabel 5.7 Kode Program *Class function* input_surat_masuk 132
Tabel 5.8 Kasus Uji *Class function* input_surat_masuk 135
Tabel 5.9 Kode Program *Class function* edit_surat_masuk 136
Tabel 5.10 Kasus Uji *Class function* edit_surat_masuk 139
Tabel 5.11 Kode Program *Class function* edit_surat_keluar 140
Tabel 5.12 Kode Program *Class function* edit_surat_keluar 142
Tabel 5.13 Kode Program *Class function* input_berita_acara 143
Tabel 5.14 Kode Program *Class function* input_berita_acara 146
Tabel 5.15 Hasil Pengujian Validasi 147
Tabel 5.16 Hasil Checklist Kesesuaian ISO 15489 153
Tabel 5.17 Hasil Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik 161
Tabel 5.18 Perhitungan Total Jawaban dikali Skor *Likert* 162
Tabel 5.19 Hasil Perhitungan Tingkat Kesesuaian Sistem Terhadap Tiap Kategori Pengujian 163
Tabel 5.20 Laporan Hasil *Checklist* Kesesuaian ISO 15489 165

DAFTAR LAMPIRAN

L1-1 Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik Terhadap Standar Sistem Pencatatan ISO 15489 171
L1-2 Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik Terhadap Standar Sistem Pencatatan ISO 15489 172
L1-3 Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik Terhadap Standar Sistem Pencatatan ISO 15489 173
L1-4 Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik Terhadap Standar Sistem Pencatatan ISO 15489 174
L1-5 Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik Terhadap Standar Sistem Pencatatan ISO 15489 175

BAB I
PENDAHULUAN

1.1. Latar Belakang

Arsip merupakan informasi terekam dalam bentuk media apapun, yang diciptakan, diterima dan dikelola oleh suatu organisasi atau individu dalam rangka pelaksanaan kegiatan atau karena kewajiban legalnya. Arsip merupakan salah satu faktor yang menunjang kelancaran penyelenggaraan kegiatan di suatu instansi. Arsip mempunyai peranan yang sangat penting karena arsip sebagai pusat ingatan, sumber informasi dalam rangka melaksanakan kegiatan administrasi yang menyangkut perencanaan, penganalisaan, pengambilan keputusan, pengendalian dan pertanggungjawaban dalam menunjang kemajuan di bidang administrasi dan manajemen [19]. Arsip sangat penting, baik dalam kehidupan sehari-hari, organisasi, bisnis, maupun lembaga pemerintahan seperti UPT Laboratorium Kesehatan Hewan Malang, karena arsip merupakan suatu aset instansi yang menyimpan berbagai informasi penting dan merupakan pertanggungjawaban kerja yang harus selalu dipelihara, dirawat dan dilindungi keberadaannya.

Unit Pelaksana Teknis (UPT) Laboratorium Kesehatan Hewan Malang merupakan sebuah institusi pemerintahan yang mempunyai tugas Dinas di bidang pemeriksaan, penyidikan dan diagnosis penyakit hewan serta pelayanan pengujian mutu produk pangan asal hewan dan mutu produk non PAH [17]. Dalam setiap proses pelayanan yang dijalankan oleh UPT Laboratorium Kesehatan Hewan Malang pasti menghasilkan arsip yang harus diorganisir, seperti surat masuk, surat keluar, laporan hasil pengujian, laporan data pegawai, sampai dengan laporan kegiatan dalam institusi tersebut. Saat ini pengelolaan semua arsip fisik masih dicatat dan disimpan dalam folder fisik. Seringkali ketika staf mencari surat-surat penting atau data asset lainnya seperti laporan hasil uji pada saat dibutuhkan memerlukan waktu yang cukup lama, mulai dari membongkar folder fisik, mencari di buku agenda lalu mencari ke tempat penyimpanan arsip, belum lagi ternyata arsip yang dicari telah dipinjam dan dikembalikan tidak sesuai pada tempatnya. Masalah tersebut tidak diketahui karena tidak adanya pencatatan peminjaman arsip, dan pengembalian arsip yang tidak teratur. Selain itu, arsip

elektronik yang berasal dari kiriman email tidak tersimpan dalam suatu penyimpanan khusus sehingga menimbulkan kemungkinan arsip tersebut terhapus secara tidak sengaja. Hampir pasti bahwa manajemen arsip adalah sebuah tantangan yang sulit dan pasti rentan terhadap kesalahan manusia. Selalu ada risiko bahwa arsip mungkin salah penempatan atau bahkan hilang mengingat proses pengarsipan yang kurang baik. Sesuai dengan Undang-Undang Nomor 43 Tahun 2009 tentang Kearsipan dan Undang-Undang Nomor 28 Tahun 2012 yang membahas alih media arsip, bahwa setiap instansi/organisasi baik pemerintah maupun swasta berkewajiban menyimpan, memelihara, dan menyelamatkan arsip, maka diperlukan manajemen pencatatan arsip elektronik yang sesuai dengan standar ISO tentang pencatatan arsip untuk menjamin ketersediaan arsip sebagai bahan bukti akuntabilitas kinerja instansi dalam penyelenggaraan kegiatan di kantor UPT Laboratorium Kesehatan Hewan Malang.

International Organization for Standarization (ISO) adalah badan standar dunia yang dibentuk untuk mengurus perkembangan dan publikasi mengenai standarisasi internasional [18]. ISO seri 15489 merupakan sebuah standar internasional dalam *Records Management* yang bertujuan untuk menjamin ketersediaan arsip dalam penyelenggaraan kegiatan sebagai bahan akuntabilitas kinerja dan alat bukti yang sah berdasarkan suatu sistem yang memenuhi syarat diantaranya adalah andal, sistematis, utuh, menyeluruh dan sesuai dengan norma, standar, prosedur, dan kriteria [7] .

Dalam penelitian ini, penulis mengembangkan sebuah solusi sistem manajemen arsip elektronik sebagai alat untuk mewujudkan efisiensi pengelolaan arsip yang mengacu pada standar sistem pencatatanarsip ISO 15489 dalam hal registrasi, pencarian, penyimpanan, pelacakan, dan pemusnahan arsip secara digital yang dapat diterapkan di kantor UPT Laboratorium Kesehatan Hewan Malang. Dengan menerapkan sistem manajemen arsip elektronik yang mengacu pada standar sistem pencatatanISO 15489, diharapkan pengelolaan arsip di kantor UPT Laboratorium Kesehatan Hewan Malang lebih terstruktur dan lebih efisien dari segi pencatatan sampai proses temu kembali ketika arsip tersebut dibutuhkan.

1.2. Rumusan Masalah

Rumusan masalah dari penelitian ini adalah sebagai berikut :

1. Bagaimana merancang Sistem Manajemen Arsip Elektronik yang mengacu pada standar sistem pencatatan ISO 15489 yang sesuai dengan prosedur standar operasional di UPT Laboratorium Kesehatan Hewan Malang?
2. Bagaimana membuat Sistem Manajemen Arsip Elektronik yang mengacu pencatatan ISO 15489 di UPT Laboratorium Kesehatan Hewan Malang?
3. Bagaimana pengujian Sistem Manajemen Arsip Elektronik yang mengacu pada standar sistem pencatatan ISO 15489 di UPT Laboratorium Kesehatan Hewan Malang?

1.3. Batasan Masalah

1. Sistem manajemen arsip elektronik yang dibuat sesuai dengan alur proses bisnis yang berlaku di UPT Laboratorium Kesehatan Hewan Malang.
2. Kategori arsip yang digunakan sesuai dengan struktur dokumen ISO 9001:2008 yaitu Pedoman Kerja, Prosedur Kerja, Instruksi Kerja dan Formulir Standar yang digunakan sebagai dasar pengklasifikasian arsip di UPT Laboratorium Kesehatan Hewan Malang.

1.4. Tujuan

Tujuan dari penelitian ini adalah sebagai berikut:

1. Menganalisa dan merancang Sistem Manajemen Arsip Elektronik sesuai prosedur standar operasional dan kebutuhan di UPT Laboratorium Kesehatan Hewan Malang dan mengacu pada standar sistem pencatatan ISO 15489.
2. Membuat Sistem Manajemen Arsip Elektronik di UPT Laboratorium Kesehatan Hewan Malang yang sesuai dengan standar ISO 15489.
3. Menguji Sistem Manajemen Arsip Elektronik berdasarkan ISO 15489 di UPT Laboratorium Kesehatan Hewan Malang.

1.5. Manfaat

Manfaat yang didapat dari rancang bangun sistem informasi ini antara lain sebagai berikut:

1. Mempermudah dalam pengelolaan arsip sesuai dengan standar prosedur yang ada di UPT Laboratorium Kesehatan Hewan Malang dalam format digital.
2. Meningkatkan efektivitas dan efisiensi kinerja karyawan dalam hal pengelolaan arsip dalam format digital.
3. Mempermudah dalam proses temu kembali arsip.

1.6. Sistematika Penulisan

Sistematika penulisan laporan skripsi ini adalah sebagai berikut :

BAB I PENDAHULUAN

Pendahuluan berisi tentang latar belakang, perumusan masalah, pembatasan masalah, tujuan, manfaat, dan sistematika penulisan.

BAB II KAJIAN PUSTAKA DAN DASAR TEORI

Bab ini membahas mengenai teori – teori yang digunakan dalam penyusunan laporan skripsi ini. Teori-teori yang digunakan sebagai landasan teori antara lain ISO 15489, *Unified Model Language* (UML), Rekayasa Perangkat Lunak, *Framework CodeIgniter*, dan Pengujian Perangkat Lunak.

BAB III METODOLOGI PENELITIAN

Metode penelitian membahas metode yang digunakan dalam penelitian yang terdiri dari studi literatur, wawancara, analisis kebutuhan, perancangan, implementasi, pengujian dan pengambilan kesimpulan.

BAB IV ANALISIS DAN PERANCANGAN

Bab ini berisi tentang alur data aplikasi, dan perancangan desain aplikasi.

BAB V IMPLEMENTASI

Bab ini berisi tentang implementasi sistem yang dibuat secara keseluruhan dari perancangan yang telah dilakukan di Bab IV.

BAB VI PENGUJIAN DAN ANALISIS

Bab ini berisi pengujian dan analisis hasil pengujian yang telah diperoleh dari pengujian terhadap aplikasi yang telah dibuat.

BAB VII PENUTUP

Bab ini berisi tentang kesimpulan dan saran yang didapat setelah melakukan analisa, desain, implementasi dan pengujian terhadap Rancang Bangun Sistem Manajemen Arsip Elektronik yang mengacu pada standar sistem pencatatan ISO 15489.

BAB II
KAJIAN PUSTAKA DAN LANDASAN TEORI

2.1. Kajian Pustaka

Dalam sebuah penelitian yang berjudul "*UP DOCUTRAK: AN ASSESSMENT OF AN IN-HOUSE DEVELOPED DOCUMENT TRACKING SYSTEM (DTS)*" menyebutkan bahwa sebagian besar proyek komputerisasi Universitas ditangani menggunakan *records*(catatan). Mayoritas dari proyek komputerisasi tersebut dapat diklasifikasikan sebagai sistem pencatatan sejak mereka menerapkan *International Organization for Standardization's* (ISO) untuk sistem pencatatan dengan kemampuan untuk menangkap, memelihara, dan menyediakan akses terhadap informasi. Sebagai sistem, proyek komputerisasi ini juga terdiri dari semua komponen yang diperlukan kebijakan, proses, orang, teknologi, dan alat-alat. Mengingat jumlah proyek dan sifat umum fungsi mereka menunjukkan adanya kemungkinan adanya standar umum proses pengembangan atau kerangka yang mengatur sistem dan catatan tersebut sebagai *Records Management Sistem* (RMS).

Di antara sistem yang disebutkan, Universitas mengembangkan *Document Tracking System* (DTS) sebagai contoh sistem pencatatan. DocuTrak adalah DTS yang menggunakan label *barcode* untuk melacak alur arsip mereka melalui berbagai kantor unit UP (*University of the Philippines*) Diliman (UPD). Berbagai jenis arsip seperti surat-menyurat atau formulir akuntansi dapat dipantau menggunakan DocuTrak. Untuk membuat arsip DocuTrak, pengguna harus mengisi kolom yang diperlukan untuk mengisi properti arsip seperti judul arsip, pengarang, jenis arsip, tindakan yang diperlukan dan jenis akses, dan nomor pelacakan yang unik. Nomor pelacakan adalah sebuah identifikasi unik yang melekat pada arsip sebagai label *barcode*. Nomor pelacakan memungkinkan kantor Unit terlibat dalam pengolahan arsip untuk membuat, menerima, bertindak atas, merilis, dan proses pelacakan arsip-arsip menggunakan DocuTrak.

Untuk memverifikasi dan memahami konteks konseptualisasi proyek DocuTrak, perencanaan proyek yang sebenarnya adalah melihat dan mengidentifikasi proyek asli tujuan dan fitur, rencana pelaksanaan proyek, dan mekanisme dukungan yang tersedia pada saat awal proyek dan pembangunan. Prosedur yang sebenarnya diambil selama pelaksanaan juga diselidiki sebagai wawasan seperti apa faktor-faktor lain dapat mempengaruhi hasil proyek. Penilaian dilakukan dengan menggunakan BRSEF dan ISO 15489. Rekomendasi didasarkan pada hasil penilaian. Penelitian ini menggunakan *Boloix 'and Robillard System Evaluation Framework* (SEF) sebagai alat. SEF sangatlah luas dan cukup fleksibel sehingga memungkinkan penyelidikan atas kepatuhan terhadap standar yang lebih spesifik seperti ISO 15489, khususnya fungsi, karakteristik, dan kepatuhan sistem pencatatan terhadap metodelogi DIRS (lihat tabel 2.1)[16].

Tabel 2.1 Perbandingan *Boloix and Robillard's Systems Evaluation Framework* dan persyaratan ISO 15489

Dimensions of the system Evaluation Framework	***ISO 15489 Requirements***
Environtment *System and Environtment* *System* *Project*	*Record System Characteristic :* *Reliability* *Integrity* *Compliance* *Comprehensiveness* *Systematic* *(Formal Process and Methodelogies)*
System (Product Features)	*Records Management Processes and Controls*
Project	*DIRS Methodelogy*

Sumber : [16]

Penilaian sistem perekaman menggunakan kriteria ISO 15489 mengungkapkan bahwa :

1. DocuTrak sistematis terutama karena sistem otomatis. Tetapi tidak memiliki kebijakan yang mendukung untuk meresmikan atau sistematisasi manajemen arsip fisiknya, serta untuk menentukan aktor dan tanggung jawab, dan sanksi untuk ketidakpatuhan. Pengarsipan sistem perlu diperbaharui.
2. Pengembangan dan pelaksanaan proses sistem ini tidak mengikuti langkah-langkah yang ditentukan dalam metodelogi DIRS. Pertama, investigasi awal tidak dilakukan. Kemudian analisis bisnis terbatas pada UPCC. Desain sistem tidak memanfaatkan umpan balik dari Unit UP yang lain. Pada catatan lain, proses pengarsipan desain tersedia namun tidak diperbarui. Sistem ini mampu mencatat *capture*, pendaftaran, klasifikasi, penyimpanan, akses, dan pelacakan. Pelacakan informasi mencerminkan lokasi arsip dan tindakan arsip tetapi informasi juga terbatas ke unit dan individu, untuk menyetujui, menolak, menerima, dan merilis masing-masing tindakan. Sistem ini juga tidak mampu mencatat pengarsipan atau pemusnahan karena kemampuan tersebut bukan bagian dari desain sistem.

2.2. UPT Laboratorium Kesehatan Hewan Malang

UPT. Laboratorium Keswan Malang berdiri sejak tahun 1975 sebagai Laboratorium diagnostik, Non Eselon berkedudukan di Jl. Raya Langsep Tahun 1996 Laboratorium Lab keswan pindah lokasi di Jl Raya Pakis Jajar. Berdasarkan PERGUB No 19 tahun 2000 menjadi BALAI LABORATORIUM KESWAN dengan eselon III. Selanjutnya berdasarkan PERGUB No 130 tahun 2008 kembali menjadi UPT Laboratorium Keswan [17].

2.2.1. Tugas Lab Keswan Malang

UPT Laboratorium Kesehatan Hewan mempunyai tugas melaksanakan Sebagaian tugas Dinas di Bidang PEMERIKSAAN, PENYIDIKAN DAN DIAGNOSIS PENYAKIT HEWAN.

Fungsi Lab Keswan Malang :

1. Pengumpulan / pengambilan speciment
2. Pemeriksaan Penyakit Hewan;
3. Diagnosis Penyakit Hewan;
4. Evaluasi Penyakit Hewan;
5. Penanggulangan Penyakit Hewan;
6. Pembinaan Penyidikan Penyakit Hewan;
7. Pelayanan Pengujian Mutu Produk Pangan asal Hewan dan Mutu produk non PAH;
8. Pelayanan Kepada Masyarakat.

2.3. Arsip

Menurut Drs. Basir Barthos dalam Manajemen Kearsipan arsip adalah : "Setiap catatan tertulis baik dalam bentuk gambar ataupun bagan yang memuat keterangan-keterangan mengenai subyek (pokok persoalan)ataupun peristiwa yang dibuat orang untuk membantu daya ingatan orang (itu) pula" [4].

Berdasarkan UU No.43 Th. 2009: "Arsip adalah rekaman kegiatan atau peristiwa dalam berbagai bentuk dan media sesuai dengan perkembangan teknologi informasi dan komunikasi yang dibuat dan diterima oleh lembaga negara, pemerintahan daerah, lembaga pendidikan, perusahaan, organisasi politik, organisasi kemasyarakatan, dan perseorangan dalam pelaksanaan kehidupan bermasyarakat, berbangsa, dan bernegara.

2.3.1. Arsip Elektronik

Pengertian arsip elektronik menurut *Information Resources Management Service of the United States General Services Administration* yang dikutip oleh Wallace, Lee, dan Schubert dalam bukunya "*Records Management : Integrated Information System*" : "*Electronic record are record just as much as paper document, and creation, maintenance, use and disposition must be managed accordingly.*

Electronically created enformation in electronic mail system, business graphic systems, digitized voice mail systems, office electronic massage and calendaring system, MIS, and Decision Support System must be reviewed".

Dalam buku tersebut juga dijelaskan contoh-contoh dari arsip elektronik tersebut :"*The operation of record system in which a computer is required for the user to create, work with, or delete record. Example of electronic records are those resides on magnetic tapes, disk and drums, on video files, and an optical disks"*[1].

2.3.2. Manajemen Arsip Elektronik

Informasi yang direkam diproses dan disimpan secara elektronik dinamakan arsip elektronik. *National Archives of Canada* memberikan definisi arsip elektronik sebagai arsip bacaan mesin yang mudah dimanipulasi, ditransmisi, dan diproses degan menggunakan komputer digital. Pada saat ini arsip elektronik sudah menjadi salah satu bagian penting dalam lingkungan pekerjaan yang bergantung pada teknologi informasi berbasis komputer. Evolusi yang cepat dalam teknologi komputer dan juga pada sistem informasi telah melahirkan isu mengenai manajemen arsip elektronik yang berkaitan dengan ketersediaan, otensitas, kelengkapan, serta nilaiguna. Di lingkungan kerja yang berbasis elektronik, beberapa perubahan perlu diperhatikan dalam bidang kearsipan mulai dari penciptaan, penggunaaan, penilaian, dan pemeliharaan. Termasuk, pengkajian kembali peranan arsiparis dan institusi kearsipan yang turut ambil bagian pada fase evolusi teknologi informasi ini.

Ada empat prinsip dalam kerangka pelaksanaan manajemen arsip elektronik menurut *International Council on Archives* [9]:

a. Arsip elektronik harus masuk dalam daur hidup sistem elektronik yang menciptakan arsip untuk menjamin penciptaan dan retensi arsip elektronik yang otentik, terpercaya, dan terpelihara.
b. Harus ada jaminan bahwa penciptanya menciptakan arsip yang otentik, terpercaya, dan terpelihara.
c. Adanya proses penilaian arsip elektronik

d. Kebutuhan akan pemeliharaan dan pengaksesan untuk menjamin arsip elektronik dapat tersedia, dapat diakses, dan dimengerti.

2.3.3. Dasar Pengelolaan Arsip Elektronik

Dalam Peraturan Pemerintah No 28 Tahun 2012 tentang Kearsipan, pada Pasal 49 menyebutkan bahwa :

1) Alih media arsip sebagaimana dimaksud dalam Pasal 48 dilaksanakan dalam bentuk dan media apapun sesuai kemajuan teknologi informasi dan komunikasi berdasarkan ketentuan peraturan perundang-undangan.
2) Dalam melakukan alih media arsip pimpinan masing-masing pencipta arsip menetapkan kebijakan alih media arsip.
3) Alih media arsip dilaksanakan dengan memperhatikan kondisi arsip dan nilai informasi.
4) Arsip yang dialihmediakan tetap disimpan untuk kepentingan hukum berdasarkan ketentuan peraturan perundang-undangan.
5) Alih media arsip diautentikasi oleh pimpinan di lingkungan pencipta arsip dengan memberikan tanda tertentu yang dilekatkan, terasosiasi atau terkait dengan arsip hasil alih media.
6) Pelaksanaan alih media arsip dinamis ditetapkan oleh pimpinan pencipta arsip.
7) Berita Berita acara alih media arsip dinamis sekurang-kurangnya memuat:
   a. waktu pelaksanaan;
   b. tempat pelaksanaan;
   c. jenis media;
   d. jumlah arsip;
   e. keterangan proses alih media yang dilakukan;
   f. pelaksana; dan
   g. penandatangan oleh pimpinan unit pengolah dan/atau unit kearsipan.
8) Daftar arsip dinamis yang dialihmediakan sekurang-kurangnya memuat:
   a. unit pengolah;

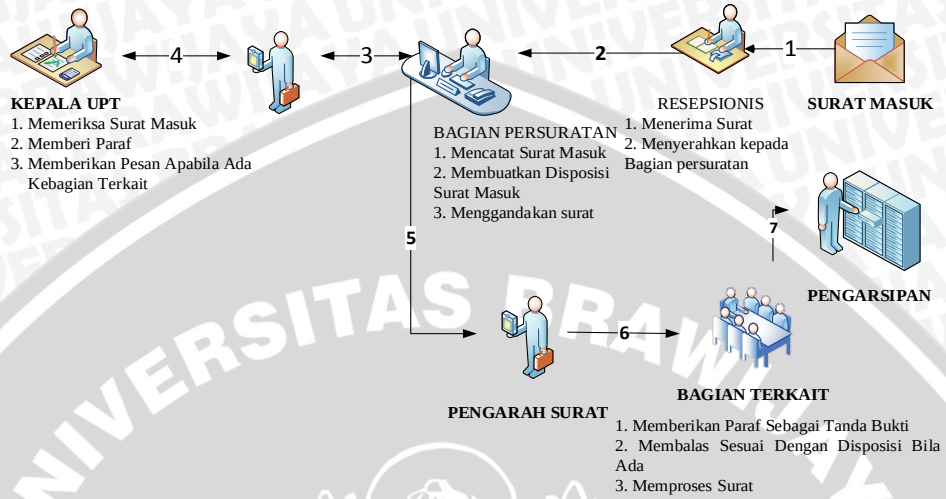
b. nomor urut;
c. jenis arsip;
d. jumlah arsip;
e. kurun waktu; dan
f. keterangan.

9) Pelaksanaan alih media arsip dinamis ditetapkan oleh pimpinan pencipta arsip.
10) Arsip hasil alih media dan hasil cetaknya merupakan alat bukti yang sah sesuai dengan ketentuan peraturan perundang-undangan.

2.4. Sistem Manajemen Arsip di Kantor UPT Laboratorium Kesehatan Hewan Malang

Berikut merupakan alur sistem manajemen pengelolaan arsip surat masuk dan surat keluar yang sedang berjalan di kantor UPT Laboratorium Kesehatan Hewan Malang. Pencatatan arsip di kantor UPT Laboratorium Kesehatan Hewan Malang hanya dilakukan secara manual pada bagian persuratan saja, arsip yang lain juga belum ada sistem pencatatan. Kantor UPT Kesehatan Hewan Malang mempunyai 3 bagian yaitu bagian pelayanan kesehatan hewan, pelayanan kesehatan masyarakat veteriner dan bagian TU. Sub bagian TU sendiri membawahi sub bagian kepegawaian, persuratan, dan administratif yang akan menghasilkan banyak arsip yang harus dikelola mulai dari pencatatan sampai dengan proses temu kembali. Gambar sistem manajemen arsip yang sedang berjalan di kantor UPT Laboratorium Kesehatan Hewan Malang dapat direpresentasikan pada gambar 2.1.

2.4.1 Alur Surat Masuk

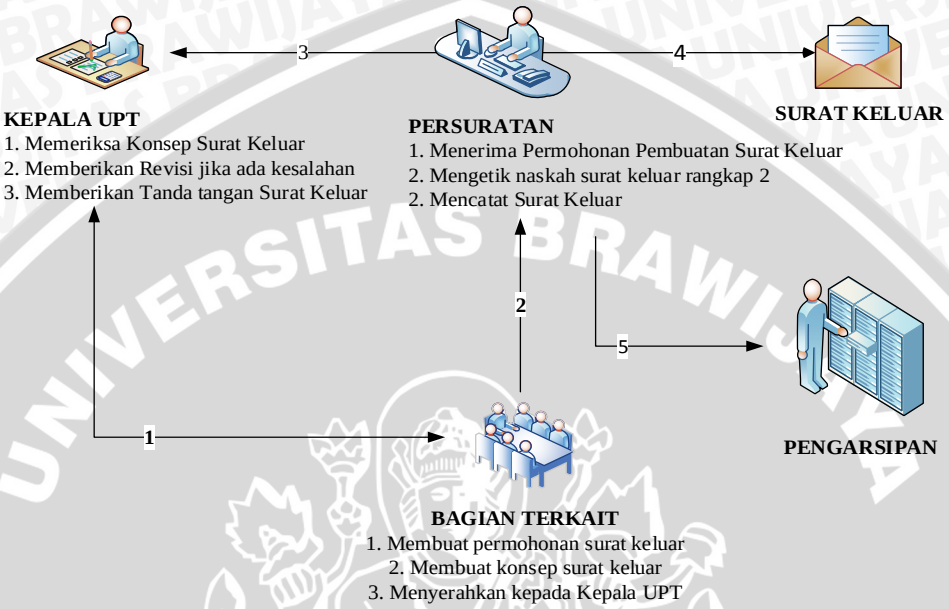


Gambar 2.1 Alur Pengelolaan Surat Masuk

Alur pengelolaan surat masuk di kantor UPT Laboratorium Kesehatan Hewan Malang

1. Surat masuk ke kantor UPT Laboratorium Kesehatan Hewan Malang diterima bagian administratif, bagian administratif menyerahkan kepada bagian persuratan.
2. Bagian persuratan mencatat surat masuk pada agenda surat masuk, mengisi kartu kendali 2 rangkap, memberi lembar disposisi, menyerahkan kepada kepala UPT untuk diketahui.
3. Kepala UPT memeriksa surat yang masuk, memberi paraf, dan mengisi disposisi jika diperlukan. Surat dikembalikan kepada administratif. Bagian persuratan menggandakan surat masuk, satu untuk didistribusikan kepada bagian yang terkait, satu untuk diarsipkan.
4. Pengarah surat mendistribusikan pada pihak yang dituju.
5. Setelah surat masuk ditindak lanjuti, surat disimpan sesuai kode klasifikasinya.

2.4.2 Alur Surat Keluar



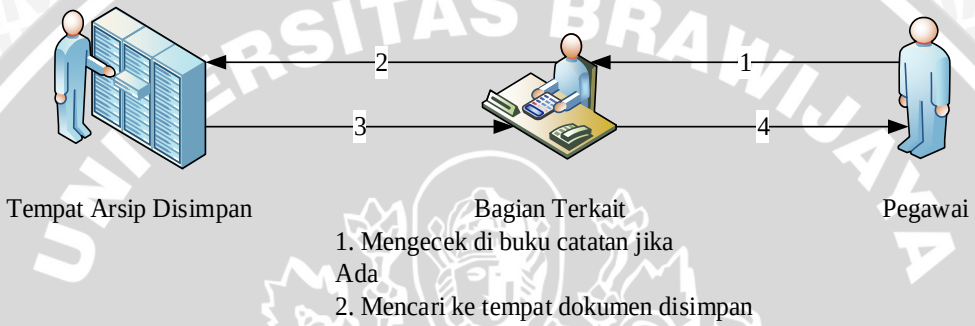
Gambar 2.2 Alur Surat Keluar

Keterangan :

1. Kepala Bagian membuat konsep surat keluar lalu menyerahkan konsep kepada kepala UPT untuk diperiksa
2. Kepala UPT memeriksa konsep surat, apabila sudah sesuai Kepala UPT akan memberikan paraf, apabila belum sesuai maka akan dikembalikan kepada bagian untuk direvisi.
3. Bagian terkait melakukan permohonan pembuatan surat keluar sesuai dengan perihal kepada persuratan.
4. Bagian Persuratan menerima permohonan pembuatan surat, kemudian membuat surat keluar rangkap 2, menyerahkan naskah surat keluar kepada Kepala.

5. Kepala UPT memeriksa naskah surat keluar, kemudian menandatangani apabila surat sudah sesuai/ benar, jika belum sesuai/benar surat dikembalikan lagi kepada persuratan untuk di revisi.
6. Apabila surat sudah sesuai, persuratan mengecek lagi, memberikan cap, nomor, satu surat untuk dikirimkan, satu surat diarsipkan.

2.4.3 Alur Pencarian Arsip

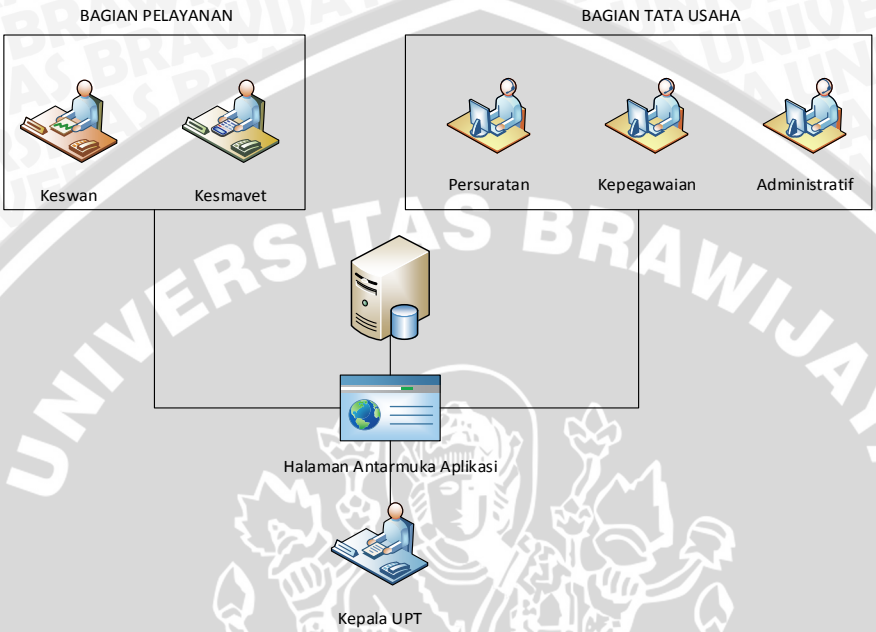


Gambar 2.3 Alur Pencarian Arsip

Keterangan

1. Pegawai menanyakan data arsip yang di cari ke bagian terkait.
2. Staf bagian akan mengecek dari buku pencatatan jika arsip tersebut berupa surat, jika bukan staf akan mencari langsung ke loker atau tempat penyimpanan karena tidak ada pencatatan arsip. Staf kemudian menayakan keperluan pencarian arsip/arsip tersebut, apabila untuk dipinjam oleh pegawai dalam instansi itu sendiri biasanya arsip tidak dicatat dan langsung dibawa, jika dari luar instansi, staf biasanya hanya mencatat pada buku agendanya bahwa arsip tersebut dipinjam.
3. Apabila untuk dilihat saja maka staf hanya menyajikan arsip/arsip yang diambil dari tempat penyimpanan, kemudian di serahkan ke pegawai dengan ketentuan arsip/arsip tidak berpindah dari ruangan.

2.1 Pemodelan Sistem Manajemen Arsip Elektronik



Gambar 2.4 Pengelolaan Arsip Elektronik Di UPT Laboratorium Kesehatan Hewan Malang

Keterangan :

1. Semua proses kegiatan pencatatan arsip dilakukan secara terkomputerisasi dan disesuiakan dengan standar ISO 15489 manajemen pencatatan arsip meliputi :
   a. Pencatatan dan registrasi arsip yang dilakukan oleh admin masing-masing bagian kemudian di validasi oleh administratif.
   b. Peminjaman arsip yang dicatat secara sistematis
   c. Jadwal retensi arsip secara otomatis dapat memberi peringatan kepada admin masing-masing bagian ketika masa retensi arsip tersebut sudah habis.
   d. Pembuatan berita acara yaitu berita acara pemusnahan arsip.
   e. Revisi arsip untuk mencegah arsip yang sudah tidak berlaku digunakan secara sengaja atau tidak sengaja.

2.5. ISO 15489

ISO singkatan dari *The International Organization for Standardization*. Seri 15489 merupakan standarisasi dalam pengelolaan *Records Management* (ISO/TR 15489-1). *The Australia Standar AS* 4390 – 1996 sebagaimana dikutip Kennedy dan Schauder mendefinisikan arsip sebagai rekod yang memiliki nilai berkelanjutan. Pengertian rekod itu sendiri adalah "…informasi terekam, dalam berbagai bentuk, termasuk data dalam sistem komputer, diciptakan atau diterima dan dipelihara oleh organisasi atau perorangan dalam suatu transaksi bisnis atau dibentuk dan dijaga sebagai bukti dari suatu aktivitas." Dengan pengertian rekod di dalam *The Australia Standar* AS 4390-1996 ini, sebagai landasan terciptanya ISO 15489 sebagai standar internasional bidang pencatatan arsip .

Tujuan ISO 15489 adalah sebagai standarisasi dalam pengelolaan arsip untuk menjamin ketersediaan informasi dalam penyelenggaraan kegiatan sebagai bahan akuntabilitas kinerja dan alat bukti yang sah berdasarkan suatu sistem yang memenuhi syarat diantaranya adalah andal, sistematis, utuh, menyeluruh dan sesuai dengan norma, standar, prosedur dan kriteria [7].

2.5.1. *Records Management Processes* (Proses Pengelolaan Catatan)

Ada beberapa hal yang harus dipahami dalam pengelolaan catatan-catatan berdasarkan standar ISO 15489, diantaranya adalah butir 9 mengenai *Records Management Processes* yang terdiri dari [8]:

1. *Capture*,

   proses penangkapan adalah proses penentuan rekod yang harus dibuat atau disimpan dalam sistem dan berlaku baik untuk rekod yang dibuat maupun yang diterima oleh organisasi. Pada tahapan penangkapan ini juga ditentukannya siapa saja yang diperbolehkan mengkases rekod tersebut dan berapa lama retensinya.

2. *Registration*,

   registrasi adalah kegiatan untuk memberikan identifikasi pada sebuah rekod ketika rekod berada dalam sistem seperti judul,

pencipta, deskripsi singkat,waktu penciptaan, dan sebagainya. Selain itu, juga terdapat waktu registrasinya. Berdasarkan ISO 15489:2001 spesifikasi registrasi harus memenuhi ketentuan sebagai berikut:

a. Penanda unik yang diberikan dari sistem
b. Tanggal dan waktu registrasi
c. Judul atau deskripsi singkat
d. Pengarang (perorangan atau badan korporasi),
e. pengirim atau penerima.

3. *Classification*,

   Klasifikasi merupakan suatu proses mengidentifikasi rekod dalam pengaturan aktivitas bisnis atau mengidentifikasi rekod ke dalam kategori menurut pengaturan secara terstruktur berdasarkan logika, metode, dan aturan ynag sesuai dengan prosedur yang ditampilkan dalam sistem klasifikasi secara sistematis. Setiap metadata arsip yang disimpan, dideskripsikan sesuai dengan pedoman yang digunakan oleh instansi tersebut. Sebelum memberkaskan arsip, terlebih dahulu membuat berkas sesuai dengan klasifikasi. Setelah itu, baru kita memberkaskan arsip sesuai dengan berkas yang ada.

4. *Access and security classification*,

   Berdasarkan ISO 15489, pengaksesan adalah hak, kesempatan, dalam arti untuk dapat mencari, menggunakan, dan menemukan kembali informasi di dalam rekod. Keamanan arsip dapat dilakukan dengan cara melakukan pengaturan hak akses. Pengaturan dan pengelolaan tingkat akses pengguna bertujuan untuk mengatur dan mengelola hak akses terhadap arsip sesuai dengan struktur organisasi dari sebuah instansi.

5. *Identification of disposition status*,

   Dalam sistem perekaman berdasarkan ISO 15489 harus ada proses pengidentifikasian status penyusutan dan periode retensi suatu arsip untuk mempermudah pengelola untuk melakukan penyusutan.

6. *Storage and handling,*

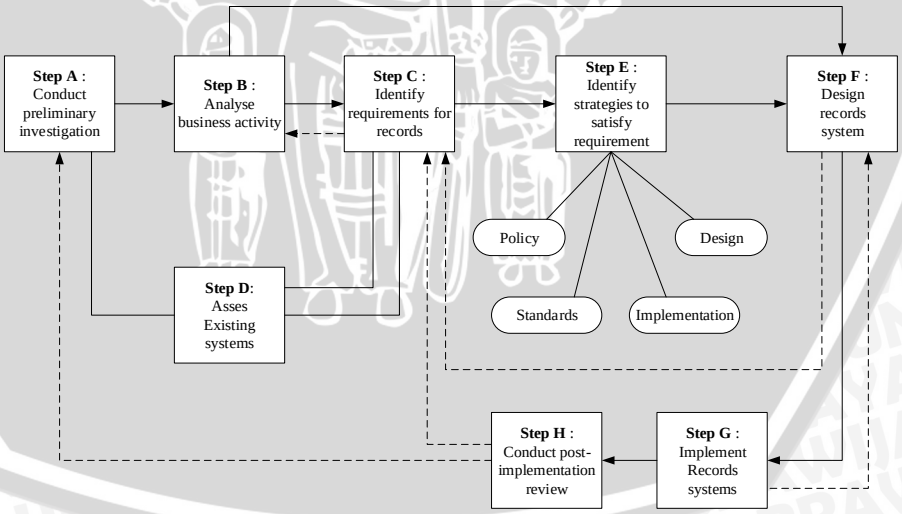
   catatan harus disimpan pada media yang memastikan ketersediaan, keandalan, keaslian dan pelestarian catatan, selama catatan tersebut dibutuhkan.

7. *Use and tracking,*

   Berdasarkan ISO 15489, penggunaan rekod adalah suatu bentuk transaksi manajemen rekod yang dibutuhkan untuk ditangkap oleh suatu sistem dan menjadi bagian dari setiap data yang diperoleh, sedangkan pelacakan adalah menciptakan, menangkap, dan mengolah informasi tentang perpindahan dan penggunaan arsip.

8. *Implementation of disposition.*

   Berdasarkan ISO 15489, penyusutan adalah proses yang berhubungan dengan pengimplementasian retensi arsip, pemusnahan atau keputusan pemindahan arsip yang diarsipkan dalam aturan penyusutan atau perangkat yang lain.



Gambar 2.5 Desain dan Implementasi Sistem Perekaman

(Design and Implementation of Records System(DIRS))

Sumber : [8]

1. ***Step A : Preliminary investigation***

Tujuan dari langkah A adalah memberikan pemahaman kepada organisasi tentang adinistrasi, hukum, bisnis, dan konteks sosial dimana ia beroperasi sehingga dapat mengidentifikasi faktor-faktor utama yang mempengaruhi kebutuhan untuk menciptakan dan penyimpanan arsip.

2. ***Step B : Analysis of Business activity***

Tujuan dari langkah ini adalah untuk mengembangkan model konseptual dari apa yang organisasi lakukan dan bagaimana melakukannya, itu akan menunjukkan bagaiaman catatan berhubungan dengan keperluan bisnis organisasi dan proses bisnisnya. Langkah ini menyediakan alat untuk melakukan dan mengarsipkan analisis bisnis secara sistematis dan membuat penggunaan terbaik dari hasil-hasil proses organisasi dan catatannya. Produk yang berasal dari langkah ini mungkin termasuk :

a. Pengarsipan yang menggambarkan bisnis dan proses bisnis organisasi,
b. Peta proses bisnis organisasi yang menunjukkan titik-titik organisasi di mana catatan diproduksi atau diterima sebagai produk atau aktivitas bisnis.

3. ***Step C : Identification of requirement for Records***

Tujuan dari langkah ini adalah untuk mengidentifikasi kebutuhan organisasi untuk membuat, menerima, dan menyimpan catatan kegiatan usaha, dan mengarsipkan persyaratan dalam bentuk terstruktur dan mudah dipelihara.

4. ***Step D : Assesment of existing system***

Tujuan dari langkah ini adalah untuk survei sistem yang ada di organisasi untuk pencatatan dan informasi lainnya sistem untuk mengukur sejauh mana mereka menangkap dan memelihara catatan dari kegiatan usaha. Ini akan memberikan dasar untuk mengembangkan sistem baru atau mendesain ulang sistem yang ada untuk memenuhi kebutuhan catatan yang telah diidentifikasi dan disepakati dalam langkah sebelumnya.

5. *Step E : Identification of strategies for satisfying records requirements*

Tujuan dari langkah ini adalah untuk menentukan kebijakan, prosedur, standar, peralatan, dan lainnya. Strategi mungkin akan termasuk :

a. Mengadopsi kebijakan dan prosedur,
b. Menyusun standar
c. Merancang komponen sistem baru, atau
d. Menerapkan sistem dengan cara yang memenuhi persyaratan diidentifikasi untuk menjaga dan menyimpan catatan.

6. *Step F : Design of a records system*

Langkah ini melibatkan pengkonversian strategi dan taktik yang dipilih pada langkah E dalam rencana sistem pencatatan yan memenuhi persyaratan yang diidentifikasi dan diarsipkan.

7. *Step G : Implementation of a records system*

Tujuan dari langkah G adalah untuk mengidentifikasi secara sistematis dan menempatkan campuran yang tepat dari strategi dan pengimplementasian rencana yang dirancang pada Langkah F. Rencana itu memberikan gambaran tentang bagaimana berbagai komponen sistem (proses,prosedur, orang dan teknologi) harus cocok satu sama lain.

8. *Step H : Post-implementation review*

Tujuan dari langkah H adalah untuk mengukur efektivitas sistem catatan, untuk mengevaluasi proses pengembangan sistem, sehingga kekurangan bisa diperbaiki. Langkah H melibatkan proses analisis apakah sistem pencatatan telah dibuat dan disusun sesuai dengan kebutuhan kegiatan bisnis dan tepat saling terkait dengan kebutuhan proses bisnis antar bagian.

2.6. Perangkat Lunak

2.6.1. Definisi Perangkat Lunak

Menurut Roger S. Pressman definisi perangkat lunak (*software*) adalah

[11] :

1. Instruksi-instruksi yang bila dieksekusi akan memberikan fungsi dan unjuk kerja yang diinginkan.
2. Struktur data yang memungkinkan program untuk memanipulasi informasi
3. Arsip-arsip yang menjelaskan pengoperasian dan penggunaan program.

2.6.2. Karakteristik Perangkat Lunak

Ketika perangkat lunak *(software*) dibuat, proses kreatif manusia(analisis, desain, konstruksi, pengujian) pada akhirnya diteruskan ke dalam bentuk fisik. Perangkat lunak (*software)* memiliki karakteristik yang berbeda dengan perangkat. Berikut adalah karakteristik perangkat lunak (software) [11]:

1. Biaya (*cost*) perangkat lunak terpusat pada perekayasaan. Jadi proyek perangkat lunak tidak dapat dijalankan seperti proses pemanufakturan.
2. Perangkat lunak tidak habis terpakai.

   Tingkat kegagalan perangkat keras (*hardware*) yang tinggi pada awalnya, sebagai akibat dari kesalahan perancangan atau kesalahan pembuatan di pabrik. Setelah diperbaiki, tingkat kegagalan menurun sampai ke suatu tingkat yang stabil untuk beberapa periode waktu. Seiring dengan berlalunya waktu, tingkat kegagalan meningkat akibat akumulasi pengaruh debu, getaran, suhu tinggi, dan banyak masalah lingkungan lainnya terhadap komponen perangkat keras (*hardware)*. Dengan kata lain, perangkat keras mulai habis terpakai.Sedangkan pada perangkat lunak, tingkat kegagalan yang tinggi, adalah sebagai akibat yang tidak diperkirakan. Setelah diperbaiki, tingkat kegagalan tersebut akan menurun sampai ke tingkat yang stabil. Namun, selama masa itu, perangkat lunak akan mengalami perubahan-perubahan (selama *maintenance)*. Dimana pada perubahan-perubahan itu, akan ditemukan kesalahan-kesalahan

baru sehingga tingkat kegagalan akan meningkat lagi. Dan lama kelamaan, perubahan-perubahan yang dilakukan dan tingkat kegagalan yang makin meningkat tersebut, membuat perangkat lunak menjadi usang.

3. Kebanyakan perangkat lunak dibuat sesuai kebutuhan, tidak dirakit dari komponen-komponen yang ada. Setelah perangkat keras dirancang, komponen-komponennya dapat dipilih dari katalog. Sedangkan, perangkat lunak tidak mempunyai katalog untuk komponen-komponennya.

2.7. Rekayasa Perangkat Lunak

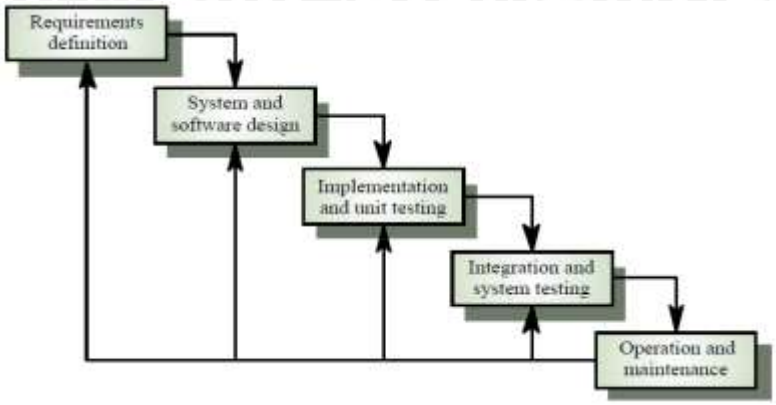
Dalam buku karangan Roger S. Pressman**,** pengertian awal dari rekayasa perangkat lunak (*Software Engineering*) diajukan oleh Fritz Bauer. Adapun definisi rekayasa perangkat lunak menurut Fritz Bauer adalah penetapan dan penggunaan prinsip-prinsip rekayasa dalam rangka mendapatkan perangkat lunak yang ekonomis, yaitu perangkat lunak yang dapat diandalkan dan bekerja secara efisien pada mesin.

