



**PERANCANGAN DAN PEMBUATAN  
APLIKASI AKUNTANSI UD. COBRA SAKTI BERBASIS WEB**

**SKRIPSI**

oleh

**Rima Novida Ayulian**

**NIM 092410101043**

**PROGRAM STUDI SISTEM INFORMASI**

**UNIVERSITAS JEMBER**

**2015**



**PERANCANGAN DAN PEMBUATAN  
APLIKASI AKUNTANSI UD COBRA SAKTI BERBASIS WEB**

**SKRIPSI**

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat

untuk menyelesaikan Program Studi Sistem Informasi (S1)

dan mencapai gelar Sarjana Komputer

oleh

**Rima Novida Ayulian**

**NIM 092410101043**

**PROGRAM STUDI SISTEM INFORMASI**

**UNIVERSITAS JEMBER**

**2015**

## PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

1. Allah SWT yang selalu memberikan rencana indah dan terbaik bagiku;
2. Muhammad Saw yang selalu menjadi panutan bagiku;
3. Ibunda Siti Mufidanah dan Ayahanda Sugiyono tercinta yang sudah memberikan doa dan dukungan serta semangat dalam menyelesaikan skripsi ini;
4. Adik-adikku Riri, Asya dan syahdan tersayang yang sudah menjadi semangat dalam hidup dalam menyelesaikan skripsi ini;
5. Guru dan Dosenku sejak taman kanak-kanak sampai dengan perguruan tinggi;
6. Seluruh teman – teman seperjuangan angkatan 2009 Program Studi Sistem Informasi Universitas Jember;
7. Almater Program Studi Sistem Informasi Universitas Jember.

## PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Rima Novida Ayulian

NIM : 092410101043

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Perancangan dan Pembuatan Aplikasi Akuntansi UD Cobra Sakti Berbasis Web”, adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya, belum pernah diajukan pada institusi mana pun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika di kemudian hari pernyataan ini tidak benar.

Jember, 25 Juni 2015

Yang menyatakan,

Rima Novida Ayulian

NIM 092410101043

**SKRIPSI**

**PERANCANGAN DAN PEMBUATAN  
APLIKASI AKUNTANSI UD COBRA SAKTI BERBASIS WEB**

oleh

Rima Novida Ayulian

NIM 092410101043

Pembimbing

Dosen Pembimbing Utama : Anang Andrianto S.T., M.T.

Dosen Pembimbing Anggota : Nelly Oktavia Adiwijaya S.Si., MT.

**PENGESAHAN**

Skripsi berjudul **Perancangan dan Pembuatan Aplikasi Akuntansi UD Cobra Sakti Berbasis Web** telah diuji dan disahkan pada :

Hari, tanggal : Kamis, 25 Juni 2015

Tempat : Program Studi Sistem Informasi, Universitas Jember

Tim Penguji:

Penguji I,

Penguji II,

Prof. Drs. Slamin, M.Comp.Sc.,Ph.D.  
NIP. 196704201992011001

M. Arief Hidayat, S.Kom, M.Kom.  
NIP. 198101232010121003

Mengesahkan

Ketua Program Studi,

Prof. Drs. Slamin, M.Comp.Sc.,Ph.D.  
NIP. 196704201992011001

## PENGESAHAN PEMBIMBING

Skripsi berjudul **Perancangan dan Pembuatan Aplikasi Akuntansi UD Cobra Sakti Berbasis Web** telah diuji dan disahkan pada:

Hari, tanggal : Kamis, 25 Juni 2015

Tempat : Program Studi Sistem Informasi, Universitas Jember

Tim Pembimbing:

Pembimbing I,

Pembimbing II,

Anang Andrianto, ST.,MT.

Nelly Oktavia A, S.si, MT.

NIP. 196906151997021002

NIP. 198410242009122008

## RINGKASAN

**Perancangan dan Pembuatan Aplikasi Akuntansi UD Cobra Sakti Berbasis Web;** Rima Novida Ayulian, 092410101043 2015, 143 HALAMAN; Program Studi Sistem Informasi Universitas Jember.

Perkembangan teknologi yang semakin pesat menjadikan kita perlunya sebuah aplikasi untuk menangani masalah keuangan perusahaan sehingga laporan keuangan tersebut dapat membantu sebuah pengambilan keputusan perusahaan untuk mencapai tujuan dan untuk menyediakan catatan lengkap mengenai pertanggungjawaban dan perlindungan kekayaan perusahaan.

UD Cobra Sakti merupakan salah satu usaha yang memiliki kendala pada masalah penentuan keuangan, contohnya seperti masih belum bisa menentukan berapa jumlah laba yang didapatkan dan berapa jumlah rugi yang diperoleh jika ada. Hal tersebut mengakibatkan pihak perusahaan mengalami kesulitan dalam pengambilan keputusan dan mengalami kesulitan dalam memberikan catatan laporan labarugi yang diperoleh, maka dari itu dibutuhkan suatu media informasi dalam bentuk aplikasi komputer.

Aplikasi akuntansi berbasis web memiliki beberapa fitur yang bisa dipakai untuk menyajikan laporan labarugi yang diinginkan. Fitur-fitur fungsional dan non fungsional yang dibuat disesuaikan dengan data-data yang telah berhasil dikumpulkan. Metode pengembangan sistem *waterfall* digunakan pada aplikasi akuntansi berbasis web. Desain dan penulisan kode program PHP menggunakan konsep OOP, dan manajemen basis data menggunakan MySQL.



## PRAKATA

Alhamdulillah, Puji syukur ke hadirat Allah SWT atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul “Perancangan dan Pembuatan Aplikasi Akuntansi UD Cobra Sakti Berbasis Web”. Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan Strata Satu (S1) pada Program Studi Sistem Informasi Universitas Jember.

Penyusunan skripsi ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Prof. Drs. Slamir, M.CompSc., Ph.D., selaku Ketua Program Studi Sistem Informasi Universitas Jember;
2. Anang Andrianto, S.T., M.T., selaku Dosen Pembimbing Utama dan Nelly Oktavia Adiwijaya, S.T., M.T., selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian, bimbingan dan arahan dalam penulisan skripsi ini;
3. Seluruh Bapak dan Ibu dosen beserta staf karyawan di Program Studi Sistem Informasi Universitas Jember;
4. Teman-teman seperjuangan angkatan 2009 Program Studi Sistem Informasi;
5. Almamater tercinta Program Studi Sistem Informasi Universitas Jember;
6. Semua pihak yang tidak dapat disebutkan satu-persatu.

Penulis juga menerima segala kritik dan saran dari semua pihak demi kesempurnaan skripsi ini. Akhirnya penulis berharap, semoga skripsi ini dapat bermanfaat.

Jember, 26 Juni 2015

Penulis

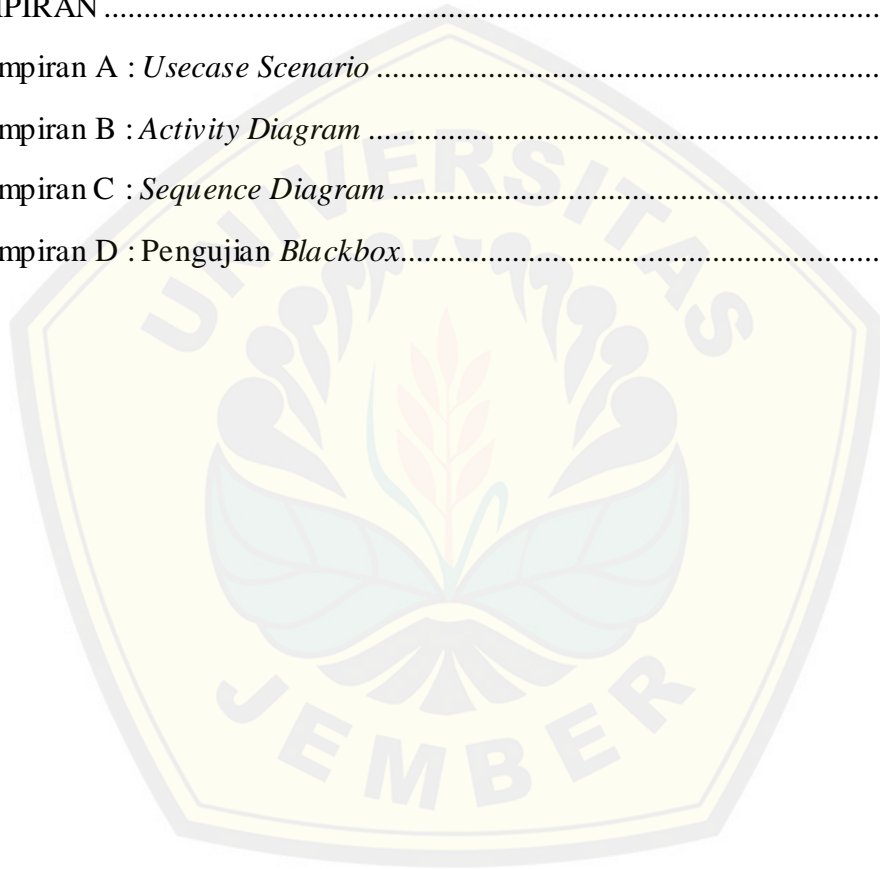
**DAFTAR ISI**

SAMPUL.....	i
JUDUL .....	ii
PERSEMBAHAN .....	iii
PERNYATAAN .....	iv
PEMBIMBING .....	v
PENGESAHAN .....	vi
PENGESAHAN PEMBIMBING.....	vii
RINGKASAN .....	viii
PRAKATA.....	ix
DAFTAR ISI.....	x
DAFTAR TABEL.....	xiv
DAFTAR GAMBAR .....	xv
BAB 1. PENDAHULUAN .....	1
1.1 Latar Belakang Masalah.....	1
1.2 Perumusan Masalah.....	3
1.3 Tujuan dan Manfaat.....	3
1.3.1 Tujuan.....	3
1.3.2 Manfaat.....	3
1.4 Batasan Masalah.....	4
1.5 Sistematika Penulisan.....	4
BAB 2. TINJAUAN PUSTAKA .....	6
2.1 Aplikasi Akuntansi.....	6
2.2 Aplikasi Berbasis Web .....	7
2.3 Laba Rugi .....	8

2.4	Model SDLC Wterfall .....	9
2.5	Proses Bisnis.....	9
BAB 3. METODOLOGI PENELITIAN .....		13
3.1	Objek Penelitian .....	13
3.2	Metode Pengembangan Aplikasi .....	13
3.2.1	Definisi Persyaratan .....	13
3.2.2	Perancangan Sistem dan Perangkat Lunak .....	14
3.2.3	Implementasi dan Pengujian Unit .....	14
3.2.4	Integrasi dan Pengujian Sistem .....	14
3.2.5	Operasi dan Pemeliharaan.....	15
3.3	Perangkat Keras .....	15
3.4	Perangkat Lunak .....	15
BAB 4. ANALISIS DAN PERANCANGAN SISTEM .....		16
4.1	Analisis Sistem.....	16
4.1.1	Kebutuhan Fungsional.....	16
4.1.2	Kebutuhan Non Fungsional.....	17
4.2	Desain Sistem.....	18
4.2.1	<i>Bussiness Process</i> .....	19
4.2.2	<i>Usecase Diagram</i> .....	20
4.2.3	<i>Usecase Scenario</i> .....	25
4.2.4	<i>Activity Diagram</i> .....	32
4.2.5	<i>Sequence Diagram</i> .....	40
4.2.6	<i>Class Diagram</i> .....	44
4.2.7	<i>Entity Relational Diagram (ERD)</i> .....	46
BAB 5. Hasil dan Pembahasan .....		47
5.1	Pengkodean Sistem.....	47

5.1.1	Menambah Data Pembelian .....	47
5.1.2	Menambah Data Penjualan.....	52
5.2	Pengujian Sitem.....	55
5.2.1	Pengujian <i>Black box</i> .....	55
5.2.2	Pengujian <i>White box</i> .....	56
5.3	Hasil Implementasi Aplikasi .....	59
5.3.1	Fitur <i>Login</i> .....	59
5.3.2	Fitur Tambah Data User .....	60
5.3.3	Fitur <i>Edit Data User</i> .....	62
5.3.4	Fitur Hapus Data User .....	63
5.3.5	Fitur <i>Add Bahan Baku</i> .....	64
5.3.6	Fitur <i>Edit Bahan Baku</i> .....	66
5.3.7	Fitur Hapus Bahan Baku .....	67
5.3.8	Fitur <i>Add Produk</i> .....	68
5.3.9	Fitur <i>Edit Produk</i> .....	70
5.3.10	Fitur Hapus Produk .....	71
5.3.11	Fitur <i>Add Tansaksi Pembelian</i> .....	72
5.3.12	Fitur <i>Edit Tansaksi Pembelian</i> .....	75
5.3.13	Fitur Hapus Transaksi Pembelian .....	75
5.3.14	Fitur <i>Add Tansaksi Penjualan</i> .....	76
5.3.15	Fitur <i>Edit Tansaksi Penjualan</i> .....	78
5.3.16	Fitur Hapus Transaksi Penjualan.....	79
5.3.17	Fitur <i>Add Tansaksi Pemasukan lain</i> .....	79
5.3.18	Fitur <i>Add Tansaksi Pengeluaran lain</i> .....	80
5.3.19	Fitur <i>view Laporan Pemasukan</i> .....	81
5.3.20	Fitur <i>View Laporan Pengeluaran</i> .....	82

