

**RANCANG BANGUN DAN ANALISIS  
SISTEM INFORMASI MANAJEMEN LABORATORIUM**

**(STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG)**

**SKRIPSI**

**Laboratorium Rekayasa Perangkat Lunak**

**Diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer**



Disusun Oleh:

**DIAN ARIYA PUJI PRATAMA**

**NIM. 125150209111017**

**KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI**

**PROGRAM STUDI INFORMATIKA / ILMU KOMPUTER**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS BRAWIJAYA**

**MALANG**

**2015**

**LEMBAR PERSETUJUAN**  
**RANCANG BANGUN DAN ANALISIS**  
**SISTEM INFORMASI MANAJEMEN LABORATORIUM**  
**(STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG)**

**SKRIPSI**

**Laboratorium Rekayasa Perangkat Lunak**

**Diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer**



**Disusun Oleh:**

**DIAN ARIYA PUJI PRATAMA**  
**NIM. 125150209111017**

**Dosen Pembimbing I**

**Dosen Pembimbing II**

**Denny Sagita R., S.Kom, M.Kom**  
**NIK. 851124 06 1 1 0250**

**Fajar Pradana, S.ST., M.Eng**  
**NIK. 871121 161 1 0371**

**LEMBAR PENGESAHAN**  
**RANCANG BANGUN DAN ANALISIS**  
**SISTEM INFORMASI MANAJEMEN LABORATORIUM**  
**(STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG)**

**SKRIPSI**

**Diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer**

**Disusun Oleh:**

**DIAN ARIYA PUJI PRATAMA**

**NIM. 125150209111017**

**Skripsi ini telah diuji dan dinyatakan lulus pada**

**Tanggal 26 Juni 2015**

**Penguji I**

**Penguji II**

**Issa Arwani, S.Kom.,M.Sc.**  
**NIP. 19830922 201212 1 003**

**Yusi Tyroni Mursitvo, S.Kom.,MS**  
**NIP. 19800228 200604 1 001**

**Penguji III**

**Agi Putra Kharisma, S.T., M.T.**

**NIK. 201304 860430 1 001**

**Mengetahui,**

**Ketua Program Studi Informatika / Ilmu Komputer**

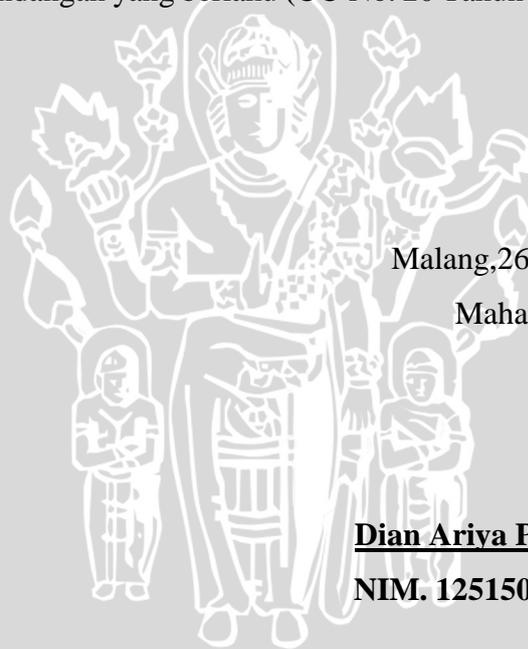
**Drs. Marji, M.T.**

**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 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 gelar akademik yang telah saya peroleh (SARJANA) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70)



Malang, 26 Juni 2015

Mahasiswa

**Dian Ariya Puji Pratama**

**NIM. 125150209111017**

## KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah SWT yang telah memberikan rahmat dan hidayahnya dengan member kekuatan dan kemudahan kepada penulis dapat sehingga dapat menyelesaikan skripsi ini dengan judul “Rancang Bangun Dan Analisis Sistem Informasi Manajemen Laboratorium Studi Kasus UPT Laboratorium Kesehatan Hewan”, sebagai salah satu syarat kelulusan pada Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya Malang.

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 menghaturkan ucapan terima kasih yang sebesar-besarnya kepada :

- Bapak Drs. Marji, MT. dan Bapak Issa Arwani, ST., MT., selaku Ketua dan Sekretaris Program Studi Informatika
- Bapak Denny Sagita R., S.Kom, M.Kom dan Fajar Pradana, S.ST., M.Eng selaku pembimbing I dan II dalam penyusunan skripsi ini.
- Mashadi dan Bu Diah novia selaku pembimbing di UPT Laboratorium Kesehatan Hewan Malang.
- Kedua Orang Tua, khususnya Ibunda 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.
- Dian Rifia Syaifudin, Erisza Puspitasari, Galang Yuniar Mahendra dan Pratiwi Endah Permata yang telah turut membantu dalam penyelesaian skripsi ini.
- Delvia Anggraini Puspitasari yang selama ini telah memberikan dukungan semangat dan kesuksesan penulis.
- Sahabat-sahabat SAP 2012 yang banyak memberikan Informasi dan bantuan dalam penyusunan skripsi ini.
- Sahabat G.Lowo yang selama ini telah memberikan dukungan semangat dan bantuan selama kuliah di Universitas Brawijaya.

- Serta semua pihak yang tidak dapat disebutkan satu persatu yang telah turut membantu baik secara langsung maupun tidak langsung dalam penyelesaian Skripsi ini.

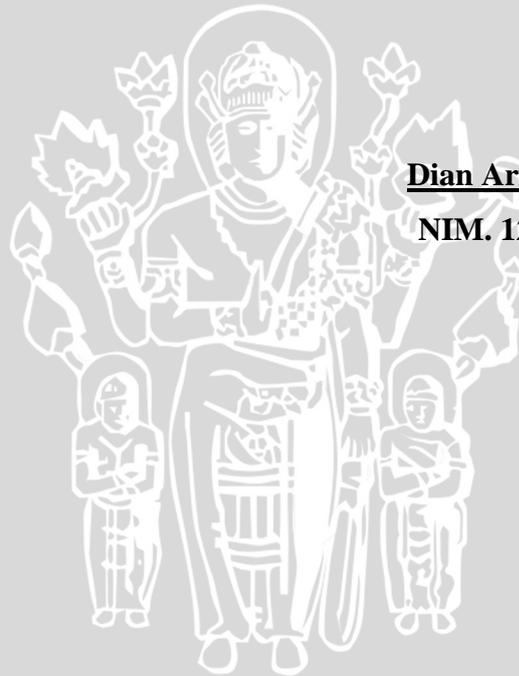
Penulis menyadari bahwa dalam skripsi ini masih banyak terdapat kekurangan, untuk itu penyusun menerima kritik dan saran yang bersifat membangun, sehingga skripsi ini bisa menjadi lebih baik. Akhir kata penulis mengucapkan terima kasih, dan semoga skripsi ini dapat bermanfaat bagi kita semua, khususnya pecinta IT.

Malang, 26 Juni 2015

**Mahasiswa**

**Dian Ariya Puji Pratama**

**NIM. 125150209111017**



## ABSTRAK

**Dian Ariya Puji Pratama. 2015. Rancang Bangun Dan Analisis Sistem Informasi Manajemen Laboratorium (Studi Kasus UPT Laboratorium Kesehatan Hewan Malang) . Fakultas Ilmu Komputer, Universitas Brawijaya, Malang.**

**Dosen Pembimbing: Bapak Denny Sagita R., S.Kom, M.Kom dan Bapak Fajar Pradana, S.ST.,M.Eng**

Dalam rangka meningkatkan kualitas pelayanan, UPT Laboratorium Kesehatan Hewan Malang mengacu pada Standar Operasional Prosedur (SOP) pelayanan berupa Standar Pelayanan Publik (SPP). 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. Salah satu tujuan UPT Laboratorium Kesehatan Hewan Malang adalah menjamin ketersediaan produksi hasil ternak tapi juga menjamin standar mutu produk hasil ternak yang aman, sehat, utuh dan halal. Proses Pelayanan di UPT Laboratorium Kesehatan Hewan Malang saat ini masih belum terkomputerisasi secara keseluruhan. Beberapa diantaranya masih menggunakan banyak berkas, mulai dari mengisi formulir pendaftaran pelayanan yang akan melakukan pengujian Sampel hingga dokumentasi proses pelayanan pengujian selesai. Dari proses ini akan menjadi penghambat proses pelayanan dalam melakukan pendataan apabila berkas formulir hilang atau tumpang tindih. Ditambah lagi dengan sulitnya pencarian data, pihak Administratif pelayanan harus membuka dokumen satu persatu untuk menemukan data yang diinginkan.

Oleh karena itu, dibuatlah aplikasi Sistem Informasi Laboratorium (SIMLAB) yang dapat memajemen proses pelayanan pengujian Sampel di UPT Laboratorium Kesehatan Malang. Dalam pengembangan SIMLAB ini menggunakan *Systems development life cycle* (SDLC) permodelan *Prototype*. Sistem ini menangani pendaftaran pengujian sampel, proses kaji ulang sampel, permintaan pengujian sampel, diagnosa sampel, rekam diagnosa hingga hasil cetak laporan pelayanan yang menangani pengujian sampel. Sistem ini merupakan sistem Informasi berbasis web dan dibuat menggunakan bahasa pemrograman PHP dengan *Framework* Laravel 4.2.

Berdasarkan hasil pengujian, proses pelayanan mulai dari pendaftaran pelayanan hingga pengolahan data rekam diagnosa Sampel menjadi lebih cepat dan mudah dilakukan. Semua pihak yang terlibat di dalam proses pelayanan tidak perlu lagi menggunakan banyak kertas. Pencarian data pelayanan juga menjadi lebih mudah dan cepat. Dapat disimpulkan bahwa sistem ini secara fungsional berfungsi 100% valid dan secara keseluruhan berjalan dengan baik. Serta hasil pengujian *User Acceptance Testing* (UAT) yang menunjukkan bahwa tingkat kemudahan penggunaan 76,58% yang berarti pengguna sangat setuju terhadap kemudahan penggunaan aplikasi.

**Kata kunci:** *Pelayanan Kesehatan Hewan, SIMLAB, SDLC, Prototype, UAT*

## ABSTRACT

**Dian Ariya Puji Pratama. 2015. *Design And Analysis Of Laboratory Management Information System (Case Studies in UPT Laboratorium Kesehatan Malang). Faculty of Computer Science, University of Brawijaya, Malang.***

**Bachelor thesis advisor: Mr. Denny Sagita R., S.Kom, M.Kom and Mr Fajar Pradana, S.ST,.M.Eng**

*In order to improving the quality, UPT Laboratorium Kesehatan Hewan Malang refers to Standard Operating Procedures (SOP) Standar Pelayanan Publik (SPP). UPT Laboratorium Kesehatan Hewan Malang is a government institutions that have responsibilities in checking, investigating and animals diagnosis and service quality testing food product from animals, and product quality non-PAH. One of the goals UPT Laboratorium Kesehatan Hewan Malang is guaranteed the availability of livestock production but also to ensure quality standard livestock products are safe, healthy , intact and halal . That process of service from UPT Laboratorium Kesehatan Hewan Malang is still not computerized completely. Some of them are still using many sheaf fill in the registration form service that will test Sampels to test service documentation of the process is completed. From this process, it will be resistance that process of service in data collection if a lot of file are missing or overlap. In addition, it is difficult to search data, the administration of service must check documents one by one to find data to be desired.*

*Therefore, Laboratory Management Information System (SIMLAB) is created that can manage the process service of sampels testing in UPT Laboratorium Kesehatan Hewan Malang. In the process of development SIMLAB used Systems development life cycle (SDLC) Prototype modeling . This System will handles registration sampels testing, the review process sampels, test request Sampels, diagnosis sampels , track record result diagnosis until the printout report services that handle testing Sampels. This system is a web-based information systems and created using PHP programming language with Laravel 4.2 Framework.*

*Based on test result, the service from registration data processing services to track record diagnosis Sampels become faster and easier. All parties that involved in the process services are no longer need to use many paper. Data search services has also become more easily and quickly. It can be concluded that the system is Functional works 100% valid and overall is doing well. As well as the result of User Acceptance Testing (UAT) testing which showed that the level for easily of user 76.68 % strongly agreed means users are ease to use application.*

**Key words :** *Animals Healty Service, SIMLAB, SDLC, Prototype, UAT*

**DAFTAR ISI**

<b>LEMBAR PERSETUJUAN .....</b>	<b>i</b>
<b>LEMBAR PENGESAHAN .....</b>	<b>ii</b>
<b>PERNYATAAN .....</b>	<b>iii</b>
<b>ORISINALITAS SKRIPSI .....</b>	<b>iii</b>
<b>KATA PENGANTAR .....</b>	<b>iv</b>
<b>ABSTRAK .....</b>	<b>vi</b>
<b>ABSTRACT .....</b>	<b>vii</b>
<b>DAFTAR ISI .....</b>	<b>viii</b>
<b>DAFTAR GAMBAR .....</b>	<b>xii</b>
<b>DAFTAR TABEL .....</b>	<b>xiv</b>
<b>DAFTAR LAMPIRAN .....</b>	<b>xvii</b>
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah .....	3
1.4 Tujuan .....	3
1.5 Manfaat .....	4
1.6 Sistematika Penulisan .....	4
<b>BAB II KAJIAN PUSTAKA DAN DASAR TEORI .....</b>	<b>6</b>
2.1 UPT Laboratorium Kesehatan Hewan Malang .....	6
2.1.1 Tugas UPT Laboratorium Kesehatan Hewan .....	6
2.2 Pengertian Laboratorium .....	7
2.3 Pengertian Sistem .....	7
2.4 Pengertian Informasi .....	7
2.5 Pengertian Sistem Informasi .....	7
2.6 Sistem Infomasi Manajemen .....	8
2.7 Sistem Manajemen Laboratorium .....	8
2.7.1 Alur Sistem Pelayanan di Kantor UPT Laboratorium Kesehatan Hewan Malang .....	9
2.8 Perangkat Lunak .....	12
2.8.1 Definisi Perangkat Lunak .....	12



2.8.2	Karakteristik Perangkat Lunak.....	12
2.9	Rekayasa Perangkat Lunak.....	13
2.9.1	Fase-Fase Rekayasa Perangkat Lunak .....	14
2.10	Model Proses Prototype ( <i>Prototyping</i> ).....	14
2.10.1.	Kelebihan Metode Prototype.....	15
2.10.2.	Kekurangan Metode Prototype.....	16
2.10.3.	Jenis Prototype. ....	16
2.11	<i>Unified Modelling Language (UML)</i> .....	17
2.11.1	<i>Usecase Diagram</i> .....	17
2.11.2	<i>Activity Diagram</i> .....	18
2.11.3	<i>Sequence Diagram</i> .....	19
2.11.4	<i>Class Diagram</i> .....	21
2.12	PHP.....	24
2.13.1.	Keunggulan PHP .....	25
2.13	MVC ( <i>Model View Controller</i> ).....	25
2.14	Framework Laravel 4.2.....	27
2.14.1.	Sejarah Framework Laravel .....	27
2.14.2.	Kelebihan Framework Laravel 4.2.....	27
2.14.3.	Fitur Framework Laravel 4.2 .....	28
2.15	Database .....	29
2.15.1.	<i>Database Management System (DBMS)</i> .....	29
2.16	Teori Normalisasi .....	30
2.17	MySQL.....	31
2.18	Pengujian Perangkat Lunak.....	32
2.18.1	<i>Functional Testing</i> .....	32
2.18.2	<i>User Acceptance Testing (UAT)</i> .....	33
<b>BAB III METODOLOGI PENELITIAN.....</b>		<b>35</b>
3.1	Identifikasi Permasalahan.....	36
3.2	Identifikasi Solusi.....	36
3.2.1	Analisis dan Pendefinisian Kebutuhan .....	38
3.2.2	Perancangan Sistem .....	39
3.2.3	Pemodelan Sistem.....	40



3.2.4	<i>Evaluasi dan Feedback</i> .....	41
3.2.5	<i>Review dan Updation</i> .....	41
3.2.6	<i>Development</i> .....	42
3.2.7	Pengujian Unit Testing.....	42
3.3	Realisasi Solusi.....	42
3.4	Analisa Hasil Realisasi Solusi .....	43
<b>BAB IV ANALISIS DAN PERANCANGAN</b> .....		44
4.1	Gambaran Umum .....	44
4.2	<i>Elisitasi</i> .....	44
4.3	Spesifikasi Sistem.....	45
4.3.1	Gambaran Sistem Lama .....	45
4.4	Spesifikasi Kebutuhan Sistem .....	46
4.4.1.	Indentifikasi Aktor .....	46
4.4.2.	Spesifikasi Kebutuhan Fungsional Sistem .....	47
4.4.3.	Spesifikasi Kebutuhan Non-Fungsional Sistem.....	53
4.5	Perancangan Prototype .....	53
4.5.1.	Perancangan Prototype Fungsional .....	54
4.5.2.	Perancangan Prototype Antarmuka.....	56
4.6	Perancangan Sistem.....	58
4.6.1.	Perancangan Usecase Diagram .....	58
4.6.2.	Skenario Usecase Diagram .....	59
4.6.3.	Perancangan Activity Diagram.....	72
4.5.3.	Perancangan Sequence Diagram.....	76
4.6.4.	Perancangan Class Diagram.....	78
4.6.5.	Perancangan Database SIMLAB.....	80
4.6.6.	Perancangan Antarmuka .....	98
<b>BAB V IMPLEMENTASI</b> .....		102
5.1	Spesifikasi Sistem.....	102
5.1.1.	Spesifikasi Perangkat Keras.....	102
5.1.2.	Spesifikasi Perangkat Lunak .....	102
5.2	Batasan-Batasan Implementasi.....	103
5.3	Implementasi Antarmuka .....	103



5.3.1. Implementasi Antarmuka Halaman Login .....	104
5.3.2. Implementasi Antarmuka Halaman Kelola Karyawan .....	104
5.3.3. Implementasi Antarmuka Halaman Kelola User .....	105
5.3.4. Implementasi Antarmuka Halaman Kelola data Hewan.....	105
5.3.5. Implementasi Antarmuka Halaman Pendaftaran Pelayanan .....	106
5.3.6. Implementasi Antarmuka Halaman Kelola Mitra .....	107
5.3.7. Implementasi Antarmuka Halaman Setting Email.....	107
5.3.8. Implementasi Antarmuka Halaman Penerimaan Sampel.....	108
5.3.9. Implementasi Antarmuka Halaman Kelola Diagnosa.....	110
5.3.10. Implementasi Antarmuka Halaman Cetak Laporan .....	112
<b>BAB VI PENGUJIAN DAN ANALISIS</b> .....	<b>113</b>
6.1 Pengujian .....	113
6.1.1. Pengujian <i>Functional Testing</i> .....	113
6.1.2. Hasil Pengujian <i>Functional Testing</i> .....	113
6.1.3. Pengujian <i>User Acceptance Testing (UAT)</i> .....	117
6.1.4. Hasil Pengujian <i>User Acceptance Testing (UAT)</i> .....	117
6.2 Analisis .....	120
6.2.1. Analisis Hasil <i>Functional Testing</i> .....	120
6.2.2. Analisis Hasil <i>User Acceptance Testing (UAT)</i> .....	120
<b>BAB VII PENUTUP</b> .....	<b>122</b>
7.1 Kesimpulan.....	122
7.2 Saran.....	122
<b>DAFTAR PUSTAKA</b> .....	<b>123</b>



## DAFTAR GAMBAR

Gambar 2.1 Gambar Alur Proses Pelayanan Pasif.....	9
Gambar 2.2 Gambar Alur Proses Pelayanan Semi Aktif.....	10
Gambar 2.3 Gambar Alur Proses Pelayanan Aktif.....	11
Gambar 2.4 Tahapan Proses Model Prototipe.....	15
Gambar 2.5 Gambar <i>Usecase</i> Diagram.....	18
Gambar 2.6 Gambar <i>Activity</i> Diagram.....	19
Gambar 2.7 Gambar <i>Sequence</i> Diagram.....	20
Gambar 2.8 Gambar Asosiasi Kelas Sederhana.....	22
Gambar 2.9 Gambar <i>Class</i> dan Asosiasi.....	22
Gambar 2.10 Gambar <i>Consultation Class</i> .....	23
Gambar 2.11 Gambar Konsep MVC.....	26
Gambar 3.1 Diagram Blok Metode Penelitian.....	35
Gambar 3.2 Diagram Blok Tahap Pembuatan Software.....	37
Gambar 4.1 <i>Usecase</i> Diagram SIMLAB.....	59
Gambar 4.2 <i>Activity</i> Diagram Login.....	72
Gambar 4.3 <i>Activity</i> Diagram Kelola Karyawan.....	73
Gambar 4.4 <i>Activity</i> Diagram Kelola Wilayah.....	74
Gambar 4.5 <i>Activity</i> Diagram Pendaftaran Pelayanan.....	75
Gambar 4.6 <i>Activity</i> Diagram Validasi Penerimaan Sampel.....	75
Gambar 4.7 <i>Sequence</i> Diagram Login.....	76
Gambar 4.8 <i>Sequence</i> Diagram Kelola Karyawan.....	76
Gambar 4.9 <i>Sequence</i> Diagram Kelola Wilayah (Provinsi).....	77
Gambar 4.10 <i>Sequence</i> Diagram Pendaftaran Pelayanan.....	77
Gambar 4.11 <i>Sequence</i> Diagram Validasi Pelayanan.....	78
Gambar 4.12 <i>Class</i> Diagram SIMLAB.....	79
Gambar 4.13 Perancangan ERD SILAB.....	81
Gambar 4.14 Perancangan Database SILAB.....	82
Gambar 4.45 Rancangan Antarmuka Halaman Login.....	99
Gambar 4.46 Rancangan Antarmuka Halaman Beranda.....	99

Gambar 4.46 Perancangan Antarmuka Halaman Tabel Data .....	100
Gambar 4.46 Perancangan Antarmuka Halaman Form Input dan Edit .....	100
Gambar 4.46 Perancangan Antarmuka Halaman Form Cetak Laporan.....	101
Gambar 5.1 Implementasi Antarmuka Halaman Login .....	104
Gambar 5.2 Implementasi Antarmuka Halaman Kelola Karyawan.....	104
Gambar 5.3 Implementasi Antarmuka Halaman Kelola User .....	105
Gambar 5.3 Implementasi Antarmuka Halaman Kelola User .....	105
Gambar 5.4 Implementasi Antarmuka Halaman Registrasi Pelayanan .....	106
Gambar 5.5 Implementasi Antarmuka Halaman Registrasi Pelayanan .....	106
(Detail Pelayanan).....	106
Gambar 5.6 Implementasi Antarmuka Halaman Kelola Mitra .....	107
(Member Personal).....	107
Gambar 5.7 Implementasi Antarmuka Halaman Setting Email.....	107
Gambar 5.8 Implementasi Antarmuka Halaman Kelola Penerimaan Sampel ...	108
Gambar 5.9 Implementasi Antarmuka Halaman Form Penerimaan Sampel .....	108
Gambar 5.10 Implementasi Antarmuka Halaman Kelola Penerimaan Sampel .	109
Gambar 5.11 Implementasi Antarmuka Halaman Validasi Penerimaan Sampel. .....	109
Gambar 5.12 Implementasi Antarmuka Halaman Kelola Data Diagnosa. ....	110
Gambar 5.13 Implementasi Antarmuka Halaman Form Diagnosa.....	110
Gambar 5.13 Implementasi Antarmuka Halaman Detail Diagnosa.....	111
Gambar 5.14 Implementasi Antarmuka Halaman Validasi Pelayanan .....	111
Gambar 5.15 Implementasi Antarmuka Halaman Form Laporan.....	112
Gambar 5.16 Hasil Rekap Laporan Kaji Ulang Permintaan .....	112
Gambar 6.3. Grafik Nilai Kemudahan Penggunaan Tiap Kategori Pertanyaan .	121

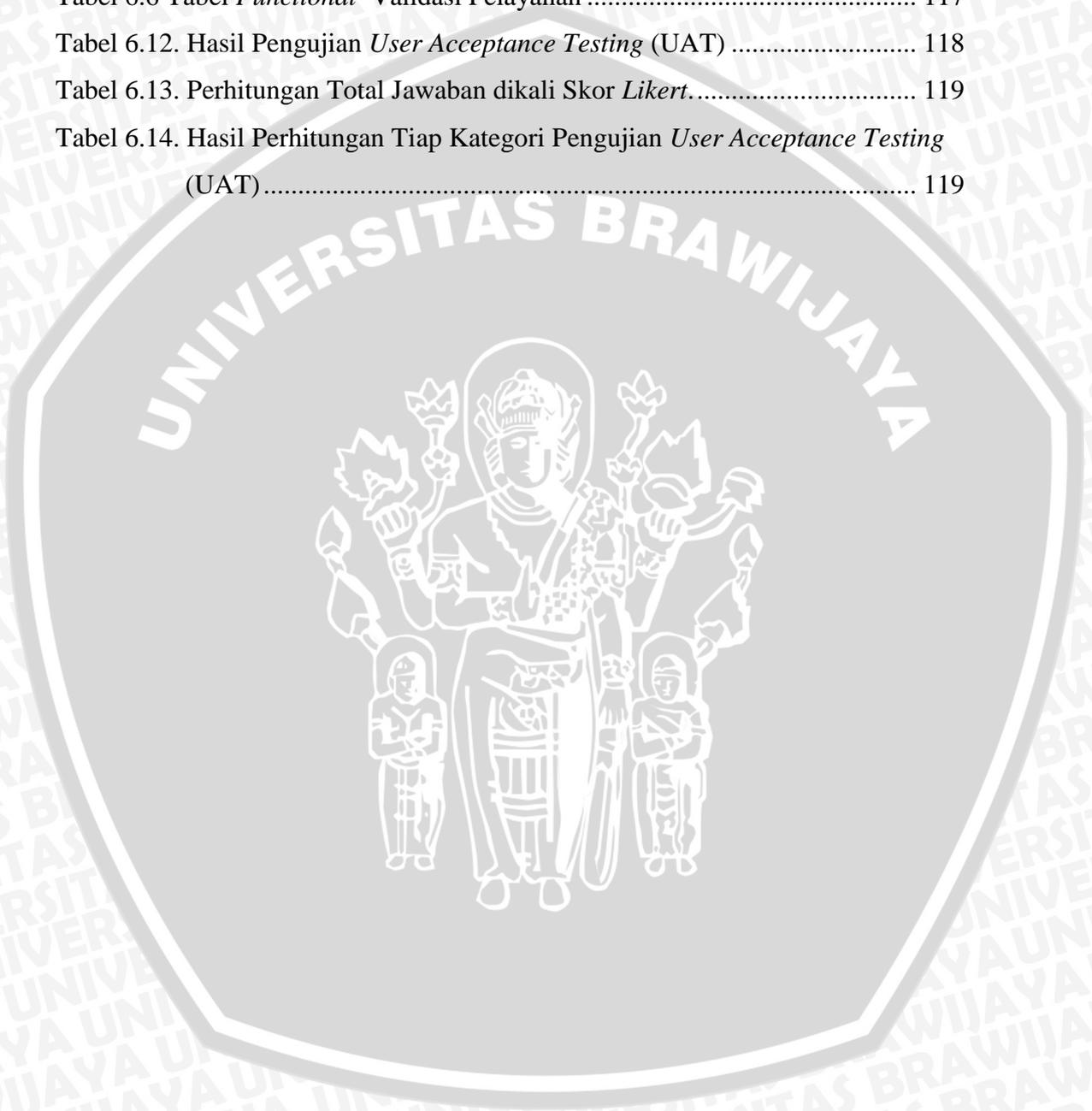


## DAFTAR TABEL

Tabel 2.3 Contoh Form UAT .....	34
Tabel 3.2 Spesifikasi Perangkat Keras .....	39
Tabel 3.3 Spesifikasi Perangkat Lunak .....	39
Tabel 4.1 Identifikasi Aktor .....	46
Tabel 4.2 Spesifikasi Kebutuhan Fungsional Sistem .....	47
Tabel 4.3 Prototype Fungsional Fitur Login .....	54
Tabel 4.4 Prototype Fungsional Fitur Validasi .....	55
Tabel 4.5 Prototype Fungsional Fitur Pelaporan .....	55
Tabel 4.6 Prototype Antarmuka Desain Antarmuka Login .....	56
Tabel 4.7 Prototype Antarmuka Desain Antarmuka Form Input dan Edit .....	57
Tabel 4.8 Prototype Antarmuka Desain Antarmuka Validasi Form .....	57
Tabel 4.9 Skenario <i>Usecase</i> Login .....	60
Tabel 4.10 Skenario Kelola Data Karyawan .....	61
Tabel 4.11 Skenario Kelola Data Wilayah .....	65
Tabel 4.12 Skenario Kelola Data Pendaftaran Pelayanan .....	68
Tabel 4.13 Skenario Validasi Penerimaan Sampel .....	71
Tabel 4.13 Tabel <i>Data type definition</i> dari tabel bagian .....	83
Tabel 4.14 Tabel <i>Data type definition</i> dari tabel bangsa .....	83
Tabel 4.15 Tabel <i>Data type definition</i> dari tabel desa .....	83
Tabel 4.16 Tabel <i>Data type definition</i> dari tabel detail pelayanan .....	84
Tabel 4.17 Tabel <i>Data type definition</i> dari tabel hasil pelayanan .....	84
Tabel 4.18 Tabel <i>Data type definition</i> dari tabel hewan .....	84
Tabel 4.19 Tabel <i>Data type definition</i> dari tabel jabatan .....	85
Tabel 4.21 Tabel <i>Data type definition</i> dari tabel <i>kaji_ulang_permintaan</i> .....	85
Tabel 4.22 Tabel <i>Data type definition</i> dari tabel karyawan .....	86
Tabel 4.23 Tabel <i>Data type definition</i> dari tabel kec .....	86
Tabel 4.24 Tabel <i>Data type definition</i> dari tabel kota .....	86
Tabel 4.25 Tabel <i>Data type definition</i> dari tabel laboratorium .....	87
Tabel 4.26 Tabel <i>Data type definition</i> dari tabel level .....	87
Tabel 4.27 Tabel <i>Data type definition</i> dari tabel member .....	87
Tabel 4.28 Tabel <i>Data type definition</i> dari tabel pelayanan .....	88

Tabel 4.29 Tabel <i>Data type definition</i> dari tabel perusahaan.....	88
Tabel 4.30 Tabel <i>Data type definition</i> dari tabel provinsi .....	88
Tabel 4.31 Tabel <i>Data type definition</i> dari tabel ras.....	89
Tabel 4.32 Tabel <i>Data type definition</i> dari tabel Sampel.....	89
Tabel 4.33 Tabel <i>Data type definition</i> dari tabel Sampel_hasil_uji.....	89
Tabel 4.34 Tabel <i>Data type definition</i> dari tabel Sampel_uji .....	90
Tabel 4.35 Tabel <i>Data type definition</i> dari tabel user.....	90
Tabel 4.36 Tabel <i>Data type definition</i> dari tabel bagian.....	91
Tabel 4.37 Tabel <i>Data filed description</i> dari tabel bangsa .....	91
Tabel 4.38 Tabel <i>Data filed description</i> dari tabel desa .....	91
Tabel 4.39 Tabel <i>Data filed description</i> dari tabel detail pelayanan.....	91
Tabel 4.40 Tabel <i>Data filed description</i> dari tabel hasil pelayanan.....	92
Tabel 4.41 Tabel <i>Data filed description</i> dari tabel hewan.....	92
Tabel 4.42 Tabel <i>Data filed description</i> dari tabel jabatan.....	92
Tabel 4.43 Tabel <i>Data filed description</i> dari tabel jabatan_fungsional.....	93
Tabel 4.44 Tabel <i>Data filed description</i> dari .....	93
Tabel 4.45 Tabel <i>Data filed description</i> dari tabel karyawan.....	94
Tabel 4.46 Tabel <i>Data filed description</i> dari tabel kec.....	94
Tabel 4.47 Tabel <i>Data filed description</i> dari tabel kota.....	94
Tabel 4.48 Tabel <i>Data filed description</i> dari tabel laboratorium.....	95
Tabel 4.49 Tabel <i>Data filed description</i> dari tabel level.....	95
Tabel 4.50 Tabel <i>Data filed description</i> dari tabel member .....	95
Tabel 4.51 Tabel <i>Data filed description</i> dari tabel pelayanan .....	96
Tabel 4.52 Tabel <i>Data filed description</i> dari tabel perusahaan .....	96
Tabel 4.53 Tabel <i>Data filed description</i> dari tabel provinsi .....	97
Tabel 4.54 Tabel <i>Data filed description</i> dari tabel ras.....	97
Tabel 4.55 Tabel <i>Data filed description</i> dari tabel Sampel .....	97
Tabel 4.56 Tabel <i>Data filed description</i> dari tabel Sampel_hasil_uji.....	97
Tabel 4.57 Tabel <i>Data filed description</i> dari tabel Sampel_uji .....	98
Tabel 4.58 Tabel <i>Data filed description</i> dari tabel user.....	98
Tabel 5.1 Spesifikasi Perangkat Keras.....	102
Tabel 5.2 Spesifikasi Perangkat Lunak.....	103

Tabel 6.1 Tabel <i>Functional Testing</i> Login.....	113
Tabel 6.2 Tabel <i>Functional</i> Kelola Karyawan.....	114
Tabel 6.3 Tabel <i>Functional</i> Pendaftaran Pelayanan .....	115
Tabel 6.4 Tabel <i>Functional</i> Kelola Penerimaan Sampel .....	116
Tabel 6.6 Tabel <i>Functional</i> Validasi Pelayanan .....	117
Tabel 6.12. Hasil Pengujian <i>User Acceptance Testing</i> (UAT) .....	118
Tabel 6.13. Perhitungan Total Jawaban dikali Skor <i>Likert</i> .....	119
Tabel 6.14. Hasil Perhitungan Tiap Kategori Pengujian <i>User Acceptance Testing</i> (UAT).....	119



**DAFTAR LAMPIRAN**

LAMPIRAN 1 .....	123
L1-1 HASIL KUSIONER UAT .....	124
L1-2 HASIL KUSIONER UAT .....	126
L1-3 HASIL KUSIONER UAT .....	128
L1-4 HASIL KUSIONER UAT .....	130
L1-5 HASIL KUSIONER UAT .....	132
L1-6 HASIL KUSIONER UAT .....	134
L1-7 HASIL KUSIONER UAT .....	136
L1-8 HASIL KUSIONER UAT .....	138
L1-9 HASIL KUSIONER UAT .....	140
L1-10 HASIL KUSIONER UAT .....	142



## BAB I PENDAHULUAN

### 1.1 Latar Belakang

Laboratorium adalah tempat atau kamar tertentu yang dilengkapi dengan peralatan untuk mengadakan percobaan [7]. Laboratorium Veteriner adalah laboratorium yang mempunyai tugas dan fungsi dalam bidang pelayanan kesehatan hewan dan kesehatan masyarakat veteriner [7]. Salah satu contohnya di Unit Pelaksanaan Teknis Laboratorium Kesehatan Hewan Malang.

Unit Pelaksanaan Teknis Laboratorium Kesehatan Hewan Malang (UPT Labkeswan Malang) merupakan salah satu dari institusi kesehatan pemerintahan yang mempunyai tugas melaksanakan sebagaimana tugas dinas dibidang pemeriksaan, penyidikan dan diagnosis penyakit hewan [13].

Prosedur pelayanan yang digunakan oleh UPT Labkeswan Malang mengacu pada Standar Operasional Prosedur (SOP) pelayanan berupa Standar Pelayanan Publik (SPP). Pengelolaan data pelayanan yang ada di UPT Labkeswan Malang yaitu berupa pelayanan pasif, pelayanan semi aktif, dan pelayanan aktif untuk prosedur pelayanan terhadap masyarakat, pemetaan penyakit sebagai data referensi penyakit yang sedang berkembang serta pengelolaan data hasil pengujian [13].

Dalam proses pelayanan di UPT Labkeswan Malang berdasarkan SOP SPP menyatakan bahwa kepastian prosedur pelayanan sehingga konsumen diberikan jaminan kepastian dari proses pelayanan. Namun dalam pelaksanaannya para petugas masih sering mengalami kesulitan dalam memantau proses pelayanan yang mereka kelola. Karena dalam sistem yang telah berjalan belum ada yang menginformasikan tentang mekanisme *monitoring* tahapan proses data sampel. Kemudian petugas administratif harus menanyakan kepada bagian terkait yang menangani data sampel tersebut. Karena harus mendatangi ruangan suatu bagian kerja yang jaraknya relatif jauh dari ruangan operasional kerja bagian administratif dan masih belum lagi mencari berkas-berkas data sampel yang telah masuk sebagai data pelayanan sehingga menghabiskan banyak waktu. Kemudian menurut prosedur dalam proses pelayanan lainnya yaitu kepastian waktu penyelesaian pelayanan. Kemudian menurut prosedur dalam proses pelayanan lainnya yaitu

kepastian waktu penyelesaian pelayanan. Dimana proses pelayanan untuk membantu konsumen dituntut dengan segera dan hasil yang akurat. Namun dalam sistem yang sedang berjalan belum ada durasi waktu yang menyatakan berapa lama proses pengujian atau pemeriksaan untuk setiap data Sampel yang sedang diuji dan sudah sampai mana proses pengujian atau pemeriksaan yang sedang dilakukan. Sehingga pihak Administratif tidak bisa memberikan Informasi yang jelas mengenai proses pengujian atau pemeriksaan kepada konsumen. Hal ini mungkin akan menimbulkan kekecewaan terhadap konsumen yang melakukan pengujian atau pemeriksaan di UPT Labkeswan Malang. Setiap proses pelayanan pasti menghasilkan laporan yang harus diorganisir. Sering terjadi penumpukan atau tumpang tindih berkas data Sampel serta dalam mencari berkas pun dan pembuatan laporan proses selanjutnya akan terhambat, sehingga kinerja para karyawan menjadi kurang efisien. Dengan demikian perlu adanya suatu Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang yang sesuai dengan penerapan SOP SPP.

Oleh karena itu, Melihat hal tersebut diatas maka diperlukan suatu solusi dalam penelitian ini dalam membangun sebuah Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang. Diharapkan dengan sistem informasi ini dapat meningkatkan kualitas pelayanan sesuai dengan SOP SPP yang diterapkan oleh UPT Labkeswan Malang.

## **1.2 Rumusan Masalah**

Rumusan masalah yang diambil dari Rancang Bangun Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang adalah sebagai berikut:

1. Bagaimana merancang dan membangun Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang ?
2. Bagaimana implementasi Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang sesuai dengan SOP Standar Pelayanan Publik (SPP) ?

3. Bagaimana melakukan pengujian pada implementasi Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang dengan menggunakan metode pengujian *Fungsional Testing* dan *User Acceptance Testing* (UAT) ?