Institute of Electrical Electronics Engineers (*IEEE*) menemukan definisi yang lebih komprehensif tentang rekayasa perangkat lunak [11] :

1. Aplikasi dari sebuah pendekatan yang sistematis, disiplin dan terukur terhadap pengembangan, pengoperasian, dan pemeliharaan (*maintenance*) perangkat lunak.
2. Studi terhadap pendekatan-pendekatan seperti pada butir pertama.

2.8 System Development Life-Cycle (Waterfall Model)

Menurut Roger S. Pressman, *classic life cycle* atau *waterfall* model, menyarankan sebuah sistematika pendekatan bertahap untuk pengembangan perangkat lunak [12]. Adapun tahapan-tahapan waterfall model ditunjukkan gambar 2.6



Gambar 2.6 Waterfall Model

Sumber : [12]

1. *Requirement* (analisis kebutuhan).

   Dalam langakah ini merupakan analisa terhadap kebutuhan sistem. Pengumpulan data dalam tahap ini bisa melakukan sebuah penelitian, wawancara atau studi literatur. Seseorang system analisis akan menggali informasi sebanyak-banyaknya dari user sehingga akan tercipta sebuah sistem komputer yang bisa melakukan tugas-tugas yang diinginkan oleh user tersebut. Tahapan ini akan menghasilkan arsip *user requirement* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan user dalam pembuatan sistem. Arsip inilah yang akan menjadi acuan sistem analisis untuk menterjemahkan kedalam bahasa pemrograman.

2. *Design System*

   Proses *design* akan menterjemahkan syarat kebutuhan kesebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat koding. Proses ini berfokus pada : struktur data, arsitektur perangkat lunak, representasi interface, dan detail (algoritma) prosedural. Tahapan ini akan menghasilkan arsip yang disebut software requirement. Arsip inilah yang akan digunakan programmer untuk melakukan aktivitas pembuatan sistemnya.

3. *Implementation & Unit Testing* (Implementasi dan Pengujian Unit)
   Coding merupakan penerjemahan *design* dalam bahasa yang bisa dikenali oleh komputer. Dilakukan oleh programmer yang akan meterjemahkan transaksi yang diminta oleh *user*. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu sistem. Dalam artian penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan testing terhadap unit sistem yang telah dibuat.
4. *Integration & System Testing* (Pengujian Secara Keseluruhan)
   Tahapan ini bisa dikatakan final dalam pembuatan sebuah sistem. Setelah melakukan analisa, *design*, pengkodean dan pengintegrasian unit maka akan dilakukan pengujian sistem secara keseluruhan. Pengujian ini dimaksudkan untuk menemukan kesalahan-kesalahan pada sistem tersebut agar bisa diperbaiki.
5. *Operation & Maintenance* (Pemeliharaan).
   Perangkat lunak yang sudah disampaikan kepada pelanggan pasti akan mengalami perubahan. Perubahan tersebut bisa karena mengalami kesalahan karena perangkat lunak harus menyesuaikan dengan lingkungan (periperal atau system operasi baru) baru, atau karena pelanggan membutuhkan perkembangan fungsional.

2.9 *Unified Modellling Language (UML)*

Menurut Sommerville, pemodelan sistem adalah proses mengembangkan *abstract models* dari suatu sistem, dengan masing – masing merepresentasikan perspektif yang berbeda dari sistem tersebut. Pemodelan sistem umumnya merepresentasikan sistem menggunakan beberapa jenis notasi grafis, yang sekarang hampir selalu didasarkan pada notasi dalam *Unified Modelling Language* (UML).

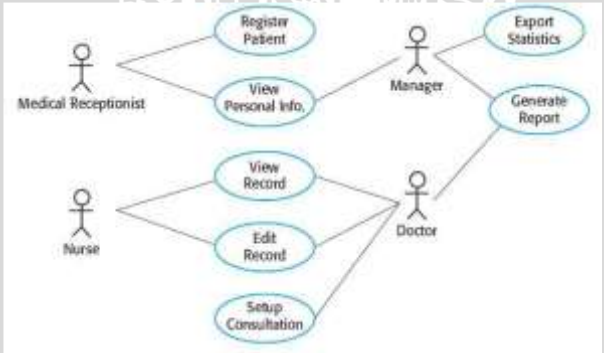
Menurut Sommerville, UML telah menjadi bahasa pemodelan standar untuk pemodelan berorientasi objek. UML memiliki tipe – tipe diagram dan juga mendukung penciptaan berbagai jenis model sistem [15].

2.9.1. *Use Case Diagram*

Pemodelan *Use Case* pada awalnya dikembangkan oleh Jacobson pada 1990-an dan dimasukkan dalam rilis pertama UML. *Use Case* secara luas digunakan untuk mendukung persyaratan dari *elicitation*. *Use Case* dapat diambil sebagai skenario sederhana yang menggambarkan apa yang pengguna harapkan dari suatu sistem.

Setiap *Use Case* merepresentasikan tugas terpisah yang melibatkan interaksi eksternal dengan sistem. *Use Case* mengidentifikasi interaksi individu antara sistem dan pengguna atau sistem lain. Setiap *Use Case* harus diarsipkan dengan deksripsi tekstual, yang kemudian dapat dikaitkan dengan model lain dalam UML yang akan membangun skenario dengan lebih terperinci [16]. Gambar 2.7 berikut merupakan contoh *Use Case* diagram konsultasi :

Aturan konsultasi mengijinkan dua atau lebih dokter, bekerja di kantor berbeda, melihat data tersimpan yang sama di waktu bersamaan. Satu dokter berinisiatif untuk membuka konsultasi dengan memilih orang yang terlibat dari menu pilihan dokter yang *on-line*. Data pasien kemudian muncul pada layar mereka namun hanya dokter pengusul yang dapat mengganti data. Sebagai tambahan, terdapat layar chat yang dibuat untuk membantu [15].



***Gambar 2.7 Gambar Use Case* Diagram**

Sumber : [15]

2.9.2. *Activity Diagram*

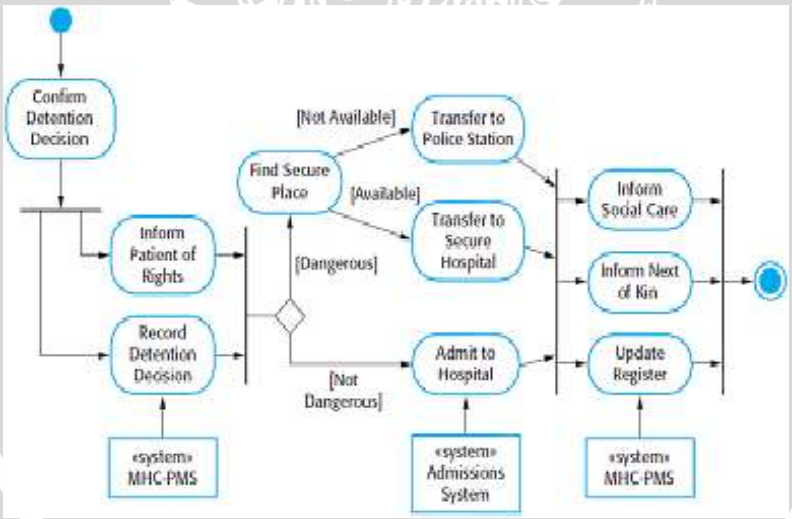
Menurut Sommerville *Activity Diagram* menunjukkan aktivitas yang terlibat di dalam proses atau dalam pengolahan data. Diagram *activity* seperti diagram state, merupakan diagram yang digunakan untuk memahami alur kerja

dari objek atau komponen yang dilakukan. Diagram *activity* dapat digunakan untuk mevisualisasikan interelasi dan interaksi antara *Use Case* yang berbeda, serta sering digunakan untuk mengasosiasikan dengan *class* yang berbeda. Kekuatan *Activity Diagram* adalah merepresentasikan *concurrent activity*.

Gambar 2.8 adalah sebuah *Activity Diagram* yang ditujukan untuk menunjukkkan aktifitas yang membuat sebuah proses sistem dan aliran kontrol dari satu aktifitas ke aktifitas yang lain.

Untuk memulai proses dimulai dengan *filled circle*, dan diakhiri dengan lambang lingkaran dalam lingkaran. Persegi dengan segi tumpul menggambarkan aktifitas.

Dalam UML *Activity Diagram*, garis panah menggambarkan alur kerja dari aktifitas satu ke aktifitas yang lain. Garis tebal digunakan untuk mengindikasi koordinasi aktifitaas. Saat alur mulai lebih dari satu aktifitas mengarah ke garis tebal maka semua aktifitas harus dilakukan sebelum melanjutkan ke tahap berikutnya. Saat alur dari garis tebal menuju aktifitas tertentu, maka aktifitas tersebut boleh dikerjakan secara paralel [16].



Gambar 2.8 Gambar *Activity Diagram*

Sumber : [15]

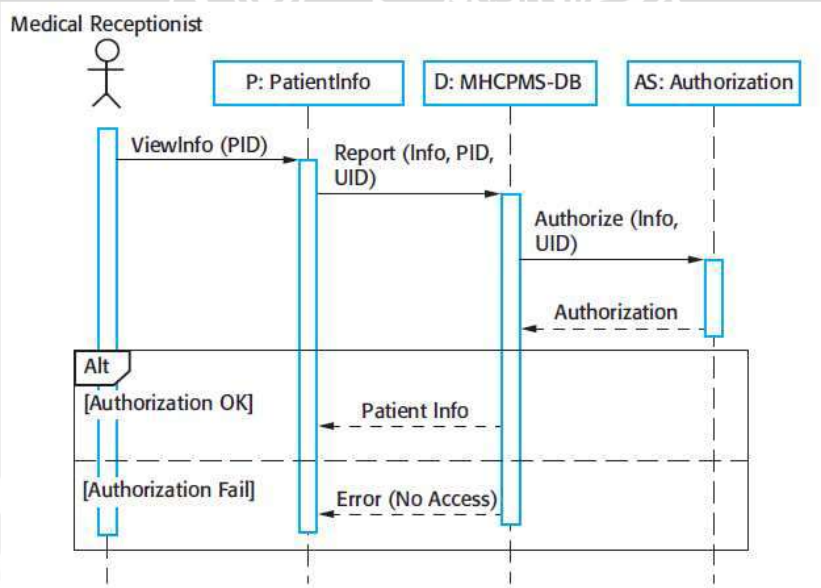
2.9.3. *Sequence Diagram*

Sequence Diagram dalam UML terutama digunakan untuk memodelkan interaksi antara aktor dan objek dalam sistem dan interaksi antara objek itu

sendiri. UML memiliki sintaks yang sangat beragam untuk *Sequence Diagram*, yang memungkinkan perbedaan tipe interaksi yang akan dimodelkan [15].

Sesuai namanya, *Sequence Diagram* menunjukkan urutan interaksi yang terjadi selama *Use Case instance*. Diagram *sequence* dapat digunakan untuk menggambarkan skenario atau rangkaian langkah – langkah yang telah dilakukan sebagai response dari sebuah *event* untuk menghasilkan output tertentu. *Sequence Diagram* menunjukkan interaksi antara aktor dan sistem dan juga antara komponen sistem [15].

Objek dan aktor yang terlibat, tercantum di bagian atas diagram dengan garis titik – titik yang ditarik secara vertikal. Interaksi antar objek ditunjukkan dengan panah yang terhubung. Persegi panjang pada garis putus – putus menunjukkan jalur *lifeline* dari objek yang bersangkutan. Anda membaca *sequence* interaksi dari atas ke bawah. Penjelasan pada tanda panah menunjukkan panggilan ke objek, parameter mereka, dan *return value*.



Gambar 2.9 Gambar *Sequence* Diagram

Sumber : [15]

Dari gambar 2.9 di atas, dapat dijelaskan bahwa :

1. Resepsionis medis memicu metode *View Info* dalam *instance* P dari kelas objek PatientInfo, penyediaan identifier pasien, PID. P adalah objek antarmuka pengguna, yang ditampilkan sebagai bentuk informasi pasien.
2. Instance P memanggil database untuk mengembalikkan informasi yang dibutuhkan, penyediaan resepsionis identifier memungkinkan pemeriksaan keamanan (pada tahap ini, tidak perduli darimana UID berasal).
3. Pemeriksaan dengan sistem autorisasi kepada pengguna yang memiliki wewenang.
4. Jika diizinkan, informasi pasien dikembalikan. Jika autorisasi gagal, maka pesan
   kesalahan akan dikembalikan.

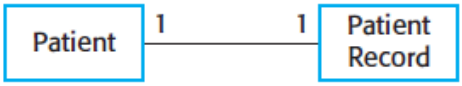
2.9.4. *Class Diagram*

Menurut Sommerville, *class* diagram menunjukkan kelas objek dalam sistem dan asosiasi antara kelas–kelas. Diagram kelas digunakan ketika mengembangkan model sistem berorientasi objek untuk menunjukkan kelas–kelas dalam sistem dan hubungan antara kelas–kelas tersebut. Sebuah kelas objek dapat dianggap sebagai definisi umum dari sebuah objek sistem. Sebuah asosiasi adalah link antara kelas yang mengindikasikan bahwa ada relasi antara kelas – kelas tersebut. Akibatnya, masing – masing kelas mungkin harus memiliki pengetahuan mengenai kelas yang berkaitan.

Ketika seorang developer sedang mengembangkan model pada tahap awal proses rekayasa perangkat lunak, objek merepresentasikan sesuatu di dunia nyata, seperti pasien, resep, dokter, dan lain – lain. Sebagai sebuah implementasi yang dibangun, biasanya seorang developer harus menentukan objek implementasi tambahan yang digunakan untuk menyediakan fungsionalitas sistem yangdibutuhkan.

Kelas diagram pada UML dapat diekspresikan dengan tingkat rincian yang berbeda. Ketika seorang developer mengembangkan sebuah model, tahap pertama biasanya melihat *world*, mengidentifikasi objek esensial, dan merepresentasikan itu sebagai *class*. Anda juga dapat mencatat adanya asosiasi

dengan menggambar garis antara kelas. Sebagai contoh gambar 2.10 adalah gambar sederhana yang menampilkan dua kelas [15].

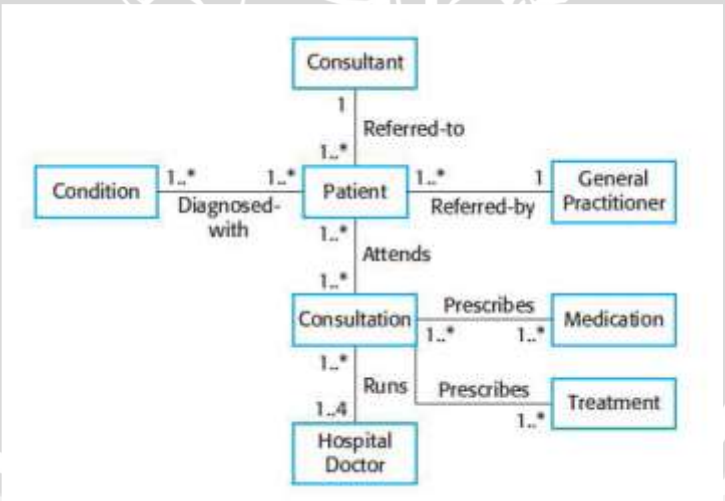


Gambar 2.10 Gambar Asosiasi Kelas Sederhana

Sumber : [15]

Pada contoh di atas dianotasikan dengan 1, yang berarti ada hubungan 1:1 diantara objek dalam kelas–kelas tersebut. Artinya, setiap pasien memiliki satu record dan setiap *record* menangani informasi hanya satu pasien.

Gambar 2.11 mengembangkan jenis *class* diagram untuk menunjukkan bahwa objek dari kelas pasien juga terlibat dalam hubungan dengan sejumlah kelas lainnya. UML juga memungkinkan peran objek untuk berpartisipasi dalam asosiasi yang ditentukan. Ketika menampilkan asosiasi antar kelas, akan lebih mudah untuk merepresentasikan kelas tersebut dengan sesederhana mungkin. Untuk mendefinisikan secara lebih rinci, anda menambahkan informasi mengenai atribut mereka (karateristik dari objek) dan operasi (hal yang dapat anda request dari objek).



Gambar 2.11 Gambar Class dan Asosiasi

Sumber : [15]

Sebagai contoh sebuah objek pasien akan memiliki atribut address dan juga mencakup operasi yang disebut change address, yang dipanggil ketika pasien menunjukkan bahwa mereka telah pindah dari satu address ke address

lain. Dalam UML, anda menunjukkan atribut dan operasi dengan mengembangkan persegi panjang yang merepresentasikan kelas. Hal tersebut diilustrasikan dalam gambar 2.12 di bawah ini:

Gambar 2.12 Gambar *Consultation Class*

Sumber : [15]

Pada gambar 2.12 dapat dirumuskan bahwa :

1. Nama dari *class* object berada di bagian atas.
2. Atribut *class* berada di bagian tengah. Hal tersebut harus juga menyertakan nama atribut, dan pilihan tipenya.
3. The operation (dipanggil dalam *methods* pada Java dan OOP lainnya), asosiasi dengan kelas objek yang letaknya di bawah persegi panjang tersebut.

Atribut dan metoda dapat memiliki salah satu sifat berikut ini :

1. *Private*, tidak dapat dipanggil dari luar *class* yang bersangkutan.
2. *Protected*, hanya dipanggil oleh *class* yang bersangkutan dan anak–anak turunannya.
3. *Public*, dapat dipanggil oleh siapa saja.

Hubungan antar *class* :

1. Asosiasi yaitu hubungan statis antar *class*. Umumnya *class* yang menggambarkan atribut berupa *class* lain, atau *class* yang harus mengetahui eksistensi *class* lain. Panah menunjukan arah *query* antar *class*.

2. Agregasi yaitu hubungan yang menyatakan bagian.
3. Pewarisan, yaitu hubungan hirarki antar *class*. *Class* dapat diturunkan dari *class* lain dan dapat mewarisi atribut dan metode *class* asalnya dan dapat menambahkan fungsionalitas baru, sehingga ia disebut anak dari *class* yang diwarisinya. Kebalikan dari pewarisan adalah generalisasi.
4. Hubungan dinamis, yaitu rangkaian pesan yang di-passing antara satu *class* kepada *class* lainnya. Hubungan dinamis dapat digambarkan dengan menggunakan diagram *sequence* [15].

2.10 Teori *Normalisasi*

Menurut Connolly dan Beg, tiga bentuk normalisasi pada awalnya yang diusulkan disebut *First Normal Form*(1NF), *Second Normal Form* (2NF), *Third Normal Form* (3NF) [CON-10]. Namun kemudian muncul bentuk-bentuk normal yang baru yaitu *Boyce-Codd Normal Form* (BCNF), *Fourth Normal Form* (4NF), dan *Fifth Normal Form* (5NF) [5].

Proses normalisasi :

1. *Unnormalized Form* (UNF) Menurut Connolly dan Beg, suatu tabel yang terdiri dari satu atau lebih kelompok yang berulang (*Repeating Group*). *Repeating Group* adalah sebuah atribut atau himpunan atribut di dalam tabel yang memiliki lebih dari satu nilai (*Multiple Value*) untuk sebuah *Primary Key* tabel tersebut.
2. *First Normal Form* (1NF) adalah sebuah relasi dimana titik temu antara baris dan kolomnya mengandung hanya satu nilai. Ada dua pendekatan umum untuk menghilangkan *repeating group* dari tabel yang tidak normal, yaitu :
   a. Menempatkan data berulang bersama salinan atribut kunci pada relasi yang berulang.
   b. Menempatkan data berulang bersama salinan atribut kunci pada relasi yang terpisah. Sebuah primary key diindentifikasi ke dalam relasi.

3. *Second Normal Form* (2NF) Menurut Connolly dan Beg, sebuah relasi yang berada pada 1NF dan setiap atribut yang bukan *primary key* berfungsi secara penuh bergantung pada *primary key*-nya.
4. *Third Normal Form* (3NF). Sebuah relasi yang berada pada 1NF dan 2NF, dan tidak ada atribut yang bukan *primary key* yang secara langsung bergantung kepada *primary key*-nya.
5. *Boyce-Codd Normal Form* (BCNF). BCNF merupakan perluasan dari 3NF. Menurut Connolly dan Beg, BCNF berdasarkan pada *functional dependencies* yang mempertimbangkan semua *candidate key* dalam sebuah relasi. Sebuah relasi dalam BCNF jika dan hanya jika setiap *determinant* adalah sebuah *candidate key*.
6. *Fourth Normal Form* (4NF), sebuah relasi yang jika dan hanya jika setiap *nontrivial multivalued dependency* A >> B, dimana A adalah *candidate key* dari relasi.
7. *Fifth Normal Form* (5NF), sebuah relasi jika dan hanya jika setiap *join dependency* (R1, R2, ... Rn) di dalam relasi R, setiap proyeksi mencakup *candidate key* dari relasi asli.

2.11 MVC *(Model View Controller)*

Model View Controller (MVC) adalah sebuah metode untuk membuat sebuah aplikasi dengan memisahkan data dari tampilan dan cara memprosesnya. MVC diperkenalkan pertama kali oleh para peneliti di XEROC PARAC yang bekerja untuk pembuatan bahasa pemrograman *Smalltalk* pada tahun 1970-1980. Metode MVC membagi aplikasi menjadi tiga bagian yaitu [6]:

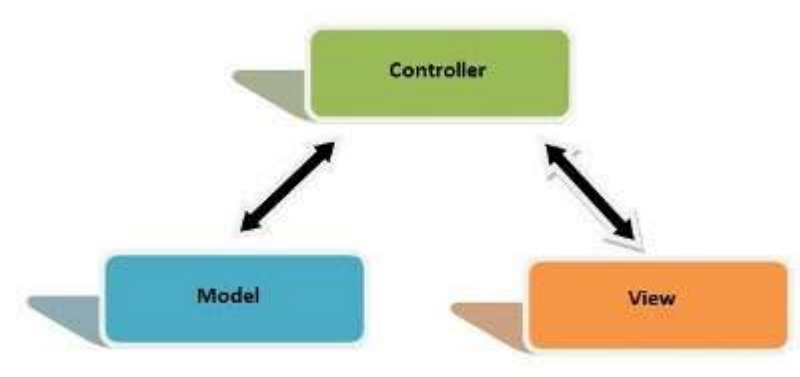
1. Model

   Berfungsi sebagai pengelola perilaku dan data pada domain aplikasi, melakukan tanggapan terhadap permintaan informasi dan merespons instruksi untuk merubah suatu kondisi (state).
2. View

   Menerjemahkan informasi yang berasal dai model ke dalam sebuah bentuk yang sesuai untuk berinteraksi dengan user. Biasanya berupa satu atau lebih elemen antarmuka user.

3. Controller

   Menerima masukan dari user dan memicu respons dengan membuat pemanggilan ke objek-objek yang ada di dalam modul model.



Gambar 2.13 Gambar Konsep MVC

Sumber : [6]

Kelebihan-kelebihan metode MVC adalah sebagai berikut [6] :

1. Bagian model memberikan penempatan detil data yang terpisah dan disebar di dalam keseluruhan aplikasi sehingga meningkatkan fleksibilitas dan pemeliharaan aplikasi.
2. Pemisahan model juga membuat objek model dapat digunakan oleh aplikasi lain dengan kebutuhan yang sama (*reuse*).
3. Pemisahan view memudahakan perakitan/integrasi dengan komponen aplikasi lainnya tanpa harus memperhatikan detil proses. Desainer hanya berkonsentrasi pada bentuk dan tampilan antarmuka pemakai (*user interface*).
4. Penggunaan Controller memungkinkan untuk perubahan proses tanpa harus mengganggu antarmuka pemakai. Detil proses disembunyikan oleh Controller sehingga tidak mengganggu presentasi ke user maupun pengelolaan data/informasi (manajemen database).

Di dalam aplikasi berbasis web dikenal beberapa jenis MVC , yaitu [6]:

1. Server Side MVC dimana seluruh proses dilakukan pada sisi server.

2. Campuran *Server Side* dan *Client Side* MVC, dimana proses dilakukan di sisi client dan sisi server, bisa menggunakan atau tidak menggunakan model dalam koneksi ke server dan biasanya memiliki kompleksitas yang tinggi.
3. *Rich Internet Application* (RIA), disebut juga *Fat Client*, merupakan aplikasi web yang memiliki kemampuan dan fungsi yang mirip aplikasi *dekstop*, memiliki bagian data sendiri (MVC tersendiri) dan hanya bagian model yang ada di bagian server.

2.12 Pengujian Perangkat Lunak

2.12.1. *Functional Testing*

Pengujian fungsional (*Functional Testing*) adalah pengujian yang dilakukan berdasarkan fungsionalitas dari program atau sistem yang akan diuji. Oleh karena itu penguji membutuhkan informasi mengenai data input dan output yang diamati, tetapi tidak perlu mengetahui bagaimana program atau sistem tersebut melakukan eksekusi. Penguji hanya berfokus pada pengujian fungsionalitas dari sistem terhadap spesifikasi. Secara tradisional, pengujian fungsional dilakukan dilaksanakan oleh tim penguji atau independen dari pengembang.

Pengujian fungsional meliputi sebarapa baik sistem melaksanakan fungsinya, termasuk perintah-perintah pengguna, manipulasi data, pencarian dan proses bisnis, pengguna layar, dan integrasi. Pengujian fungsional juga meliputi permukaan yang jelas dari jenis fungsi-fungsi, serta operasi *back-end* (seperti, keamanan dan bagaimana meningkatkan sistem) [13].

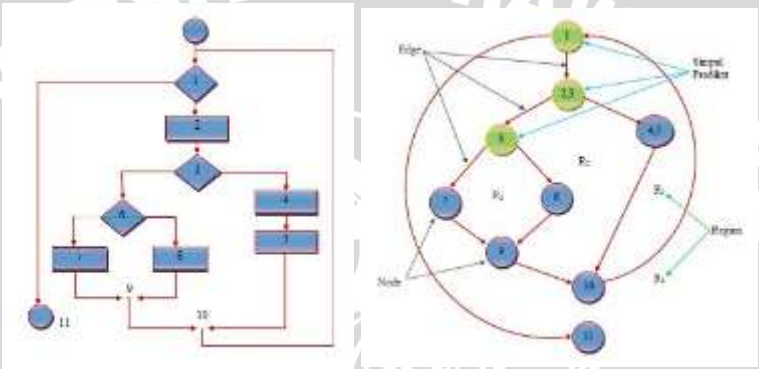
2.12.2. *White-Box Testing (pengujian kotak putih)*

Pengujian white-box (*glass box*), adalah metode desain *test case* yang menggunakan struktur kontrol desain prosedural untuk memperoleh *test case*. Dengan menggunakan metode pengujian white-box, perekayasa sistem dapat melakukan test case untuk memberikan jaminan bahwa :

1. Semua jalur independen pada suatu modul ditelusuri minimal 1 kali
2. Semua jalur keputusan logis *True/False* dilalui

3. Semua *loop* dieksekusi pada batas yang tercantum dan batas operasionalnya
4. Struktur data internal digunakan agar validitas terjamin

Pengujian *white-box* bisa dilakukan dengan pengujian basis path, metode ini merupakan salah satu teknik pengujian struktur kontrol untuk menjamin semua statemen dalam setiap jalur independen program dieksekusi minimal 1 kali. Perhitungan jalur independen dapat dilakukan melalui matrix Cyclomatic complexity. Sebelum menghitung nilai *Cyclomatic complexity*, harus diterjemahkan desain prosedural ke grafik alir, kemudian dibuat flow graphnya, seperti pada gambar 2.14.



Gambar 2.14 *Flowchart* dan *Flow Graph* Jalur Independen

Sumber : [3]

Dari gambar flowgraph di atas didapat :

a. Path 1 = 1 – 11
b. Path 2 = 1 – 2 – 3 – 4 – 5 – 10 – 1 – 11
c. Path 3 = 1 – 2 – 3 – 6 – 8 – 9 – 10 – 1 – 11
d. Path 4 = 1 – 2 – 3 – 6 – 7 – 9 – 10 – 1 – 11

Path 1,2,3,4 yang telah didefinisikan diatas merupakan basis set untuk diagram alir. *Cyclomatic complexity* digunakan untuk mencari jumlah path dalam satu flowgraph. Dapat dipergunakan rumusan sebagai berikut :

1. Jumlah region grafik alir sesuai dengan *cyclomatic complexity*.
2. *Cyclomatic complexity* V(G) untuk grafik alir dihitung dengan rumus:

$$V(G) = E - N + 2 \quad \ldots\ldots\ldots\ldots (1)$$

Dimana:

E = jumlah edge pada grafik alir

N = jumlah node pada grafik alir

Dari Gambar di atas dapat dihitung *cyclomatic complexity*:

1. Flowgraph mempunyai 4 region
2. V(G) = 11 edge – 9 node + 2 = 4

Jadi *cyclomatic complexity* untuk flowgraph adalah 4

Biasanya batas maksimum nilai yang dapat diterima *Cyclomatic complexity* adalah 10. Nilai-nilai lain, seperti 15 atau 20 juga telah diusulkan. Terlepas dari batas yang tepat, jika *Cyclomatic complexity* melebihi 20, hal tersebut harus dipertimbangkan. Prosedur dengan kompleksitas yang tinggi *cyclomatic* harus disederhanakan atau dipecah menjadi beberapa prosedur yang lebih kecil [3].

2.12.3. *Black-Box Testing*

Yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji harus dibuat dengan kasus benar dan kasus salah [14].

2.12.4. Pengujian Kesesuaian Sistem Terhadap Standar ISO

1. *Checklist* ISO 15489

Pedoman ini dikembangkan untuk membantu organisasi dalam membangun sebuah program audit yang baik, yang akan menunjukkan dan memberikan jaminan kesesuaian dengan standar praktek yang baik yang ditetapkan untuk manajemen kearsipan. Standar ini mengizinkan cara yang berbeda untuk memperoleh jaminan kesesuaian dalam menyediakan lampiran, jumlah *checklist* yang digunakan sebagai ukuran, dan tes kesesuaian dalam komponen utama

(siklus hidup) manajemen arsip yang mencakup penciptaan, retensi, pemeliharaan, dan pemusnahan .

Checklist meliputi petunjuk untuk standar praktek yang baik yang diperlukan sebagai pedoman, termasuk : Kode Etik NHS *Management Records*, jaminan pemeliharaan arsip NHS, ISO 15489 – standar pencatatan internasional dan informasi elemen manajemen siklus hidup dari informasi *NHS Governance Toolkit* [2].

Tabel 2.2 Contoh *Form Checklist* ISO 15489

Checklist 3.1 : Creation of Records				
Objective: Records are created as relevant to the organisations clinical and corporate activities and captured into the appropriate record keeping systems upon creation or receipt.				
Good Practice Measure	*Evidence*	*Compliance Yes/No/Partial*	*Action Required Yes/No*	*Follow-up Date*
1. There is guidance on what constitutes a record and what should be done to safeguard it and make it accessible via a record keeping system within each department/ Guidance should include: Naming conventions and metadata requirements (i.e. title, subject, name of creator, date created, locality etc.)				
2. There is specific provision within the organisation's guidance for the capture of electronic information (e.g. e-mails)				
3. The organisation has established a record keeping system (e.g. an electronic				

record management system) to manage its records.				
4. The record keeping or record management system records the physical location of each record set.				

Sumber : [2]

Tabel 2.3 ***Example of an Audit Outcomes Report***

Title :	*Records Management Compliance Audit: Archiving, Disposal and Destruction of Records*
Location / Record Type:	*Estates Department*
Distribution of Report:	*Director of Estates* *Records Manager*
Objectives:	*To ensure that archiving, disposal and destruction of the Trust's estates records is undertaken in accordance with established procedures and in compliance with the NHS Records Management Code of Practice.*
Audit Scope & Methodology:	*The audit was carried out using a checklist of good practice and through examination, on a test basis of records held in estates. These included:* *List record types examined. E.g.* *a) Destruction register* *b) Maintenance schedules*
Sample Selection:	*100% of records held for each type*
Summary Findings:	*The Estates dept has comprehensive procedures for archiving, disposal and destruction of records that*

	comply with good practice guidelines. These, however, need to be updated to reflect retention schedules recently approved by the Board. *The development of a local records inventory has commenced which, amongst other features, will allow minimum retention periods and disposal details to be entered.* *Destruction schedules have only been introduced in the last year and indicate that only records of a relatively minor or subsidiary nature have been destroyed in the year. The method of destruction was not recorded. A number of key records in the sample have been retained for longer than the minimum periods recommended in the NHS Records Management Code of Practice (Part 2) and in several cases, are no longer required. These are to be scheduled and approved for confidential destruction.*
Appendices:	*Checklist of compliance and action* *Supplementary working papers*

Sumber : [2]

2. Pengujian Kualitas ISO

Pengujian kualitas ISO 9126 ini terdiri dari dua bagian yaitu tingkat kualitas masing - masing aspek berdasarkan empat karakteristik ISO 9126. Dari 5 responden yang mengisi kuesioner untuk pengujian kualitas perangkat lunak

prototype Dashboard Information System. Tanggapan Responden terhadap tingkat kualitas prototipe *Dashboard Information System* berdasarkan jawaban responden terhadap indikator kualitas software menurut ISO 9126, dapat diukur dengan menggunakan rumus [6] :

$$\%\ \boldsymbol{Skor\ Aktual} = \frac{\boldsymbol{Skor\ Aktual}}{\boldsymbol{Skor\ Ideal}} \times \mathbf{100}\%$$

Keterangan :

1) Skor aktual adalah jawaban seluruh responden atas kuesioner yang telah diajukan
2) Skor ideal adalah nilai tertinggi atau semua responden diasumsikan memilih jawaban dengan skor tertinggi.

Selanjutnya hasil tersebut diolah dan dihitung dengan kriteria hasil tersebut diolah dan hitung dengan kreteria yang telah ditetapkan dalam rancangan penelitian, yaitu :

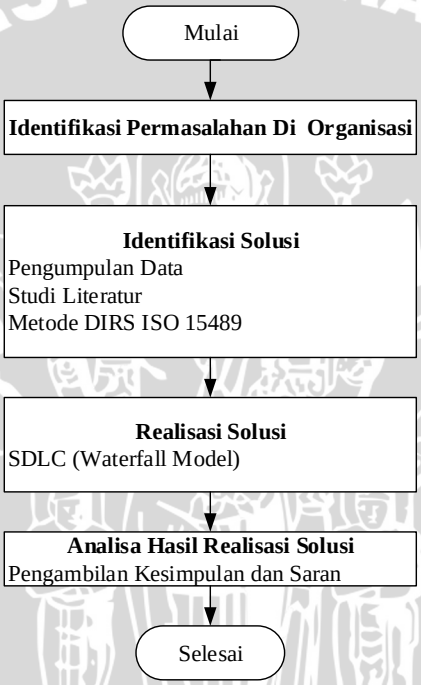
Tabel 2.2 Kriteria Presentase Tanggapan Responden terhadap Skor Ideal

% Jumlah Skor	Kriteria
20,00 % – 36,00 %	Tidak Baik
36,01 % – 52,00 %	Kurang Baik
52,01 % – 68,00 %	Cukup
68,01 % – 84,00 %	Baik
84,01 % – 100 %	Sangat Baik

Sumber : [6]

BAB III
METODOLOGI PENELITIAN

Pada bab ini dijelaskan metodologi yang akan ditempuh dalam penelitian. Langkah-langkah yang dilakukan dalam penyusunan laporan penelitian ini meliputi identifikasi permasalahan, identifikasi solusi, realisasi solusi, dan analisa hasil realisasi solusi. Untuk lebih memudahkan dalam pengerjaan aplikasi maka dibuat sebuah alur kerja yang dapat direpresentasikan dalam *flowchart* pada gambar 3.1



Gambar 3.1 Diagram Blok Metode Penelitian

3.1. Identifikasi Permasalahan

Merupakan proses untuk mendefinisikan masalah-masalah yang sering muncul di organisasi. Proses ini dilakukan dengan cara observasi dan wawancara dengan pegawai di UPT Laboratorium Kesehatan Hewan Malang yang mempunyai kewenangan untuk mengelola arsip, yaitu Kepala UPT, Kepala Sub Bagian Tata Usaha, Kepala Bagian Keswan, dan Kepala Bagian Kesmavet

3.2. Identifikasi Solusi

Proses identifikasi solusi m3elibatkan kegiatan studi literatur dan pengumpulan data sehingga diperoleh sebuah solusi untuk mengatasi permasalahan di organisasi.

Studi literatur dilakukan dengan pengumpulan serta pemahaman materi dan informasi yang digunakan sebagai dasar pembuatan sistem manajemen arsip elektronik. Literatur yang diperlukan dalam penyusunan skripsi meliputi :

1. ISO 15489 : 2001
2. Perangkat Lunak
3. Rekayasa Perangkat Lunak
4. *System Development Life-Cycle* (*Waterfall Model*)
5. *Unified Modelling Language* (UML)
6. Teori Normalisasi
7. *Model-View-Controller* (MVC)
8. Pengujian Perangkat Lunak

Metode yang digunakan untuk proses pengumpulan data dalam penelitian ini adalah sebagai berikut:

a. Teknik Observasi

   Dalam penelitian ini peneliti menggunakan teknik observasi Non Partisipan, dimana peneliti akan mengamati proses bisnis alur arsip di UPT Laboratorium Kesehatan Hewan Malang yang melibatkan pegawai UPT.

b. Teknik Wawancara

   Wawancara yang dilakukan kepada pegawai UPT untuk mendapatkan informasi mengenai alur sistem manajemen arsip yang sedang berjalan, serta kebutuhan perangkat lunak dan perancangan sistem yang diharapkan.

c. Teknik Pengarsipan

Data yang akan diambil melalui teknik ini yaitu data-data yang berhubungan dengan alur proses manajemen arsip di UPT Laboratorium Kesehatan Hewan Malang seperti catatan surat, agenda kegiatan, disposisi surat, dan sebagainya.

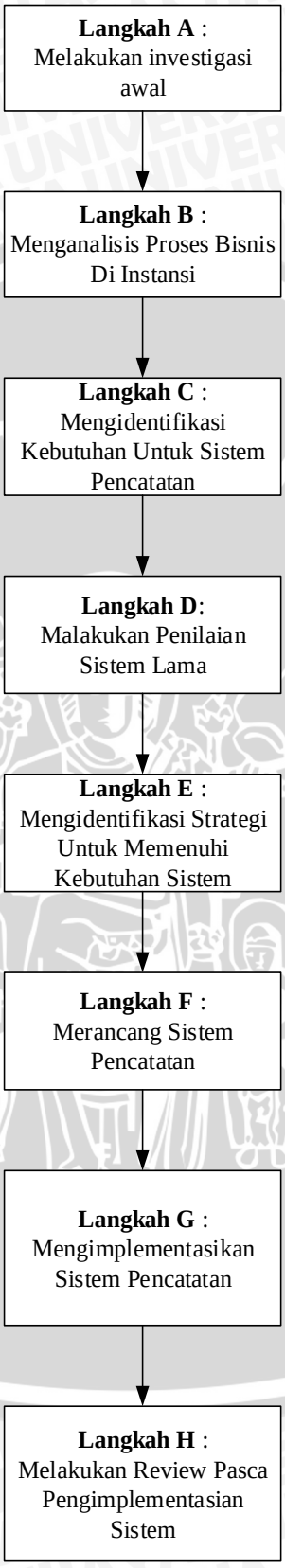
Dari proses identifikasi permasalahan dan pengumpulan data, maka dapat diambil kesimpulan bahwa kantor UPT Laboratorium Kesehatan Hewan Malang membutuhkan sebuah sistem yang dapat memanajemen arsip yang lebih terstruktur dan sistematis, karena proses pencatatan arsip fisik dan elektronik masih dilakukan dengan pencatatan manual dan tempat penyimpanan yang tidak beraturan sehingga ketersediaan arsip sebagai akuntabilitas kinerja organisasi tidak terjamin. Oleh karena itu, dibutuhkan sebuah standar sebagai acuan untuk proses pengelolaan rekod arsip agar manajemen arsip lebih terstruktur dan sistematis. ISO seri 15489 bisa menjadi sebuah solusi yang dapat digunakan sebagai acuan konsep standar manajemen pencatatan arsip agar pengelolaan arsip lebih terstruktur dan sistematis, karena di dalam ISO 15489 terdapat butir-butir yang mengatur tentang manajemen rekod arsip dengan lingkungan penggunanya. Dari proses identifikasi permasalahan tersebut, maka pembuatan sebuah perangkat lunak/*software* untuk manajemen arsip yang mengacu pada ISO 15489 dapat menjadi salah satu solusi alternatif yang bisa diterapkan.

Adapun tahapan realisasi solusi sistem manajemen rekod arsip sesuai dengan konsep ISO 15489 *Design and Implementation of Records System* (DIRS *methodelogy*) dapat dilihat pada gambar 3.2.

Tahapan-tahapan konsep DIRS ISO 15489 dijelaskan sebagai berikut :

1. Langkah A: Melakukan Investigasi Awal

   Investigasi awal merupakan proses untuk mengidentifikasi faktor-faktor utama yang mempengaruhi kebutuhan untuk menciptakan dan penyimpanan arsip pada organisasi. Proses ini dilakukan dengan cara wawancara dengan pegawai UPT Laboratorium Kesehatan Hewan Malang yang terlibat langsung dengan proses pengelolaan arsip.



Gambar 3.2 Metode DIRS ISO 15489

2. Langkah B: Menganalisis Bisnis Proses Di Instansi

   Proses ini dilakukan dengan cara observasi untuk melakukan dan mengarsipkan analisis bisnis secara sistematis dan membuat penggunaan terbaik dari hasil-hasil proses organisasi dan catatannya. Tujuan dari langkah ini adalah untuk mengembangkan model konseptual dari apa yang organisasi lakukan dan bagaimana melakukannya, itu akan menunjukkan bagaiamana catatan berhubungan dengan keperluan bisnis organisasi dan proses bisnisnya. Produk yang berasal dari langkah ini mungkin termasuk :

   a. pengarsipan yang menggambarkan bisnis dan proses bisnis organisasi,
   b. Peta proses bisnis organisasi yang menunjukkan titik-titik organisasi di mana catatan diproduksi atau diterima sebagai produk atau aktivitas bisnis.

3. Langkah C: Mengidentifikasi Kebutuhan untuk Sistem Pencatatan

   Proses ini dilakukan dengan cara mengumpulkan dan menganalisis data hasil wawancara dan observasi yang telah dilakukan pada langkah A dan B untuk mengidentifikasi kebutuhan organisasi untuk membuat, menerima, menyimpan catatan kegiatan usaha, dan mengarsipkan kebutuhan dalam bentuk terstruktur dan mudah dipelihara.

4. Langkah D: Melakukan Penilaian Pada Sistem Sebelumnya

   Langkah ini dilakukan dengan cara mengamati jalannya alur sistem yang ada di organisasi dalam hal pencatatan dan informasi lainnya untuk mengukur sejauh mana mereka menangkap dan memelihara catatan dari kegiatan usaha. Proses Ini akan memberikan dasar untuk mengembangkan sistem baru atau mendesain ulang sistem yang ada untuk memenuhi kebutuhan catatan yang telah diidentifikasi dan disepakati dalam langkah sebelumnya.

5. Langkah E: Mengidentifikasi Strategi Untuk Memenuhi Kebutuhan Sistem

Tujuan dari langkah ini adalah untuk menentukan kebijakan, prosedur, standar, peralatan, dan taktik lainnya yang harus diadopsi oleh organisasi untuk memastikan bahwa sistem tersebut dapat membuat dan menyimpan catatan yang diperlukan untuk aktivitas bisnisnya. Langkah ini dilakukan dengan cara berdiskusi dengan pihak yang berwenang dalam pengelolaan arsip di UPT Laboratorium Kesehatan Hewan Malang. Strategi mungkin akan termasuk :

a. Mengadopsi kebijakan dan prosedur,
b. Menyusun standar
c. Merancang komponen sistem baru, atau
d. Menerapkan sistem dengan cara yang memenuhi persyaratan diidentifikasi untuk menjaga dan menyimpan catatan.

Ketika langkah ini selesai, akan ada pendekatan yang terencana, sistematis dan sesuai dengan penciptaan, penangkapan, pemeliharaan, dan penggunaan catatan yang akan memberikan dasar untuk mendesain ulang sistem pencatatan.

6. Langkah F: Merancang Sistem Pencatatan

   Langkah ini melibatkan pengkonversian strategi dan taktik yang dipilih pada langkah E dalam rencana sistem pencatatan yan memenuhi persyaratan yang diidentifikasi dan diarsipkan pada langkah C dan solusi dari kekurangan sistem manajemen pencatatan dalam organisasi yang diidentifikasi pada langkah D. Langkah ini melibatkan :

   a. merancang perubahan sistem saat ini, dalam hal proses dan prakteknya,
   b. mengadaptasi atau mengintegrasikan solusi teknologi, dan
   c. menentukan bagaimana cara terbaik untuk memasukkan perubahan ini untuk meningkatkan pengelolaan catatan dalam sebuah organisasi

7. Langkah G: Mengimplementasikan Sistem Pencatatan

   Proses ini dilakukan untuk mengidentifikasi secara sistematis dan menempatkan campuran yang tepat dari strategi dan pengimplementasian rencana yang dirancang pada Langkah F. Rencana itu memberikan gambaran tentang bagaimana berbagai komponen sistem (proses, prosedur, orang dan teknologi) harus cocok satu sama lain. Setelah menyelesaikan langkah ini, organisasi harus mengintegrasikan praktik perbaiakan manajemen pencatatan ke dalam organisasi dengan gangguan minimal untuk aktivitas bisnis; memberikan kontribusi terhadap kebutuhan organisasi untuk kualitas akreditasi; dan memanfaatkan investasi jangka panjang yang telah dibuat dalam langkah A sampai dengan langkah F.