5.3.21	Fitur <i>View</i> Laporan LabaRugi.....	83
BAB 6.	Penutup.....	85
6.1	Kesimpulan.....	85
6.2	Saran.....	85
DAFTAR PUSTAKA	.....	86
LAMPIRAN	.....	87
Lampiran A	: <i>Usecase Scenario</i> .....	87
Lampiran B	: <i>Activity Diagram</i> .....	103
Lampiran C	: <i>Sequence Diagram</i> .....	113
Lampiran D	: Pengujian <i>Blackbox</i> .....	121



## DAFTAR TABEL

Tabel 4.1 Kebutuhan Fungsional Sistem.....	16
Tabel 4.2 Kebutuhan Non Fungsional Sistem.....	18
Tabel 4.3 Deskripsi <i>Usecase Diagram</i> .....	22
Tabel 4.4 Hak Akses Pengguna.....	24
Tabel 4.5 Usecase Scenario "Menambah Data Pembelian" .....	25
Tabel 4.6 Usecase Scenario "Manajemen Data Pembelian" .....	27
Tabel 4.7 <i>Usecase Scenario</i> "Menampilkan Laporan Pemasukan" .....	29
Tabel 4.8 <i>Usecase Scenario</i> "Menampilkan Laporan Pengeluaran".....	31
Tabel 4.9 <i>Activity Diagram</i> "Menambah Data Pembelian" .....	33
Tabel 4.10 <i>Activity Diagram</i> "Menambah Data Penjualan" .....	35
Tabel 4.11 <i>Activity Diagram</i> "Menampilkan Laporan Pemasukan".....	37
Tabel 4.12 <i>Activity Diagram</i> "Menampilkan Laporan Pengeluaran" .....	39
Tabel 5.1 Test Case Fitur Login.....	59

## DAFTAR GAMBAR

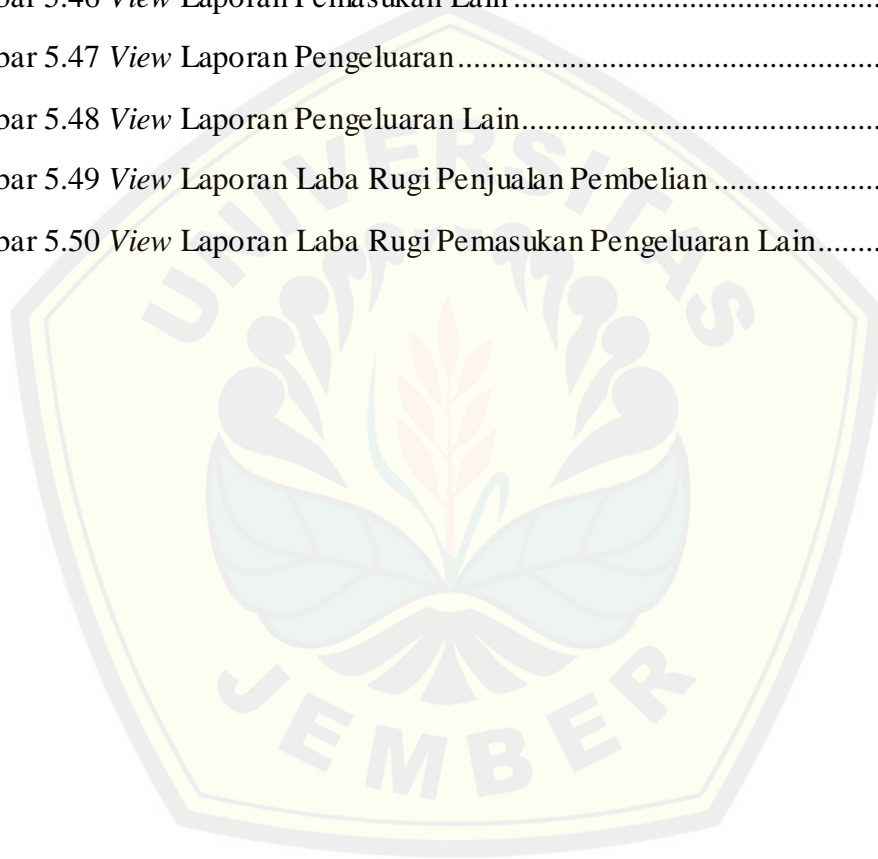
Gambar 2.1 Siklus Hidup Perangkat Lunak .....	9
Gambar 4.1 <i>Business Process</i> .....	20
Gambar 4.2 <i>Use Case Diagram</i> .....	21
Gambar 4.3 <i>Sequence Diagram</i> Menambah Data Pembelian .....	41
Gambar 4.4 <i>Sequence Diagram</i> Menambah Data Pembelian .....	41
Gambar 4.5 <i>Sequence Diagram</i> Menambah Data Penjualan .....	42
Gambar 4.6 <i>Sequence Diagram</i> Menampilkan Laporan Pemasukan .....	43
Gambar 4.7 <i>Sequence Diagram</i> Menampilkan Laporan Pengeluaran .....	44
Gambar 4.8 <i>Class Diagram</i> Aplikasi Akuntansi .....	45
Gambar 4.9 <i>Entity Relational Diagram</i> Aplikasi Akuntansi .....	46
Gambar 5.1 Kode <i>Controller</i> Menambah Data Pembelian .....	47
Gambar 5.2 Kode Model Menambah Data Pembelian .....	48
Gambar 5.3 Kode <i>Controller</i> Menambah Data Pembelian .....	48
Gambar 5.4 Kode Model Menambah Data Pembelian .....	49
Gambar 5.5 Kode Model Menambah Data Pembelian .....	49
Gambar 5.6 Kode Model Menambah Data Pembelian .....	51
Gambar 5.7 Kode <i>Controller</i> Menambah Data Penjualan .....	52
Gambar 5.8 Kode Model Menambah Data Penjualan .....	52
Gambar 5.9 Kode <i>Controller</i> Menambah Data Penjualan .....	53
Gambar 5.10 Kode Model Menambah Data Penjualan .....	53
Gambar 5.11 Kode Model Menambah Data Penjualan .....	54
Gambar 5.12 Kode Model Menambah Data Penjualan .....	55
Gambar 5.13 Listing Program Fitur <i>Login</i> .....	56
Gambar 5.14 Diagram Alir Fitur <i>Login</i> .....	57



Gambar 5.15 Halaman <i>Login</i> .....	60
Gambar 5.16 Fitur Tambah Data <i>User</i> .....	61
Gambar 5.17 Peringatan Kolom Kosong Tambah Data <i>User</i> .....	61
Gambar 5.18 Tambah Data <i>User</i> Sukses .....	62
Gambar 5.19 Halaman <i>Edit User</i> .....	63
Gambar 5.20 Peringatan Hapus Data <i>User</i> .....	64
Gambar 5.21 <i>Add Bahan Baku</i> .....	65
Gambar 5.22 Peringatan Kolom Kosong <i>Add Bahan Baku</i> .....	65
Gambar 5.23 <i>Add Bahan Baku</i> Sukses.....	66
Gambar 5.24 Halaman <i>Edit Bahan Baku</i> .....	67
Gambar 5.25 Peringatan Hapus Bahan Baku.....	68
Gambar 5.26 <i>Add Produk</i> .....	69
Gambar 5.27 Peringatan Kolom Kosong <i>Add Produk</i> .....	69
Gambar 5.28 <i>Add Produk</i> Sukses .....	70
Gambar 5.29 Halaman <i>Edit Produk</i> .....	71
Gambar 5.30 Peringatan Hapus Produk .....	72
Gambar 5.31 <i>Add Transaksi Pembelian</i> .....	73
Gambar 5.32 Peringatan Kolom Kosong <i>Add Transaksi Pembelian</i> .....	73
Gambar 5.33 <i>List Transaksi Pembelian</i> .....	74
Gambar 5.34 <i>Detail List Transaksi Pembelian</i> .....	74
Gambar 5.35 Halaman <i>Edit Pembelian</i> .....	75
Gambar 5.36 Hapus Transaksi Pembelian .....	76
Gambar 5.37 <i>Add Transaksi Penjualan</i> .....	76
Gambar 5.38 Peringatan Kolom Kosong <i>Add Transaksi Penjualan</i> .....	77
Gambar 5.39 <i>List Transaksi Penjualan</i> .....	77
Gambar 5.40 <i>Detail List Transaksi Penjualan</i> .....	78



Gambar 5.41 Halaman <i>Edit</i> Penjualan .....	78
Gambar 5.42 Hapus Transaksi Pembelian .....	79
Gambar 5.43 <i>Add</i> Transaksi Pemasukan Lain .....	80
Gambar 5.44 <i>Add</i> Transaksi Pengeluaran Lain.....	80
Gambar 5.45 <i>View</i> Laporan Pemasukan .....	81
Gambar 5.46 <i>View</i> Laporan Pemasukan Lain .....	81
Gambar 5.47 <i>View</i> Laporan Pengeluaran.....	82
Gambar 5.48 <i>View</i> Laporan Pengeluaran Lain.....	82
Gambar 5.49 <i>View</i> Laporan Laba Rugi Penjualan Pembelian .....	83
Gambar 5.50 <i>View</i> Laporan Laba Rugi Pemasukan Pengeluaran Lain.....	84



## **BAB 1. PENDAHULUAN**

Bab ini adalah awal untuk membuat buku Tugas Akhir ini. Pada bab ini akan ditulis tentang latar belakang, perumusan masalah, tujuan dan manfaat, batasan masalah dan sistematika penulisan buku.

### **1.1 Latar Belakang Masalah**

Perusahaan adalah suatu bentuk usaha yang melakukan kegiatan secara tetap dan terus menerus dengan tujuan memperoleh keuntungan atau laba. Proses mencatat, mengklasifikasi, meringkas, mengolah dan menyajikan data, transaksi serta kejadian yang berhubungan dengan keuangan yang dapat digunakan oleh orang dengan mudah dimengerti untuk pengambilan suatu keputusan sering disebut sebagai akuntansi.

Akuntansi sering disebut sebagai bahasa bisnis (business language), atau lebih tepatnya sebagai bahasa pengambilan keputusan (Muawanah & Poernawati, 2008, hal. 1). Fungsi utama akuntansi adalah sebagai informasi keuangan suatu organisasi. Dari laporan akuntansi kita bisa melihat posisi keuangan suatu organisasi beserta perubahan yang terjadi di dalamnya. Akuntansi dibuat secara kualitatif dengan satuan ukuran uang. Informasi mengenai keuangan sangat dibutuhkan khususnya oleh pihak manajer atau manajemen untuk membantu membuat keputusan suatu organisasi.

UD Cobra Sakti merupakan usaha yang bergerak di bidang penjualan tas, dompet, sepatu, jaket yang terbuat dari bahan kulit ular. Usaha ini berada di Dusun Komis Wetan desa Wonosobo RT 001 RW 007 kecamatan Srono kabupaten Banyuwangi. UD Cobra Sakti memasarkan produksinya ke art shop yang ada di Bali. Seiring dengan berjalannya waktu UD Cobra Sakti terus mengalami perkembangan, pemesanan barang semakin meningkat. Tidak hanya dari Bali namun sudah sampai

ke luar negeri seperti Mesir, Korea dll. Semakin besarnya pesanan UD Cobra Sakti harus memiliki manajemen pencatatan transaksi yang tepat.

Pada perusahaan UD Cobra Sakti proses pencatatan keuangannya kurang terinci, sehingga terjadi banyak kesalahan dalam menentukan pendapatan. UD Cobra Sakti masih menggunakan pencatatan secara manual yaitu dengan menggunakan pencatatan pada buku. Proses pengklasifikasian transaksi keluar masuk penjualan juga masih belum tepat. Bukti-bukti transaksi yang ada setiap bulannya juga masih belum teringkas dalam satu wadah. Selain itu, pada UD Cobra Sakti proses mengolah data dalam perusahaan, yang mencakup pengumpulan, pemerosesan, dan pengawasan hasil olahan masih belum terlaksana. Proses penyajian data transaksi pendapatan keuangan pada UD Cobra Sakti ini juga kurang baik, karena UD Cobra Sakti tidak pernah mengetahui berapa jumlah laba rugi yang diperoleh selama proses transaksi perbulannya.

Kondisi yang menjadikan perlunya sebuah aplikasi untuk menangani masalah keuangan perusahaan sehingga laporan keuangan tersebut dapat membantu sebuah pengambilan keputusan perusahaan untuk mencapai tujuan dan untuk menyediakan catatan lengkap mengenai pertanggungjawaban dan perlindungan kekayaan perusahaan. Peranan teknologi informasi terhadap perkembangan akuntansi yang pertama karena efisien, penghematan waktu dan biaya, kedua karena termasuk peningkatan efektifitas, mencapai hasil atau output laporan keuangan dengan benar.

Penelitian aplikasi akuntansi sebelumnya telah dilakukan, namun pada penelitian sebelumnya yang berjudul Pembuatan Sistem Informasi Keuangan Berbasis Web pada UD. Saputra Jaya (Irnama Sari, 2011) tidak menggunakan metode pengembangan sistem. Penelitian ilmiah yang akan penulis lakukan menggunakan metode pengembangan sistem model waterfall. Model waterfall dipilih pada penelitian ini karena document pengembangan sistem sangat terorganisir, karena

setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya.

## **1.2 Perumusan Masalah**

Berdasarkan uraian diatas maka permasalahan yang timbul dalam penelitian ini adalah bagaimana pengembangan Aplikasi Akuntansi pada UD Cobra Sakti.