### 1.3 Batasan Masalah

Batasan Masalah dari pembuatan Skripsi Rancang Bangun Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang adalah sebagai berikut:

1. Sistem Informasi di bangun sesuai dengan standart pelayanan di UPT Laboratorium Kesehatan Hewan Malang yaitu mengacu pada SOP Standar Pelayanan Publik (SPP).
2. Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang hanya mengacu pada proses pelayanan pengujian sampel mulai dari pendaftaran pengujian sampel, proses kaji ulang sampel, permintaan pengujian sampel, diagnosa sampel, rekam diagnosa hingga hasil cetak laporan pelayanan yang menangani pengujian sample.
3. Bahasa pemrograman yang digunakan yaitu HTML, PHP dengan *Framework* Laravel 4.2 dan XAMPP sebagai server localhost serta basis data MySQL.
4. Pengujian sistem menggunakan metode pengujian *Fungsional Testing* dan *User Acceptance Testing* (UAT).

### 1.4 Tujuan

Berikut tujuan dari Rancang Bangun Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang:

1. Untuk menganalisa perancangan dan membangun Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang.
2. Untuk mengimplementasikan Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang sesuai dengan SOP Standar Pelayanan Publik (SPP).

3. Untuk melakukan pengujian pada implementasi Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang berdasarkan metode pengujian *Functional Testing* dan *User Acceptance Testing* (UAT).

### 1.5 Manfaat

Manfaat dalam penelitian Rancang Bangun Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang:

1. Mempermudah dalam pengembangan sistem karena memiliki dokumentasi sistem yang terstruktur sehingga bisa dikembangkan berdasarkan perencanaan yang telah dibuat sebelumnya.
2. Mempermudah dalam pengelolaan data pelayanan.
3. Mempermudah pelaporan pelayanan dan hasil uji sampel.
4. Digunakan sebagai alat bantu memonitoring pemetaan penyakit terhadap Sampel data yang telah diujikan.
5. Digunakan sebagai alat bantu dalam proses pelayanan di UPT Laboratorium Kesehatan Hewan Malang.

### 1.6 Sistematika Penulisan

Sistematika penulisan laporan skripsi ini 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 landasan teori yang digunakan dalam penyusunan laporan skripsi ini. Landasan teori yang digunakan antara lain, Pelayanan Publik), Sistem Informasi , Tugas Lab Keswan Malang, Standar Operasional Prosedur Standar Pelayanan Publik (SOP SPP), Pengertian Laboratorium, Sistem Informasi Manajemen, Sistem Manajemen Laboratorium, Alur Sistem Pelayanan di Kantor UPT Laboratorium Kesehatan Hewan Malang, Perangkat Lunak,

Rekayasa Perangkat Lunak, Model Proses Prototipe (Prototype), MVC (*Model View Controller*), *Framework* Laravel 4.2, Pengujian Perangkat Lunak, *Functional Testing* dan *User Acceptance Testing* (UAT).

### **BAB III METODOLOGI PENELITIAN**

Metode penelitian membahas metode yang digunakan dalam penelitian yang terdiri dari identifikasi permasalahan, identifikasi solusi, realisasi solusi, dan analisa hasil realisasi solusi.

### **BAB IV ANALISIS DAN PERANCANGAN**

Bab ini berisi tentang alur data aplikasi, perancangan desain aplikasi, dan pembuatan 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 Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang.

## BAB II

### KAJIAN PUSTAKA DAN DASAR TEORI

#### 2.1 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 [13].

##### 2.1.1 Tugas UPT Laboratorium Kesehatan Hewan.

UPT Laboratorium Kesehatan Hewan mempunyai tugas melaksanakan Sebagai tugas dinas dibidang pemeriksaan, penyidikan dan diagnosis penyakit hewan [13].

- a) Fungsi Lab Keswan Malang :
- b) Pengumpulan / pengambilan specimen
- c) Pemeriksaan Penyakit Hewan;
- d) Diagnosis Penyakit Hewan;
- e) Evaluasi Penyakit Hewan;
- f) Penanggulangan Penyakit Hewan;
- g) Pembinaan Penyidikan Penyakit Hewan;
- h) Pelayanan Pengujian Mutu Produk Pangan asal Hewan dan Mutu produk non PAH;
- i) Pelayanan Kepada Masyarakat

## 2.2 Pengertian Laboratorium

Laboratorium adalah suatu tempat ruangan untuk melakukan penelitian, eksperimen atau pengukuran. Fasilitas yang dimiliki sebuah laboratorium tergantung dari jenis kegiatan yang berlangsung di laboratorium [7]. Salah satu contohnya adalah Laboratorium Veteriner. Laboratorium Veteriner adalah laboratorium yang mempunyai tugas dan fungsi dalam bidang pelayanan kesehatan hewan dan kesehatan masyarakat veteriner [7].

## 2.3 Pengertian Sistem

Pendekatan sistem yang lebih menekankan pada prosedur mendefinisikan sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkelompok dan bekerjasama untuk melakukan kegiatan pencapaian sasaran tertentu. Makna dari prosedur sendiri, yaitu urutan yang tepat dari tahapan-tahapan instruksi yang menerangkan apa (*what*) yang harus dikerjakan, siapa (*who*) yang mengerjakan, kapan (*when*) dikerjakan dan bagaimana (*how*) mengerjakannya. Sedangkan pendekatan yang menekankan pada komponen mendefinisikan “sistem” sebagai berikut. Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu [8].

## 2.4 Pengertian Informasi

Informasi adalah arti dari hubungan dan penafsiran data yang mengizinkan seseorang untuk membuat keputusan. Informasi dikatakan berharga jika Informasi itu mempengaruhi proses pengambilan keputusan lebih baik. Sasaran utama dari sistem Informasi juga dapat berarti beberapa kesatuan yang tak terukur yang dapat mengurangi ketidakpastian tentang suatu peristiwa atau langkah [8].

## 2.5 Pengertian Sistem Informasi

Sistem Informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan [8].

## 2.6 Sistem Infomasi Manajemen

Menurut Barry E.Cushing, SIM adalah “Suatu sistem Informasi manajemen adalah kumpulan dari manusia dan sumber daya modal di dalam suatu organisasi yang bertanggung jawab mengumpulkan dan mengolah data untuk menghasilkan Informasi yang berguna untuk semua tingkatan manajemen di dalam kegiatan perencanaan dan pengendalian” [5].

Menurut Frederick H.Wu SIM adalah “Sistem Informasi Manajemen adalah kumpulan-kumpulan dari sistem-sistem yang menyediakan Informasi untuk mendukung manajemen”[5].

Menurut Gordon.B Davis, dalam buku ‘Analisis dan Desain Informasi’ SIM, adalah : “Sistem Informasi Manajemen merupakan suatu sistem yang melakukan fungsi-fungsi untuk menyediakan semua Informasi yang mempengaruhi semua operasi organisasi”[5].

Menurut George M.Scott, dalam buku “Prinsip-prinsip SIM’ adalah : ‘Sistem Informasi Manajemen adalah serangkaian Sub-sistem Informasi yang menyeluruh dan terkoordinasi dan secara rasional terpadu yang mampu yang mampu mentransformasi data sehingga menjadi Informasi lewat serangkaian cara guna meningkatkan produktivitas yang sesuai dengan gaya dan sifat manajer atas dasar criteria mutu yang telah ditetapkan”[5].

Jadi dari beberapa definisi tersebut,dapat dirangkum bahwa Sistem Informasi Manajemen adalah kumpulan dari interaksi sistem-sistem Informasi yang menghasilkan Informasi yang berguna untuk semua tingkatan manajemen.

## 2.7 Sistem Manajemen Laboratorium

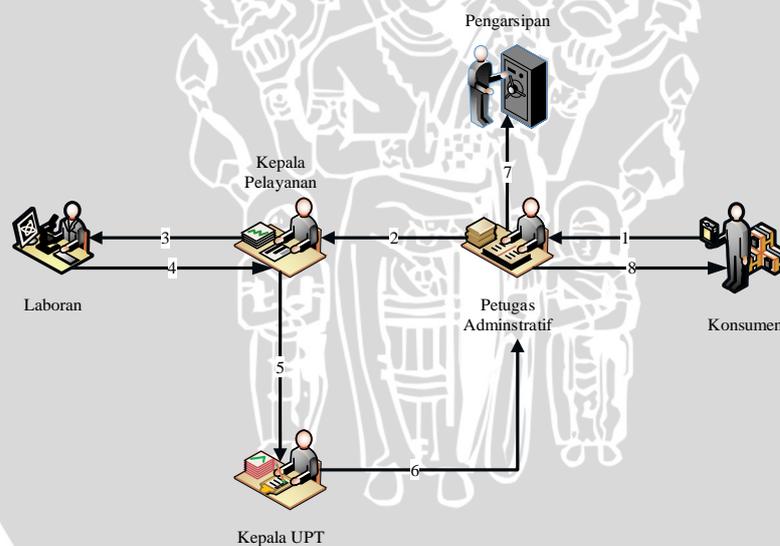
Dalam pendahuluan ISO 17025 tentang sistem mutu laboratorium, “Organisasi laboratorium perlu diarahkan dan dikendalikan secara sistematis dan transparan agar bisa berhasil. Keberhasilan dapat dicapai melalui pengimplementasian dan pemeliharaan sistem manajemen yang didesain untuk selalu memperbaiki efektivitas dan efisiensi kinerjanya sambil mempertimbangkan kebutuhan semua pihak berkepentingan” [7].

### 2.7.1 Alur Sistem Pelayanan di Kantor UPT Laboratorium Kesehatan Hewan Malang.

Sistem Informasi laboratorium kesehatan Hewan adalah prosedur sistematis untuk mengumpulkan, menyimpan, mempertahankan, mengolah, mengambil dan memvalidasi data yang dibutuhkan oleh laboratorium kesehatan tentang kegiatan pelayanannya untuk pengambilan keputusan manajemen. Tujuan utama dari sistem Informasi laboratorium kesehatan hewan adalah mengumpulkan, mengolah dan menyajikan data dengan lebih akurat, informatif, sesuai dengan kebutuhan, tepat waktu dan sesuai dengan Standar Operasional Prosedur pelayanan.

Dalam alur sistem pelayanan di UPT Laboratorium Kesehatan Hewan Malang memiliki jenis pelayanan sebagai berikut :

#### a) Pelayanan Pasif.



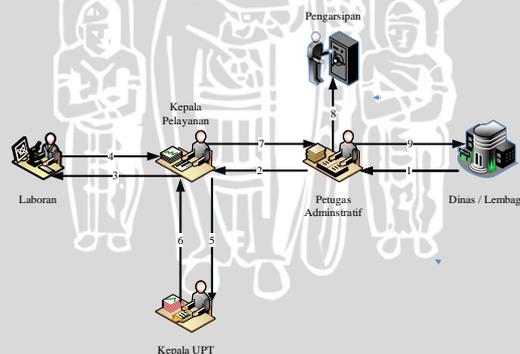
**Gambar 2.1 Gambar Alur Proses Pelayanan Pasif.**

Berikut adalah penjelasan prosedur pelayanan pasif sesuai dengan pada Gambar 2.1:

1. Konsumen melakukan registrasi dengan mengisi formulir dan data Sampel yang akan di ujiakan. Kemudian diserahkan ke petugas administratif pelayanan.

2. Petugas Administratif akan mencatat data tersebut dan menyerahkan ke bagian pelayanan untuk di kategorikan pengujiannya. Pengujian Sampel antara Kesehatan Hewan (Keswan) dengan Kesehatan Masyarakat Verterier (Kesmavet).
3. Setelah di kategorikan oleh Petugas pelayanan data beserta Sampel akan di serahkan ke bagian laboratorium untuk dilakukan pemeriksaan Sampel.
4. Kemudian hasil uji Sampel dari pemeriksaan laboran akan di berikan kepada kepala pelayanan untuk pengesahan hasil pemeriksaan, lalu membuat surat hasil pemeriksaan.
5. Melalui tahap pengecekan di bagian pelayanan kemudian di serahkan ke kepala UPT untuk di sahkan sehingga surat hasil uji bisa di dapatkan oleh konsumen.
6. Surat yang telah disahkan akan digandakan oleh petugas administratif di serahkan ke pengarsipan dan yang asli di berikan kepada konsumen.
7. Surat yang di gandakan akan di simpan sebagai bukti serta data record sebagai arsip.
8. Surat hasil uji di terbitkan dan diberikan kepada konsumen.

#### b) Pelayanan Semi Aktif.



**Gambar 2.2 Gambar Alur Proses Pelayanan Semi Aktif.**

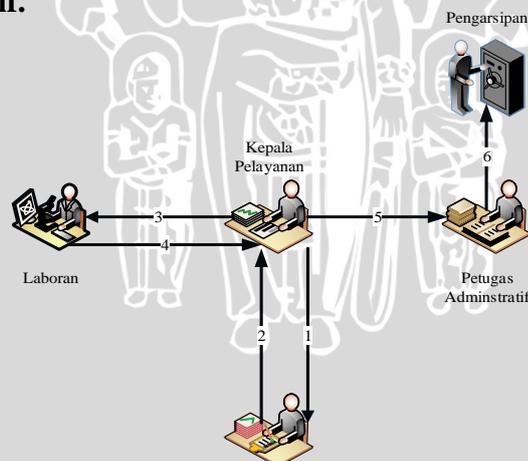
Berikut adalah penjelasan prosedur pelayanan semi aktif sesuai dengan pada Gambar 2.2 :

1. Lembaga/Dinas/Institusi memberikan surat permohonan pemeriksaan Sampel uji.
2. Petugas Administratif akan mencatat data tersebut dan menyerahkan ke bagian pelayanan untuk di kategorikan pengujiannya. Pengujian Sampel

antara Kesehatan Hewan (Keswan) dengan Kesehatan Masyarakat Verterier (Kesmavet).

3. Setelah di kategorikan oleh Petugas pelayanan data beserta Sampel akan di serahkan ke bagian laboratorium untuk dilakukan pemeriksaan Sampel.
4. Kemudian hasil uji Sampel dari pemeriksaan laboran akan di berikan kepada kepala pelayanan untuk pengesahan hasil pemeriksaan, lalu membuat surat hasil pemeriksaan.
5. Melalui tahap pengecekan di bagian pelayanan kemudian di serahkan ke kepala UPT untuk di sahkan sehingga surat hasil uji bisa di dapatkan oleh Lembaga / Dinas / Istitusi.
6. Surat yang telah disahkan akan digandakan oleh petugas administratif di serahkan ke pengarsipan dan yang asli di berikan kepada Lembaga / Dinas / Istitusi.
7. Surat yang di gandakan akan di simpan sebagai bukti serta data record sebagai arsip.
8. Surat haisl uji di terbitkan dan diberikan kepada Lembaga/Dinas/Istitusi.

### c) Pelayanan Aktif.



**Gambar 2.3 Gambar Alur Proses Pelayanan Aktif.**

Berikut adalah penjelasan prosedur pelayanan aktif sesuai dengan pada Gambar 2.3 :

1. Program Kerja pada bagian Pelayanan untuk melakukan sidak atau pemeriksaan kepada masyarakat di suatu daerah.

2. Meminta persetujuan Petugas Administratif akan mencatat data tersebut dan menyerahkan ke bagian pelayanan untuk di kategorikan pengujiannya. Pengujian Sampel antara Kesehatan Hewan (Keswan) dengan Kesehatan Masyarakat Veteriner (Kesmavet).
3. Setelah di kategorikan oleh Petugas pelayanan data beserta Sampel akan di serahkan ke bagian laboratorium untuk dilakukan pemeriksaan Sampel.
4. Kemudian hasil uji sampel dari pemeriksaan laboran akan di berikan kepada kepala pelayanan untuk pengesahan hasil pemeriksaan, lalu membuat surat hasil pemeriksaan.
5. Melalui tahap pengecekan di bagian pelayanan kemudian di serahkan ke kepala UPT untuk di sahkan sehingga sesuai dengan hasil program kerja bagian pelayanan.
6. Surat yang telah disahkan akan digandakan oleh petugas administratif di serahkan ke pengarsipan sebagai bukti hasil program kerja bagian pelayanan.

## 2.8 Perangkat Lunak

### 2.8.1 Definisi Perangkat Lunak

Menurut Roger S. Pressman, definisi perangkat lunak (*software*) adalah [9] :

1. Instruksi - instruksi yang bila dieksekusi akan memberikan fungsi dan untuk kerja yang diinginkan.
2. Struktur data yang memungkinkan program untuk memanipulasi Informasi .
3. Dokumen-dokumen yang menjelaskan pengoperasian dan penggunaan program.

### 2.8.2 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 keras. Berikut adalah karakteristik perangkat lunak (*software*) [9]:

Biaya (*cost*) perangkat lunak terpusat pada perenkayaan. Jadi proyek perangkat lunak tidak dapat dijalankan seperti proses pamanufaktur.

1. 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.

2. 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.

## 2.9 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 [9]:

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.

### 2.9.1 Fase-Fase Rekayasa Perangkat Lunak

Fase-fase yang berhubungan dengan rekayasa perangkat lunak, dapat dikategorikan ke dalam tiga tahapan umum yaitu [9]

#### 1. Fase Pendefinisian

Selama fase ini, pelaku rekayasa perangkat lunak berusaha mengidentifikasi data apa yang akan diproses, fungsi dan cara kerja seperti apa yang diinginkan, perilaku sistem seperti apa yang diinginkan, tampilan apa yang akan dibuat, desain apa yang harus ada, dan kriteria validasi apa yang dibutuhkan untuk mendefinisikan sebuah Informasi yang valid.

#### 2. Fase Pengembangan

Selama proses pengembangan, pelaku rekayasa perangkat lunak berusaha mendefinisikan bagaimana data disusun, bagaimana fungsi diimplementasikan di dalam arsitektur perangkat lunak, bagaimana detail-detail prosedural diimplementasikan, bagaimana tampilan dikarakterisasi, bagaimana desain diterjemahkan ke dalam bahasa pemrograman, dan bagaimana pengujian akan dilakukan.

#### 3. Fase Pendukung

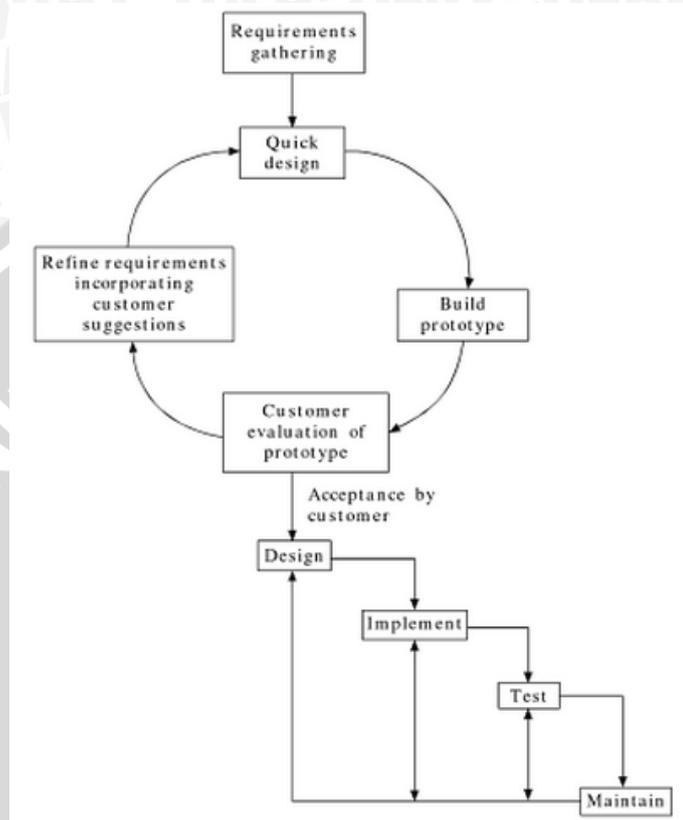
Pada fase ini dilakukan pembetulan kesalahan, untuk meningkatkan kualitas perangkat lunak, sehingga dapat memenuhi kebutuhan dari pelanggan.

### 2.10 Model Proses Prototipe (Prototyping)

Menurut Roger S. Pressman, Seringkali pelanggan mendefinisikan jumlah sasaran perangkat lunak secara umum, tetapi tidak bisa mendefinisikan spesifikasi kebutuhan yang rinci untuk fungsi-fungsi dan fitur-fitur yang nantinya akan dimiliki perangkat lunak yang akan dikembangkan [9].

Meskipun pembuatan prototype dapat digunakan sebagai model yang berdiri sendiri, pembuatan protipe lebih umum digunakan sebagai teknik yang

dapat diimplementasikan di dalam konteks setiap model proses perangkat lunak yang telah dan akan diimplementasikan. Berikut adalah paradigma pembuatan prototipe seringkali membantu tim pengembangan perangkat lunak pada Gambar 2.4.



**Gambar 2.4 Tahapan Proses Model Prototipe.**

**Sumber : [6]**

### **2.10.1. Kelebihan Metode Prototype**

- Adanya komunikasi yang baik antara pengembang dan pelanggan.
- Pengembangan dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan
- Lebih menghemat waktu dalam pengembangan system
- Penerapan menjadi lebih mudah karena pemakai mengetahui apa yang diharapkannya

### 2.10.2. Kekurangan Metode Prototype.

- a) Resiko tinggi yaitu untuk masalah-masalah yang tidak terstruktur dengan baik, ada perubahan yang besar dari waktu ke waktu, dan adanya persyaratan data yang tidak menentu.
- b) Interaksi pemakai penting. Sistem harus menyediakan dialog *on-line* antara pelanggan dan komputer.
- c) Hubungan pelanggan dengan komputer yang disediakan mungkin tidak mencerminkan teknik perancangan yang baik.

### 2.10.3. Jenis Prototype.

Ada berbagai jenis prototipe perangkat lunak yang digunakan dalam pengembangan *software* sebagai berikut [6]:

#### a. *Throwaway / Rapid Prototyping*

*Throwaway prototyping* juga disebut sebagai *rapid* atau *prototyping* cepat selesai. Jenis *prototyping* ini menggunakan upaya yang sangat sedikit dengan analisis kebutuhan minimum untuk membangun prototipe. Setelah persyaratan yang sebenarnya dipahami, prototipe tersebut akan dibuang dan dikembangkan dengan pemahaman yang lebih jelas tentang kebutuhan pengguna.

#### b. *Evolutionary Prototyping*

*Evolutionary Prototyping* juga disebut sebagai *breadboard Prototyping* didasarkan pada membangun prototipe fungsional yang sebenarnya dengan fungsi minimal di awal. Menggunakan *Evolutionary Prototyping* hanya persyaratan dipahami dengan baik termasuk dalam prototipe dan persyaratan ditambahkan ketika mereka mengerti.

#### c. *Incremental Prototyping*

*Incremental Prototyping* mengacu pada beberapa prototipe fungsional dari berbagai sub sistem dan kemudian semua prototipe yang tersedia dibangun secara lengkap.

#### d. *Ekstrim Prototyping*

*Ekstrim prototyping* digunakan dalam domain pengembangan web. Ini terdiri dari tiga tahap berurutan. Pertama, prototipe dasar disajikan dalam format html. Kemudian pengolahan data disimulasikan menggunakan lapisan layanan prototipe. Akhirnya layanan diimplementasikan dan terintegrasi untuk prototipe akhir. Proses ini disebut *Ekstrim Prototyping* digunakan untuk menarik perhatian pada tahap kedua dari proses, di mana UI berfungsi penuh dikembangkan dengan sangat sedikit berkaitan dengan layanan yang sebenarnya.

### 2.11 Unified Modelling Language (UML)

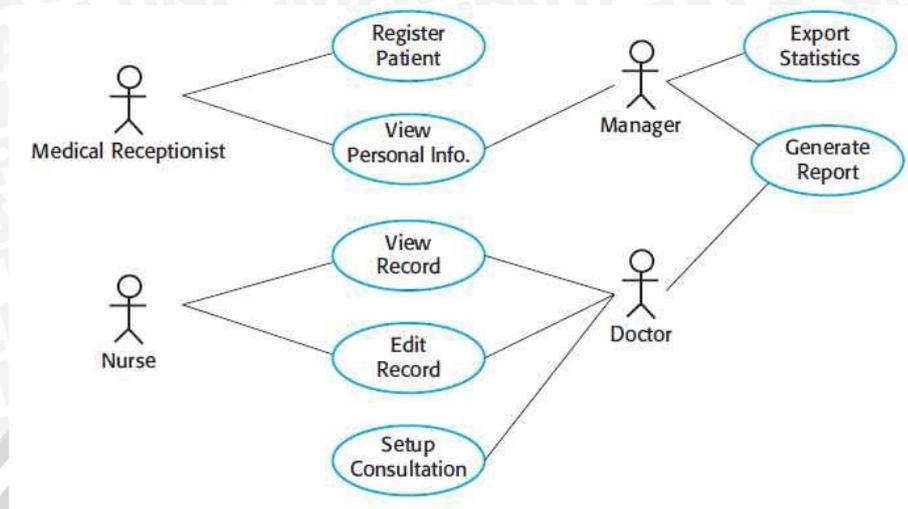
Menurut Sommerville pemodelan sistem adalah proses mengembangkan abstract models dari suatu sistem, dengan masing – masing merepresentasikan perspektif yang berbeda dari sistem tersebut. Pemodelan sistem umumnya merepresentasikan sistem menggunakan beberapa jenis notasi grafis, yang sekarang hampir selalu didasarkan pada notasi dalam *Unified Modelling Language* (UML) [10]. 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 [11].

#### 2.11.1 *Usecase Diagram*

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

Setiap *Usecase* merepresentasikan tugas terpisah yang melibatkan interaksi eksternal dengan sistem. *Usecase* mengidentifikasi interaksi individu antara sistem dan pengguna atau sistem lain. Setiap *Usecase* harus didokumentasikan dengan deksripsi tekstual, yang kemudian dapat dikaitkan dengan model lain dalam UML

yang akan membangun skenario dengan lebih terperinci [11]. Gambar 2.5 merupakan contoh *Usecase Diagram* konsultasi :



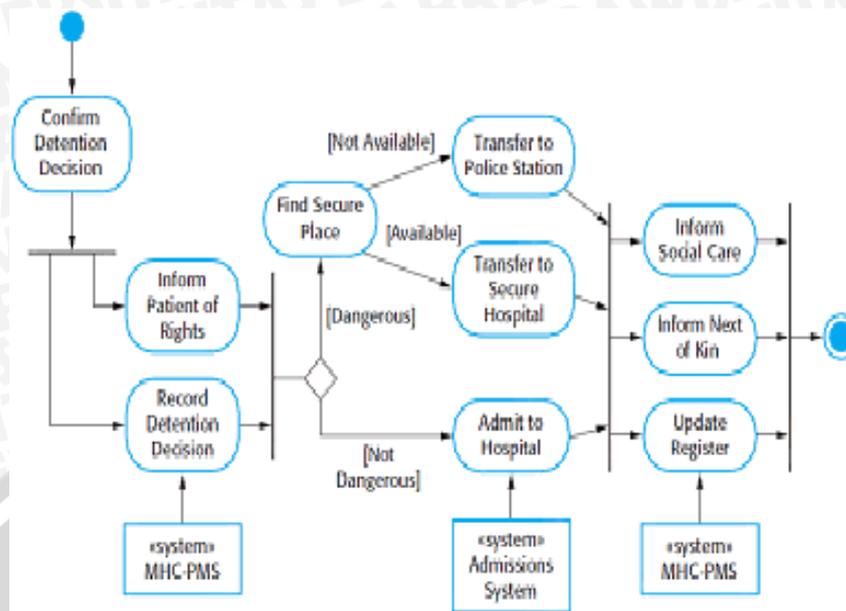
**Gambar 2.5 Gambar *Usecase Diagram***

**Sumber :** [11]

Aturan konsultasi mengizinkan 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 [11].

### 2.11.2 *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 memvisualisasikan interelasi dan interaksi antara *Usecase* yang berbeda, serta sering digunakan untuk mengasosiasikan dengan *Class* yang berbeda. Kekuatan Diagram *Activity* adalah merepresentasikan concurrent *Activity*.



**Gambar 2.6 Gambar Activity Diagram**

**Sumber :** [11].

Gambar 2.6 adalah sebuah *Activity Diagram* yang ditujukan untuk menunjukkan 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 mengindikasikan koordinasi aktifitas. 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 [11].

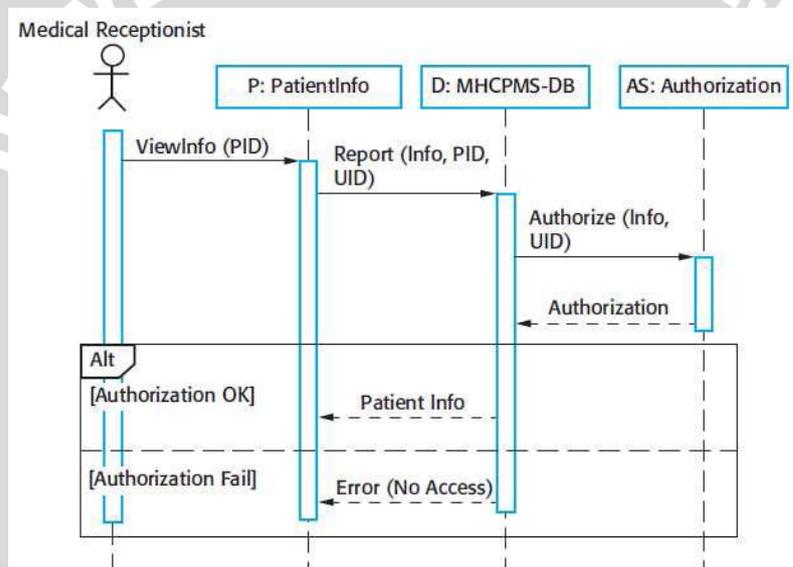
### 2.11.3 Sequence Diagram

*Sequence Diagram* dalam UML yang 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. [11].

Sesuai namanya, *Sequence Diagram* menunjukkan urutan interaksi yang terjadi selama *Usecase 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 [11].

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 2.7** Gambar *Sequence Diagram*

Sumber : [11]

Dari Gambar 2.7, 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 mengembalikan Informasi yang dibutuhkan, penyediaan resepsionis identifier memungkinkan pemeriksaan keamanan (pada tahap ini, tidak peduli darimana UID berasal).

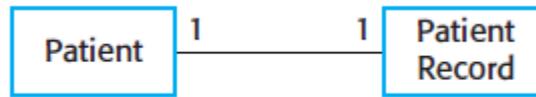
3. Pemeriksaan dengan sistem otorisasi kepada pengguna yang memiliki wewenang.
4. Jika diizinkan, Informasi pasien dikembalikan. Jika otorisasi gagal, maka pesan kesalahan akan dikembalikan.

#### 2.11.4 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 yang dibutuhkan.

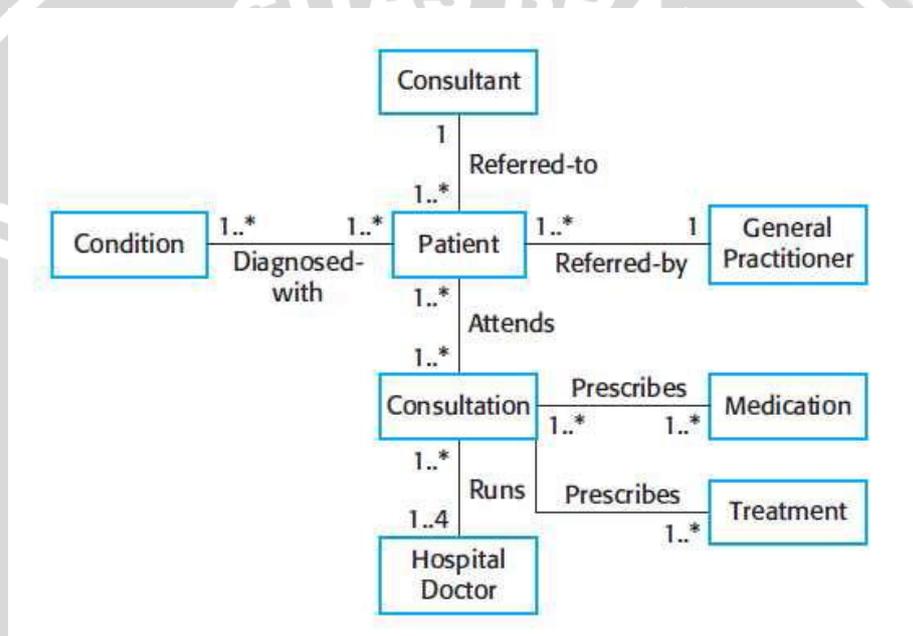
*Class Diagram* pada UML dapat diekspresikan dengan tingkat rincian yang berbeda. Ketika seorang developer mengembangkan sebuah model, tahap pertama biasanya melihat world, mengidentifikasi objek esensial, dan merepresentasikan itu sebagai *Class* . Cara paling sederhana penulisannya adalah mendeskripsikan nama kelas dalam sebuah kotak. Anda juga dapat mencatat adanya asosiasi dengan menggambar garis antara kelas. Sebagai contoh gambar dibawah adalah gambar sederhana yang menampilkan dua kelas, menunjukkan adanya hubungan antar record pasien dan pasien itu sendiri [11].



Gambar 2.8 Gambar Asosiasi Kelas Sederhana

Sumber : [11]

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 2.9 Gambar Class dan Asosiasi

Sumber : [11]

Gambar 2.9 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). 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 2.10 :



**Gambar 2.10 Gambar Consultation Class**  
Sumber : [11]

Pada Gambar 2.10 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 menunjukkan 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* [11].

## 2.12 PHP

Menurut Agus Saputra dalam bukunya yang berjudul Membangun Aplikasi E-Library untuk Panduan Skripsi, PHP atau yang memiliki kepanjangan PHP Hypertext Preprocessor merupakan suatu bahasa pemrograman yang difungsikan untuk membangun suatu website dinamis. PHP menyatu dengan kode HTML, maksudnya adalah beda kondisi. HTML digunakan sebagai pembangun atau pondasi dari kerangka layout web, sedangkan PHP difungsikan sebagai prosesnya sehingga dengan adanya PHP tersebut, web akan sangat mudah di-maintenance.

PHP berjalan pada sisi server sehingga PHP disebut juga sebagai bahasa Server Side Scripting. Artinya bahwa dalam setiap/untuk menjalankan PHP, wajib adanya web server.

PHP ini bersifat open source sehingga dapat dipakai secara cuma-cuma dan mampu lintas platform, yaitu dapat berjalan pada sistem operasi Windows maupun Linux. PHP juga dibangun sebagai modul pada web server apache dan sebagai binary yang dapat berjalan sebagai CGI.

### 2.13.1. Keunggulan PHP

Ada beberapa alasan yang menjadi dasar pertimbangan mengapa menggunakan PHP.

1. Mudah dipelajari, alasan tersebut menjadi salah satu alasan utama untuk menggunakan PHP, Pemula pun akan mampu untuk menjadi web master PHP.
2. Mampu Lintas *Platform*, artinya PHP dapat / mudah diaplikasikan ke berbagai platform OS(*Operating Sytem*) dan hampir semua *browser* juga mendukung PHP.
3. Free alias Gratis, bersifat *Open Source*.
4. PHP memiliki tingkat akses yang cepat.
5. Didukung oleh beberapa macam web server, PHP mendukung beberapa web server, seperti Apache, IIS, Lighttpd, Xitami.
6. Mendukung database, PHP mendukung beberapa database, baik yang gratis maupun yang berbayar, seperti MySQL, PostgreSQL, mSQL, Informix, SQL server, Oracle.

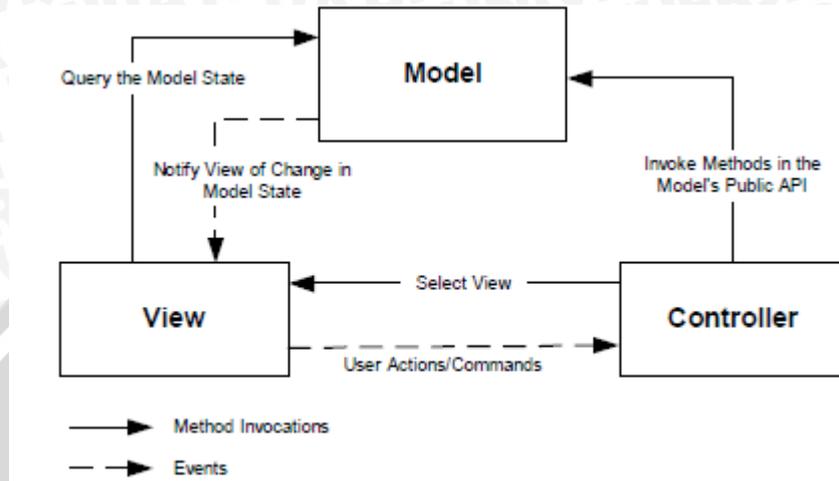
### 2.13 MVC (*Model View Controller*)

*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 XEROX PARC yang bekerja untuk pembuatan bahasa pemrograman *Smalltalk* pada tahun 1970-1980. Metode MVC membagi aplikasi menjadi tiga bagian yaitu [4]:

1. *Model*  
Befungsi 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 2.12 Gambar Konsep MVC**

**Sumber : [4]**

Kelebihan-kelebihan metode MVC adalah sebagai berikut [4] :

1. Bagian model memberikan penempatan detail 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* memudahkan perakitan/integrasi dengan komponen aplikasi lainnya tanpa harus memperhatikan detail proses. Desainer hanya berkonsentrasi pada bentuk dan tampilan antarmuka pemakai (user interface).
4. Penggunaan Controller memungkinkan untuk perubahan proses tanpa harus mengganggu antarmuka pemakai. Detail 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 [4]:

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.

## 2.14 *Framework* Laravel 4.2

### 2.14.1. Sejarah *Framework* Laravel

Sejarah *Framework* Laravel di buat oleh Tylor Otwell, Proyek Laravel dimulai pada april 2011. Awal mula proyek ini dibuat, karena Otwell sendiri tidak menemukan *Framework* yang *up-to-date* dengan versi PHP. Mengembangkan *Framework* yang sudah ada juga bukan ide yang bagus, karena keterbatasan sumber daya. Dikarenakan beberapa keterbatasan tersebut, Otwell membuat sendiri *Framework* dengan nama Laravel. Oleh karean itu, Laravel mensyaratkan PHP versi 5.3 ke atas. Laravel di rilis dibawah lisensi MIT. Laravel dibangun dengan konsep MVC (*Model-View -Controller*), Kemudian Laravel dilengkapi juga command line tools yang bernama “*Artisan*” yang digunakan untuk packaging bundle dan instalasi bundle melalui command prompt [3].

### 2.14.2. Kelebihan *Framework* Laravel 4.2

Ada beberapa alasan yang menjadi dasar pertimbangan mengapa menggunakan *Framework* Laravel [2] :

- 1) *Expresif*, Ketika melihat sintaks laravel, seorang programmer diharapkan akan langsung tahu kegunaan dari sintaks tersebut meskipun belum pernah mempelajarinya apalagi menggunakannya.
- 2) *Simple*, Laravel begitu simple karena dengan adanya *Eloquent* ORM.