8. Langkah H: Melakukan Review Pasca Pengimplementasian Sistem

   Langkah H melibatkan :

   a. Proses analisis apakah sistem pencatatan telah dibuat dan disusun sesuai dengan kebutuhan kegiatan bisnis dan tepat saling terkait dengan kebutuhan proses bisnis antar bagian.
   b. Wawancara manajemen, staf dan pemangku kepentingan lainnya.
   c. Melakukan survei
   d. Memeriksa pengarsipan yang dikembangkan selama fase awal dari proyek pengembangan sistem.

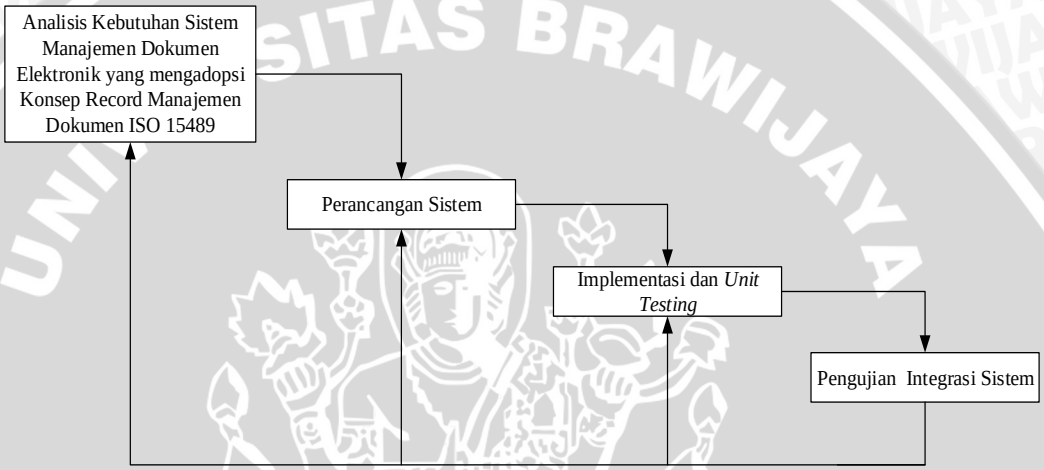
   Tinjauan pasca-pelaksanaan akan meminimalkan resiko kegagalan sistem dan dari waktu ke waktu akan mengantisipasi perubahan persyaratan yang signifikan untuk kebutuhan sistem pencatatan yang dibutuhkan di kantor UPT Laboratorium Kesehatan Hewan Malang.

3.3. Realisasi Solusi

Realisasi solusi merupakan implementasi proses pembuatan sistem dari tahap identifikasi solusi. Implementasi aplikasi dilakukan dengan mengacu pada perancangan perangkat lunak yang sudah di desain pada bab analisis kebutuhan

dan perancangan perangkat lunak. Implementasi perangkat lunak dilakukan dengan menggunakan bahasa pemrograman PHP dengan framework CodeIgniter. Bootstrap digunakan sebagai *framework* desain tampilan, MySql sebagai basis data, dan prinsip-prinsip ISO 15489 sebagai acuan untuk proses manajemen pencatatan arsip elektronik di dalam perangkat lunak yang dikembangkan.

Untuk tahapan pembuatan perangkat lunak, penulis menggunakan konsep SDLC (*Waterfall Model*) dapat dilihat pada gambar 3.3 berikut :



Gambar 3.3 Diagram Blok Tahap Pembuatan Software

Tahap pembuatan perangkat lunak yang mengacu pada proses pengembangan sistem SDLC (*Waterfall Model*) pada gambar 3.3 dapat dijabarkan sebagai berikut :

3.3.1 Pendefinisian Kebutuhan

Pendefinisian kebutuhan dilakukan dengan menentukan kebutuhan apa saja yang diperlukan sesuai dengan data yang didapatkan pada saat proses pengumpulan data yang telah dilakukan sebelumnya kemudian merancangnya dalam bentuk sistem manajemen arsip elektronik yang mengacu pada standar sistem pencatatan ISO 15489. Proses analisis kebutuhan sistem dalam penelitian ini meliputi :

a. Pemodelan *Use Case* serta *Activity Diagram* untuk proses bisnis sebelum implementasi sistem dan proses bisnis sesudah implementasi sistem.

b. Pemodelan aktivitas sistem.

Metode analisis yang digunakan adalah metode *Object Oriented Analysis* dengan menggunakan bahasa pemodelan UML (*Unified Modelling Language*). Kebutuhan yang digunakan dalam pembuatan sistem ini meliputi:

1) Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang terkait dengan fungsi produk, misalnya sistem harus mampu mencetak laporan. Sistem manajemen arsip elektronik ini memiliki fitur antara lain sebagai berikut :

a) Sistem dapat mengelola data *user* beserta dengan hak aksesnya, *user* disini merupakan karyawan UPT Laboratorium Kesehatan Hewan Malang sesuai dengan bagian dan jabatannya masing-masing,
b) Sistem dapat mengelola data surat masuk beserta disposisinya, memberikan informasi laporan jumlah surat masuk dalam jangka waktu harian, bulanan, tahunan dan dapat dicetak dalam bentuk format PDF.
c) Sistem dapat mengelola data surat keluar, beserta dengan informasi laporan jumlah surat keluar dalam jangka waktu harian, bulanan, tahunan dan dapat dicetak dalam bentuk format PDF.
d) Sistem dapat mengelola data arsip, mengatur klasifikasi dan kategori arsip sesuai dengan kebijakan manajemen arsip yang berlaku di UPT Laboratorium Kesehatan Hewan Malang,
e) Sistem dapat memastikan perubahan revisi arsip dari setiap arsip yang telah di *upload* ke dalam sistem,
f) Sistem dapat mengelola peminjaman arsip untuk memberi informasi pada proses penemuan kembali arsip,
g) Sistem dapat mengelola pemusnahan arsip sesuai dengan Jadwal Retensi Arsip yang telah dibuat.

Tabel 3.1 Kebutuhan Fungsional

Kebutuhan Fungsional	
Super Admin	*Create, read, update, delete* data user beserta seluruh fungsionalitas sistem.
Admin Kepegawaian	*Create, read, update, delete* data data arsip yang ada dalam bagian kepegawaian, data karyawan, data bagian, data jabatan, dan data jabatan fungsional, peminjaman arsip, disposisi surat masuk, validasi usul musnah arsip.
Admin Pelayanan.	*Create, read, update, delete* data arsip yang ada dalam bagian pelayanan, data revisi arsip, peminjaman arsip, disposisi surat masuk, validasi usul musnah arsip.
Admin Persuratan	*Create, read, update, delete* data surat masuk dan surat keluar, konsep surat keluar, disposisi, data revisi arsip, peminjaman arsip, validasi usul musnah arsip.
Administratif	*Create, read, update, delete* data kategori arsip, berita acara pemusnahan arsip, disposisi surat masuk, klasifikasi arsip.
Kepala UPT	Kelola disposisi surat masuk, lihat laporan arsip.

2) *Development requirement* adalah kebutuhan yang terkait *tools* untuk pengembangan sistem informasi baik perangkat keras maupun perangkat lunak. Berikut adalah kebutuhan pengembangan sistem :

Tabel 3.2 Kebutuhan Pengembangan

Kebutuhan Pengembangan	
Perangkat Keras	Personal Computer dengan kebutuhan minimal : 1. Processor Core 2 Duo 2. RAM 2 GB
Perangkat Lunak	1. Sistem Operasi Microsoft Windows. 2. Dreamweaver sebagai tools editor kode. 3. Astah Professional sebagai *tools* perancangan UML. 4. Microsoft Visio sebagai *tools* perancangan sistem. 5. Web Server Apache. 6. Framework CodeIgniter. 7. RDBMS MySql

3.3.2 Perancangan Sistem

Tahap perancangan ini terdiri dari perancangan pemodelan proses bisnis, *Use Case Diagram, Activity Diagram, Sequence Diagram, Class Diagram* dan struktur *database.*

1. Perancangan Pemodelan Bisnis Baru

   Setelah penerapan Sistem Informasi Manajemen Arsip, diharapkan pengelolaan arsip dari seluruh bagian yang ada di Kantor UPT Laboratorium Kesehatan Hewan Malang lebih terstruktur dan meningkatkan efektivitas dan efisiensi organisasi.

2. Perancangan *Use Case Diagram*

   Tahap perancangan pertama yaitu perancangan *Use Case diagram* untuk mendeskripsikan interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Serta untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu [14].

3. Perancangan *Activity Diagram*

   Perancangan *Activity Diagram* untuk menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. *Activity Diagram* menggambarkan aktivitas sistem bukan apa yang dilakukan actor, jadi aktivitas yang dapat dilakukan oleh sistem. *Activity Diagram* juga banyak digunakan untuk mendefinisikan hal-hal berikut [14] :

   a) Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
   b) Urutan atau pengelompokan tampilan dari sistem / *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
   c) Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
   d) Rancangan menu yang ditampilkan pada perangkat lunak.

4. Perancangan *Sequence Diagram*

   Diagram sekuen menggambarkan kelakuan objek pada *Use Case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram *sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah *Use Case*. Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian *Use Case* yang memiliki proses sendiri atau yang penting semua *Use Case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga

semakin banyak *Use Case* yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak.

5. Perancangan *Class Diagram*

   Perancangan *Class diagram* ini dilakukan untuk merancang *Class* yang nantinya akan diimplementasikan pada framework CodeIgniter yang menerapkan pola MVC. Dengan perancangan *Class diagram* ini, maka akan mempermudah peneliti dalam mengembangkan sistem sesuai dengan kebutuhan dan mengimplementasikannya pada *class-class* di CodeIgniter.

6. Perancangan *Entity Relationship Diagram (ERD)* Perancangan ERD digunakan untuk pemodelan awal basis data relasional. ERD biasanya memiliki hubungan *binary* (satu relasi menghubungkan dua buah entitas). Beberapa metode perancangan ERD menileransi hubungan relasi *ternary* (satu relasi menghubungkan tiga buah relasi) atau N-ary (satu relasi menghubungkan banyak entitas) [14].

3.3.3 Implementasi dan Pengujian Unit

Implementasi aplikasi dilakukan dengan mengacu pada pengidentifikasian kebutuhan dan perancangan desain perangkat lunak yang sudah di rancang sebelumnya.

Pengujian unit dilakukan untuk mengetahui apakah kinerja setiap unit dari sistem manajemen arsip elektronik ini telah memenuhi spesifikasi kebutuhan yang diperlukan dan juga untuk mendeteksi kesalahan (*bug*) dari sistem. Pengujian dilakukan dengan dua cara, yaitu:

1. *Functional Test*

   Dalam aplikasi web, pengujian fungsional digunakan untuk menguji apakah link serta menu-menu dalam aplikasi sudah berjalan dengan baik atau belum.

2. *Checklist* ISO 15489

   Checklist ISO 15489 digunakan untuk menilai apakah sistem yang telah diterapkan sudah sesuai dengan persyaratan ISO 15489 atau belum.

3.3.4 Pengujian Integrasi Sistem

Sesuatu yang dibuat haruslah diujicobakan. Demikian juga dengan *software*. Semua fungsi-fungsi *software* harus diujicobakan, agar *software* bebas dari *error*, dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya. Pengujian integrasi adalah pengujian terhadap unit-unit program yang saling berhubungan (terintegrasi) dengan fokus pada masalah *interfacing*.

3.4. Analisa Hasil Realisasi Solusi

Analisa hasil realisasi solusi merupakan proses pengambilan kesimpulan yang dilakukan setelah semua tahapan perancangan sistem, implementasi sistem dan pegujian sistem telah selesai dilakukan. Analisa hasil realisasi solusi didapatkan dengan melakukan pengujian analisis dan evaluasi pada pengimplementasian sistem yang telah dibangun terhadap dampak yang dirasakan oleh kantor UPT Laboratorium Kesehatan Hewan Malang dan kesesuaian sistem yang telah dibangun terhadap standar sistem pencatatan ISO 15489

BAB IV
ANALISIS DAN PERANCANGAN

4.1. Gambaran Umum

Sistem Manajemen Arsip Elektronik berdasarkan ISO 15489 di UPT Laboratorium Kesehatan Hewan Malang merupakan sistem yang dibangun untuk mendukung proses bisnis dan efisiensi kinerja organisasi yang dilakukan di kantor UPT Laboratorium Kesehatan Hewan Malang dalam hal pengelolaan arsip. Setiap aktor yakni karyawan atau staf memiliki hak akses yang berbeda terhadap sistem tergantung pada jabatan dan bagian. Super Admin mempunyai tanggung jawab untuk memberikan validasi hak akses kepada user yang mempunyai wewenang dalam pengelolaan arsip. Modul-modul yang ada pada sistem admin mempunyai hubungan dan keterkaitan satu sama lain karena dalam melakukan transaksi kegiatan saling membutuhkan data.

4.2. Elisitasi

Dari hasil penelitian yang dilakukan dengan cara wawancara kepada pihak Kantor UPT Laboratorium Kesehatan Hewan Malang, yaitu Kepala UPT dan Kasubag Tata Usaha, maka diperoleh hasil elisitasi kebutuhan perangkat lunak yang akan digunakan sebagai dasar pembangunan dan pengembangan sebuah sistem manajemen arsip elektronik yang mengacu pada standar sistem pencatatanarsip ISO 15489. Berikut kebutuhan fungsional sistem yang telah diajukan oleh Kasubag Tata Usaha dan Kepala UPT Laboratorium Kesehatan Hewan Malang:

1) Sistem dapat mengelola data *user* beserta dengan hak aksesnya, *user* disini merupakan karyawan UPT Laboratorium Kesehatan Hewan Malang sesuai dengan bagian dan jabatannya masing-masing,
2) Sistem dapat mengelola data konsep surat keluar, mulai dari pembuatan konsep surat sampai konsep surat tersebut dicetak.
3) Sistem dapat mengelola data surat keluar, beserta dengan informasi laporan jumlah surat keluar dalam jangka waktu harian, bulanan, tahunan dan dapat dicetak dalam bentuk format PDF

4) Sistem dapat mengelola data arsip, mengatur klasifikasi dan kategori arsip sesuai dengan kebijakan manajemen arsip yang berlaku di UPT Laboratorium Kesehatan Hewan Malang,
5) Sistem dapat memastikan perubahan revisi arsip dari setiap arsip yang telah di *upload* ke dalam sistem,
6) Sistem dapat mengelola peminjaman arsip untuk memberi informasi pada proses penemuan kembali arsip,
7) Sistem dapat mengelola pemusnahan arsip sesuai dengan Jadwal Retensi Arsip yang telah dibuat.

Sedangkan hasil elisitasi kebutuhan perangkat lunak berdasarkan penelitian yang berkaitan dengan standar sistem pencatatan ISO 15489, sistem pencatatan harus mempunyai karakteristik [9] :

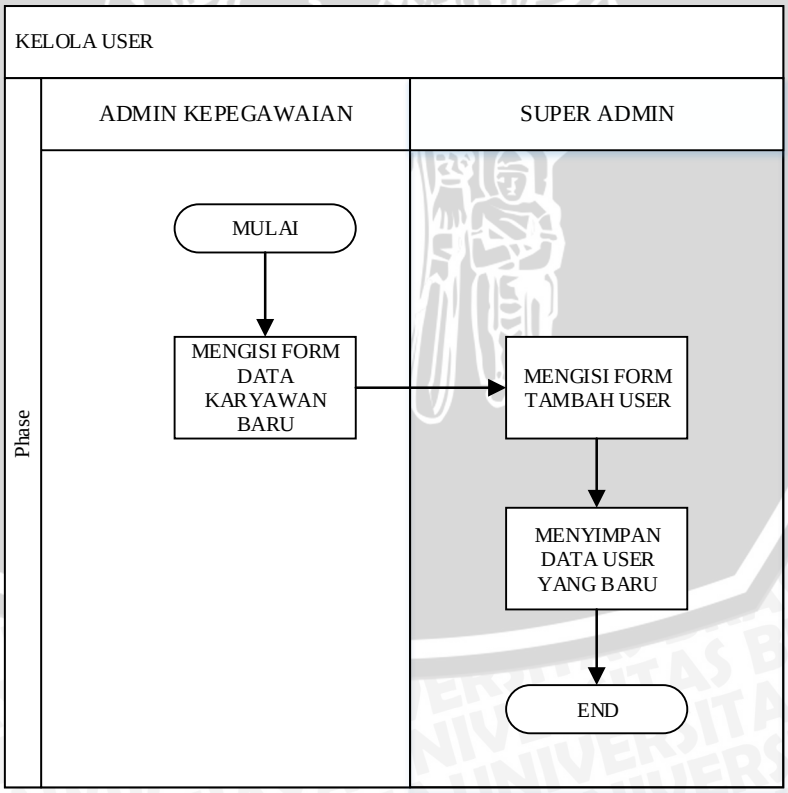
1. Kehandalan (*Reliability*).
   a. Sistem pencatatan harus dapat menampung semua data arsip dalam lingkup kegiatan usaha.
   b. Pengelolaan arsip yang mencerminkan proses bisnis yang yang dilakukan oleh pencipta arsip.
   c. Melindungi arsip dari pengubahan dan penghapusan yang tidak sah.
   d. Berfungsi sebagai sumber utama informasi dari konten arsip yang telah dicatat.
   e. Dapat diakses kapanpun saat informasi arsip dibutuhkan.
2. Keutuhan (*Integrity*). Langkah-langkah pengendalian seperti pengaturan hak akses, verifikasi pengguna, kewenangan penghapusan dan keamanan arsip harus diimplementasikan untuk mencegah akses yang tidak sah, perusakan, perubahan atau penghapusan arsip.
3. Kesesuaian (*Compliance*). Sistem catatan harus dikelola sesuai dengan semua persyaratan proses bisnis instansi saat ini, peraturan yang sedang berjalan dan harapan karyawan di mana organisasi beroperasi. Bagian yang mempunyai catatan harus memahami bagaimana persyaratan ini mempengaruhi proses bisnis yang mereka lakukan.

4. Kelengkapan (*Comprehensiveness*). Sistem pencatatan harus dapat mengelola semua arsip hasil dari aktifitas bisnis organisasi, atau bagian organisasi dimana sistem di operasikan.
5. Sistematis (*Systematic*). Arsip yang dicatat harus diketahui kapan dicatat, dirubah atau dihapus oleh bagian yang mempunyai wewenang. Harus ada riwayat dari setiap perubahan dan penghapusan arsip yang telah dicatat dalam sistem.

4.3. *Swimlane Diagram* Proses Bisnis Sistem

1. Kelola Data User

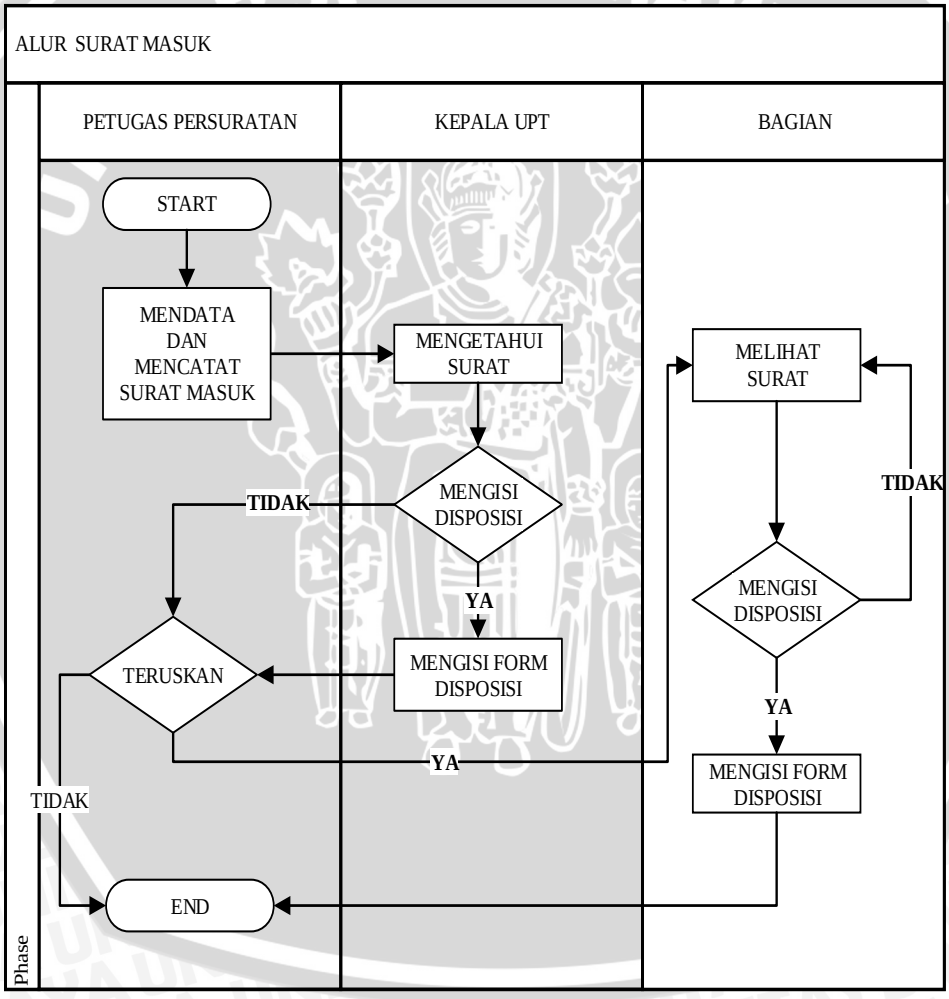
Alur proses bisnis kelola data user dimulai dari admin kepegawaian memasukkan data user yang baru, setalah data tersimpan, super admin memvalidasi dan memberikan *username* dan *password*. Alur kelola data user ditunjukkan pada gambar 4.1.



Gambar 4.1 Alur Kelola Data User

2. Kelola Data Surat Masuk

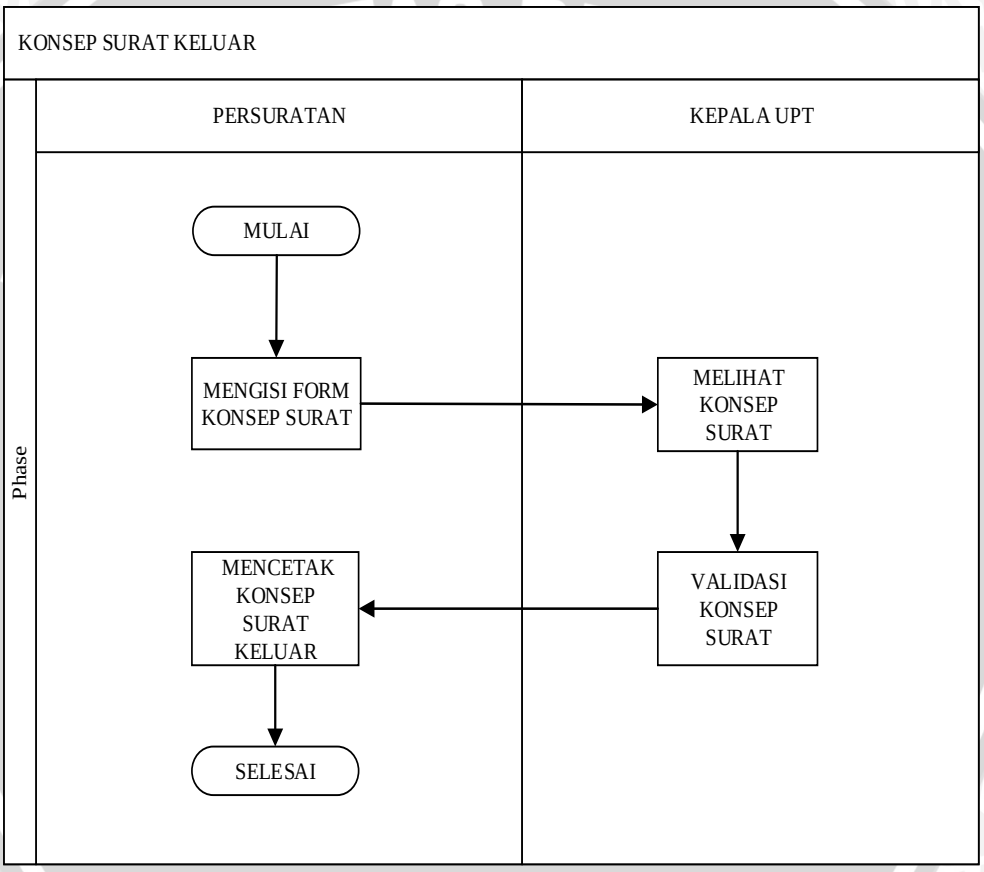
Alur kelola data surat masuk pada sistem yang akan dibuat dimulai dari admin persuratan mencatat data surat masuk, setelah data tersimpan, Kepala UPT menerima notifikasi bahwa ada surat masuk. Kepala UPT bisa mengetahui surat masuk dan mengisi form disposisi sebagai perintah tindak lanjut atas surat masuk tersebut. Ketika Kepala UPT mengetahui surat masuk, admin persuratan menerima notifikasi bahwa surat masuk telah dibaca. Alur kelola data surat masuk ditunjukkan pada gambar 4.2.



Gambar 4.2 Alur Kelola Data Surat Masuk

3. Kelola Data Konsep Surat Keluar

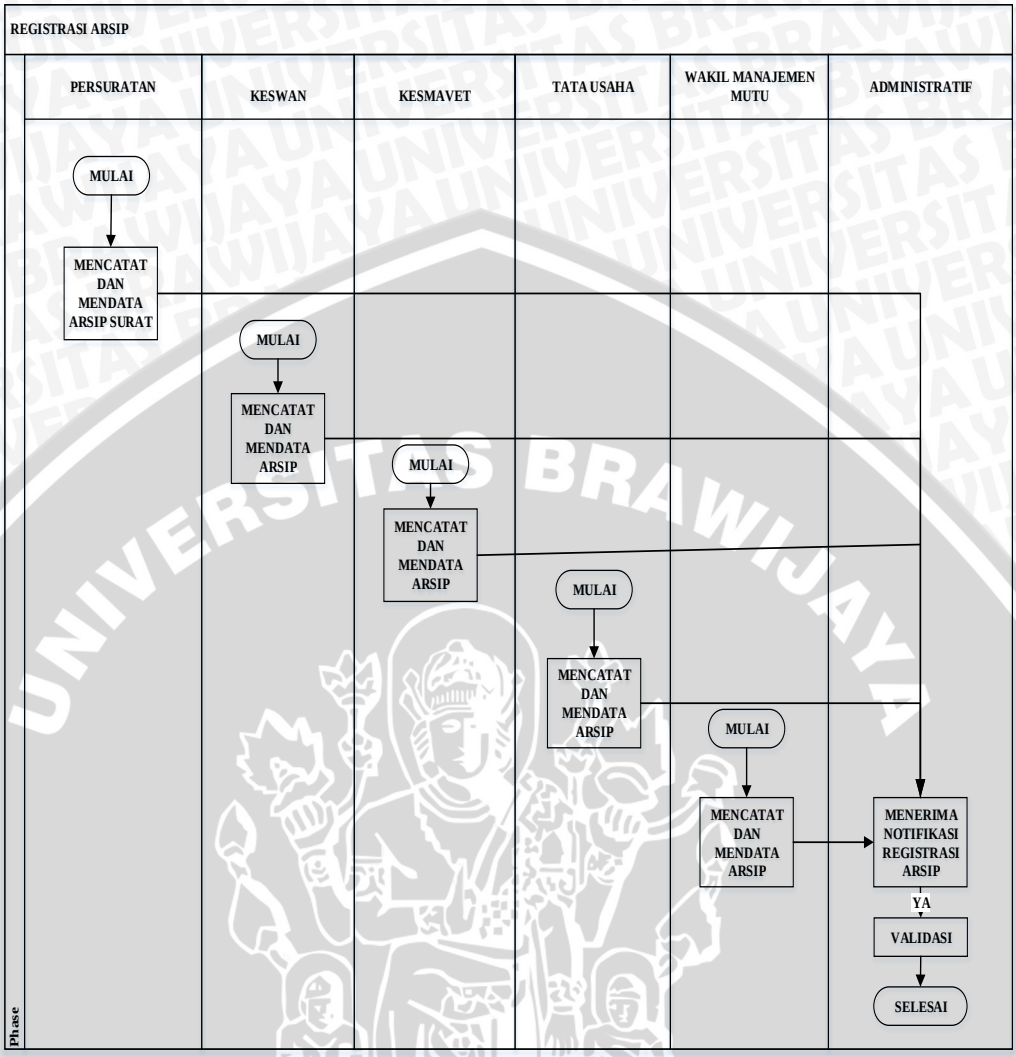
Alur kelola data konsep surat keluar dimulai dari admin persuratan membuat konsep surat keluar, setelah data tersimpan Kepala UPT menerima notifikasi konsep surat keluar baru, setelah konsep surat keluar disetujui, konsep surat keluar otomatis mendapat nomor urut surat. Admin Persuratan kemudian mencetak konsep surat keluar. Alur kelola data konsep surat keluar ditunjukkan pada gambar 4.3.



Gambar 4.3 Alur Kelola Konsep Surat Keluar

4. Kelola Data Arsip

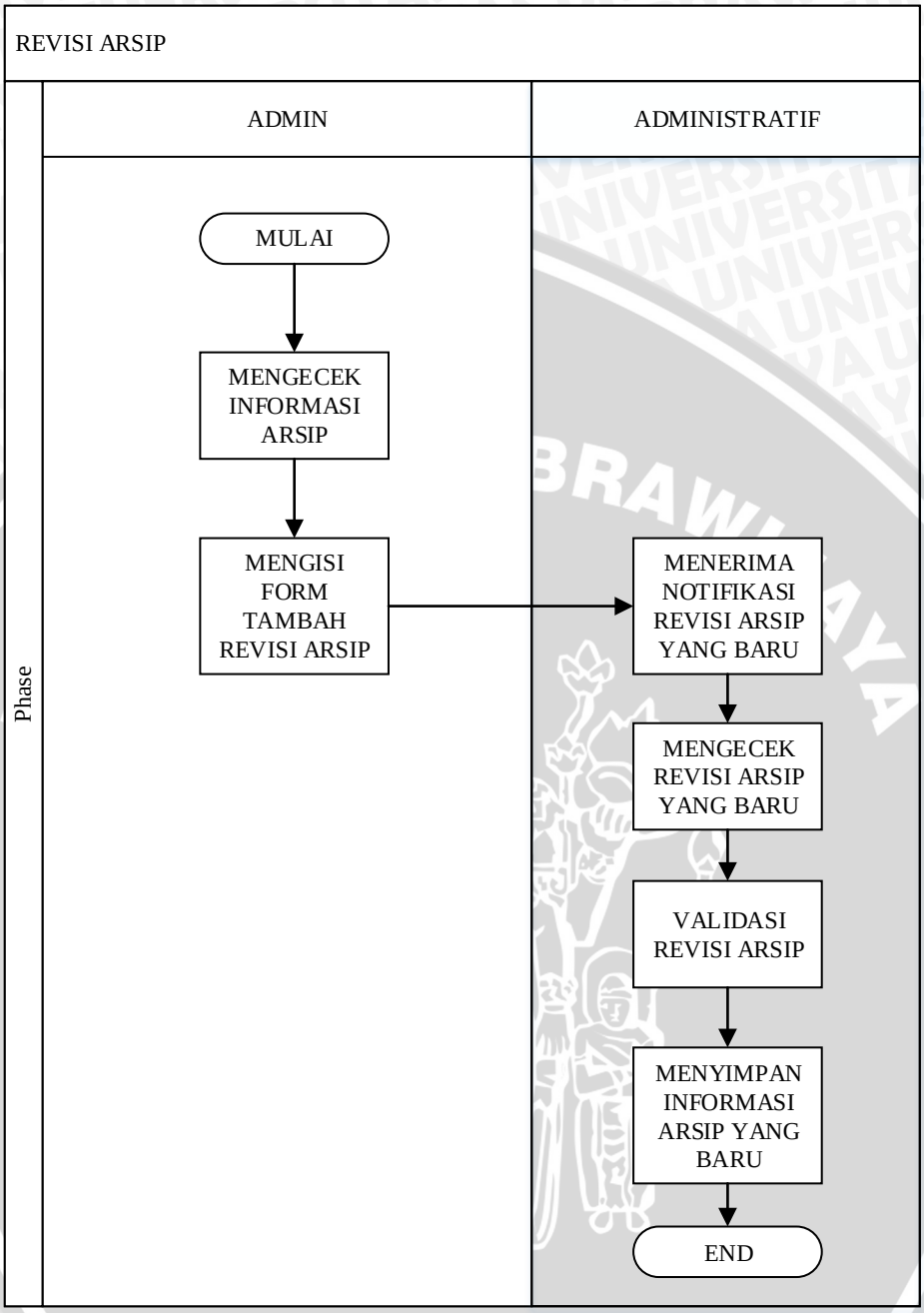
Alur kelola data arsip dimulai dari admin dari setiap bagian bisa menambah data arsip di bagian masing-masing untuk di registrasi, setelah data tersimpan administratif menerima notifikasi data arsip baru untuk di validasi. Alur kelola data arsip ditunjukkan pada gambar 4.4.



Gambar 4.4 Alur Kelola Data Arsip

5. Kelola Data Revisi Arsip

Alur revisi arsip dimulai dari admin yang mempunyai hak akses arsip mengisi form revisi arsip, setelah data tersimpan administratif menerima notifikasi revisi arsip baru dan mencocokkan dengan revisi arsip yang asli sebelum dilakukan proses validasi. Alur kelola data revisi arsip ditunjukkan pada gambar 4.5.

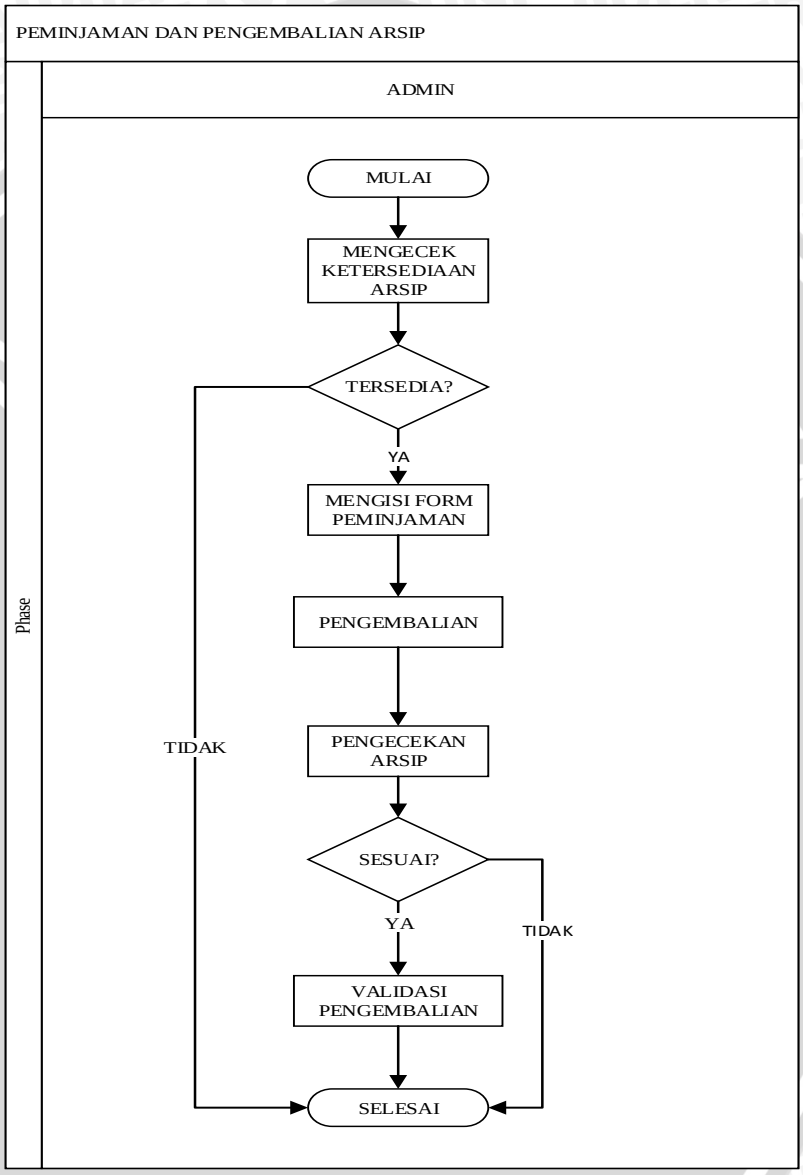


Gambar 4.5 Alur Kelola Data Revisi Arsip

6. Kelola Data Peminjaman Arsip

Alur peminjaman arsip dimulai dari admin setiap bagian mengecek ketersediaan arsip, jika arsip dalam keadaan tersedia (tidak terpinjam), admin mengisi form peminjaman, kemudian dalam proses pengembalian, dilakukan pengecekan arsip, jika arsip yang telah dikembalikan sesuai

dengan yang dipinjam, maka dilakukan validasi pengembalian arsip di dalam sistem. Alur kelola data peminjaman dan pengembalian arsip ditunjukkan pada gambar 4.6.

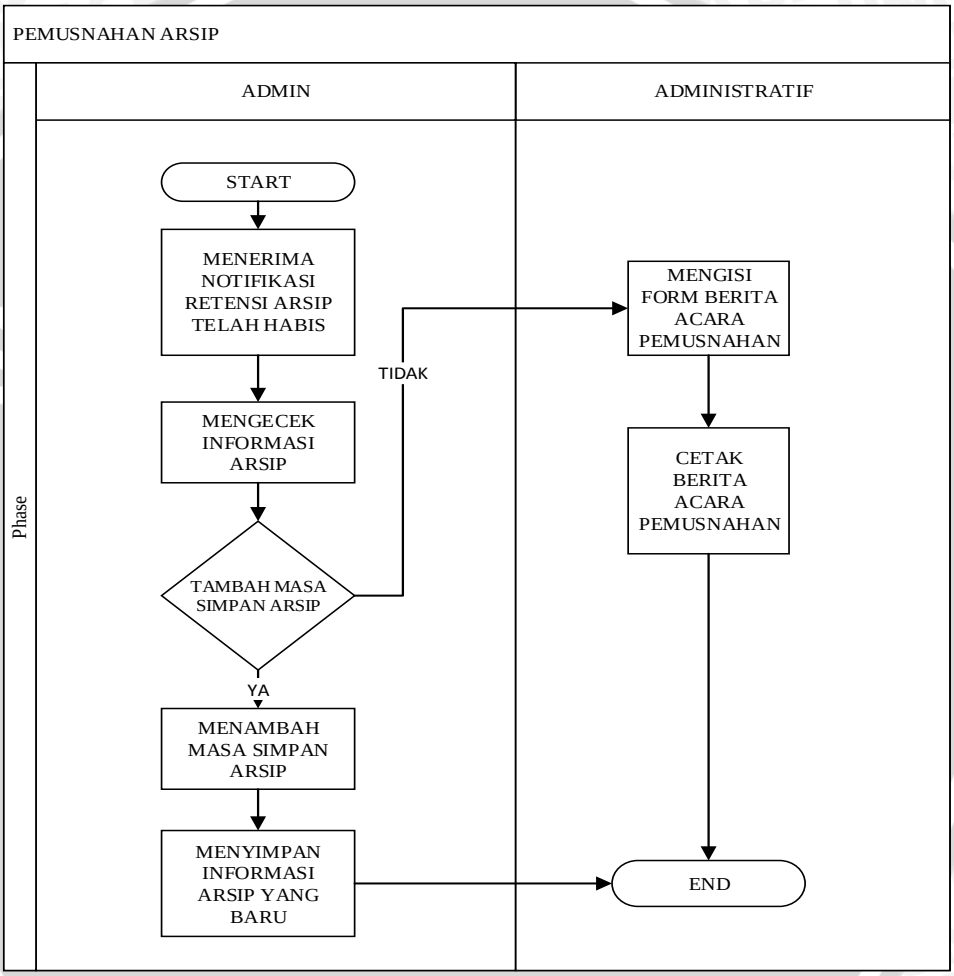


Gambar 4.6 Alur Kelola Data Peminjaman dan Pengembalian Arsip

7. Kelola Data Arsip Usul Musnah

Alur arsip usul musnah dimulai dari admin menerima notifikasi bahwa retensi arsip telah habis, admin mengecek informasi arsip. Jika admin menambah masa simpan arsip, maka arsip kembali aktif dan

disimpan kembali dalam data arsip. Arsip yang tidak ditambah masa simpannya maka akan masuk ke dalam berita acara pemusnahan untuk dimusnahkan. Administratif mengisi form berita acara kemudian mencetak berita acara yang telah dibuat. Alur kelola data user ditunjukkan pada gambar 4.7.



Gambar 4.7 Alur Kelola Data Arsip Usul Musnah

4.4. Identifikasi Aktor dan Fungsionalitas Sistem

Tabel 4.1 Identifikasi Aktor dan Fungsionalitas Sistem

No.	Aktor	Deskripsi
1.	**Super Admin**	Super Admin adalah administrator sistem yang dapat mengelola semua fitur/fungsionalitas sistem, meliputi

		kelola user, validasi user, kelola data bagian, kelola kategori arsip, kelola data karyawan, kelola data jabatan, kelola data jabatan fungsional, kelola data revisi, validasi revisi arsip, kelola data arsip, kelola data surat masuk, kelola data surat keluar, dan validasi data arsip.
2.	**Administratif**	Administratif adalah petugas yang dapat mengelola fitur/fungsionalitas sistem meliputi, kelola data kategori arsip, kelola klasifikasi arsip, validasi data arsip, validasi revisi, kelola berita acara.
3.	**Admin Kepegawaian**	Admin Kepegawaian adalah administrator sistem yang dapat mengelola fitur/fungsionalitas sistem, meliputi kelola data arsip, daftar arsip usul musnah, kelola revisi arsip, kelola data karyawan, kelola data bagian, kelola data jabatan, dan kelola data jabatan fungsional.
4.	**Admin Persuratan**	Admin Persuratan adalah administrator sistem yang dapat mengelola fitur/fungsionalitas sistem, meliputi kelola konsep surat keluar, data arsip yaitu surat masuk, surat keluar, dan daftar arsip usul musnah.
5.	**Admin Pelayanan**	Admin Pelayanan adalah administrator sistem yang dapat mengelola fitur/fungsionalitas sistem kelola data arsip yang ada di bagian pelayanan, kelola revisi arsip, kelola daftar arsip usul musnah.
6.	**Kepala UPT**	Kepala UPT adalah pimpinan di UPT Laboratorium Kesehatan Hewan Malang. Aktor memiliki kewenangan untuk lihat laporan, mengetahui disposisi surat masuk, dan validasi konsep surat keluar.