## **1.3 Tujuan dan Manfaat**

Adapun Tujuan dan Manfaat pada penelitian ini adalah sebagai berikut:

### **1.3.1 Tujuan**

Tujuan yang akan dicapai dalam penelitian ini adalah:

1. Merancang dan membuat sebuah sistem yang dapat mengelola dan menyajikan laporan keuangan pada UD. Cobra Sakti.
2. Sebagai bahan dasar untuk menyelesaikan karya tulis skripsi.

### **1.3.2 Manfaat**

Manfaat yang diharapkan dari penelitian ini adalah:

#### **1. Bagi Penulis**

Meningkatkan keilmuan tentang Sistem Informasi sekaligus sebagai media bagi penyelesaian Tugas Akhir untuk jenjang S-1 pada Program Studi Sistem Informasi, Universitas Jember.

#### **2. Bagi UD. Cobra Sakti**

Dengan dibangunnya Sistem Informasi Akuntansi di UD. Cobra Sakti, maka dapat membantu dalam pembuatan laporan keuangan yang pada awalnya masih menggunakan cara manual.

#### 1.4 Batasan Masalah

Adapun batasan masalah dari penelitian ini yaitu:

- a. Tidak membahas masalah penggajian karyawan per orang melainkan hanya dicantumkan secara global.
- b. Sistem dapat dibuka optimal pada browser chrome.
- c. Terdapat tiga level user yang diperkenankan mengakses sistem (administrator, kasir dan staff).
- d. Sistem yang dibuat telah sesuai dengan permintaan UD. Cobra Sakti dan sesuai dengan kebutuhan pengguna.

#### 1.5 Sistematika Penulisan

Sistematika dan keruntutan penyusunan tugas akhir ini disusun sebagai berikut:

1. Pendahuluan  
Bab pendahuluan terdiri dari latar belakang, rumusan masalah, tujuan dan manfaat, batasan masalah dan Sistematika penulisan.
2. Tinjauan Pustaka  
Bab ini akan menjelaskan mengenai tinjauan pustaka, definisi, dan penjelasan metode dan teori sesuai dengan permasalahan yang akan dibahas.
3. Metodologi Penelitian  
Bab ini membahas tentang tahapan penelitian yang akan dilakukan dan diawali dengan menganalisa kebutuhan, Perancangan (*Design*), Pengkodean (*Coding*), Uji Coba (*Testing*), serta *Operation and Maintenance*.
4. Analisis dan Perancangan Sistem  
Bab ini akan menguraikan pembahasan Desain dan Perancangan sistem yang akan di rancang.

5. Hasil dan Pembahasan

Bab ini akan menjelaskan hasil dari penelitian dan pembahasan yang telah dilakukan

6. Penutup

Bab ini berisi Kesimpulan dan Saran.



## BAB 2. TINJAUAN PUSTAKA

Pada bab ini menguraikan secara detail mengenai teori yang mendasari penjelasan tentang isi dari bahan serta konsep yang akan dijadikan dalam kerangka pemikiran dalam penelitian.

### 2.1 Aplikasi Akuntansi

Aplikasi merupakan suatu kelompok file (form, class, report) yang bertujuan untuk melakukan aktivitas tertentu yang saling terkait (Santoso, 2000, hal. 9). Lebih jelasnya aplikasi itu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju. Aplikasi biasanya berupa perangkat lunak yang berbentuk software yang berisi kesatuan perintah atau program yang dibuat untuk melaksanakan sebuah pekerjaan yang diinginkan.

Akuntansi adalah suatu proses pencatatan, penggolongan, peringkasan dan pelaporan atas transaksi keuangan perusahaan serta implementasinya (Kusrini & Koniyo, 2007, hal. 16). Jadi akuntansi adalah sebagai informasi keuangan suatu organisasi. Dari laporan akuntansi kita bisa melihat posisi keuangan suatu organisasi beserta perubahan yang terjadi didalamnya. Akuntansi dibuat secara kualitatif dengan satuan ukuran uang. Informasi mengenai keuangan sangat dibutuhkan khususnya oleh pihak manajer manajemen untuk membantu membuat keputusan suatu organisasi.

Akuntansi bertujuan untuk menyiapkan suatu laporan keuangan yang akurat agar dapat dimanfaatkan oleh para manajer, pengambil kebijakan, dan pihak berkepentingan lainnya, seperti pemegang saham, kreditur, atau pemilik. Akuntansi seringkali dinyatakan sebagai bahasa perusahaan yang berguna untuk memberikan informasi yang berupa data-data keuangan perusahaan yang dapat



digunakan guna pengambilan keputusan. Setiap perusahaan memerlukan dua macam informasi tentang perusahaannya yaitu informasi mengenai nilai perusahaan dan informasi tentang laba/rugi usaha. Untuk memperoleh informasi-informasi tersebut diatas, pengusaha hendaknya mengadakan catatan yang teratur mengenai transaksi-transaksi yang dilakukan perusahaan yang dinyatakan dalam satuan uang. Pada suatu laporan akuntansi harus mencantumkan nama perusahaan, nama laporan, dan tanggal penyusunan atau jangka waktu laporan tersebut untuk memudahkan orang lain memahaminya.

Aplikasi yang dirancang untuk memudahkan aktivitas dan pencatatan akuntansi ke dalam modul-modul spesifik seperti pembelian, penjualan, buku besar dan lain-lain disebut dengan aplikasi akuntansi. Aplikasi ini bisa merupakan aplikasi yang dikembangkan sendiri oleh perusahaan, atau dibeli dari pihak ke tiga yang menyediakannya, atau dapat pula merupakan kombinasi dari keduanya. Aplikasi akuntansi ditujukan untuk memudahkan kita dalam menghasilkan laporan keuangan. Dengan aplikasi akuntansi, pemilik perusahaan ataupun seorang akuntan dapat menyajikan laporan yang dibutuhkan dengan waktu yang sangat cepat. Bila dibandingkan dengan cara kerja manual akuntansi, aplikasi akuntansi bisa lebih menghemat waktu.

## 2.2 Aplikasi Berbasis Web

Menurut (Prasetyo, 2008) Web adalah sebuah sistem dengan informasi yang disajikan dalam bentuk teks, gambar, suara, dan lain-lain yang tersimpan dalam sebuah server Web Internet yang disajikan dalam bentuk hiperteks. Melalui web, setiap pemakai internet bisa mengakses informasi-informasi di situs web yang tidak hanya berupa teks, tetapi juga dapat berupa gambar, suara, film, animasi.

Aplikasi berbasis web merupakan aplikasi yang dibuat dengan memanfaatkan mekanisme dan aplikasi yang sudah ada pada sistem web (www). Ada banyak aplikasi berbasis web yang tersedia saat ini, antara lain : Chatting, sistem informasi,



email, forum, penjualan online, dll. Dengan penggunaan sistem berbasis web akan didapatkan beberapa keunggulan :

1. Pengaksesan dapat dilakukan dimanapun.
2. Dapat diakses hanya dengan menggunakan web browser.
3. Tidak perlu menginstal aplikasi disetiap komputer.

### 2.3 Laba Rugi

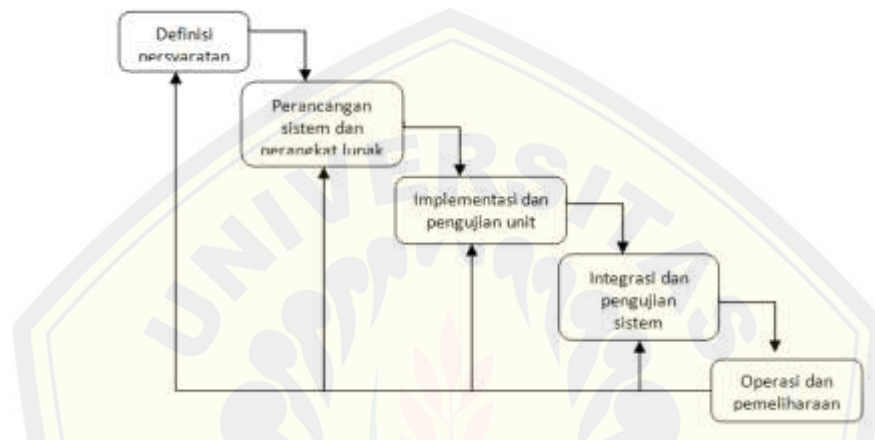
Di suatu akhir periode akuntansi perusahaan ada dua hasil yang sering terjadi, yaitu laba atau rugi. Laba rugi adalah pendapatan dikurangi seluruh pengeluaran atau pengorbanan yang telah dikeluarkan (Ir. Kuswandi, 2008). Pengeluaran yang dimaksud umumnya dikategorikan sebagai beban/biaya. Laporan Laba-Rugi dapat dibuat dengan cara semua pendapatan dikelompokkan tersendiri di bagian atas dan dijumlahkan, kemudian semua beban dikelompokkan tersendiri di bagian bawah dan dijumlahkan. Jumlah pendapatan dikurangi jumlah beban, selisihnya merupakan laba bersih atau rugi bersih.

$$\text{LABA (RUGI)} = \text{PENDAPATAN} - \text{BEBAN/BIAYA}$$

Laba rugi biasanya digunakan sebagai alat untuk mengetahui kemajuan yang dicapai perusahaan dan juga mengetahui berapakah hasil bersih atau laba yang didapat dalam suatu periode. Laba rugi juga sering dimanfaatkan sebagai ukuran untuk menilai prestasi perusahaan. Hasil akhir dari suatu laporan laba rugi adalah keuntungan bersih atau kerugian. Laporan laba-rugi ini cukup penting keberadaannya, karena laporan ini dapat dijadikan alat untuk memprediksi arus kas dimasa mendatang, banyak pemakai laporan keuangan yang memakai laporan laba-rugi ini untuk memprediksi arus kas masa depan, seperti para investor dan kreditor.

## 2.4 Model SDLC Waterfall

Model SDLC *Waterfall* sering juga disebut sekuensial linier (*sequential linier*) atau alur hidup klasik. Model waterfall merupakan sebuah model pengembangan software yang bersifat sekuensial dan terdiri dari 5 tahap yang saling terkait dan mempengaruhi seperti terlihat pada gambar berikut.



Gambar 2.1 Siklus Hidup Perangkat Lunak

(Hanum, 2003, hal. 43)

Model waterfall dipilih pada penelitian ini karena kualitas dari sistem yang dihasilkan akan baik. Ini dikarenakan oleh pelaksanaannya secara bertahap sehingga tidak terfokus pada tahapan tertentu. Document pengembangan sistem sangat terorganisir, karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya.

## 2.5 Proses Bisnis

UD Cobra Sakti merupakan jenis usaha yang bergerak di bidang kerajinan yang berbahan dasar kulit ular. Bahan baku utama berupa kulit ular yang didapat dari bekerjasama dengan pengepul di daerah Kalimantan dan Sumatra. Bahan baku ini diolah menjadi beberapa produk berupa tas, dompet, sepatu, jaket, sandal dan lain-lain. Bahan baku kulit ular akan melalui beberapa tahapan proses sebelum menjadi

produk yang mempunyai nilai ekonomis tinggi. Tahap pertama dari proses tersebut adalah penyamakan. Penyamakan merupakan proses pengolahan bahan baku kulit ular menjadi barang setengah jadi yang akan diproses pada tahap selanjutnya. Pengolahan bahan baku ini melalui beberapa proses pengolahan, seperti pelemasan kulit ular yang dilakukan dengan cara direndam, pewarnaan kulit ular, pengeringan kulit ular dan proses penyetrikaan kulit ular kering yang sudah berwarna.

Adapun macam-macam bahan baku dalam proses ini adalah sebagai berikut:

1. Kulit Ular
2. ZA
3. Oropon
4. Garam
5. FA
6. Chrom
7. Pewarna
8. Soda kue
9. Sintan
10. Minyak
11. Paku
12. SN
13. Kapur

Tahap kedua adalah proses pembuatan kulit ular menjadi barang jadi. Pada proses ini kulit ular akan melalui beberapa tahap, antara lain proses pemotongan kulit ular sesuai dengan pola yang diinginkan, penyemprotan lem pada kulit ular yg sudah dipotong, pemasangan magnet bila menggunakan magnet, pemasangan releting jika pola yang dibuat menggunakan resleting, pemasangan accesories bila menggunakan accesories dan terakhir proses penjahitan kulit ular menjadi barang jadi.

Adapun macam-macam bahan baku pada proses ini adalah sebagai berikut:

1. Ring D
2. Ring kotak
3. Ring jalan
4. Ring o
5. Kaitan
6. Keling jamur
7. Keling jamur
8. Keling gepeng
9. Kaki bulat
10. Kaki pyramid
11. Magnet tanam
12. Magnet temple
13. Kunci
14. Mata ayam
15. Rantai
16. Resleting
17. Kepala resleting
18. AHE
19. CCI
20. Spon busa
21. Kertas karton
22. Solex
23. Lem
24. Latex
25. Sumbu kompor
26. Tali kur
27. Kain kaos
28. Kain suede

- 29. Benang
- 30. Plastik bungkus
- 31. Pengawet

Tahap selanjutnya yaitu tahap ketiga dimana pada tahap ini merupakan proses transaksi. Pada proses transaksi ini dilakukan dengan cara memasukkan setiap transaksi kedalam sebuah buku nota secara terinci dan dilakukan proses penjumlahan. Tahap keempat merupakan tahap terakhir dalam proses ini tahapan tersebut adalah penjualan barang jadi kepada konsumen. Penjualan ini juga bisa disebut sebagai proses pengiriman barang. Tahap pengiriman barang ini dilakukan dengan menggunakan jasa pengiriman barang, sehingga konsumen dapat menerima secara langsung barang yang dipesan.