- 3) *Accessibile*, Laravel pun dibuat dengan dokumentasi yang lengkap setiap rilis versi terbarunya.

### 2.14.3. Fitur *Framework* Laravel 4.2

*Framework* Laravel 4.2 memiliki beberapa fitur unggulan, antara lain [3]:

- 1) *Bundles*  
Yaitu sebuah fitur dengan sistem pengemasan modular dan berbagai *bundle* telah tersedia untuk digunakan dalam aplikasi anda.
- 2) *Eloquent ORM*  
Merupakan penerapan PHP lanjut dari pola “*Active Record*” menyediakan metode internal untuk mengatasi kendala hubungan antara objek database. Penggunaan *query* Laravel fluent didukung *Eloquent*.
- 3) *Application Logic*  
Merupakan bagian dari aplikasi yang dikembangkan, baik menggunakan *Controllers* maupun sebagai bagian dari deklarasasi *Route*. Sintaks yang digunakan untuk mendefinisikannya mirip dengan yang digunakan *Framework Sinatra*.
- 4) *Reverse Routing*  
Mendefinisikan hubungan antara *Link* dan *Route*, Sehingga jika suatu saat ada perubahan pada *route* secara otomatis akan tersambung dengan *link* yang relevan. Ketika *Link* yang dibuat dengan menggunakan nama-nama dari *route* yang ada, secara otomatis Laravel akan membuat *URI* yang sesuai.
- 5) *Restful Controller*  
Memberikan sebuah option (pilihan) untuk memisahkan logika dalam melayani *HTTP GET* dan permintaan *POST*.
- 6) *Class Auto Loading*  
Menyediakan otomatisasi loading untuk *Class -Class* PHP, tanpa membutuhkan pemeriksaan manual terhadap jalur masuknya. Fitur ini mencegah loading yang tidak perlu.
- 7) *View Composers*  
Kode unit logical yang dapat dijalankan ketika sebuah *View* di-load.

8) *Ioc Container*

Memungkinkan untuk objek baru yang dihasilkan dengan mengikuti prinsip control pembalik, dengan pilihan contoh dan referensi dari objek baru sebagai Singletons.

9) *Migrations*

Menyediakan versi sistem control untuk skema database, sehingga memungkinkan untuk menghubungkan perubahan adalah basis kode aplikasi dan keperluan yang dibutuhkan dalam merubah tata letak database, mempermudah dalam penempatan dan memperbaiki aplikasi.

10) *Unit Testing*

Mempunyai peran penting dalam *Framework* Laravel , dimana unit testing ini mempunyai banyak tes untuk mendekteksi dan mencegah regresi.

11) *Automatic Pagination*

Menyederhanakan tugas dari penerapan halaman, mengartikan penerapan yang manual dengan metode otomatis yang terintergrasi ke Laravel.

## 2.15 Database

Database adalah koleksi berbagai data secara logis, terkait dan deskripsi yang dirancang untuk memenuhi kebutuhan Informasi dari suatu organisasi. Aplikasi database hanyalah sebuah program yang berinteraksi dengan database di beberapa titik dalam pelaksanaannya. Keuntungan yang didapat dengan menggunakan database adalah mengurangi pengulangan data. Database tidak hanya menyimpan data operasional, namun juga deskripsi data-data tersebut.

### 2.15.1. Database Management System (DBMS)

Menurut Conolly & Begg , DBMS merupakan sistem software yang dapat memungkinkan user untuk mendefinisikan, membuat, memelihara dan mengontrol akses terhadap database. DBMS menyediakan fasilitas-fasilitas berikut ini :

1. *Data Definition Language (DDL)*

Memungkinkan user untuk menspesifikasi tipe data, struktur dan constraint data yang akan disimpan ke dalam database.

## 2. *Data Manipulation Language* (DML)

Memungkinkan user untuk insert, update, delete dan menerima data dari database. Menggunakan suatu *query language*. *Query language* yang paling umum digunakan yaitu *Structured Query Language* (SQL).

## 3. Menyediakan akses kontrol terhadap database. Misalnya :

### a) Sistem Keamanan

Mencegah user yang tidak diberi kuasa untuk mengakses database.

### b) Integritas Sistem

Konsistensi data yang disimpan.

### c) Sistem Kontrol Persetujuan

Memungkinkan shared access database

## 2.16 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).

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* diidentifikasi 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  $A \twoheadrightarrow B$ , dimana A adalah *candidate key* dari relasi.
7. *Fifth Normal Form (5NF)*, sebuah relasi jika dan hanya jika setiap join dependency  $(R_1, R_2, \dots, R_n)$  di dalam relasi R, setiap proyeksi mencakup *candidate key* dari relasi asli.

### 2.17 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread* dan multi-user. MySQL adalah *Relational Database Management System (RDBMS)* yang didistribusikan secara gratis dibawah lisensi GPL (*General Public Lisenca*). MySQL sebenarnya merupakan turunan dari SQL (*Structured Query Laguage*). SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Sebagai database server, MySQL dapat dikatakan lebih unggul dibandingkan database server lainnya dalam *query* data. Hal ini terbukti untuk *query* yang dilakukan oleh single user, kecepatan *query* MySQL bisa sepuluh kali lebih cepat dari PostgreSQL dan lima kali lebih cepat dibandingkan interbase. Selain itu MySQL memiliki beberapa keistimewaan, antara lain :

1. *Portability*. MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac OS X Server, Solaris, Amiga, dan lainnya.
2. *Open Source*, MySQL didistribusikan secara gratis dibawah lisensi GPL.
3. *Multiuser*. MySQL dapat digunakan oleh beberapa user dalam waktu bersamaan tanpa mengalami masalah atau konflik.
4. *Performance Tuning*. MySQL memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana.
5. *Column Types*. MySQL memiliki tipe kolom yang sangat kompleks seperti *signed/unsigned integer, float, double, char, text, date, timestamp*, dan lain-lain.
6. *Command dan Functions*. MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah *select* dan *Where* dalam *query*.
7. *Security*. MySQL memiliki beberapa lapisan sekuritas seperti level subnetmask, nama host, dan ijin akses user dengan sistem perizinan yang mendetail dan password terenkripsi.
8. *Scalability dan Limits*. MySQL mampu menangani database dalam skala besar, dengan jumlah records lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris.
9. *Conectivity*. MySQL dapat melakukan koneksi dengan client menggunakan protokol TCP/IP, Unix Soket (UNIX), atau Named Pipes (NT).
10. Struktur Tabel. MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani *ALTER TABLE*, dibandingkan database lainnya semacam PostgreSQL ataupun Oracle.

## 2.18 Pengujian Perangkat Lunak

### 2.18.1 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 seberapa 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) [11].

### 2.18.2 *User Acceptance Testing (UAT)*

Pada jenis pengujian ini, perangkat lunak akan diserahkan kepada pengguna untuk mengetahui apakah perangkat lunak memenuhi harapan pengguna dan bekerja seperti yang diharapkan. Pada pengembangan perangkat lunak, user acceptance test (UAT), juga disebut pengujian *beta (beta testing)*, pengujian aplikasi (*application testing*), dan pengujian pengguna akhir (*end user testing*) adalah tahapan pengembangan perangkat lunak diuji pada “dunia nyata” yang dimaksudkan oleh pengguna. UAT dapat dilakukan dengan in-house testing dengan membayar relawan atau subjek pengujian menggunakan perangkat lunak atau biasanya mendistribusikan perangkat lunak secara luas dengan melakukan pengujian versi yang tersedia secara gratis untuk diunduh melalui web. Pengalaman awal pengguna akan diteruskan kembali kepada para pengembang yang membuat perubahan sebelum akhirnya melepaskan perangkat lunak komersial.

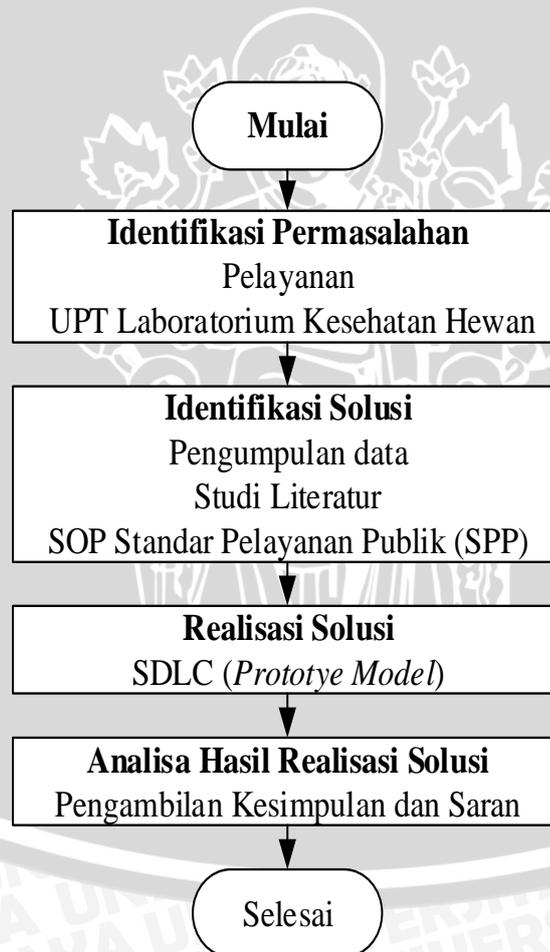
Tabel 2.3 Contoh Form UAT

KUISIONER UAT						
Alternatif Jawaban:						
STM = Sangat Tidak Mudah		S = Sedang		SM = Sangat Mudah		
TM = Tidak Mudah		M = Mudah				
NO	PERTANYAAN	JUMLAH				
		STM	TM	S	M	SM
<b>1</b>	<b><i>Learnability</i></b>					
1.1	Apakah tampilan menu dalam aplikasi mudah dikenali?					
1.2	Apakah aplikasi mudah dioperasikan?					
1.3	Apakah aplikasi dapat dengan mudah dipelajari?					
1.4	Apakah informasi yang disediakan oleh aplikasi ini mudah dimengerti?					
<b>2</b>	<b><i>Efficiency</i></b>					
2.1	Apakah dengan diterapkannya aplikasi ini dapat membuat pengelolaan pelayanan lebih efisien?					
2.2	Apakah pencarian dokumen dengan menggunakan aplikasi menjadi lebih mudah?					
2.3	Apakah penggunaan menu atau fitur aplikasi mudah digunakan?					
<b>3</b>	<b><i>Memorability</i></b>					
3.1	Apakah langkah-langkah mengelola pelayanan dengan aplikasi ini mudah diingat?					
3.2	Apakah data (informasi dan file) tersimpan dengan baik?					
3.3	Apakah penggunaan aplikasi mudah diingat setelah lama tidak menggunakannya?					
<b>4</b>	<b><i>Errors</i></b>					
4.1	Apakah dapat dengan mudah menghindari kesalahan dalam menggunakan aplikasi?					
4.2	Apakah pesan kesalahan atau peringatan dalam penggunaan aplikasi tampil?					
<b>5</b>	<b><i>Satisfaction</i></b>					
5.1	Apakah aplikasi mempunyai fungsi dan kapabilitas (kemampuan) sesuai yang diharapkan?					
5.2	Aplikasi membuat pengelolaan pelayanan menjadi efektif?					
5.3	Tata letak halaman (layout) dan informasi dalam layar halaman sangat jelas?					
5.4	Apakah aplikasi nyaman digunakan?					
5.5	Secara keseluruhan, apakah penggunaan aplikasi ini memuaskan?					

### BAB III

## METODOLOGI PENELITIAN

Pada bab ini dijelaskan metodologi yang akan ditempuh dalam penelitian. Metode penelitian yang digunakan adalah studi kasus dan pengembangan karena penelitian lebih difokuskan pada pengembangan suatu aplikasi dengan studi kasus di UPT Laboratorium Kesehatan Hewan Malang. Langkah-langkah yang dilakukan 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 berikut :



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.

### 3.2 Identifikasi Solusi

Identifikasi solusi dilakukan dengan cara pengumpulan data sehingga diperoleh sebuah solusi untuk mengatasi permasalahan di organisasi. Metode yang digunakan untuk proses pengumpulan data dalam penelitian ini adalah sebagai berikut:

#### a. Teknik Observasi

Teknik observasi merupakan metode mengumpulkan data dengan mengamati langsung di lapangan. Dalam penelitian ini peneliti menggunakan teknik observasi Non Partisipan, dimana peneliti akan mengamati proses bisnis alur pelayanan laboratorium 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 pelayanan laboratorium yang sedang berjalan, serta kebutuhan perangkat lunak dan perancangan sistem yang diharapkan.

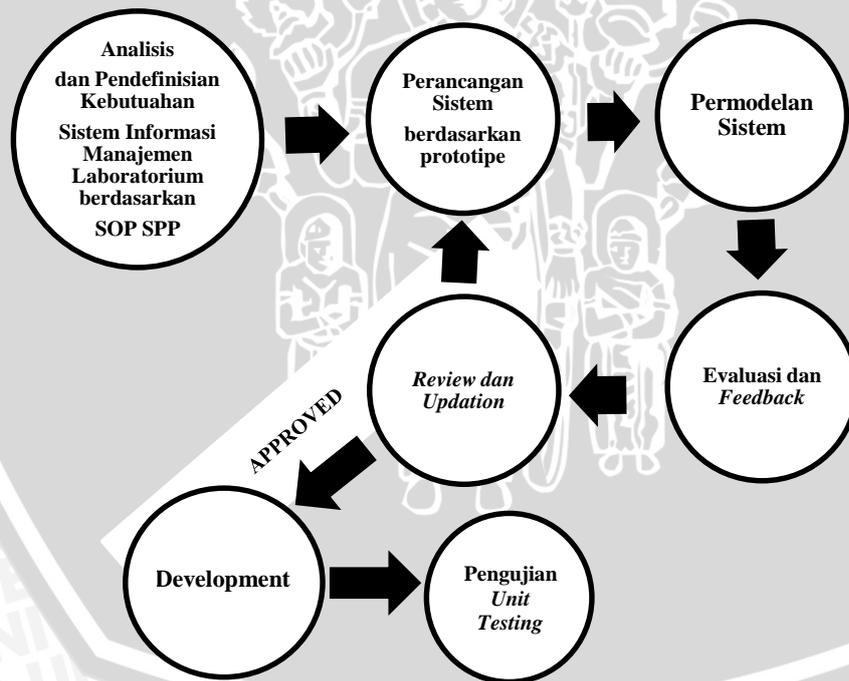
#### c. Teknik Dokumentasi

Data yang akan diambil melalui teknik ini yaitu data-data yang berhubungan dengan alur proses manajemen pelayanan laboratorium di UPT Laboratorium Kesehatan Hewan Malang seperti data pelayanan, pemeriksaan, penyidikan, diagnosis penyakit hewan, dokumen hasil pengujian dan sebagainya serta melakukan Studi literatur untuk pengumpulan serta pemahaman materi dan Informasi . Literatur yang diperlukan dalam penyusunan skripsi meliputi :

1. SOP SPP Pelayanan Di UPT Laboratorium Kesehatan Hewan Malang
2. Sistem Infomasi Manajemen
3. Aplikasi berbasis web
4. PHP dengan konsep MVC
5. *Framework* Laravel 4.2

6. MySQL
7. Apache
8. Rekayasa Perangkat Lunak

Dari proses identifikasi permasalahan dan pengumpulan data, maka dapat diambil kesimpulan bahwa kantor UPT Laboratorium Kesehatan Hewan Malang membutuhkan sebuah sistem yang dapat memajemen pengelolaan data pelayanan laboratorium yang lebih terstruktur dan sistematis. Mengacu pada permasalahan tersebut, pembuatan sebuah perangkat lunak/ software untuk manajemen data pelayanan yang mengadopsi standar konsep manajemen dapat menjadi salah satu solusi alternatif. Salah satu standar konsep manajemen pelayanan adalah Standar Operasional Prosedure Standar Pelayanan Publik (SOP SPP). Adapun tahapan pembuatan sebuah perangkat lunak dengan konsep prototipe (*Prototyping Model*) dapat dilihat Pada Gambar 3.2 berikut :



**Gambar 3.2 Diagram Blok Tahap Pembuatan Software**

### 3.2.1 Analisis dan Pendefinisian Kebutuhan

Analisis kebutuhan dilakukan dengan menentukan kebutuhan apa saja yang diperlukan sesuai dengan data yang didapatkan pada saat proses pengumpulan data yang telah dilakukan sebelumnya kemudian merancanginya dalam bentuk sistem Informasi manajemen laboratorium. Proses analisis kebutuhan sistem dalam penelitian ini meliputi :

1. Pemodelan *Usecase* serta *Activity* Diagram untuk proses bisnis pengimplementasi sistem
2. 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:

#### a) Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang terkait dengan fungsi produk, misalnya sistem harus mampu mencetak laporan, sistem harus mampu menampilkan grafik, dan lain-lain. Sistem Informasi manajemen laboratorium ini memiliki fitur antara lain sebagai berikut :

- 1) Rekam Diagnosa data Sampel sebagai Informasi riwayat pengujian data Sampel.
- 2) Memberikan Informasi batas waktu penyelesaian pegujian dan hasil pengujian data Sampel.
- 3) Memberikan notifikasi setiap alur prosesnya sehingga mempermudah pengelolaan dan mengetahui runtutan proses pelyanan.
- 4) Monitoring alur proses pelayanan laboratorium untuk mengetahui dimana posisi dokumen.

#### b) Kebutuhan Pengembangan (*Development Requirement*)

Development requirement adalah kebutuhan yang terkait tools untuk pengembangan sistem Informasi baik perangkat keras maupun perangkat lunak. Kebutuhan pengembangan penyusunan sistem ini adalah sebagai berikut:

### 1. Spesifikasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan dalam proses pembuatan Sistem Informasi Manajemen Laboratorium (SIMLAB) dijelaskan pada tabel berikut.

**Tabel 3.2 Spesifikasi Perangkat Keras**

Nama Komponen	Spesifikasi
Processor	Intel(R) Core(TM) i5-4210U CPU @ 1.70GHz
RAM	4 GB DDR3
Harddisk	500 GB SATA3
System Manufacturer	ASUSTeK COMPUTER INC
VGA Card	NVIDIA GeForce 820M

### 2. Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam proses pembuatan Sistem Informasi Manajemen Laboratorium (SIMLAB) dijelaskan pada tabel berikut.

**Tabel 3.3 Spesifikasi Perangkat Lunak**

Perangkat Lunak	Spesifikasi
Sistem Operasi	Windows 8.1 Pro 64-bit
Bahasa Pemrograman	PHP extension: mysql ( <i>Framework</i> Laravel 4.2)
Editor Tools	Dreamweaver Adobe CS 6

#### 3.2.2 Perancangan Sistem

Perancangan sistem yaitu membangun *prototyping* dengan membuat perancangan sementara yang berfokus pada penyajian kepada pelanggan. Untuk mengatasi ketidakserasian antara pelanggan dan pengembang, maka dibutuhkan kerjasama yang baik diantara keduanya sehingga pengembang akan mengetahui dengan benar apa yang diinginkan pelanggan dengan tidak mengesampingkan segi-segi teknis dan pelanggan akan mengetahui proses-proses dalam menyelesaikan sistem yang diinginkan. Dengan demikian akan menghasilkan sistem sesuai dengan jadwal waktu penyelesaian yang telah ditentukan.

### 3.2.3 Pemodelan Sistem

Tahap permodelan sistem ini terdiri dari pemodelan proses bisnis, *Usecase* Diagram, *Activity* Diagram, *Sequence* Diagram, *Class* Diagram dan struktur database.

#### 1. Pemodelan Bisnis Baru

Setelah penerapan Sistem Informasi Manajemen Pelayanan Laboratorium, diharapkan manajemen / pengelolaan data pelayanan laboratorium yang ada di Kantor UPT Laboratorium Kesehatan Hewan Malang lebih terstruktur, meningkatkan efektivitas dan efisiensi organisasi.

#### 2. Pemodelan *Usecase* Diagram

Tahap perancangan pertama yaitu perancangan *Usecase* Diagram untuk mendeskripsikan interaksi antara satu atau lebih aktor dengan SIMLA, Serta untuk mengetahui fitur dan fungsi dari SIMLAB yang akan dibangun.

#### 3. Pemodelan *Activity* Diagram

Perancangan *Activity* Diagram untuk menggambarkan aliran kerja (workflow) atau aktivitas atau proses bisnis yang ada pada SIMLAB. *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 :

- Proses bisnis SIMLAB dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan berdasarkan permintaan dari pihak UPT Laboratorium Kesehatan Hewan Malang.
- Urutan atau pengelompokan tampilan dari sistem / user interface dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan pada SIMLAB.
- Rancangan menu yang ditampilkan pada SIMLAB.

#### 4. Pemodelan *Sequence* Diagram

*Sequence* Diagram menggambarkan kelakuan objek pada *Usecase* dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek pada SIMLAB. Oleh karena itu objek-objek yang terlibat disesuaikan dengan *Usecase* Diagram perancangan SIMLAB. Banyaknya

Diagram *sequence* yang harus digambar adalah minimal sebanyak pendefinisian *Usecase* yang memiliki proses sendiri atau yang penting semua *Usecase* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada Diagram *sequence* SIMLAB.

#### 5. **Pemodelan Class Diagram**

Perancangan *Class* Diagram ini dilakukan untuk merancang *Class* yang nantinya akan diimplementasikan pada *Framework* Laravel 4.2 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 Laravel *Framework* 4.2.

#### 6. **Pemodelan Entity Relationship Diagram**

Perancangan Entity Relationship Diagram (ERD) digunakan untuk mendefinisikan perancangan basis data pada SIMLAB. Tujuannya untuk memudahkan pihak UPT Laboratorium Kesehatan Hewan Malang dalam memahami basis data yang digunakan dalam SIMLAB.

#### 3.2.4 Evaluasi dan *Feedback*

Evaluasi aplikasi dilakukan dengan mengidentifikasi kebutuhan dan perancangan desain perangkat lunak yang sudah di rancang sebelumnya. Pihak UPT Laboratorium Kesehatan Hewan Malang mengevaluasi apakah SIMLAB yang sudah jadi sudah sesuai dengan yang diharapkan secara fungsional dan desain perencanaan sebelum dibuat. apabila belum sesuai maka akan ada *feedback* dari pihak UPT Laboratorium Kesehatan Hewan Malang sehingga pengembang bisa mendaur ulang SIMLAB tersebut berdasarkan permintaan kebutuhan yang akan dipenuhi.

#### 3.2.5 *Review* dan *Updation*

*Review* dilakukan dari desain perancangan SIMLAB apakah sudah memenuhi kebutuhan yang diperlukan atau masih ada pengembangan selanjutnya kepada pihak UPT Laboratorium Kesehatan Hewan Malang. Apabila sudah memenuhi kriteria kebutuhan maka dapat dilakukan development sistem. Apabila belum sesuai dengan kebutuhan dari pihak UPT Laboratorium Kesehatan Hewan

Malang dapat melakukan *Updation* atau perubahan berupa penambahan yang disarankan sesuai dengan perencanaan yang telah dibuat.

### 3.2.6 Development

Development yaitu melakukan pembuatan SIMLAB berdasarkan prototype yang telah oleh pihak UPT Laboratorium Kesehatan Hewan Malang dengan pengembang. Sehingga kebutuhan sistem akan terpenuhi sesuai dengan permintaan UPT Laboratorium Kesehatan Hewan Malang terhadap pembuatan sistem.

### 3.2.7 Pengujian Unit Testing.

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

#### 1. *Functional Test*

Pengujian fungsional digunakan untuk menguji secara fungsionalitas apakah SIMLAB sudah berjalan dengan permintaan proses bisnis yang diajukan oleh pihak UPT Laboratorium Kesehatan Hewan Malang dengan menggunakan cek list fungsional.

#### 2. *User Acceptance Test*

User Acceptance Test yang dilakukan oleh pihak UPT Laboratorium Kesehatan Hewan Malang dari sistem tersebut untuk memastikan penilaian tentang usability pada SIMLAB berdasar nilai persentase tiap kategori pengujian, yaitu *learnbility, efficiency, memorability, errors, dan satisfaction*.

### 3.3 Realisasi Solusi

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

perancangan sistem. Implementasi SIMLAB dilakukan dengan menggunakan bahasa pemrograman PHP dengan *Framework* Laravel 4.2. Bootstrap digunakan sebagai *Framework* desain tampilan, MySql sebagai basis data, dan SOP SPP sebagai acuan dari alur proses manajemen pelayanan laboratorium di dalam perangkat lunak yang dikembangkan.

### 3.4 Analisa Hasil Realisasi Solusi

Analisa hasil realisasi solusi merupakan proses pengambilan kesimpulan yang dilakukan setelah semua tahapan perancangan sistem, implementasi sistem dan pengujian sistem telah selesai dilakukan. Analisa hasil realisasi solusi didapatkan dengan melakukan pengujian analisis dan evaluasi pada pengimplementasian SIMLAB yang telah dibangun terhadap dampak yang dirasakan oleh Kantor UPT Laboratorium Kesehatan Hewan Malang dan kesesuaian SOP SPP yang digunakan.



## BAB IV

### ANALISIS DAN PERANCANGAN

#### 4.1 Gambaran Umum

Sistem Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang ini merupakan sistem yang dibangun untuk mendukung proses bisnis pelayanan dan efisiensi kinerja organisasi yang dilakukan di kantor UPT Laboratorium Kesehatan Hewan Malang. Modul sistem disesuaikan dengan standast operasional prosedur kerja setiap bagian yang ada di kantor UPT Laboratorium Kesehatan Hewan Malang, meliputi Kepegawaian, Pelayanan (Kesmavet dan Keswan), Laboratorium, dan Administratif. Setiap aktor yakni karyawan atau staf memiliki hak akses yang berbeda terhadap sistem tergantung pada jabatan yang dimiliki. Modul-modul yang ada pada sistem terintegrasi menjadi satu dalam pelayanan terpadu. Fitur-fitur yang disediakan akan menunjang kebutuhan pelayanan dan membantu kinerja dari karyawan sehingga lebih mudah, cepat dan optimal dalam melakukan kegiatannya.

#### 4.2 *Elisitasi*

Elisitasi ini dilakukan dengan harapan agar proses menjadi lebih cepat dan terarah. Dari hasil penelitian yang dilakukan dengan cara wawancara kepada pihak Kantor UPT Laboratorium Kesehatan Hewan Malang, yaitu Kepala UPT dan Kepala Pelayanan, maka diperoleh hasil elisitasi kebutuhan perangkat lunak yang akan digunakan sebagai dasar pembangunan dan pengembangan sebuah sistem manajemen laboratorium yang mengacu pada standar pelayanan publik (SPP). Berikut kebutuhan fungsional sistem yang telah diajukan oleh Kepala Pelayanan 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 Sampel berdasarkan kategori pelayanan Kesmavet dan Keswan, Sampel diambil berdasarkan Bahan Asal Hewan

(BAH) dan Hasil Bahan Asal Hewan / Olahan (HBAH), Metode Pengujian di sesuaikan dengan pelayanan yang tersedia.

- 3) Sistem dapat mengelola data pendaftaran member yang melakukan pengujian baik secara personal atau permintaan dari lembaga / instansi.
- 4) Sistem dapat mengelola data kaji ulang permintaan pengujian, penerimaan contoh Sampel, surat permohonan pengujian serta surat hasil pengujian.
- 5) Sistem dapat mengelola data diagnosa Sampel dan rekam diagnosa setiap member yang telah melakukan pengujian beserta pembuatan laporan berupa PDF.
- 6) Sistem dapat mengelola data rekap laporan berdasarkan kaji ulang permintaan, surat permintaan pengujian, hasil pengujian, rekap laporan pelayanan berdasarkan periode setiap daerah berupa PDF.
- 7) Sistem dapat mengelola data arsip, mengatur klasifikasi dan kategori arsip sesuai dengan kebijakan manajemen arsip yang berlaku di UPT Laboratorium Kesehatan Hewan Malang.

### **4.3 Spesifikasi Sistem**

Tahap ini akan menjelaskan analisa terhadap sistem yang sedang berjalan dan sistem yang akan dibuat perancangannya tentang pelaksanaan yang berkaitan dengan sistem Informasi manajemen laboratorium untuk pelayanan yang ada di UPT Laboratorium Kesehatan Hewan Malang.

#### **4.3.1 Gambaran Sistem Lama**

Berdasarkan hasil wawancara dengan pihak UPT Laboratorium Kesehatan Hewan Malang, dapat diidentifikasi bahwa lembaga tersebut belum memiliki suatu media yang memenuhi kebutuhan pelayanan dalam menjalankan tugasnya. Proses bisnis yang berjalan keseluruhan masih manual belum terkomputerisasi sepenuhnya sehingga sering mengalami kesulitan dalam melakukan tugas pelayanan. Kondisi yang ada saat ini antara lain :

1. Administratif melakukan pendaftaran bagi member personal dan instansi/lembaga yang melakukan pelayanan pengujian Sampel.

2. Administratif melakukan pendaftaran berdasarkan formulir kaji ulang permintaan kebagian pelayanan sesuai dengan permintaan member.
3. Petugas Pelayanan mengklasifikasikan pelayanan keswan atau kesmavet berdasarkan formulir penerimaan Sampel dari pendaftaran kaji ulang permintaan dari administratif dan membuat surat permintaan pengujian kepihak laboratorium.
4. Laboran menerima Sampel dan surat penerimaan Sampel untuk dilakukan pengujian / diagnosa Sampel.
5. Laboran memberikan hasil pengujian kepihak pelayanan untuk dibuatkan surat hasil pelayanan.
6. Petugas Pelayanan mengkatagorikan rekam pengujian pada riwayat member dan membuat surat hasil pelayanan.
7. Petugas pelayanan memberikan surat hasil pelayanan kepada administratif.
8. Kelengkapan dan keabsahan pelayanan diserahkan kepada kepala UPT Laboratorium Kesehatan Hewan sebelum diserahkan kepada member.

#### 4.4 Spesifikasi Kebutuhan Sistem

Berikut ini adalah spesifikasi kebutuhan pada Sistem Informasi Manajemen Laboratorium sebagai manajemen data pelayanan di UPT Laboratorium Kesehatan Hewan Malang.

##### 4.4.1. Identifikasi Aktor.

Identifikasi Aktor sistem terdiri yang masing-masing akan dideskripsikan sebagai berikut :

**Tabel 4.1 Identifikasi Aktor.**

No	Aktor	Deskripsi
1	<b>Super Admin</b>	Super Admin adalah administrator sistem Informasi yang dapat mengelola semua fitur / fungsionalitas sistem, meliputi kelola Data User, kelola data karyawan, kelola data jabatan, kelola data jabatan fungsional, kelola member, kelola lembaga, kelola wilayah, kelola data bagian, kelola Sampel, kelola data uji Sampel, kelola data hasil Sampel, kelola data hewan, kelola data ras, kelola data pelayanan, dan validasi rekam diagnosa.

2	<b>Admin Kepegawaian</b>	adalah petugas yang dapat mengelola fitur / fungsionalitas, meliputi kelola data karyawan, kelola data bagian, kelola data jabatan, dan kelola data jabatan fungsional.
3	<b>Administratif</b>	Administratif adalah petugas yang dapat mengelola fitur/fungsionalitas, meliputi melihat data karyawan, kelola data member, kelola data lembaga, kelola data wilayah dan kelola data pelayanan.
4	<b>Kepala UPT</b>	Kepala UPT merupakan pimpinan di UPT Laboratorium Kesehatan Hewan Malang. Aktor memiliki kewenangan untuk validasi data pelayanan.
5	<b>Admin Keswan</b>	Keswan adalah petugas yang dapat mengelola fitur / fungsionalitas, meliputi kelola data hewan, kelola data ras, kelola data Sampel, kelola data uji Sampel, kelola data hasil uji dan validasi data pelayanan.
6	<b>Admin Kesmavet</b>	Kesmavet adalah petugas yang dapat mengelola fitur / fungsionalitas, meliputi kelola data hewan, kelola produksi pangan asal hewan, kelola data ras, kelola data Sampel, kelola data uji Sampel, kelola data hasil uji dan validasi data pelayanan.
7	<b>Laboran</b>	Laboran adalah petugas yang dapat mengelola fitur / fungsionalitas, meliputi kelola data laboratorium dan kelola data rekam diagnosa.

#### 4.4.2. Spesifikasi Kebutuhan Fungsional Sistem

Daftar kebutuhan sistem terdiri dari keutuhan fungsional dan keutuhan non-fungsional yang masing-masing akan dispesifikasikan sebagai berikut :

**Tabel 4.2 Spesifikasi Kebutuhan Fungsional Sistem**

No	No SRS	Kebutuhan	Aktor
1	SRS_1_01	Login	Super Admin, Kepala UPT, Admin Kepegawaian, Administratif, Admin Keswan, Admin Kesmavet dan Laboran

2	SRS_1_02	Logout	Super Admin, Kepala UPT, Admin Kepegawaian, Administratif, Admin Keswan, Admin Kesmavet dan Laboran
3	SRS_2_01	Melihat Data Profile user	Super Admin, Kepala UPT, Admin Kepegawaian, Administratif, Admin Keswan, Admin Kesmavet dan Laboran
4	SRS_3_01	Melihat Data Karyawan	Super Admin, Kepala UPT dan Admin Kepegawaian
5	SRS_3_02	Menambah Data Karyawan	Super Admin dan Admin Kepegawaian
6	SRS_3_03	Mengubah Data Karyawan	Super Admin dan Admin Kepegawaian
7	SRS_3_04	Menghapus Data Karyawan	Super Admin dan Admin Kepegawaian
8	SRS_3_05	Menambah Data Jabatan	Super Admin dan Admin Kepegawaian
9	SRS_3_06	Mengubah Data Jabatan	Super Admin dan Admin Kepegawaian
10	SRS_3_07	Menghapus Data Jabatan	Super Admin dan Admin Kepegawaian
11	SRS_3_08	Menambah Data Jabatan Fungsional	Super Admin dan Admin Kepegawaian
12	SRS_3_09	Mengubah Data Jabatan Fungsional	Super Admin dan Admin Kepegawaian
13	SRS_3_10	Menghapus Data Jabatan Fungsional	Super Admin dan Admin Kepegawaian

14	SRS_3_11	Menambah Data Bagian	Super Admin dan Admin Kepegawaian
15	SRS_3_12	Mengubah Data Bagian	Super Admin dan Admin Kepegawaian
16	SRS_3_13	Menghapus Data Bagian	Super Admin dan Admin Kepegawaian
17	SRS_4_01	Menambah Data Level	Super Admin
18	SRS_4_02	Mengubah Data Level	Super Admin
19	SRS_4_03	Menghapus Data Level	Super Admin
20	SRS_4_04	Melihat Data User	Super Admin
21	SRS_4_05	Menambah Data User	Super Admin
22	SRS_4_06	Mengubah Data User	Super Admin
23	SRS_4_07	Menghapus Data User	Super Admin
24	SRS_4_08	Validasi Data User	Super Admin
25	SRS_5_01	Menambah Data Provinsi, Kota, Kecamatan dan Desa	Super Admin dan Administratif
26	SRS_5_02	Mengubah Data Provinsi, Data Provinsi, Kota, Kecamatan dan Desa	Super Admin dan Administratif
27	SRS_5_03	Menghapus Data Provinsi	Super Admin dan Administratif
28	SRS_5_04	Menambah Data Kota	Super Admin dan Administratif
29	SRS_5_05	Mengubah Data Kota	Super Admin dan Administratif
30	SRS_5_06	Menghapus Data Kota	Super Admin dan Administratif
31	SRS_5_07	Menambah Data Kecamatan	Super Admin dan Administratif
32	SRS_5_08	Mengubah Data Kecamatan	Super Admin dan Administratif

33	SRS_5_09	Menghapus Data Kecamatan	Super Admin dan Administrasitif
34	SRS_5_10	Menambah Data Desa	Super Admin dan Administrasitif
35	SRS_5_11	Mengubah Data Desa	Super Admin dan Administrasitif
36	SRS_5_12	Menghapus Data Desa	Super Admin dan Administrasitif
37	SRS_6_01	Menambah Data Hewan	Super Admin, Admin Keswan dan Admin Kesmavet
38	SRS_6_02	Mengubah Data Hewan	Super Admin, Admin Keswan dan Admin Kesmavet
39	SRS_6_03	Menghapus Data Hewan	Super Admin, Admin Keswan dan Admin Kesmavet
40	SRS_6_04	Menambah Data Bangsa	Super Admin, Admin Keswan dan Admin Kesmavet
41	SRS_6_05	Mengubah Data Bangsa	Super Admin, Admin Keswan dan Admin Kesmavet
42	SRS_6_06	Menghapus Data Bangsa	Super Admin, Admin Keswan dan Admin Kesmavet
43	SRS_6_07	Menambah Data Ras	Super Admin, Admin Keswan dan Admin Kesmavet
44	SRS_6_08	Mengubah Data Ras	Super Admin, Admin Keswan dan Admin Kesmavet

45	SRS_6_09	Menghapus Data Ras	Super Admin, Admin Keswan dan Admin Kesmavet
46	SRS_7_01	Menambah Data Sampel	Super Admin, Admin Keswan dan Admin Kesmavet
47	SRS_7_02	Mengubah Data Sampel	Super Admin, Admin Keswan dan Admin Kesmavet
48	SRS_7_03	Menghapus Data Sampel	Super Admin, Admin Keswan dan Admin Kesmavet
49	SRS_7_04	Menambah Data Metode Pengujian	Super Admin, Admin Keswan dan Admin Kesmavet
50	SRS_7_05	Mengubah Data Metode Pengujian	Super Admin, Admin Keswan dan Admin Kesmavet
51	SRS_7_06	Menghapus Data Metode Pengujian	Super Admin, Admin Keswan dan Admin Kesmavet
52	SRS_7_07	Menambah Data Hasil Pengujian	Super Admin, Admin Keswan dan Admin Kesmavet
53	SRS_7_08	Mengubah Data Pengujian	Super Admin, Admin Keswan dan Admin Kesmavet
54	SRS_7_09	Menghapus Data Pengujian	Super Admin, Admin Keswan dan Admin Kesmavet
55	SRS_8_01	Kelola Data Diagnosa	Super admin dan Laboran

56	SRS_8_02	Melihat Data Rekam Diagnosa	Super Admin, Kepala UPT, Administratif, Admin Keswan, Admin Kesmavet dan Laboran
57	SRS_8_03	Validasi Data Diagnosa	Super admin dan Laboran
58	SRS_9_01	Melihat Data Pendaftaran pelayanan	Super admin dan Administratif
59	SRS_9_02	Menambah Data Pendaftaran pelayanan	Super admin dan Administratif
60	SRS_9_03	Mengubah Data Pendaftaran pelayanan	Super admin dan Administratif
61	SRS_9_04	Menghapus Data Pendaftaran pelayanan	Super admin dan Administratif
62	SRS_9_05	Kelola Data Daftar Penerimaan Sampel	Super Admin, Admin Keswan dan Admin Kesmavet
63	SRS_9_06	Menambah Data laboratorium	Super Admin, Admin Keswan dan Admin Kesmavet
64	SRS_9_07	Mengubah Data laboratorium	Super Admin, Admin Keswan dan Admin Kesmavet
65	SRS_9_08	Menghapus Data Laboratorium	Super Admin, Admin Keswan dan Admin Kesmavet
66	SRS_9_09	Validasi Data Pelayanan	Super Admin dan Kepala UPT
67	SRS_9_10	Validasi Data Kaji Ulang Sampel	Sper Admin dan Administratif
68	SRS_9_11	Validasi Data Penerimaan Sampel	Super Admin, Admin Keswan dan Admin Kesmavet

69	SRS_10_01	Menambah Data Member	Super Admin dan Administrasitif
70	SRS_10_02	Mengubah Data Member	Super Admin dan Administrasitif
71	SRS_10_03	Menghapus Data Member	Super Admin dan Administrasitif
72	SRS_10_04	Menambah Data Instansi	Super Admin dan Administrasitif
73	SRS_10_05	Mengubah Data Instansi	Super Admin dan Administrasitif
74	SRS_10_06	Menghapus Data Instansi	Super Admin dan Administrasitif
75	SRS_10_07	Cetak Laporan	Super Admin , Administratif, Admin Keswan dan Admin Kesmavet

#### 4.4.3. Spesifikasi Kebutuhan Non-Fungsional Sistem

Kebutuhan non-fungsional untuk SIMLAB adalah *Usability*. Sistem harus memenuhi kemudahan penggunaan (*Usability*) dari berbagai aspek seperti (*Learnability*) kemudahan dipelajari, (*Efficiency*) efisiensi, (*Memorability*) kemudahan diingat, (*Errors*) tingkat kesalahan, dan (*Satisfaction*) kepuasan penggunaan.