4.5. Daftar Kebutuhan Perangkat Lunak

Daftar kebutuhan sistem terdiri dari kebutuhan fungsional dan kebutuhan fungsional yang masing-masing akan dispesifikasikan menurut prioritas pada tabel 4.2 sebagai berikut :

Tabel 4.2 Spesifikasi Kebutuhan Fungsional

No	No SRS	Kebutuhan	Aktor
1	SRS_6_01	Tambah Data klasifikasi arsip	Super Admin, Administratif
2	SRS_6_02	Ubah Data klasifikasi arsip	Super Admin, Administratif
3	SRS_6_03	Hapus Data klasifikasi arsip	Super Admin, Administratif
4	SRS_6_04	Lihat Data klasifikasi arsip	Super Admin, Administratif
5	SRS_6_05	Tambah kategori arsip	Super Admin, Administratif
6	SRS_6_06	Ubah kategori arsip	Super Admin, Administratif
7	SRS_6_07	Hapus kategori arsip	Super Admin, Administratif
8	SRS_6_08	Lihat kategori arsip	Super Admin, Administratif
9	SRS_4_01	Tambah data arsip	Super Admin, Admin Persuratan, Admin Kepegawaian, Admin Pelayanan,
10	SRS_4_02	Hapus data arsip	Super Admin, Admin Persuratan, Admin Kepegawaian, Admin Pelayanan.
11	SRS_4_03	Ubah data arsip	Super Admin, Admin Persuratan, Admin Kepegawaian, Admin Pelayanan.
12	SRS_4_04	Lihat data arsip	Super Admin, Admin kepegawaian, Admin Persuratan, Admin Pelayanan.
13	SRS_6_01	Validasi registrasi arsip	Super Admin, Administratif
14	SRS_5_01	Tambah data surat masuk	Super Admin, Admin Persuratan
15	SRS_5_02	Hapus data surat masuk	Super Admin, Admin Persuratan
16	SRS_5_03	Ubah data surat masuk	Super Admin, Admin Persuratan
17	SRS_5_04	Lihat data surat masuk	Super Admin, Admin Persuratan
18	SRS_5_05	Tambah data surat keluar	Super Admin, Admin Persuratan
19	SRS_5_06	Hapus data surat keluar	Super Admin, Admin Persuratan
20	SRS_5_07	Ubah data surat keluar	Super Admin, Admin Persuratan
21	SRS_5_08	Lihat data surat keluar	Super Admin, Admin Persuratan
22	SRS_5_09	Tambah data konsep surat keluar	Super Admin, Admin Persuratan

23	SRS_5_10	Hapus data konsep surat keluar	Super Admin, Admin Persuratan
24	SRS_5_11	Ubah data konsep surat keluar	Super Admin, Admin Persuratan
25	SRS_5_12	Lihat data konsep surat keluar	Super Admin, Admin Persuratan
26	SRS_4_11	Daftar Usul Arsip Musnah	Super Admin, Admin Persuratan, Admin Kepegawaian, Admin Pelayanan.
27	SRS_6_09	Tambah Data Berita Acara Pemusnahan	Super Admin, Administratif
28	SRS_6_10	Lihat Data Berita Acara Pemusnahan	Super Admin, Administratif
29	SRS_6_11	Cetak Berita Acara Pemusnahan	Super Admin, Administratif
30	SRS_4_05	Tambah data revisi	Super Admin, Admin Persuratan, Admin Kepegawaian, Admin Pelayanan.
31	SRS_4_06	Lihat data revisi	Super Admin, Admin Persuratan, Admin Kepegawaian, Admin Pelayanan.
32	SRS_6_12	Validasi revisi arsip	Super Admin, Administratif
33	SRS_4_07	Tambah Data Peminjaman	Super Admin, Admin Persuratan, Admin Kepegawaian, Admin Pelayanan.
34	SRS_4_08	Hapus Data Peminjaman	Super Admin, Admin Persuratan, Admin Kepegawaian, Admin Pelayanan.
35	SRS_4_09	Lihat Data Peminjaman	Super Admin, Admin Persuratan, Admin Kepegawaian, Admin Pelayanan.
36	SRS_7_01	Lihat Laporan	Super Admin, Kepala UPT
37	SRS_3_13	Tambah data bagian	Super Admin, Admin kepegawaian
38	SRS_3_14	Hapus data bagian	Super Admin, Admin kepegawaian
39	SRS_3_15	Ubah data bagian	Super Admin, Admin kepegawaian
40	SRS_3_16	Lihat data bagian	Super Admin, Admin kepegawaian
41	SRS_3_01	Tambah data karyawan	Super Admin, Admin kepegawaian
42	SRS_3_02	Hapus data karyawan	Super Admin, Admin kepegawaian
43	SRS_3_03	Ubah data karyawan	Super Admin, Admin kepegawaian

44	SRS_3_04	Lihat data karyawan	Super Admin, Admin kepegawaian
45	SRS_3_05	Tambah data jabatan	Super Admin, Admin kepegawaian
46	SRS_3_06	Hapus data jabatan	Super Admin, Admin kepegawaian
47	SRS_3_07	Ubah data jabatan	Super Admin, Admin kepegawaian
48	SRS_3_08	Lihat data jabatan	Super Admin, Admin kepegawaian
49	SRS_3_09	Tambah jabatan fungsional	Super Admin, Admin kepegawaian
50	SRS_3_10	Hapus jabatan fungsional	Super Admin, Admin kepegawaian
51	SRS_3_11	Ubah jabatan fungsional	Super Admin, Admin kepegawaian
52	SRS_3_12	Lihat jabatan fungsional	Super Admin, Admin kepegawaian
53	SRS_1_01	Login	Super Admin, Administratif, Admin kepegawaian, Admin Persuratan, Admin Pelayanan, Kepala UPT
54	SRS_1_04	Logout	Super Admin, Administratif, Admin Kepegawaian, Admin Persuratan, Admin Pelayanan, Kepala UPT
55	SRS_8_01	Tambah User	Super Admin
56	SRS_8_02	Hapus User	Super Admin
57	SRS_8_03	Ubah User	Super Admin
58	SRS_8_04	Lihat User	Super Admin

Tabel 4.3 Spesifikasi Kebutuhan Non-fungsional

Kebutuhan Non-fungsional	
Reliability	Sistem dapat diakses selama berada dalam jangkauan jaringan LAN di UPT Laboratorium Kesehatan Hewan Malang

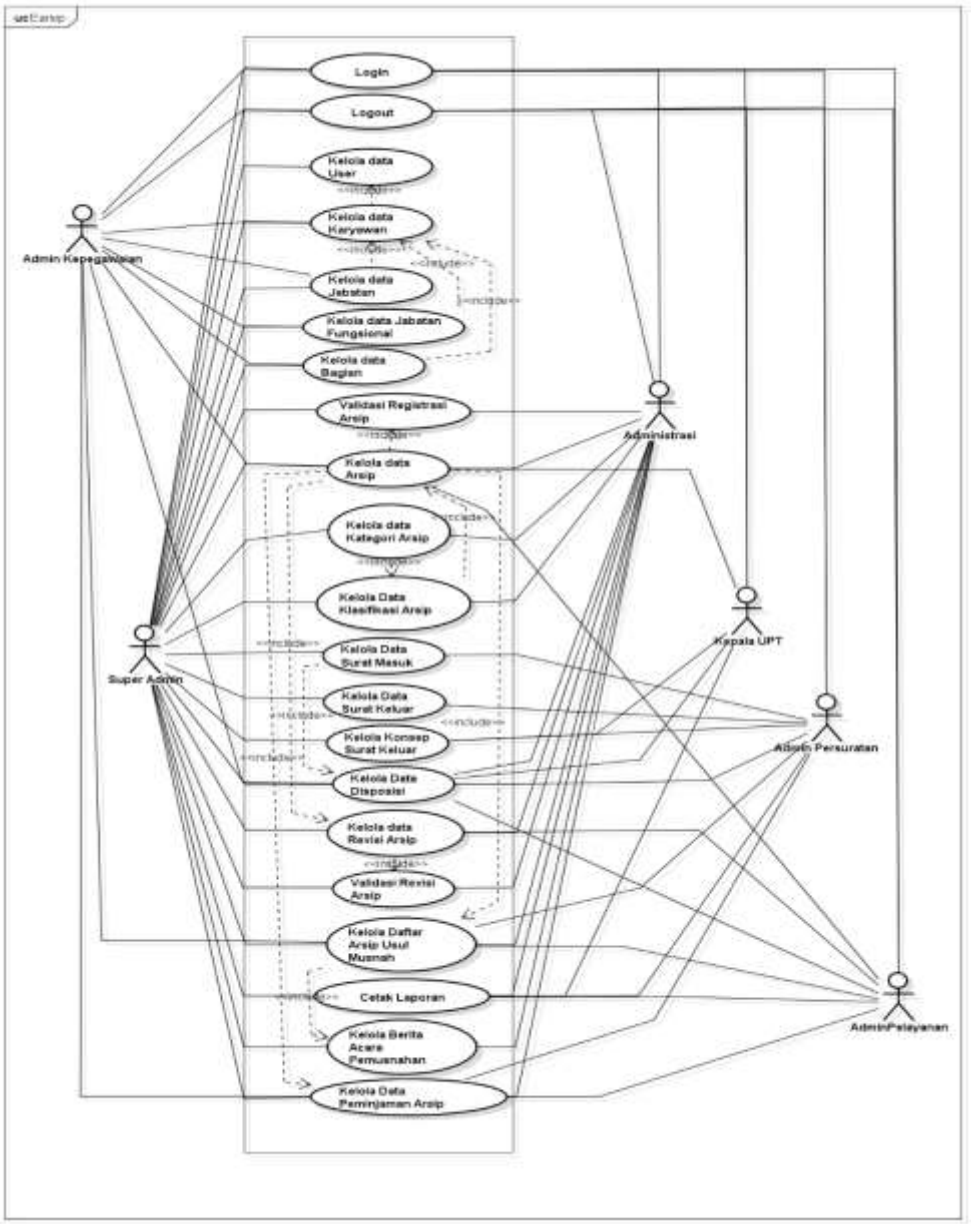
4.6. Analisis Kebutuhan Sistem

Proses analisis kebutuhan dilakukan dengan mengacu pada standar ISO 15489 yang mengatur tentang *record* arsip yang akan disesuaikan dengan proses bisnis yang sedang berjalan di kantor UPT Laboratorium Kesehatan Hewan

Malang. Proses analisis kebutuhan dilakukan dengan cara menggambarkan proses bisnis instansi dalam bentuk *Use Case Diagram* dan *Activity Diagram*. Analisis kebutuhan ini bertujuan untuk menggambarkan kebutuhan-kebutuhan yang harus disediakan sistem agar memenuhi kebutuhan pengguna.

Use Case diagram perancangan sistem manajemen arsip elektronik yang mengacu pada standar sistem pencatatan ISO 15489 ditunjukkan pada gambar 4.8.

4.6.1. *Use Case Diagram* Sistem Manajemen Arsip Elektronik



Gambar 4.8 Usecase Diagram Sistem Manajemen Arsip Elektronik

Dalam *Use Case* digambarkan setiap proses yang terdapat dalam sistem sesuai dengan aktor yang dapat mengelola proses tersebut. Setiap proses pada *Use Case* kepegawaian dijelaskan dalam skenario *Use Case* pada tabel 4.4 s.d 4.13.

4.6.2. *Skenario Use Case*

1. Skenario *Use Case* Kelola Data User

Tabel 4.4 Skenario *Use Case* Kelola Data User

Nomor SRS :	**SRS_8_01, SRS_8_02, SRS_8_03, SRS_8_04**	
Nama *Use Case* :	**Kelola User**	
Aktor :	Super Admin	
Deskripsi :	*Use Case* ini digunakan untuk mendeskripsikan kegiatan mengelola data User dan memberikan hak akses berdasarkan *privilege*nya.	
Kondisi Awal :	Aktor login sebagai super admin kelola User sebelum melakukan aktivitas mengelola data User. Data User yang dikelola hanya yang terdaftar pada data karyawan.	
Skenario Normal :	Aksi Aktor :	Reaksi Sistem :
	Memasukkan Data User 1. Super Admin masuk ke modul User. 4. Super Admin memilih menu tambah User. 6. Super Admin memasukkan data User. 7. Super Admin menekan tombol “simpan”.	2. Sistem menampilkan modul User. 3. Sistem menampilkan daftar User. 5. Sistem menampilkan form tambah User. 8. Sistem menyimpan data Aktor yang baru dan memberikan status penyimpanan. 9. Sistem menampilkan data User yang baru tersebut di daftar User

	Mengubah Data User 1. Aktor masuk ke modul User.	
		2. Sistem menampilkan modul User. 3. Sistem menampilkan daftar User.
	4. Aktor memilih data User yang akan di ubah. 5. Aktor menekan tombol aksi “edit”.	
		6. Sistem menampilkan pesan validasi hapus data
	7. Aktor mengubah data User pada form ubah data. 8. Aktor menekan tombol “simpan” pada data User yang telah di ubah.	
		9. Sistem menampilkan pesan bahwa data berhasil disimpan
	Menghapus Data User 1. Aktor masuk ke modul User.	
		2. Sistem menampilkan modul User. 3. Sistem menampilkan daftar User.
	4. Aktor memilih data User yang akan di hapus. 5. Aktor menekan tombol aksi “hapus”.	
		6. Sistem menampilkan pesan validasi hapus data
	7. Aktor menekan tombol “YA”	
		8. Sistem menampilkan pesan bahwa data berhasil dihapus.
	Validasi Data User 1. Aktor masuk ke dalam sistem.	

	3. Aktor menekan tombol aksi detail notifikasi validasi data arsip. 6. Aktor memeriksa dan mengubah status data arsip 8. Aktor menekan tombol "YA" **Lihat Data User** 1. Aktor masuk ke modul User. 4. Aktor memilih data User dan menekan tombol aksi detail.	2. Sistem menampilkan notifikasi validasi data user. 4. Sistem menampilkan data arsip yang akan di validasi. 5. Sistem menampilkan detail data arsip. 7. Sistem menampilkan pesan validasi. 9. Sistem menyimpan data validasi arsip yang baru dan memberikan informasi status penyimpanan 2. Sistem menampilkan modul User. 3. Sistem menampilkan data User. 5. Sistem menampilkan detail data User.
Skenario Alternatif :	Aksi Aktor :	Reaksi Sistem :
	Memasukkan Data User 1. Super Admin masuk ke modul User. 4. Super Admin memilih menu tambah User.	2. Sistem menampilkan modul User. 3. Sistem menampilkan daftar User. 5. Sistem menampilkan form tambah User.

	6. Super Admin memasukkan data User. 7. Super Admin menekan tombol “simpan”.	8. Memeriksa valid tidaknya data masukkan. 9. Menampilkan pesan bahwa data masukkan tidak valid.
	10. Memperbaiki data masukkan yang tidak valid.	11. Memeriksa valid tidaknya data masukkan. 12. Menyimpan data User. 13. Menampilkan pesan “data berhasil disimpan”.
	Mengubah Data User 1. Aktor masuk ke modul User.	2. Sistem menampilkan modul User.
	4. Aktor memilih data User yang akan di ubah. 5. Aktor menekan tombol aksi “edit”.	3. Sistem menampilkan daftar User.
	7. Aktor mengubah data User pada form ubah data. 8. Aktor menekan tombol “simpan” pada data User yang telah di ubah.	6. Sistem menampilkan data yang telah diubah dan disimpan.
	11. Memperbaiki data masukkan yang diubah dan tidak valid.	9. Memberikan valid tidaknya data masukkan. 10. Menampilkan pesan bahwa data masukan tidak valid.

		12. Memeriksa valid tidaknya data masukkan. 13. Menyimpan data User. 14. Menampilkan pesan "data berhasil diubah".
Kondisi Akhir :	Jika *Use Case* sukses, aktor bisa masuk ke sistem untuk melakukan proses tambah data User baru, ubah data User, validasi data User, hapus data User dan lihat detil data User.	

2. Skenario *Use Case Login*

Tabel 4.5 Skenario *Use Case* Login

Nomor SRS :	**SRS_1_01**	
Nama *Use Case* :	**Login**	
Aktor :	Super Admin, Kepala UPT, Administratif, Admin Kepegawaian, Admin Persuratan, Admin Kegiatan, Admin Pelayanan.	
Deskripsi :	*Use Case* ini digunakan untuk mendeskripsikan kegiatan autentikasi *user* sebelum masuk ke system berdasarkan *user privilege*nya.	
Kondisi Awal :	Aktor belum melakukan login, sehingga belum bisa masuk ke sistem	
Skenario Normal :	Aksi Aktor :	Reaksi Sistem :
	1. Memasukkan data berupa username dan password.	2. Mengecek valid tidaknya data masukan dengan memeriksa ke tabel user. Kemudian masuk kedalam sistem
Skenario Alternatif :	Alternatif ke -1	
	Aksi Aktor :	Reaksi Sistem :
	1. Memasukkan data berupa username dan password yang salah.	2. Memeriksa verifikasi tidaknya data masukan. 3. Sistem akan menampilkan pesan kesalahan, aktor diminta untuk

	4. Masukkan ulang username dan password dengan benar. 5. Kemudian klik login.	memasukkan ulang username dan password. 6. Memeriksa valid tidaknya data masukkan. 7. Apabila valid akan masuk kedalam sistem.
	Alternatif ke – 2	
	Aksi Aktor :	Reaksi Sistem :
	1. Klik lupa password 3. Isi form lupa password dengan menggunakan nip, password baru dan alamat email yang telah terdaftar. 4. Kemudian klik kirim.	**2.** Menampilkan Form lupa password. 5. Memeriksa valid tidaknya data masukkan. 6. Apabila valid akan masuk akan muncul pesan “data terkirim”. Kemudian menunggu hingga data telah divalidasi oleh Super Admin. 7. Apabila tidak valid maka akan muncul pesan “data yang anda masukkan tidak terdaftar, Mohon permintaan baru password baru ke Super Admin”.
Kondisi Akhir :	Jika *Use Case* sukses, Aktor bisa masuk ke sistem untuk melakukan kegiatannya berdasarkan hak akses yang telah diberikan.	

3. Skenario *Use Case* Kelola Data Arsip

Tabel 4.6 Skenario *Use Case* Kelola Data Arsip

Nomor SRS :	**SRS_4_01, SRS_4_02, SRS_4_03, SRS_3_04**	
Nama *Use Case* :	**Kelola Data Arsip**	
Aktor :	Super Admin, Admin Kepegawaian, Admin Kegiatan, Admin Pelayanan.	
Deskripsi :	*Use Case* ini digunakan untuk mendeskripsikan kegiatan mengelola seluruh data arsip.	
Kondisi Awal :	Kegiatan pengelolaan data arsip dapat dilakukan ketika ada hasil pelaporan dari proses yang yang telah dikerjakan dalam bagian. Aktor login sebagai admin sesuai haik aksesnya, kemudian memilih menu arsip.	
Skenario Normal :	Aksi Aktor :	Reaksi Sistem :
	Memasukkan Data Arsip 1. Aktor masuk ke modul arsip. 4. Aktor memilih menu tambah arsip. 6. Aktor memasukkan data arsip. 7. Aktor menekan tombol “simpan”. **Mengubah Data Arsip** 1. Aktor masuk ke modul arsip.	2. Sistem menampilkan modul arsip. 3. Sistem menampilkan daftar arsip. 5. Sistem menampilkan form tambah arsip. 8. Sistem menyimpan data arsip yang baru dan memberikan status penyimpanan. 9. Sistem menampilkan data arsip yang baru tersebut di daftar arsip.

	4. Aktor memilih data arsip yang akan di ubah. 5. Aktor menekan tombol aksi “edit”. 7. Aktor mengubah data arsip pada form ubah data. 8. Aktor menekan tombol “simpan” pada data arsip yang telah di ubah. **Menghapus Data Arsip** 1. Aktor masuk ke modul arsip. 4. Aktor memilih data arsip yang akan di ubah. 5. Aktor menekan tombol aksi “hapus”. 7. Aktor menekan tombol “YA” **Cari Data Arsip** 1. Aktor masuk ke modul arsip. 4. Aktor mengetikkan keyword data arsip di form pencarian.	2. Sistem menampilkan modul arsip. 3. Sistem menampilkan daftar arsip 6. Sistem menampilkan form ubah data arsip. 9. Sistem menyimpan data arsip yang telah di ubah dan menampilkan status update data. 2. Sistem menampilkan modul arsip. 3. Sistem menampilkan daftar arsip. 6. Sistem menampilkan pesan validasi hapus data. 8. Sistem menampilkan pesan bahwa data berhasil dihapus. 2. Sistem menampilkan modul arsip. 3. Sistem menampilkan arsip.

	Lihat Data Arsip 1. Aktor masuk ke modul arsip 4. Aktor memilih data arsip dan menekan tombol aksi detail.	5. Sistem melakukan pencarian arsip sesuai dari keyword yang di masukkan. 6. Sistem menampilkan hasil pencarian. 2. Sistem menampilkan modul arsip. 3. Sistem menampilkan data arsip. 5. Sistem menampilkan detail data arsip
Skenario Alternatif :	Aksi Aktor :	Reaksi Sistem :
	Memasukkan Data Arsip 1. Aktor masuk ke modul arsip. 4. Aktor memilih menu tambah arsip. 6. Aktor memasukkan data arsip. 7. Aktor menekan tombol “simpan”. 9. Memperbaiki data masukkan sampai data yang dimasukkan valid	 2. Sistem menampilkan modul arsip. 3. Sistem menampilkan daftar arsip. 5. Sistem menampilkan form tambah arsip. 8. Sistem memberikan pesan valid tidaknya data masukkan. 10. Memeriksa valid tidaknya data masukkan.

		11. Menampilkan pesan "data berhasil disimpan". 12. Sistem menyimpan data arsip yang baru dan memberikan status penyimpanan. 13. Sistem menampilkan data arsip yang baru tersebut di daftar arsip.
	Mengubah Data Arsip 1. Aktor masuk ke modul arsip.	2. Sistem menampilkan modul arsip. 3. Sistem menampilkan daftar arsip
	4. Aktor memilih data arsip yang akan di ubah. 5. Aktor menekan tombol aksi "edit".	6. Sistem menampilkan form ubah data arsip.
	7. Aktor mengubah data arsip pada form ubah data. 8. Aktor menekan tombol "*edit*" pada data arsip yang telah di ubah.	9. Sistem memberikan pesan valid tidaknya data masukkan.
	10. Memperbaiki data masukkan sampai data yang dimasukkan valid.	11. Sistem memeriksa valid tidaknya data masukkan. 12. Sistem menampilkan pesan "data berhasil diubah". 13. Sistem menyimpan data arsip yang telah di ubah dan menampilkan status update data.
Kondisi Akhir :	Jika *Use Case* sukses, aktor bisa masuk ke sistem untuk melakukan proses tambah data arsip baru, ubah data arsip, hapus data arsip dan lihat detil data arsip.	

4. Skenario *Use Case* Kelola Daftar Usul Musnah Arsip

Tabel 4.7 Skenario *Use Case* Daftar Usul Musnah Arsip.

Nomor SRS :	SRS_4_11	
Nama *Use Case* :	Daftar Usul Musnah Arsip	
Aktor :	Super Admin, Admin Kepegawaian, Pelayanan.	
Deskripsi :	*Use Case* ini digunakan untuk mendeskripsikan kegiatan kelola daftar usul musnah arsip yang telah habis masa retensinya.	
Skenario Normal :	Aksi Aktor :	Reaksi Sistem :
	Validasi Retensi Arsip 1. Aktor masuk ke dalam sistem	2. Sistem menampilkan notifikasi daftar usul musnah arsip
	3. Aktor mengeklik notifikasi daftar usul musnah arsip	4. Sistem menampilkan list data daftar usul musnah yang akan di validasi.
	5. Aktor memeriksa dan memilih data arsip yang akan dimusnahkan atau ditambah retensinya.	**6.** Sistem menampilkan pesan validasi.
	7. Aktor menekan tombol "YA"	8. Sistem menyimpan data validasi daftar usul musnah arsip yang baru dan memberikan informasi status penyimpanan.
Kondisi Akhir :	Jika *Use Case* sukses, aktor bisa masuk ke sistem untuk melakukan validasi arsip usul musnah yang telah habis masa retensinya.	

5. Skenario *Use Case* Kelola Data Peminjaman Arsip

Tabel 4.8 Skenario *Use Case* Kelola Data Peminjaman Arsip

Nomor SRS :	**SRS_4_07, SRS_4_08, SRS_4_09**	
Nama *Use Case* :	**Kelola Data Peminjaman Arsip**	
Aktor :	Super Admin, Admin Kepegawaian, Admin Pelayanan, Admin Persuratan.	
Deskripsi :	*Use Case* ini digunakan untuk mendeskripsikan kegiatan pengelolaan data peminjaman arsip.	
Kondisi Awal :	Aktor login sebagai admin lalu memilih menu peminjaman.	
Skenario Normal :	Aksi Aktor :	Reaksi Sistem :
	Memasukkan Data Peminjaman 1. Aktor masuk ke modul peminjaman. 3. Aktor memilih menu tambah peminjaman. 5. Aktor memasukkan data peminjaman. 6. Aktor menekan tombol "simpan".	2. Sistem menampilkan menu peminjaman. 4. Sistem menampilkan form tambah peminjaman. 7. Sistem menyimpan data peminjaman yang baru dan memberikan status penyimpanan. 8. Sistem menampilkan data peminjaman yang baru tersebut di daftar peminjaman.
	Menghapus Data Peminjaman 1. Aktor masuk ke modul peminjaman. 4. Aktor memilih data peminjaman yang akan di hapus.	2. Sistem menampilkan modul peminjaman. 3. Sistem menampilkan daftar peminjaman.

	5. Aktor menekan tombol aksi “hapus”. 7. Aktor menekan tombol “YA” **Lihat Data Peminjaman** 1. Aktor masuk ke modul peminjaman. 3. Aktor memilih data peminjaman dan menekan tombol aksi detail	6. Sistem menampilkan pesan validasi hapus data 8. Sistem menampilkan pesan bahwa data berhasil dihapus. 2. Sistem menampilkan modul peminjaman. 4. Sistem menampilkan data peminjaman. 5. Sistem menampilkan detail data peminjaman.
Skenario Alternatif :	Aksi Aktor :	Reaksi Sistem :
	Memasukkan Data Peminjaman 1. Aktor masuk ke modul peminjaman. 4. Aktor memilih menu tambah peminjaman. 6. Aktor memasukkan data peminjaman. 7. Aktor menekan tombol “simpan”.	2. Sistem menampilkan modul peminjaman. 3. Sistem menampilkan daftar peminjaman. 5. Sistem menampilkan form tambah karyawan. 8. Sistem memeriksa valid tidaknya data masukkan. 9. Sistem mengeluarkan pesan bahwa data masukkan tidak valid.

	10. Aktor memperbaiki data masukkan yang tidak valid.	11. Sistem memeriksa valid tidaknya data masukkan. 12. Menyimpan data peminjaman. 13. Menampilkan pesan "data berhasil disimpan".
Kondisi Akhir :	Jika *Use Case* sukses, aktor bisa masuk ke sistem untuk melakukan proses tambah data peminjaman baru, ubah data peminjaman, cari data peminjaman, hapus data peminjaman dan lihat detail data peminjaman.	

6. Skenario *Use Case* Kelola Disposisi Surat Masuk

Tabel 4.9 Skenario Kelola Disposisi Surat Masuk

Nomor SRS :	**SRS_7_01**	
Nama *Use Case* :	**Kelola Disposisi Surat Masuk**	
Aktor :	Kepala UPT	
Deskripsi :	*Use Case* ini digunakan untuk mendeskripsikan kegiatan mengetahui disposisi surat masuk yang telah diinputkan oleh bagian persuratan.	
Kondisi Awal :	Aktor login sebagai Kepala UPT, aktor menerima notifikasi data disposisi surat masuk yang telah diinputkan oleh bagian persuratan.	
Skenario Normal :	Aksi Aktor :	Reaksi Sistem :
	Mengetahui Disposisi Surat Masuk 1. Aktor masuk ke dalam sistem	2. Sistem menampilkan notifikasi disposisi surat masuk
	3. Aktor memilih notifikasi disposisi surat masuk	4. Sistem menampilkan data disposisi surat masuk untuk diketahui. 5. Sistem menampilkan detail data disposisi surat masuk.
	6. Aktor memeriksa dan mengubah status data disposisi surat masuk	

		7. Sistem menyimpan data disposisi surat masuk yang telah diketahui oleh kepala upt.
Kondisi Akhir :	Jika *Use Case* sukses, aktor bisa masuk ke sistem untuk mengetahui disposisi surat masuk yang telah diinputkan oleh admin bagian persuratan.	

7. Skenario *Use Case* Validasi Registrasi Arsip

Tabel 4.10 Skenario *Use Case* Validasi Registrasi Arsip

Nomor SRS :	**SRS_6_09, SRS_6_10**	
Nama *Use Case* :	**Validasi Registrasi Arsip**	
Aktor :	Super Admin, Administratif	
Deskripsi :	*Use Case* ini digunakan untuk mendeskripsikan kegiatan validasi data revisi arsip yang telah di inputkan oleh administratif.	
Kondisi Awal :	Aktor login sebagai Administratif, aktor menerima notifikasi validasi registrasi arsip yang telah di inputkan oleh admin tiap bagian.	
Skenario Normal :	Aksi Aktor :	Reaksi Sistem :
	Validasi Data Registrasi Arsip 1. Aktor masuk ke dalam sistem 3. Aktor mengeklik notifikasi validasi registrasi arsip.	 2. Sistem menampilkan notifikasi validasi registrasi arsip 4. Sistem menampilkan registrasi arsip yang akan di validasi.

	6. Aktor memeriksa dan mengubah status data registrasi arsip. 8. Aktor menekan tombol "YA"	5. Sistem menampilkan detail data arsip. 7. Sistem menampilkan pesan validasi. 9. Sistem menyimpan data validasi registrasi data arsip yang baru dan memberikan informasi status penyimpanan.
Kondisi Akhir :	Jika *Use Case* sukses, aktor bisa masuk ke sistem untuk melakukan validasi data registrasi arsip yang telah diinputkan oleh tiap admin bagian.	

8. Skenario *Use Case* Kelola Berita Acara

Tabel 4.11 Skenario *Use Case* kelola Data Berita Acara

Nomor SRS :	**SRS_6_10**	
Nama *Use Case* :	**Kelola Data Berita Acara**	
Aktor :	Super Admin, Admnistratif	
Deskripsi :	*Use Case* ini digunakan untuk mendeskripsikan kegiatan pengelolaan data berita acara.	
Skenario Normal :	Aksi Aktor :	Reaksi Sistem :
	Tambah Berita Acara 1. Aktor masuk ke modul berita acara 3. Aktor menekan button tambah data 5. Aktor memasukkan data berita acara 6. Aktor menekan *button* simpan	 2. Sistem menampilkan data berita acara 4. Sistem menampilkan form tambah data berita acara

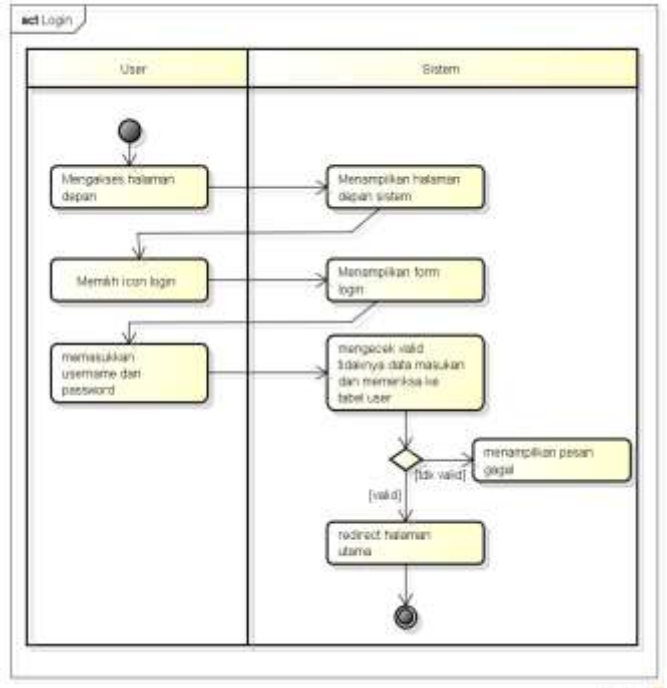
	Lihat Data Berita Acara 1. Aktor masuk ke modul berita acara. 3. Aktor memilih data berita acara dan menekan tombol aksi detail **Cetak Berita Acara** 7. Aktor memilih modul berita acara. 3. Aktor memilih data berita acara dan menekan tombol aksi *print*	7. Sistem menyimpan data berita acara. 8. Sistem menampilkan data berita acara yang baru tersebut di daftar berita acara. 2. Sistem menampilkan data berita acara. 4. Sistem menampilkan detail data berita acara. 2. Sistem menampilkan data berita acara. 4. Sistem menampilkan konfirmasi pencetakan berita acara.
Kondisi Akhir :	Jika *Use Case* sukses, aktor bisa masuk ke sistem untuk melakukan validasi arsip usul musnah yang telah habis masa retensinya.	

4.6.3. Perancangan *Activity Diagram*

Dibawah ini merupakan gambar *Activity Diagram* yang menggambarkan proses interaksi antata user dengan sistem manajemen arsip elektronik berdasarkan ISO 15489 di UPT Laboratorium Kesehatan Hewan Malang.

1. *Activity Diagram* Login

Activity Diagram Login merupakan detail proses dari *Use Case* login, dalam activity login pada gambar 4.13 berikut dijelaskan alur proses bagaimana user masuk ke dalam sistem.



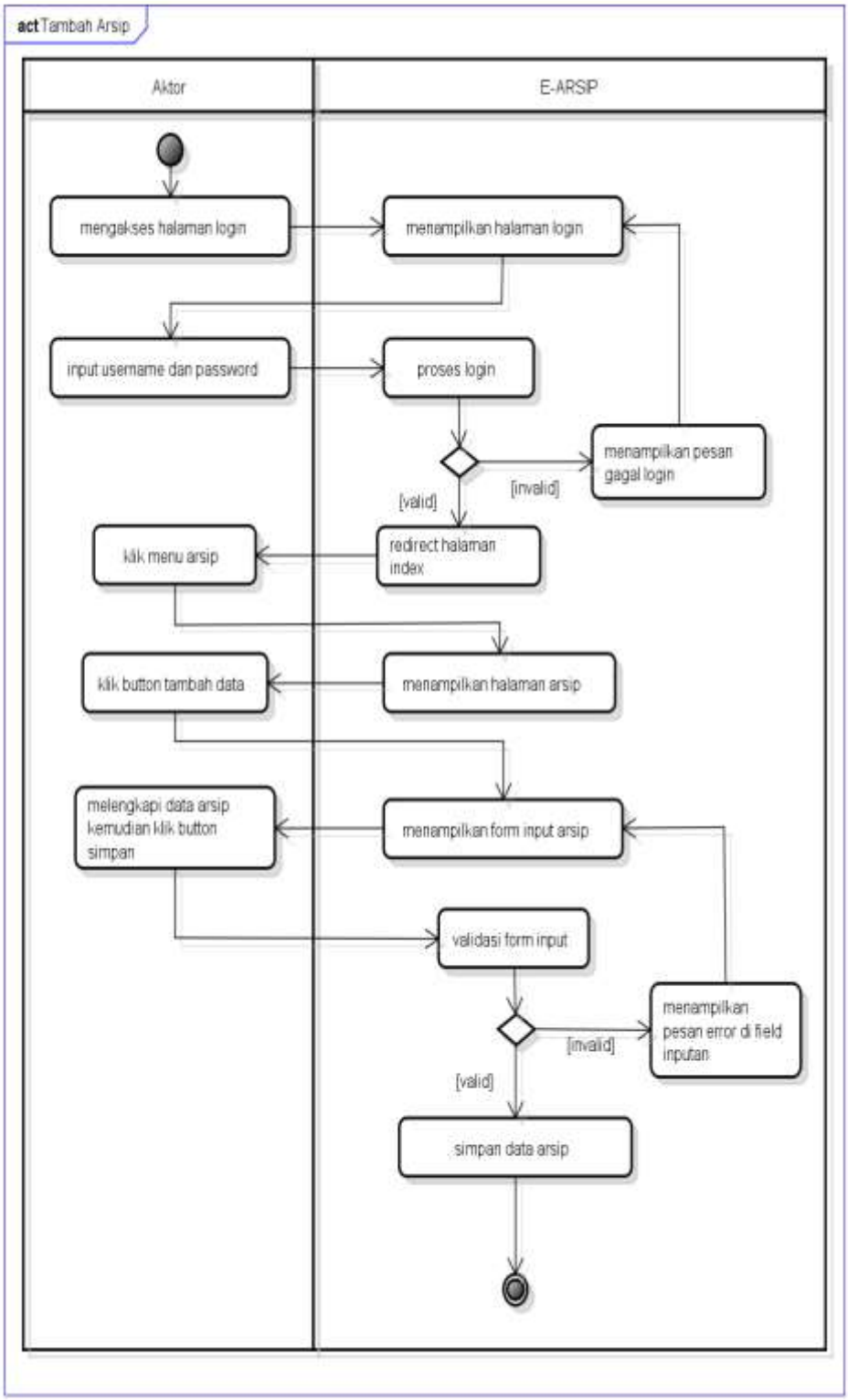
Gambar 4.9 *Activity Diagram* Login

2. *Activity Diagram* Tambah Data Arsip

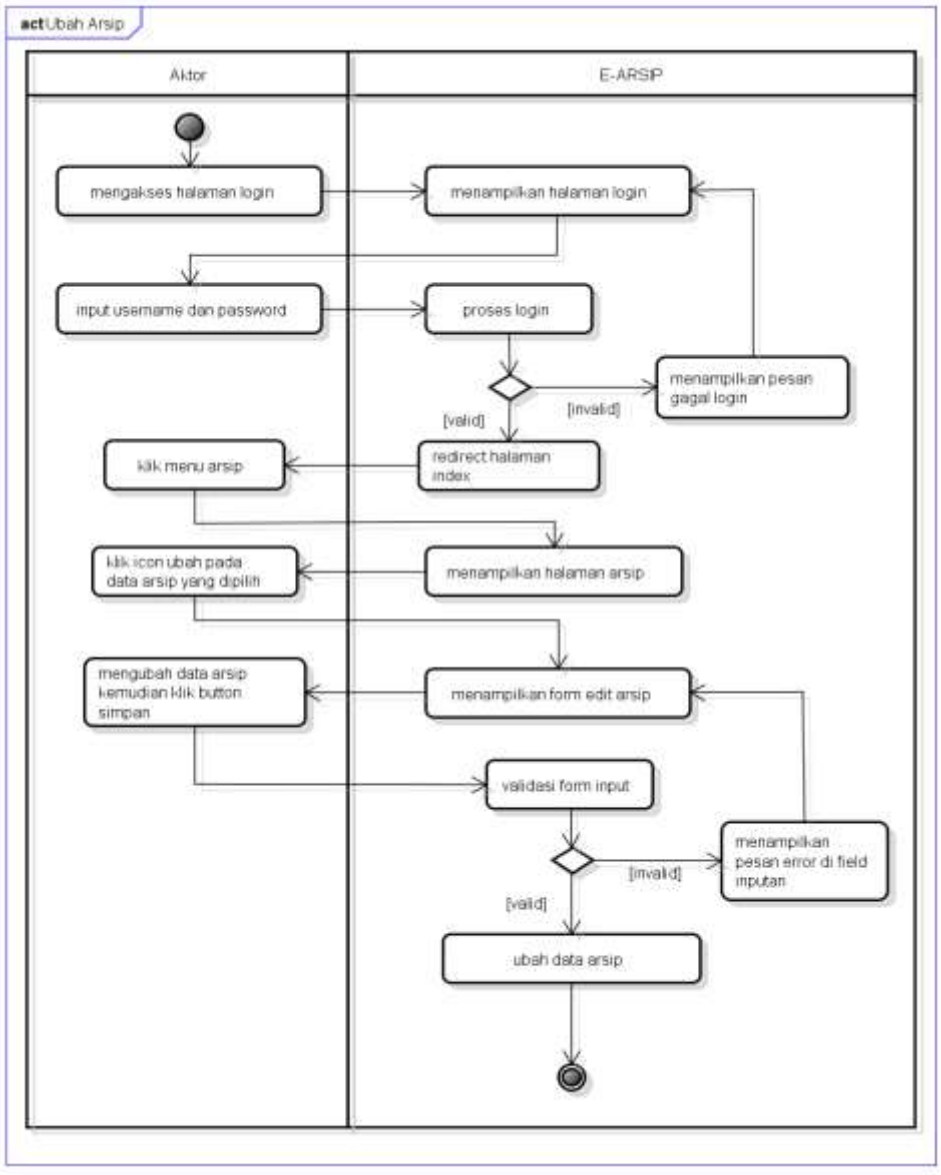
Activity Diagram tambah data arsip merupakan detail proses dari *Use Case* kelola data arsip, dalam *Activity Diagram* tambah data arsip pada gambar 4.10 berikut dijelaskan alur proses bagaimana admin dapat menambah data arsip mulai dari admin klik button tambah data sampai dengan data arsip disimpan dalam database sistem.

3. *Activity Diagram* Ubah Data Arsip

Activity Diagram ubah data arsip merupakan detail proses dari *Use Case* kelola data arsip, dalam *Activity Diagram* ubah data arsip pada gambar 4.11 berikut dijelaskan alur proses bagaimana admin dapat mengubah data arsip mulai dari admin klik button ubah pada halaman data arsip yang dipilih sampai dengan perubahan data arsip disimpan dalam database sistem.



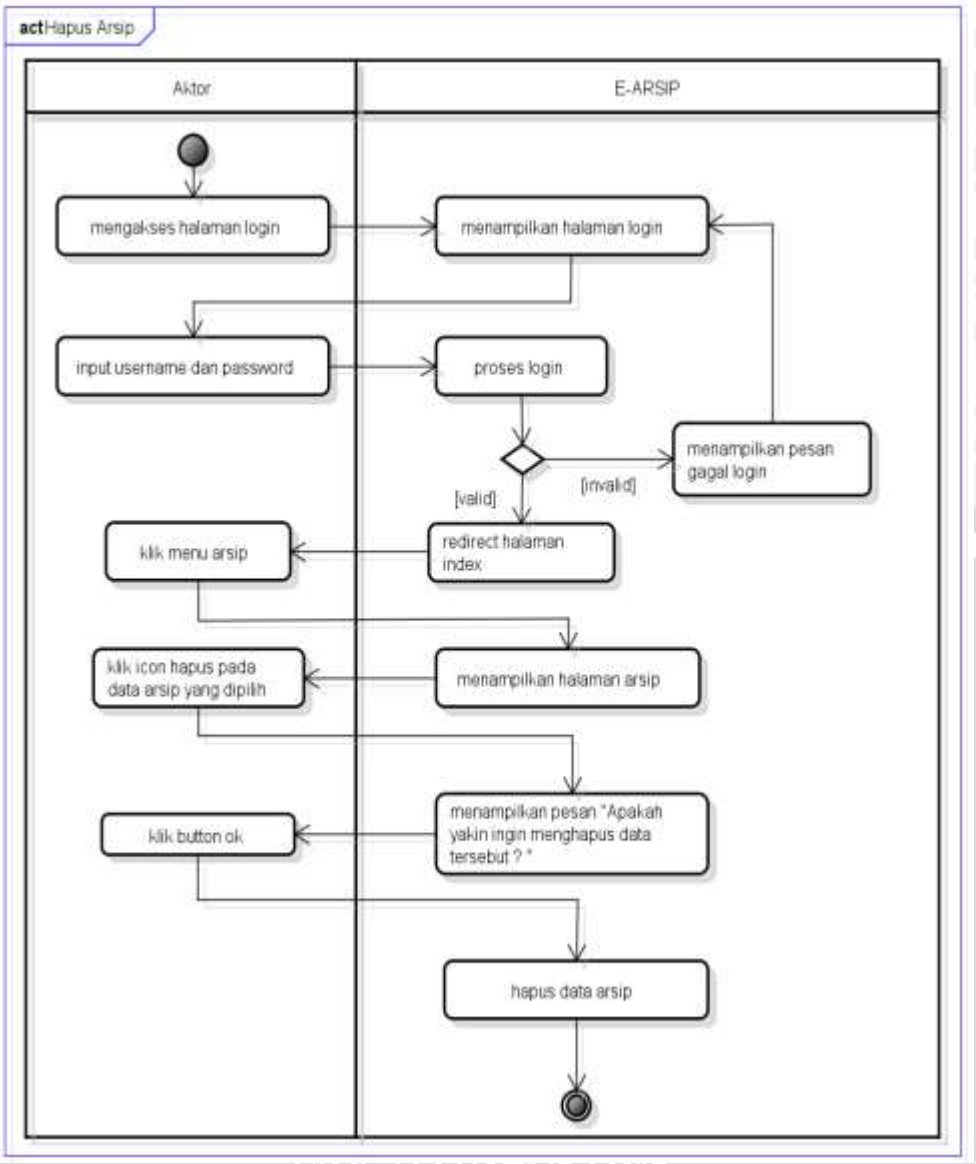
Gambar 4.10 *Activity Diagram* Tambah Data Arsip



Gambar 4.11 *Activity Diagram* Ubah Data Arsip

4. *Activity Diagram Hapus Data Arsip*

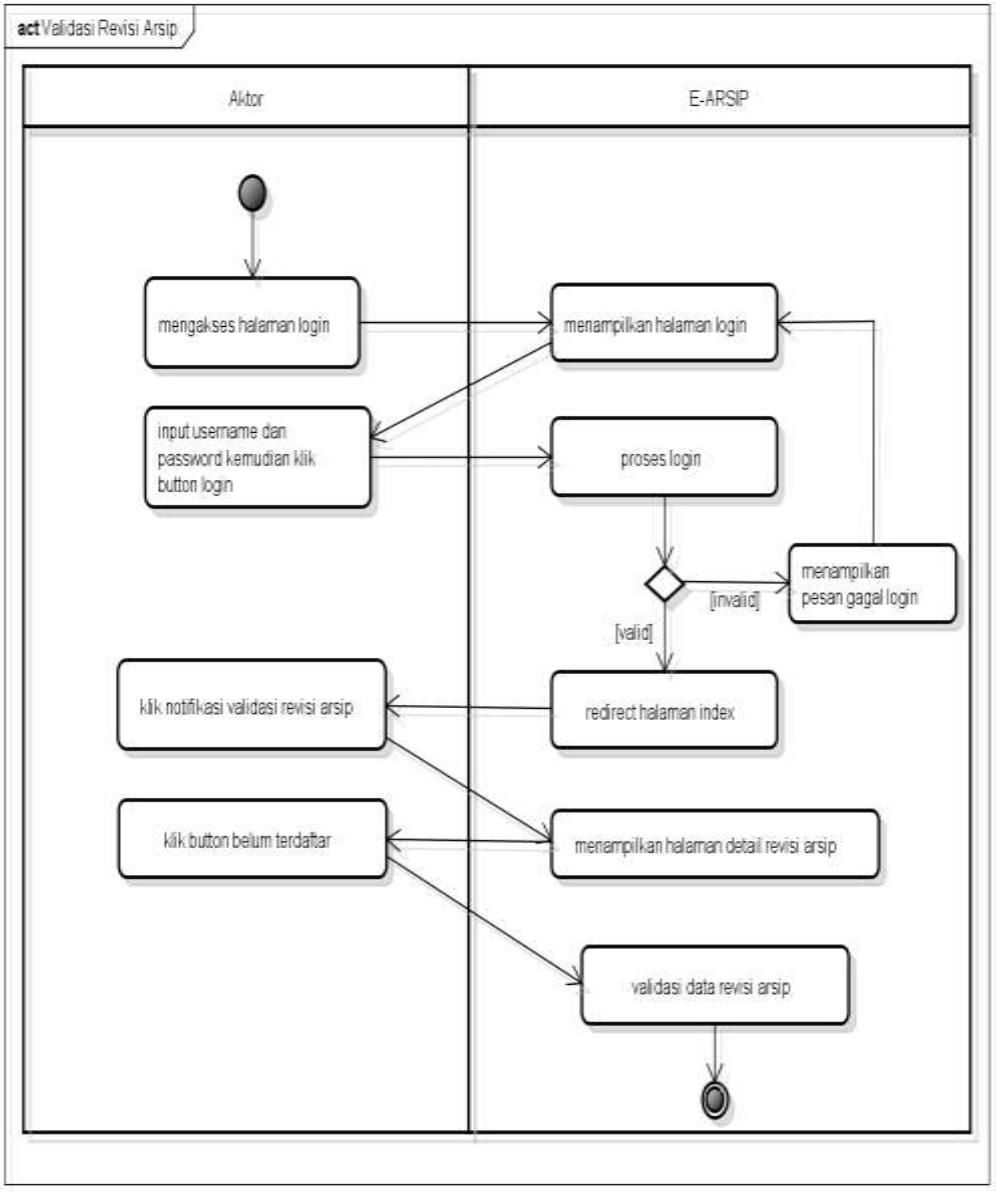
Activity Diagram hapus data arsip merupakan detail proses dari *Use Case* kelola data arsip, dalam *Activity Diagram* hapus data arsip pada gambar 4.13 berikut dijelaskan alur proses bagaimana kepala UPT dapat menghapus data arsip mulai dari menekan icon hapus sampai data arsip terhapus dari database



Gambar 4.12 *Activity Diagram* Hapus Data Arsip

5. *Activity Diagram* Validasi Revisi Arsip

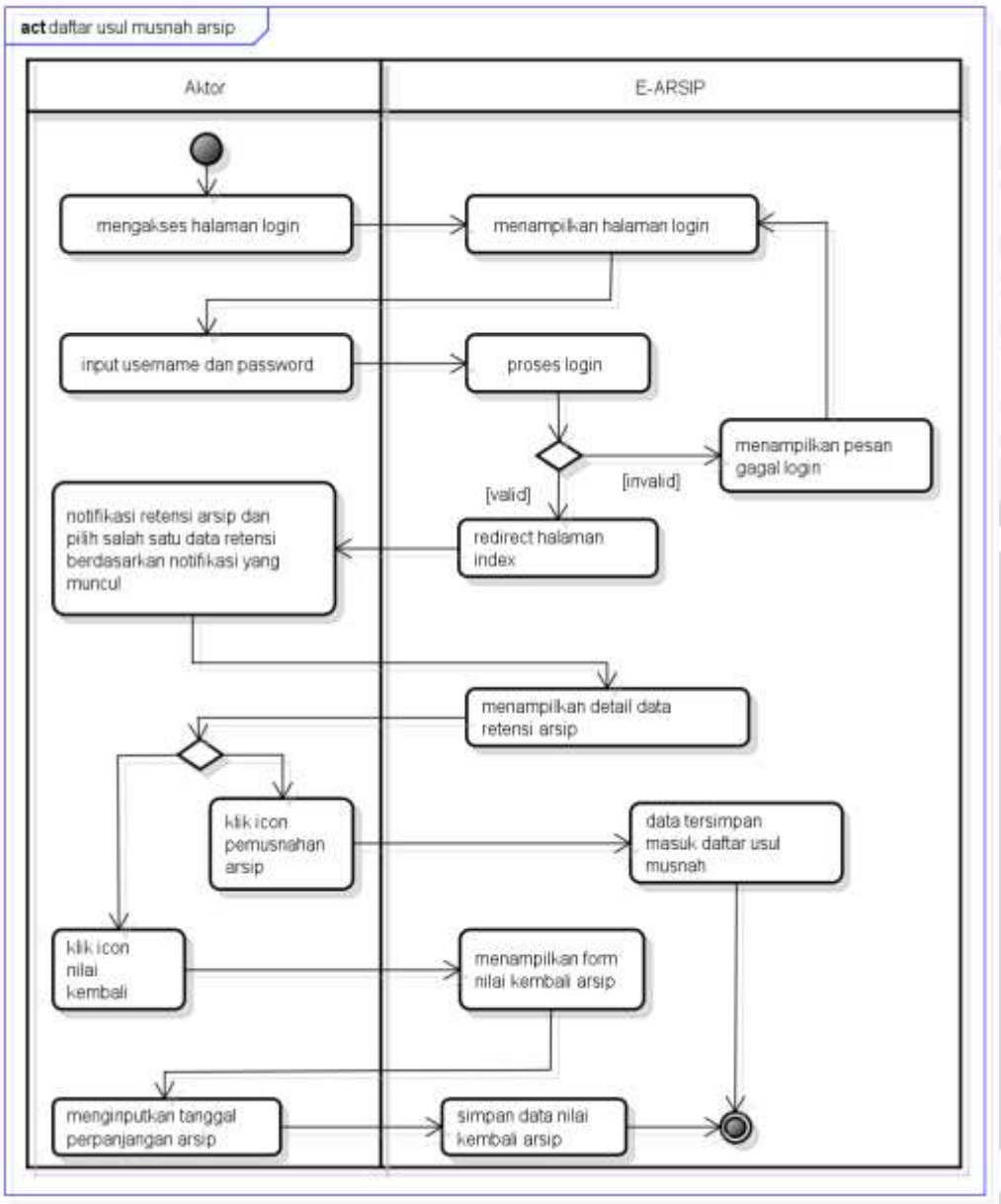
Activity Diagram validasi revisi arsip merupakan detail proses dari *Use Case* validasi revisi arsip, dalam *Activity Diagram* validasi revisi arsip pada gambar 4.13 berikut dijelaskan alur proses bagaimana kepala UPT dapat validasi revisi arsip mulai dari menerima notifikasi validasi revisi arsip kemudian data disimpan dalam database.