Pada Aplikasi Akuntansi UD Cobra Sakti Berbasis Web ini memiliki tiga aktor, antara lain admin, staff dan kasir. Admin dalam sistem memiliki hak penuh untuk menggunakan sistem. Staff hanya memiliki hak dapat memanajemen data produk, memanajemen data bahan baku dan menambah data pembelian. Sedangkan kasir memiliki hak menambah data penjualan dan menambah pemasukan dan pengeluaran lain.

## BAB 3. METODOLOGI PENELITIAN

Penelitian ini menggunakan metode kualitatif, yang merupakan pendekatan yang temuan-temuan penelitiannya tidak diperoleh melalui prosedur statistik atau bentuk perhitungan lainnya, prosedur ini menghasilkan temuan-temuan yang diperoleh dari data-data yang dikumpulkan dengan menggunakan beragam sarana. Sarana itu meliputi pengamatan dan wawancara, namun bisa juga mencakup dokumen, buku, kaset video, dan bahkan data yang telah dihitung untuk tujuan lain, misalnya data sensus.

Pada pelaksanaannya peneliti membagi menjadi beberapa tahap meliputi tahapan-tahapan perencanaan sistem, tahap implementasi dan metode pengembangan perangkat lunak.

### 3.1 Objek Penelitian

Penelitian dilakukan di UD. Cobra Sakti bertempat di dusun Komis Wetan desa Wonosobo RT 001 RW 007 kecamatan Srono kabupaten Banyuwangi. Pada penelitian ini peneliti mengambil data bahan baku, data produk dan transaksi pada perusahaan.

### 3.2 Metode Pengembangan Aplikasi

Metode pengembangan aplikasi pada penelitian ini menggunakan pemodelan *waterfall* dengan beberapa tahapan seperti definisi persyaratan, perancangan sistem dan perangkat lunak, implementasi dan pengujian unit, integrasi dan pengujian sistem dan operasi dan pemeliharaan.

#### 3.2.1 Definisi Persyaratan

Tahap ini merupakan tahap analisa kebutuhan aplikasi akuntansi UD Cobra Sakti. Proses pengumpulan kebutuhan dilakukan secara insentif untuk

mespesifikasikan kebutuhan perangkat lunak agar mudah dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Pengumpulan data dilakukan dengan cara melakukan kunjungan wawancara secara langsung pada UD Cobra Sakti.

### 3.2.2 Perancangan Sistem dan Perangkat Lunak

Aplikasi akuntansi pada UD Cobra Sakti ini menggunakan beberapa diagram perancangan perangkat lunak yang dibuat sesuai data yang telah dianalisis sebelumnya dan fitur sistem yang akan dibangun antara lain *Bussiness Proses*, *Use Case Diagram*, *Use Case Scenario*, *Sequence Diagram*, *Class Diagram* dan *Entity Relationship Diagram* (ERD).

### 3.2.3 Implementasi dan Pengujian Unit

Pada tahap pengodean dilakukan penulisan kode program sesuai desain yang telah dibuat sebelumnya. Beberapa proses pengodean yang dilakukan antara lain :

- (1) Penulisan kode program menggunakan bahasa pemrograman PHP (*Page Hyper Text Pre-Processor*).
- (2) Manajemen data menggunakan Basis Data MySQL.
- (3) Menggunakan XAMMP sebagai server.

### 3.2.4 Integrasi dan Pengujian Sistem

Pada tahap pengujian aplikasi akuntansi UD Cobra Sakti menggunakan dua metode pengujian, yaitu *white box testing* dan *black box testing*. *White box testing* dilakukan dengan cara melihat kedalam modul untuk mengevaluasi kode-kode program yang telah ditulis apakah ada kesalahan. Sedangkan untuk *black box testing* pengujian dilakukakn dengan menitikberatkan pada fungsionalitas dari aplikasi yang dibuat. Pengujian *black box testing* ini dilakukan untuk mengevaluasi fungsi, *interfae*, struktur data, serta performa aplikasi. Pengujian akan dilakukan oleh pengembang, Dosen Pembimbing Utama, Dosen Pembimbing Pendamping, dan pihak UD Cobra Sakti.



### 3.2.5 Operasi dan Pemeliharaan

Setelah aplikasi telah selesai dibangun maka aplikasi telah dapat di implementasikan. Implementasi sistem menunjukkan bahwa pembangunan sistem telah selesai dikerjakan dan pertama kalinya aplikasi akan dipergunakan di UD. Cobra Sakti. Kegiatan selanjutnya adalah pemeliharaan aplikasi yang akan dilakukan secara berkala.

### 3.3 Perangkat Keras

Kebutuhan yang diperlukan untuk mengakses Aplikasi Akuntansi UD Cobra Sakti ini adalah PC atau Laptop dengan spesifikasi sebagai berikut:

1. Satu unit *computer processor*
2. RAM (*Random Access Memory*) minimal 1GB
3. *Monitor*
4. *Keyboard*
5. *Mouse*

### 3.4 Perangkat Lunak

Perangkat lunak yang digunakan untuk memfasilitasi perkembangan sistem, yaitu:

1. *Sistem Operasi Windows XP/7/8/Vista*
2. *NetBeans IDE* versi 7.3.1 sebagai PHP editor
3. *Google Chrome* sebagai Browsing
4. XAMPP untuk manajemen *database*



## BAB 4. ANALISIS DAN PERANCANGAN SISTEM

Bab ini akan membahas tentang analisis dan perancangan sistem untuk membuat aplikasi akuntansi UD Cobra Sakti berbasis web dengan menggunakan model *waterfall* yaitu suatu proses pembuatan situs web secara terstruktur dan berurutan dimulai dari penentuan masalah, analisa kebutuhan, perancangan implementasi, integrasi, uji coba sistem, penempatan situs web dan pemeliharaan.

### 4.1 Analisis Sistem

Pada tahap pertama dilakukan pendefinisian kebutuhan untuk merancang dan membangun aplikasi akuntansi UD Cobra Sakti. Literatur-literatur dianalisis sehingga kebutuhan sistem didefinisikan ke dalam kebutuhan fungsional dan kebutuhan non fungsional.

#### 4.1.1 Kebutuhan Fungsional

Kebutuhan fungsional di definisikan sebagai proses yang dilakukan perangkat lunak supaya inputan atau masukan data diterima dan di proses untuk menghasilkan keluaran sesuai dengan kebutuhan. Kebutuhan fungsional dalam Aplikasi Akuntansi UD Cobra Sakti digambarkan pada Gambar 4.1

Tabel 4.1 Kebutuhan Fungsional Sistem

(Sumber: Hasil Analisa, 2015)

SRSF_ID	Identifikasi
SRSF 01	Sistem dapat mengenali user yang berhak atau tidak untuk menggunakan sistem dengan adanya fitur Login.

SRSF 02	Sistem dapat melakukan manajemen terhadap data barang( menambah, menghapus, dan mengedit).
SRSF 03	Sistem dapat mengelola data produk.
SRSF 04	Sistem dapat mengelola data bahan baku.
SRSF 05	Sistem dapat mengelola data user.
SRSF 06	Sistem dapat melakukan transaksi penjualan.
SRSF 07	Sistem dapat melakukan transaksi pembelian.
SRSF 08	Sistem dapat menambahkan transaksi pemasukan lain yang dilakukan oleh perusahaan.
SRSF 09	Sistem dapat menambahkan transaksi pengeluaran lain yang dilakukan oleh perusahaan.
SRSF 10	Sistem dapat menampilkan laporan transaksi pemasukan yang dilakukan oleh kasir.
SRSF 11	Sistem dapat menampilkan laporan transaksi pengeluaran yang dilakukan oleh kasir.
SRSF 12	Sistem dapat menampilkan laporan transaksi laba rugi yang dilakukan oleh kasir.
SRSF 13	

#### 4.1.2 Kebutuhan Non Fungsional

Kebutuhan non fungsional adalah kebutuhan sistem yang tidak secara langsung berhubungan dengan fungsi dan kebutuhan sistem. Kebutuhan non fungsional digambarkan pada Tabel 4.2

Tabel 4.2 Kebutuhan Non Fungsional Sistem

(Sumber: Hasil Analisa, 2015).

SRSFN_ID	Parameter	Identifikasi
SRSNF_01	<i>Availability</i>	Sistem harus selalu dapat diakses dari mana saja dan kapan saja selama 24 jam non-stop.
SRSNF_02	<i>Respon Time</i>	Sistem harus merespon permintaan pengguna paling lama 15 detik.
SRSNF_03	<i>user friendly</i>	Tampilan sistem mudah di mengerti pengguna agar pengguna tidak kesulitan dalam mengoperasikannya.
SRSNF_04	<i>Realibility</i>	Sistem berjalan sesuai dengan kebutuhan pengguna.

## 4.2 Desain Sistem

Pada tahapan desain sistem ini menjelaskan bagaimana proses perancangan aplikasi tersebut ke dalam sebuah desain, diagram-diagram yang nantinya akan dipergunakan sebagai acuan dalam mentransformasikan desain-desain tersebut ke dalam bahasa pemrograman. Desain untuk sistem informasi optimasi penjadwalan pekerjaan proyek ini menggunakan *Unified Modeling Language* (UML) yang dirancang menggunakan konsep *Object-Oriented Programming* (OOP). Pemodelan UML yang digunakan antara lain:

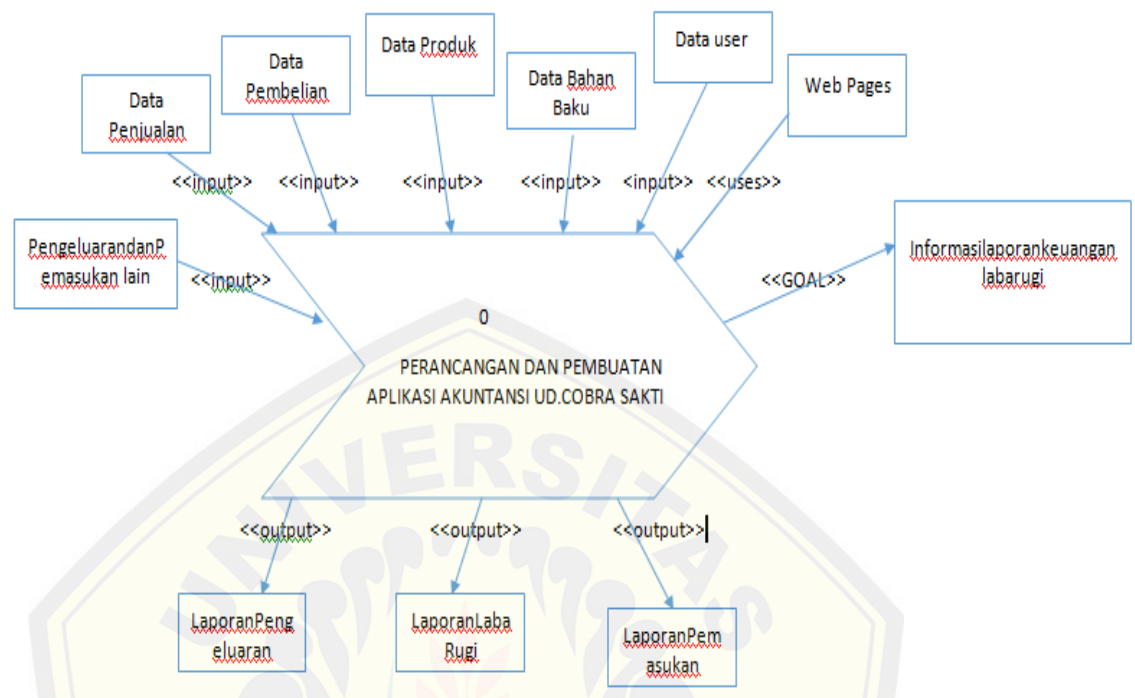
- a. *Business Process*
- b. *Use Case Diagram*
- c. *Use Case Scenario*

- d. *Activity Diagram*
- e. *Sequence Diagram*
- f. *Class Diagram, dan*
- g. *Entity Relationship Diagram (ERD)*

Diharapkan dengan gambaran diagram-diagram di atas dapat menggambarkan fungsi-fungsi aplikasi yang akan dibangun.