#### 4.5 Perancangan *Prototype*

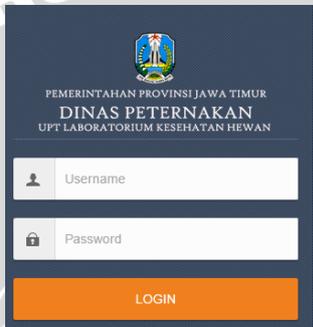
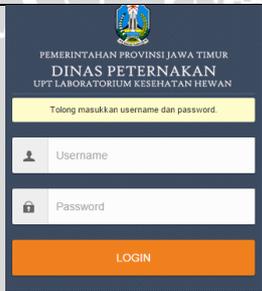
Pada tahap ini dilakukan perancangan *prototype* SIMLAB berdasarkan dari hasil analisa dan perminataan dari user. Rancangan *prototype* sistem ini terdiri dari perancangan *prototype* fungsional dan antarmuka.

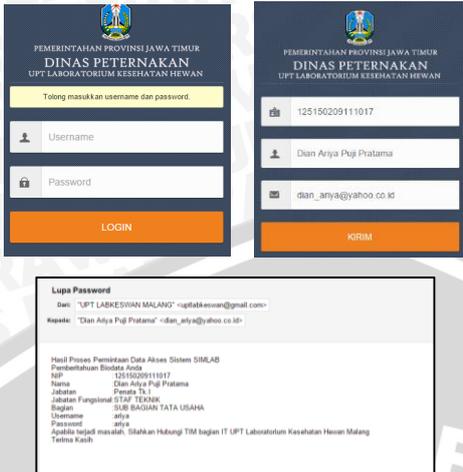
#### 4.5.1. Perancangan *Prototype* Fungsional

Pada tahap ini dilakukan perancangan prototype fungsional pada SIMLAB yang bertujuan untuk mengetahui tahapan-tahapan perubahan secara fungsionalitas (fitur) atau proses kerja SIMLAB. Beberapa perubahan yang terjadi digambarkan oleh prototype berikut ini.

##### 1. Fitur Login

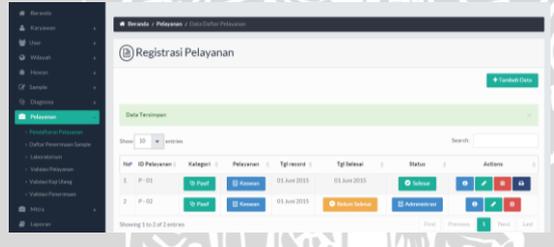
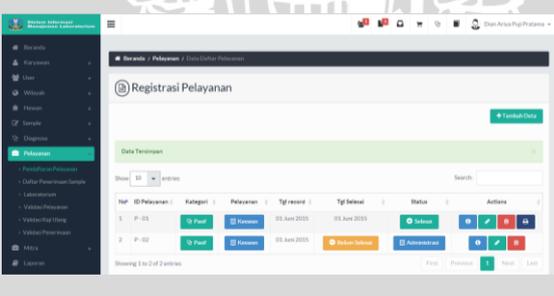
**Tabel 4.3 Prototype Fungsional Fitur Login**

No	Versi	Gambar	Deskripsi
1	Beta 1		Dapat melakukan proses login saja.
2	Beta 2		Dapat melakukan proses login dan terdapat pesan error

3	Final		<p>Dapat melakukan proses login, terdapat pesan error dan terdapat fungsi lupa password yang langsung dikirim ke email dari account user yang didaftarkan.</p>
---	-------	--	--

## 2. Fitur Validasi

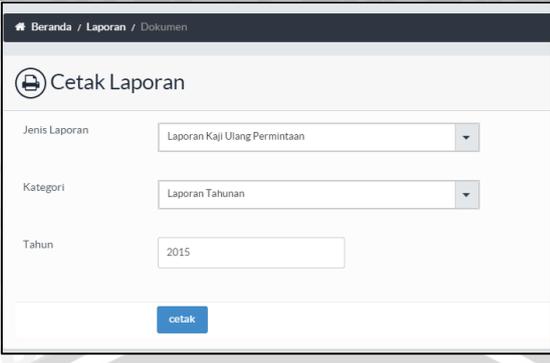
**Tabel 4.4 Prototype Fungsional Fitur Validasi**

No	Versi	Gambar	Deskripsi
1	Beta 1		<p>Terdapat pesan setelah melakukan input pendaftaran pelayanan.</p>
2	Final		<p>Terdapat notifikasi dan pesan setelah melakukan input pendaftaran pelayanan.</p>

## 3. Fitur Pelaporan

**Tabel 4.5 Prototype Fungsional Fitur Pelaporan**

No	Versi	Gambar	Deskripsi
1	Beta 1		<p>Pelaporan satu persatu</p>

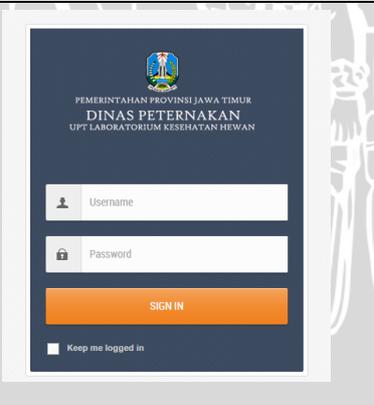
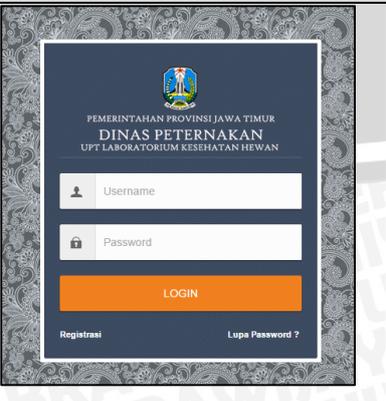
2	Final		Terdapat Rakap Pelaporan
---	-------	--	--------------------------

#### 4.5.2. Perancangan *Prototype* Antarmuka

Pada tahap ini dilakukan perancangan prototype desain antarmuka pada SIMLAB yang bertujuan untuk mengetahui tahapan-tahapan perubahan tampilan dari SIMLAB. Beberapa perubahan yang terjadi digambarkan oleh prototype berikut ini.

##### 1. Desain Antarmuka Login

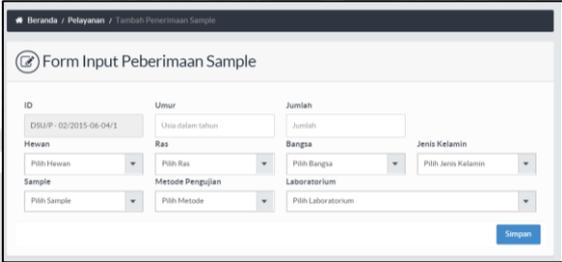
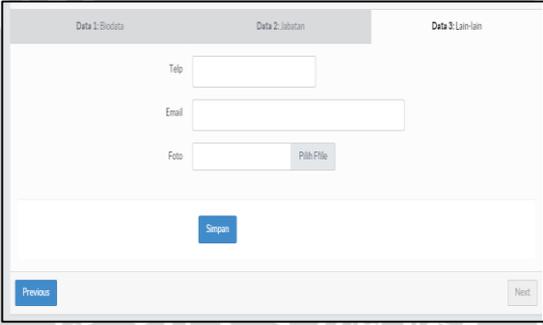
**Tabel 4.6 Prototype Antarmuka Desain Antarmuka Login**

No	Versi	Gambar	Deskripsi
1	Beta 1		<ul style="list-style-type: none"> <li>✓ Background polos berwarna abu-abu.</li> <li>✓ Button sign in</li> </ul>
2	Final		<ul style="list-style-type: none"> <li>✓ Background batik.</li> <li>✓ Button Login</li> <li>✓ Menu Registrasi dan Lupa Password</li> </ul>



## 2. Desain Antarmuka Form Input dan Edit

Tabel 4.7 Prototype Antarmuka Desain Antarmuka Form Input dan Edit

No	Versi	Gambar	Deskripsi
1	Beta 1		<ul style="list-style-type: none"> <li>✓ Field input hanya posisi vertikal</li> </ul>
2	Final		<ul style="list-style-type: none"> <li>✓ Field input posisi vertikal</li> <li>✓ Field input posisi ber tab-tab</li> </ul>

## 3. Desain Antarmuka Validasi Form

Tabel 4.8 Prototype Antarmuka Desain Antarmuka Validasi Form

No	Versi	Gambar	Deskripsi
1	Beta 1		<ul style="list-style-type: none"> <li>✓ Button Icon validasi hanya di tabel record data</li> </ul>
2	Final		<ul style="list-style-type: none"> <li>✓ Button notifikasi.</li> <li>✓ Button Validasi di detail laporan atau form</li> </ul>

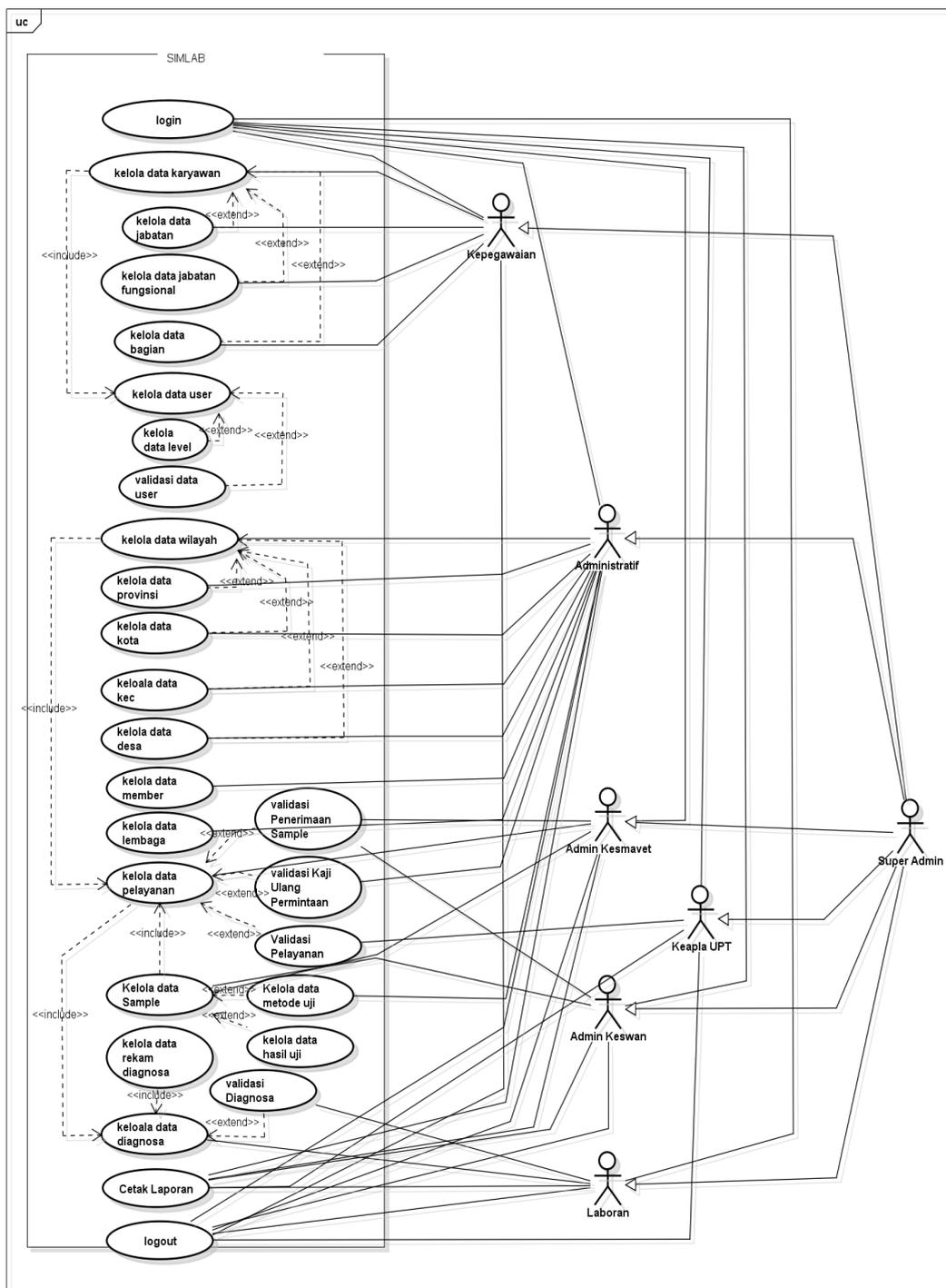
I Jumlah contoh yang dikirim :		<input checked="" type="checkbox"/> Memenuhi	<input type="checkbox"/> Tidak Memenuhi
II Kondisi Contoh :		<input checked="" type="checkbox"/> Segar	<input type="checkbox"/> Beku <input type="checkbox"/> Dengan Pengawet <input type="checkbox"/> Rusak
III Pemenuhan Terhadap Sumber daya :		<input checked="" type="checkbox"/> Tersedia	<input type="checkbox"/> Tidak Tersedia
1. Alat :		<input checked="" type="checkbox"/> Tersedia	<input type="checkbox"/> Tidak Tersedia
2. Bahan :		<input checked="" type="checkbox"/> Tersedia	<input type="checkbox"/> Tidak Tersedia
3. Metode Uji :		<input checked="" type="checkbox"/> Tersedia	<input type="checkbox"/> Tidak Tersedia
4. Standart Uji :		<input checked="" type="checkbox"/> Tersedia	<input type="checkbox"/> Tidak Tersedia
5. Personil :		<input checked="" type="checkbox"/> Tersedia	<input type="checkbox"/> Tidak Tersedia
Manejer Administrasi		MT Keswan / MT Kesmavet	
		<input checked="" type="checkbox"/> Validasi Administratif	<input type="checkbox"/> Penerimaan Sample Terlebih Dahulu

## 4.6 Perancangan Sistem

Pada tahap ini dilakukan perancangan sistem yang akan dibuat berdasarkan dari analisa yang telah ditentukan sebelumnya. Rancangan sistem ini terdiri dari perancangan *Usecase Diagram*, skenario *Class Diagram*, perancangan *Activity Diagram*, perancangan *Sequence Diagram*, dan perancangan database serta perancangan *Class Diagram*.

### 4.6.1. Perancangan *Usecase Diagram*

Diagram *Usecase* merupakan pemodelan untuk kelakuan (behavior) sistem Informasi yang akan dibuat. *Usecase* menggambarkan interaksi antar satu atau lebih aktor serta fungsionalitas yang disediakan sistem. Berikut adalah fungsionalitas aktor dalam Sistem Informasi Manajemen Laboratorium.



Gambar 4.1 Usecase Diagram SIMLAB

#### 4.6.2. Skenario Usecase Diagram

Berikut adalah skenario jalannya masing-masing *Usecase* yang telah di definisikan sebelumnya:

a. Skenario *Usecase* LoginTabel 4.9 Skenario *Usecase* Login

Nomor SRS	<b>SRS_1_01</b>	
Nama <i>Usecase</i> :	<b>Login</b>	
Aktor :	Super Admin, Kepala UPT, Administratif, Admin Kepegawaian , Admin Laboran, Admin Keswan dan Admin Kesmavet	
Deskripsi :	<i>Usecase</i> ini digunakan untuk mendeskripsikan kegiatan autentikasi Aktor sebelum masuk ke system berdasarkan user privileginya.	
Kondisi Awal :	Mengakses halaman form login	
Skenario Normal :	Aksi Aktor :	Reaksi Sistem :
	1. Memasukkan data berupa username dan password. Kemudian klik button login.	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. Kemudian klik button login.  4. Masukkan ulang username dan password dengan benar. Kemudian klik login.	2. Memeriksa verifikasi tidaknya data masukan. 3. Sistem akan menampilkan pesan kesalahan, aktor diminta untuk memasukkan ulang username dan password. 5. Memeriksa valid tidaknya data masukan. 6. 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. Kemudian klik kirim.	2. Menampilkan Form lupa password.  4. Memeriksa valid tidaknya data masukan. 5. Apabila valid akan masuk akan muncul pesan “data terkirim”. Kemudian menunggu

		<p>hingga data telah divalidasi oleh Super Admin.</p> <p>6. Apabila tidak valid maka akan muncul pesan “data yang anda masukkan tidak terdaftar, Mohon permintaan baru password baru ke Super Admin”.</p>
Kondisi Akhir :	<p>Jika <i>Usecase</i> sukses, Aktor bisa masuk ke sistem untuk melakukan kegiatannya berdasarkan hak akses yang telah diberikan.</p>	

**b. Skenario Kelola Data Karyawan**

**Tabel 4.10 Skenario Kelola Data Karyawan**

Nomor SRS	<b>SRS_3_01, SRS_3_02, SRS_3_03 dan SRS_3_04</b>	
Nama <i>Usecase</i> :	Kelola Data Karyawan	
Aktor :	Super Admin dan Admin Kepegawaian	
Deskripsi :	Usecase ini digunakan untuk mendeskripsikan kegiatan mengelola data karyawan.	
Kondisi Awal :	Aktor mengakses halaman kelola karyawan sebelum melakukan aktivitas mengelola data karyawan.	
Skenario Normal :	Tambah Data Karyawan	
	Aksi Aktor :	Reaksi Sistem :
	<p>1. Aktor masuk ke menu Karyawan.</p> <p>4. Aktor memilih button tambah data.</p>	<p>2. Sistem menampilkan modul karyawan.</p> <p>3. Sistem menampilkan daftar data karyawan.</p> <p>5. Sistem menampilkan form tambah data karyawan.</p>
	<p>6. Aktor memasukkan data karyawan. Kemudian klik button “simpan”.</p>	<p>7. Apabila inputan yang di masukkan sesuai maka sistem menampilkan pesan notifikasi data berhasil disimpan sedangkan apabila ada inputan yang di masukkan tidak sesuai maka akan keluar pesan validasi form dan pesan</p>

		notifikasi data sudah ada.
<b>Ubah Data Karyawan</b>		
	<b>Aksi Aktor :</b>	<b>Reaksi Sistem :</b>
	<ol style="list-style-type: none"> <li>1. Aktor masuk ke menu Karyawan.</li> <li>4. Aktor memilih icon ubah pada data karyawan yang dipilih.</li> <li>6. Aktor mengubah data karyawan. Kemudian klik button "simpan".</li> </ol>	<ol style="list-style-type: none"> <li>2. Sistem menampilkan modul Karyawan.</li> <li>3. Sistem menampilkan daftar data karyawan.</li> <li>5. Sistem menampilkan form ubah data karyawan.</li> <li>7. Apabila inputan yang di masukkan sesuai maka sistem menampilkan pesan notifikasi data berhasil diubah sedangkan apabila ada inputan yang di masukkan tidak sesuai maka akan keluar pesan validasi form dan pesan notifikasi data sudah ada.</li> </ol>
<b>Hapus Data Karyawan</b>		
	<b>Aksi Aktor :</b>	<b>Reaksi Sistem :</b>
	<ol style="list-style-type: none"> <li>1. Aktor masuk ke menu Karyawan</li> </ol>	<ol style="list-style-type: none"> <li>2. Sistem menampilkan modul karyawan</li> <li>3. Sistem menampilkan daftar data karyawan.</li> </ol>

	<p>4. Aktor memilih icon hapus pada data karyawan yang dipilih.</p> <p>6. Aktor memilih opsi ya atau tidak.</p>	<p>5. Sistem akan menampilkan pesan peringatan hapus data. Pesan peringatan hapus “Apakah anda ingin menghapus data ini ?”.</p> <p>7. Apabila Aktor memilih ya maka akan muncul pesan data berhasil dihapus sedangkan apabila Aktor memilih tidak maka data kembali ke daftar data karyawan.</p>
	<b>Cari Data Karyawan</b>	
	<b>Aksi Aktor :</b>	<b>Reaksi Sistem :</b>
	<p>1. Aktor masuk ke menu karyawan.</p> <p>4. Aktor mengetikkan keyword di form pencarian.</p>	<p>2. Sistem menampilkan modul karyawan.</p> <p>3. Sistem menampilkan data karyawan.</p> <p>5. Sistem melakukan pencarian data member sesuai dari keyword yang di masukkan.</p> <p>6. Sistem menampilkan hasil pencarian.</p>
	<b>Lihat Detil Data Karyawan</b>	
	<b>Aksi Aktor :</b>	<b>Reaksi Sistem :</b>
	<p>1. Aktor masuk ke menu Karyawan.</p> <p>4. Aktor memilih icon detil pada data karyawan yang dipilih.</p>	<p>2. Sistem menampilkan modul karyawan.</p> <p>3. Sistem menampilkan data karyawan.</p> <p>5. Sistem menampilkan detil data karyawan.</p>
Skenario Alternatif :	<b>Tambah Data Karyawan</b>	
	<b>Aksi Aktor :</b>	<b>Reaksi Sistem :</b>
	<p>1. Aktor masuk ke menu Karyawan.</p> <p>4. Aktor memilih button tambah data.</p>	<p>2. Sistem menampilkan modul karyawan.</p> <p>3. Sistem menampilkan daftar data karyawan.</p> <p>5. Sistem menampilkan form tambah data karyawan.</p>

	<p>6. Aktor memasukkan data karyawan dan beberapa data kosong. Kemudian klik button “simpan”.</p> <p>8. Aktor memasukkan data karyawan lagi dan melengkapi data yang terdapat pesan error di form input. Kemudian klik button “simpan”.</p>	<p>7. Mempilkan pesan error di form input karyawan.</p> <p>9. Apabila inputan yang di masukkan sesuai maka sistem menampilkan pesan notifikasi data berhasil diubah sedangkan apabila ada inputan yang di masukkan tidak sesuai maka akan keluar pesan validasi form dan pesan notifikasi data sudah ada.</p>
<b>Ubah Data Karyawan</b>		
<b>Aksi Aktor :</b>		<b>Reaksi Sistem :</b>
	<p>1. Aktor masuk ke menu Karyawan.</p> <p>4. Aktor memilih icon ubah pada data yang akan di ubah.</p> <p>6. Aktor mengubah data karyawan. Kemudian klik button “simpan”.</p> <p>9. Memperbaiki data masukkan yang tidak valid.</p>	<p>2. Sistem menampilkan modul Karyawan.</p> <p>3. Sistem menampilkan daftar data karyawan.</p> <p>5. Sistem menampilkan form ubah data karyawan.</p> <p>7. Memeriksa valid tidaknya data masukkan.</p> <p>8. Mengeluarkan pesan bahwa data masukkan tidak valid.</p> <p>10. Memeriksa valid tidaknya data masukkan.</p> <p>11. Menyimpan data karyawan.</p> <p>12. Menampilkan pesan “data berhasil diubah”.</p>
<p><b>Kondisi Akhir :</b></p>	<p>Jika <i>Usecase</i> sukses, aktor bisa masuk ke sistem untuk melakukan proses tambah data karyawan baru, ubah data karyawan, cari data karyawan, hapus data karyawan dan lihat detil data karyawan.</p>	

c. Skenario *Usecase* Kelola Wilayah

Tabel 4.11 Skenario Kelola Data Wilayah

Nomor SRS	<b>SRS_5_01, SRS_5_02, SRS_5_03, SRS_5_04, SRS_5_05, SRS_5_06, SRS_5_07, SRS_5_08, SRS_5_09, SRS_5_10, SRS_5_11, dan SRS_5_12</b>	
Nama <i>Usecase</i> :	Kelola Wilayah	
Aktor :	Super admin dan Administratif	
Deskripsi :	<i>Usecase</i> ini digunakan untuk mendeskripsikan kegiatan mengelola data wilayah pengambilan sampel.	
Kondisi Awal :	<ul style="list-style-type: none"> <li>✓ Aktor mengakses halaman kelola wilayah Provinsi sebelum melakukan aktivitas mengelola data wilayah provinsi.</li> <li>✓ Aktor sudah menginputkan data wilayah Provinsi sebelum melakukan aktivitas mengelola data wilayah Kota dan mengakses halaman kelola wilayah Kota.</li> <li>✓ Aktor sudah menginputkan data kelola wilayah Provinsi dan Kota sebelum melakukan aktivitas mengelola data wilayah Kec/Kab serta mengakses halaman kelola wilayah Kec/Kab</li> <li>✓ Aktor sudah menginputkan data kelola wilayah Provinsi, Kota dan kec/kelurahan sebelum melakukan aktivitas mengelola data wilayah Desa serta mengakses halaman kelola wilayah Desa.</li> </ul>	
Skenario Normal :	Tambah Data Wilayah	
	Aksi Aktor :	Reaksi Sistem :
	<ol style="list-style-type: none"> <li>1. User masuk ke menu Wilayah.</li> <li>4. User memilih button tambah data.</li> <li>6. User memasukkan data wilayah. Kemudian klik button “simpan”.</li> </ol>	<ol style="list-style-type: none"> <li>2. Sistem menampilkan modul wilayah.</li> <li>3. Sistem menampilkan daftar data wilayah.</li> <li>5. Sistem menampilkan form tambah data wilayah.</li> <li>7. Apabila inputan yang di masukkan sesuai maka sistem menampilkan pesan notifikasi data berhasil disimpan sedangkan apabila ada inputan yang di masukkan tidak sesuai maka akan keluar pesan validasi form dan pesan notifikasi data sudah ada.</li> </ol>

Ubah Data Wilayah	
Aksi Aktor :	Reaksi Sistem :
<ol style="list-style-type: none"> <li>1. User masuk ke menu wilayah.</li> <li>4. User memilih icon ubah pada data wilayah yang dipilih.</li> <li>6. User mengubah data wilayah. Kemudian klik button “simpan”.</li> </ol>	<ol style="list-style-type: none"> <li>2. Sistem menampilkan modul wilayah.</li> <li>3. Sistem menampilkan daftar data wiayah.</li> <li>5. Sistem menampilkan form ubah data wilayah.</li> <li>7. Apabila inputan yang di masukkan sesuai maka sistem menampilkan pesan notifikasi data berhasil diubah sedangkan apabila ada inputan yang di masukkan tidak sesuai maka akan keluar pesan validasi form dan pesan notifikasi data sudah ada.</li> </ol>
Hapus Data Wilayah	
Aksi Aktor :	Reaksi Sistem :
<ol style="list-style-type: none"> <li>1. User masuk ke menu wilayah.</li> <li>4. User memilih icon hapus pada data yang dipilih.</li> <li>6. User memilih opsi ya atau tidak.</li> </ol>	<ol style="list-style-type: none"> <li>2. Sistem menampilkan modul wilayah.</li> <li>3. Sistem menampilkan daftar wilayah.</li> <li>5. Sistem akan menampilkan pesan peringatan hapus data. Pesan peringatan hapus “Apakah anda ingin menghapus data ini ?”.</li> <li>7. Apabila user memilih ya maka akan muncul pesan data berhasil dihapus sedangkan apabila user memilih tidak maka data kembali ke daftar data wilayah.</li> </ol>

Cari Data Wilayah		
Aksi Aktor :		Reaksi Sistem :
1. User masuk ke menu wilayah.		2. Sistem menampilkan modul wilayah.
4. User mengetikkan keyword di form pencarian.		3. Sistem menampilkan data wilayah.
		5. Sistem melakukan pencarian data wilayah sesuai dari keyword yang di masukkan.
		6. Sistem menampilkan hasil pencarian.
Skenario Alternatif : Tambah Data Wilayah		
Aksi Aktor :		Reaksi Sistem :
1. User masuk ke menu Wilayah.		2. Sistem menampilkan modul Wilayah.
4. User memilih icon ubah pada data wilayah yang dipilih.		3. Sistem menampilkan daftar data Wilayah.
6. User mengubah data Wilayah Kemudian klik button "simpan".		5. Sistem menampilkan form ubah data Wilayah.
9. Memperbaiki data masukkan yang tidak valid.		7. Memeriksa valid tidaknya data masukkan.
		8. Mengeluarkan pesan bahwa data masukkan tidak valid.
		10. Memeriksa valid tidaknya data masukkan.
		11. Menyimpan data Wilayah.
		12. Menampilkan pesan "data berhasil disimpan".
Ubah Data Wilayah		
Aksi Aktor :		Reaksi Sistem :
1. User masuk ke menu Wilayah.		2. Sistem menampilkan modul Wilayah.
		3. Sistem menampilkan daftar data Wilayah.

	<p>4. User memilih icon ubah pada data wilayah yang dipilih.</p> <p>6. User mengubah data Wilayah. Kemudian klik button “simpan”.</p> <p>9. Memperbaiki data masukkan yang tidak valid.</p>	<p>5. Sistem menampilkan form ubah data Wilayah.</p> <p>7. Memeriksa valid tidaknya data masukkan.</p> <p>8. Mengeluarkan pesan bahwa data masukkan tidak valid.</p> <p>10. Memeriksa valid tidaknya data masukkan.</p> <p>11. Menyimpan data Wilayah.</p> <p>12. Menampilkan pesan “data berhasil diubah”.</p>
Kondisi Akhir :	Jika <i>Usecase</i> sukses, aktor bisa masuk ke sistem untuk melakukan proses tambah data wilayah baru, ubah data wilayah, cari wilayah, dan hapus data wilayah .	

d. Skenario *Usecase* Kelola pendaftaran Pelayanan

**Tabel 4.12 Skenario Kelola Data Pendaftaran Pelayanan**

Nomor SRS	SRS_9_01, SRS_9_02, SRS_9_03, SRS_9_04	
Nama <i>Usecase</i> :	Kelola Pendaftaran Pelayanan	
Aktor :	Super admin dan Administratif	
Deskripsi :	<i>Usecase</i> ini digunakan untuk mendeskripsikan kegiatan mengelola data pelayanan berdasarkan kategori pelayanan pengujian Sampel yaitu pelayanan aktif, pelayanan semi aktif dan pelayanan pasif. Jenis Pelayanan Kesmavet dan Keswan.	
Kondisi Awal :	<p>✓ Member telah terdaftar</p> <p>✓ Aktor Mengakses halaman pendaftaran pelayanan.</p>	
Skenario Normal :	Tambah Data Pelayanan	
	Aksi Aktor :	Reaksi Sistem :
	1. User masuk ke menu pendaftaran pelayanan.	<p>2. Sistem menampilkan modul pendaftaran pelayanan.</p> <p>3. Sistem menampilkan daftar data pendaftaran pelayanan.</p>

	<p>4. Aktor klik Button tambah data.</p> <p>6. User memasukkan data pendaftaran pelayanan. Kemudian klik button “simpan”.</p>	<p>5. Sistem menampilkan form tambah data pendaftaran pelayanan.</p> <p>7. Apabila inputan yang di masukkan sesuai maka sistem menampilkan pesan notifikasi data berhasil disimpan sedangkan apabila ada inputan yang di masukkan tidak sesuai maka akan keluar pesan validasi form.</p>
Ubah Data Pelayanan		
	Aksi Aktor :	Reaksi Sistem :
	<p>1. Aktor masuk ke menu pendaftaran pelayanan.</p> <p>4. Aktor memilih icon ubah pada data pendaftaran pelayanan yang dipilih.</p> <p>6. Aktor mengubah data pendaftaran pelayanan. Kemudian klik button “simpan”.</p>	<p>2. Sistem menampilkan modul pendaftaran pelayanan.</p> <p>3. Sistem menampilkan daftar data pendaftaran pelayanan.</p> <p>5. Sistem menampilkan form ubah data pendaftaran pelayanan.</p> <p>7. Apabila inputan yang di masukkan sesuai maka sistem menampilkan pesan notifikasi data berhasil diubah sedangkan apabila ada inputan yang di masukkan tidak sesuai maka akan keluar pesan validasi form.</p>

Hapus Data Pelayanan	
Aksi Aktor :	Reaksi Sistem :
1. Aktor masuk ke menu pendaftaran pelayanan.	2. Sistem menampilkan modul pendaftaran pelayanan
4. Aktor memilih icon hapus pada data pendaftaran pelayanan yang akan dihapus.	3. Sistem menampilkan daftar endafataran pelayanan. 5. Sistem akan menampilkan pesan peringatan hapus data. Pesan peringatan hapus “Apakah anda ingin menghapus data ini ?”.
6. Aktor memilih opsi ya atau tidak.	7. Apabila Aktor memilih ya maka akan muncul pesan data berhasil dihapus sedangkan apabila Aktor memilih tidak maka data kembali ke daftar data pelayanan.
Cari Data Pelayanan	
Aksi Aktor :	Reaksi Sistem :
1. Aktor masuk ke menu pelayanan.	2. Sistem menampilkan modul pelayanan. 3. Sistem menampilkan data pelayanan.
4. Aktor mengetikkan keyword di form pencarian.	5. Sistem melakukan pencarian data pendaftaran pelayanan sesuai dari keyword yang di masukkan. 6. Sistem menampilkan hasil pencarian.
Lihat Detil Data Pelayanan	
Aksi Aktor :	Reaksi Sistem :
1. Aktor masuk ke menu pelayanan.	2. Sistem menampilkan modul pelayanan. 3. Sistem menampilkan data pelayanan.
4. Aktor memilih icon detil pada data yang akan dilihat.	5. Sistem menampilkan detil data pendaftaran pelayanan.

Kondisi Akhir :	Jika <i>Usecase</i> sukses, aktor bisa masuk ke sistem untuk melakukan proses tambah data pendaftaran pelayanan baru, ubah data pendaftaran pelayanan, cari data pendaftaran pelayanan, hapus data pendaftaran pelayanan dan lihat detail data pendaftaran pelayanan.
-----------------	---

e. Skenario *Usecase* Validasi Penerimaan Sampel

**Tabel 4.13 Skenario Validasi Penerimaan Sampel.**

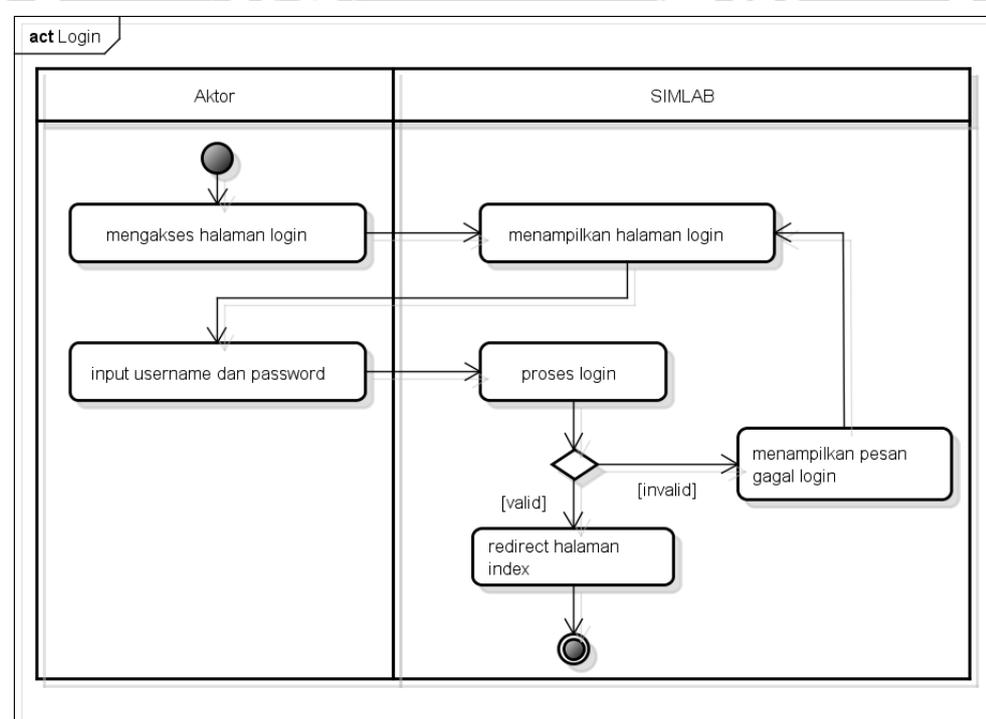
Nomor SRS	<b>SRS_9_11</b>	
Nama <i>Usecase</i> :	<b>Validasi Penerimaan Sampel</b>	
Aktor :	Super Admin, Admin Keswan dan Admin Kesmavet	
Deskripsi :	<i>Usecase</i> ini digunakan untuk mendeskripsikan kegiatan autentikasi data sampel sebelum dilakukan pengujian dan masuk ke laboratorium.	
Kondisi Awal :	<ul style="list-style-type: none"> <li>✓ Telah Melakukan pendaftaran pelayanan</li> <li>✓ Membawa sampel pengujian</li> <li>✓ Data penerimaan sampel telah terdaftar.</li> </ul>	
Skenario Normal :	Aksi Aktor :	Reaksi Sistem :
	<ol style="list-style-type: none"> <li>1. Melihat notifikasi penerimaan sampel.</li> <li>2. Klik notifikasi penerimaan sampel</li> <li>3. Pilih salah satu data yang akan dilakukan validasi penerimaan sample.</li> <li>5. Klik button validasi</li> </ol>	<ol style="list-style-type: none"> <li>4. Menampilkan detail data penerimaan sampel.</li> <li>6. Data penerimaan sampel tervalidasi.</li> </ol>
Skenario Alternatif :	Aksi Aktor :	Reaksi Sistem :
	<ol style="list-style-type: none"> <li>1. Klik menu validasi penerimaan sample.</li> <li>3. Ketikan keywor id pelayanan di form search.</li> <li>5. Klik icon validasi penerimaan sampel.</li> </ol>	<ol style="list-style-type: none"> <li>2. Menampilkan data validasi penerimaan sampel.</li> <li>4. Menampilkan data penerimaan sampel yang dicari. Apabila tidak ada data tidak tampil.</li> <li>6. Data penerimaan sampel tervalidasi</li> </ol>
Kondisi Akhir :	Jika <i>Usecase</i> sukses, aktor bisa masuk ke sistem untuk melakukan proses validasi penerimaan sampel.	