Gambar 4.13 *Activity Diagram* Validasi Revisi Arsip

6. *Activity Diagram* Kelola Daftar usul Musnah Arsip

Activity Diagram kelola daftar usul musnah arsip merupakan detail proses dari *Use Case* kelola data daftar usul musnah arsip, dalam *Activity Diagram* kelola daftar usul musnah arsip pada gambar 4.14 berikut dijelaskan alur proses bagaimana admin menerima notifikasi bahwa retensi arsip telah habis sampai keputusan arsip tersebut akan dimusnahkan atau ditambah masa simpannya.



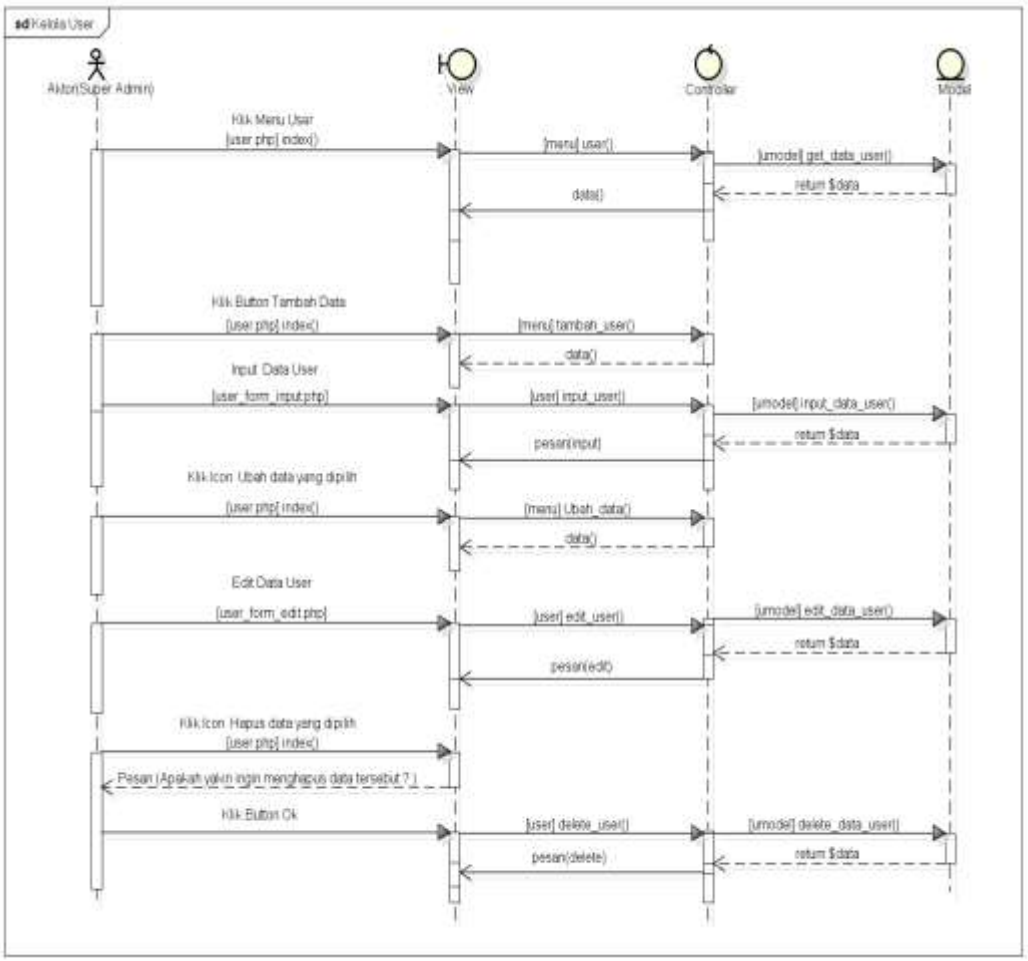
Gambar 4.14 Kelola Daftar Usul Musnah Arsip

4.6.4. Perancangan *Sequence Diagram*

Dibawah ini merupakan gambar *Sequence Diagram* untuk proses pada Sistem Manajemen Arsip Elektronik pada UPT Laboratorium Kesehatan Hewan Malang. Pada bagian ini, akan digambarkan alur jalannya sistem dari beberapa fitur yang ada pada sistem.

1. *Sequence Diagram* Kelola Data *User*

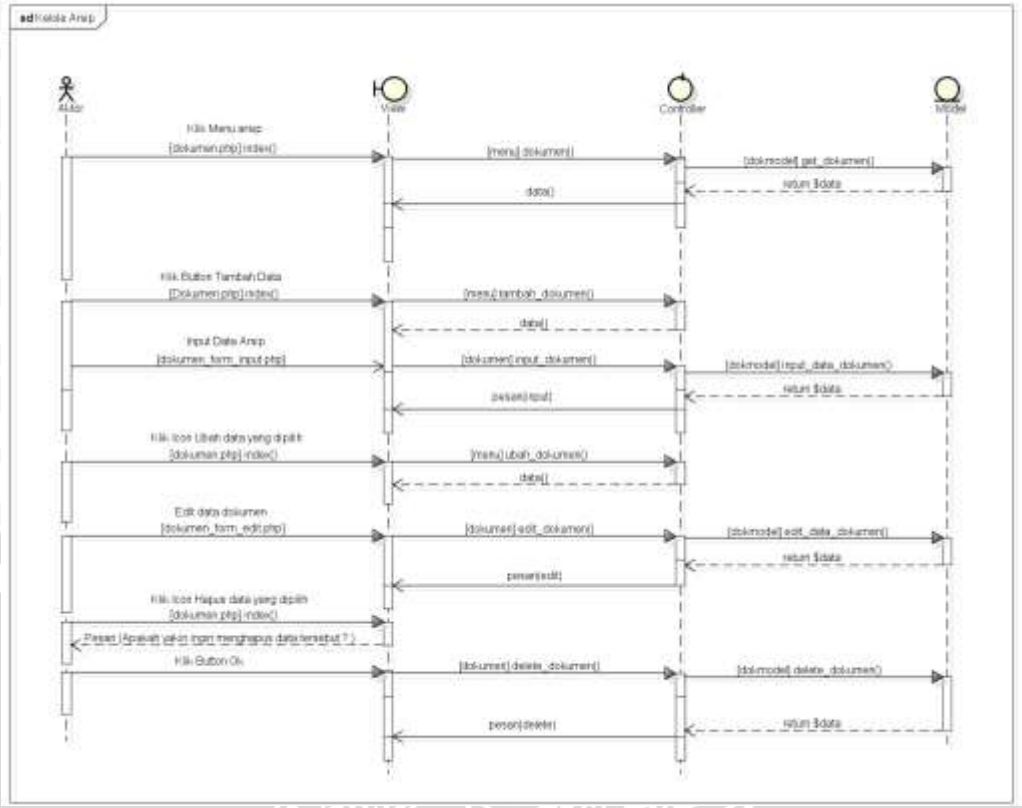
Sequence Diagram kelola data user ini melibatkan interaksi yang dimulai dari super admin sebagai aktor yang menggunakan view(antarmuka), controller menu dan controller user serta model yang diberi nama umodel. *Sequence Diagram* kelola data user dapat dilihat pada gambar 4.15



Gambar 4.15 *Sequence Diagram* Kelola Data User

2. *Sequence Diagram* Kelola Data Arsip

Sequence Diagram kelola data user ini melibatkan interaksi yang dimulai dari admin sebagai aktor yang menggunakan view(antarmuka), controller menu dan controller dokumen serta model yang diberi nama dokmodel. *Sequence Diagram* kelola data user dapat dilihat pada gambar 4.16



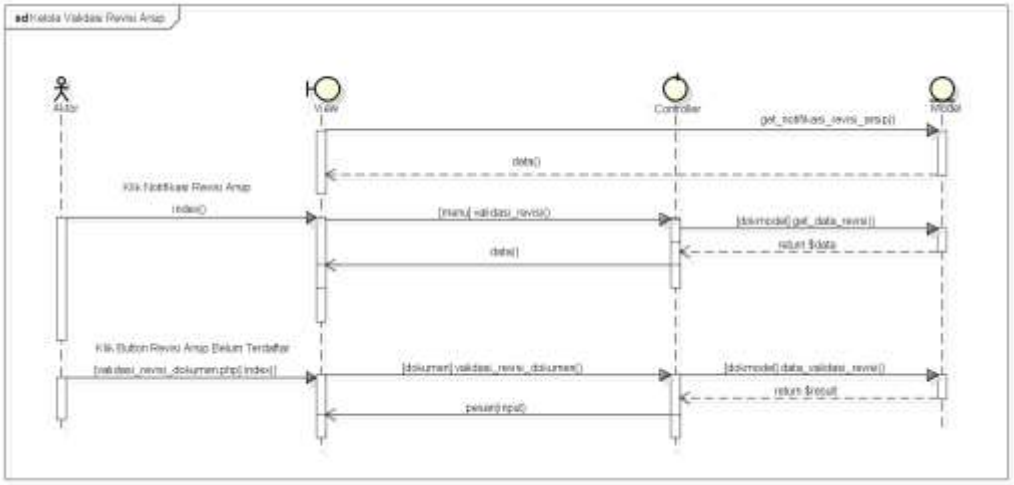
Gambar 4.16 *Sequence Diagram* Kelola Data Arsip

3. *Sequence Diagram* Validasi Revisi Arsip

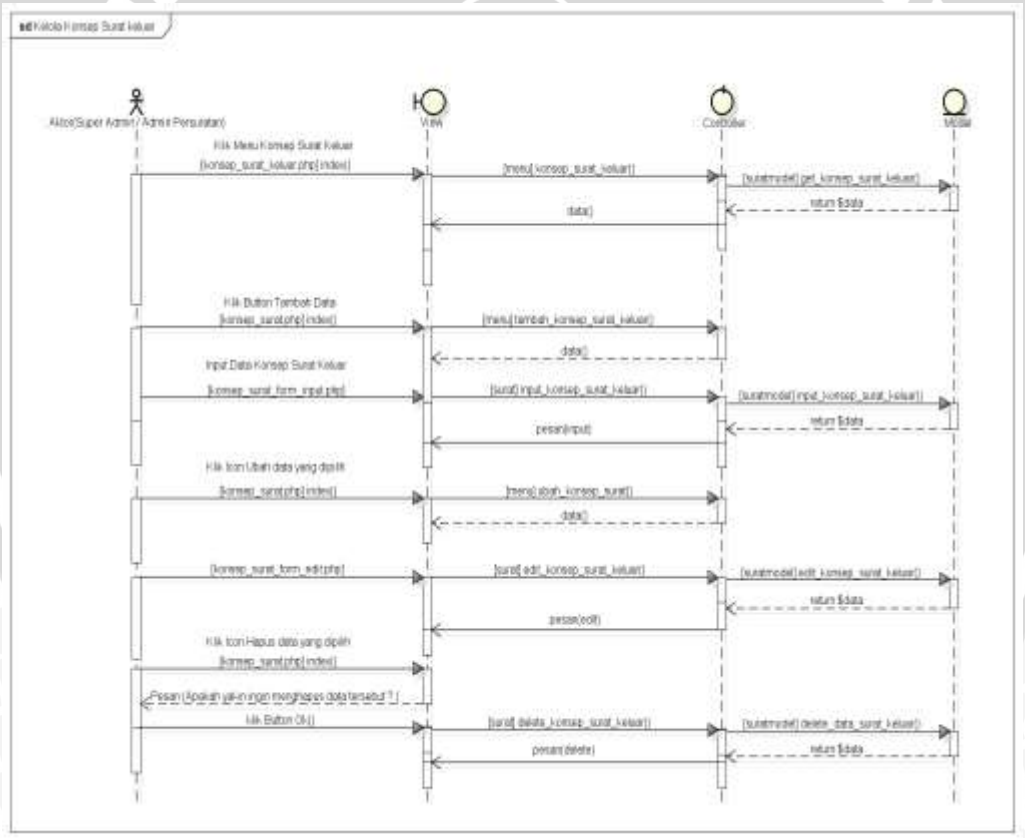
Sequence Diagram validasi revisi arsip pada gambar 4.17 melibatkan interaksi yang dimulai dari user sebagai aktor yang menggunakan view(antarmuka) validasi_revisi_dokumen.php, controller menu dan dokumen serta model yang diberi nama dokmodel.

4. *Sequence Diagram* Kelola Data Konsep Surat Keluar

Sequence Diagram pada gambar 4.18 merupakan *Sequence Diagram* yang menjelaskan proses kelola data konsep surat keluar. Dimulai dari aktor super admin dan bagian persuratan mengakses *form* antarmuka (*view*)konsep_surat_keluar dan melibatkan *controller* menu dan surat, serta model yang diberi nama suratmodel.



Gambar 4.17 *Sequence Diagam* Validasi Revisi Arsip

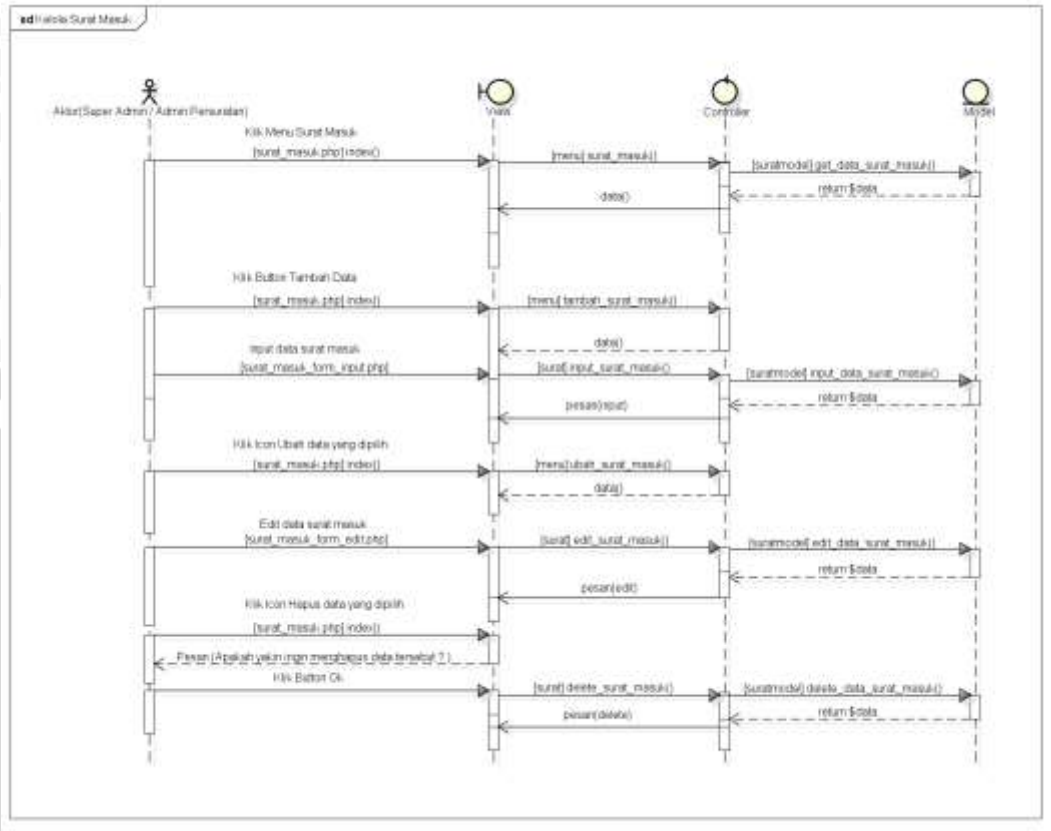


Gambar 4.18 *Sequence Diagram* Kelola Data Konsep Surat Keluar

5. *Sequence Diagram* Kelola Data Surat Masuk

Sequence Diagram pada gambar 4.19 merupakan *Sequence Diagram* yang menjelaskan proses kelola data surat masuk. Dimulai dari aktor persuratan

mengakses form antarmuka (*view)* surat_masuk.php, *controller* menu dan *controller* surat, serta model yang diberi nama suratmodel.



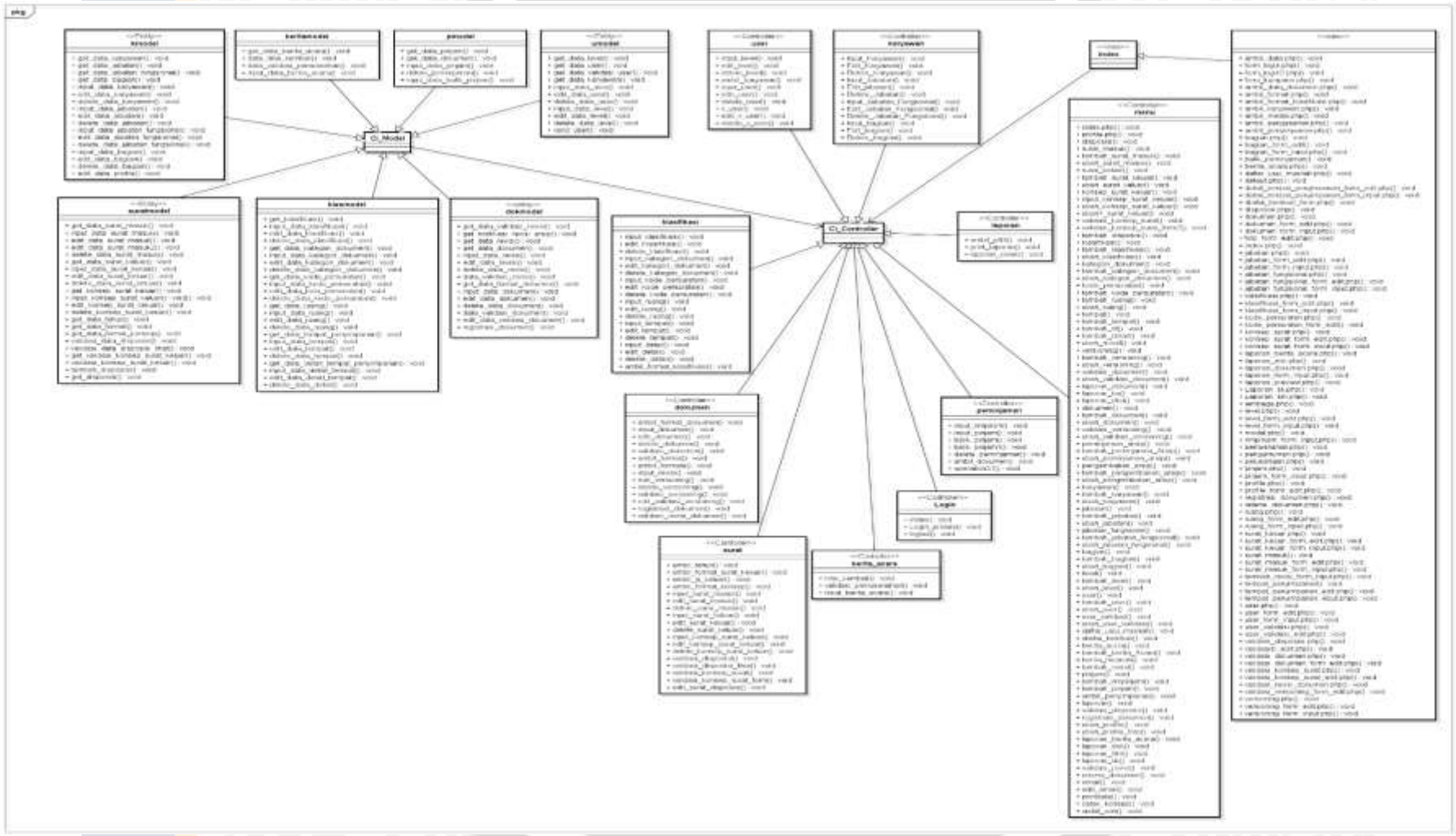
Gambar 4.19 *Sequence Diagram* Kelola Data Surat Masuk

4.6.5. Perancangan *Class diagram*

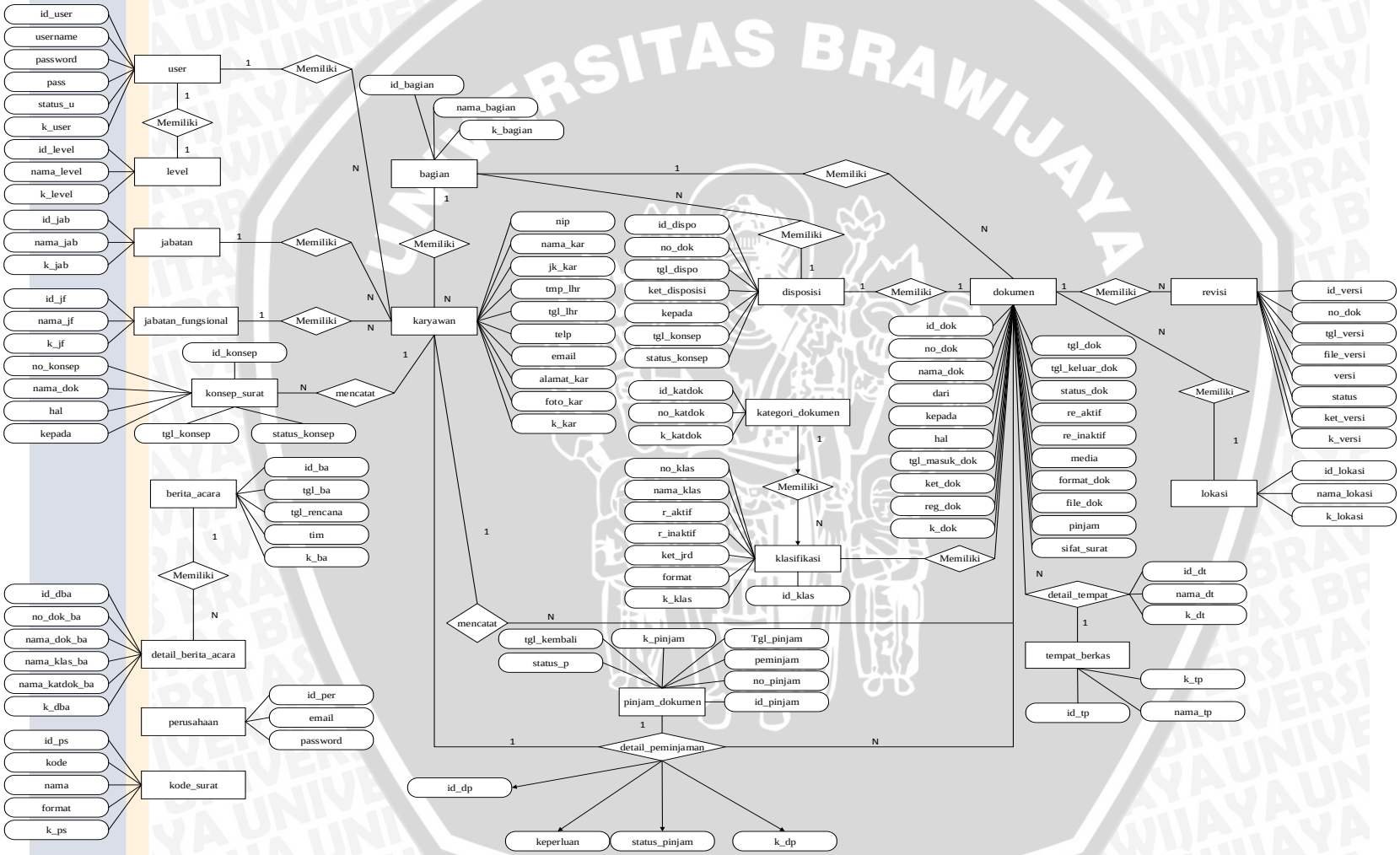
Perancangan *Class diagram* ini dilakukan untuk merancang *Class* yang nantinya akan diimplementasikan pada framework CodeIgniter yang menerapkan pola MVC. Perancangan *Class Diagram* sistem manajemen arsip elektronik dapat dilihat pada gambar 4.20

4.6.6. Perancangan *Entity Relationship Diagram (ERD)*

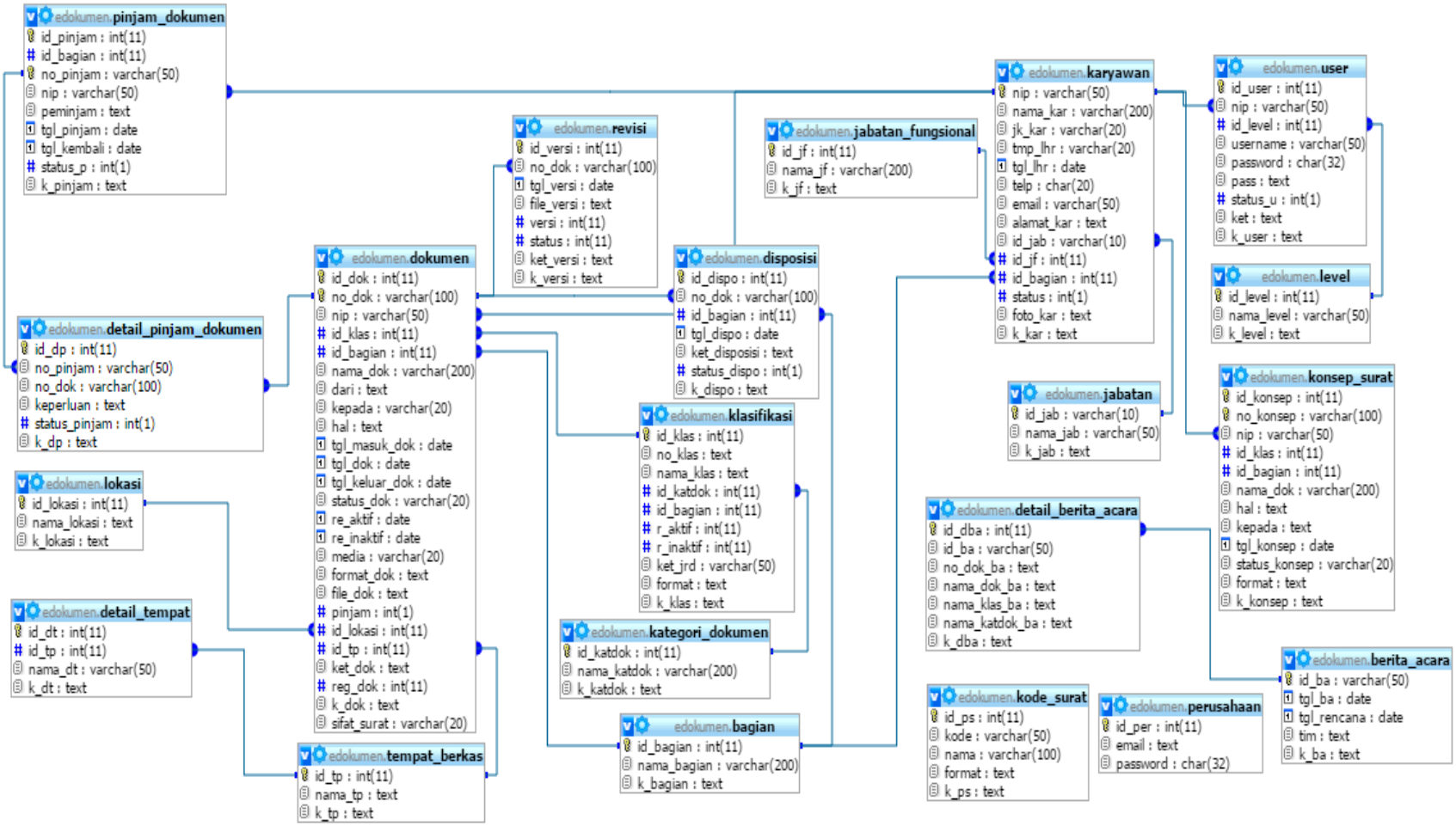
ERD akan menggambarkan hubungan entitas satu dengan lainnya. Secara umum ERD dari struktur basis data pada Sistem Manajemen Arsip Elektronik ditunjukkan dalam gambar 4. 21



Gambar 4.20 *Class Diagram* Sistem Manajemen Arsip Elektronik



Gambar 4.21 *Entity Relationship Diagram* Sistem Manajemen Arsip Elektronik



Gambar 4.22 Rancangan Database Sistem Manajemen Arsip Elektronik

4.6.7. *Data Type Definition*

Data Type Definition menjelaskan secara rinci mengenai tipe data yang digunakan setiap kolom pada masing-masing tabel basis data sistem. Tabel-tabel yang terdapat dalam basis data sistem adalah :

1. Tabel bagian berisi id bagian dan kelengkapan data bagian yang digunakan untuk mendefinisikan bagian yang ada di UPT Laboratorium Kesehatan Hewan Malang. *Data Type Definition* dari tabel bagian ditunjukkan pada tabel 4.18.

Tabel 4.12 Tabel *Data Type Definition* dari tabel bagian

Field	Type	Null	Key	Default	Extra
id_bagian	int(11)	No	PK	*None*	AI
nama_bagian	varchar(200)	No		*None*	
k_bagian	text	No		*None*	

2. Tabel berita_acara berisi atribut berita acara yang digunakan untuk menyimpan data berita acara pemusnahan arsip.

Tabel 4.13 Tabel *Data Type Definition* dari tabel berita_acara

Field	Type	Null	Key	Default	Extra
id_ba	varchar(50)	No	PK	*None*	
tgl_ba	date	No		*None*	
tgl_rencana	date	No		*None*	
cara_musnah	text	No		*None*	
Tim	text	No		*None*	
k_ba	text	No		*None*	

3. Tabel detail_berita_acara berisi atribut detail berita acara yang digunakan untuk menyimpan data detail berita acara pemusnahan arsip.

Tabel 4.14 Tabel *Data Type Definition* dari tabel detail_berita_acara

Field	Type	Null	Key	Default	Extra
id_dba	int(11)	No	PK	*None*	AI
id_ba	varchar(50)	No		*None*	

no_dok_ba	text	No		*None*	
nama_dok_ba	text	No		*None*	
nama_klas_ba	text	No		*None*	
nama_katdok_ba	text	No		*None*	
k_dba	text	No		*None*	

4. Tabel detail_pinjam_dokumen berisi data yang digunakan untuk menyimpan data peminjaman arsip.

Tabel 4.15 Tabel *Data Type Definition* dari tabel detail_pinjam_dokumen

Field	Type	Null	Key	Default	Extra
id_dp	Int(11)	No	PK	*None*	
no_pinjam	varchar(50)	No		*None*	
no_dok	varchar(100)	No		*None*	
Keperluan	text	No		*None*	
status_pinjam	Int(11)	No		*None*	
k_dp	text	No		*None*	

5. Tabel detail_tempat digunakan sebagai penyimpanan data detail tempat penyimpanan arsip.

Tabel 4.16 Tabel *Data Type Definition* dari tabel detail_tempat

Field	Type	Null	Key	Default	Extra
id_dt	int(11)	No	PK	*None*	
id_tp	int(11)	No		*None*	
nama_dt	varchar(50)	No		*None*	
k_dt	text	No		*None*	

6. Tabel disposisi digunakan sebagai tempat penyimpanan data disposisi surat masuk.

Tabel 4.17 Tabel *Data Type Definition* dari tabel disposisi

Field	Type	Null	Key	Default	Extra
id_dispo	int(11)	No		*None*	

no_dok	varchar(100)	No		*None*	
id_bagian	int(11)	No		*None*	
tgl_dispo	date	No		*None*	
ket_disposisi	Text	No		*None*	
status_dispo	int(1)	No		*None*	
k_dispo	Text	No		*None*	

7. Tabel dokumen digunakan sebagai tempat penyimpanan arsip beserta metadata nya.

Tabel 4.18 Tabel *Data Type Definition* dari tabel dokumen

Field	Type	Null	Key	Default	Extra
id_dok	int(11)	No	PK	*None*	AI
no_dok	varchar(100)	No	PK	*None*	
nip	varchar(50)	No		*None*	
id_klas	int(11)	No	FK	*None*	
id_bagian	int(11)	No	FK	*None*	
nama_dok	varchar(200)	No		*None*	
dari	text	No		*None*	
kepada	varchar(20)	No		*None*	
hal	text	No		*None*	
tgl_masuk_dok	date	No		*None*	
tgl_dok	date	No		*None*	
tgl_keluar_dok	date	No		*None*	
status_dok	varchar(20)	No		*None*	
re_aktif	date	No		*None*	
re_inaktif	date	No		*None*	

media	varchar(20)	No		*None*	
format_dok	text	No		*None*	
file_dok	text	No		*None*	
pinjam	int(1)	No		*None*	
id_lokasi	int(11)	No		*None*	
id_tp	int(11)	No	FK	*None*	
id_dt	int(11)	No	FK	*None*	
ket_dok	text	No		*None*	
reg_dok	int(11)	No		*None*	
k_dok	text	No		*None*	
sifat_surat	varchar(20)	No		*None*	
catatan	text	No		*None*	

8. Tabel jabatan digunakan sebagai tempat penyimpanan data jabatan

Tabel 4.19 Tabel *Data Type Definition* dari tabel jabatan

Field	Type	Null	Key	Default	Extra
id_jab	varchar(10)	No	PK	*None*	
nama_jab	varchar(50)	No		*None*	
k_jab	text	No		*None*	

9. Tabel jabatan_fungsional berisi data jabatan fungsional.

Tabel 4.20 Tabel *Data Type Definition* dari tabel jabatan_fungsional

Field	Type	Null	Key	Default	Extra
id_jf	int(11)	No	PK	*None*	
nama_jf	varchar(200)	No		*None*	
k_jf	text	No		*None*	

10. Tabel karyawan berisi data karyawan yang ada di UPT Laboratorium Kesehatan Hewan Malang.

Tabel 4.21 *Tabel Data Type Definition* dari tabel karyawan

Field	Type	Null	Key	Default	Extra
nip	varchar(50)	No	PK	*None*	
nama_kar	varchar(200)	No		*None*	
jk_kar	varchar(20)	No		*None*	
tmp_lhr	varchar(20)	No		*None*	
tgl_lhr	date	No		*None*	
telp	char(20)	No		*None*	
email	varchar(50)	No		*None*	
alamat_kar	text	No		*None*	
id_jab	varchar(10)	No		*None*	
id_jf	int(11)	No		*None*	
id_bagian	int(11)	No		*None*	
status	int(1)	No		*None*	
foto_kar	text	No		*None*	
k_kar	text	No		*None*	

11. Tabel kategori_dokumen digunakan sebagai tempat penyimpanan data kategori arsip.

Tabel 4.21 Tabel *Data Type Definition* dari tabel kategori_dokumen

Field	Type	Null	Key	Default	Extra
id_katdok	int(11)	No	PK	*None*	
nama_katdok	varchar(200)	No		*None*	
k_katdok	text	No		*None*	

12. Tabel klasifikasi digunakan sebagai tempat penyimpanan data klasifikasi arsip

Tabel 4.22 Tabel *Data Type Definition* dari tabel klasifikasi

Field	Type	Null	Key	Default	Extra

id_klas	int(11)	No	PK	*None*	AI
no_klas	text	No		*None*	
nama_klas	text	No		*None*	
id_katdok	int(11)	No		*None*	
id_bagian	int(11)	No		*None*	
r_aktif	int(11)	No		*None*	
r_inaktif	int(11)	No		*None*	
ket_jrd	varchar(50)	No		*None*	
format	text	No		*None*	
k_klas	text	No		*None*	

13. Tabel kode_surat digunakan sebagai tempat penyimpanan data kode persuratan.

Tabel 4.23 Tabel *Data Type Definition* dari tabel kode_surat

Field	Type	Null	Key	Default	Extra
id_ps	int(11)	No		*None*	AI
kode	varchar(50)	No		*None*	
nama	varchar(100)	No		*None*	
format	text	No		*None*	
k_ps	text	No		*None*	

14. Tabel konsep_surat digunakan sebagai tempat penyimpanan data konsep surat.

Tabel 4.24 Tabel *Data Type Definition* dari tabel konsep_surat

Field	Type	Null	Key	Default	Extra
id_konsep	int(11)	No	PK	*None*	AI
no_konsep	varchar(100)	No	PK	*None*	
nip	varchar(50)	No		*None*	
id_klas	int(11)	No	FK	*None*	
id_bagian	int(11)	No	FK	*None*	
nama_dok	varchar(200)	No		*None*	

hal	text	No		*None*	
kepada	text	No		*None*	
tgl_konsep	date	No		*None*	
status_konsep	varchar(20)	No		*None*	
format	text	No		*None*	
js_keluar	varchar(50)	No		*None*	
no_dok	varchar(100)	No		*None*	
lampiran	int(11)	No		*None*	
alamat	text	No		*None*	
ttd	varchar(20)	No		*None*	
sifat_surat	varchar(10)	No		*None*	
k_konsep	text	No		*None*	

15. Tabel level berisi data level user yang bisa mengakses sistem

Tabel 4.25 *Tabel Data Type Definition* dari tabel level

Field	Type	Null	Key	Default	Extra
id_level	int(11)	No	PK	*None*	AI
nama_level	varchar(50)	No		*None*	
k_level	text	No		*None*	

16. Tabel lokasi digunakan sebagai tempat penyimpanan data ruang penyimpanan arsip.

Tabel 4.26 Tabel *Data Type Definition* dari tabel level

Field	Type	Null	Key	Default	Extra
id_lokasi	int(11)	No	PK	*None*	AI
nama_lokasi	text	No		*None*	
k_lokasi	text	No		*None*	

17. Tabel perusahaan digunakan sebagi tempat penyimpanan data email perusahaan.

Tabel 4.27 Tabel *Data Type Definition* dari tabel perusahaan

Field	Type	Null	Key	Default	Extra
id_per	int(11)	No	PK	*None*	AI
email	Text	No		*None*	
password	char(32)	No		*None*	

18. Tabel pinjam_dokumen berisi data peminjaman arsip.

Tabel 4.28 Tabel *Data Type Definition* dari tabel pinjam_dokumen

Field	Type	Null	Key	Default	Extra
id_pinjam	int(11)	No	PK	*None*	AI
id_bagian	int(11)	No	FK	*None*	
no_pinjam	varchar(50)	No		*None*	
nip	varchar(50)	No		*None*	
peminjam	text	No		*None*	
tgl_pinjam	date	No		*None*	
tgl_kembali	date	No		*None*	
status_p	int(1)	No		*None*	
k_pinjam	text	No		*None*	

19. Tabel revisi digunakan sebagai tempat penyimpanan data riwayat revisi arsip.

Tabel 4.29 T*abel Data Type Definition* dari tabel revisi

Field	Type	Null	Key	Default	Extra
id_versi	int(11)	No	PK	*None*	AI
no_dok	varchar(100)	No	FK	*None*	
tgl_versi	date	No		*None*	
file_versi	Text	No		*None*	
versi	int(11)	No		*None*	
status	int(11)	No		*None*	
ket_versi	Text	No		*None*	
k_versi	Text	No		*None*	

20. Tabel tempat_berkas digunakan sebagai tempat penyimpanan data tempat berkas untuk menyimpan arsip fisik.

Tabel 4.30 Tabel *Data Type Definition* dari tabel revisi

Field	Type	Null	Key	Default	Extra
id_tp	int(11)	No	PK	*None*	AI
nama_tp	Text	No		*None*	
k_tp	Text	No		*None*	

21. Tabel user digunakan untuk menyimpan data user yang dapat mengakses sistem.

Tabel 4.31 Tabel *Data Type Definition* dari tabel revisi

Field	Type	Null	Key	Default	Extra
id_user	int(11)	No	PK	*None*	AI
nip	varchar(50)	No		*None*	
id_level	int(11)	No	FK	*None*	
username	varchar(50)	No		*None*	
password	char(32)	No		*None*	
pass	Text	No		*None*	
status_u	int(1)	No		*None*	
ket	Text	No		*None*	
k_user	Text	No		*None*	

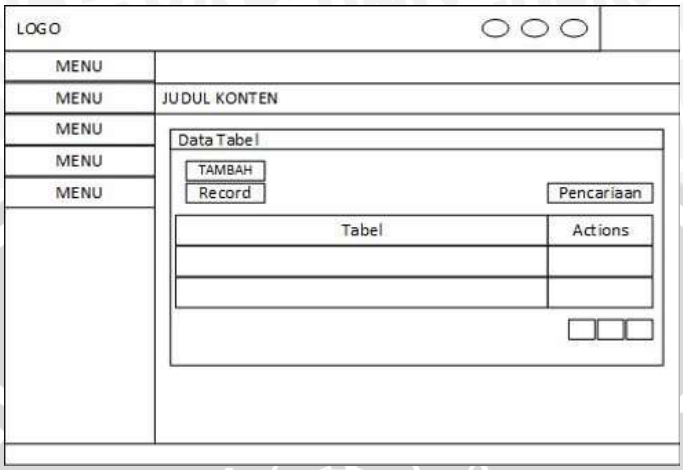
4.6.8. Perancangan Antar Muka

1. Desain Tampilan halaman Login

Gambar 4.23 Desain Tampilan halaman *Login*

2. Desain Tampilan Halaman Data Tabel

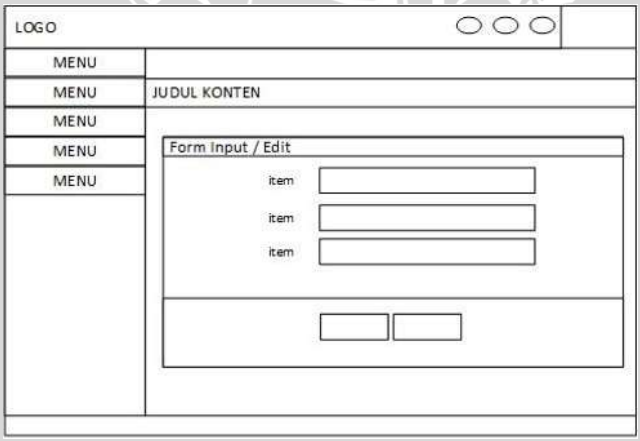
Halaman data tabel merupakan halaman yang berisi tentang data konten yang telah di inputkan ke dalam sistem.



Gambar 4.24 Desain Tampilan Halaman Data Tabel

3. Desain Tampilan Halaman Tambah/ Ubah Data

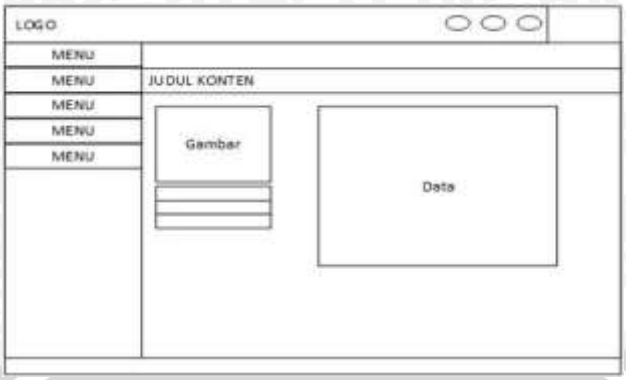
Halaman berikut merupakan tampilan halaman untuk menambah atau mengubah data ke dalam sistem.