#### **4.2.1 *Bussiness Process***

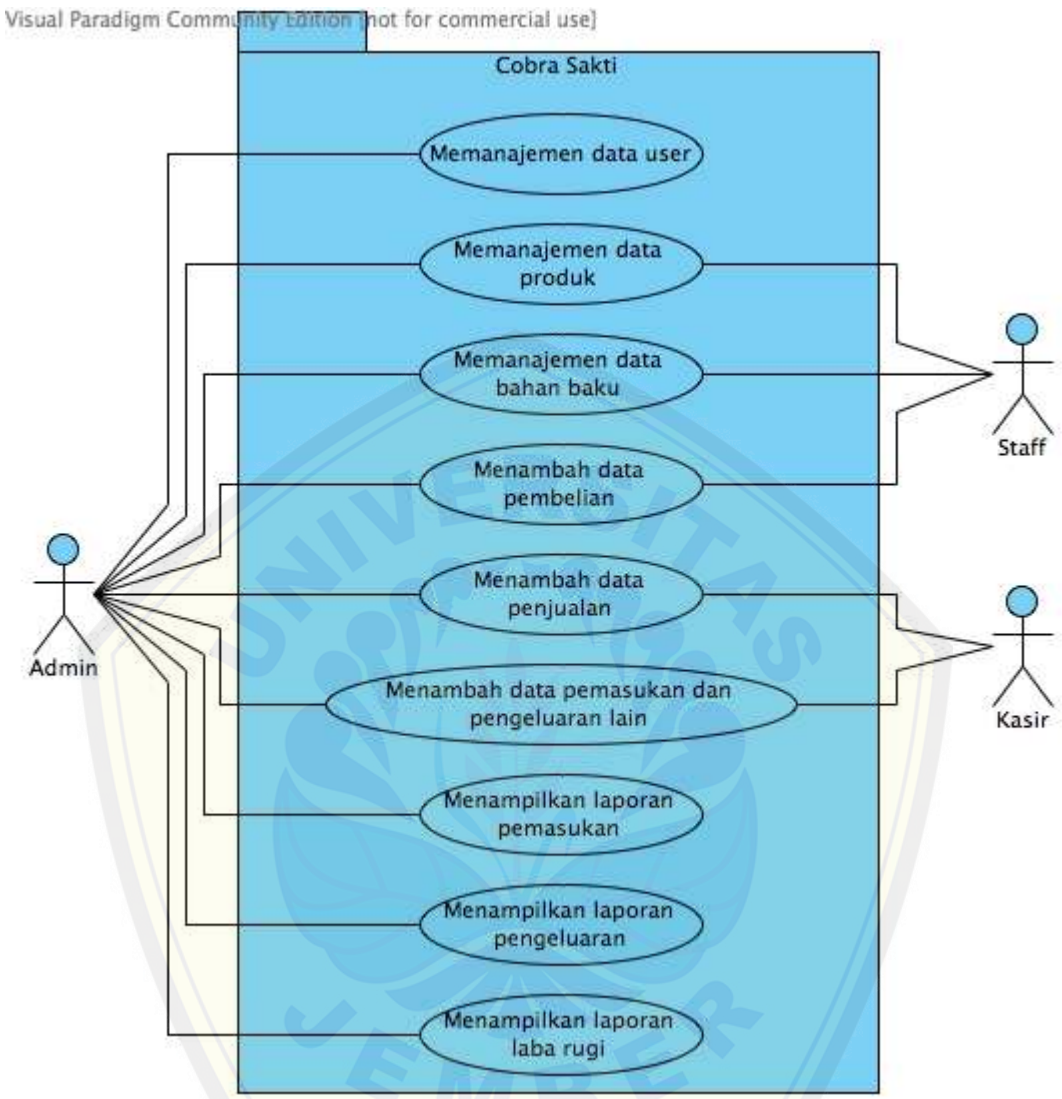
*Bussines Process* dari Aplikasi Akuntansi UD Cobra Sakti ini merupakan sebuah kumpulan proses yang dilakukan untuk mencapai hasil yang diinginkan user yaitu mengetahui setiap transaksi penjualan dan pembelian setiap harinya sehingga pemilik perusahaan bisa mendapatkan berapa jumlah laba rugi yang diperoleh perusahaan setiap bulannya. Di dalam sistem tersebut terdapat komponen untuk mencapai tujuan berupa inputan yaitu data penjualan, data pembelian, data produk, data bahan baku dan data user sehingga akan dihasilkan data keluaran (*output*) berupa laporan pemasukan, laporan pengeluaran dan laporan laba rugi dilihat pada Gambar 4.1



Gambar 4.1 *Business Process*  
 Sumber : (Hasil Analisis,2015).

**4.2.2 Usecase Diagram**

*Use Case Diagram* adalah penggambaran fitur dan aktor yang terdapat pada sistem yang dibuat sesuai dengan hak akses yang dimiliki oleh masing – masing aktor. *Usecase diagram* dalam pembuatan Aplikasi Akuntansi UD Cobra Sakti ini ada 9 dan terdiri dari 3 aktor yaitu administrator, kasir dan staff yang terdapat pada Gambar 4.2, untuk deskripsi *usecase diagram* pada Gambar 4.2 terdapat pada tabel 4.3 dan Hak akses Pengguna di tabel 4.4.



Gambar 4.2 Use Case Diagram

Sumber : (Hasil Analisis,2015).

Tabel 4.3 Deskripsi *Usecase Diagram*

(Sumber: Hasil Analisa, 2015).

<i>Usecase</i>	Deskripsi
<i>Login</i>	Merupakan <i>usecase</i> yang menjelaskan proses <i>user</i> sebagai admin dan pemilik untuk melakukan login atau autentifikasi masuk ke <i>system</i> sesuai dengan hak akses.
Memanejemenn Data <i>User</i>	Merupakan <i>usecase</i> yang menjelaskan proses <i>user</i> sebagai admin untuk menambah, mengedit dan menghapus data user ke sistem.
Memanejemen Data Produk	Merupakan <i>usecase</i> yang menjelaskan proses <i>user</i> sebagai staff untuk menambah, mengedit dan menghapus data produk di sistem.
Memanejemen Data Bahan Baku	Merupakan <i>usecase</i> yang menjelaskan proses <i>user</i> sebagai staff untuk menambah, mengedit dan menghapus data bahan baku di system
Menambah Data Pembelian	Merupakan <i>usecase</i> yang menjelaskan proses <i>user</i> sebagai kasir untuk menambahkan, mengedit dan menghapus data pembelian perusahaan ke sistem.
Menambah Data Penjualan	Merupakan <i>usecase</i> yang menjelaskan proses <i>user</i> sebagai kasir untuk menambahkan, mengedit dan menghapus data penjualan perusahaan ke sistem.



Menampilkan Laporan Pemasukan	Merupakan <i>usecase</i> yang menjelaskan proses <i>user</i> sebagai kasir untuk mengelola laporan pemasukan di setiap transaksi yang dilakukan perusahaan ke sistem.
Menampilkan Laporan Pengeluaran	Merupakan <i>usecase</i> yang menjelaskan proses <i>user</i> sebagai kasir untuk mengelola laporan pengeluaran di setiap transaksi yang dilakukan perusahaan ke sistem.
Menampilkan Laporan Laba Rugi	Merupakan <i>usecase</i> yang menjelaskan proses <i>user</i> sebagai kasir untuk mengelola laporan laba rugi di setiap transaksi yang dilakukan perusahaan ke sistem.
Menambah Pemasukan dan Pengeluaran lain	Merupakan <i>usecase</i> yang menjelaskan proses <i>user</i> sebagai kasir untuk menambahkan data pemasukan dan pengeluaran lain perusahaan ke sistem.



Tabel 4.4 Hak Akses Pengguna

(Sumber: Hasil Analisa, 2014).

Aktor	Akses Pengguna
Admin	Menambah data user Mengedit data user Menghapus data user Melihat laporan pemasukan Melihat laporan pengeluaran Melihat laporan laba rugi Menambah data produk Mengedit data produk Menghapus data produk Menambah data bahan baku Mengedit data bahan baku Menghapus data bahan baku Menambah data pembelian Mengedit data pembelian Menghapus data pembelian Menambah data penjualan Mengedit data penjualan Menghapus data penjualan Menambahkan data pemasukan dan pengeluaran lain
Staff	Menambah data produk Mengedit data produk Menghapus data produk Menambah data bahan baku Mengedit data bahan baku

	Menghapus data bahan baku Menambah data pembelian Mengedit data pembelian Menghapus data pembelian
Kasir	Menambah data penjualan Mengedit data penjualan Menghapus data penjualan Menambahkan data pemasukan dan pengeluaran lain

#### 4.2.3 Usecase Scenario

*Usecase Scenario* menggambarkan alur cerita pada setiap *usecase* yang telah digambarkan dalam *usecase diagram* dengan lebih detail.

##### 4.2.3.1 Usecase Scenario “Manambah Data Pembelian”

*Usecase scenario* ini menjelaskan alur untuk Menambah Pembelian. *Usecase scenario* “Manambah Data Pembelian” dapat dilihat pada tabel 4.5.

Tabel 4.5 Usecase Scenario "Menambah Data Pembelian"

(Sumber: Hasil Penelitian,2015).

<b>ID</b>	CB-01
<b>Name</b>	Menambah Data Pembelian
<b>Participating Actor</b>	Admin
<b>Entry Condition</b>	Admin ingin menginput data Pembelian
<b>Exit Condition</b>	Admin selesai menginput data Pembelian
<b>Skenario Utama</b> <b>“Input Data Pembelian”</b>	

<b>Admin</b>	<b>Sistem</b>
1. Memilih Menu Pembelian	
	2. Menampilkan form input Pembelian
3. Pilih Bahan Produksi	
	4. Menampilkan form input Pembelian dengan bahan dan harga
5. Input Jumlah	
6. Klik Submit	
	7. Menampilkan table transaksi Pembelian
8. Klik Bayar	
	9. Menampilkan table transaksi
<b>Skenario Alternatif</b>	
<b>“Pengisian Form Pembelian Baru yang Tidak Benar”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Pengisian form tambah Pembelian masih belum lengkap	
	7. Menampilkan alert gagal “Please fill out this field”
	8. Menampilkan form input Pembelian dengan bahan dan harga
<b>Skenario Alternatif</b>	
<b>“Memilih Tombol Cancel”</b>	

Admin	Sistem
6. Klik Cancel	
	7. Kembali ke halaman list Pembelian dan menampilkan tabel data Pembelian, tombol Edit dan Delete, dan detail

#### 4.2.3.2 Usecase Scenario “Menambah Data Penjualan”

Usecase scenario ini menjelaskan alur untuk Menambah Penjualan. Usecase scenario “Menambah Data Penjualan” dapat dilihat pada tabel 4.6.

Tabel 4.6 Usecase Scenario "Manajemen Data Pembelian"

(Sumber: Hasil penelitian, 2014).

<b>ID</b>	CB-01	
<b>Name</b>	Menambah Data Penjualan	
<b>Participating Actor</b>	Admin	
<b>Entry Condition</b>	Admin ingin menginput data Penjualan	
<b>Exit Condition</b>	Admin selesai menginput data Penjualan	
<b>Skenario Utama</b>		
<b>“Input Data Penjualan”</b>		
	<b>Admin</b>	<b>Sistem</b>
	1. Memilih Menu Penjualan	
		2. Menampilkan form input penjualan

3. Pilih Bahan Produksi	
4. Input Jumlah	
5. Klik Submit	
	6. Menampilkan table transaksi penjualan
7. Input discount	
8. Klik Bayar	
	9. Menampilkan form total pembayaran
10. Menginputkan jumlah uang yang dibayarkan	
11. Klik Ok	
	12. Menampilkan total kembalian
13. Klik Ok	
	14. Menampilkan form penjualan
<b>Skenario Alternatif</b>	
<b>“Pengisian Form Penjualan Baru yang Tidak Benar”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Klik submit	
	7. Menampilkan alert gagal
	1. Menampilkan form input penjualan

<b>Skenario Alternatif</b>	
<b>“Memilih Tombol Cancel”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Klik Cancel	
	7. Kembali ke halaman list Penjualan dan menampilkan tabel data Penjualan, tombol Edit dan Delete, dan detail

#### 4.2.3.3 Usecase Scenario “Menampilkan Laporan Pemasukan”

*Usecase scenario* ini menjelaskan alur untuk Menampilkan Laporan Pemasukan. . *Usecase scenario* “Menampilkan Laporan Pemasukan” dapat dilihat pada tabel 4.7.

Tabel 4.7 *Usecase Scenario* "Menampilkan Laporan Pemasukan"

(Sumber:Hasil penelitian,2015).

<b>ID</b>	CB-01
<b>Name</b>	Menampilkan Laporan Pemasukan
<b>Participating Actor</b>	Admin
<b>Entry Condition</b>	Admin ingin melihat laporan pemasukan dan laporan pemasukan lain
<b>Exit Condition</b>	Admin selesai melihat laporan pemasukan dan laporan pemasukan lain
<b>Skenario Utama</b>	

<b>“Laporan Pemasukan”</b>	
<b>Admin</b>	<b>Sistem</b>
1. Klik laporan pemasukan	
	2. Menampilkan halaman laporan pemasukan
3. Input tanggal tampilan laporan	
4. Klik produk penjualan	
5. Klik tampilkan	
	6. Menampilkan laporan sesuai tanggal
7. Klik cetak	
	8. Menampilkan halaman pdf
9. Klik pemasukan lain	
10. Klik tampilkan	
	11. Menampilkan laporan pemasukan lain sesuai tanggal
12. Klik cetak	
	13. Menampilkan halaman pdf
14. Klik close	
	15. Menampilkan halaman laporan pemasukan



#### 4.2.3.4 Usecase Scenario “Menampilkan Laporan Pengeluaran”

*Usecase scenario* ini menjelaskan alur untuk menampilkan laporan pengeluaran. *Usecase scenario* “Menampilkan Laporan Pengeluaran” dapat dilihat pada tabel 4.8.

Tabel 4.8 *Usecase Scenario* "Menampilkan Laporan Pengeluaran"

(Sumber:Hasil penelitian,2014).