#### 4.6.3. Perancangan *Activity Diagram*.

Pembuatan *Activity Diagram* ini bertujuan untuk menggambarkan urutan aktivitas dari proses dari setiap *Usecase* yang ada. Berikut merupakan gambar dari masing-masing *Activity Diagram*.

##### a) *Activity Diagram Login*

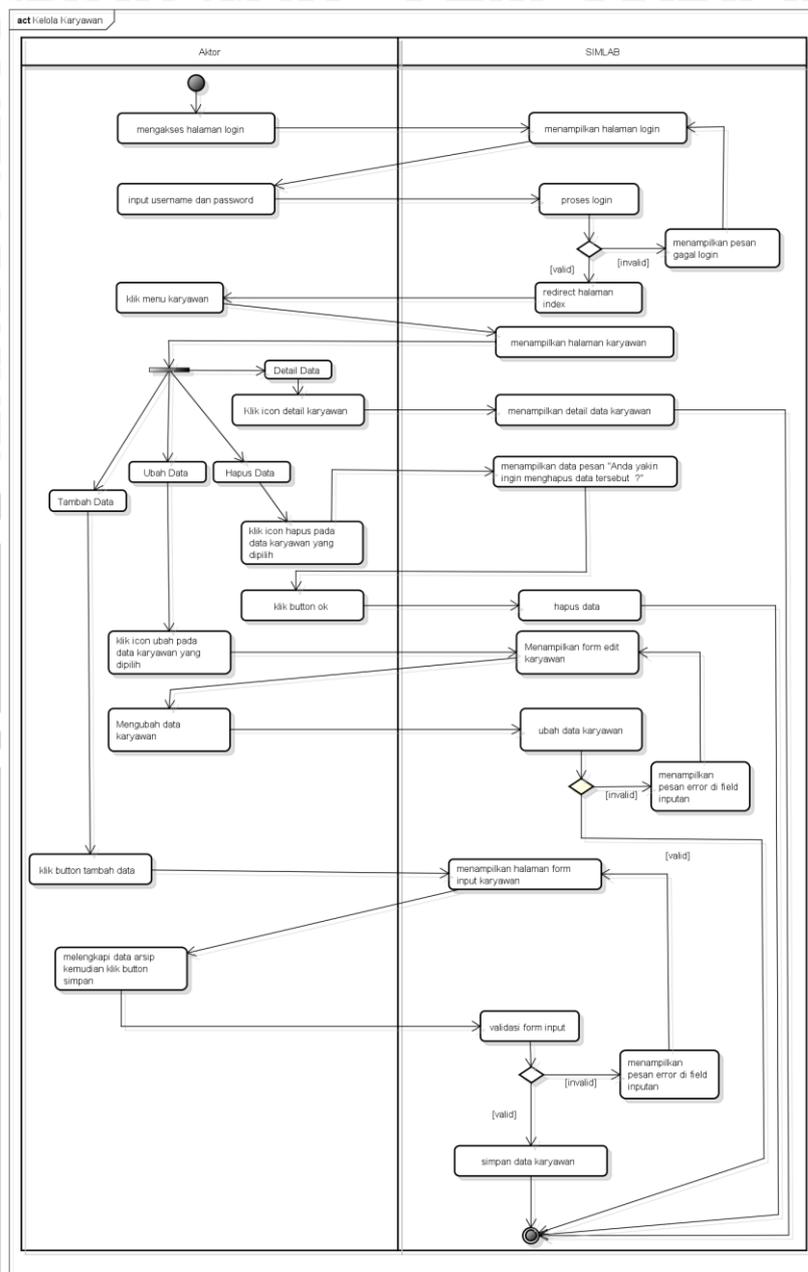
Pada Gambar 4.2 merupakan gambar *Activity Diagram* login yang menjelaskan aktifitas dilakukan oleh aktor untuk masuk ke dalam halaman sistem masing-masing user.



Gambar 4.2 *Activity Diagram Login*

##### b) *Activity Diagram Kelola Karyawan*

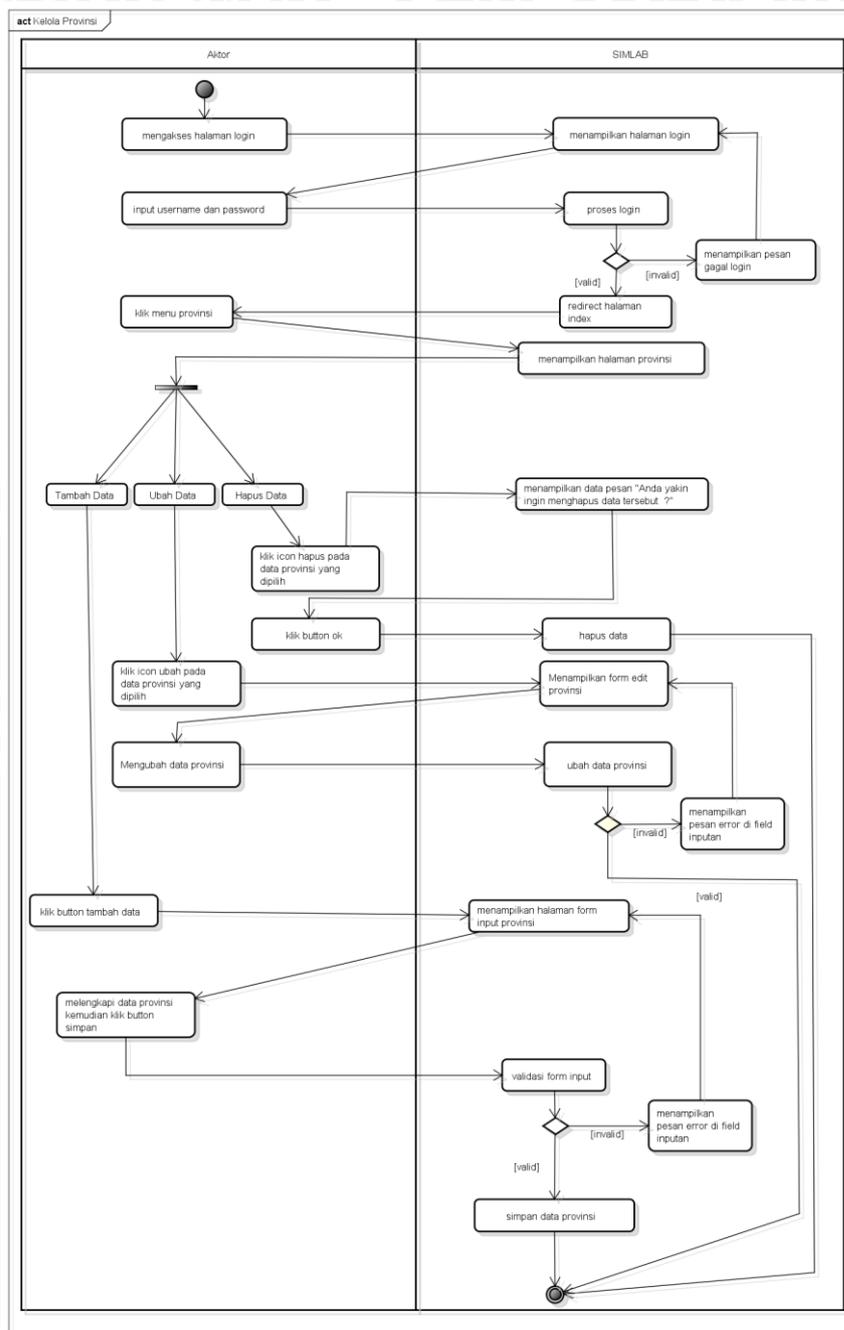
Pada Gambar 4.3 merupakan gambar *Activity Diagram* kelola karyawan yang menjelaskan aktifitas yang dilakukan oleh aktor (super admin dan admin kepegawaian) dengan sistem dalam mengelola data karyawan atau staf yang ada di UPT Laboratorium Kesehatan Hewan Malang.



Gambar 4.3 Activity Diagram Kelola Karyawan

c) Activity Diagram Kelola Wilayah (Provinsi)

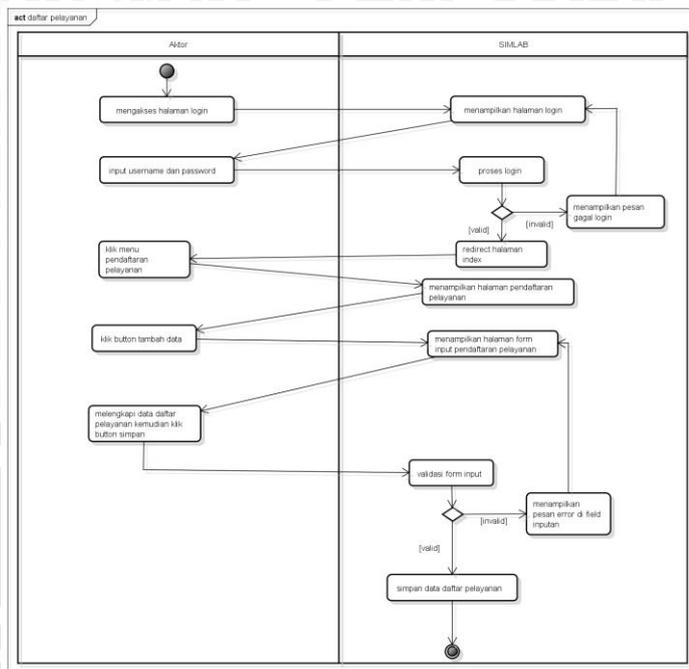
Pada Gambar 4.4 merupakan gambar Activity Diagram kelola wilayah (Provinsi) yang menjelaskan aktifitas yang dilakukan oleh aktor (super admin dan Administrasitif) dengan sistem dalam mengelola data wilayah yang asal sampel berasal sebelum dilakukan pengujian.



Gambar 4.4 Activity Diagram Kelola Wilayah

d) Activity Diagram Pendaftaran Pelayanan

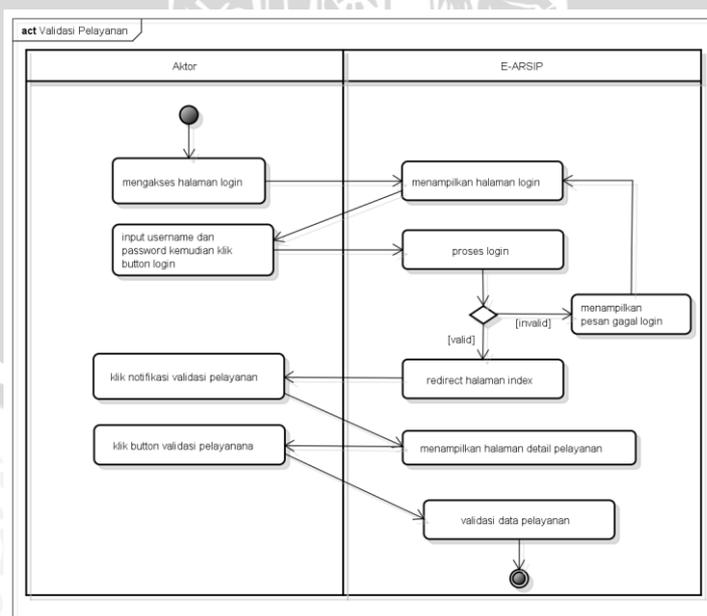
Pada Gambar 4.5 merupakan gambar Activity Diagram pendaftaran pelayanan yang menjelaskan aktifitas yang dilakukan oleh aktor (Super Admin dan Administrastitif) dengan sistem dalam pendaftaran pelayanan pengujian sample dengan mengkategorikan jenis pelayanan kepada member yang melakukan permintaan pengujian sampel.



Gambar 4.5 Activity Diagram Pendaftaran Pelayanan

e) Activity Diagram Validasi Penerimaan Sampel

Pada Gambar 4.6 merupakan gambar Activity Diagram validasi penerimaan sampel yang menjelaskan aktifitas yang dilakukan oleh aktor (Super Admin, Admin Keswan dan Admin Kesmavet) dengan sistem dalam mengesahkan penerimaan sampel dari member ke bagian pelayanan.



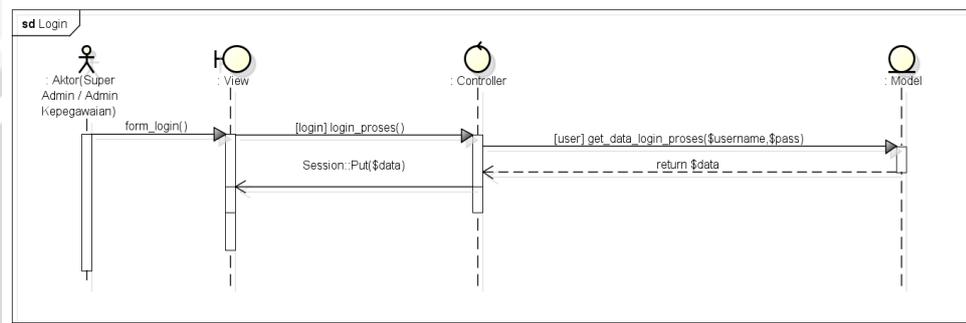
Gambar 4.6 Activity Diagram Validasi Penerimaan Sampel

### 4.5.3. Perancangan *Sequence Diagram*.

*Sequence Diagram* menggambarkan interaksi antar masing-masing objek pada setiap *Usecase* dalam urutan waktu. Interaksi ini berupa pengiriman serangkaian data antar objek-objek yang saling berinteraksi.

#### a) *Sequence Diagram* Login

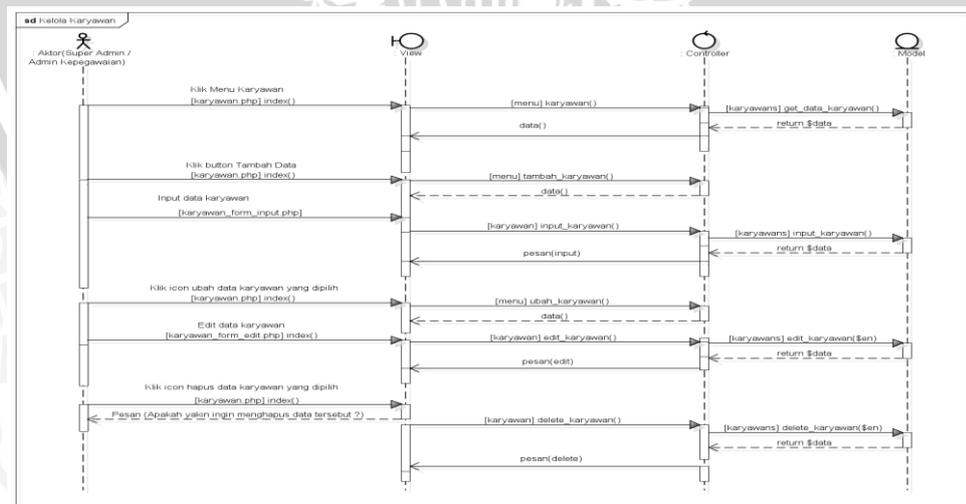
Pada Gambar 4.7 merupakan gambar *Sequence Diagram* login yang menjelaskan interaksi aktor dengan sistem dalam melakukan login masuk ke dalam sistem sesuai dengan hak akses user.



**Gambar 4.7 *Sequence Diagram* Login**

#### b) *Sequence Diagram* Kelola Karyawan

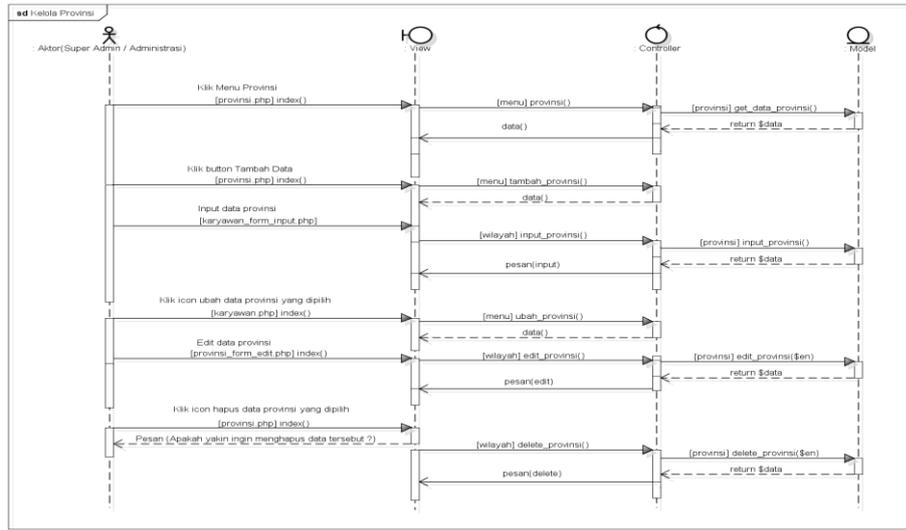
Pada Gambar 4.8 merupakan gambar *Sequence Diagram* kelola karyawan yang menjelaskan interaksi aktor (Super Admin dan Admin Kepegawaian) dengan sistem dalam mengelola data karyawan.



**Gambar 4.8 *Sequence Diagram* Kelola Karyawan**

c) *Sequence Diagram* Kelola Wilayah (Provinsi)

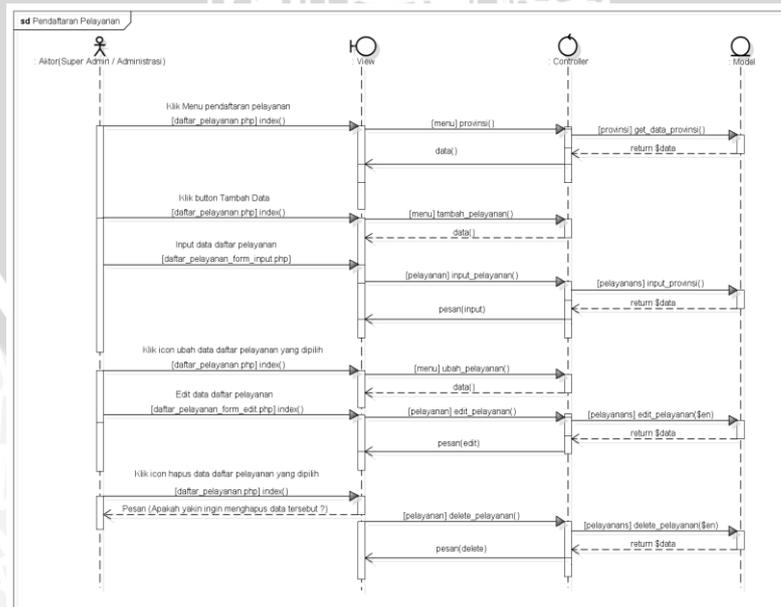
Pada Gambar 4.9 merupakan gambar *Sequence Diagram* kelola wilayah yang menjelaskan interaksi aktor (Super Admin dan Administratif) dengan sistem dalam mengelola data wilayah provinsi.



**Gambar 4.9** *Sequence Diagram* Kelola Wilayah (Provinsi)

d) *Sequence Diagram* Pendaftaran Pelayanan

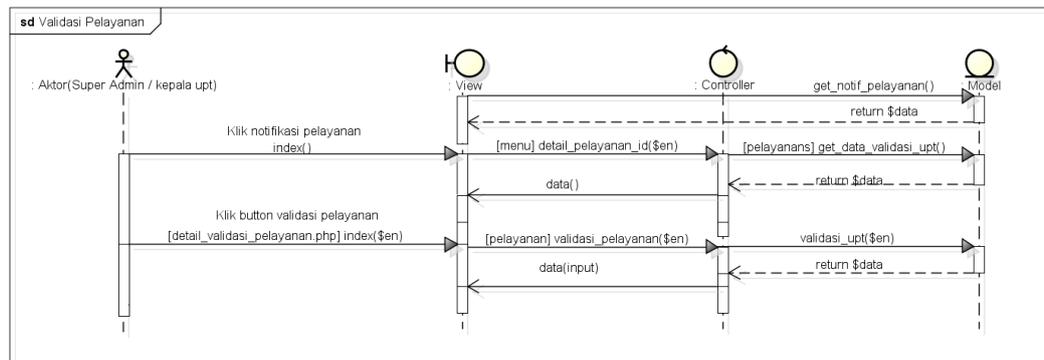
Pada Gambar 4.10 merupakan gambar *Sequence Diagram* pendaftaran pelayanan yang menjelaskan interaksi aktor (Super Admin dan Administratif) dengan sistem dalam mengelola data pelayanan.



**Gambar 4.10** *Sequence Diagram* Pendaftaran Pelayanan

e) *Sequence* Diagram Validasi Pelayanan

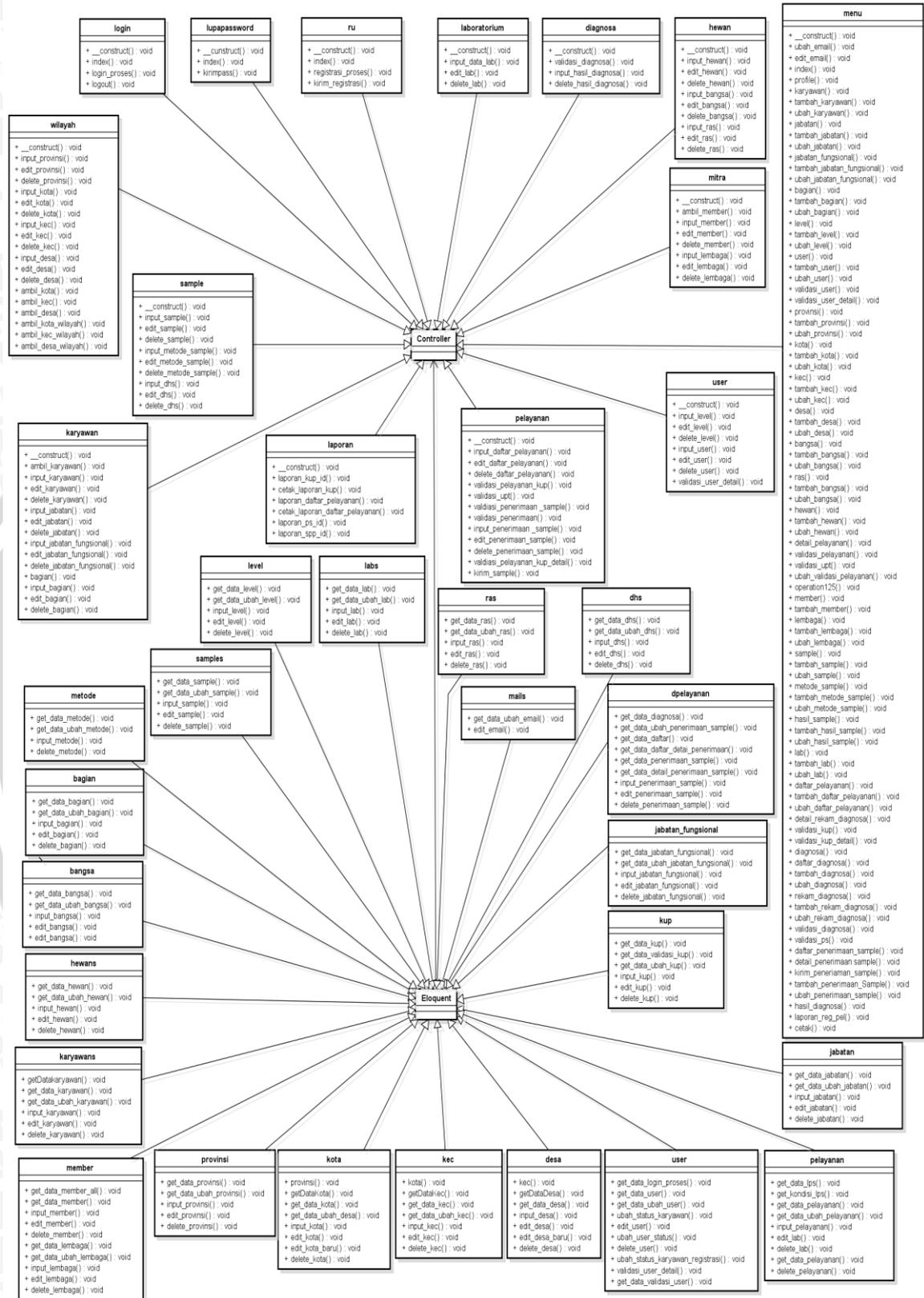
Pada Gambar 4.11 merupakan gambar *Sequence* Diagram validasi pelayanan menjelaskan interaksi aktor (super admin dan kepala UPT) dengan sistem dalam pengesahan data pelayanan.



**Gambar 4.11** *Sequence* Diagram Validasi Pelayanan

#### 4.6.4. Perancangan *Class* Diagram

Perancangan *Class* Diagram ini dilakukan untuk merancang *Class* yang nantinya akan diimplementasikan pada *Framework* Laravel yang menerapkan pola MVC. Dengan perancangan *Class* Diagram ini, maka akan mempermudah peneliti dalam mengembangkan sistem sesuai dengan kebutuhan pada *Class* yang ada di laravel 4.2 seperti pada Gambar 4.12.



Gambar 4.12 Class Diagram SIMLAB



#### 4.6.5. Perancangan Database SIMLAB

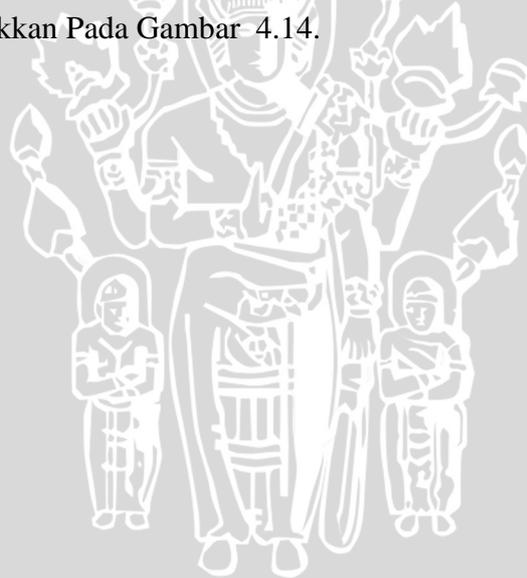
Perancangan basis data dilakukan agar SIMLAB menjadi basis data yang efisien dalam penggunaan ruang penyimpanan, cepat dalam pengaksesan. Perancangan basis data dapat dilakukan dengan menggunakan perancangan database, ERD, *Data Object Description* dan physical data model.

##### 1) Perancangan Database

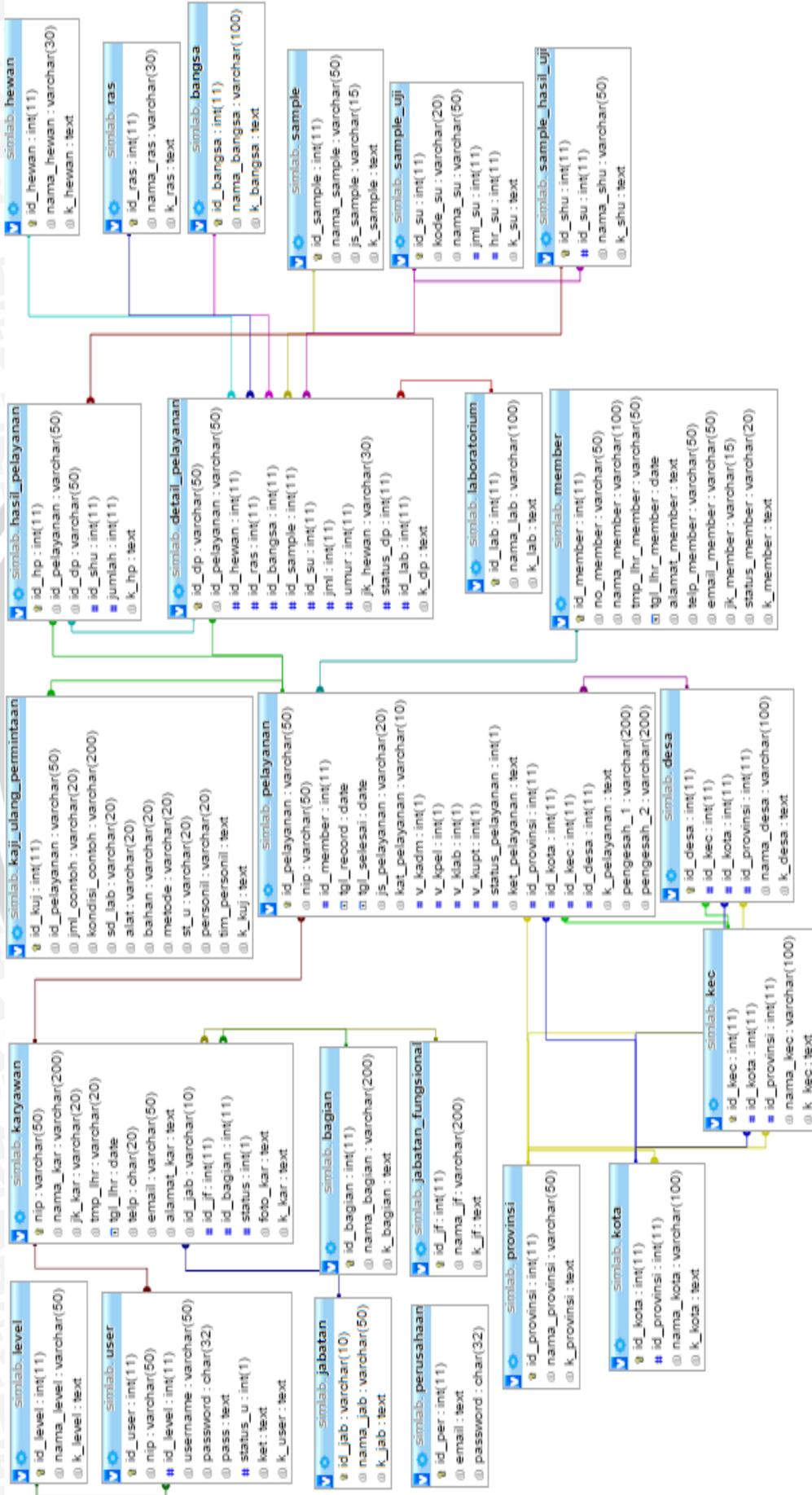
Untuk mendapatkan struktur tabel yang baik dan sesuai dengan kebutuhan SIMLAB, maka basis data dirancang dengan menggunakan desain database seperti pada Gambar 4.13.

##### 2) Entity Relationship Diagram (ERD)

ERD digunakan untuk menggambarkan entitas-entitas secara umum yang terdapat pada perancangan SIMLAB. Secara umum ERD dari struktur basis data pada SIMLAB ditunjukkan Pada Gambar 4.14.







Gambar 4.14 Perancangan Database SILAB

### 3) Perancangan *Data Object Description*

*Data Object Description* menjelaskan secara rinci mengenai atribut - atribut yang dimiliki oleh masing-masing tabel yang ada pada basis data sesuai dengan Entity Relationship Diagram. *Data Object Description* dapat dipisahkan menjadi dua bagian, yaitu *Data type definition* dan data field description.

#### a. *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 data terkait bagian satuan unit kerja di UPT Laboratorium Kesehatan Hewan Malang. *Data type definition* dari tabel bagian ditunjukkan pada Tabel 4.13.

**Tabel 4.13 Tabel *Data type definition* dari tabel bagian**

Key	Field	Type
PK	id_bagian	int(11)
	nama_bagian	varchar(200)
	k_bagian	Text

2. Tabel bangsa berisi data terkait klasifikasi dari hewan. *Data type definition* dari tabel bangsa ditunjukkan pada Tabel 4.14.

**Tabel 4.14 Tabel *Data type definition* dari tabel bangsa**

Key	Field	Type
PK	id_bangsa	int(11)
	nama_bangsa	varchar(100)
	k_bangsa	Text

3. Tabel desa berisi data terkait bagian wilayah penerimaan Sampel. *Data type definition* dari tabel desa ditunjukkan pada Tabel 4.15.

**Tabel 4.15 Tabel *Data type definition* dari tabel desa**

Key	Field	Type
PK	id_desa	int(11)
FK	id_kec	int(11)
FK	id_kota	int(11)
FK	id_provinsi	int(11)
	nama_desa	varchar(100)
	k_desa	Text

4. Tabel `detail_pelayanan` berisi data terkait detail dari proses pelayanan. *Data type definition* dari tabel `detail_pelayanan` ditunjukkan pada Tabel 4.16.

**Tabel 4.16** Tabel *Data type definition* dari tabel detail pelayanan

Key	Field	Type
<b>PK</b>	<code>id_dp</code>	<code>varchar(50)</code>
<b>FK</b>	<code>id_pelayanan</code>	<code>varchar(50)</code>
<b>FK</b>	<code>id_hewan</code>	<code>int(11)</code>
<b>FK</b>	<code>id_ras</code>	<code>int(11)</code>
<b>FK</b>	<code>id_bangsa</code>	<code>int(11)</code>
<b>FK</b>	<code>id_Sampel</code>	<code>int(11)</code>
<b>FK</b>	<code>id_su</code>	<code>int(11)</code>
<b>FK</b>	<code>id_shu</code>	<code>int(11)</code>
<b>FK</b>	<code>id_lab</code>	<code>int(11)</code>
	<code>Jml</code>	<code>int(11)</code>
	<code>Umur</code>	<code>int(11)</code>
	<code>jk_hewan</code>	<code>varchar(30)</code>
	<code>status_dp</code>	<code>int(11)</code>
	<code>k_dp</code>	<code>Text</code>

5. Tabel `hasil_pelayanan` berisi data terkait hasil dari proses pelayanan. *Data type definition* dari tabel `hasil_pelayanan` ditunjukkan pada Tabel 4.17.

**Tabel 4.17** Tabel *Data type definition* dari tabel hasil pelayanan

Key	Field	Type
<b>PK</b>	<code>id_hp</code>	<code>int(11)</code>
<b>FK</b>	<code>id_pelayanan</code>	<code>varchar(50)</code>
<b>FK</b>	<code>id_dp</code>	<code>varchar(50)</code>
<b>FK</b>	<code>id_shu</code>	<code>int(11)</code>
	<code>Jumlah</code>	<code>int(11)</code>
	<code>Hasil</code>	<code>varchar(20)</code>
	<code>k_hp</code>	<code>Text</code>

6. Tabel `hewan` berisi data terkait jenis hewan dari Sampel. *Data type definition* dari tabel `hewan` ditunjukkan pada Tabel 4.18.

**Tabel 4.18** Tabel *Data type definition* dari tabel hewan

Key	Field	Type
<b>PK</b>	<code>id_hewan</code>	<code>int(11)</code>
	<code>nama_hewan</code>	<code>varchar(30)</code>
	<code>k_hewan</code>	<code>text</code>

7. Tabel jabatan berisi data terkait identitas struktur golongan dan pangkat karyawan. *Data type definition* dari tabel jabatan ditunjukkan pada Tabel 4.19.

**Tabel 4.19** Tabel *Data type definition* dari tabel jabatan

Key	Field	Type
PK	id_jab	varchar(10)
	nama_jab	varchar(50)
	k_jab	Text

8. Tabel jabatan\_fungsional berisi data terkait tugas dan fungsi dari jabatan karyawan. *Data type definition* dari tabel jabatan\_fungsional ditunjukkan pada Tabel 4.20.

**Tabel 4.20** Tabel *Data type definition* dari tabel jabatan\_fungsional

Key	Field	Type
PK	id_jf	int(11)
	nama_jf	varchar(200)
	k_jf	Text

9. Tabel kaji\_ulang\_permintaan berisi data terkait formulir permintaan pengujian Sampel. *Data type definition* dari tabel kaji\_ulang\_permintaan ditunjukkan pada Tabel 4.21.

**Tabel 4.21** Tabel *Data type definition* dari tabel kaji\_ulang\_permintaan

Key	Field	Type
PK	id_kuj	int(11)
FK	id_pelayanan	varchar(50)
	jml_contoh	varchar(20)
	kondisi_contoh	varchar(200)
	sd_lab	varchar(20)
	Alat	varchar(20)
	Bahan	varchar(20)
	Metode	varchar(20)
	st_u	varchar(20)
	Personil	varchar(20)
	tim_personil	Text
	k_kuj	Text

10. Tabel karyawan berisi data terkait biodata karyawan. *Data type definition* dari tabel karyawan ditunjukkan pada Tabel 4.22.

**Tabel 4.22** Tabel *Data type definition* dari tabel karyawan

Key	Field	Type
<b>PK</b>	nip	varchar(50)
	nama_kar	varchar(200)
	jk_kar	varchar(20)
	tmp_lhr	varchar(20)
	tgl_lhr	Date
	telp	char(20)
	email	varchar(50)
	alamat_kar	Text
<b>FK</b>	id_jab	varchar(10)
<b>FK</b>	id_jf	int(11)
<b>FK</b>	id_bagian	int(11)
	Status	int(1)
	foto_kar	Text
	k_kar	Text

11. Tabel kec berisi data terkait wilayah dari pengujian Sampel. *Data type definition* dari tabel kec ditunjukkan pada Tabel 4.23.

**Tabel 4.23** Tabel *Data type definition* dari tabel kec

Key	Field	Type
<b>PK</b>	id_kec	int(11)
<b>FK</b>	id_kota	int(11)
<b>FK</b>	id_provinsi	int(11)
	nama_kec	varchar(100)
	k_kec	Text

12. Tabel kota berisi data terkait wilayah dari pengujian Sampel. *Data type definition* dari tabel kota ditunjukkan pada Tabel 4.24.

**Tabel 4.24** Tabel *Data type definition* dari tabel kota.

Key	Field	Type
<b>PK</b>	id_kota	int(11)
<b>FK</b>	id_provinsi	int(11)
	nama_kota	varchar(100)
	k_kota	Text

13. Tabel `laboratorium` berisi data terkait tempat pengujian Sampel atau laboratorium pelayanan yang ada di UPT Laboratorium Kesehatan Hewan. *Data type definition* dari tabel `laboratorium` ditunjukkan pada Tabel 4.25.

**Tabel 4.25** Tabel *Data type definition* dari tabel `laboratorium`

Key	Field	Type
PK	<code>id_lab</code>	<code>int(11)</code>
	<code>nama_lab</code>	<code>varchar(100)</code>
	<code>k_lab</code>	<code>text</code>

14. Tabel `level` berisi data terkait tingkatan hak akses user dalam penggunaan aplikasi. *Data type definition* dari tabel `level` ditunjukkan pada Tabel 4.26.