Gambar 4.25 Desain Tampilan Halaman Data Tabel

4. Desain Tampilan Halaman Profil

Halaman profile merupakan tampilan halaman yang berisi data detail data user.



Gambar 4.26 Desain Tampilan Halaman Profil

5. Desain Halaman Laporan

Halaman laporan merupakan halaman yang berisikan data yang akan di cetak dengan parameter atau kategori tertentu sebelum data ditampilkan dan di cetak berupa pdf.

Gambar 4.27 Desain Halaman Laporan

.

BAB V
IMPLEMENTASI DAN PENGUJIAN

Tahap implementasi merupakan tahap penerapan atau pembuatan aplikasi dari perancangan sistem yang telah dilakukan di Bab IV. Berikut rancang bangun aplikasi dari tiap hak akses *user*, yaitu super admin, administratif, admin persuratan, admin kepegawaian, dan admin pelayanan. Pembahasan implementasi dalam bab ini meliputi beberapa tahap, yaitu spesifikasi perangkat lunak, batasan-batasan implementasi, dan implementasi pada antarmuka (*interface*) aplikasi.

5.1 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak yang dipakai dalam proses pengembangan Sistem Manajemen Arsip UPT Laboratorium Kesehatan Hewan Malang dijelaskan pada tabel 5.1.

Tabel 5.1. Spesifikasi Perangkat Lunak

Bahasa Pemrograman	PHP 5.2.9, HTML
Framework	Codeigniter 2.1.4, Bootstrap 2.30
RDBMS	MySQL 5.1.41, phpMyAdmin 3.2.4
Web Server	Apache 2.2.14

5.2 Batasan – batasan Implementasi

Beberapa batasan dalam mengimplementasikan Sistem Informasi Manajemen Arsip adalah sebagai berikut:

1. Sistem Informasi Manajemen UPT Laboratorium Kesehatan Hewan Malang ini dirancang menggunakan *framework* CodeIgniter dan Bootstrap.
2. Sistem menggunakan *web server* Apache.
3. Sistem menggunakan *Relational Data Base Management System* (RDBMS) MySQL

5.3 Implementasi Basis Data

Implentasi basis data dilakukan menggunakan *Relational Data Base Management System* (RDBMS) MySQL. Implementasi basis data direalisasikan dalam bentuk tabel, seperti pada Gambar 5.1 berikut ini, sesuai dengan perancangan basis data dalam bentuk rancangan *database* yang ditunjukkan pada bab perancangan.

Gambar 5.1. Implementasi Basis Data

5.4 Implementasi Antarmuka (*User Interface*)

5.4.1. Implementasi HTML dan CSS pada Antarmuka

HTML digunakan untuk merancang semua komponen halaman web, serta menampilkan informasi yang dibutuhkan dalam bentuk halaman web yang dibuka di *browser*. CSS digunakan untuk mengatur komponen halaman web dalam HTML dengan format tertentu sesuai dengan desain halaman yang diinginkan. Berikut implementasi HTML pada *view* dokumen.php.

Tabel 5.2. Implementasi HTML pada view dokumen.php.

No.	Kode Program
1	`<div class="panel-body">`
2	` `
3	`<div align='right'>`
4	

```
5   <a href="<?php echo base_url(); ?>sistem_c/menu/tambah_dokumen"
6   class="btn btn-primary">
7         <i class="entypo-plus"></i>
8         Tambah Data
9   </a>
10  <a href="<?php echo base_url(); ?>sistem_c/menu/laporan_dok"
11  target="_blank" class="btn btn-primary">
12        <i class="entypo-print"></i>
13        Print </a>
14  <table class="table table-bordered datatable" id="table-1">
15      <thead>
16          <tr>
17              <th width="10">No</th>
18              <th align="center">No Arsip</th>
19              <th align="center">Nama Arsip</th>
20              <th align="center">Kategori Arsip</th>
21              <th align="center">Batas Waktu</th>
22              <th align="center">Masa Arsip</th>
23              <th align="center">Tgl Arsip</th>
24              <th align="center">Tgl Record</th>
25              <th align="center">Status Arsip</th>
26              <th align="center">Peminjaman Arsip</th>
27              <th align="center" width="200">Actions</th>
28          </tr>
29      </thead>
30      <tbody>
31  <?php
32  $x=1;
33  foreach ($ambil->result_array() as $row) {
34  ?>
35        <tr class="odd gradeX">
36              <td width="10"><?php echo $x++; ?></td>
37              <td><?php echo $row['no_dok']; ?></td>
                <td><?php echo $row['nama_dok']; ?></td>
                <td><?php echo $row['nama_katdok']; ?></td>
```

Tabel diatas merupakan potongan kode HTML di *view* dokumen yang menampilkan potongan kode HTML dengan potongan kode PHP didalamnya. Baris ke-4 sampai baris ke 8 merupakan *script* untuk memanggil form tambah data arsip menggunakan HTML. Baris ke-9 sampai baris ke-12 merupakan *script* untuk

menampilkan laporan data arsip. Pada baris ke-13 merupakan potongan kode HTML untuk membuat tabel data dokumen yang akan ditampilkan.

Sedangkan untuk membuat jenis atau desain *form search* dokumen maupun desain tabel data dokumen, dan desain komponen/pembatas halaman dengan menggunakan CSS. Gambar 5.3 merupakan implementasi CSS pada *view*.

Tabel 5.3. Potongan Kode CSS neon.css pada *Class* dokumen.

No.	Kode program
1	`.label {`
2	`  border: 1px solid #000;`
3	`  }`
4	`  .table {`
5	`  border-collapse: collapse !important;`
6	`  }`
7	`  .table-bordered th,`
8	`  .table-bordered td {`
9	`  border: 1px solid #ddd !important;`
10	`  }`
11	`}`
12	`*,`
13	`*:before,`
14	`*:after {`
15	`  -webkit-box-sizing: border-box;`
16	`  -moz-box-sizing: border-box;`
17	`  box-sizing: border-box;`
18	`}`
19	`html {`
20	`  font-size: 62.5%;`
21	`  -webkit-tap-highlight-color: rgba(0, 0, 0, 0);`
22	`}`
23	`body {`
24	`  font-family: "Noto Sans", sans-serif, "Helvetica Neue",`
25	`Helvetica, Arial, sans-serif;`
26	`  font-size: 12px;`
27	`  line-height: 1.428571429;`
28	`  color: #060606;`
29	`  background-color: #ffffff;`
30	`}`
31	`input,`
	`button,`

```
select,
textarea {
  font-family: inherit;
  font-size: inherit;

```

Beberapa potongan kode CSS dari neon.css diatas digunakan untuk mengatur komponen halaman web dalam tag <div *class*="header">. <div *class*="body">., dan <div *class*="footer">. Sedangkan untuk mengatur komponen lain seperti form, digunakan CSS dari Bootstrap. Gambar 5.4 merupakan potongan kode CSS dari Bootstrap yang digunakan dalam implementasi antarmuka.

Tabel 5.4. Potongan Kode CSS pada bootstrap-modal.css.

No.	Kode program
1	`.modal {`
2	`outline: none;`
3	`position: absolute;`
4	`margin-top: 0;`
5	`top: 50%;`
6	`overflow: visible; /* allow content to popup out (i.e`
7	`tooltips) */`
8	`}`
9	
10	`.modal.fade {`
11	`top: -100%;`
12	`-webkit-transition: opacity 0.3s linear, top 0.3s ease-`
13	`out, bottom 0.3s ease-out, margin-top 0.3s ease-out;`
14	`-moz-transition: opacity 0.3s linear, top 0.3s ease-`
15	`out, bottom 0.3s ease-out, margin-top 0.3s ease-out;`
16	`-o-transition: opacity 0.3s linear, top 0.3s ease-`
17	`out, bottom 0.3s ease-out, margin-top 0.3s ease-out;`
18	`transition: opacity 0.3s linear, top 0.3s ease-`
19	`out, bottom 0.3s ease-out, margin-top 0.3s ease-out;`
20	`}`
21	`.modal.fade.in {`
22	`top: 50%;`
23	`}`
24	
25	`.modal-body {`

```
26          max-height: none;
27          overflow: visible;
28  }
29
30  .modal.modal-absolute {
31          position: absolute;
31          z-index: 950;
    }
```

5.4.2. Antarmuka (*User Interface*)

1. Antarmuka Halaman Awal (*Login*)

Berikut tampilan halaman awal ketika aplikasi diakses.

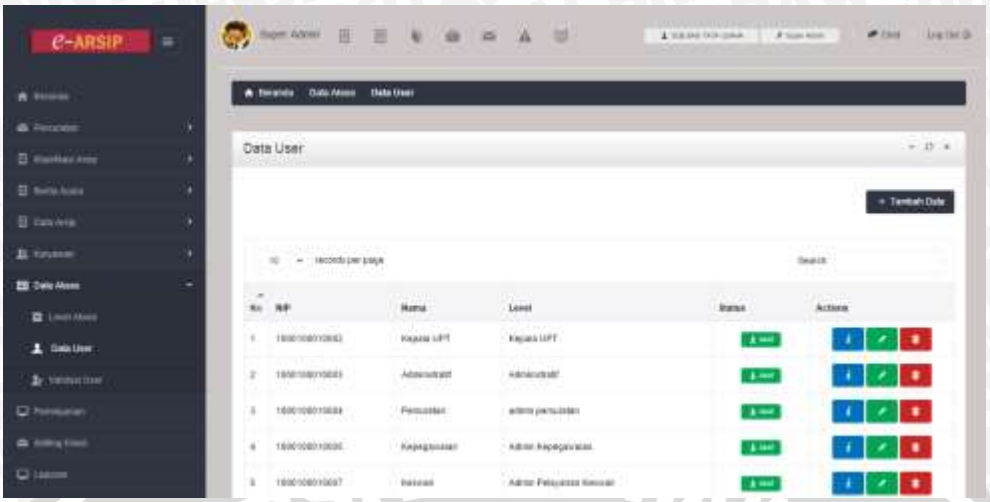
Gambar 5.2 Halaman Login

Sumber: Implementasi

Tiap user melakukan login terlebih dahulu untuk dapat mengakses sistem dengan mengisi *username* dan *password*.

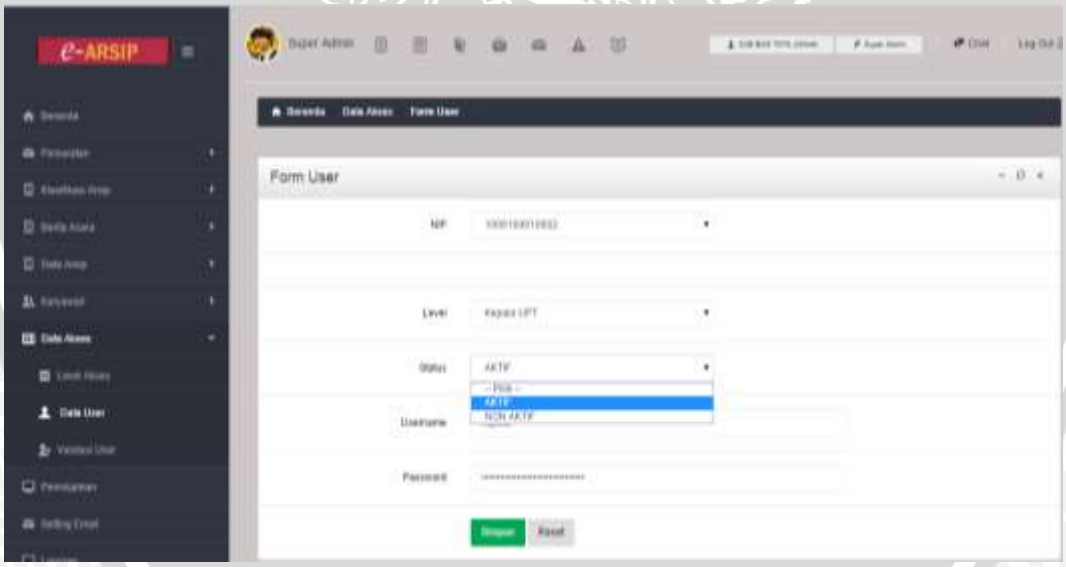
2. Halaman Super Admin

Gambar berikut merupakan gambar antarmuka (*interface*) halaman super admin Sistem Manajemen Arsip UPT Laboratorium Kesehatan Hewan Malang. Pada halaman *login* super admin terdapat menu pengelolaan yang lengkap mulai dari kelola user sampai cetak laporan.



Gambar 5.3. Halaman Menu Data *User*

Gambar 5.4 merupakan gambar halaman form validasi data user. Validasi disini merupakan penambahan status aktif kepada user agar bisa masuk ke dalam sistem.



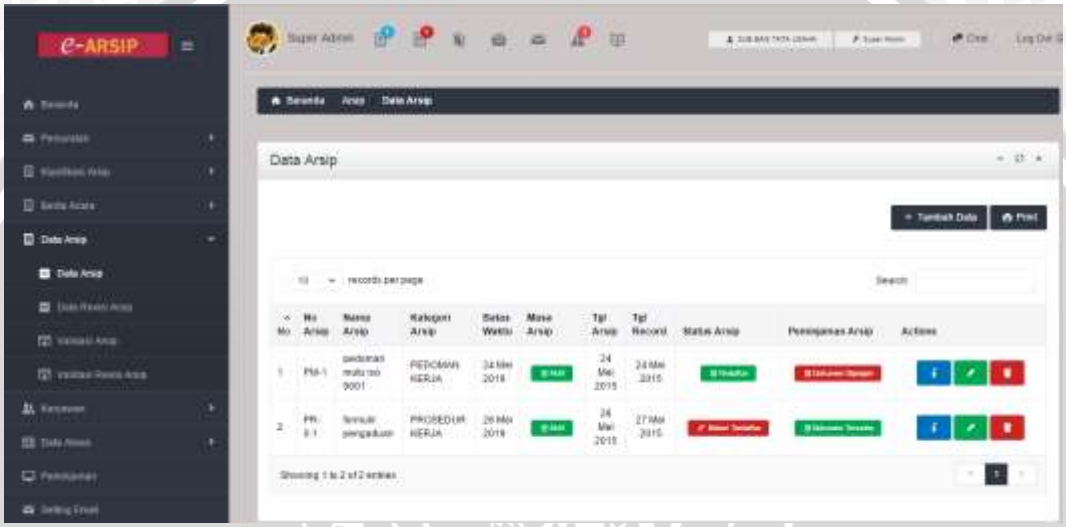
Gambar 5.4. Halaman Form Validasi Data User

Super admin dapat mengelola modul *user*. Pada tabel *user* ditampilkan data *user* berupa *username*, *level*, nip, dan nama karyawan. Admin dapat menambah, mengubah, dan menghapus data *user*. User disini merupakan karyawan yang telah di daftarkan oleh admin kepegawaian.

3. Halaman Admin

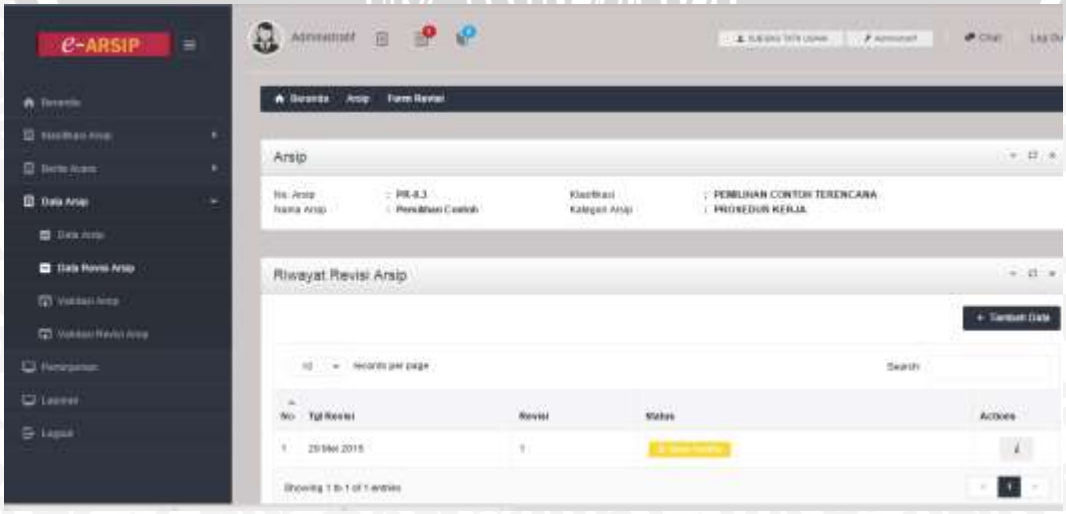
Halaman admin yang dimaksud adalah halaman admin secara umum yang ditampilkan pada admin kepegawaian, admin pelayanan, administratif, dan kepala UPT untuk mengelola arsip pada bagian masing-masing.

Gambar 5.5 merupakan gambar halaman menu data arsip yang dapat diakses oleh admin umum selain admin persuratan. Halaman ini digunakan untuk mellihat data arsip yang telah di inputkan oleh bagian masing-masing.



Gambar 5.5. Halaman Admin Menu Data Arsip

Halaman admin secara umum terdapat menu data arsip, revisi arsip, daftar arsip usul musnah, dan peminjaman arsip. Halaman 5.6 merupakan halaman data revisi arsip.



Gambar 5.6. Halaman Admin Data Revisi Arsip

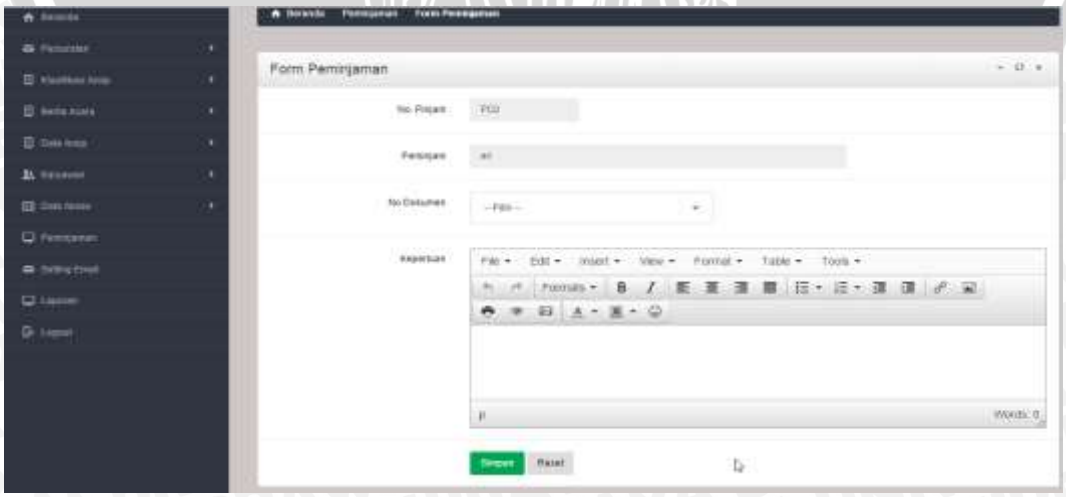
Untuk memberi peringatan kepada admin bahwa masa simpan(retensi) arsip telah habis, maka sistem dapat memberikan notifikasi bahwa masa simpan arsip telah habis. Gambar notifikasi retensi arsip ditunjukkan pada gambar 5.7

Gambar 5.7 Halaman Admin Masa Simpan Arsip Telah Habis

Arsip yang telah habis masa simpannya dapat diperpanjang dengan menu nilai kembali arsip seperti yang ditunjukkan pada gambar 5.8

Gambar 5.8 Halaman Admin Perpanjang Masa Simpan Arsip

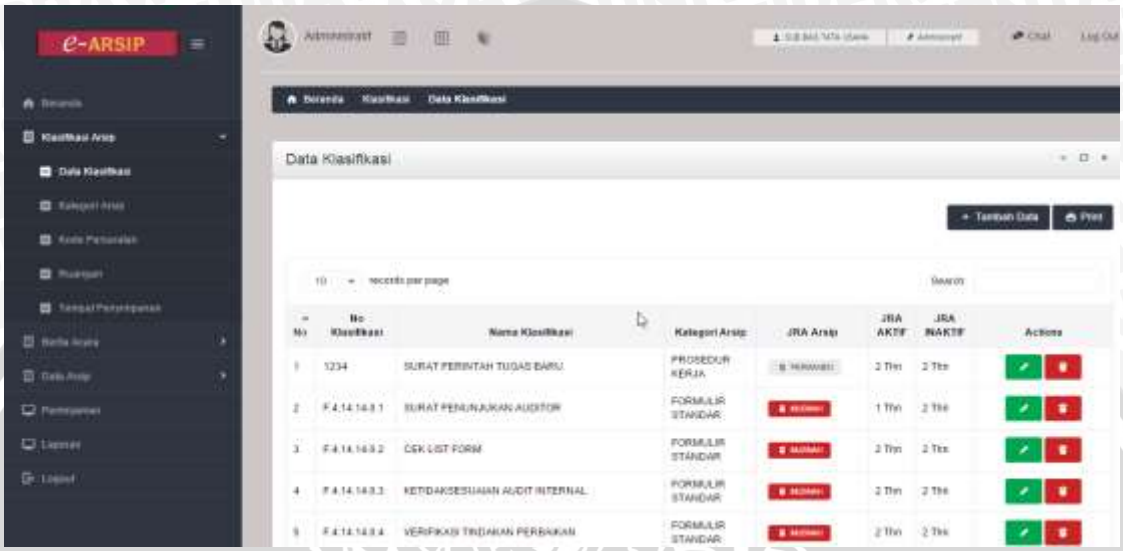
Gambar 5.9 merupakan gambar halaman admin form peminjaman arsip



Gambar 5.9 Halaman Admin Form Peminjaman Arsip

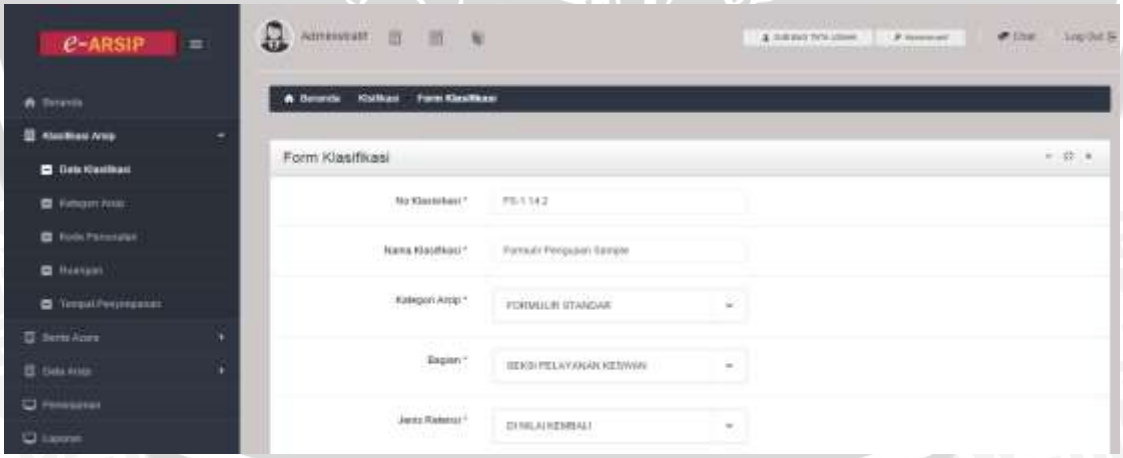
4. Halaman Administratif

Pada halaman administratif ditampilkan modul pengelolaan arsip seperti klasifikasi arsip, validasi registrasi arsip, validasi revisi arsip, dan mencetak berita acara pemusnahan arsip. Halaman 5.10 merupakan halaman data klasifikasi arsip.



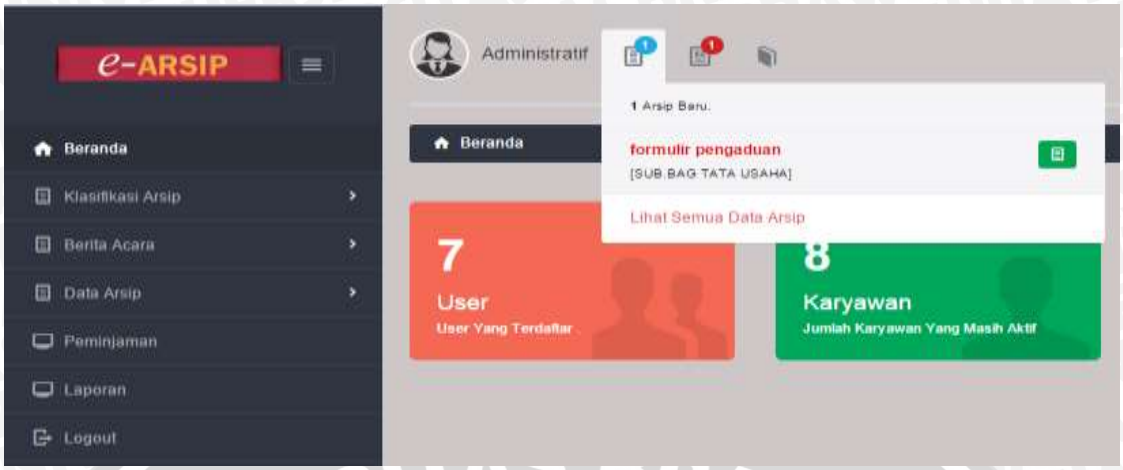
Gambar 5.10 Halaman Data Klasifikasi Arsip

Gambar 5.11 merupakan gambar halaman form tambah data klasifikasi arsip



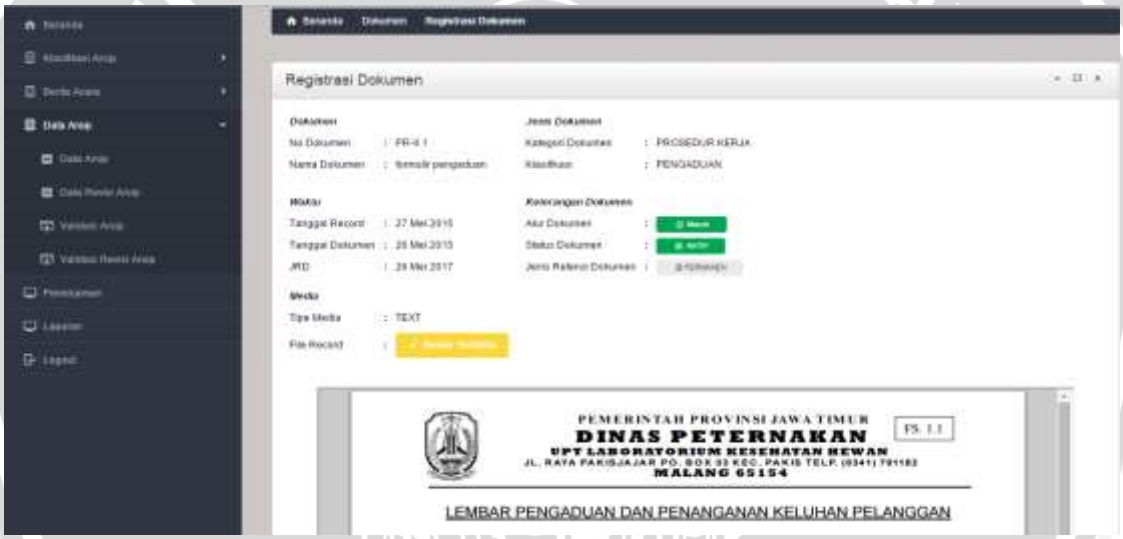
Gambar 5.11 Halaman *Form* Tambah Data Klasifikasi Arsip

Arsip yang telah di inputkan oleh bagian masing-masing harus di validasi terlebih dahulu oleh administratif. Gambar 5.12 merupakan gambar halaman notifikasi validasi arsip baru



Gambar 5.12 Halaman Admin Notifikasi Validasi Arsip Baru

Gambar 5.13 merupakan gambar halaman preview validasi arsip baru



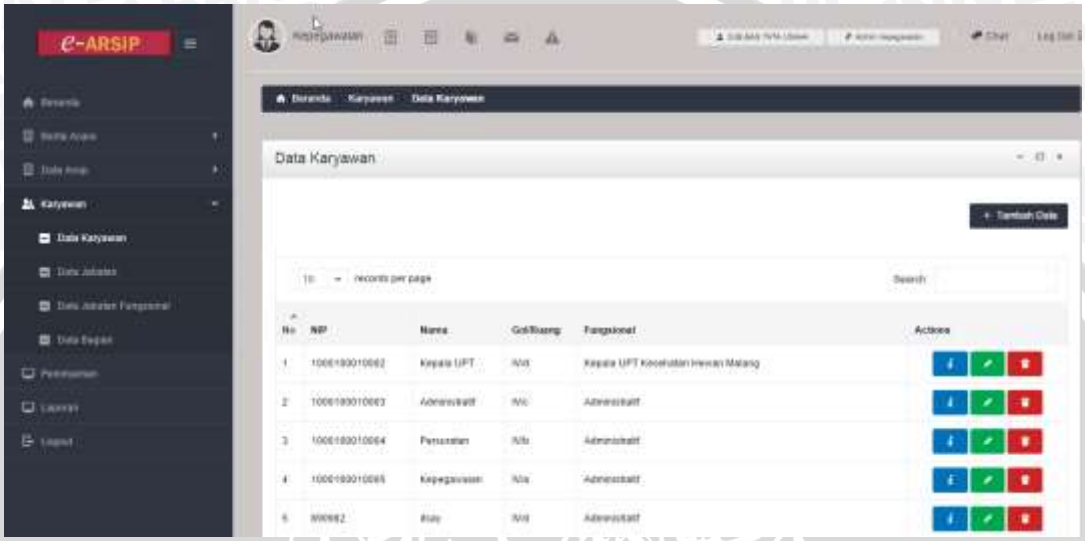
Gambar 5.13 Halaman Preview Arsip

Gambar 5.14 merupakan gambar halaman notifikasi revisi arsip baru

Gambar 5.14 Halaman Notifikasi Revisi Arsip

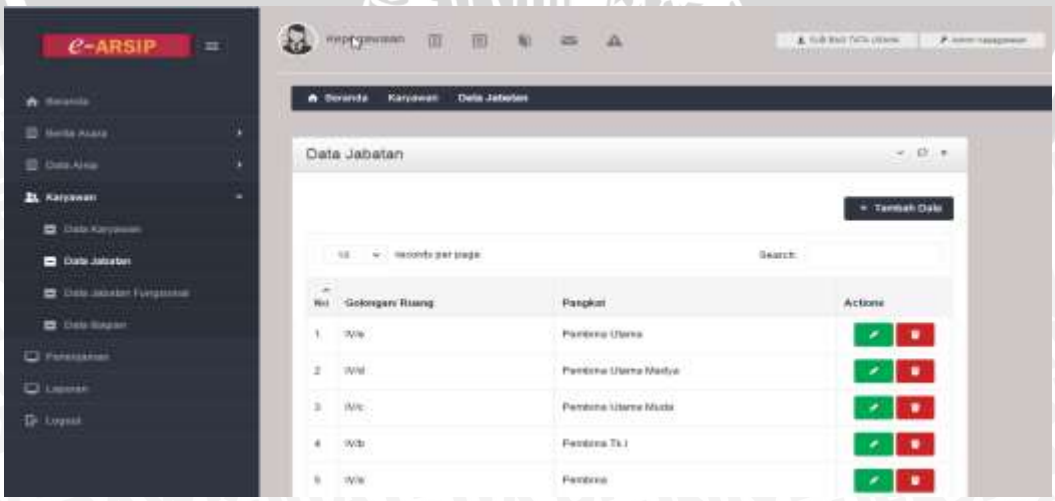
5. Halaman Admin Kepegawaian

Dalam halaman admin kepegawaian, terdapat menu pengelolaan data yang berhubungan dengan kepegawaian seperti data karyawan, jabatan, jabatan fungsional, dan data bagian. Halaman admin kepegawaian data karyawan ditunjukkan pada gambar 5.15



Gambar 5.15 Halaman Admin Kepegawaian Data Karyawan

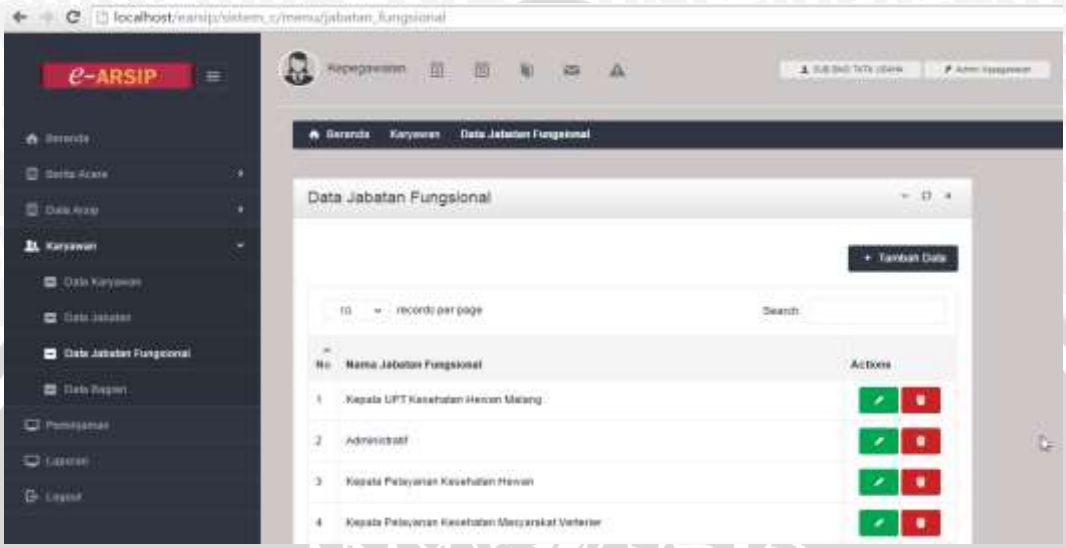
Gambar 5.16 merupakan gambar halaman data jabatan yang ada di UPT Laboratorium Kesehatan Hewan Malang.



Gambar 5.16 Halaman Data Jabatan

Pada halaman menu jabatan, admin bisa menambahkan, mengedit atau jabatan yang ada di UPT Laboratorium Kesehatan Hewan Malang.

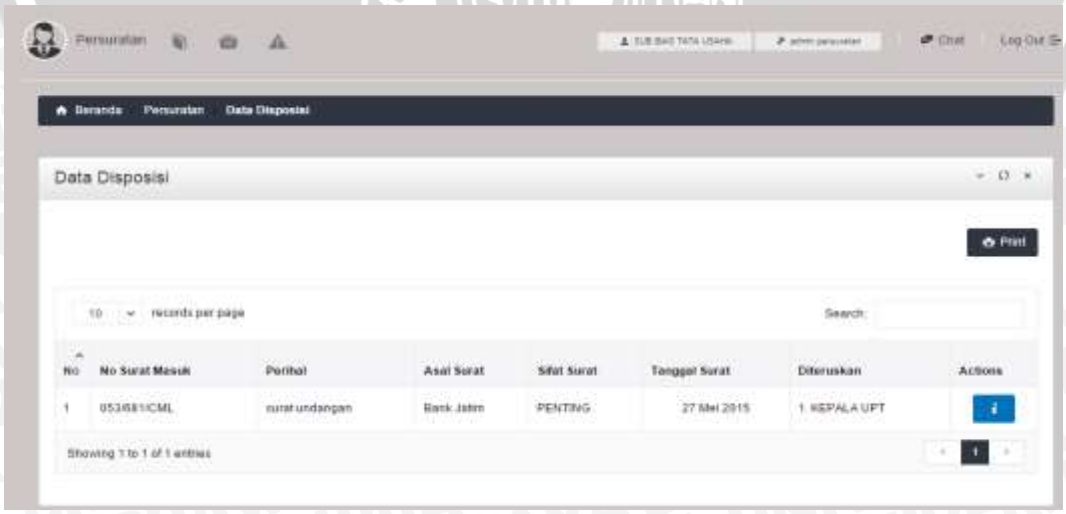
Gambar 5.17 merupakan gambar halaman admin kepegawaian data jabtan fungsional.



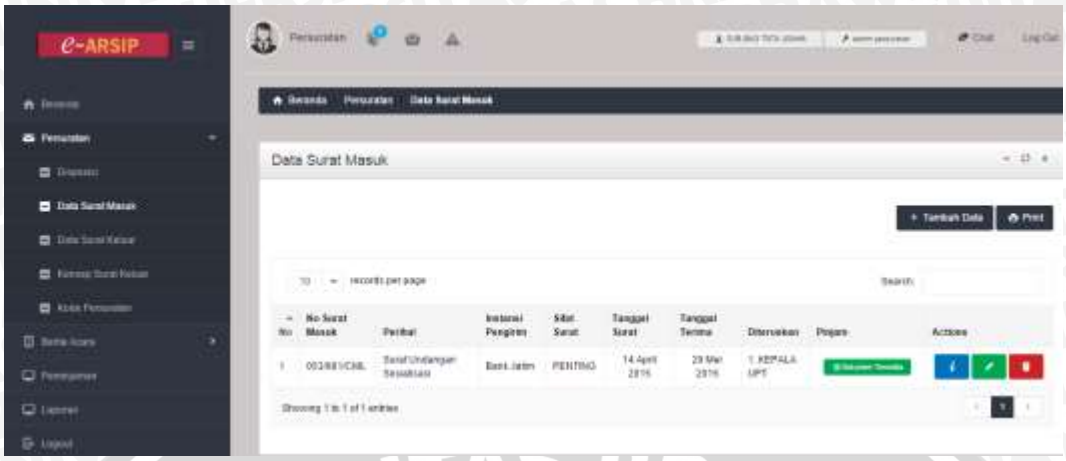
Gambar 5.17 Halaman Admin Kepegawaian Data Jabatan Fungsional

6. Halaman Admin Persuratan

Admin Persuratan dapat mengelola menu-menu yang berhubungan dengan pengelolaan surat, yaitu disposisi, data surat masuk, sata surat keluar, dan data konsep surat keluar. Halaman data disposisi ditunjukkan pada gambar 5.18

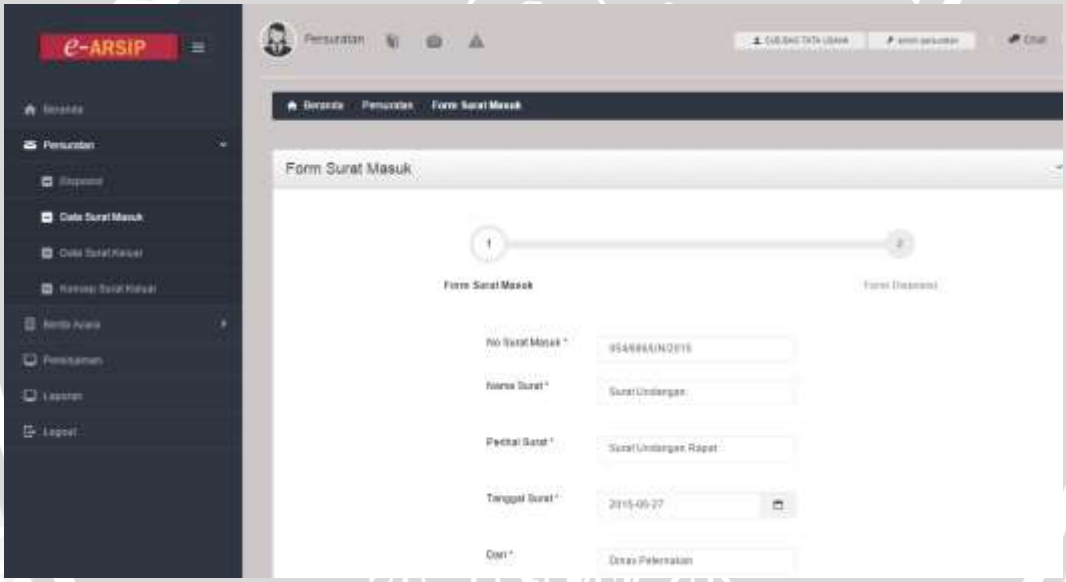


Gambar 5.18 Halaman Persuratan Data Disposisi.



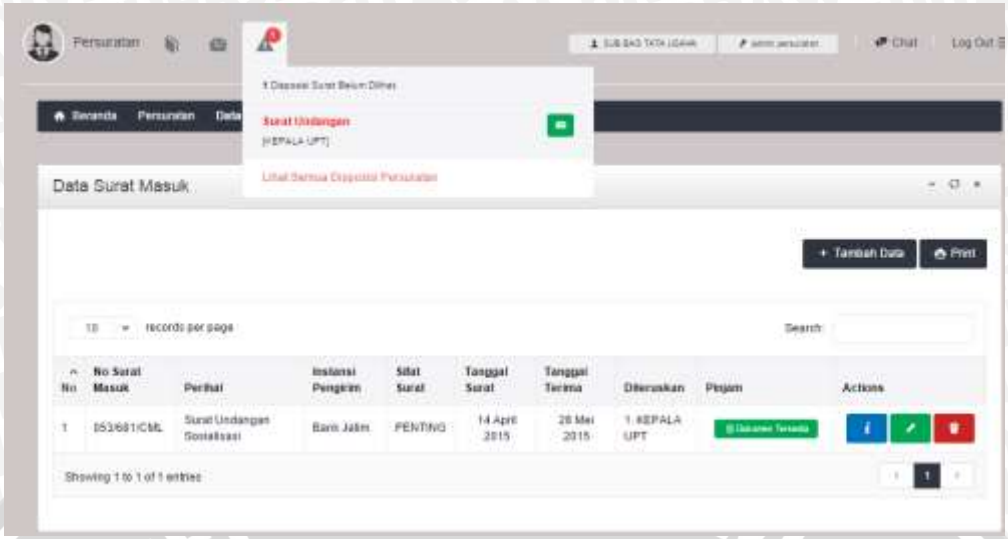
Gambar 5.19 Halaman Data Surat Masuk.

Halaman form tambah data sura masuk ditunjukkan pada gambar 5.20



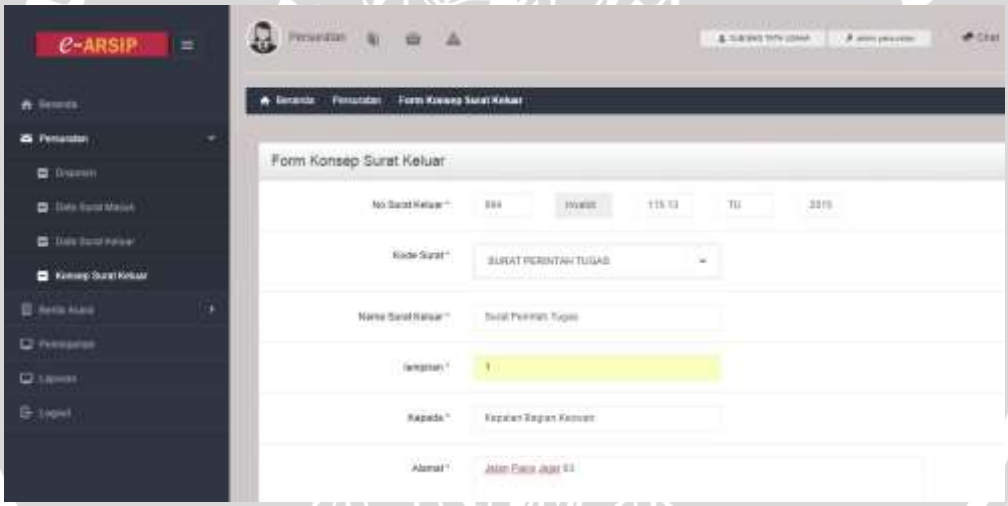
Gambar 5.20 Halaman Form Tambah Data Surat Masuk.