<b>ID</b>	CB-01	
<b>Name</b>	Menampilkan Laporan Pengeluaran	
<b>Participating Actor</b>	Admin	
<b>Entry Condition</b>	Admin ingin melihat laporan pengeluaran dan laporan pengeluaran lain	
<b>Exit Condition</b>	Admin selesai melihat laporan pengeluaran dan laporan pengeluaran lain	
<b>Skenario Utama</b>		
<b>“Laporan Pengeluaran”</b>		
	<b>Admin</b>	<b>Sistem</b>
	1. Klik laporan pengeluaran	
		2. Menampilkan halaman laporan pengeluaran
	3. Input tanggal tampilan laporan	
	4. Klik pembelian bahan baku	
	5. Klik tampilkan	

	6. Menampilkan laporan sesuai tanggal
7. Klik cetak	
	8. Menampilkan halaman pdf
9. Klik pengeluaran lain	
10. Klik tampilkan	
	11. Menampilkan laporan pengeluaran lain sesuai tanggal
12. Klik cetak	
	13. Menampilkan halaman pdf
14. Klik close	
	15. Menampilkan halaman laporan pengeluaran

#### 4.2.4 Activity Diagram

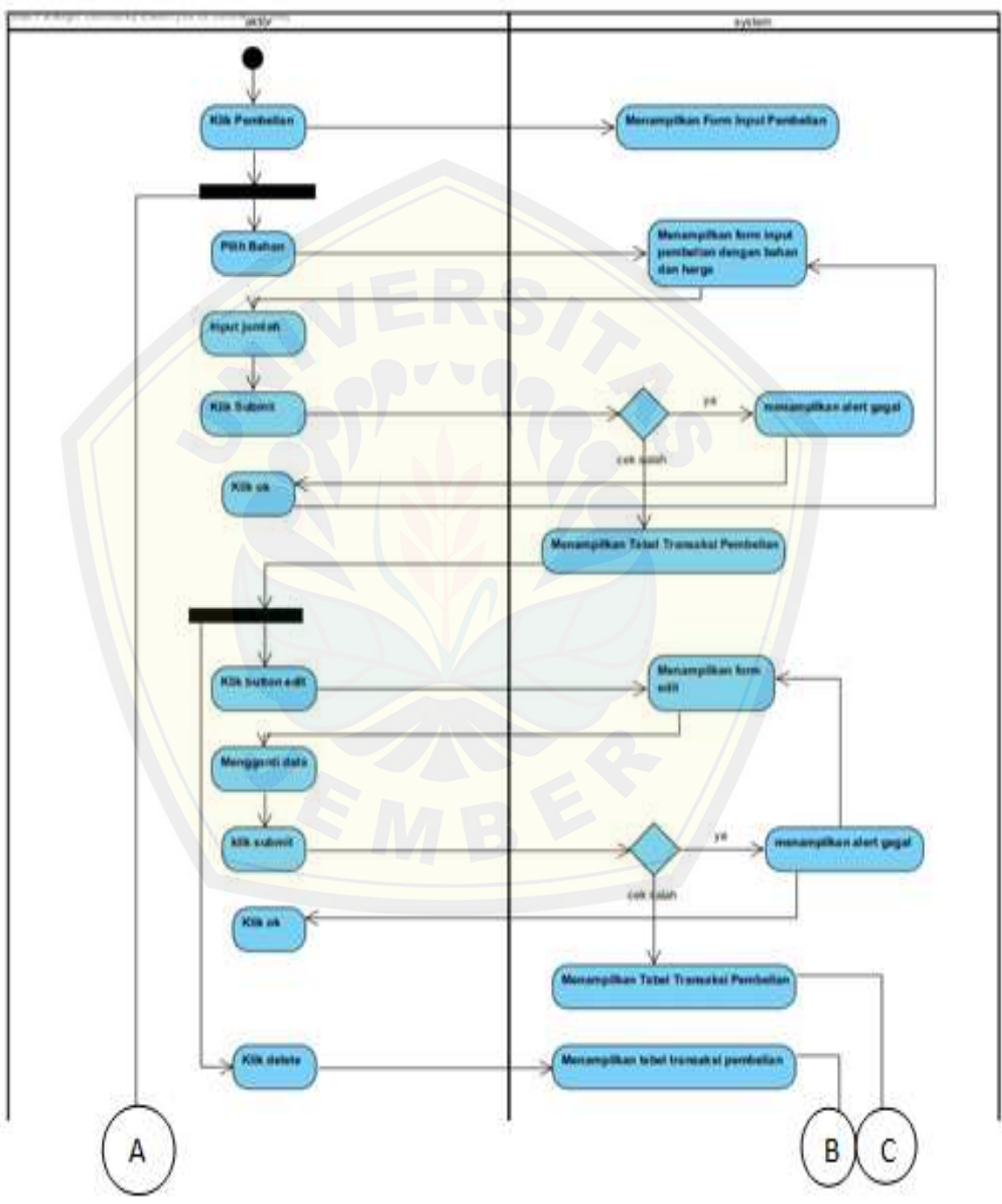
Setelah melakukan alur *usecase scenario* maka selanjutnya merancang *activity diagram*. *Activity diagram* merupakan penggambaran *flow of events* yang menunjukkan *control* aliran dari *activity* ke *activity*.

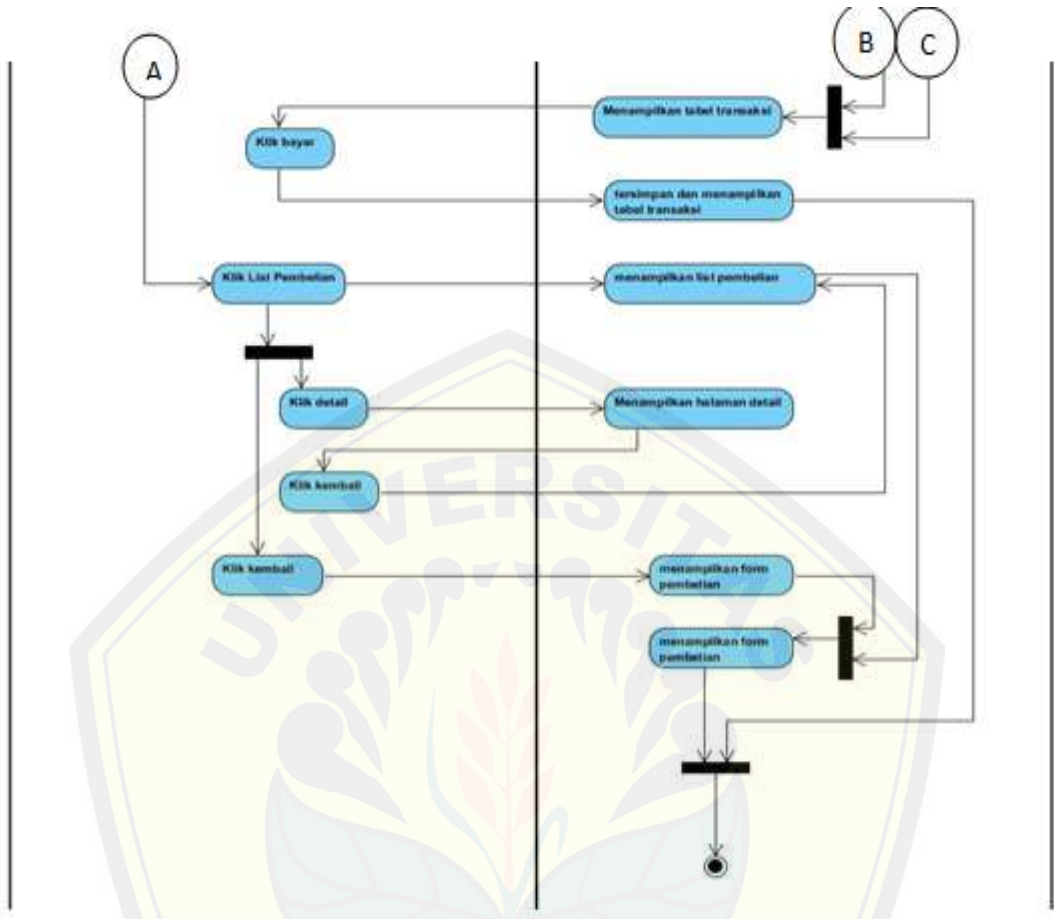
##### 4.2.4.1 Activity Diagram Menambah Data Pembelian

Setelah menggambarkan alur kerja pada *usecase scenario*, maka dilanjutkan dengan menggambarkan kedalam *activity diagram*. *Activity Diagram* pada fitur Manajemen Data Pembelian merupakan penggambaran dari alur kerja pada *usecase scenario*. Berikut penggambaran *activity diagram* menambah data pembelian dapat dilihat pada tabel 4.9.

Tabel 4.9 Activity Diagram “Menambah Data Pembelian”

(Sumber : Hasil Analisis, 2015)