**Tabel 4.26** Tabel *Data type definition* dari tabel `level`

Key	Field	Type
PK	<code>id_level</code>	<code>int(11)</code>
	<code>nama_level</code>	<code>varchar(50)</code>
	<code>k_level</code>	<code>text</code>

15. Tabel `member` berisi data terkait member secara personal dan instansi. *Data type definition* dari tabel `member` ditunjukkan pada Tabel 4.27.

**Tabel 4.27** Tabel *Data type definition* dari tabel `member`

Key	Field	Type
PK	<code>id_member</code>	<code>int(11)</code>
	<code>no_member</code>	<code>varchar(50)</code>
	<code>nama_member</code>	<code>varchar(100)</code>
	<code>tmp_lhr_member</code>	<code>varchar(50)</code>
	<code>tgl_lhr_member</code>	<code>date</code>
	<code>alamat_member</code>	<code>text</code>
	<code>telp_member</code>	<code>varchar(50)</code>
	<code>email_member</code>	<code>varchar(50)</code>
	<code>jk_member</code>	<code>varchar(15)</code>
	<code>status_member</code>	<code>varchar(20)</code>
	<code>k_member</code>	<code>text</code>

16. Tabel pelayanan berisi data terkait . *Data type definition* dari tabel pelayanan ditunjukkan pada Tabel 4.28.

**Tabel 4.28** Tabel *Data type definition* dari tabel pelayanan

Key	Field	Type
<b>PK</b>	id_pelayanan	varchar(50)
<b>FK</b>	Nip	varchar(50)
<b>FK</b>	id_member	int(11)
	tgl_record	date
	tgl_selesai	date
	js_pelayanan	varchar(20)
	kat_pelayanan	varchar(10)
	v_kadm	int(1)
	v_kpel	int(1)
	v_klab	int(1)
	v_kupt	int(1)
	status_pelayanan	int(1)
	ket_pelayanan	text
<b>FK</b>	id_provinsi	int(11)
<b>FK</b>	id_kota	int(11)
<b>FK</b>	id_kec	int(11)
<b>FK</b>	id_desa	int(11)
	pengesah_1	varchar(200)
	pengesah_2	varchar(200)
	k_pelayanan	text

17. Tabel perusahaan berisi data terkait email pengiriman pesan sebagai sarana pemberitahuan kepada karyawan atau member . *Data type definition* dari tabel perusahaan ditunjukkan pada Tabel 4.29.

**Tabel 4.29** Tabel *Data type definition* dari tabel perusahaan

Key	Field	Type
<b>PK</b>	id_per	int(11)
	email	text
	password	char(32)

18. Tabel provinsi berisi data terkait wilayah dari pengujian Sampel. *Data type definition* dari tabel provinsi ditunjukkan pada Tabel 4.30.

**Tabel 4.30** Tabel *Data type definition* dari tabel provinsi

Key	Field	Type
<b>PK</b>	id_provinsi	int(11)
	nama_provinsi	varchar(50)
	k_provinsi	text

19. Tabel `ras` berisi data terkait klasifikasi dari hewan. *Data type definition* dari tabel `ras` ditunjukkan pada Tabel 4.31.

**Tabel 4.31** Tabel *Data type definition* dari tabel `ras`

Key	Field	Type
<b>PK</b>	<code>id_ras</code>	<code>int(11)</code>
	<code>nama_ras</code>	<code>varchar(30)</code>
	<code>k_ras</code>	<code>text</code>

20. Tabel `Sampel` berisi data terkait Sampel dari hewan baik berupa Bahan Asal Hewan (Bah) atau Bahan Asal Hasil Hewan. *Data type definition* dari tabel `Sampel` ditunjukkan pada Tabel 4.32.

**Tabel 4.32** Tabel *Data type definition* dari tabel `Sampel`

Key	Field	Type
<b>PK</b>	<code>id_Sampel</code>	<code>int(11)</code>
	<code>nama_Sampel</code>	<code>varchar(50)</code>
	<code>js_Sampel</code>	<code>varchar(15)</code>
	<code>k_Sampel</code>	<code>text</code>

21. Tabel `Sampel_hasil_uji` berisi data terkait hasil metode pengujian dari `Sampel`. *Data type definition* dari tabel `Sampel_hasil_uji` ditunjukkan pada Tabel 4.33.

**Tabel 4.33** Tabel *Data type definition* dari tabel `Sampel_hasil_uji`

Key	Field	Type
<b>PK</b>	<code>id_shu</code>	<code>int(11)</code>
<b>FK</b>	<code>id_su</code>	<code>int(11)</code>
	<code>nama_shu</code>	<code>varchar(50)</code>
	<code>k_shu</code>	<code>text</code>

22. Tabel `Sampel_uji` berisi data terkait metode pengujian dari Sampel.

*Data type definition* dari tabel `Sampel_uji` ditunjukkan pada Tabel 4.34.

**Tabel 4.34** Tabel *Data type definition* dari tabel `Sampel_uji`

Key	Field	Type
<b>PK</b>	<code>id_su</code>	<code>int(11)</code>
	<code>nama_su</code>	<code>varchar(50)</code>
	<code>jml_su</code>	<code>int(11)</code>
	<code>hr_su</code>	<code>int(11)</code>
	<code>k_su</code>	<code>text</code>

23. Tabel `user` berisi data terkait hak akses berdasarkan data user yang telah ditetapkan. *Data type definition* dari tabel `user` ditunjukkan pada Tabel 4.35.

**Tabel 4.35** Tabel *Data type definition* dari tabel `user`

Key	Field	Type
<b>PK</b>	<code>id_user</code>	<code>int(11)</code>
<b>FK</b>	<code>nip</code>	<code>varchar(50)</code>
	<code>id_level</code>	<code>int(11)</code>
	<code>username</code>	<code>varchar(50)</code>
	<code>password</code>	<code>char(32)</code>
	<code>pass</code>	<code>text</code>
	<code>status_u</code>	<code>int(1)</code>
	<code>ket</code>	<code>text</code>
	<code>k_user</code>	<code>text</code>

#### b. Data Field Description

Data field description menjelaskan keterangan seluruh kolom (field) dalam masing-masing tabel basis data sistem. Data field description basis data sistem Informasi Manajemen Laboratorium (SIMLAB) pada UPT Laboratorium Kesehatan Hewan Malang dijelaskan sebagai berikut:

1. Tabel `bagian` berisi data terkait bagian satuan unit kerja di UPT Laboratorium Kesehatan Hewan Malang. *Data field description* dari tabel `bagian` ditunjukkan pada Tabel 4.36.

Tabel 4.36 Tabel *Data filed description* dari tabel bagian

Table Name	No	Field	Description
bagian	1	<b>id_bagian</b>	Sebagai id bagian unit kerja
	2	<b>nama_bagian</b>	Nama bagian unit kerja
	3	<b>k_bagian</b>	Sebagai id yang dienkripsi

2. Tabel bangsa berisi data terkait klasifikasi dari hewan. *Data filed description* dari tabel bangsa ditunjukkan pada Tabel 4.37.

Tabel 4.37 Tabel *Data filed description* dari tabel bangsa

Table Name	No	Field	Description
bangsa	1	<b>id_bangsa</b>	Sebagai id bangsa dari hewan
	2	<b>nama_bangsa</b>	Nama bangsa dari hewan
	3	<b>k_bangsa</b>	Sebagai id yang dienkripsi

3. Tabel desa berisi data terkait bagian wilayah penerimaan Sampel. *Data filed description* dari tabel desa ditunjukkan pada Tabel 4.38.

Tabel 4.38 Tabel *Data filed description* dari tabel desa

Table Name	No	Field	Description
desa	1	<b>id_desa</b>	Sebagai id wilayah daerah desa
	2	<b>id_kec</b>	Sebagai id wilayah daerah kecamatan
	3	<b>id_kota</b>	Sebagai id wilayah daerah kota
	4	<b>id_provinsi</b>	Sebagai id wilayah daerah provinsi
	5	<b>nama_desa</b>	Nama wilayah daerah desa
	6	<b>k_desa</b>	Sebagai id yang dienkripsi

4. Tabel detail\_pelayanan berisi data terkait detail dari proses pelayanan. *Data filed description* dari tabel detail\_pelayanan ditunjukkan pada Tabel 4.39.

Tabel 4.39 Tabel *Data filed description* dari tabel detail pelayanan

Table Name	No	Field	Description
detail_pelayanan	1	<b>id_dp</b>	Sebagai id detail pelayanan
	2	<b>id_pelayanan</b>	Sebagai id pelayanan
	3	<b>id_hewan</b>	Sebagai id hewan
	4	<b>id_ras</b>	Sebagai id ras
	5	<b>id_bangsa</b>	Sebagai id bangsa
	6	<b>id_sampel</b>	Sebagai id Sampel
	7	<b>id_su</b>	Sebagai id metode uj
	8	<b>id_shu</b>	Sebagai id hasil metode uji
	9	<b>id_lab</b>	Sebagai id laboratorium
	10	<b>Jml</b>	Jumlah Sampel
	11	<b>Umur</b>	Umur hewan sebagai Sampel

	12	<b>jk_hewan</b>	Jenis kelamin dari hewan
	13	<b>status_dp</b>	Status detail pelayanan
	14	<b>k_dp</b>	Sebagai id yang dienkrripsi

5. Tabel hasil\_pelayanan berisi data terkait hasil dari proses pelayanan. *Data filed description* dari tabel hasil\_pelayanan ditunjukkan pada Tabel 4.40.

**Tabel 4.40 Tabel Data filed description dari tabel hasil pelayanan**

Table Name	No	Field	Description
hasil_pelayanan	1	<b>id_hp</b>	Sebagai id dari hasil pelayanan
	2	<b>id_pelayanan</b>	Sebagai id pelayanan
	3	<b>id_dp</b>	Sebagai id detail detail pelayanan
	4	<b>id_shu</b>	Sebagai id dari hasil pelayanan
	5	<b>Jumlah</b>	Jumlah dari hasil pelayanan
	6	<b>Hasil</b>	Hasil pelayanan
	7	<b>k_hp</b>	Sebagai id yang dienkrripsi

6. Tabel hewan berisi data terkait jenis hewan dari Sampel. *Data filed description* dari tabel hewan ditunjukkan pada tabel 4.41.

**Tabel 4.41 Tabel Data filed description dari tabel hewan**

Table Name	No	Field	Description
hewan	1	<b>id_hewan</b>	Sebagai id dari hewan
	2	<b>nama_hewan</b>	Nama dari hewan
	3	<b>k_hewan</b>	Sebagai id yang dienkrripsi

7. Tabel jabatan berisi data terkait identitas struktur golongan dan pangkat karyawan. *Data filed description* dari tabel jabatan pada Tabel 4.42..

**Tabel 4.42 Tabel Data filed description dari tabel jabatan**

Table Name	No	Field	Description
jabatan	1	<b>id_jab</b>	Sebagai id golongan dan pangkat dari karyawan
	2	<b>nama_jab</b>	Nama golongan dan pangkat dari karyawan
	3	<b>k_jab</b>	Sebagai id yang dienkrripsi

8. Tabel jabatan\_fungsional berisi data terkait tugas dan fungsi dari jabatan karyawan. *Data filed description* dari tabel jabatan\_fungsional ditunjukkan pada Tabel 4.43.

**Tabel 4.43 Tabel Data filed description dari tabel jabatan\_fungsional**

Table Name	No	Field	Description
Jabatan_fungsional	1	id_jf	Sebagai id jabatan fungsional kerja dari karyawan
	2	nama_jf	Nama jabatan fungsional kerja dari karyawan
	3	k_jf	Sebagai id yang dienkrpsi

9. Tabel *kaji\_ulang\_permintaan* berisi data terkait formulir permintaan pengujian Sampel. *Data filed description* dari tabel *kaji\_ulang\_permintaan* ditunjukkan pada Tabel 4.44.

**Tabel 4.44 Tabel Data filed description dari tabel kaji\_ulang\_permintaan**

Table Name	No	Field	Description
kaji_ulang_permintaan	1	id_kuj	Sebagai id kaji ulang permintaan
	2	id_pelayanan	Sebagai id kaji ulang permintaan
	3	jml_contoh	Jumlah contoh Sampel masuk
	4	kondisi_contoh	Kondisi Sampel masuk
	5	sd_lab	Keterangan tersedianya lab
	6	Alat	Keterangan tersedianya alat
	7	Bahan	Keterangan tersedianya bahan
	8	Metode	Keterangan metode pengujian
	9	st_u	Keterangan standart uji
	10	Personil	Keterangan tersedianya lab
	11	tim_personil	Nama personil pengujian
	12	k_kuj	Sebagai id yang dienkrpsi

10. Tabel *karyawan* berisi data terkait biodata karyawan. *Data filed description* dari tabel *karyawan* ditunjukkan pada Tabel 4.45.

**Tabel 4.45 Tabel Data filed description dari tabel karyawan**

Table Name	No	Field	Description
karyawan	1	nip	Sebagai id karyawan
	2	nama_kar	Nama dari karyawan
	3	jk_kar	Jenis kelamin dari karyawan
	4	tmp_lhr	Tempat asal lahir karyawan
	5	tgl_lhr	Tanggal lahir karyawan
	6	telp	Telepon yang dapat dihubungi
	7	email	Alamat email karyawan
	8	alamat_kar	Alamat tinggal karyawan
	9	id_jab	Sebagai id jabatan
	10	id_jf	Sebagai id jabatan fungsional
	11	id_bagian	Sebagai id bagian unit kerja
	12	Status	Status karyawan
	13	foto_kar	Foto karyawan
	14	k_kar	Sebagai id yang dienkrripsi

11. Tabel kec berisi data terkait wilayah dari pengujian Sampel. *Data filed description* dari tabel kec ditunjukkan pada tabel 4.46.

**Tabel 4.46 Tabel Data filed description dari tabel kec**

Table Name	No	Field	Description
kec	1	id_kec	Sebagai id wilayah daerah kamatan
	2	id_kota	Sebagai id wilayah daerah kota
	3	id_provinsi	Sebagai id wilayah daerah provinsi
	4	nama_kec	Nama wilayah kecamatan
	5	k_kec	Sebagai id yang dienkrripsi

12. Tabel kota berisi data terkait wilayah dari pengujian Sampel. *Data filed description* dari tabel kota ditunjukkan pada tabel 4.47.

**Tabel 4.47 Tabel Data filed description dari tabel kota.**

Table Name	No	Field	Description
kota	1	id_kota	Sebagai id wilayah daerah kota
	2	id_provinsi	Sebagai id wilayah daerah provinsi
	3	nama_kota	Nama wilayah kota
	4	k_kota	Sebagai id yang dienkrripsi

13. Tabel laboratorium berisi data terkait tempat pengujian Sampel atau laboratorium pelayanan yang ada di UPT Laboratorium Kesehatan Hewan.

*Data filed description* dari tabel laboratorium ditunjukkan pada tabel 4.48

**Tabel 4.48 Tabel Data filed description dari tabel laboratorium**

Table Name	No	Field	Description
Laboratorium	1	id_lab	Sebagai id laboratorium
	2	nama_lab	Nama Laboratorium
	3	k_lab	Sebagai id yang dienkripsi

14. Tabel level berisi data terkait tingkatan hak akses user dalam penggunaan aplikasi. *Data filed description* dari tabel level ditunjukkan pada tabel 4.49

**Tabel 4.49 Tabel Data filed description dari tabel level**

Table Name	No	Field	Description
level	1	id_level	Sebagai id level akses
	2	nama_level	Nama level akses
	3	k_level	Sebagai id yang dienkripsi

15. Tabel member berisi data terkait member secara personal dan instansi.

*Data filed description* dari tabel member ditunjukkan pada tabel 4.50.

**Tabel 4.50 Tabel Data filed description dari tabel member**

Table Name	No	Field	Description
member	1	id_member	Sebagai id member
	2	no_member	Nomor identitas / lembaga
	3	nama_member	Nama member
	4	tmp_lhr_member	Tempat lahir member
	5	tgl_lhr_member	Tanggal lahir member
	6	alamat_member	Alamat member
	7	telp_member	Telepon member
	8	email_member	Alamat email member
	9	jk_member	Jenis kelamin member
	10	status_member	Status member
	11	k_member	Sebagai id yang dienkripsi

16. Tabel pelayanan berisi data terkait daftar pelayanan. *Data filed description* dari tabel pelayanan ditunjukkan pada tabel 4.51.

**Tabel 4.51** Tabel *Data filed description* dari tabel pelayanan

Table Name	No	Field	Description
Pelayanan	1	id_pelayanan	Sebagai id pelayanan
	2	Nip	Sebagai id karyawan
	3	id_member	Sebagai id member
	4	tgl_record	Tanggal Record Pelayanan
	5	tgl_selesai	Tanggal Selesai Pelayanan
	6	js_pelayanan	Jenis Pelayanan
	7	kat_pelayanan	Kategori Pelayanan
	8	v_kadm	Status validasi Administratif
	9	v_kpel	Status validasi Kepala Pelayanan
	10	v_klab	Status validasi Laboran
	11	v_kupt	Status validasi Kepala UPT
	12	status_pelayanan	Status Pelayanan
	13	ket_pelayanan	Keterangan pelayanan
	14	id_provinsi	Sebagai id provinsi
	15	id_kota	Sebagai id kota
	16	id_kec	Sebagai id kecamatan
	17	id_desa	Sebagai id desa
	18	Pengesah_1	Data Pengesah Pelayanan kepala pelayanan
	19	Pengesah_2	Data Pengesah Pelayanan kepala UPT
	20	k_pelayanan	Sebagai id yang dienkrpsi

17. Tabel perusahaan berisi data terkait email pengiriman pesan sebagai sarana pemberitahuan kepada karyawan atau member . *Data filed description* dari tabel perusahaan ditunjukkan pada tabel 4.52.

**Tabel 4.52** Tabel *Data filed description* dari tabel perusahaan

Table Name	No	Field	Description
Perusahaan	1	id_per	Sebagai id perusahaan
	2	Email	Alamat email yang digunakan mengirim pesan perubahan password dan hasil pelayanan
	3	Password	Sebagai id yang dienkrpsi

18. Tabel provinsi berisi data terkait wilayah dari pengujian Sampel. *Data filed description* dari tabel provinsi ditunjukkan pada tabel 4.53.

**Tabel 4.53 Tabel Data filed description dari tabel provinsi**

Table Name	No	Field	Description
Provinsi	1	id_provinsi	Sebagai id wilayah daerah provinsi
	2	nama_provinsi	Nama wilayah dari provinsi
	3	k_provinsi	Sebagai id yang dienkripsi

19. Tabel ras berisi data terkait klasifikasi dari hewan. *Data filed description* dari tabel ras ditunjukkan pada tabel 4.54.

**Tabel 4.54 Tabel Data filed description dari tabel ras**

Table Name	No	Field	Description
ras	1	id_ras	Sebagai id ras
	2	nama_ras	Nama dari ras
	3	k_ras	Sebagai id yang dienkripsi

20. Tabel Sampel berisi data terkait Sampel dari hewan baik berupa Bahan Asal Hewan (Bah) atau Bahan Asal Hasil Hewan. *Data filed description* dari tabel Sampel ditunjukkan pada tabel 4.55.

**Tabel 4.55 Tabel Data filed description dari tabel sampel**

Table Name	No	Field	Description
Sampel	1	id_Sampel	Sebagai dari Sampel
	2	nama_Sampel	Nama dari Sampel
	3	js_Sampel	Jenis Sampel dari pelayanan kesmavet atau keswan
	4	k_Sampel	Sebagai id yang dienkripsi

21. Tabel Sampel\_hasil\_uji berisi data terkait hasil metode pengujian dari Sampel. *Data filed description* dari tabel Sampel\_hasil\_uji ditunjukkan pada tabel 4.56.

**Tabel 4.56 Tabel Data filed description dari tabel sampel\_hasil\_uji**

Table Name	No	Field	Description
Sampel_hasil_uji	1	id_shu	Sebagai id dari hasil Sampel
	2	id_su	Sebagai dari metode pengujian
	3	nama_shu	Nama dari hasil pengujian
	4	k_shu	Sebagai id yang dienkripsi

22. Tabel Sampel\_uji berisi data terkait metode pengujian dari Sampel. *Data filed description* dari tabel Sampel\_uji ditunjukkan pada tabel 4.57.

**Tabel 4.57** Tabel *Data filed description* dari tabel sampel\_uji

Table Name	No	Field	Description
Sampel_uji	1	id_su	Sebagai id dari metode pengujian
	2	nama_su	Nama dari Sampel pengujian
	3	jml_su	Jumlah batas dari Sampel pengujian
	4	hr_su	Periode Sampel berdasarkan jumlah Sampel
	5	k_su	Sebagai id yang dienkripsi

23. Tabel user berisi data terkait hak akses berdasarkan data user yang telah ditetapkan. *Data filed description* dari tabel user ditunjukkan pada tabel 4.58.

**Tabel 4.58** Tabel *Data filed description* dari tabel user

Table Name	No	Field	Description
user	1	id_user	Sebagai id dari user
	2	nip	NIP dari karyawan
	3	id_level	Sebagia id dari level
	4	username	Username alamat akun
	5	password	Passsword alamat akun yang ter enkripsi
	6	pass	Passsword alamat akun
	7	status_u	Status dari user
	8	ket	Keterangan user
	9	k_user	Sebagai id yang dienkripsi

#### 4.6.6. Perancangan Antarmuka

Pada bagian ini akan dijelaskan tentang perancangan antarmuka aplikasi SIMLAB. Antarmuka perangkat lunak ini digambarkan dengan desain web yang nantinya akan menjadi tampilan website SIMLAB. Berikut merupakan rancangan antarmuka SIMLAB.

##### 1. Rancangan Antarmuka Halaman Login

Halaman Login merupakan halaman awal sistem yang menampilkan form login dengan isian berupa *username* dan *password*, dan sebuah tombol Login.

Berikut ini merupakan rancangan antarmuka halaman Login.

**Gambar 4.45 Rancangan Antarmuka Halaman Login**

**2. Rancangan Antarmuka Halaman Beranda**

Halaman beranda merupakan halaman awal setelah user melakukan login. Halaman beranda ini menampilkan halaman awal setelah melakukan login. Berikut ini merupakan rancangan antarmuka halaman beranda.

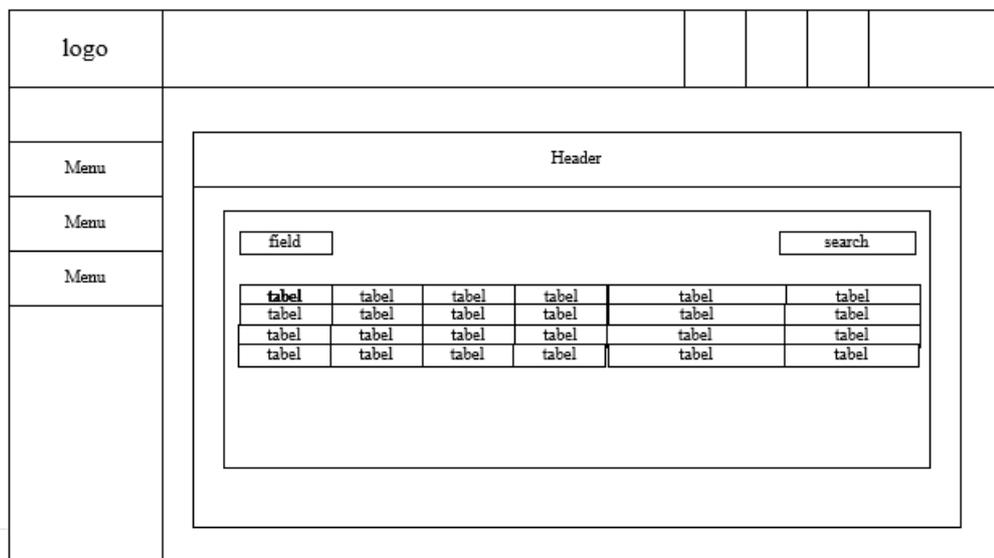
logo															
Menu	<table border="1"> <tr> <td colspan="5">Header</td> </tr> <tr> <td colspan="5">Konten</td> </tr> </table>					Header					Konten				
Header															
Konten															
Menu															
Menu															

**Gambar 4.46 Perancangan Antarmuka Halaman Beranda**

**3. Rancangan Antarmuka Halaman Tabel Data**

Halaman tabel data merupakan tampilan untuk menampung data yang telah tersimpan. Berikut ini merupakan rancangan antarmuka tabel data.

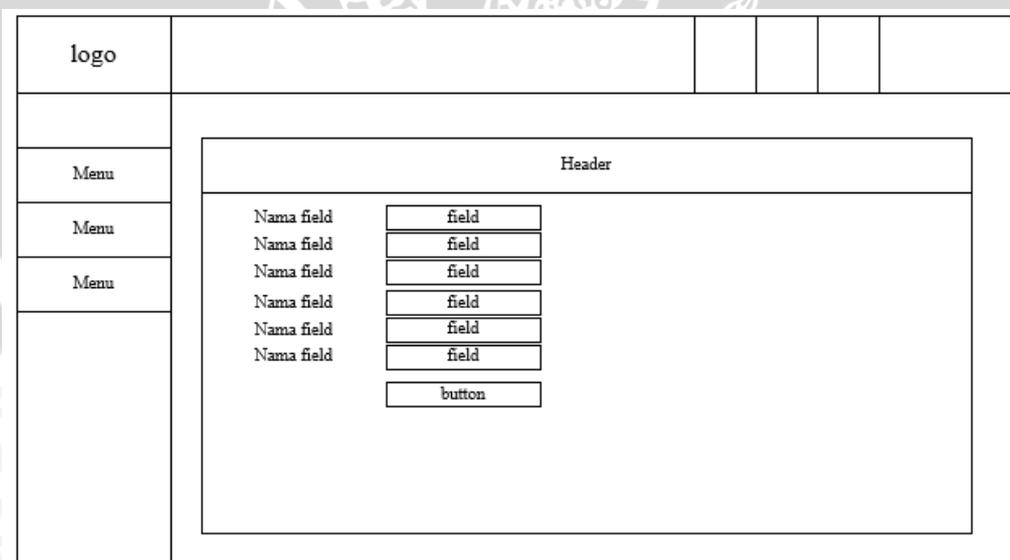




Gambar 4.46 Perancangan Antarmuka Halaman Tabel Data

4. Rancangan Antarmuka Halaman Form Input dan Edit

Halaman form input dan edit merupakan tampilan untuk memasukan dan mengubah data. Berikut ini merupakan rancangan antarmuka form input dan edit.



Gambar 4.46 Perancangan Antarmuka Halaman Form Input dan Edit

### 5. Rancangan Antarmuka Halaman Form Cetak Laporan

Halaman form cetak laporan merupakan tampilan untuk mencetak data rekap laporan proses pelayanan. Berikut ini merupakan rancangan antarmuka form cetak laporan.

logo				
	Header			
Menu	Nama field	field		
Menu	Nama field	pilih	v	
Menu		button		

**Gambar 4.46 Perancangan Antarmuka Halaman Form Cetak Laporan**

## BAB V

### IMPLEMENTASI

Bab ini membahas tahapan implementasi Sistem Informasi Manajemen Laboratorium (SIMLAB). Pembahasan terdiri atas penjelasan tentang spesifikasi sistem, batasan-batasan dalam implementasi, dan implementasi antarmuka.

#### 5.1 Spesifikasi Sistem

Hasil analisis kebutuhan dan perancangan perangkat lunak yang telah diuraikan pada bab 4 menjadi acuan untuk melakukan implementasi menjadi sistem yang dapat berfungsi sesuai dengan kebutuhan. Sistem Informasi Manajemen Laboratorium (SIMLAB) dibuat dalam lingkungan implementasi yang terdiri dari lingkungan perangkat keras dan perangkat lunak.

##### 5.1.1. Spesifikasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan dalam proses pembuatan Sistem Informasi Manajemen Laboratorium (SIMLAB) dijelaskan pada tabel berikut.

**Tabel 5.1 Spesifikasi Perangkat Keras**

Nama Komponen	Spesifikasi
Processor	Intel(R) Core(TM) i5-4210U CPU @ 1.70GHz
RAM	4 GB DDR3
Harddisk	500 GB SATA3
System Manufacturer	ASUSTeK COMPUTER INC
VGA Card	NVIDIA GeForce 820M

##### 5.1.2. Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam proses pembuatan Sistem Informasi Manajemen Laboratorium (SIMLAB) dijelaskan pada tabel berikut.

Tabel 5.2 Spesifikasi Perangkat Lunak

Perangkat Lunak	Spesifikasi
Sistem Operasi	Windows 8.1 Pro 64-bit
Bahasa Pemrograman	PHP extension: mysql ( <i>Framework</i> Laravel 4.2)
Basis Data	MySQL client version: 5.6.21
Web Server	Apache/2.4.10 (Win32) OpenSSL/1.0.1i PHP/5.6.3
Database Administrator Tool	PHPMyAdmin

## 5.2 Batasan-Batasan Implementasi

Beberapa batasan dalam mengimplementasikan Sistem Informasi Manajemen Laboratorium (SIMLAB) adalah sebagai berikut:

1. Input yang diterima oleh sistem berupa data terkait biodata karyawan, Member serta pelayanan pengujian Sampel.
2. Output yang diterima pengguna berupa hasil pelayanan.
3. Platform pembuatan yang digunakan adalah PHP dengan *Framework* Laravel 4.2.
4. Database disimpan dalam MySQL.
5. Sistem bersifat dinamis, yaitu menyediakan fasilitas insert, update, dan delete.

## 5.3 Implementasi Antarmuka

Implementasi perancangan antarmuka dilakukan sesuai perancangan yang sudah dibuat sebelumnya. Implementasi perancangan antarmuka Sistem Informasi Manajemen Laboratorium (SIMLAB) menggunakan hasil langsung capture halaman dari aplikasi tersebut.

### 5.3.1. Implementasi Antarmuka Halaman Login

Admin sistem (Aktor) yang akan menggunakan aplikasi pada Sistem Informasi Manajemen (SIMLAB) harus melalui proses autentifikasi login. Halaman login mempunyai dua kolom untuk username dan password. Implementasi antarmuka kebutuhan fungsional login ditunjukkan pada Gambar 5.1.

**Gambar 5.1 Implementasi Antarmuka Halaman Login**

### 5.3.2. Implementasi Antarmuka Halaman Kelola Karyawan

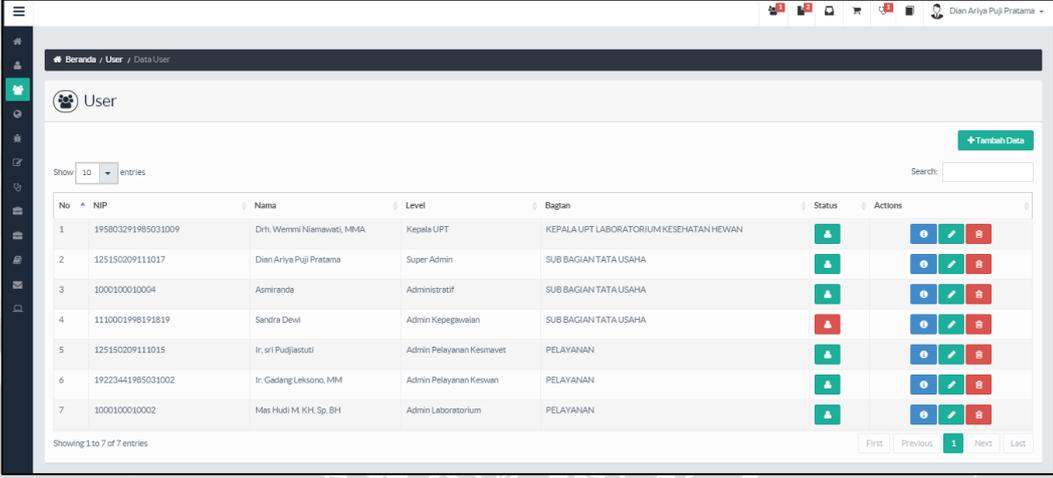
Berikut adalah halaman yang digunakan oleh aktor (Super admin dan Admin Kepegawaian) untuk melihat dan mengelola data karyawan. Implementasi antarmuka halaman jadwal ketersediaan ditunjukkan pada Gambar 5.2.

No	NIP	Nama	Gol/Ruang	Fungsional	Actions
1	195803291985001009	Drh. Wernni Niamawati, MHA	IV/d	KEPALA UPT LABORATORIUM KESEHATAN HEWAN	[Edit] [Delete]
2	125150209111017	Dian Ariya Puji Pratama	III/d	STAF TEKNIK	[Edit] [Delete]
3	1000100010004	Asmiranda	III/b	KEPALA SUB BAGIAN TATA USAHA	[Edit] [Delete]
4	111000198191819	Sandra Dewi	IV/b	STAF ADMINISTRASI	[Edit] [Delete]
5	1000100010002	Mas Hudi M. KH. Sp. BH	IV/a	KEPALA LABORATORIUM	[Edit] [Delete]
6	19223441985031002	Ir. Gedang Leksono, MM	III/b	KEPALA SEKSI PELAYANAN KESEHATAN HEWAN	[Edit] [Delete]
7	125150209111015	Ir. Sri Pudjastuti	III/c	KEPALA SEKSI PELAYANAN KESEHATAN MASYARAKAT VETERINER	[Edit] [Delete]
8	11100019819181912	Drh. Samsula Prilwantini	IV/c	KEPALA PELAYANAN	[Edit] [Delete]

**Gambar 5.2 Implementasi Antarmuka Halaman Kelola Karyawan**

### 5.3.3. Implementasi Antarmuka Halaman Kelola User

Berikut adalah halaman yang digunakan oleh aktor (Super admin) untuk melihat dan mengelola data user. Halaman ini menentukan hak akses user lain berdasarkan level akses yang diberikan oleh super admin. Implementasi antarmuka halaman kelola data user ditunjukkan pada Gambar 5.3.

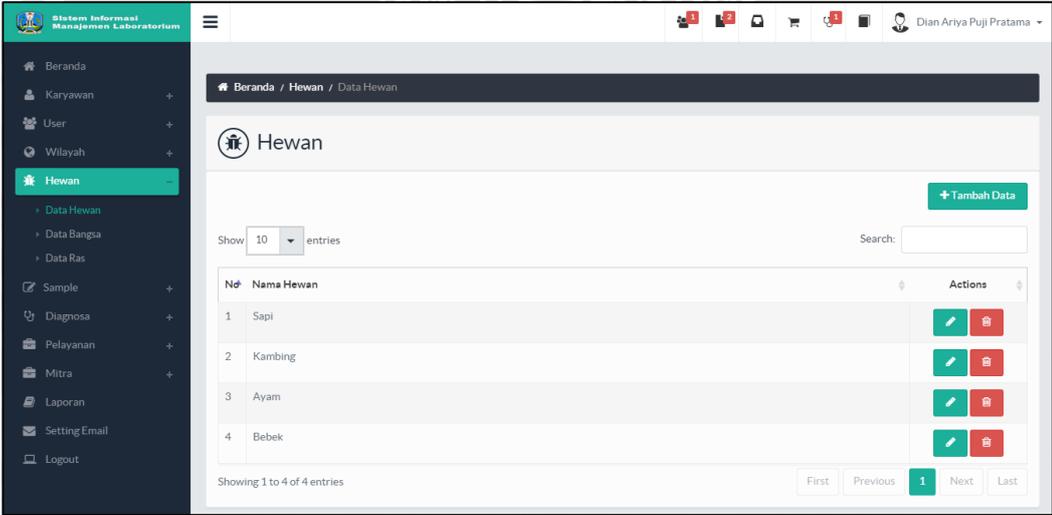


No	NIP	Nama	Level	Bagian	Status	Actions
1	195903291985031009	Drh. Werni Niamawati, MMA	Kepala UPT	KEPALA UPT LABORATORIUM KESKIHATAN HEWAN	▲	✎ ✖
2	125150209111017	Dian Ariya Puji Pratama	Super Admin	SUB BAGIAN TATA USAHA	▲	✎ ✖
3	1000100010004	Asmiranda	Administratif	SUB BAGIAN TATA USAHA	▲	✎ ✖
4	1110001998191819	Sandra Devi	Admin Kepegawaian	SUB BAGIAN TATA USAHA	▲	✎ ✖
5	125150209111015	Ir. Sri Pudjastuti	Admin Pelayanan Kesmavet	PELAYANAN	▲	✎ ✖
6	19223441985031002	Ir. Gading Leksono, MM	Admin Pelayanan Keswan	PELAYANAN	▲	✎ ✖
7	1000100010002	Mas Hudi M.KH.Sp. BH	Admin Laboratorium	PELAYANAN	▲	✎ ✖

**Gambar 5.3 Implementasi Antarmuka Halaman Kelola User**

### 5.3.4. Implementasi Antarmuka Halaman Kelola data Hewan

Berikut adalah halaman yang digunakan oleh aktor (Super admin kesmavet dan keswan) untuk melihat dan mengelola data hewan yang akan menjadi Sampel pengujian. Implementasi antarmuka halaman kelola data hewan ditunjukkan pada Gambar 5.3.

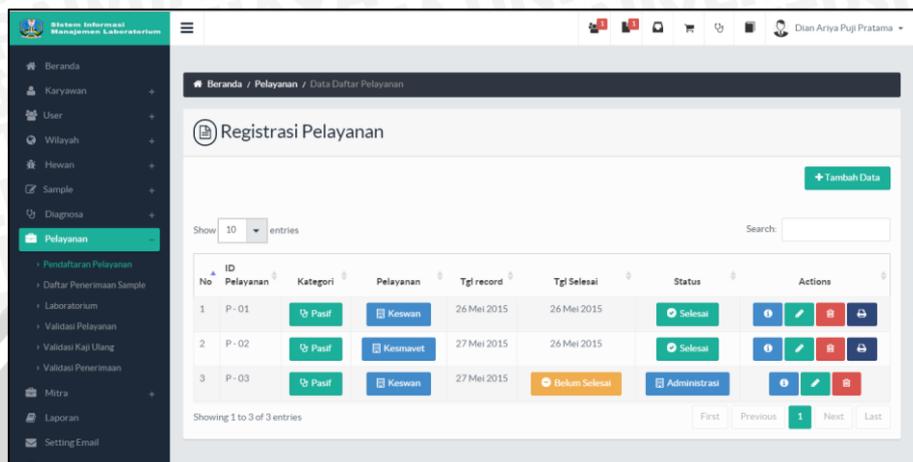


No	Nama Hewan	Actions
1	Sapi	✎ ✖
2	Kambing	✎ ✖
3	Ayam	✎ ✖
4	Bebek	✎ ✖

**Gambar 5.3 Implementasi Antarmuka Halaman Kelola User**

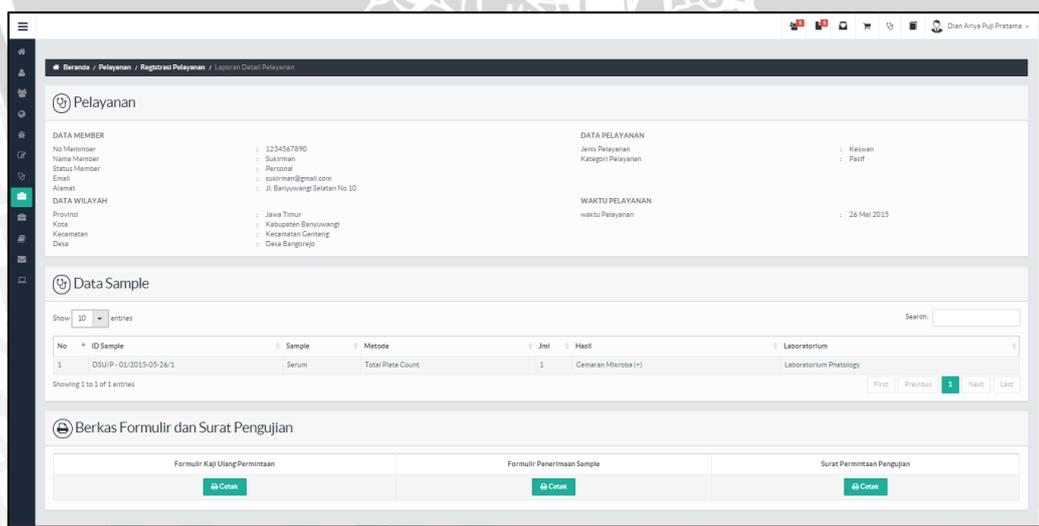
### 5.3.5. Implementasi Antarmuka Halaman Pendaftaran Pelayanan

Berikut adalah halaman yang digunakan oleh aktor (Super admin dan Admin Kepegawaian) untuk melihat dan mengelola data karyawan. Implementasi antarmuka halaman jadwal ketersediaan ditunjukkan pada Gambar 5.4.