Gambar 5.21 merupakan halaman notifikasi yang memberitahukan bahwa disposisi surat belum dibaca yang hanya dapat diakses oleh admin persuratan dan super admin



Gambar 5.21 Halaman Notifikasi Disposisi Surat Belum Dibaca

Gambar 5.22 merupakan gambar halaman form tambah konsep surat keluar



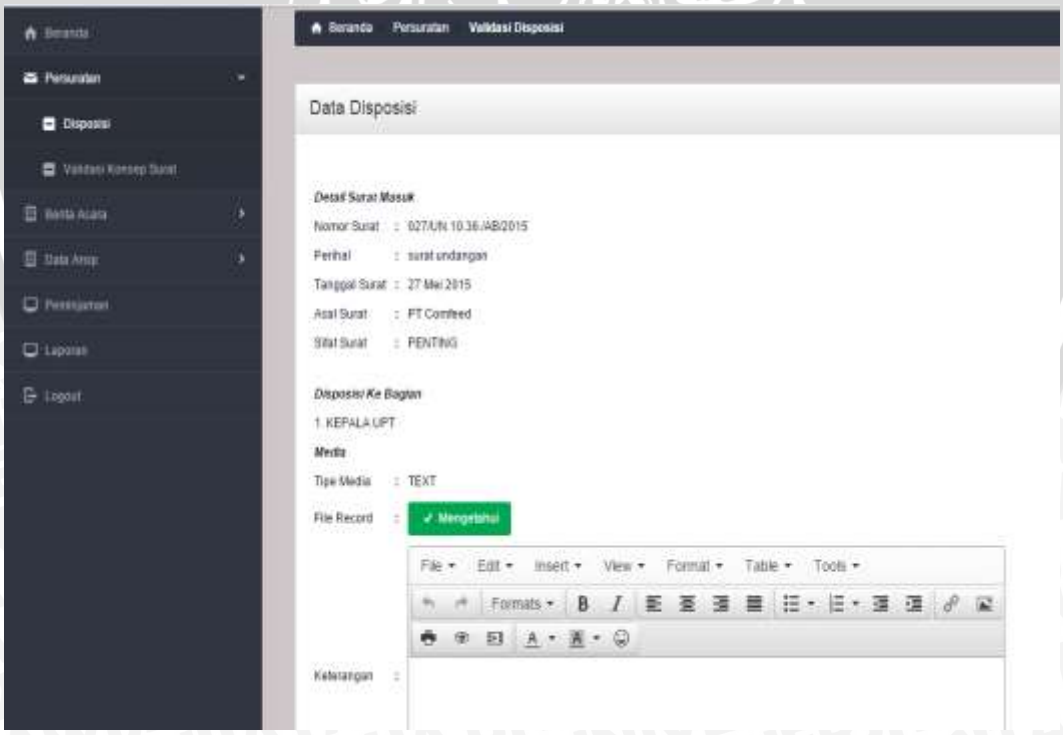
Gambar 5.22 Halaman Form Tambah Konsep Surat Keluar

7. Halaman Kepala UPT

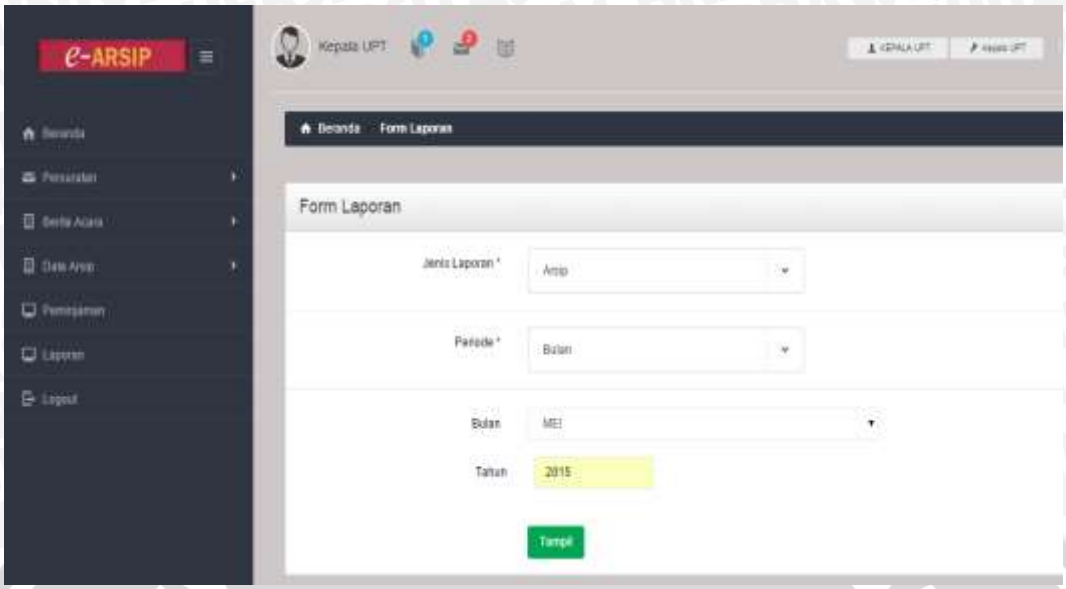
Kepala UPT memiliki hak akses secara spesifik untuk mengetahui disposisi surat masuk, validasi konsep surat keluar, dan lihat laporan. Halaman notifikasi disposisi surat masuk baru ditnjukkan pada gambar 5.23

Gambar 5.23 Halaman Notifikasi Disposisi Surat Masuk Baru.

Setelah menerima notifikasi, kepala UPT dapat melihat data disposisi surat masuk dan mengetahui surat masuk. Form data disposisi ditunjukkan pada gambar 5.24. Form disposisi digunakan untuk melihat informasi surat masuk yang telah di inputkan oleh bagian persuratan. Dalam form disposisi terdapat button "Mengetahui" yang digunakan sebagai validasi bahwa surat masuk telah dibaca.

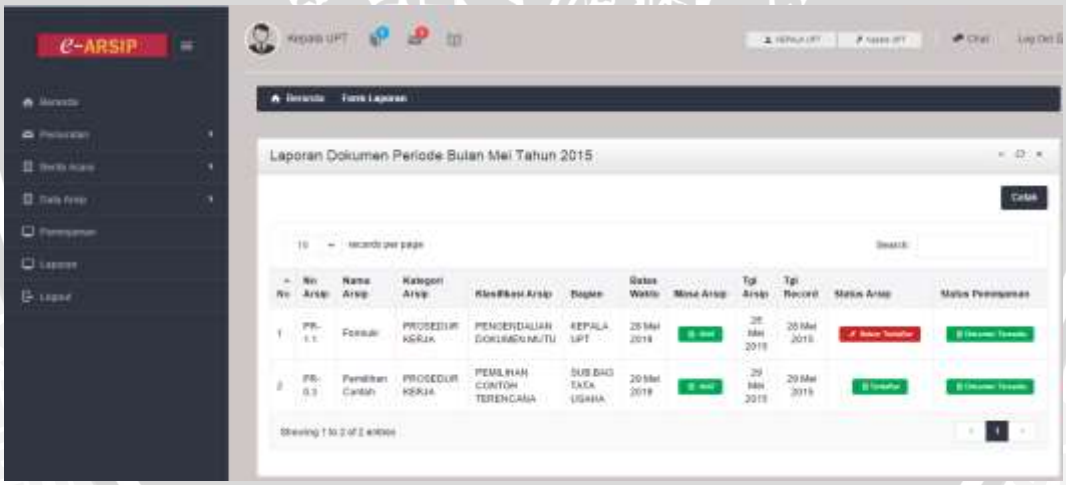


Gambar 5.24 Halaman Form Disposisi Surat Masuk.



Gambar 5.25 Halaman Pilih Laporan

Gambar 5.26 merupakan gambar halaman lihat laporan. Pada halaman ini ditampilkan data laporan yang telah dipilih berdasarkan periode waktu.



Gambar 5.26 Halaman Lihat Laporan

Data laporan yang telah dipilih dapat dicetak dalam format PDF. Gambar 5.27 merupakan halaman cetak laporan dalam format PDF.

localhost/earsip/sistem_c/laporan/laporan_cetak

PEMERINTAH PROVINSI JAWA TIMUR
DINAS PETERNAKAN
UPT LABORATORIUM KESEHATAN HEWAN
JL. RAYA PAKISJAJAR PO. BOX 03 KEC. PAKIS TELP. (0341) 791182
MALANG 65154

LAPORAN ARSIP

Periode : Tahun 2015 Tanggal Cetak : 2015-05-29

No	No Arsip	Nama Arsip	Klasifikasi	Kategori	Bagian	Tgl Arsip
1	PR-1.1	Formulir	PENGENDALIAN DOKUMEN MUTU	PROSEDUR KERJA	KEPALA UPT	28 Mei 2015
2	PR-03	Pemilihan Contoh	PEMILIHAN CONTOH TERENCANA	PROSEDUR KERJA	SUB.BAG TATA USAHA	29 Mei 2015

Gambar 5.27 Halaman Cetak Laporan

5.5 Pengujian

Proses pengujian dilakukan melalui tiga tahapan (strategi) yaitu pengujian unit, pengujian validasi dan pengujian kesesuaian standar ISO 15489 .

5.5.1. Pengujian Unit

Pengujian unit fokus pada usaha verifikasi pada unit yang terkecil pada desain perangkat lunak (komponen atau modul perangkat lunak). Setiap unit perangkat lunak diuji agar dapat diperiksa apakah aliran masukan (*input*) dan keluaran (*output*) dari unit sudah sesuai dengan yang diinginkan [ROS-13]. Pada pengujian unit digunakan metode pengujian *basis path testing*, yaitu metode pengujian dengan langkah-langkah pengujian memodelkan alur logika program pada suatu *flow graph*, menentukan *cyclomatic complexity*, menentukan basis set dari jalur-jalur independen, dan membuat kasus uji (*test case*) pada setiap basis set yang telah ditentukan.

1. *Function* input_dokumen() dari *Class Controller* Dokumen

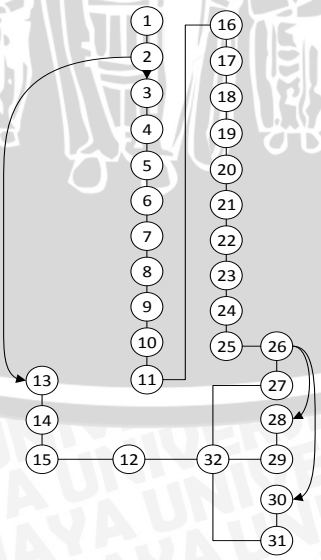
Function input_dokumen() merupakan *method* dari *class controller* dokumen yang berfungsi untuk menambah / input data dokumen melalui halaman dokumen. Pada *class function* input_dokumen() seleksi kondisi *if* pertama untuk menambah data dokumen jika data berhasil di post dan seleksi kondisi *if* kedua

seleksi data media arsip. Tabel 5.5 merupakan kode program function input_dokumen dan gambar 5.28 merupakan pemodelan *flowgraph* pada fungsi input_dokumen().

Tabel 5.5 Kode Program function_input_dokumen()

Kode Program	Node
function input_dokumen()	(1)
{	
if($this->input->post()){	(2)
$bagian = $this->input->post('bagian');	(3)
$cek = $this->db->query("select nama_bagian from bagian where	
id_bagian ='$bagian' ");	(4)
$row = $cek->row_array();	(5)
$config['*upload*_path'] ='data/dokumen/'.$row['nama_bagian'].'/';	
$config['allowed_types']= 'mp4|pdf|docx|doc|xlx|avi|flv|mkv|mp3';	
$config['max_size'] = '9000000000000000';	(6)
$config['overwrite'] = FALSE;	(7)
$config['remove_spaces'] = TRUE;	(8)
$this->load->library('*upload*', $config);	(9)
$this->*upload*->do_*upload*('file');	(10)
$getcon = $this->*upload*->data();	(11)
$namafile = $getcon['file_name'];	(12)
$this->load->model('dokmodel');	(13)
$this->dokmodel->input_data_dokumen($namafile);	(14)
$this->session->set_flashdata('input', 'input');	
redirect('sistem_c/menu/dokumen');	
} else {	(15)
$data['div'] = $this->session->userdata('bagian');	
$data['msg'] ='';	
$data['mod'] ='dok';	
$data['page'] ='fdok';	
$this->load->view('sistem_v/index', $data);	(16)
}	
}	(17)
function input_data_dokumen($namafile)	(18)
{	
$no = $this->input->post('no');	(19)
$nama = $this->input->post('nama');	
$sifat = $this->input->post('sifat');	
$klas = $this->input->post('klas');	
$tmp = $this->input->post('tmp');	
$cek1 = $this->db->query("select id_klas from klasifikasi where no_klas='$klas' ");	(20)
$row1 = $cek1->row_array();	(21)
$klasi = $row1['id_klas'];	
$bagian = $this->input->post('bagian');	
$hal = $this->input->post('hal');	
$tgl_dok = $this->input->post('tgl_masuk');	
$nip = $this->input->post('nip');	
$media = $this->input->post('media');	
$tgl = date("Y-m-d");	
$cek = $this->db->query("select r_aktif,r_inaktif from klasifikasi where no_klas='$klas' ");	(22)
$row = $cek->row_array();	

```
$guna          = $this->input->post('guna');                          (23)
$detail        = $this->input->post('detail');
$lokasi        = $this->input->post('lokasi');
$tp            = $this->input->post('tp');
$catatan       = $this->input->post('catatan');
$ra            = $row['r_aktif'];
$re            = $row['r_inaktif'];
$tg            = date_create($tgl_dok);
date_add($tg,date_interval_create_from_date_string(""".$ra."           (24)
years"));
$raktif        = date_format($tg,"Y-m-d");
$tgl           = date_create($raktif);
date_add($tgl1,date_interval_create_from_date_string(""".$re."         (25)
years"));
$rinaktif      = date_format($tgl1,"Y-m-d");
$format        = $this->input->post('format');
$fi            = $namafile;                                           (26)
if($this->input->post('media')=='TEXT')
{                                                                     (27)
  $ket = $hal.'/'.$sifat;
}                                                                     (28)
elseif($this->input->post('media')=='AUDIO')
{                                                                     (29)
  $ket = $hal.'/'.$sifat;
}                                                                     (30)
elseif($this->input->post('media')=='VIDEO')
{                                                                     (31)
  $ket = $hal.'/'.$sifat;
}
$k_dok = md5(($this->input->post('nama')).($this->input-
>post('tgl_masuk')).($this->input->post('no')));                      (32)
$this->db->query("insert into dokumen
values('','$no','$nip','$klasi','$bagian','$nama','','','','$tg
l_dok','$tgl','','Aktif','$raktif','$rinaktif','$media','','$fi
','1','$lokasi','$tmp','$detail','$ket','1','$k_dok','','$catat
an')");
}
```



Gambar 5.28 *Flow Graph Function* input_data_dokumen dari *Class Controller* dokumen

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

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

$V(G) = 34 - 32 + 2$

$V(G) = 4$

Jadi nilai *cyclomatic complexity* untuk *flowgraph class function* input_berita_acara adalah 2. Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan, maka ditentukan satu buah basis set dari jalur independen, yaitu :

Jalur 1 : 1 – 2 – 13– 14 – 15

Jalur 2 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 16 – 17 – 18 – 19 – 20 – 21 – 22 – 23– 24 – 25 – 26 – 27 – 32 – 12 – 15

Jalur 3 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 16 – 17 – 18 – 19 – 20 – 21 – 22 – 23 – 24 – 25 – 26– 28 – 29 – 32 – 12 – 15

Jalur 4 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 16 – 17 – 18 – 19 – 20 – 21 – 22 – 23 – 24 – 25 – 26– 30 – 31– 32 – 12 – 15

Tabel 5.6 Kasus Uji *function_input*_dokumen()

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Jika file arsip tidak ter-*upload*	Kembali ke dokumen_form_input. php	Menampilkan form input arsip
2	Jika file arsip ter-*upload* dengan media berupa text	Menyimpan data arsip yang berupa text dan menampilkan pesan data berhasil disimpan	Menyimpan data arsip yang berupa text dan menampilkan pesan data berhasil disimpan

3	Jika file arsip ter-*upload* dengan media berupa audio	Menyimpan data arsip yang berupa audio dan menampilkan pesan data berhasil disimpan	Menyimpan data arsip yang berupa audio dan menampilkan pesan data berhasil disimpan
4	Jika file arsip ter-*upload* dengan media berupa video	Menyimpan data arsip yang berupa video dan menampilkan pesan data berhasil disimpan	Menyimpan data arsip yang berupa video dan menampilkan pesan data berhasil disimpan

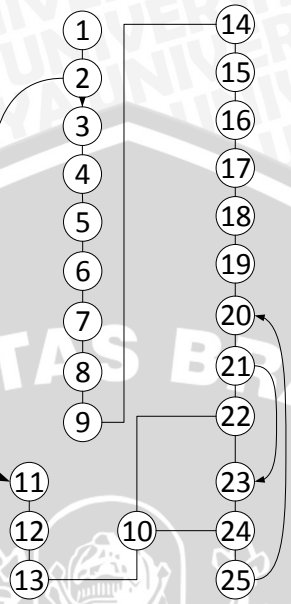
2. Pengujian Unit untuk *function_input_surat masuk()*

Class function input_surat_masuk() merupakan fungsi untuk menambahkan data surat masuk. Pada *class function* input_surat() masuk seleksi kondisi *if* pertama untuk menambah data surat masuk jika data berhasil di post, seleksi kondisi *if* kedua menambah data jika nama file sama dan pengulangan *for* untuk mengulang data bagian sebagai disposisi. Tabel 5.7 dan gambar 5.29 merupakan tabel dan dan gambar yang menampilkan kode program *function* input_surat_masuk().

Tabel 5.7 Kode Program *Class function* input_surat_masuk

```
function input_surat_masuk()                                        (1)
{
  $file = $this->input->post('file');                               (2)
  if($this->input->post()){                                         (3)
      $config['upload_path']      = 'data/dokumen/SUB.BAG TATA
USAHA/';
      $config['allowed_types']    = 'pdf|docx|doc|xlx';
      $config['max_size']         = '90000000000000000';
      $config['overwrite']        = FALSE;
      $config['remove_spaces']    = TRUE;
      $this->load->library('upload', $config);                      (4)
      $this->upload->do_upload('file');                             (5)
      $getcon = $this->upload->data();                              (6)
      $namafile = $getcon['file_name'];                             (7)
      $this->load->model('suratmodel');                             (8)
      $this->suratmodel->input_data_surat_masuk($namafile);
      $this->session->set_flashdata('input', 'input');              (9)
      redirect('sistem_c/menu/surat_masuk');                        (10)
  } else {                                                          (11)
      $data['msg']  ='';
      $data['mod']  ='surat';
      $data['page'] ='fsm';
      $data['div']  = $this->session->userdata('divisi');
      $this->load->view('sistem_v/index', $data);
```

```
  }                                                                    (12)
}                                                                      (13)

function input_data_surat_masuk($namafile)                             (14)
{
  $no     = $this->input->post('no');                                  (15)
  $sifat = $this->input->post('sifat');
  $nip        = $this->input->post('nip');
  $nama       = $this->input->post('nama');
  $hal        = $this->input->post('hal');
  $lokasi     = $this->input->post('lokasi');
  $tmp        = $this->input->post('tmp');
  $dari       = $this->input->post('dari');
  $klas       = $this->input->post('klas');
  $detail     = $this->input->post('detail');
  $tgl_masuk  = $this->input->post('tgl_masuk');
  $tgl        = date("Y-m-d");
  $fi         = $namafile;
  $bag        = $this->input->post('bag');
  $result     = $this->db->query("SELECT * FROM klasifikasi where      (16)
                id_klas='$klas'");
  $row        = $result->row_array();
  $ra         = $row['r_aktif'];
  $re         = $row['r_inaktif'];
  $tg         = date_create($tgl_masuk);

date_add($tg,date_interval_create_from_date_string("".$ra."           (17)
years"));
  $raktif     = date_format($tg,"Y-m-d");
  $tg1        = date_create($raktif);

date_add($tg1,date_interval_create_from_date_string("".$re."          (18)
years"));
  $rinaktif   = date_format($tg1,"Y-m-d");
  $ket        = $this->input->post('ket');
  $k_dok      = md5(($this->input->post('nama')).($this->input-
>post('tgl_masuk')));

$this->db->query("insert into dokumen
values('','$no','$nip','$klas','4','$nama','$dari','','$hal','$t      (19)
gl_masuk','$tgl','','Aktif','$raktif','$rinaktif','TEXT','','$fi
','','$lokasi','$tmp','$detail','$ket','1','$k_dok','$sifat','')
");

 $bagian     = $this->input->post('bagian');
 $jml        = sizeof($bagian);

for($x=0;$x<$jml;$x++){                                                (20)
 if (($jml-1)==$x) {                                                   (21)
     $k_dispo= md5(($this->input->post('nama')).
     ($this->input->post('tgl_masuk')).                                (22)
      ($bagian[$x]));
  $this->db->query("insert into disposisi
  values('','$no','$bagian[$x]','$tgl','1','1','$k_dispo')");
} else {                                                               (23)
   $k_dispo   = md5(($this->input->post('nama')).
  ($this->input->post('tgl_masuk')).($bagian[$x]));                    (24)
  $this->db->query("insert into disposisi
  values('','$no','$bagian[$x]','$tgl','1','1','$k_dispo')");
                  }                                                    (25)
}

}
```



Gambar 5.29 *Flowgraph Class* function input_surat_masuk()

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

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

$V(G) = 27 - 25 + 2$

$V(G) = 4$

Jadi nilai *cyclomatic complexity* untuk *flowgraph class function* input_berita_acara adalah 4. Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan, maka ditentukan satu buah basis set dari jalur independen, yaitu :

Jalur 1 : 1 – 2 – 11– 12 – 13

Jalur 2 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 14 – 15 – 16 – 17 – 18 – 19 – 20 – 21 – – 22 – 10 – 13

Jalur 3 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 14 – 15 – 16 – 17– 18 – 19 – 20 – 21 – 23 – 24 – 25 – 20 – 21– 23 – 24 – 10 – 13

Jalur 4 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 14 – 15 – 16 – 17– 18 – 19 – 20 – 21 – 23 – 24 – 10 – 13

Tabel 5.8 Kasus Uji *Class function* input_surat_masuk

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Jika file surat masuk tidak ter-*upload*	Kembali ke halaman surat_masuk_form_input.php	Menampilkan halaman form input surat masuk
2	Jika file surat masuk ter-*upload* dan disposisi berjumlah 1 bagian	Menyimpan data *array* dan *post* surat masuk serta menampilkan pesan data berhasil disimpan	Menyimpan data surat masuk dan menampilkan pesan data berhasil disimpan
3	Jika file surat masuk ter-*upload* dan disposisi berjumlah lebih dari 2 bagian	Menyimpan data *array* dan *post* surat masuk serta menampilkan pesan data berhasil disimpan	Menyimpan data surat masuk dan menampilkan pesan data berhasil disimpan
4	Jika file surat masuk ter-*upload* dan disposisi berjumlah 2 bagian	Menyimpan data *array* dan *post* surat masuk serta menampilkan pesan data berhasil disimpan	Menyimpan data surat masuk dan menampilkan pesan data berhasil disimpan

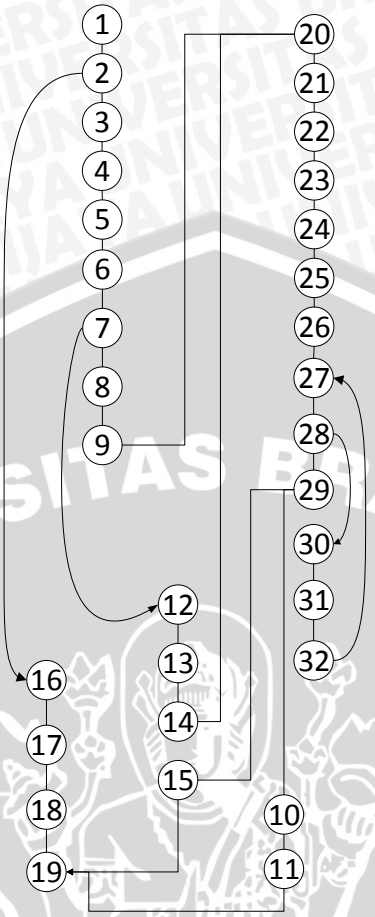
3. Pengujian Unit untuk *function edit_surat_masuk()*

Class function edit_surat_masuk() merupakan fungsi untuk mengubah data surat masuk. Pada *class function* edit_surat masuk() seleksi kondisi *if* untuk mengubah data surat masuk, pengulangan *for* untuk, dan seleksi kondisi *if*. Tabel 5.9 dan gambar 5.30 merupakan tabel dan gambar yang menampilkan kode program *function* edit_surat_masuk().

Tabel 5.9 Kode Program *Class function* edit_surat_masuk

```
function edit_surat_masuk($en)                                          (1)
{
if($this->input->post()){                                               (2)
    $config['upload_path'] = 'data/dokumen/SUB.BAG TATA                 (3)
USAHA/';
    $config['allowed_types'] = 'pdf|docx|doc|xlx';
    $config['max_size']             = '10000000000000000';
    $this->load->library('upload', $config);                            (4)
    $this->upload->do_upload('file');                                   (5)
    $getcon  = $this->upload->data();                                   (6)
    $namafile = $getcon['file_name'];                                   (7)
    if($namafile!="")
    {                                                                   (8)
        $this->load->model('suratmodel');                               (9)
        $this->suratmodel->edit_data_surat_masuk($en);
        $this->db->query(" update dokumen set file_dok
='$namafile' where k_dok='$en'");                                       (10)
        $this->session->set_flashdata('edit', 'edit');                  (11)
        redirect('sistem_c/menu/surat_masuk');                          (12)
  }else{                                                                (13)
        $this->load->model('suratmodel');                               (14)
        $this->suratmodel->edit_data_surat_masuk2($en);
        $this->session->set_flashdata('edit', 'edit');                  (15)
        redirect('sistem_c/menu/surat_masuk');
    }                                                                   (16)
} else {                                                                (17)
  $data['msg']      ='';
  $data['en']       ='$en';
  $data['mod']      ='surat';
  $data['page']     ='fsme';
  $data['div']      = $this->session->userdata('divisi');
  $this->load->view('sistem_v/index', $data);                           (18)
}                                                                       (19)
}                                                                       (20)
function edit_data_surat_masuk($en)
{                                                                       (21)
    $no             = $this->input->post('no');
    $sifat          = $this->input->post('sifat');
    $nip            = $this->input->post('nip');
    $nama           = $this->input->post('nama');
    $hal            = $this->input->post('hal');
    $lokasi         = $this->input->post('lokasi');
    $tmp            = $this->input->post('tmp');
    $dari           = $this->input->post('dari');
    $klas           = $this->input->post('klas');
    $detail         = $this->input->post('detail');
    $tgl_masuk      = $this->input->post('tgl_masuk');
    $tgl            = date("Y-m-d");
    $bag            = $this->input->post('bag');                        (22)
    $result  = $this->db->query("SELECT * FROM klasifikasi
where id_klas='$klas'");
    $row            = $result->row_array();
    $ra                 = $row['r_aktif'];
    $re                 = $row['r_inaktif'];
    $tg                 = date_create($tgl_masuk);
                                                                        (23)
date_add($tg,date_interval_create_from_date_string(".$ra."
years"));
    $raktif         = date_format($tg,"Y-m-d");
    $tg1            = date_create($raktif);
                                                                        (24)
```

```
date_add($tg1,date_interval_create_from_date_string(".$re."
years"));
    $rinaktif= date_format($tg1,"Y-m-d");
    $ket            = $this->input->post('ket');
    $k_dok          = md5(($this->input->post('nama')).($this-
>input->post('tgl_masuk')));
    $result1  = $this->db->query("SELECT no_dok FROM dokumen
where k_dok='$en'");
    $row1           = $result1->row_array();                    (25)
                                                                (26)
    $this->db->query("delete from disposisi where
no_dok='$row1[no_dok]'");
    $this->db->query(" update dokumen set
                            no_dok          ='$no',
                            nip             ='$nip,
                            id_klas         ='$klas',
                            nama_dok        ='$nama',
                            dari            ='$dari',
                            hal             ='$hal',
                            tgl_masuk_dok   ='$tgl_masuk',
                            tgl_dok         ='$tgl',
                            re_aktif        ='$raktif',
                            re_inaktif      ='$rinaktif',
                            id_lokasi       ='$lokasi',
                            id_tp           ='$tmp',
                            id_dt           ='$detail',
                            reg_dok         ='1'
                       where
                            k_dok='$en'");                      (27)
        $bagian         = $this->input->post('bagian');
    $jml = sizeof($bagian);                                     (28)
    for($x=0;$x<$jml;$x++){
        if (($jml-1)==$x) {                                     (29)
            $k_dispo    = md5(($this->input-
>post('nama')).($this->input-
>post('tgl_masuk')).($bagian[$x]));
            $this->db->query("insert into disposisi             (30)
    values('','$no','$bagian[$x]','$tgl','1','1','$k_dispo')");
        } else {
            $k_dispo    = md5(($this->input-
>post('nama')).($this->input-
>post('tgl_masuk')).($bagian[$x]));                             (31)
            $this->db->query("insert into disposisi
    values('','$no','$bagian[$x]','$tgl','1','1','$k_dispo')");
        }                                                       (32)
    }
}
```



Gambar 5.30 *Flowgraph* Class *function* edit_surat_masuk()

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

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

$V(G) = 35 - 32 + 2$

$V(G) = 5$

Jadi nilai *cyclomatic complexity* untuk *flowgraph class function* edit_surat_masuk adalah 4. Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan, maka ditentukan satu buah basis set dari jalur independen, yaitu :

Jalur 1 : 1 – 2 – 16 – 17 – 18 – 19

Jalur 2 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 12 –13 – 14 – 20 – 21 – 22– 23 – 24 – 25 – 26 – 27 – 28 – 29 – 15 – 19

Jalur 3 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 12 – 13 – 14 – 20 – 21 – 22– 23 – 24 – 25 – 26 – 27 – 28 – 30 – 31 – 32 – 27 – 28 – 29 – 15 – 19

Jalur 4 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 20 – 21 – 22– 23 – 24 – 25 – 26 – 27 – 28 – 29 – 10 – 11 – 19

Jalur 5 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 20 – 21 – 22– 23 – 24 – 25 – 26 – 27 – 28 – 30 – 31 – 32 – 27 – 28 – 29 – 10 – 11 – 19

Tabel 5.10 Kasus Uji *Class function* edit_surat_masuk

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Jika file surat masuk tidak ter-*upload*	Kembali ke halaman surat_masuk_form _edit.php	Menampilkan halaman form edit surat masuk
2	Jika data surat masuk ter-*update* ,nama data file surat masuk sama dan disposisi bagian berjumlah 1	Menyimpan data *array* dan *post* surat masuk serta menampilkan pesan data berhasil diubah	Menyimpan data surat masuk dan menampilkan pesan data berhasil diubah
3	Jika data surat masuk ter-*update* , nama data file surat masuk sama dan data disposisi bagian berjumlah sebanyak data yang diinputkan.	Menyimpan data *array* dan *post* surat masuk serta menampilkan pesan data berhasil diubah	Menyimpan data surat masuk dan menampilkan pesan data berhasil diubah
4	Jika data surat masuk ter-*update*, nama data file surat masuk tidak	Menyimpan data *array* dan *post* surat masuk serta	Menyimpan data surat masuk dan

	sama dan data disposisi bagian berjumlah 1	menampilkan pesan data berhasil diubah	menampilkan pesan data berhasil diubah
5	Jika data surat masuk ter-*update* , nama data file surat masuk tidak sama dan data disposisi bagian berjumlah sebanyak data yang diinputkan.	Menyimpan data *array* dan *post* surat masuk serta menampilkan pesan data berhasil diubah	Menyimpan data surat masuk dan menampilkan pesan data berhasil diubah

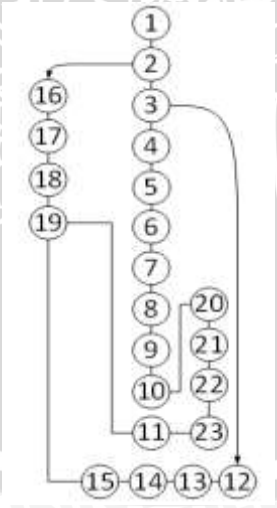
4. Pengujian Unit *edit_surat_keluar*

Class function edit_surat_keluar merupakan fungsi untuk mengubah data surat keluar. Pada *class function* input_surat keluar seleksi kondisi *if* pertama untuk jika data di postkan, Seleksi kondisi *if* kedua untuk mengubah data file surat keluar jika nama file sama. Tabel 5.11 dan gambar 5.31 merupakan tabel dan gambar yang menampilkan kode program *function* edit_surat_ keluar.

Tabel 5.11 Kode Program *Class function* edit_surat_keluar

Kode Program	Node
function edit_surat_keluar($en)	(1)
{	
if($this->input->post()){	(2)
$file = $this->input->post('file');	(3)
if($file!=""){	(4)
$config['*upload*_path'] = 'data/dokumen/SUB.BAG TATA USAHA/';	(5)
$config['allowed_types'] = 'pdf|docx|doc|xlx';	
$config['max_size'] = '1000000000000000';	
$this->load->library('*upload*', $config);	(6)
$this->*upload*->do_*upload*('file');	(7)
$getcon = $this->*upload*->data();	(8)
$namafile = $getcon['file_name'];	
$this->load->model('suratmodel');	(9)
$this->suratmodel->edit_data_surat_keluar($en);	(10)
$this->db->query(" update dokumen set file_dok ='$namafile' where k_dok ='$en'");	
$this->session->set_flashdata('edit', 'edit');	
redirect('sistem_c/menu/surat_keluar');	(11)
}else{	(12)
$this->load->model('suratmodel');	(13)
$this->suratmodel->edit_data_surat_keluar($en);	(14)
$this->session->set_flashdata('edit', 'edit');	
redirect('sistem_c/menu/surat_keluar');	(15)
}} else {	(16)

```
      $data['msg']  ='';                                                    (17)
      $data['en']          ='$en';
      $data['mod']  ='surat';
      $data['page'] ='fske';
      $data['div']  = $this->session->userdata('divisi');
      $this->load->view('sistem_v/index', $data);                           (18)
  }}                                                                        (19)
function edit_data_surat_keluar($en)                                       (20)
{
    $urut           = $this->input->post('urut');                           (21)
    $no             = $this->input->post('no');
    $p              = $this->input->post('p');
    $b              = $this->input->post('b');
    $thn            = $this->input->post('thn');
    $no2            = $this->input->post('klas');
    $p              = $this->input->post('p');
    $b              = $this->input->post('b');
    $thn            = $this->input->post('thn');
    $tgl            = date('Y-m-d');
    $no_su          = $urut."/".$no."/".$p."/".$b."/".$thn;
    $nama           = $this->input->post('nama');
    $tgl_keluar     = $this->input->post('tgl_dok');
    $k_dok    = md5(($this->input->post('nama')).($this->input-
>post('no')));
    $this->db->query(" update dokumen set                                   (22)
                        no_dok          ='$no_su',
                        id_klas         ='$no2',
                        nama_dok        ='$nama',
                        tgl_keluar_dok  ='$tgl_keluar',
                        tgl_dok         ='$tgl',
                        format_dok      ='',
                        k_dok           ='$k_dok'
                      where
                        k_dok='$en'");}                                     (23)
```



Gambar 5.31 *Flowgraph Class function* edit_surat_keluar()

Pemodelan dalam *flow graph* yang telah dilakukan terhadap *class function* edit_surat_keluar menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan V(G) = E – N + 2, dimana V(G) merupakan jumlah

kompleksitas siklomatis, E merupakan sisi atau edge (garis penghubung antar node) dan N merupakan jumlah simpul (node).

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

$V(G) = 24 - 23 + 2$

$V(G) = 3$

Jadi nilai *cyclomatic complexity* untuk *flowgraph class function* input_berita_acara adalah 4. Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan, maka ditentukan satu buah basis set dari jalur independen, yaitu :

Jalur 1 : 1 – 2 – 16 – 17 – 18 – 19

Jalur 2 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 20 – 21 – 23 – 11 – 19

Jalur 3 : 1 – 2 – 3 – 12 – 13 – 14 – 15 – 19

Tabel 5.12 Kode Program *Class function* edit_surat_keluar

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Jika file surat keluar tidak ter-*upload*	Kembali ke halaman surat_keluar_form_edit. php	Menampilkan halaman form edit surat keluar
2	Jika data surat keluar ter-*update* dan nama data file surat keluar yang sama	Menyimpan data *array* dan *post* surat keluar serta menampilkan pesan data berhasil diubah	Menyimpan data surat keluar dan menampilkan pesan data berhasil diubah
3	Jika data surat keluar ter-*update* dan nama data file surat keluar yang sama	Menyimpan data *array* dan *post* surat keluar serta menampilkan pesan data berhasil diubah	Menyimpan data surat keluar dan menampilkan pesan data berhasil diubah

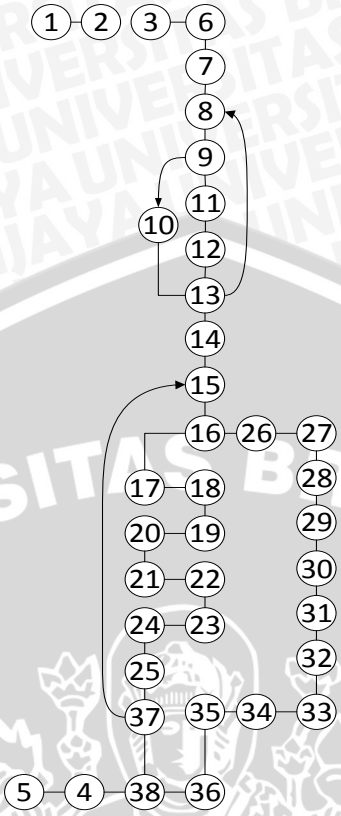
5. Pengujian Unit *function input_berita_acara*

Class function input_berita_acara merupakan fungsi untuk menambahkan data berita acara pemusnahan arsip. Pada *class function* input_ berita_acara seleksi kondisi *if* untuk seleksi data saksi lebih dari satu orang, pengulangan *for* untuk mengulang data saksi sebanyak jumlah saksi yang di inputkan, dan seleksi kondisi *if* untuk seleksi data arsip yang akan dimusnahkan. Tabel 5.13 dan gambar 5.32 merupakan tabel dan dan gambar yang menampilkan kode program *function* input_ berita_acara

Tabel 5.13 Kode Program *Class function* input_berita_acara

Kode Program	Node
function input_berita_acara()	(1)
{	
$this->load->model('beritamodel');	(2)
$this->beritamodel->input_data_berita_acara();	(3)
$this->session->set_flashdata('input', 'input');	
redirect('sistem_c/menu/berita_acara');	(4)
}	(5)
function input_data_berita_acara()	(6)
{	
$no = $this->input->post('no');	(7)
$tgl = $this->input->post('tgl');	
$cp = $this->input->post('cp');	
$saksi1 = $this->input->post('saksi1');	
$saksi2 = $this->input->post('saksi2');	
$tgl = $this->input->post('tgl');	
$date = date('Y-m-d');	
$b = count($cp);	
$k_ba = md5($this->input->post('no').$tgl);	
$s = $saksi1.'\|'.$saksi2;	
for($i=0;$i<$b;$i++)	(8)
{	
if($i==0)	(9)
{	
$c = $cp[$i];	(10)
}	
else	(11)
{	
$c = $c.'\|'.$cp[$i];	(12)
}	
}	(13)
	(14)
$this->db->query("insert into berita_acara	
values('$no','$date','$date','$c','$s','$k_ba')");	
$dok = $this->input->post('dok');	
$jml = sizeof($dok);	
for($x=0;$x<$jml;$x++){	(15)
if (($jml-1)==$x) {	(16)
$cek = $this->db->query("SELECT * FROM	(17)
dokumen d,klasifikasi k,kategori_dokumen kd,bagian b	
WHERE	
d.id_klas=k.id_klas and	
d.id_bagian=b.id_bagian and	
k.id_katdok=kd.id_katdok and	

```
                        d.no_dok='$dok[$x]'");
      $row    = $cek->row_array();
      $nama   = $row['nama_dok'];
      $klas   = $row['nama_klas'];
      $katdok= $row['nama_katdok'];
      $kode   = md5($nama.$klas.$katdok.$no);

      $this->db->query("insert into detail_berita_acara                (18)
values('','$no','$dok[$x]','$nama','$klas','$katdok','$kode')");
                        $this->db->query("delete from dokumen
where no_dok='$dok[$x]'");
      $url = base_url();
      $pathFile =
$url."data/dokumen/".$row['nama_bagian']."/".$row['file_dok'];    (19)
      $cek1   = $this->db->query("SELECT * from disposisi where
no_dok='$dok[$x]'");
if($cek1->num_rows() > 0){                                        (20)
$this->db->query("delete from disposisi where                     (21)
no_dok='$dok[$x]'");}
$cek2  = $this->db->query("SELECT * from revisi where
no_dok='$dok[$x]'");
      if($cek2->num_rows() > 0){                                  (22)
      $this->db->query("delete from revisi where                  (23)
no_dok='$dok[$x]'");
      $pathFile1 =
$url."data/dokumen/".$row['nama_bagian']."/".$row['file_versi'];  (24)
      }
      } else {                                                    (25)
      $cek    = $this->db->query("SELECT * FROM
                dokumen d,klasifikasi k,kategori_dokumen
kd,bagian b
                WHERE d.id_klas=k.id_klas and                     (26)
                      d.id_bagian=b.id_bagian and                 (27)
                      k.id_katdok=kd.id_katdok and
                      d.no_dok='$dok[$x]'");
      $row    = $cek->row_array();
      $nama   = $row['nama_dok'];
      $klas   = $row['nama_klas'];
      $katdok= $row['nama_katdok'];                               (28)
      $kode   = md5($nama.$klas.$katdok.$no);

      $this->db->query("insert into detail_berita_acara
values('','$no','$dok[$x]','$nama','$klas','$katdok','$kode')");
      $this->db->query("delete from dokumen where
no_dok='$dok[$x]'");                                              (29)
      $url = base_url();
      $pathFile =                                                 (30)
$url."data/dokumen/".$row['nama_bagian']."/".$row['file_dok'];
      $cek1   = $this->db->query("SELECT * from disposisi where
no_dok='$dok[$x]'");
      if($cek1->num_rows() > 0){ $this->db->query("delete from    (31)
disposisi where no_dok='$dok[$x]'");}
      $cek2  = $this->db->query("SELECT * from revisi where       (32)
no_dok='$dok[$x]'");                                              (33)
      if($cek2->num_rows() > 0){                                  (34)
      $this->db->query("delete from revisi where                  (35)
no_dok='$dok[$x]'");
      $pathFile1 =                                                (36)
$url."data/dokumen/".$row['nama_bagian']."/".$row['file_versi'];
      }                                                           (37)
  }                                                               (38)
  }
}
```



Gambar 5.32 *Flowgraph Class function* input_berita_acara()

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

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

$V(G) = 40 - 38 + 2$

$V(G) = 4$

Jadi nilai *cyclomatic complexity* untuk *flowgraph class function* input_berita_acara adalah 4. Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan, maka ditentukan satu buah basis set dari jalur independen, yaitu :

Jalur 1 : 1 – 2 – 3 – 6 – 7 – 8 – 9 – 11 – 12 – 13 – 8 – 9 – 11 – 12 – 13 – 14 – 15 – 16 – 26 – 27 – 28 – 29 – 30 – 31 – 32 – 33 – 34 – 35 – 36 – 38 – 4 – 5

Jalur 2 : 1 – 2 – 3 – 6 – 7 – 8 – 9 – 11 – 12 – 13 – 14 – 15 – 16 – 26 – 27 – 28 – 29 – 30 – 31 – 32 – 33 – 34 – 35 – 36 – 38 – 4 – 5

Jalur 3 : 1 – 2 – 3 – 6 – 7 – 8 – 9 – 10 – 13 – 8 – 9 – 10 – 13 – 14 – 15 – 16 – 17 – 18 – 19 – 20 – 21 – 22 – 23 – 24 – 25 – 37 – 15 – 16 – 17 – 18 – 19 – 20 – 21 – 22 – 23 – 24 – 25 – 37 – 38 – 4 – 5

Jalur 4 : 1 – 2 – 3 – 6 – 7 – 8 – 9 – 10 – 13 – 14 – 15 – 16 – 17 – 18 – 19 – 20 – 21 – 22 – 23 – 24 – 25 – 37 – 38 – 4 – 5

Tabel 5.14 Kode Program *Class function* input_berita_acara

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Input data berita acara jika saksi lebih dari 1 orang dan menghapus 1 data arsip	Menyimpan data *array* dan *post* berita acara serta menampilkan pesan data berhasil dihapus	Menyimpan data surat keluar dan menampilkan pesan data berhasil hapus
2	Input data berita acara jika saksi 1 orang dan menghapus data sebanyak data arsip yang akan di musnahkan	Menyimpan data *array* dan *post* berita acara serta menampilkan pesan data berhasil dihapus	Menyimpan data surat keluar dan menampilkan pesan data berhasil hapus
3	Input data berita acara jika saksi lebih dari 1 orang dan menghapus data sebanyak data arsip yang akan di musnahkan	Menyimpan data *array* dan *post* berita acara serta menampilkan pesan data berhasil dihapus	Menyimpan data surat keluar dan menampilkan pesan data berhasil hapus
4	Input data berita acara jika saksi lebih dari 1	Menyimpan data *array* dan *post* berita acara serta	Menyimpan data surat keluar dan

	orang dan jumlah arsip sama dengan 1	menampilkan pesan data berhasil dihapus	menampilkan pesan data berhasil hapus

5.5.2. Pengujian Validasi

Pengujian validasi bertujuan untuk mengetahui apakah sistem yang sudah dibangun telah sesuai dengan spesifikasi yang ada pada perancangan. *Item – item* yang telah dirumuskan dalam daftar kebutuhan dan merupakan hasil analisis kebutuhan akan menjadi acuan untuk melakukan pengujian validasi. Pengujian validasi menggunakan metode *black box*, karena tidak diperlukan konsentrasi terhadap jalur jalannya algoritma program dan lebih ditekankan untuk menemukan konformitas antara kinerja sistem dengan daftar kebutuhan.