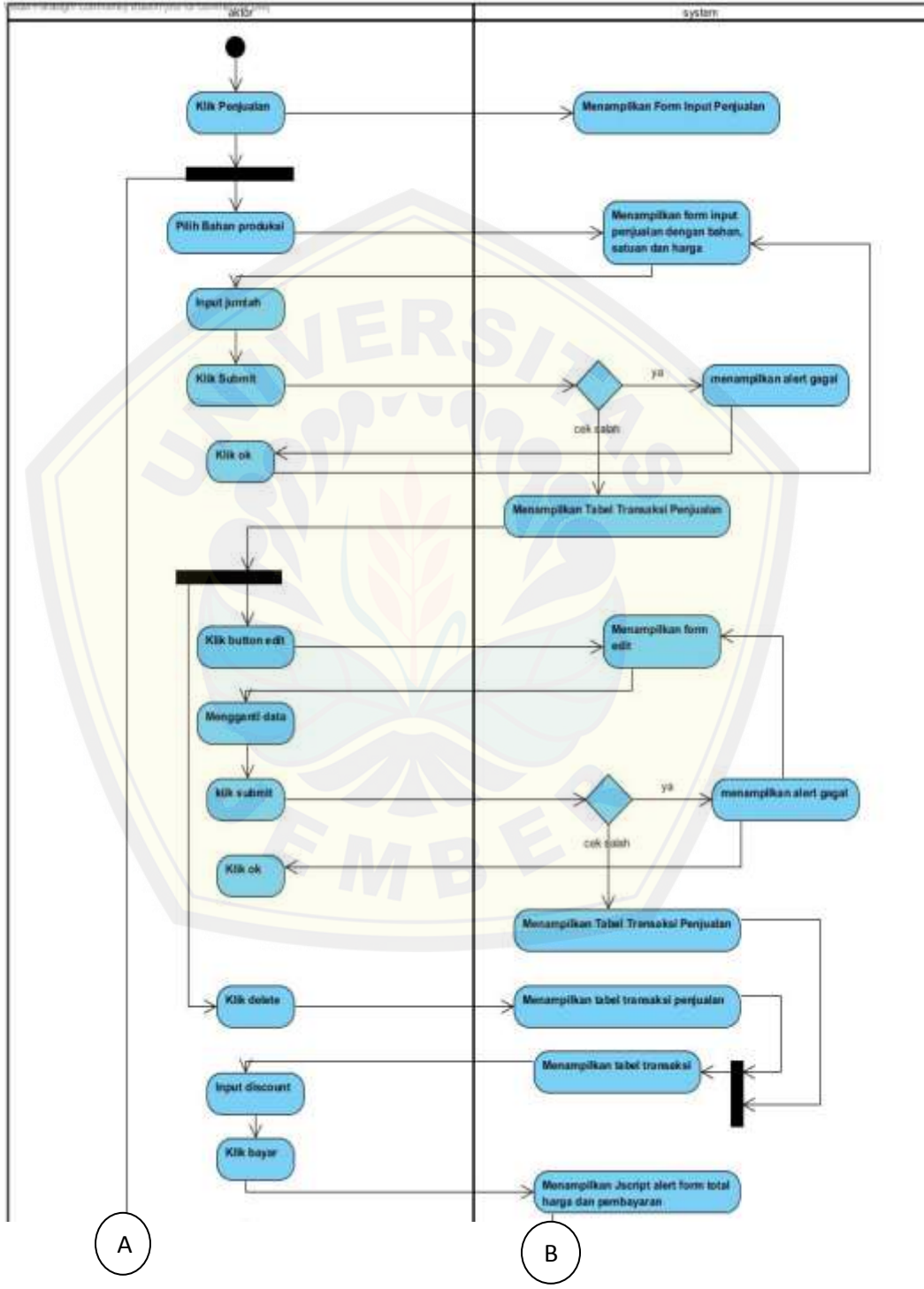


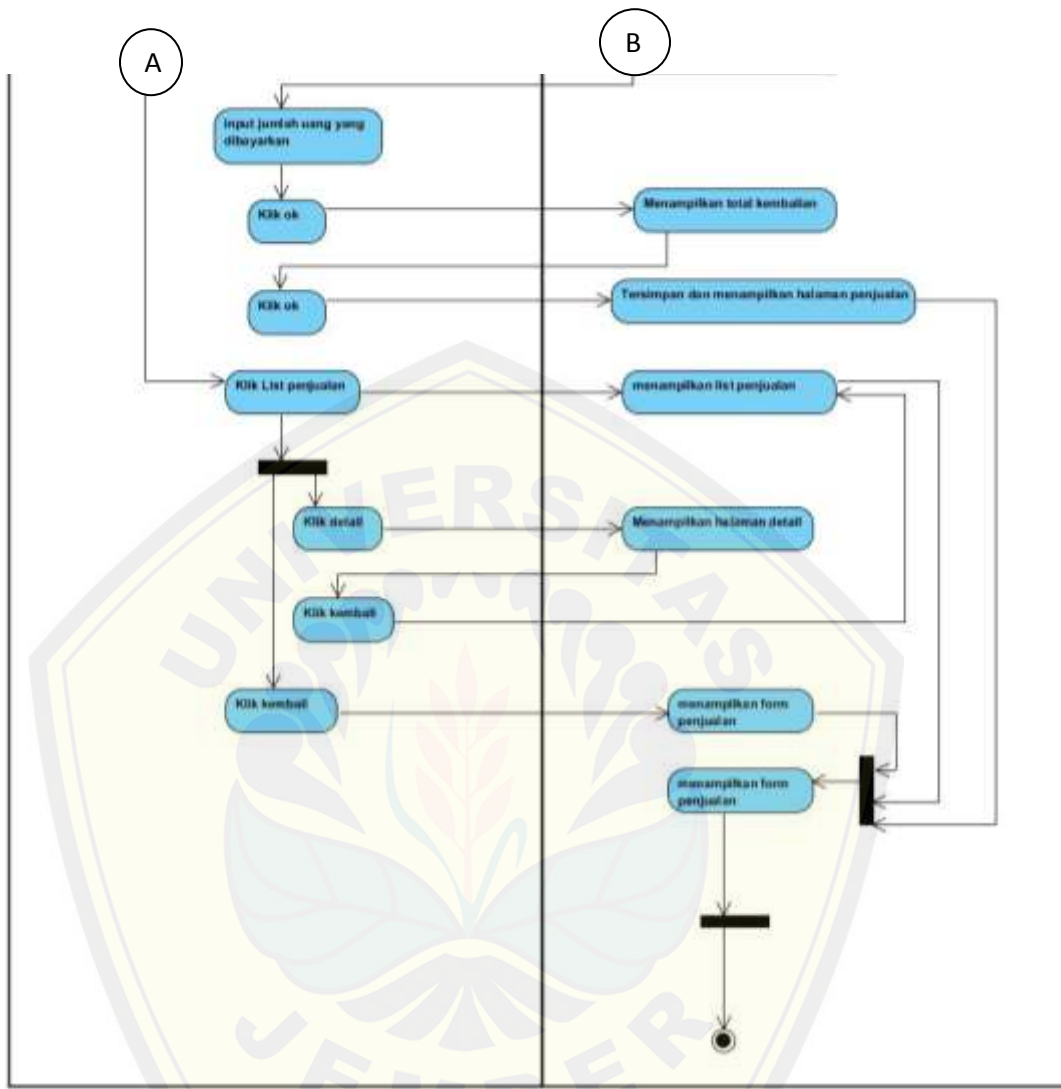
4.2.4.2 Activity Diagram Menambah Data Penjualan

Setelah menggambarkan alur kerja pada *usecase scenario*, maka dilanjutkan dengan menggambarkan kedalam *activity diagram*. *Activity Diagram* pada fitur Manajemen Penjualan (Input Data Penjualan) merupakan penggambaran dari alur kerja pada *usecase scenario*. Berikut penggambaran *activity diagram* menambah data penjualan dapat dilihat pada tabel 4.10.

Tabel 4.10 Activity Diagram “Menambah Data Penjualan”

(Sumber : Hasil Analisis, 2015).



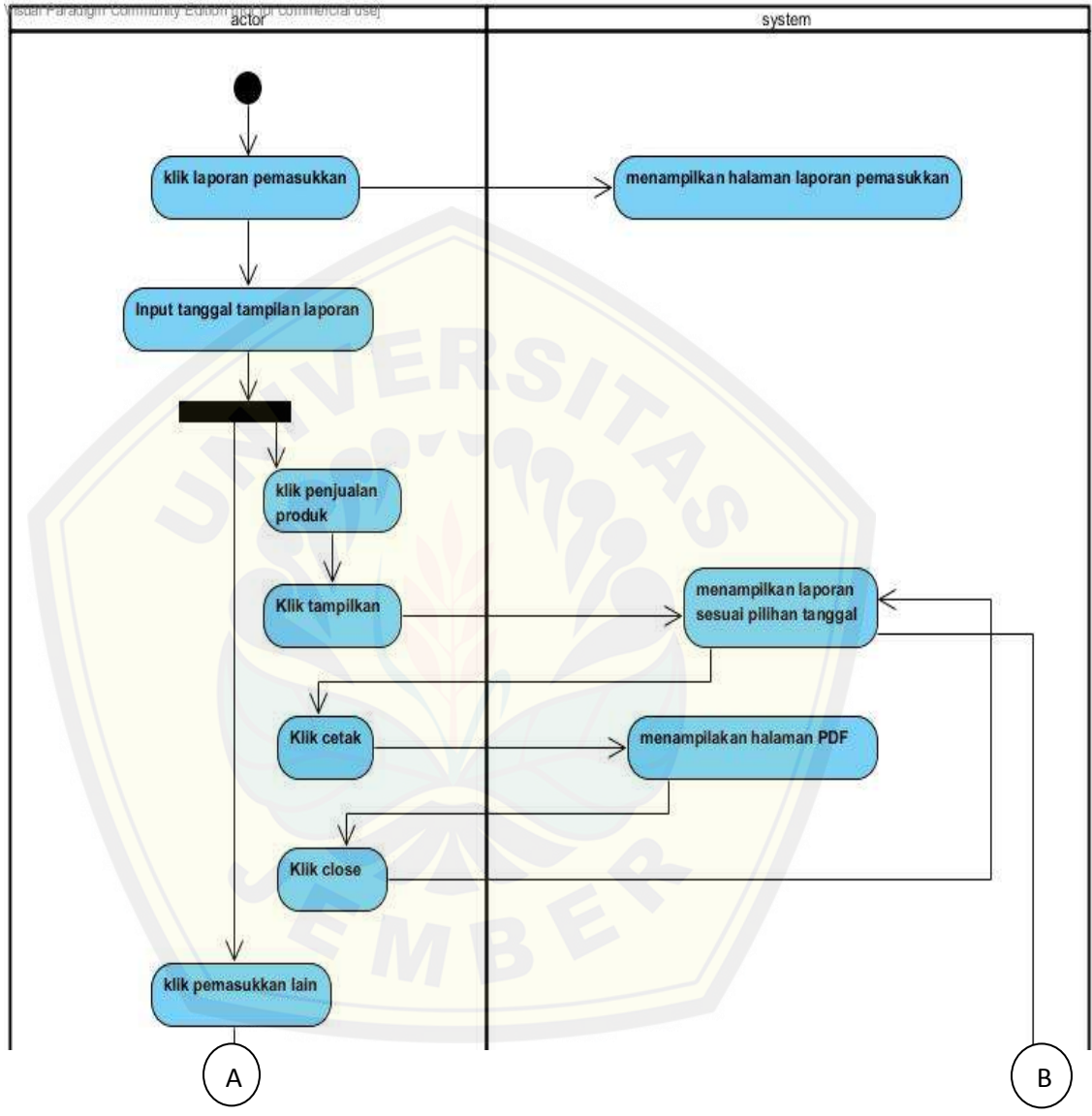


**4.2.4.3 Activity Diagram Menampilkan Laporan Pemasukan**

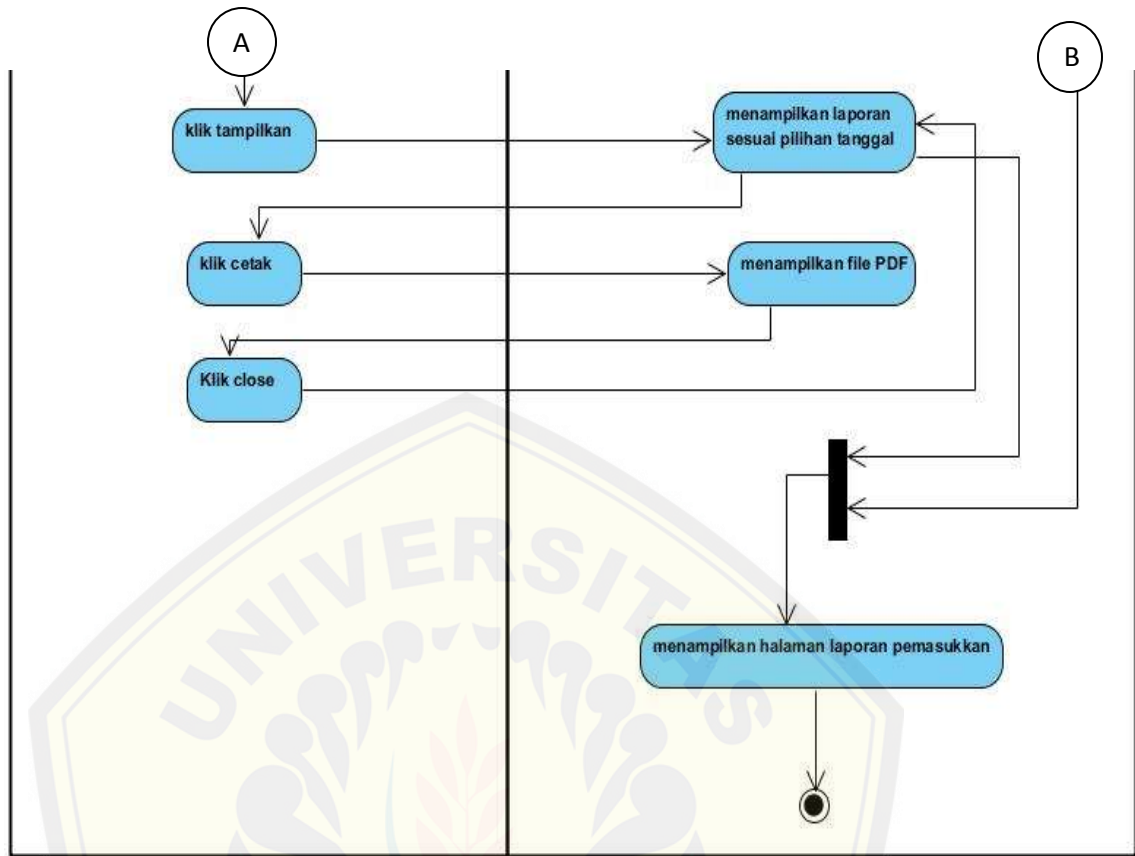
Setelah menggambarkan alur kerja pada *usecase scenario*, maka dilanjutkan dengan menggambarkan kedalam *activity diagram*. *Activity Diagram* pada fitur Pemasukan Lain merupakan penggambaran dari alur kerja pada *usecase scenario*. Berikut penggambaran *activity diagram* menampilkan laporan pemasukan dapat dilihat pada tabel 4.12.

Tabel 4.11 Activity Diagram “Menampilkan Laporan Pemasukan”

(Sumber : Hasil Analisis, 2015)