**Gambar 5.4 Implementasi Antarmuka Halaman Registrasi Pelayanan**

Apabila telah selesai proses pelayanan bisa di lihat detail pelayanan seperti Pada Gambar 5.5. Kemudian aktor bisa mencetak formulir dan surat sesuai dengan kebutuhan. Implementasi antarmuka halaman detail pendaftaran pelayanan sebagai berikut.

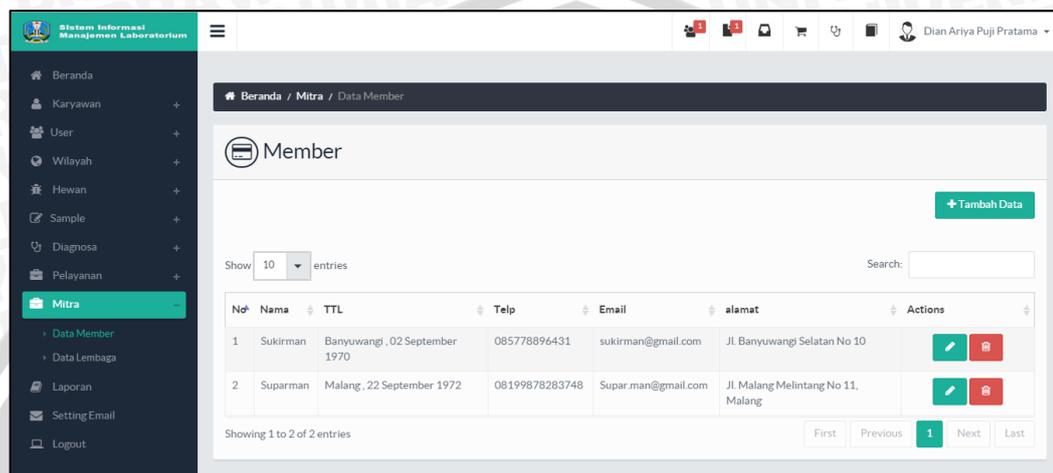


**Gambar 5.5 Implementasi Antarmuka Halaman Registrasi Pelayanan (Detail Pelayanan)**



### 5.3.6. Implementasi Antarmuka Halaman Kelola Mitra

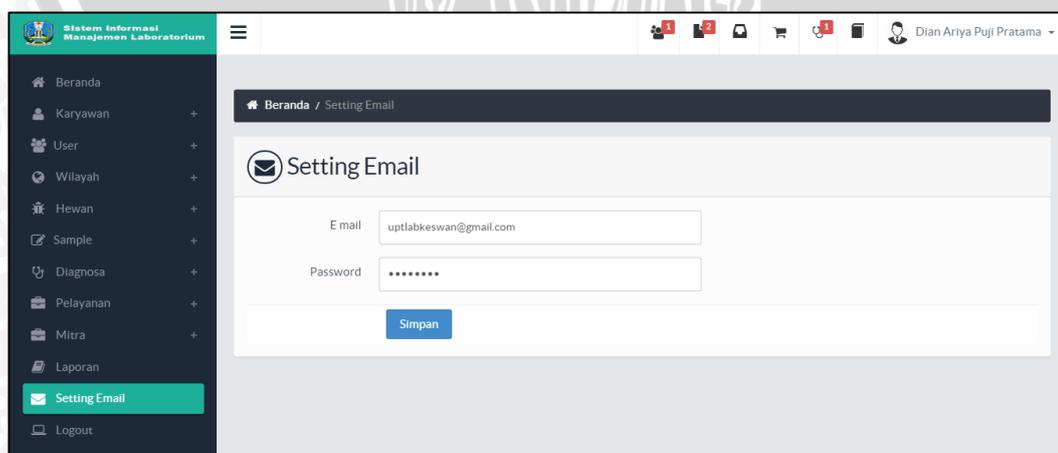
Berikut adalah halaman yang digunakan oleh aktor (Super admin dan Administrasitif) untuk melihat dan mengelola data member personal maupun lembaga. Pada Gambar 5.6 merupakan halaman yang mengelola data member personal.



**Gambar 5.6 Implementasi Antarmuka Halaman Kelola Mitra (Member Personal)**

### 5.3.7. Implementasi Antarmuka Halaman Setting Email.

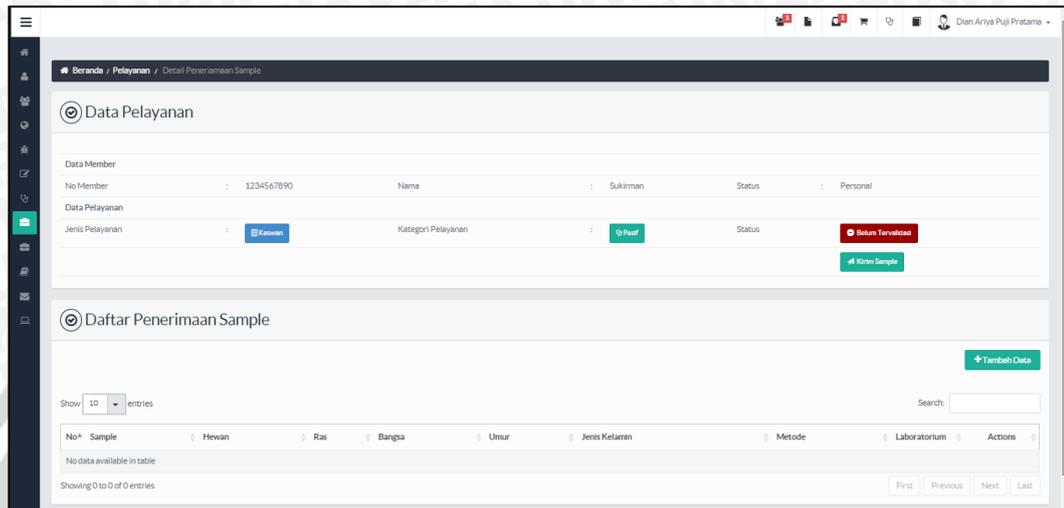
Berikut adalah halaman yang digunakan oleh aktor (Super Admin) untuk melihat dan mengubah alamat email sebagai tempat mengirimkan informasi kepada user tentang pelayanan, lua password user dan registrasi user. Implementasi antarmuka halaman setting email ditunjukkan pada Gambar 5.7.



**Gambar 5.7 Implementasi Antarmuka Halaman Setting Email**

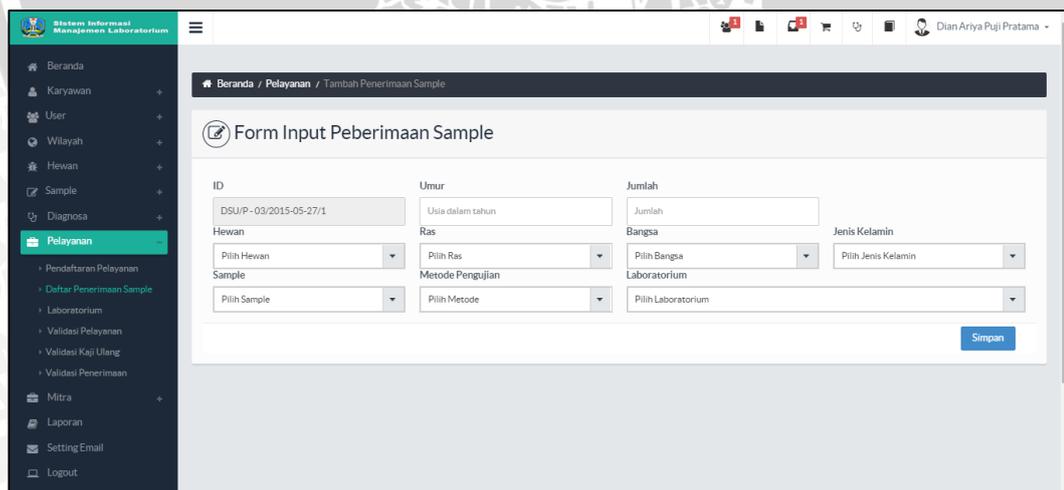
### 5.3.8. Implementasi Antarmuka Halaman Penerimaan Sampel

Berikut adalah halaman yang digunakan oleh aktor (Super admin/ keswan / kesmavet) untuk melihat dan mengelola data penerimaan Sampel. Pada Gambar 5.8 merupakan data penerimaan Sampel ketika data belum di kirim kan.



Gambar 5.8 Implementasi Antarmuka Halaman Kelola Penerimaan Sampel

Pada Gambar 5.9 merupakan form input penerimaan Sampel yang berfungsi untuk menambahkan data Sampel yang akan diujikan beserta metode yang di ajukan sebagai pengujian.



Gambar 5.9 Implementasi Antarmuka Halaman Form Penerimaan Sampel

Data yang telah di inputkan akan dikirim ke labororan berdasarkan pengiriman Sampel dari kesmavet / keswan. Pada Gambar 5.10 merupakan proses setelah melakukan pengiriman Sampel.

The screenshot displays a web application interface for sample management. The top section, titled "Data Pelayanan", shows customer information: No. Member (1234567890), Name (Sukiman), and Status (Personal). Below this, service details are listed: Jenis Pelayanan (Keswan), Kategori Pelayanan (Puaf), and Status (Tervalidasi). The bottom section, "Daftar Penerimaan Sample", features a table with columns for No, Sample, Hewan, Ras, Bangsa, Umur, Jenis Kelamin, Metode, and Laboratorium. A single entry is visible: No. 1, Sample Serum, Hewan Kambing, Ras Petelur, Bangsa Manella, Umur 1, Jenis Kelamin Jantan, Metode Total Plate Count, and Laboratorium Laboratorium Patology. The interface includes navigation buttons like "First", "Previous", "Next", and "Last".

**Gambar 5.10 Implementasi Antarmuka Halaman Kelola Penerimaan Sampel**

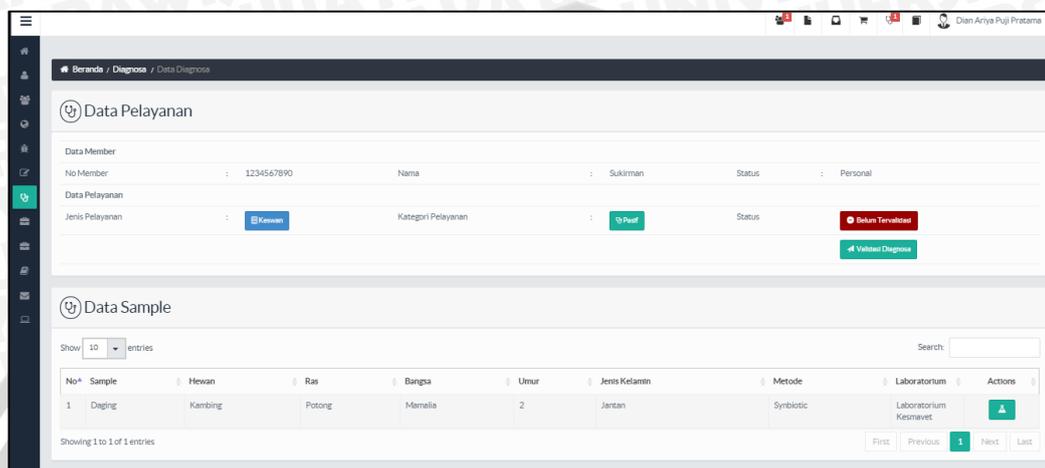
Setelah proses mengirim data Sampel selanjutnya yaitu proses validasi data penerimaan Sampel. Apabila sudah divalidasi maka akan tertulis status "tervalidasi" apabila belum maka status berwarna merah dan tersedia tombol validasi seperti pada Gambar 5.11.

This screenshot shows the same interface as Gambar 5.10, but with a different sample entry. The "Data Pelayanan" section remains the same. In the "Daftar Penerimaan Sample" table, the entry is: No. 1, Sample Daging, Hewan Kambing, Ras Potong, Bangsa Manella, Umur 2, Jenis Kelamin Jantan, Metode Sybiotic, and Laboratorium Laboratorium Kesmavet. The "Status" field in this entry is "Belum Tervalidasi" (Not Validated) in red, and a green "Validasi Penerimaan" button is visible next to it. The rest of the interface, including the navigation buttons, is identical to the previous screenshot.

**Gambar 5.11 Implementasi Antarmuka Halaman Validasi Penerimaan Sampel.**

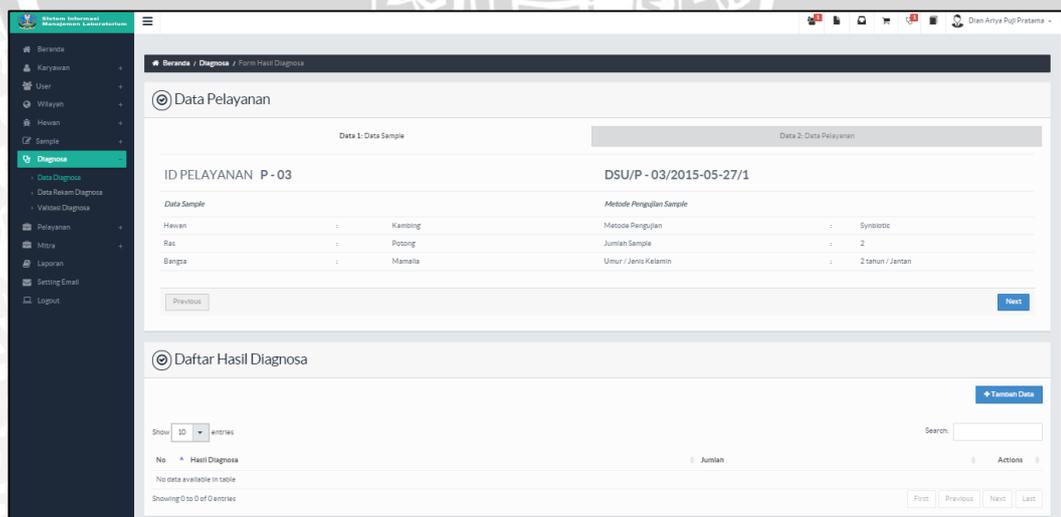
### 5.3.9. Implementasi Antarmuka Halaman Kelola Diagnosa

Berikut adalah halaman yang digunakan oleh aktor (Super admin dan Admin laboran) untuk melihat dan mengelola data diagnosa Sampel. Implementasi antarmuka halaman kelola data diagnosa ditunjukkan pada Gambar 5.12.



**Gambar 5.12 Implementasi Antarmuka Halaman Kelola Data Diagnosa.**

Pada Gambar 5.13 adalah halaman yang digunakan oleh aktor (Super admin dan Admin laboran) untuk mengelola data diagnosa Sampel. Memberikan keterangan hasil dari pengujian Sampel berdasarkan jumlah yang di tetapkan di permintaan pengujian Sampel.



**Gambar 5.13 Implementasi Antarmuka Halaman Form Diagnosa.**

Apabila data telah di masukkan maka akan seperti Pada Gambar 5.13. dimana jumlah permintaan pengujian akan di sesuaikan dengan jumlah yang akan di diagnosa apabila sesuai maka tombol tambah data diagnosa sudah terselesaikan.

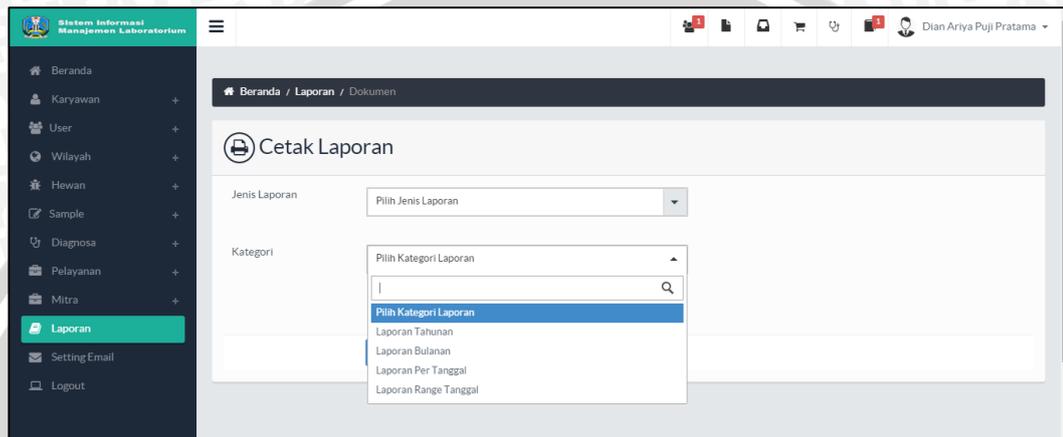
**Gambar 5.13 Implementasi Antarmuka Halaman Detail Diagnosa**

Kemudian apabila sudah di ketahui hasil pengujian maka kepala UPT Laboratorium Kesehatan Hewan Malang akan mengecek hasil pelayanan dan memberikan validasi sebagai tanda proses pelayanan telah selesai. Implementasi antarmuka halaman validasi pelayanan ditunjukkan pada Gambar 5.14.

**Gambar 5.14 Implementasi Antarmuka Halaman Validasi Pelayanan**

### 5.3.10. Implementasi Antarmuka Halaman Cetak Laporan

Berikut adalah halaman yang digunakan oleh aktor (semua aktor kecuali admin kepegawaian) untuk melihat rekap data laporan pelayanan mulai dari kaji ulang permintaan, penerimaan Sampel, hasil diagnosa dan hasil pelayanan. Implementasi antarmuka halaman form cetak laporan ditunjukkan pada Gambar 5.15.



**Gambar 5.15 Implementasi Antarmuka Halaman Form Laporan**

Berikut adalah rekap laporan dari proses pelayanan diantaranya adalah rekap kaji ulang permintaan, rekap penerimaan Sampel, rekap diagnosa dan rekap pelayanan. Laporan yang di cetak berdasarkan kategori periode seperti tahun, bulan dan tahun, tanggal dan range tanggal. Pada Gambar 5.16 merupakan hasil rekap laporan proses pelayanan.

 <b>PEMERINTAH PROVINSI JAWA TIMUR</b> <b>DINAS PETERNAKAN</b> <b>UPT LABORATORIUM KESEHATAN HEWAN</b> JL. RAYA PAKISJAR PO. BOX 03 KEC. PAKIS TELP. (0341) 791182 <b>MALANG 65154</b>								
								Tercetak pada : 27 Mei 2015
Laporan : Rekap Kaji Ulang Permintaan								
Berdasarkan : Tahun 2015								
No	ID Pelayanan	Kuota Sample	Kondisi Sample	Sumber Daya Lab				
				Alat	Bahan	Metode Uji	Standar Uji	Personil
1	P - 01	Memenuhi	Segar	Tersedia	Tersedia	Tersedia	Tersedia	Tersedia
2	P - 02	Memenuhi	Bekul/Dengan Pengawet	Tersedia	Tersedia	Tersedia	Tersedia	Tersedia
3	P - 03	Memenuhi	Segar	Tersedia	Tersedia	Tersedia	Tersedia	Tersedia

**Gambar 5.16 Hasil Rekap Laporan Kaji Ulang Permintaan**

## BAB VI PENGUJIAN DAN ANALISI

### 6.1 Pengujian

#### 6.1.1. Pengujian *Functional Testing*

*Functional Testing* akan dilakukan dengan cara memberikan input pada komponen yang diuji, kemudian diperiksa hasil dari output. Status validasi dikatakan valid apabila output yang dihasilkan sesuai dan status validasi dikatakan tidak valid apabila output yang dihasilkan tidak sesuai atau terdapat error.

#### 6.1.2. Hasil Pengujian *Functional Testing*.

Hasil dari *Functional Testing* dapat dilihat pada tabel-tabel dibawah ini:

##### 1) Hasil Pengujian *Functional Testing* Login

**Tabel 6.1 Tabel *Functional Testing* Login**

<b>Login dengan data benar</b>	
Kondisi awal:	<ul style="list-style-type: none"> <li>• User belum melakukan login</li> <li>• User mengakses halaman awal</li> </ul>
Skenario:	<ol style="list-style-type: none"> <li>1. Mengetikkan username dan password yang sudah terdaftar</li> <li>2. Menekan tombol “Login”</li> </ol>
Hasil yang diharapkan:	Masuk ke dalam halaman utama user
Hasil Pengujian:	Sesuai harapan
Kesimpulan:	Valid
<b>Login dengan data salah</b>	
Kondisi awal:	<ul style="list-style-type: none"> <li>• User belum melakukan login</li> <li>• User mengakses halaman awal</li> </ul>
Skenario:	<ol style="list-style-type: none"> <li>1. Mengetikkan username dan password yang salah atau belum terdaftar</li> <li>2. Menekan tombol “Login”</li> </ol>
Hasil yang diharapkan:	Menampilkan pesan kesalahan
Hasil Pengujian:	Sesuai harapan
Kesimpulan:	Valid

2) Hasil Pengujian *Functional Testing* Kelola KaryawanTabel 6.2 Tabel *Functional* Kelola Karyawan

<b>Tambah Data Karyawan</b>	
Kondisi awal:	<ul style="list-style-type: none"> <li>• Aktor sudah melakukan login</li> <li>• Aktor mengakses halaman Karyawan</li> </ul>
Skenario:	<ol style="list-style-type: none"> <li>1. Klik button “Tambah Data”</li> <li>2. Menginputkan data karyawan pada form input karyawan</li> <li>3. Klik button “Simpan”</li> </ol>
Hasil yang diharapkan:	Data karyawan berhasil ditambahkan
Hasil Pengujian:	Sesuai harapan
Kesimpulan:	Valid
<b>Ubah Data Karyawan</b>	
Kondisi awal:	<ul style="list-style-type: none"> <li>• Aktor sudah melakukan login</li> <li>• Aktor mengakses halaman karyawan</li> <li>• Memilih Data karyawan yang tersedia</li> </ul>
Skenario:	<ol style="list-style-type: none"> <li>1. Klik icon pensil pada salah satu data karyawan</li> <li>2. Mengubah data karyawan pada form edit karyawan</li> <li>3. Kik button “Simpan”</li> </ol>
Hasil yang diharapkan:	Data karyawan berhasil diubah
Hasil Pengujian:	Sesuai harapan
Kesimpulan:	Valid
<b>Hapus Data Karyawan</b>	
Kondisi awal:	<ul style="list-style-type: none"> <li>• Aktor sudah melakukan login</li> <li>• Aktor mengakses halaman karyawan</li> <li>• Memilih Data karyawan yang tersedia</li> </ul>
Skenario:	<ol style="list-style-type: none"> <li>1. Klik icon tempat sampah (trash) pada salah satu data karyawan.</li> <li>2. Muncul pesan “Anda yakin ingin menghapus data tersebut ?”.</li> <li>3. Klik Ok.</li> </ol>
Hasil yang diharapkan:	Data karyawan berhasil terhapus
Hasil Pengujian:	Sesuai harapan
Kesimpulan:	Valid

3) Hasil Pengujian *Functional Testing* Pendaftaran PelayananTabel 6.3 Tabel *Functional* Pendaftaran Pelayanan

<b>Tambah Data Pendaftaran Pelayanan</b>	
Kondisi awal:	<ul style="list-style-type: none"> <li>• Aktor sudah melakukan login</li> <li>• Data member tersedia</li> <li>• Data wilayah tersedia</li> <li>• Aktor mengakses halaman Pendaftaran Pelayanan</li> </ul>
Skenario:	<ol style="list-style-type: none"> <li>1. Klik button “Tambah Data”</li> <li>2. Menginputkan data pendaftaran pelayanan pada form input pendaftaran pelayanan</li> <li>3. Klik button “Simpan”</li> </ol>
Hasil yang diharapkan:	Data pendaftaran pelayanan berhasil ditambahkan
Hasil Pengujian:	Sesuai harapan
Kesimpulan:	Valid
<b>Ubah Data Pendaftaran Pelayanan</b>	
Kondisi awal:	<ul style="list-style-type: none"> <li>• Aktor sudah melakukan login</li> <li>• Data pelayanan yang di ubah hanya data yang dipilih oleh aktor dengan ketentuan dari kepala pelayanan.</li> <li>• Data member tersedia</li> <li>• Data wilayah tersedia</li> <li>• Aktor mengakses halaman Pendaftaran Pelayanan</li> </ul>
Skenario:	<ol style="list-style-type: none"> <li>1. Klik icon pensil pada salah satu data pendaftaran pelayanan.</li> <li>2. Mengubah data pendaftaran pelayanan pada form edit pendaftaran pelayanan.</li> <li>3. Kik button “Simpan”</li> </ol>
Hasil yang diharapkan:	Data pendaftaran pelayanan berhasil diubah
Hasil Pengujian:	Sesuai harapan
Kesimpulan:	Valid
<b>Hapus Data Pendaftaran Pelayanan</b>	
Kondisi awal:	<ul style="list-style-type: none"> <li>• Aktor sudah melakukan login</li> <li>• Aktor mengakses halaman pendaftaran pelayanan.</li> <li>• Memilih Data pendaftaran pelayanan yang tersedia</li> </ul>
Skenario:	<ol style="list-style-type: none"> <li>1. Klik icon tempat sampah (trash) pada salah satu data pendaftaran pelayanan.</li> <li>2. Muncul pesan “Anda yakin ingin menghapus data tersebut ?”.</li> <li>3. Klik Ok.</li> </ol>
Hasil yang diharapkan:	Data pendaftaran pelayanan berhasil terhapus

Hasil Pengujian:	Sesuai harapan
Kesimpulan:	Valid

#### 4) Hasil Pengujian *Functional Testing* Kelola Penerimaan Sampel

Tabel 6.4 Tabel *Functional* Kelola Penerimaan Sampel

<b>Tambah Data Penerimaan Sampel</b>	
Kondisi awal:	<ul style="list-style-type: none"> <li>• Aktor sudah melakukan login</li> <li>• Data pendaftaran pelayanan yang di inputkan</li> <li>• Data sampel, metode, hewan, ras, bangsa dan laboratorium yang tersedia</li> <li>• Aktor mengakses halaman penerimaan sample.</li> </ul>
Skenario:	<ol style="list-style-type: none"> <li>1. Klik notifikasi Penerimaan sampel</li> <li>2. Menginputkan data penerimaan sampel pada form input penerimaan sampel</li> <li>3. Klik button “Simpan”</li> </ol>
Hasil yang diharapkan:	Data penerimaan sampel berhasil ditambahkan
Hasil Pengujian:	Sesuai harapan
Kesimpulan:	Valid
<b>Ubah Data Penerimaan Sample</b>	
Kondisi awal:	<ul style="list-style-type: none"> <li>• Aktor sudah melakukan login</li> <li>• Data penerimaan sampel yang di ubah hanya data yang dipilih oleh aktor dengan ketentuan dari kepala pelayanan.</li> <li>• Data pendaftaran pelayanan yang di inputkan</li> <li>• Data sampel, metode, hewan, ras, bangsa dan laboratorium yang tersedia</li> <li>• Aktor mengakses halaman penerimaan sample.</li> </ul>
Skenario:	<ol style="list-style-type: none"> <li>1. Klik icon pensil pada salah satu data pendaftaran pelayanan.</li> <li>2. Mengubah data pendaftaran pelayanan pada form edit pendaftaran pelayanan.</li> <li>3. Kik button “Simpan”</li> </ol>
Hasil yang diharapkan:	Data pendaftaran pelayanan berhasil diubah
Hasil Pengujian:	Sesuai harapan
Kesimpulan:	Valid
<b>Hapus Data Penerimaan Sampel</b>	
Kondisi awal:	<ul style="list-style-type: none"> <li>• Aktor sudah melakukan login</li> <li>• Aktor mengakses halaman pendaftaran pelayanan.</li> <li>• Memilih Data pendaftaran pelayanan yang tersedia</li> </ul>
Skenario:	<ol style="list-style-type: none"> <li>4. Klik icon tempat sampah (trash) pada salah satu data pendaftaran pelayanan.</li> </ol>

	5. Muncul pesan “Anda yakin ingin menghapus data tersebut ?” 6. Klik Ok.
Hasil yang diharapkan:	Data pendaftaran pelayanan berhasil terhapus
Hasil Pengujian:	Sesuai harapan
Kesimpulan:	Valid

## 5) Hasil Pengujian *Functional Testing* Validasi Pelayanan

**Tabel 6.6** Tabel *Functional* Validasi Pelayanan

Validasi Pelayanan	
Kondisi awal:	<ul style="list-style-type: none"> <li>• Aktor sudah melakukan login</li> <li>• Telah tervalidasi Kaji Uji Permintaan</li> <li>• Telah tervalidasi Penerimaan Sampel</li> <li>• Telah tervalidasi Diagnosa</li> <li>• Aktor mengakses data berdasarkan notifikasi validasi pelayanan</li> </ul>
Skenario:	<ol style="list-style-type: none"> <li>1. Klik notifikasi validasi pelayanan</li> <li>2. Tampil detail pelayanan</li> <li>3. Klik button validasi pelayanan</li> </ol>
Hasil yang diharapkan:	Data pelayanan berhasil tervalidasi
Hasil Pengujian:	Sesuai harapan
Kesimpulan:	Valid

### 6.1.3. Pengujian *User Acceptance Testing* (UAT)

Pengujian ini ditujukan untuk mengetahui prosentase kemudahan pada Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Malang dapat membantu pengguna.

### 6.1.4. Hasil Pengujian *User Acceptance Testing* (UAT)

Pengujian dilakukan dengan memberikan kuesioner kepada pengguna yaitu meliputi staf karyawan UPT Laboratorium Kesehatan Hewan Malang sebanyak 10 orang responden. Tabel Kuisoner Pengujian UAT selengkapnya dapat dilihat pada Lampiran 1 L1-1 sampai L1-10. Pernyataan yang digunakan pada pengujian ini beserta hasil kuisoner ditunjukkan pada tabel 6.12.

Tabel 6.12. Hasil Pengujian *User Acceptance Testing* (UAT)

NO	PERTANYAAN	JUMLAH				
		STM	TM	S	M	SM
<b>1</b>	<b><i>Learnability</i></b>					
1.1	Apakah tampilan menu dalam aplikasi mudah dikenali?	0	0	2	6	2
1.2	Apakah aplikasi mudah dioperasikan?	0	0	3	5	2
1.3	Apakah aplikasi dapat dengan mudah dipelajari?	0	1	1	7	1
1.4	Apakah Informasi yang disediakan oleh aplikasi ini mudah dimengerti?	0	0	0	9	1
<b>2</b>	<b><i>Efficiency</i></b>					
2.1	Apakah dengan diterapkannya aplikasi ini dapat membuat pengelolaan pelayanan lebih efisien?	0	2	2	5	1
2.2	Apakah pencarian data dengan menggunakan aplikasi menjadi lebih mudah?	0	0	0	8	2
2.3	Apakah penggunaan menu atau fitur aplikasi mudah digunakan?	0	0	0	9	1
<b>3</b>	<b><i>Memorability</i></b>					
3.1	Apakah langkah-langkah mengelola pelayanan dengan aplikasi ini mudah diingat?	0	1	2	6	1
3.2	Apakah data tersimpan dengan baik?	0	0	0	10	0
3.3	Apakah penggunaan aplikasi mudah diingat setelah lama tidak menggunakannya?	0	1	3	5	1
<b>4</b>	<b><i>Errors</i></b>					
4.1	Apakah dapat dengan mudah menghindari kesalahan dalam menggunakan aplikasi?	0	0	2	6	1
4.2	Apakah pesan kesalahan atau peringatan dalam penggunaan aplikasi tampil?	0	0	1	8	1
<b>5</b>	<b><i>Satisfaction</i></b>					
5.1	Apakah aplikasi mempunyai fungsi dan kapabilitas (kemampuan) sesuai yang diharapkan?	0	0	3	5	2
5.2	Aplikasi membuat pengelolaan pelayanan menjadi efektif ?	0	0	4	4	2
5.3	Tata letak halaman ( <i>layout</i> ) dan Informasi dalam layar halaman sangat jelas?	0	0	4	5	1
5.4	Apakah aplikasi nyaman digunakan?	0	0	5	4	1
5.5	Secara keseluruhan, apakah penggunaan aplikasi ini memuaskan?	0	0	3	7	0
	Total	0	5	35	109	20

Dari hasil pengujian diatas didapatkan 0 jawaban “Sangat Tidak Mudah”, 5 jawaban “Tidak Mudah”, 35 jawaban “Sedang”, 109 jawaban “Mudah”, dan 20 jawaban “Sangat Mudah”. Dari hasil skor kuesioner tersebut, kemudian dicari nilai

persentase tiap kategori pengujian, yaitu *learnbility*, *efficiency*, *memorability*, *errors*, dan *satisfaction*.

Untuk mencari skor jawaban tiap kategori, dihitung total skor jawaban di kategori tersebut dikali skor *Likert*. Kemudian dicari skor maksimum dengan mengalikan jumlah jawaban tiap pertanyaan dengan jumlah pertanyaan di tiap kategori. Berikut hasil perhitungan skor total jawaban dikali dengan skor *Likert*.

**Tabel 6.13. Perhitungan Total Jawaban dikali Skor *Likert*.**

No	Kategori	Jawaban					Total
		STM (1)	TM (2)	S (3)	M (4)	SM (5)	
1.	<b>Learnability</b>	0 x 1	1 x 2	6 x 3	27 x 4	6 x 5	158
2.	<b>Efficiency</b>	0 x 1	2 x 2	2 x 3	22 x 4	4 x 5	118
3.	<b>Memorability</b>	0 x 1	2 x 2	5 x 3	21 x 4	2 x 5	113
4.	<b>Errors</b>	0 x 1	0 x 2	3 x 3	14 x 4	2 x 5	75
5.	<b>Satisfaction</b>	0 x 1	0 x 2	19 x 3	25 x 4	6 x 5	187
6.	<b>Keseluruhan</b>	0 x 1	5 x 2	35 x 3	109 x 4	20 x 5	651

Sedangkan nilai maksimum didapatkan dari skor tertinggi *Likert* dikali jumlah responden pada tiap kategori. Hasil perhitungan persentase ditunjukkan pada Tabel 6.14.

**Tabel 6.14. Hasil Perhitungan Tiap Kategori Pengujian *User Acceptance Testing* (UAT)**

No	Kategori	Pertanyaan	Total Skor x Skor <i>Likert</i>	Nilai Max	Hasil Skor
1.	<b>Learnability</b>	1.1 – 1.4	158	200	$(158/200) \times 100\% = 79\%$
2.	<b>Efficiency</b>	2.1 – 2.3	118	150	$(118/150) \times 100\% = 78,67\%$
3.	<b>Memorability</b>	3.1 – 3.3	113	150	$(113/150) \times 100\% = 75,33\%$
4.	<b>Errors</b>	4.1 – 4.2	75	100	$(75/100) \times 100\% = 75\%$
5.	<b>Satisfaction</b>	5.1 – 5.5	187	250	$(187/250) \times 100\% = 74,80\%$
6.	<b>Keseluruhan</b>	1.1 – 5.5	651	850	$(651/850) \times 100\% = 76,58\%$

Untuk menilai hasil pengujian terlebih dahulu ditentukan interval skor interpretasi persen yaitu dengan rumus interval  $(I) = 100 / \text{jumlah skor } \textit{Likert}$ . Maka interval sistem ini adalah  $(I) = 100/5 = 20$ , dan interpretasi skor berdasarkan interval adalah :

- Angka 0% - 19.99% = Sangat Tidak Mudah
- Angka 20% - 39.99% = Tidak Mudah
- Angka 40% - 59.99% = Sedang
- Angka 60% - 79.99% = Mudah
- Angka 80% - 100% = Sangat Mudah

Selanjutnya dihitung hasil dengan membagi total skor dengan skor maksimum, kemudian dikali 100%. Hasil =  $(651/850) \times 100\%$  maka didapat hasil akhir 76.58%. Berdasarkan interval, hasil tersebut masuk dalam kategori “mudah” digunakan.

## 6.2 Analisis

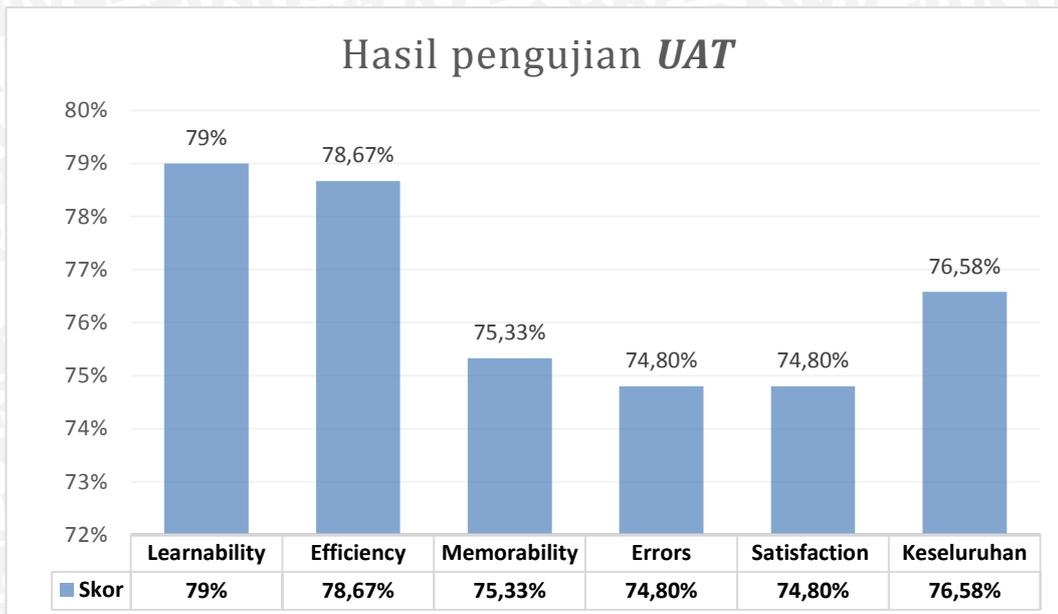
Proses analisis bertujuan untuk mendapatkan kesimpulan dari hasil pengujian Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang. Proses analisis yang dilakukan meliputi analisis pengujian fungsional unit testing dan analisis pengujian *User Acceptance Testing* (UAT).