Tabel 5.15 Hasil Pengujian Validasi

No Uji	No. SRS	Kebutuhan	Kasus Uji	Status
1.	SRS_6_01	Tambah Data klasifikasi arsip	Klik *button* tambah data, masukkan data klasifikasi, klik *button* simpan	Valid
2.	SRS_6_02	Ubah Data klasifikasi arsip	Klik *icon* ubah pada data klasifikasi yang ingin diubah, ubah data klasifikasi sesuai form, klik *button* simpan	Valid
3.	SRS_6_03	Hapus Data klasifikasi arsip	Klik *icon* hapus pada data klasifikasi yang ingin dihapus, klik Ok	Valid
4.	SRS_6_04	Lihat Data klasifikasi arsip	Pilih menu klasifikasi arsip, pilih Data Klasifikasi.	Valid
5.	SRS_6_05	Tambah kategori arsip	Klik *button* tambah data, masukkan data kategori arsip, klik *button* simpan	Valid
6.	SRS_6_06	Ubah kategori arsip	Klik *icon* ubah pada data kategori yang ingin diubah,	Valid

			ubah data kategori sesuai form, klik *button* simpan	
7.	SRS_6_07	Hapus kategori arsip	Klik *icon* hapus pada data kategori yang ingin dihapus, klik Ok	Valid
8.	SRS_6_08	Lihat kategori arsip	Pilih menu klasifikasi arsip, pilih Kategori Arsip.	Valid
9.	SRS_4_01	Tambah data arsip	Klik *button* tambah data, masukkan data arsip, klik *button* simpan	Valid
10.	SRS_4_02	Hapus data arsip	Klik *icon* hapus pada data arsip yang ingin dihapus, klik Ok	Valid
11.	SRS_4_03	Ubah data arsip	Klik *icon* ubah pada data arsip yang ingin diubah, ubah data arsip sesuai form, klik *button* simpan	Valid
12.	SRS_4_04	Lihat data arsip	Pilih menu Data arsip, pilih Arsip.	Valid
13.	SRS_6_01	Validasi registrasi arsip	klik notifikasi arsip baru, klik *button* belum terdaftar pada form detail arsip.	Valid
14.	SRS_5_01	Tambah data surat masuk	Klik *button* tambah data, masukkan data surat masuk, klik *button* simpan	Valid
15.	SRS_5_02	Hapus data surat masuk	Klik *icon* hapus pada data surat masuk yang ingin dihapus, klik Ok	Valid
16.	SRS_5_03	Ubah data surat masuk	Klik *icon* ubah pada data surat masuk yang ingin diubah,	Valid

			ubah data surat masuk sesuai form, klik *button* simpan	
17.	SRS_5_04	Lihat data surat masuk	Pilih menu Persuratan, pilih surat masuk.	Valid
18.	SRS_5_05	Tambah data surat keluar	Klik *button* tambah data, masukkan data surat keluar, klik *button* simpan	Valid
19.	SRS_5_06	Hapus data surat keluar	Klik *icon* hapus pada data surat keluar yang ingin dihapus, klik Ok	Valid
20.	SRS_5_07	Ubah data surat keluar	Klik *icon* ubah pada data surat keluar yang ingin diubah, ubah data surat keluar sesuai form, klik *button* simpan	Valid
21.	SRS_5_08	Lihat data surat keluar	Pilih menu Persuratan, pilih surat keluar.	Valid
22.	SRS_5_09	Tambah data konsep surat keluar	Klik *button* tambah data, masukkan konsep surat, klik *button* simpan	Valid
23.	SRS_5_10	Hapus data konsep surat keluar	Klik *icon* hapus pada data konsep surat yang ingin dihapus, klik Ok	Valid
24.	SRS_5_11	Ubah data konsep surat keluar	Klik *icon* ubah pada data konsep surat yang ingin diubah, ubah konsep surat sesuai form, klik *button* simpan	Valid
25.	SRS_5_12	Lihat data konsep surat keluar	Pilih menu Persuratan, pilih Data Konsep Surat Keluar.	Valid

26.	SRS_4_13	Daftar Arsip Usul Musnah	Klik notifikasi retensi arsip, klik *icon* hapus pada form penilaian arsip	Valid
27.	SRS_6_09	Tambah Data Berita Acara	Klik *button* tambah data, masukkan data berita acara pemusnahan arsip, klik *button* simpan	Valid
28.	SRS_6_10	Lihat Data Berita Acara	Pilih menu Berita Acara Pemusnahan, pilih Berita Acara Pemusnahan	Valid
29.	SRS_6_11	Cetak Berita Acara	Klik *icon* print pada halaman berita acara pemusnahan	Valid
30.	SRS_4_05	Tambah data revisi	Klik *button* tambah data, masukkan data revisi, klik *button* simpan	Valid
31.	SRS_4_06	Lihat data revisi	Pilih menu Data Arsip, pilih Data Revisi Arsip.	Valid
32.	SRS_6_12	Validasi revisi arsip	Klik notifikasi revisi arsip, klik *button* belum terdaftar pada form revisi arsip.	Valid
33.	SRS_4_09	Tambah Data Peminjaman	Klik *button* tambah data, masukkan data peminjaman, klik *button* simpan	Valid
34.	SRS_4_10	Hapus Data Peminjaman	Klik *icon* hapus pada data peminjaman yang ingin dihapus, klik Ok	Valid
35.	SRS_4_11	Lihat Data Peminjaman	Pilih menu Peminjaman, pilih Peminjaman.	Valid
36.	SRS_7_01	Lihat Laporan	Pilih menu data yang akan dilihat, klik *icon* print.	Valid

37.	SRS_3_13	Tambah data bagian	Klik *button* tambah data, masukkan data bagian, klik *button* simpan	Valid
38.	SRS_3_14	Hapus data bagian	Klik *icon* hapus pada data bagian yang ingin dihapus, klik Ok	Valid
39.	SRS_3_15	Ubah data bagian	Klik *icon* ubah pada data bagian yang ingin diubah, ubah data bagian sesuai form, klik *button* simpan	Valid
40.	SRS_3_16	Lihat data bagian	Pilih menu Data Karyawan, pilih Data Bagian.	Valid
41.	SRS_3_01	Tambah data karyawan	Klik *button* tambah data, pada halaman data karyawan, masukkan data karyawan, klik *button* simpan	Valid
42.	SRS_3_02	Hapus data karyawan	Klik *icon* hapus pada data karyawan yang ingin dihapus, klik Ok	Valid
43.	SRS_3_03	Ubah data karyawan	Klik *icon* ubah pada data karyawan yang ingin diubah, ubah data karyawan sesuai form, klik *button* simpan	Valid
44.	SRS_3_04	Lihat data karyawan	Pilih menu Data Karyawan, pilih Data Karyawan.	Valid
45.	SRS_3_05	Tambah data jabatan	Klik *button* tambah data, masukkan data jabatan, klik *button* simpan	Valid
46.	SRS_3_06	Hapus data jabatan	Klik *icon* hapus pada data jabatan yang ingin dihapus, klik Ok	Valid

47.	SRS_3_07	Ubah data jabatan	Klik *icon* ubah pada data jabatan yang ingin diubah, ubah data jabatan sesuai form, klik *button* simpan	Valid
48.	SRS_3_08	Lihat data jabatan	Pilih menu Data Karyawan, pilih Data Jabatan.	Valid
49.	SRS_3_09	Tambah jabatan fungsional	Klik *button* tambah data, masukkan data jabatan fungsional, klik *button* simpan	Valid
50.	SRS_3_10	Hapus jabatan fungsional	Klik *icon* hapus pada data jabatan fungsional yang ingin dihapus, klik Ok	Valid
51.	SRS_3_11	Ubah jabatan fungsional	Klik *icon* ubah pada data jabatan fungsional yang ingin diubah, ubah data jabatan fungsional sesuai form, klik *button* simpan	Valid
52.	SRS_3_12	Lihat jabatan fungsional	Pilih menu Data Karyawan, pilih Jabatan Fungsional.	Valid
53.	SRS_1_01	Login	Masukkan username dan password user yang sesuai dengan level admin, masukkan username dan password yang salah	Valid
54.	SRS_1_04	Logout	Klik *icon* yang bertuliskan logout	Valid
55.	SRS_8_01	Tambah User	Klik *button* tambah data, pada halaman data user masukkan data user, klik *button* simpan	Valid
56.	SRS_8_02	Hapus User	Klik *icon* hapus pada data user yang ingin dihapus, klik Ok	Valid

57.	SRS_8_03	Ubah User	Klik *icon* ubah pada data user yang ingin diubah, ubah data user sesuai form, klik *button* simpan	Valid
58.	SRS_8_04	Lihat User	Pilih menu Data User, pilih Data User.	Valid

5.5.3. Pengujian Checklist ISO 15489 dan Kemampuan Sistem Terhadap Kesesuaian Standar Sistem Pencatatan ISO 15489

1. Pengujian *Checklist* ISO 15489

Checklist ISO 15489 bertujuan untuk memeriksa apakah sistem manajemen arsip UPT Laboratorium Kesehatan Hewan Malang yang sedang berjalan saat ini sudah sesuai dengan standar pencatatan ISO 15489 yang mencakup pengarsipan, pemusanahan, dan penghancuran arsip.

Tabel 5.16 Hasil Checklist Kesesuaian ISO 15489

Checklist 1 : Penciptaan Arsip			
Tujuan : Arsip diciptakan dan diterima oleh bagian yang bersangkutan dan di catat oleh bagian yang berwenang.			
Pedoman Praktek Yang Baik	Bukti	Kesesuaian (Ya/Tidak/ Sebagian)	Diperlukan Tindakan Ya/Tidak
1. Terdapat panduan terhadap apa itu arsip dan apa yang harus dilakukan untuk menjaga dan membuatnya dapat diakses melalui sistem pencatatan di setiap bagian.	Panduan secara resmi belum ada	Sebagian	Ya
2. Ada ketentuan khusus dalam merekam informasi elektronik (misalnya email)	Sudah ada ketentuan bagaimana mengelola arsip elektronik	Ya	Tidak
3. Instansi telah memiliki sebuah sistem pencatatan (*electronic record*	Sudah ada	Ya	Tidak

management system) untuk mengelola arsip nya			
4. Sistem pencatatan mencatat lokasi fisik setiap arsip yang dicatat dalam sistem	Sistem mencatat lokasi dimana arsip fisik disimpan	Ya	Tidak

Checklist 2 : Tempat Penyimpanan Arsip			
Tujuan : Penyimpanan arsip yang akurat dan dilakukan pengecekan setiap tahun			
Pedoman Praktek Yang Baik	Bukti	Kesesuaian (Ya/Tidak/ Sebagian)	Diperlukan Tindakan Ya/Tidak
1. Strategi penyimpanan arsip telah disetujui oleh instansi	Kepala UPT telah mengetahui tempat penyimpanan arsip baik arsip asli maupun arsip elektronik.	Ya	Tidak
2. Penyimpanan arsip dibedakan berdasarkan bagiannya	Prakteknya penyimpanan arsip kertas disimpan di bagian masing-masing, pada sistem file arsip dibedakan dalam folder beradasarkan bagiannya.	Ya	Tidak
3. Penyimpanan arsip dibedakan antara arsip elektronik dan arsip kertas	Jelas berbeda	Ya	Tidak
4. Penyimpanan arsip di cek dan di *update* setiap tahun	Setiap tahun diadakan pengecekan terhadap penyimpanan arsip kertas dan dan di tambahkan penanda setiap tahunnya	Ya	Tidak
5. Setiap lokasi penyimpanan mempunyai penanda unik	Dalam sebuah rak penyimpanan arsip terdapat penanda tahun untuk membedakan penyimpanan arsip setiap tahunnya	Ya	Tidak

Checklist 3 : Penyimpanan Arsip			
Tujuan : Sistem pencatatan dilindungi dari akses yang tidak sah, pengrusakan atau pencurian data.			
Pedoman Praktek Yang Baik	Bukti	Kesesuaian (Ya/Tidak/ Sebagian)	Diperlukan Tindakan Ya/Tidak
1. Tempat penyimpanan yang dialokasikan untuk menyimpan arsip fisik memiliki ruang yang cukup untuk mengakomodasi bertambahnya jumlah arsip.	Adanya lemari kayu, lemari besi dan folder-folder untuk mengantisipasi bertambahnya volume arsip	Ya	Tidak
2. Area penyimpanan untuk arsip fisik sesuai dengan standar yang disepakati untuk memastikan arsip aman dari bahaya lingkungan atau biologis, misalnya basah, kebakaran, banjir atau kontaminasi bahan kimia.	Adanya lemari besi untuk menyimpan arsip-arsip penting dan rahasia.	Ya	Tidak
3. Area penyimpanan untuk arsip elektronik (termasuk file server) memastikan catatan yang aman dari bahaya lingkungan atau biologis, misalnya basah, kebakaran, banjir, atau kontaminasi bahan kimia.	Komputer yang digunakan sebagai server diletakkan dalam tempat khusus	Ya	Tidak
4. Akses untuk masuk ke dalam sistem dibatasi untuk	Hak akses untuk masuk ke dalam sistem berbeda-beda sesuai	Ya	Tidak

mencegah usaha pengubahan dan pencurian data	dengan bagian masing-masing		

Checklist 4 : Pemusnahan Arsip

Tujuan : Arsip disimpan dan dimusnahkan sesuai dengan jadwal retensi arsip

Pedoman Praktek Yang Baik	Bukti	Kesesuaian (Ya/Tidak/ Sebagian)	Diperlukan Tindakan Ya/Tidak
1. Telah ada prosedur untuk pengarsipan dan pemusnahan arsip untuk jenis arsip yang berbeda.	Prosedur secara resmi masih mengikuti kebijakan Dinas Peternakan	Ya	Tidak
2. Instansi memiliki kebijakan tentang jadwal retensi arsip yang telah dibuat dan disepakati oleh instansi (termasuk arsip elektronik)	UPT mempunyai jadwal retensi arsip yang telah disetujui sesuai kebijakan yang berlaku pada Dinas Peternakan	Ya	Tidak
3. Pemusnahan arsip dilakukan secara teratur (sesuai dengan masa simpan arsip)	Telah ada jadwal retensi arsip yang digunakan sebagai dasar pemusnahan arsip yang telah habis masa simpannya.	Ya	Tidak
4. Keputusan pemusnahan arsip disetujui dan didokumentasikan.	Ada pengesahan untuk setiap berita acara pemusnahan arsip	Ya	Tidak

Checklist 5 : Arsip Elektronik

Tujuan : Sistem Manajememen Arsip elektronik yang ditentukan, diciptakan, dirancang, dan dioperasikan untuk mencegah akses yang tidak sah, korupsi, dan pengrusakan data.

Pedoman Praktek Yang Baik	Bukti	Kesesuaian (Ya/Tidak/ Sebagian)	Diperlukan Tindakan Ya/Tidak
1. Prosedur pencatatan sistem manajemen arsip elektronik harus mencakup : a. Tanggung jawab	Sistem manajemen arsip elektronik yang dibuat mencakup semua persayaratan	Ya	Tidak

	b. Perekaman data c. Informasi retensi d. Informasi pemusnahan e. *Back up* data f. Pemeliharaan g. Keamanan dan perlindungan h. Alur proses i. Pengubahan data informasi j. Waktu pencatatan k. Data suara dan video l. Kontrol revisi dokumen	prosedur pencatatan arsip elektronik		
2.	Ada pemeriksaan untuk setiap data arsip yang dicatat ke dalam sistem (untuk menghindari penipuan dan tindakan berbahaya lainnya)	Ada bagian yang bertugas untuk memvalidasi arsip yang baru dicatat ke dalam sistem	Ya	Tidak
3.	Tingkat akses yang tersedia untuk sistem telah didokumentasikan dan disetujui dan hanya mengizinkan staf dengan hak akses yang sah untuk memasukkan arsip baru dan mengubah informasi arsip yang sudah ada.	Hak akses sistem dibedakan sesuai dengan bagian masing-masing	Ya	Tidak
4.	Setiap riwayat perubahan arsip dicatat untuk mengantisipasi adanya audit	Pencatatan, revisi dan pemusnahan arsip didokumentasikan	Ya	Tidak
5.	Informasi audit yang dikumpulkan dan dipelihara harus : a. Dihasilkan secara otomatis	Ada riwayat dari setiap pencatatan arsip baru, revisi, dan pemusnahan arsip	Ya	Tidak

oleh sistem manajemen elektronik. b. Waktu yang akurat c. Tersedia untuk pemeriksaaan oleh petugas yang berwenang d. Disimpan dengan aman untuk mencegah perubahan data e. Tidak dimodifikasi	sesuai dengan tanggal pemrosesan.		
6. Arsip elektronik disimpan sesuai dengan jadwal retensi arsip yang telah disetujui oleh instansi.	Ketika masa simpan telah habis, maka sistem manajemen arsip menampilkan notifikasi untuk memberi peringatan.	Ya	Tidak

Checklist 6 : Keamanan dan Kerahasiaan Arsip			
Tujuan : Hak akses untuk melihat informasi arsip dibatasi sesuai kebijakan instansi.			
Pedoman Praktek Yang Baik	Bukti	Kesesuaian (Ya/Tidak/ Sebagian)	Diperlukan Tindakan Ya/Tidak
1. Pelanggaran kerahasiaan arsip, hilangnya catatan, dll dicatat sebagai insiden keamanan dan dikelola dengan tepat.	Belum ada kebijakan resmi yang disepakati mengenai pelanggaran kerahasiaan arsip	Sebagian	Ya
2. Sistem manajemen arsip dikembangkan dengan instansi lain, sehingga memungkinkan penggunaan berbagi informasi arsip rahasia	Sistem manajemen arsip baru berjalan di dalam instansi saja	Tidak	Ya

3. Seluruh staf menyadari tanggung jawab mereka mengenai arsip rahasia	Seluruh staf mempunyai hak akses yang berbeda	Ya	Tidak

Checklist 7 : Kehandalan Informasi Arsip			
Tujuan : Setiap bagian mempunyai prosedur untuk menjamin kehandalan informasi arsip masing-masing			
Pedoman Praktek Yang Baik	Bukti	Kesesuaian (Ya/Tidak/ Sebagian)	Diperlukan Tindakan Ya/Tidak
1. Staf memvalidasi hubungan informasi arsip dengan arsip yang lainnya	Untuk pencatatan surat keluar, harus disesuaikan dengan konsep surat yang telah disetujui oleh kepala UPT	Ya	Tidak
2. Terdapat pemeriksaan yang dilakukan untuk mengkonfirmasi bahwa arsip yang dicatat adalah arsip yang telah dibuat atau yang diterima oleh bagian yang berwenang.	Ada validasi pendaftaran arsip yang baru dicatat dalam sistem	Ya	Tidak
3. Adanya training dan pelatihan tentang tindak lanjut terhadap ketidaksesuaian.	Belum ada	Tidak	Ya

Checklist 8 : Manajemen Arsip			
Tujuan : Manajemen Arsip diatur, didokumentasikan, direncanakan, dan dilaksanakan sesuai dengan kebijakan yang berlaku di instansi			
Pedoman Praktek Yang Baik	Bukti	Kesesuaian (Ya/Tidak/ Sebagian)	Diperlukan Tindakan Ya/Tidak
1. Instansi memiliki kebijakan	Instansi menerapkan ISO 9001:2008, dalam	Ya	Tidak

	manajemen arsip yang telah disetujui.	ISO 9001 manajemen dokumen adalah hal yang mutlak		
2.	Instansi memiliki dewan yang bertanggung jawab atas strategi sistem manajemen arsip yang mempunyai tugas untuk menyampaikan kebijakan.	Belum ada dewan khusus yang menangani kearsipan yang ditunjuk secara resmi	Sebagian	Ya
3.	Kebijakan manajemen arsip mencakup arsip elektronik dan arsip fisik.	Telah dibuat sistem manajemen arsip elektronik untuk membantu pengelolaan arsip fisik	Ya	Tidak
4.	Kebijakan manajemen arsip ditinjau secara rutin	Diagendakan dilakukan peninjauan setahun sekali	Ya	Tidak

Checklist 9 : Pelatihan Manajemen Arsip			
Tujuan : Manajemen arsip diatur, didokumentasikan, direncanakan, dan dilaksanakan sesuai dengan kebijakan yang berlaku di instansi			
Pedoman Praktek Yang Baik	Bukti	Kesesuaian (Ya/Tidak/ Sebagian)	Diperlukan Tindakan Ya/Tidak
1. Pelatihan manajemen arsip termasuk dalam rencana pendidikan, pelatihan dan pengembangan.	Belum diadakan pelatihan manajemen kearsipan secara resmi	Sebagian	Ya
2. Staf mengerti apa yang mereka catat, bagaimana harus dicatat dan mengapa mereka mencatat	Staf mengerti apa fungsi dan tujuan dari arsip mereka masing-masing	Ya	Tidak
3. Staf dilatih utuk megidentifikasi dan memperbaiki kesalahan yang mungkin terjadi saat proses pencatatan	Dilakukan pelatihan cara penggunaan sistem manajemen arsip elektronik (oleh pembuat sistem)	Ya	Tidak

4.	Staf diarahkan untuk menjadi pengguna akhir catatan	Staf mempunyai tanggung jawab penuh atas arsip mereka masing-masing	Ya	Tidak
5.	Ada ketentuan prosedur peninjauan rutin terkait kebutuhan pelatihan dalam arsip dan teknologi informasi	Belum ada	Tidak	Ya

2. Pengujian Kemampuan Sistem Terhadap Kesesuaian Standar Sistem Pencatatan ISO 15489

Pengujian dilakukan dengan memberikan kuesioner kepada pengguna yaitu karyawan UPT Laboratorium Kesehatan Hewan Malang sebagai pengguna aplikasi meliputi kepala UPT, bagian persuratan, administratif, dan bagian pelayanan sebanyak 5 orang responden. Pernyataan yang digunakan pada pengujian ini beserta hasil kuesioner ditunjukkan pada tabel 6.12.

Tabel 5.17 Hasil Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik

NO	PERTANYAAN	JUMLAH			
		STS	TS	S	SS
1	***Reliability***				
1.1	Apakah aplikasi dapat mencatat semua data arsip?	0	0	5	0
1.2	Apakah aplikasi dapat menampilkan data arsip sewaktu-waktu saat arsip dibutuhkan?	0	1	4	0
1.3	Apakah metadata arsip yang ditampilkan dapat memberikan informasi dari arsip yang dibutuhkan?	0	0	5	0
2	***Integrity***				
2.1	Apakah sistem dapat membedakan hak akses user dalam mengelola data arsip?	0	0	5	0
2.2	Apakah sistem mendukung keamanan arsip dari pengubahan ilegal?	0	0	5	0
2.3	Apakah data arsip (informasi dan file) arsip tersimpan dengan baik?	0	1	4	0
3	***Compliance***				

3.1	Apakah pengelolaan arsip dalam sistem sudah sesuai dengan proses bisnis yang sedang berjalan pada instansi?	0	1	4	0
3.2	Apakah alur pengelolaan arsip pada sistem sama dengan alur pengelolaan arsip instansi?	0	1	4	0
4	***Comprehensiveness***				
4.1	Apakah semua data arsip hasil aktifitas bisnis organisasi bisa masuk ke dalam sistem?	0	0	4	1
4.2	Apakah arsip dokumen (informasi dan file) arsip tersimpan dengan baik?	0	1	3	1
4.3	Apakah metadata arsip membantu dalam proses penemuan kembali data arsip?	0	0	5	0
5	***Systematic***				
5.1	Apakah sistem mencatat riwayat pengubahan data arsip?	0	0	5	0
5.2	Apakah sistem mencatat riwayat pemusnahan arsip?	0	0	5	0
5.3	Apakah pencarian data arsip dapat dilakukan dengan mudah?	0	1	4	0
	Total	**0**	**6**	**62**	**2**

Dari hasil pengujian yang didapatkan dari perhitungan *quesioner* pada lampiran 1 didapatkan 0 jawaban "Sangat Tidak Sesuai", 6 jawaban "Tidak Sesuai", 62 jawaban "Sesuai", dan 2 jawaban "Sangat Sesuai". Dari hasil skor kuesioner tersebut, kemudian dicari nilai persentase tingkat kesesuaian tiap kategori pengujian, yaitu *reliability*, *integrity*, dan *compliance*, *comprehensiveness*, dan *systematic*.

Untuk mencari skor jawaban tiap kategori, dihitung total skor jawaban di kategori tersebut dikali skor *likert*. Kemudian di cari skor maksimum dengan mengalikan jumlah jawaban tiap pertanyaan dengan jumlah pertanyaan di tiap kategori. Hasil perhitungan skor total jawaban dikali dengan skor *likert* ditujukkan pada tabel 5.19

Tabel 5.18 Perhitungan Total Jawaban dikali Skor *Likert*.

No	Kategori	Jawaban				Total
		STS (1)	TS (2)	S (3)	SS (4)	
1.	*Reliability*	0 x 1	1 x 2	14 x 3	0 x 4	44
2.	*Integrity*	0 x 1	1 x 2	14 x 3	0 x 4	44
3.	*Compliance*	0 x 1	2 x 2	8 x 3	0 x 4	28
4.	*Comprehensiveness*	0 x 1	1 x 2	12 x 3	2 x 4	46

5.	*Systematic*	0 x 1	1 x 2	14 x 3	0 x 4	44
6.	**Keseluruhan**	0 x 1	6 x 2	62 x 3	2 x 4	206

Sedangkan nilai maksimum didapatkan dari skor tertinggi likert dikali jumlah responden pada tiap kategori. Hasil perhitungan persentase ditunjukkan pada tabel 5.19

Tabel 5.19 Hasil Perhitungan Tingkat Kesesuaian Sistem Terhadap Tiap Kategori Pengujian

No.	Kategori	Pernyataan	Total Skor x Skor Likert	Nilai Max	Hasil Skor
1.	*Reliability*	1.1 – 1.3	44	60	(44/60) x 100% = 73%
2.	*Integrity*	2.1 – 2.3	44	60	(44/60) x 100% = 73%
3.	*Compliance*	3.1 – 3.2	28	40	(28/40) x 100% = 70%
4.	*Comprehensiveness*	4.1 – 4.3	46	60	(46/60) x 100% = 76%
5.	*Systematic*	5.1 – 5.3	44	60	(44/60) x 100% = 73%
6.	Keseluruhan	1.1 – 4.3	206	280	(175/240) x 100% = 73.57%

Untuk menilai hasil tingkat kesesuaian terlebih dahulu ditentukan interval skor interpretasi persen yaitu dengan rumus interval (I) = 100 / jumlah skor likert. Maka interval sistem ini adalah (I) = 100/4 = 25, dan interpretasi skor berdasarkan interval adalah :

- Angka 0% - 24.99% = Sangat Tidak Sesuai
- Angka 25% - 49.99% = Tidak Sesuai
- Angka 50% - 74.99% = Sesuai
- Angka 75% - 100% = Sangat Sesuai

Selanjutnya dihitung hasil dengan membagi total skor dengan skor maksimum, kemudian dikali 100%. Hasil = (206/280) x 100%, maka didapat hasil akhir 73.57%. Berdasarkan interval, hasil tersebut masuk dalam kategori “Sesuai” dengan standar sistem pencatatan ISO 15489.

5.6 Analisis

Proses analisis bertujuan untuk mendapatkan kesimpulan dari hasil pengujian Sistem Manajemen Arsip Elektronik Yang Mengacu Pada Standar sistem pencatatan ISO 15489. Proses analisis yang dilakukan meliputi analisis pengujian unit, analisis pengujian validasi dan analisis pengujian sistem terhadap kesesuian sistem pencatatan ISO 15489.

5.6.1. Analisis Hasil Pengujian Unit

Proses analisis terhadap pengujian unit dilakukan dengan melihat kesesuain fungsi dari implementasi unit modul yang diuji dengan hasil perancangan perangkat lunak yang telah dirancang sebelumnya. Dari pengujian yang dilakukan, jumlah jalur pada *method* yang digambarkan dalam *flowgraph* telah sesuai dengan perhitungan rumus *cyclomatic complexity*. Setiap kasus uji yang dibuat berdasarkan jumlah jalur tersebut telah diuji dan memberikan hasil sesuai dengan hasil yang diharapkan. Berdasarkan hal tersebut, dapat diambil kesimpulan bahwa unit modul dari program sudah memenuhi kebutuhan fungsional.

5.6.2. Analisis Hasil Pengujian Validasi

Proses analisis terhadap hasil pengujian validasi dilakukan dengan melihat kesesuaian antara hasil kinerja sistem dengan daftar kebutuhan. Berdasarkan hasil pengujian validasi dapat disimpulkan bahwa implementasi dan fungsionalitas Sistem Manajemen Arsip Elektronik Yang Mengacu Pada Standar sistem pencatatan ISO 15489 telah memenuhi kebutuhan yang telah dijabarkan pada tahap analisis kebutuhan serta memperoleh hasil 95% valid.

5.6.3. Analisis Hasil Pengujian Sistem Terhadap Kesesuaian ISO 15489

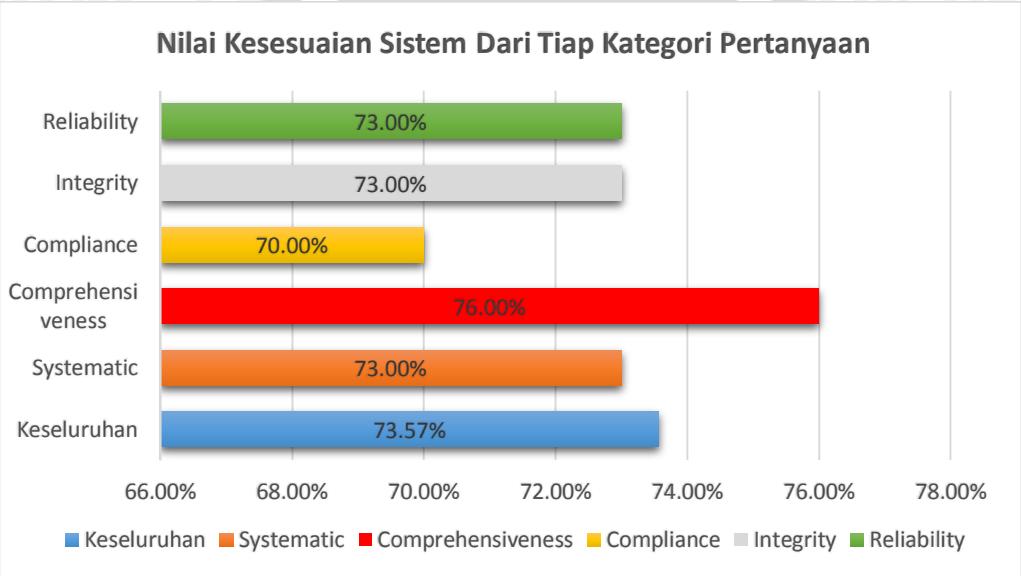
1. Proses analisis penilaian dilakukan dengan melihat pengimplementasian proses manajemen kearsipan yang sedang berjalan saat ini pada UPT Laboratorium Kesehatan Hewan Malang terhadap butir 9 mengenai *Records*

Management Processes ISO 15489. Kemudian didapatkan laporan seperti pada tabel 5.20

Tabel 5.20 Laporan Hasil *Checklist* Kesesuaian ISO 15489

Judul:	Kesesuaian Manajemen Kearsipan Audit: Pengarsipan, Pemusnahan, dan Penghancuran Arsip
Lokasi:	UPT Laboratorium Kesehatan Hewan Malang
Distribusi Laporan:	Kepala UPT Laboratorium Kesehatan Hewan Malang
Tujuan:	Untuk memastikan bahwa pengarsipan, pemusnahan, dan pengahancuran arsip yang terpercaya dan dilakukan sesuai dengan prosedur yang ditetapkan dan sesuai dengan kode etik manajemen arsip UPT Laboratorium Kesehatan Hewan Malang.
Ruang Lingkup & Metodologi Audit:	Audit ini dilakukan dengan menggunakan *checklist* praktek yang sesuai standar dan melalui pemeriksanaan, yang didasarkan pada basis tes dalam ruang lingkup kearsipan. Jenis arsip yang diperiksa misalnya: a. Daftar Pemusnahan Arsip b. Formulir Standar
Pemilihan *Sample*:	100% dari arsip yang dihasilkan dari setiap jenis proses bisnis
Kesimpulan yang didapat:	UPT Laboratorium Kesehatan Hewan Malang belum memiliki prosedur yang resmi untuk pengarsipan, penyusutan, dan pemusnahan arsip yang sesuai dengan standar praktek yang baik. Oleh karena itu diperlukan tindak lanjut agar pengarsipan di UPT Laboratorium Kesehatan Hewan Malang sesuai dengan standar pengelolaan arsip.

2. Proses analisis penilaian dilakukan dengan melihat hasil kinerja sistem terhadap kesesuaian dari karakteristik standar sistem pencatatan ISO 15489 yang telah dijadikan variabel pengujian yaitu kehandalan (*reliability*), keutuhan (*integrity*), kesesuaian (*compliance*), kelengkapan(*comprehensiveness*), dan sistematis (*systematic*) dengan hasil perhitungan yang dapat dilihat pada gambar grafik 5.33



Gambar 5.33 Grafik Nilai Kesesuaian Sistem Dari Tiap Kategori Pertanyaan

Pada hasil perhitungan tingkat kesesuaian secara keseluruhan diperoleh hasil sebesar 73.57%. Hasil ini menunjukkan bahwa Sistem Manajemen Arsip Elektronik secara keseluruhan sesuai dengan standar sistem pencatatan ISO 15489.

BAB VI
PENUTUP

6.1. Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian Sistem Manajemen Arsip Elektronik Yang Mengacu Pada Standar sistem pencatatan ISO 15489 dapat diambil kesimpulan :

1. Implementasi sistem ini telah sesuai dengan spesifikasi perancangan pada bab IV. Hal ini terlihat dari hasil pengujian unit dengan metode *white box* dan *black box*. Pengujian *white box* memperoleh hasil bahwa unit modul dari program sudah memenuhi kebutuhan fungsional. Dan pengujian *black box* dengan metode pengujian validasi memperoleh hasil 95% valid.
2. Hasil pengujian kesesuaian sistem terhadap karakteristik standar sistem pencatatanISO 15489 pada kategori *reliability* sebesar 73%, kategori *integrity* sebesar 73%, *compliance* sebesar 70%, *comprehensiveness* sebesar 76%, dan *systematic* sebesar 73%. Sedangkan tingkat kesesuaian sistem secara keseluruhan sebesar 73.57%, dimana hasil tersebut masuk pada interval “Sesuai”. Hal ini menunjukkan bahwa sistem sudah sesuai dengan karakteristik standar sistem pencatatan ISO 15489 berdasarkan pendapat pengguna.

6.2. Saran

Saran yang dapat diberikan untuk mengembangkan Sistem Manajemen Arsip Elektronik lebih lanjut antara lain:

1. Menambah fitur pencarian berdasarkan isi konten dari file PDF yang di *upload*.
2. Menambahkan fasilitas *scan* data arsip secara langsung dalam sistem.

DAFTAR PUSTAKA

[1] Alamsyah, Nandang Deliarnoor. 2008. *Rekod Elektonik dan Hukum*. Disampaikan dalam Seminar Internasional INDONESIA-MALAYSIA.

[2] An Approach to Records Management Audit. 2007. (Online). http://systems.hscic.gov.uk/infogov/records/it04a.pdf, (diakses tanggal 3 Juli 2015).

[3] Anggoro, Dani, dan Muhammad Lutfi Aksani. 2015. *Dashboard Information System Sebagai Pendukung Keputusan Dalam Penjualan Tiket Pesawat studi Kasus : PT Nurindo Tour.* Jurnal Sistem Informasi, Vol 5 No. 3, Maret 2015

[4] Barthos, Basir. 2003. *Manajemen kearsipan : untuk lembaga negara, swasta dan perguruan tinggi / oleh Basir Barthos*. Bumi Aksara : Jakarta.

[5] Connolly, Thomas, and Carolyn Begg. 2010. *Database Systems: A Practical Approach to Design, Implementation, and Management, Fifth Edition*. Boston : Pearson Education.

[6] Hendraputra, Ade., et all. 2009. *Jaminan Mutu Sistem Informasi*. Bandung: Politeknik Telkom.

[7] ISO 15489-1. 2001. *Information and documentation-Records management-Part 1: General*. Switzerland: Geneva 20.

[8] ISO 15489-2. 2001. *Information and documentation-Records management-Part 2: Guidelines*. Switzerland: Geneva 20.

[9] Kustinawati, Sri. 2010. *Peranan Manajemen Arsip Dinamis*. Arsiparis Pelaksana Lanjutan, Universitas Sriwijaya.

[10] Pressman, Roger S. 2001. *Software Engineering : A Practitioner Approach*. McGraw-Hill Companies.

[11] Pressman, Roger S. 2012. *Rekayasa Perangkat Lunak(Pendekatan Praktisi) Edisi 7 : Buku 1*.Yogyakarta : Andi Offset.

[12] Rahmadiansyah, Dudi dan Dedy Irwan. 2012. *Implementasi Metode Model View Controller Menggunakan Framework Code Igniter dalam Pengembangan Aplikasi Manajemen Depo Petikemas pada Unit Usaha Belawan Logistic Center*. (Online), ISBN 978-602-19873-0-6, (http://prosiding-snastikomti.stth-medan.ac.id/index.php/doc_download/16-implementasi-metode-model-view-controller-menggunakan-framework-code-igniter-dalam-pengembangan-aplikasi-manajemen-depo-petikemas-pada-unit-usaha-belawan-logistic-center, (diakses tanggal 05 November 2014).

[13] Rosa, AS dan Salahuddin, M. 2013. *Rekayasa Perangkat Lunak*. Bandung: Informatika.

[14] Simmarmata, Janner. 2010. *Rekayasa Perangkat Lunak*. Yogyakarta: Andi Offset.

[15] Sommerville, Ian. 2011. *Software Engineering, Ninth Edition*. New York : Addison-Wesley.

[16] Sueno, Bernadette D. 2009. *UP Docutrak : An Assessment of an in-house developed document tracking system(DTS)*. *Journal of Phillippine Librarianship*

[17] Unit Pelaksana Teknis Laboratorium Kesehatan Hewan Malang. 2014. Tupoksi(Online)

http://disnak.jatimprov.go.id/web/uptlab/labkesehatanhewanmalang.
(diakses tanggal 02 November 2014)

[18] Wibisono, Agus. 2011. *Apa itu Standarisasi Internasional (ISO)* (Online) http://aguswibisono.com/2011/apa-itu-standarisasi-internasional-iso-sni/. (diakses tanggal 2 November 2014).

[19] Wursanto. 1991. *Kearsipan I*. Yogyakarta: Kanisisus

LAMPIRAN

L1-1 Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik Terhadap Standar Sistem Pencatatan ISO 15489

Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik Terhadap Standar Sistem Pencatatan ISO 15489

Nama : Arik Hermanto
Jabatan : Persuratan

Pilihan Jawaban :

STS : Sangat Tidak Setuju **TS : Tidak Setuju**
S : Setuju **SS : Sangat Setuju**

NO	PERTANYAAN	JUMLAH			
		STS	TS	S	SS
1	***Reliability***				
1.1	Apakah aplikasi dapat mencatat semua data arsip?			✓	
1.2	Apakah aplikasi dapat menampilkan data arsip sewaktu-waktu saat arsip dibutuhkan?			✓	
1.3	Apakah metadata arsip yang ditampilkan dapat memberikan informasi dari arsip yang dibutuhkan?			✓	
2	***Integrity***				
2.1	Apakah sistem dapat membedakan hak akses user dalam mengelola data arsip?			✓	
2.2	Apakah sistem mendukung keamanan arsip dari pengubahan ilegal?			✓	
2.3	Apakah data arsip (informasi dan file) arsip tersimpan dengan baik?			✓	
3	***Compliance***				
3.1	Apakah pengelolaan arsip dalam sistem sudah sesuai dengan proses bisnis yang sedang berjalan pada instansi?			✓	
3.2	Apakah alur pengelolaan arsip pada sistem sama dengan alur pengelolaan arsip instansi?			✓	
4	***Comprehensiveness***				
4.1	Apakah semua data arsip hasil aktifitas bisnis organisasi bisa masuk ke dalam sistem?				✓
4.2	Apakah arsip dokumen (informasi dan file) arsip tersimpan dengan baik?				✓
4.3	Apakah metadata arsip membantu dalam proses penemuan kembali data arsip?			✓	
5	***Systematic***				
5.1	Apakah sistem mencatat riwayat pengubahan data arsip?			✓	
5.2	Apakah sistem mencatat riwayat pemusnahan arsip?			✓	
5.3	Apakah pencarian data arsip dapat dilakukan dengan mudah?			✓	

L1-2 Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik Terhadap Standar Sistem Pencatatan ISO 15489

Nama : Winarti, SP
Jabatan : Kasubag TU

Pilihan Jawaban :

STS : Sangat Tidak Setuju **TS : Tidak Setuju**

S : Setuju **SS : Sangat Setuju**

NO	PERTANYAAN	JUMLAH			
		STS	TS	S	SS
1	***Reliability***				
1.1	Apakah aplikasi dapat mencatat semua data arsip?			✓	
1.2	Apakah aplikasi dapat menampilkan data arsip sewaktu-waktu saat arsip dibutuhkan?		✓		
1.3	Apakah metadata arsip yang ditampilkan dapat memberikan informasi dari arsip yang dibutuhkan?			✓	
2	***Integrity***				
2.1	Apakah sistem dapat membedakan hak akses user dalam mengelola data arsip?			✓	
2.2	Apakah sistem mendukung keamanan arsip dari pengubahan ilegal?			✓	
2.3	Apakah data arsip (informasi dan file) arsip tersimpan dengan baik?		✓		
3	***Compliance***				
3.1	Apakah pengelolaan arsip dalam sistem sudah sesuai dengan proses bisnis yang sedang berjalan pada instansi?		✓		
3.2	Apakah alur pengelolaan arsip pada sistem sama dengan alur pengelolaan arsip instansi?		✓		
4	***Comprehensiveness***				
4.1	Apakah semua data arsip hasil aktifitas bisnis organisasi bisa masuk ke dalam sistem?			✓	
4.2	Apakah arsip dokumen (informasi dan file) arsip tersimpan dengan baik?		✓		
4.3	Apakah metadata arsip membantu dalam proses penemuan kembali data arsip?			✓	
5	***Systematic***				
5.1	Apakah sistem mencatat riwayat pengubahan data arsip?			✓	
5.2	Apakah sistem mencatat riwayat pemusnahan arsip?			✓	
5.3	Apakah pencarian data arsip dapat dilakukan dengan mudah?		✓		

L1-3 Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik Terhadap Standar Sistem Pencatatan ISO 15489

Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik Terhadap Standar Sistem Pencatatan ISO 15489

Nama : Drh. Sri Pangastuti

Jabatan : Kepala Bagian Kesmavet

Pilihan Jawaban :

STS : Sangat Tidak Setuju **TS : Tidak Setuju**

S : Setuju **SS : Sangat Setuju**

NO	PERTANYAAN	JUMLAH			
		STS	TS	S	SS
1	***Reliability***				
1.1	Apakah aplikasi dapat mencatat semua data arsip?			✓	
1.2	Apakah aplikasi dapat menampilkan data arsip sewaktu-waktu saat arsip dibutuhkan?			✓	
1.3	Apakah metadata arsip yang ditampilkan dapat memberikan informasi dari arsip yang dibutuhkan?			✓	
2	***Integrity***				
2.1	Apakah sistem dapat membedakan hak akses user dalam mengelola data arsip?			✓	
2.2	Apakah sistem mendukung keamanan arsip dari pengubahan ilegal?			✓	
2.3	Apakah data arsip (informasi dan file) arsip tersimpan dengan baik?			✓	
3	***Compliance***				
3.1	Apakah pengelolaan arsip dalam sistem sudah sesuai dengan proses bisnis yang sedang berjalan pada instansi?			✓	
3.2	Apakah alur pengelolaan arsip pada sistem sama dengan alur pengelolaan arsip instansi?			✓	
4	***Comprehensiveness***				
4.1	Apakah semua data arsip hasil aktifitas bisnis organisasi bisa masuk ke dalam sistem?			✓	
4.2	Apakah arsip dokumen (informasi dan file) arsip tersimpan dengan baik?			✓	
4.3	Apakah metadata arsip membantu dalam proses penemuan kembali data arsip?			✓	
5	***Systematic***				
5.1	Apakah sistem mencatat riwayat pengubahan data arsip?			✓	
5.2	Apakah sistem mencatat riwayat pemusnahan arsip?			✓	
5.3	Apakah pencarian data arsip dapat dilakukan dengan mudah?			✓	

L1-4 Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik Terhadap Standar Sistem Pencatatan ISO 15489

Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik Terhadap Standar Sistem Pencatatan ISO 15489

Nama : Vinda Mustika

Jabatan : Administratif

Pilihan Jawaban :

STS : Sangat Tidak Setuju **TS : Tidak Setuju**

S : Setuju **SS : Sangat Setuju**

NO	PERTANYAAN	JUMLAH			
		STS	TS	S	SS
1	***Reliability***				
1.1	Apakah aplikasi dapat mencatat semua data arsip?			✓	
1.2	Apakah aplikasi dapat menampilkan data arsip sewaktu-waktu saat arsip dibutuhkan?			✓	
1.3	Apakah metadata arsip yang ditampilkan dapat memberikan informasi dari arsip yang dibutuhkan?			✓	
2	***Integrity***				
2.1	Apakah sistem dapat membedakan hak akses user dalam mengelola data arsip?			✓	
2.2	Apakah sistem mendukung keamanan arsip dari pengubahan ilegal?			✓	
2.3	Apakah data arsip (informasi dan file) arsip tersimpan dengan baik?			✓	
3	***Compliance***				
3.1	Apakah pengelolaan arsip dalam sistem sudah sesuai dengan proses bisnis yang sedang berjalan pada instansi?			✓	
3.2	Apakah alur pengelolaan arsip pada sistem sama dengan alur pengelolaan arsip instansi?			✓	
4	***Comprehensiveness***				
4.1	Apakah semua data arsip hasil aktifitas bisnis organisasi bisa masuk ke dalam sistem?			✓	
4.2	Apakah arsip dokumen (informasi dan file) arsip tersimpan dengan baik?			✓	
4.3	Apakah metadata arsip membantu dalam proses penemuan kembali data arsip?			✓	
5	***Systematic***				
5.1	Apakah sistem mencatat riwayat pengubahan data arsip?			✓	
5.2	Apakah sistem mencatat riwayat pemusnahan arsip?			✓	
5.3	Apakah pencarian data arsip dapat dilakukan dengan mudah?			✓	

L1-5 Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik Terhadap Standar Sistem Pencatatan ISO 15489

Pengujian Kesesuaian Sistem Manajemen Arsip Elektronik Terhadap Standar Sistem Pencatatan ISO 15489

Nama : Drh. Novia Rahasti M

Jabatan : Kabag Keswan

Pilihan Jawaban :

STS : Sangat Tidak Setuju **TS : Tidak Setuju**

S : Setuju **SS : Sangat Setuju**

NO	PERTANYAAN	JUMLAH			
		STS	TS	S	SS
1	***Reliability***				
1.1	Apakah aplikasi dapat mencatat semua data arsip?			✓	
1.2	Apakah aplikasi dapat menampilkan data arsip sewaktu-waktu saat arsip dibutuhkan?			✓	
1.3	Apakah metadata arsip yang ditampilkan dapat memberikan informasi dari arsip yang dibutuhkan?			✓	
2	***Integrity***				
2.1	Apakah sistem dapat membedakan hak akses user dalam mengelola data arsip?			✓	
2.2	Apakah sistem mendukung keamanan arsip dari pengubahan ilegal?			✓	
2.3	Apakah data arsip (informasi dan file) arsip tersimpan dengan baik?			✓	
3	***Compliance***				
3.1	Apakah pengelolaan arsip dalam sistem sudah sesuai dengan proses bisnis yang sedang berjalan pada instansi?			✓	
3.2	Apakah alur pengelolaan arsip pada sistem sama dengan alur pengelolaan arsip instansi?			✓	
4	***Comprehensiveness***				
4.1	Apakah semua data arsip hasil aktifitas bisnis organisasi bisa masuk ke dalam sistem?			✓	
4.2	Apakah arsip dokumen (informasi dan file) arsip tersimpan dengan baik?			✓	
4.3	Apakah metadata arsip membantu dalam proses penemuan kembali data arsip?			✓	
5	***Systematic***				
5.1	Apakah sistem mencatat riwayat pengubahan data arsip?			✓	
5.2	Apakah sistem mencatat riwayat pemusnahan arsip?			✓	
5.3	Apakah pencarian data arsip dapat dilakukan dengan mudah?			✓	

repository.ub.ac.id

UNIVERSITAS BRAWIJAYA

repository.ub.ac.id

UNIVERSITAS BRAWIJAYA