### 6.2.1. Analisis Hasil *Functional Testing*

Proses analisis terhadap hasil pengujian fungsional unit testing dilakukan dengan melihat kesesuaian antara hasil kinerja sistem dengan daftar kebutuhan. Berdasarkan hasil pengujian dapat disimpulkan bahwa implementasi dan fungsionalitas Sistem Informasi Manajemen Laboratorium (SIMLAB) telah memenuhi kebutuhan yang telah dijabarkan di tahap analisis kebutuhan serta memperoleh hasil 100% valid.

### 6.2.2. Analisis Hasil Pengujian *User Acceptance Testing* (UAT)

Proses analisis terhadap hasil pengujian dilakukan dengan melihat persentase tiap kategori dari pengujian *User Acceptance Testing* (UAT). Dari perhitungan yang dilakukan, didapatkan hasil presentase seperti ditunjukkan pada grafik berikut.



**Gambar 6.3. Grafik Nilai Kemudahan Penggunaan Tiap Kategori Pertanyaan**

Pada hasil perhitungan secara keseluruhan diperoleh hasil sebesar 76.58%. Hasil ini menunjukkan bahwa Sistem Informasi Manajemen Laboratorium di UPT Laboratorium Kesehatan Hewan Malang secara keseluruhan mudah digunakan.

## BAB VII PENUTUP

### 7.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan, maka dapat diambil kesimpulan:

1. Implementasi Sistem Informasi Manajemen Laboratorium telah dibuat sesuai dengan perancangan pada bab 4.
2. Implementasi semua fungsi yang terdapat pada perancangan Sistem Informasi Manajemen Laboratorium telah berjalan dengan baik dan berfungsi 100% valid.
3. Sistem Informasi Manajemen Laboratorium (SIMLAB) memenuhi kemudahan penggunaan. Hal tersebut didasarkan pada hasil pengujian UAT yang menunjukkan bahwa tingkat kemudahan penggunaan 76,58 % yang berarti pengguna sangat setuju terhadap kemudahan penggunaan aplikasi.

### 7.2 Saran

Adapun beberapa saran untuk pengembangan Sistem Informasi Manajemen Laboratorium studi kasus di UPT Laboratorium Kesehatan Hewan Malang, antara lain:

1. Data Sistem Informasi Manajemen Laboratorium diintegrasikan dengan data server di DINAS PETERNAKAN provinsi Jawa Timur.
2. Seiring dengan perubahan kebutuhan, maka diperlukan adanya penyesuaian sistem agar Sistem Informasi Manajemen Laboratorium dapat berjalan secara maksimal.

## DAFTAR PUSTAKA

- [1] Aris.M Garniadi. 2009. Rancang Bangun Perangkat Lunak Sistem Informasi Laboratorium Fasilkom UNSRI. JURNAL ILMIAH GENERIC VOLUME 4, NOMOR 1. Palembang: Fakultas Ilmu Komputer Universitas Sriwijaya.
- [2] Dangar Hardik. 2013. Learning Laravel 4 Application Development. Packt Publishing Ltd. Livery Place 35 Livery Street Birmingham 3 2PB, UK. ISBN 978-1-78328-057-5.
- [3] Dangar Hardik. 2014. Laravel 4 Documentation. <http://www.mediafire.com/view/gdwvlbntt0td2fb/14.pdf>, (di akses pada tanggal 3 Maret 2015).
- [4] Hidayat, Arief. Surarso, Bayu. 2012. Penerapan Arsitektur Model View Controller (Mvc) Dalam Rancang Bangun Sistem Kuis Online Adaptif. Seminar Nasional Teknologi Informasi dan Komunikasi. ISSN 2089-9815. Yogyakarta.
- [5] Jogiyanto. 2005. Analisis dan Desain Pendekatan Terstruktur Teori dan Praktik Aplikasi Bisnis. Penerbit Andi Offset. Yogyakarta.
- [6] Mall, Rajib. 2004. *Fundamentals Of Software Engineering* Second Edition. Department of Computer Science And Engineering. Indian Institute Of Technology Kharagpur. New Delhi.
- [7] Indonesia. Peraturan Pemerintah Republik Indonesia Nomor 47 Tahun 2014. Tentang Pengendalian Dan Penanggulangan Penyakit Hewan.
- [8] Riyanto, EP., Prinali, & Indelarko, Hendi. 2009. Pengembangan Aplikasi Sistem Informasi Geografis. Yogyakarta: Gava Media.
- [9] Pressman, Roger S. 2012. *Rekayasa Perangkat Lunak (Pendekatan Praktisi)* Edisi 7 : Buku 1. Yogyakarta : Andi.
- [10] Rosa, A.S. & Shalahuddin, M. 2013. *Rekayasa Perangkat Lunak*. Bandung: Informatika.
- [11] Sommerville, Ian. 2011. *Software Engineering*, Ninth Edition. New York : Addison-Wesley.
- [12] Simarmata, Janner. 2010. *Rekayasa Perangkat Lunak*. Yogyakarta: Andi.
- [13] Unit Pelaksana Teknis Laboratorium Kesehatan Hewan Malang. 2014. Tupoksi (Online) <http://disnak.jatimprov.go.id/web/uptlab/labkesehatan/hewanmalang>. (diakses tanggal 02 November 2014)

## Lampiran 1

### L1-1 Hasil Kusioner UAT

#### SISTEM INFORMASI MANAJEMEN LABORATORIUM STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG

##### *User Acceptance Testing (UAT)*

**Bapak/Ibu yang terhormat,**

Demi peningkatan kualitas pelayanan di UPT Laboratorium Kesehatan Hewan Malang, kami mohon bantuan Bapak/Ibu untuk memberikan informasi dibawah ini. Semua keterangan dan jawaban yang diperoleh semata-mata hanya untuk kepentingan penelitian dan dijamin kerahasiaannya. Oleh sebab itu jawaban Bapak/Ibu/sdr berikan besar sekali artinya bagi kelancaran penelitian ini.

Isilah pertanyaan berikut ini dengan memberi tanda (√) pada jawaban yang sesuai dengan pendapat bapak/ibu, dengan alternative jawaban yang tersedia. Atas bantuan Bapak/Ibu responden, kami mengucapkan terimakasih.

##### **IDENTITAS RESPONDEN:**

Nama : Dth. Novia Parasiti M.  
Umur : 40  
Jenis Kelamin : Wanita  
Jabatan : Kasek pelayanan Keswan

TM = Tidak Mudah

M = Mudah

NO	PERTANYAAN	JUMLAH				
		STM	TM	S	M	SM
<b>1</b>	<b><i>Learnability</i></b>					
1.1	Apakah tampilan menu dalam aplikasi mudah dikenali?				✓	
1.2	Apakah aplikasi mudah dioperasikan?				✓	
1.3	Apakah aplikasi dapat dengan mudah dipelajari?				✓	
1.4	Apakah informasi yang disediakan oleh aplikasi ini mudah dimengerti?				✓	
<b>2</b>	<b><i>Efficiency</i></b>					
2.1	Apakah dengan diterapkannya aplikasi ini dapat membuat pengelolaan pelayanan lebih efisien?				✓	
2.2	Apakah pencarian dokumen dengan menggunakan aplikasi menjadi lebih mudah?				✓	
2.3	Apakah penggunaan menu atau fitur aplikasi mudah digunakan?				✓	
<b>3</b>	<b><i>Memorability</i></b>					
3.1	Apakah langkah-langkah mengelola pelayanan dengan aplikasi ini mudah diingat?					✓
3.2	Apakah data (informasi dan file) tersimpan dengan baik?				✓	
3.3	Apakah penggunaan aplikasi mudah diingat setelah lama tidak menggunakannya?				✓	
<b>4</b>	<b><i>Errors</i></b>					
4.1	Apakah dapat dengan mudah menghindari kesalahan dalam menggunakan aplikasi?				✓	
4.2	Apakah pesan kesalahan atau peringatan dalam penggunaan aplikasi tampil?				✓	
<b>5</b>	<b><i>Satisfaction</i></b>					
5.1	Apakah aplikasi mempunyai fungsi dan kapabilitas (kemampuan) sesuai yang diharapkan?				✓	
5.2	Aplikasi membuat pengelolaan pelayanan menjadi efektif ?				✓	
5.3	Tata letak halaman (layout) dan informasi dalam layar halaman sangat jelas?				✓	
5.4	Apakah aplikasi nyaman digunakan?				✓	
5.5	Secara keseluruhan, apakah penggunaan aplikasi ini memuaskan?				✓	



## L1-2 Hasil Kusioner UAT

### SISTEM INFORMASI MANAJEMEN LABORATORIUM STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG

#### *User Acceptance Testing (UAT)*

#### **Bapak/Ibu yang terhormat,**

Demi peningkatan kualitas pelayanan di UPT Laboratorium Kesehatan Hewan Malang, kami mohon bantuan Bapak/Ibu untuk memberikan informasi dibawah ini. Semua keterangan dan jawaban yang diperoleh semata-mata hanya untuk kepentingan penelitian dan dijamin kerahasiaannya. Oleh sebab itu jawaban Bapak/Ibu/sdr berikan besar sekali artinya bagi kelancaran penelitian ini.

Isilah pertanyaan berikut ini dengan memberi tanda (√) pada jawaban yang sesuai dengan pendapat bapak/ibu, dengan alternative jawaban yang tersedia. Atas bantuan Bapak/Ibu responden, kami mengucapkan terimakasih.

#### **IDENTITAS RESPONDEN:**

Nama : Winarti Murhika  
Umur : 30  
Jenis Kelamin : Wanita  
Jabatan : Kasubag TU

**Alternatif Jawaban:**

**STM** = Sangat Tidak Mudah

**S** = Sedang

**SM** = Sangat Mudah

**TM** = Tidak Mudah

**M** = Mudah

NO	PERTANYAAN	JUMLAH				
		STM	TM	S	M	SM
<b>1</b>	<b><i>Learnability</i></b>					
1.1	Apakah tampilan menu dalam aplikasi mudah dikenali?				✓	
1.2	Apakah aplikasi mudah dioperasikan?				✓	
1.3	Apakah aplikasi dapat dengan mudah dipelajari?					✓
1.4	Apakah informasi yang disediakan oleh aplikasi ini mudah dimengerti?				✓	
<b>2</b>	<b><i>Efficiency</i></b>					
2.1	Apakah dengan diterapkannya aplikasi ini dapat membuat pengelolaan pelayanan lebih efisien?			✓		
2.2	Apakah pencarian dokumen dengan menggunakan aplikasi menjadi lebih mudah?				✓	
2.3	Apakah penggunaan menu atau fitur aplikasi mudah digunakan?				✓	
<b>3</b>	<b><i>Memorability</i></b>					
3.1	Apakah langkah-langkah mengelola pelayanan dengan aplikasi ini mudah diingat?				✓	
3.2	Apakah data (informasi dan file) tersimpan dengan baik?				✓	
3.3	Apakah penggunaan aplikasi mudah diingat setelah lama tidak menggunakannya?			✓		
<b>4</b>	<b><i>Errors</i></b>					
4.1	Apakah dapat dengan mudah menghindari kesalahan dalam menggunakan aplikasi?			✓		
4.2	Apakah pesan kesalahan atau peringatan dalam penggunaan aplikasi tampil?				✓	
<b>5</b>	<b><i>Satisfaction</i></b>					
5.1	Apakah aplikasi mempunyai fungsi dan kapabilitas (kemampuan) sesuai yang diharapkan?				✓	
5.2	Aplikasi membuat pengelolaan pelayanan menjadi efektif ?			✓		
5.3	Tata letak halaman (layout) dan informasi dalam layar halaman sangat jelas?			✓		
5.4	Apakah aplikasi nyaman digunakan?			✓		
5.5	Secara keseluruhan, apakah penggunaan aplikasi ini memuaskan?				✓	



## L1-3 Hasil Kusioner UAT

### KUESIONER PENELITIAN

#### SISTEM INFORMASI MANAJEMEN LABORATORIUM STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG

##### *User Acceptance Testing (UAT)*

**Bapak/Ibu yang terhormat,**

Demi peningkatan kualitas pelayanan di UPT Laboratorium Kesehatan Hewan Malang, kami mohon bantuan Bapak/Ibu untuk memberikan informasi dibawah ini. Semua keterangan dan jawaban yang diperoleh semata-mata hanya untuk kepentingan penelitian dan dijamin kerahasiaannya. Oleh sebab itu jawaban Bapak/Ibu/sdr berikan besar sekali artinya bagi kelancaran penelitian ini.

Isilah pertanyaan berikut ini dengan memberi tanda (√) pada jawaban yang sesuai dengan pendapat bapak/ibu, dengan alternative jawaban yang tersedia. Atas bantuan Bapak/Ibu responden, kami mengucapkan terimakasih.

##### **IDENTITAS RESPONDEN:**

Nama : *Desi Purnama Sari Amd*  
Umur : *38*  
Jenis Kelamin : *Wanita*  
Jabatan : *Penyelia Kormacet*

NO	PERTANYAAN	JUMLAH				
		STM	TM	S	M	SM
<b>1</b>	<b><i>Learnability</i></b>					
1.1	Apakah tampilan menu dalam aplikasi mudah dikenali?				✓	
1.2	Apakah aplikasi mudah dioperasikan?			✓		
1.3	Apakah aplikasi dapat dengan mudah dipelajari?			✓		
1.4	Apakah informasi yang disediakan oleh aplikasi ini mudah dimengerti?				✓	
<b>2</b>	<b><i>Efficiency</i></b>					
2.1	Apakah dengan diterapkannya aplikasi ini dapat membuat pengelolaan pelayanan lebih efisien?		✓			
2.2	Apakah pencarian dokumen dengan menggunakan aplikasi menjadi lebih mudah?				✓	
2.3	Apakah penggunaan menu atau fitur aplikasi mudah digunakan?				✓	
<b>3</b>	<b><i>Memorability</i></b>					
3.1	Apakah langkah-langkah mengelola pelayanan dengan aplikasi ini mudah diingat?			✓		
3.2	Apakah data (informasi dan file) tersimpan dengan baik?				✓	
3.3	Apakah penggunaan aplikasi mudah diingat setelah lama tidak menggunakannya?			✓		
<b>4</b>	<b><i>Errors</i></b>					
4.1	Apakah dapat dengan mudah menghindari kesalahan dalam menggunakan aplikasi?				✓	
4.2	Apakah pesan kesalahan atau peringatan dalam penggunaan aplikasi tampil?				✓	
<b>5</b>	<b><i>Satisfaction</i></b>					
5.1	Apakah aplikasi mempunyai fungsi dan kapabilitas (kemampuan) sesuai yang diharapkan?			✓		
5.2	Aplikasi membuat pengelolaan pelayanan menjadi efektif?			✓		
5.3	Tata letak halaman (layout) dan informasi dalam layar halaman sangat jelas?			✓		
5.4	Apakah aplikasi nyaman digunakan?			✓		
5.5	Secara keseluruhan, apakah penggunaan aplikasi ini memuaskan?			✓		



## L1-4 Hasil Kusioner UAT

### SISTEM INFORMASI MANAJEMEN LABORATORIUM STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG

#### *User Acceptance Testing (UAT)*

*Bapak/Ibu yang terhormat,*

Demi peningkatan kualitas pelayanan di UPT Laboratorium Kesehatan Hewan Malang, kami mohon bantuan Bapak/Ibu untuk memberikan informasi dibawah ini. Semua keterangan dan jawaban yang diperoleh semata-mata hanya untuk kepentingan penelitian dan dijamin kerahasiaannya. Oleh sebab itu jawaban Bapak/Ibu/sdr berikan besar sekali artinya bagi kelancaran penelitian ini.

Isilah pertanyaan berikut ini dengan memberi tanda (√) pada jawaban yang sesuai dengan pendapat bapak/ibu, dengan alternative jawaban yang tersedia. Atas bantuan Bapak/Ibu responden, kami mengucapkan terimakasih.

#### IDENTITAS RESPONDEN:

Nama : eko Setiawan  
Umur : 32  
Jenis Kelamin : Laki - Laki  
Jabatan : staf TU

NO	PERTANYAAN	JUMLAH				
		STM	TM	S	M	SM
<b>1</b>	<b><i>Learnability</i></b>					
1.1	Apakah tampilan menu dalam aplikasi mudah dikenali?					✓
1.2	Apakah aplikasi mudah dioperasikan?					✓
1.3	Apakah aplikasi dapat dengan mudah dipelajari?				✓	
1.4	Apakah informasi yang disediakan oleh aplikasi ini mudah dimengerti?				✓	
<b>2</b>	<b><i>Efficiency</i></b>					
2.1	Apakah dengan diterapkannya aplikasi ini dapat membuat pengelolaan pelayanan lebih efisien?				✓	
2.2	Apakah pencarian dokumen dengan menggunakan aplikasi menjadi lebih mudah?					✓
2.3	Apakah penggunaan menu atau fitur aplikasi mudah digunakan?				✓	
<b>3</b>	<b><i>Memorability</i></b>					
3.1	Apakah langkah-langkah mengelola pelayanan dengan aplikasi ini mudah diingat?				✓	
3.2	Apakah data (informasi dan file) tersimpan dengan baik?				✓	
3.3	Apakah penggunaan aplikasi mudah diingat setelah lama tidak menggunakannya?				✓	
<b>4</b>	<b><i>Errors</i></b>					
4.1	Apakah dapat dengan mudah menghindari kesalahan dalam menggunakan aplikasi?				✓	
4.2	Apakah pesan kesalahan atau peringatan dalam penggunaan aplikasi tampil?				✓	
<b>5</b>	<b><i>Satisfaction</i></b>					
5.1	Apakah aplikasi mempunyai fungsi dan kapabilitas (kemampuan) sesuai yang diharapkan?					✓
5.2	Aplikasi membuat pengelolaan pelayanan menjadi efektif?					✓
5.3	Tata letak halaman (layout) dan informasi dalam layar halaman sangat jelas?				✓	
5.4	Apakah aplikasi nyaman digunakan?				✓	
5.5	Secara keseluruhan, apakah penggunaan aplikasi ini memuaskan?				✓	

## L1-5 Hasil Kusioner UAT

### SISTEM INFORMASI MANAJEMEN LABORATORIUM STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG

#### *User Acceptance Testing (UAT)*

#### **Bapak/Ibu yang terhormat,**

Demi peningkatan kualitas pelayanan di UPT Laboratorium Kesehatan Hewan Malang, kami mohon bantuan Bapak/Ibu untuk memberikan informasi dibawah ini. Semua keterangan dan jawaban yang diperoleh semata-mata hanya untuk kepentingan penelitian dan dijamin kerahasiaannya. Oleh sebab itu jawaban Bapak/Ibu/sdr berikan besar sekali artinya bagi kelancaran penelitian ini.

Isilah pertanyaan berikut ini dengan memberi tanda (√) pada jawaban yang sesuai dengan pendapat bapak/ibu, dengan alternative jawaban yang tersedia. Atas bantuan Bapak/Ibu responden, kami mengucapkan terimakasih.

#### **IDENTITAS RESPONDEN:**

Nama : Gulis Suyanto, SPet  
Umur : 40  
Jenis Kelamin : Laki - laki  
Jabatan : Laboran

**Alternatif Jawaban:**

**STM** = Sangat Tidak Mudah  
**TM** = Tidak Mudah

**S** = Sedang  
**M** = Mudah

**SM** = Sangat Mudah

NO	PERTANYAAN	JUMLAH				
		STM	TM	S	M	SM
<b>1</b>	<b><i>Learnability</i></b>					
1.1	Apakah tampilan menu dalam aplikasi mudah dikenali?					✓
1.2	Apakah aplikasi mudah dioperasikan?					✓
1.3	Apakah aplikasi dapat dengan mudah dipelajari?				✓	
1.4	Apakah informasi yang disediakan oleh aplikasi ini mudah dimengerti?				✓	
<b>2</b>	<b><i>Efficiency</i></b>					
2.1	Apakah dengan diterapkannya aplikasi ini dapat membuat pengelolaan pelayanan lebih efisien?				✓	
2.2	Apakah pencarian dokumen dengan menggunakan aplikasi menjadi lebih mudah?				✓	
2.3	Apakah penggunaan menu atau fitur aplikasi mudah digunakan?				✓	
<b>3</b>	<b><i>Memorability</i></b>					
3.1	Apakah langkah-langkah mengelola pelayanan dengan aplikasi ini mudah diingat?				✓	
3.2	Apakah data (informasi dan file) tersimpan dengan baik?				✓	
3.3	Apakah penggunaan aplikasi mudah diingat setelah lama tidak menggunakannya?				✓	
<b>4</b>	<b><i>Errors</i></b>					
4.1	Apakah dapat dengan mudah menghindari kesalahan dalam menggunakan aplikasi?				✓	
4.2	Apakah pesan kesalahan atau peringatan dalam penggunaan aplikasi tampil?					✓
<b>5</b>	<b><i>Satisfaction</i></b>					
5.1	Apakah aplikasi mempunyai fungsi dan kapabilitas (kemampuan) sesuai yang diharapkan?					✓
5.2	Aplikasi membuat pengelolaan pelayanan menjadi efektif ?					✓
5.3	Tata letak halaman (layout) dan informasi dalam layar halaman sangat jelas?				✓	
5.4	Apakah aplikasi nyaman digunakan?					✓
5.5	Secara keseluruhan, apakah penggunaan aplikasi ini memuaskan?				✓	



## L1-6 Hasil Kusioner UAT

### KUESIONER PENELITIAN

#### SISTEM INFORMASI MANAJEMEN LABORATORIUM STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG

##### *User Acceptance Testing (UAT)*

**Bapak/Ibu yang terhormat,**

Demi peningkatan kualitas pelayanan di UPT Laboratorium Kesehatan Hewan Malang, kami mohon bantuan Bapak/Ibu untuk memberikan informasi dibawah ini. Semua keterangan dan jawaban yang diperoleh semata-mata hanya untuk kepentingan penelitian dan dijamin kerahasiaannya. Oleh sebab itu jawaban Bapak/Ibu/sdr berikan besar sekali artinya bagi kelancaran penelitian ini.

Isilah pertanyaan berikut ini dengan memberi tanda (√) pada jawaban yang sesuai dengan pendapat bapak/ibu, dengan alternative jawaban yang tersedia. Atas bantuan Bapak/Ibu responden, kami mengucapkan terimakasih.

##### IDENTITAS RESPONDEN:

Nama : *Narhudi*  
Umur : *28*  
Jenis Kelamin : *Laki-Laki*  
Jabatan : *Administratif*

KUISIONER UAT

Alternatif Jawaban:

STM = Sangat Tidak Mudah

S = Sedang

SM = Sangat Mudah

TM = Tidak Mudah

M = Mudah

NO	PERTANYAAN	JUMLAH				
		STM	TM	S	M	SM
<b>1</b>	<b><i>Learnability</i></b>					
1.1	Apakah tampilan menu dalam aplikasi mudah dikenali?				✓	
1.2	Apakah aplikasi mudah dioperasikan?				✓	
1.3	Apakah aplikasi dapat dengan mudah dipelajari?				✓	
1.4	Apakah informasi yang disediakan oleh aplikasi ini mudah dimengerti?				✓	
<b>2</b>	<b><i>Efficiency</i></b>					
2.1	Apakah dengan diterapkannya aplikasi ini dapat membuat pengelolaan pelayanan lebih efisien?				✓	
2.2	Apakah pencarian dokumen dengan menggunakan aplikasi menjadi lebih mudah?				✓	
2.3	Apakah penggunaan menu atau fitur aplikasi mudah digunakan?				✓	
<b>3</b>	<b><i>Memorability</i></b>					
3.1	Apakah langkah-langkah mengelola pelayanan dengan aplikasi ini mudah diingat?				✓	
3.2	Apakah data (informasi dan file) tersimpan dengan baik?				✓	
3.3	Apakah penggunaan aplikasi mudah diingat setelah lama tidak menggunakannya?				✓	
<b>4</b>	<b><i>Errors</i></b>					
4.1	Apakah dapat dengan mudah menghindari kesalahan dalam menggunakan aplikasi?				✓	
4.2	Apakah pesan kesalahan atau peringatan dalam penggunaan aplikasi tampil?				✓	
<b>5</b>	<b><i>Satisfaction</i></b>					
5.1	Apakah aplikasi mempunyai fungsi dan kapabilitas (kemampuan) sesuai yang diharapkan?				✓	
5.2	Aplikasi membuat pengelolaan pelayanan menjadi efektif?				✓	
5.3	Tata letak halaman (layout) dan informasi dalam layar halaman sangat jelas?				✓	
5.4	Apakah aplikasi nyaman digunakan?				✓	
5.5	Secara keseluruhan, apakah penggunaan aplikasi ini memuaskan?				✓	



## L1-7 Hasil Kusioner UAT

### KUESIONER PENELITIAN

#### SISTEM INFORMASI MANAJEMEN LABORATORIUM STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG

##### *User Acceptance Testing (UAT)*

*Bapak/Ibu yang terhormat,*

Demi peningkatan kualitas pelayanan di UPT Laboratorium Kesehatan Hewan Malang, kami mohon bantuan Bapak/Ibu untuk memberikan informasi dibawah ini. Semua keterangan dan jawaban yang diperoleh semata-mata hanya untuk kepentingan penelitian dan dijamin kerahasiaannya. Oleh sebab itu jawaban Bapak/Ibu/sdr berikan besar sekali artinya bagi kelancaran penelitian ini.

Isilah pertanyaan berikut ini dengan memberi tanda (✓) pada jawaban yang sesuai dengan pendapat bapak/ibu, dengan alternative jawaban yang tersedia. Atas bantuan Bapak/Ibu responden, kami mengucapkan terimakasih.

##### **IDENTITAS RESPONDEN:**

Nama : Sri Agustini  
Umur : 28  
Jenis Kelamin : Wanita  
Jabatan : Administratif



**KUISIONER UAT**

**Alternatif Jawaban:**

**STM** = Sangat Tidak Mudah

**S** = Sedang

**SM** = Sangat Mudah

**TM** = Tidak Mudah

**M** = Mudah

NO	PERTANYAAN	JUMLAH				
		STM	TM	S	M	SM
<b>1</b>	<b><i>Learnability</i></b>					
1.1	Apakah tampilan menu dalam aplikasi mudah dikenali?				✓	
1.2	Apakah aplikasi mudah dioperasikan?				✓	
1.3	Apakah aplikasi dapat dengan mudah dipelajari?				✓	
1.4	Apakah informasi yang disediakan oleh aplikasi ini mudah dimengerti?					✓
<b>2</b>	<b><i>Efficiency</i></b>					
2.1	Apakah dengan diterapkannya aplikasi ini dapat membuat pengelolaan pelayanan lebih efisien?				✓	
2.2	Apakah pencarian dokumen dengan menggunakan aplikasi menjadi lebih mudah?					✓
2.3	Apakah penggunaan menu atau fitur aplikasi mudah digunakan?					✓
<b>3</b>	<b><i>Memorability</i></b>					
3.1	Apakah langkah-langkah mengelola pelayanan dengan aplikasi ini mudah diingat?				✓	
3.2	Apakah data (informasi dan file) tersimpan dengan baik?				✓	
3.3	Apakah penggunaan aplikasi mudah diingat setelah lama tidak menggunakannya?				✓	
<b>4</b>	<b><i>Errors</i></b>					
4.1	Apakah dapat dengan mudah menghindari kesalahan dalam menggunakan aplikasi?				✓	
4.2	Apakah pesan kesalahan atau peringatan dalam penggunaan aplikasi tampil?				✓	
<b>5</b>	<b><i>Satisfaction</i></b>					
5.1	Apakah aplikasi mempunyai fungsi dan kapabilitas (kemampuan) sesuai yang diharapkan?				✓	
5.2	Aplikasi membuat pengelolaan pelayanan menjadi efektif?				✓	
5.3	Tata letak halaman (layout) dan informasi dalam layar halaman sangat jelas?					✓
5.4	Apakah aplikasi nyaman digunakan?				✓	
5.5	Secara keseluruhan, apakah penggunaan aplikasi ini memuaskan?				✓	



## L1-8 Hasil Kusioner UAT

### KUESIONER PENELITIAN

#### SISTEM INFORMASI MANAJEMEN LABORATORIUM STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG

##### *User Acceptance Testing (UAT)*

###### ***Bapak/Ibu yang terhormat,***

Demi peningkatan kualitas pelayanan di UPT Laboratorium Kesehatan Hewan Malang, kami mohon bantuan Bapak/Ibu untuk memberikan informasi dibawah ini. Semua keterangan dan jawaban yang diperoleh semata-mata hanya untuk kepentingan penelitian dan dijamin kerahasiaannya. Oleh sebab itu jawaban Bapak/Ibu/sdr berikan besar sekali artinya bagi kelancaran penelitian ini.

Isilah pertanyaan berikut ini dengan memberi tanda (√) pada jawaban yang sesuai dengan pendapat bapak/ibu, dengan alternative jawaban yang tersedia. Atas bantuan Bapak/Ibu responden, kami mengucapkan terimakasih.

###### **IDENTITAS RESPONDEN:**

Nama : *Dh. Sni Pangastuti*  
Umur : *40*  
Jenis Kelamin : *Wanita*  
Jabatan : *Kasir Pelayanan Kesehatan*



KUISIONER UAT

Alternatif Jawaban:

STM = Sangat Tidak Mudah

S = Sedang

SM = Sangat Mudah

TM = Tidak Mudah

M = Mudah

NO	PERTANYAAN	JUMLAH				
		STM	TM	S	M	SM
<b>1</b>	<b><i>Learnability</i></b>					
1.1	Apakah tampilan menu dalam aplikasi mudah dikenali?				✓	
1.2	Apakah aplikasi mudah dioperasikan?				✓	
1.3	Apakah aplikasi dapat dengan mudah dipelajari?				✓	
1.4	Apakah informasi yang disediakan oleh aplikasi ini mudah dimengerti?				✓	
<b>2</b>	<b><i>Efficiency</i></b>					
2.1	Apakah dengan diterapkannya aplikasi ini dapat membuat pengelolaan pelayanan lebih efisien?				✓	
2.2	Apakah pencarian dokumen dengan menggunakan aplikasi menjadi lebih mudah?				✓	
2.3	Apakah penggunaan menu atau fitur aplikasi mudah digunakan?				✓	
<b>3</b>	<b><i>Memorability</i></b>					
3.1	Apakah langkah-langkah mengelola pelayanan dengan aplikasi ini mudah diingat?				✓	
3.2	Apakah data (informasi dan file) tersimpan dengan baik?				✓	
3.3	Apakah penggunaan aplikasi mudah diingat setelah lama tidak menggunakannya?				✓	
<b>4</b>	<b><i>Errors</i></b>					
4.1	Apakah dapat dengan mudah menghindari kesalahan dalam menggunakan aplikasi?				✓	
4.2	Apakah pesan kesalahan atau peringatan dalam penggunaan aplikasi tampil?				✓	
<b>5</b>	<b><i>Satisfaction</i></b>					
5.1	Apakah aplikasi mempunyai fungsi dan kapabilitas (kemampuan) sesuai yang diharapkan?				✓	
5.2	Aplikasi membuat pengelolaan pelayanan menjadi efektif?				✓	
5.3	Tata letak halaman (layout) dan informasi dalam layar halaman sangat jelas?				✓	
5.4	Apakah aplikasi nyaman digunakan?				✓	
5.5	Secara keseluruhan, apakah penggunaan aplikasi ini memuaskan?				✓	



## L1-9 Hasil Kusioner UAT

### KUESIONER PENELITIAN

#### SISTEM INFORMASI MANAJEMEN LABORATORIUM STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG

##### *User Acceptance Testing (UAT)*

*Bapak/Ibu yang terhormat,*

Demi peningkatan kualitas pelayanan di UPT Laboratorium Kesehatan Hewan Malang, kami mohon bantuan Bapak/Ibu untuk memberikan informasi dibawah ini. Semua keterangan dan jawaban yang diperoleh semata-mata hanya untuk kepentingan penelitian dan dijamin kerahasiaannya. Oleh sebab itu jawaban Bapak/Ibu/sdr berikan besar sekali artinya bagi kelancaran penelitian ini.

Isilah pertanyaan berikut ini dengan memberi tanda (√) pada jawaban yang sesuai dengan pendapat bapak/ibu, dengan alternative jawaban yang tersedia. Atas bantuan Bapak/Ibu responden, kami mengucapkan terimakasih.

##### **IDENTITAS RESPONDEN:**

Nama : Vinda Mustika  
Umur : 27  
Jenis Kelamin : Wanita  
Jabatan : Administratif

## KUISIONER UAT

## Alternatif Jawaban:

STM = Sangat Tidak Mudah

S = Sedang

SM = Sangat Mudah

TM = Tidak Mudah

M = Mudah

NO	PERTANYAAN	JUMLAH				
		STM	TM	S	M	SM
<b>1</b>	<b><i>Learnability</i></b>					
1.1	Apakah tampilan menu dalam aplikasi mudah dikenali?			✓		
1.2	Apakah aplikasi mudah dioperasikan?			✓		
1.3	Apakah aplikasi dapat dengan mudah dipelajari?		✓			
1.4	Apakah informasi yang disediakan oleh aplikasi ini mudah dimengerti?				✓	
<b>2</b>	<b><i>Efficiency</i></b>					
2.1	Apakah dengan diterapkannya aplikasi ini dapat membuat pengelolaan pelayanan lebih efisien?		✓			
2.2	Apakah pencarian dokumen dengan menggunakan aplikasi menjadi lebih mudah?				✓	
2.3	Apakah penggunaan menu atau fitur aplikasi mudah digunakan?				✓	
<b>3</b>	<b><i>Memorability</i></b>					
3.1	Apakah langkah-langkah mengelola pelayanan dengan aplikasi ini mudah diingat?		✓			
3.2	Apakah data (informasi dan file) tersimpan dengan baik?				✓	
3.3	Apakah penggunaan aplikasi mudah diingat setelah lama tidak menggunakannya?		✓			
<b>4</b>	<b><i>Errors</i></b>					
4.1	Apakah dapat dengan mudah menghindari kesalahan dalam menggunakan aplikasi?			✓		
4.2	Apakah pesan kesalahan atau peringatan dalam penggunaan aplikasi tampil?			✓		
<b>5</b>	<b><i>Satisfaction</i></b>					
5.1	Apakah aplikasi mempunyai fungsi dan kapabilitas (kemampuan) sesuai yang diharapkan?			✓		
5.2	Aplikasi membuat pengelolaan pelayanan menjadi efektif?			✓		
5.3	Tata letak halaman (layout) dan informasi dalam layar halaman sangat jelas?			✓		
5.4	Apakah aplikasi nyaman digunakan?			✓		
5.5	Secara keseluruhan, apakah penggunaan aplikasi ini memuaskan?			✓		

## L1-10 Hasil Kusioner UAT

### KUESIONER PENELITIAN

#### SISTEM INFORMASI MANAJEMEN LABORATORIUM STUDI KASUS UPT LABORATORIUM KESEHATAN HEWAN MALANG

##### *User Acceptance Testing (UAT)*

*Bapak/Ibu yang terhormat,*

Demi peningkatan kualitas pelayanan di UPT Laboratorium Kesehatan Hewan Malang, kami mohon bantuan Bapak/Ibu untuk memberikan informasi dibawah ini. Semua keterangan dan jawaban yang diperoleh semata-mata hanya untuk kepentingan penelitian dan dijamin kerahasiannya. Oleh sebab itu jawaban Bapak/Ibu/sdr berikan besar sekali artinya bagi kelancaran penelitian ini.

Isilah pertanyaan berikut ini dengan memberi tanda (√) pada jawaban yang sesuai dengan pendapat bapak/ibu, dengan alternative jawaban yang tersedia. Atas bantuan Bapak/Ibu responden, kami mengucapkan terimakasih.

##### IDENTITAS RESPONDEN:

Nama : Drh. Anjarini DP.  
Umur : 32  
Jenis Kelamin : Wanita  
Jabatan : Penyeka Keswan

## KUISIONER UAT

## Alternatif Jawaban:

STM = Sangat Tidak Mudah  
TM = Tidak Mudah

S = Sedang  
M = Mudah

SM = Sangat Mudah

NO	PERTANYAAN	JUMLAH				
		STM	TM	S	M	SM
<b>1</b>	<b><i>Learnability</i></b>					
1.1	Apakah tampilan menu dalam aplikasi mudah dikenali?			✓		
1.2	Apakah aplikasi mudah dioperasikan?			✓		
1.3	Apakah aplikasi dapat dengan mudah dipelajari?				✓	
1.4	Apakah informasi yang disediakan oleh aplikasi ini mudah dimengerti?				✓	
<b>2</b>	<b><i>Efficiency</i></b>					
2.1	Apakah dengan diterapkannya aplikasi ini dapat membuat pengelolaan pelayanan lebih efisien?			✓		
2.2	Apakah pencarian dokumen dengan menggunakan aplikasi menjadi lebih mudah?				✓	
2.3	Apakah penggunaan menu atau fitur aplikasi mudah digunakan?				✓	
<b>3</b>	<b><i>Memorability</i></b>					
3.1	Apakah langkah-langkah mengelola pelayanan dengan aplikasi ini mudah diingat?			✓		
3.2	Apakah data (informasi dan file) tersimpan dengan baik?				✓	
3.3	Apakah penggunaan aplikasi mudah diingat setelah lama tidak menggunakannya?			✓		
<b>4</b>	<b><i>Errors</i></b>					
4.1	Apakah dapat dengan mudah menghindari kesalahan dalam menggunakan aplikasi?				✓	
4.2	Apakah pesan kesalahan atau peringatan dalam penggunaan aplikasi tampil?				✓	
<b>5</b>	<b><i>Satisfaction</i></b>					
5.1	Apakah aplikasi mempunyai fungsi dan kapabilitas (kemampuan) sesuai yang diharapkan?			✓		
5.2	Aplikasi membuat pengelolaan pelayanan menjadi efektif?			✓		
5.3	Tata letak halaman (layout) dan informasi dalam layar halaman sangat jelas?			✓		
5.4	Apakah aplikasi nyaman digunakan?			✓		
5.5	Secara keseluruhan, apakah penggunaan aplikasi ini memuaskan?			✓		