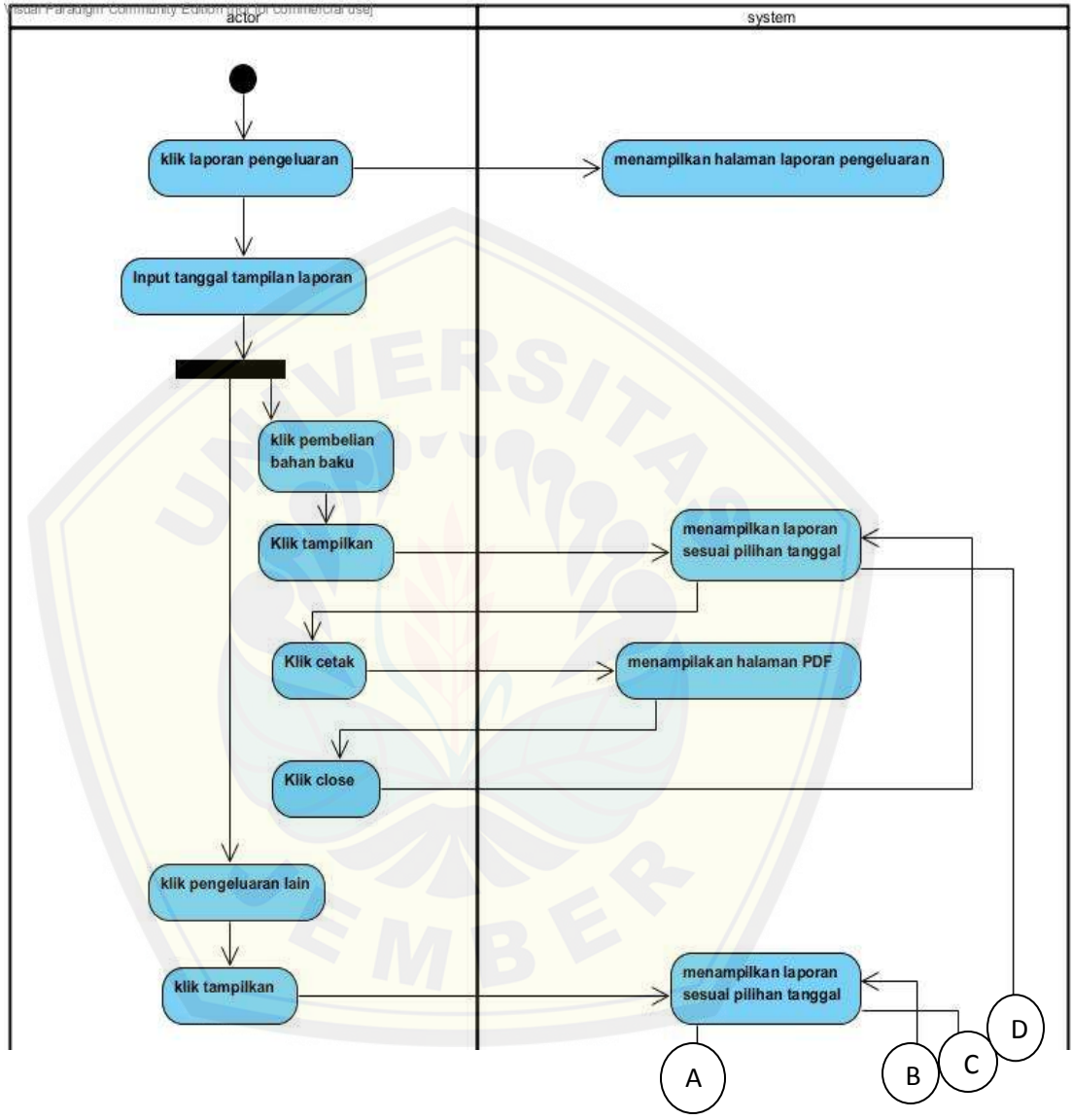


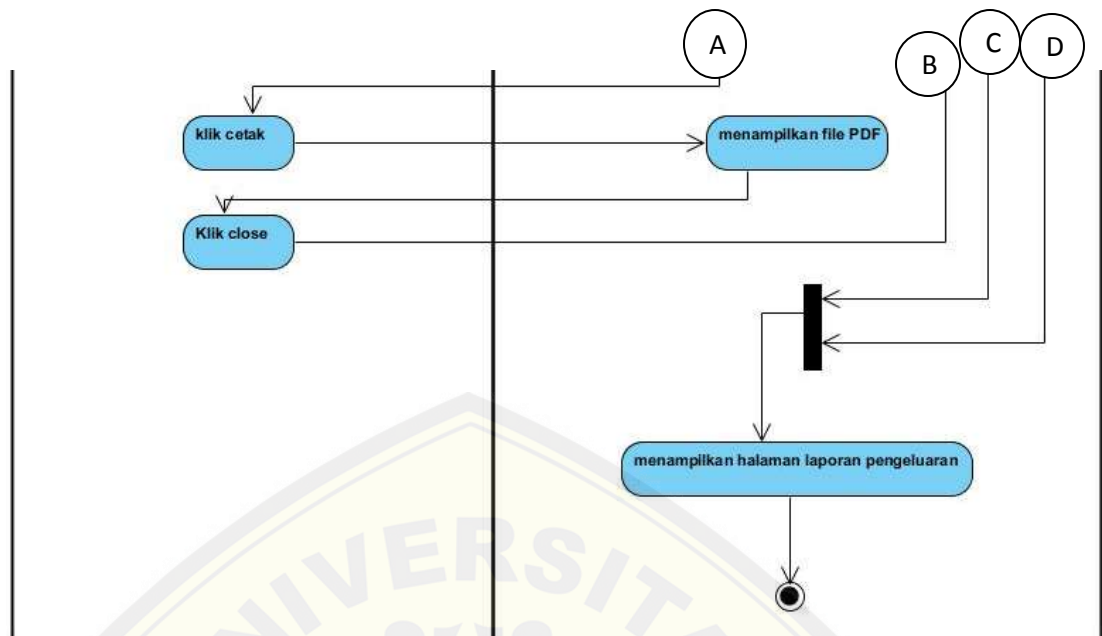
**4.2.4.4 Activity Diagram Menampilkan Laporan Pengeluaran**

Setelah menggambarkan alur kerja pada *usecase scenario*, maka dilanjutkan dengan menggambarkan kedalam *activity diagram*. *Activity Diagram* pada fitur Pengeluaran Lain merupakan penggambaran dari alur kerja pada *usecase scenario*. Berikut penggambaran *activity diagram* menampilkan laporan pengeluaran dapat dilihat pada tabel 4.12.

Tabel 4.12 Activity Diagram “Menampilkan Laporan Pengeluaran”

(Sumber : Hasil Analisis, 2015).



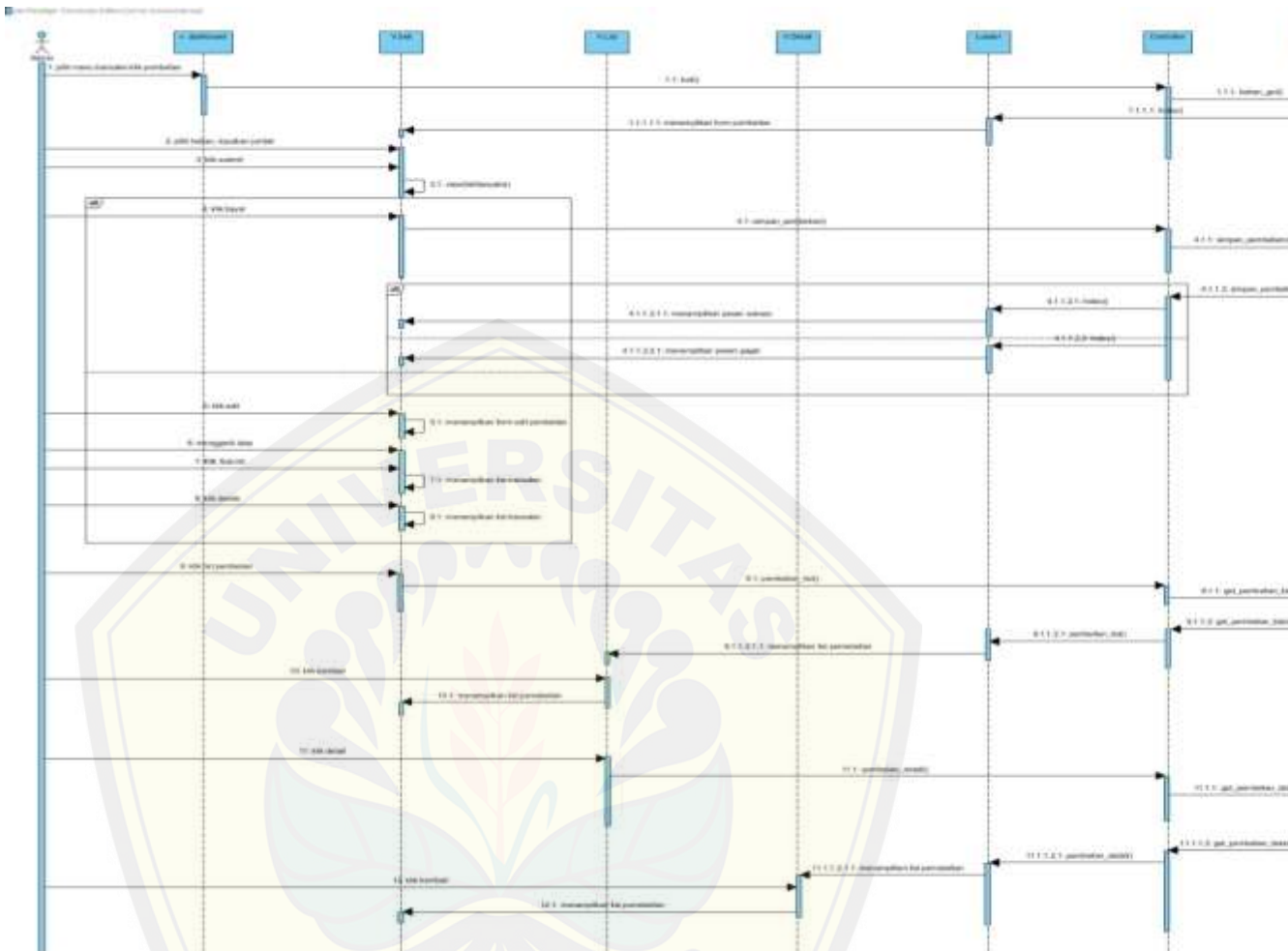


#### 4.2.5 Sequence Diagram

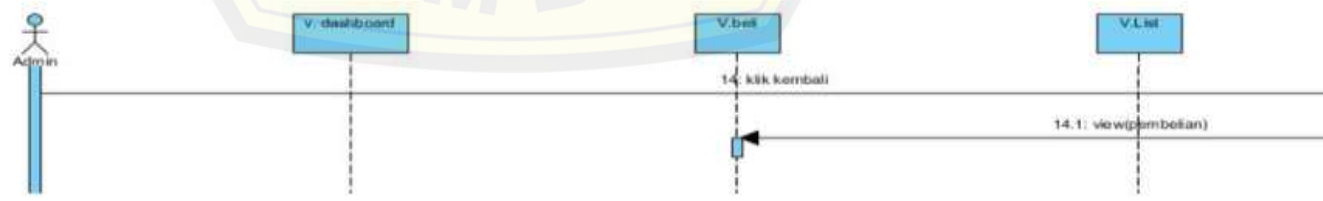
Setelah melakukan alur *usecase scenario* dan *activity diagram* maka selanjutnya merancang *sequence diagram*. *Sequence diagram* itu akan berhubungan langsung dengan implementasi coding. Menggambarkan suatu hubungan dan interaksi yang terjadi antara *view*, *controller* dan *model*.

##### 4.2.5.1 Sequence Diagram Menambah Data Pembelian

Setelah menggambarkan alur kerja pada *usecase scenario* dan *activity diagram* maka dilanjutkan dengan menggambarkan kedalam *sequence diagram*. Berikut penggambaran *sequence diagram* Menambah Data Pembelian dapat dilihat pada gambar 4.3.



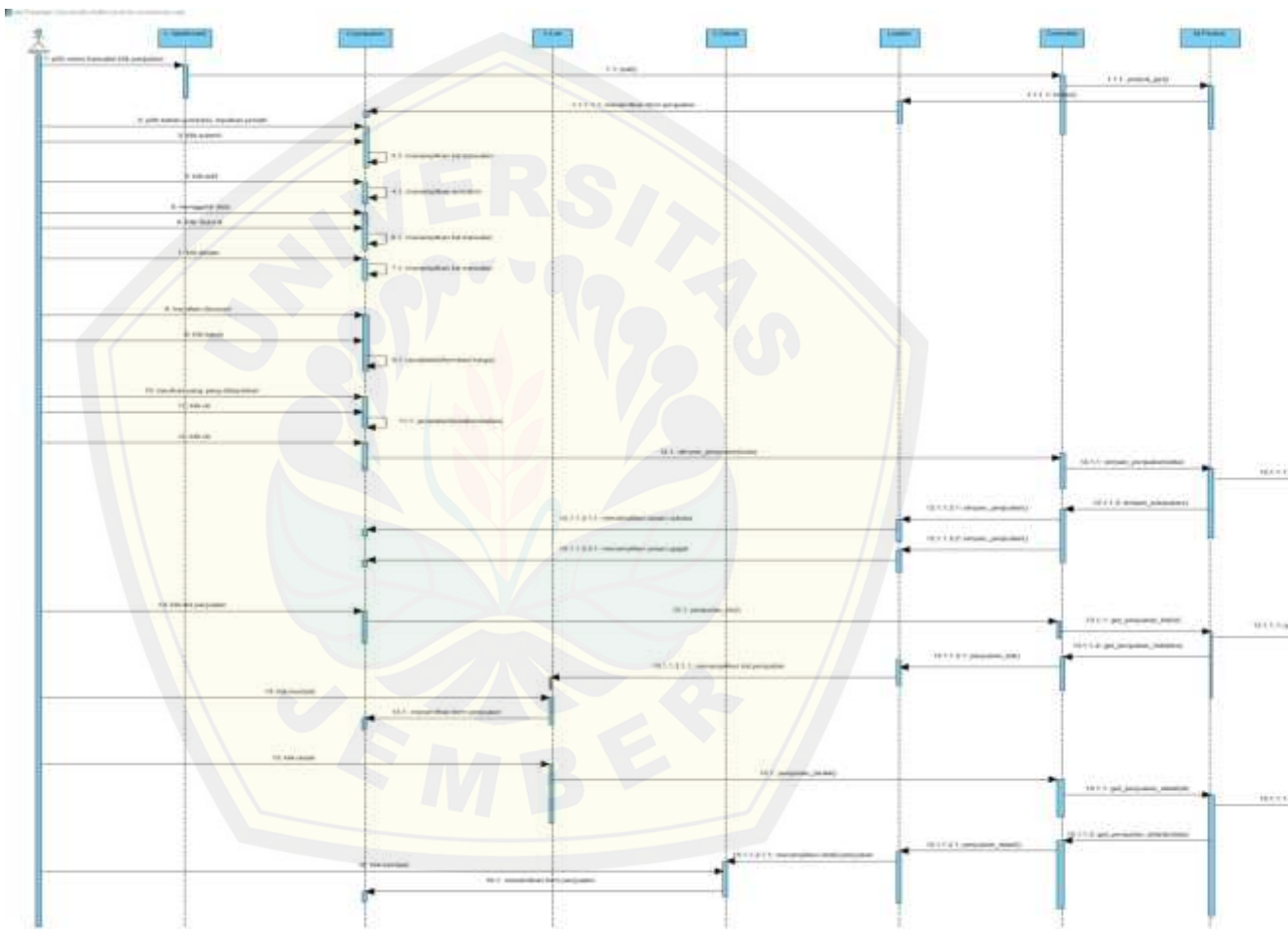
Gambar 4.3 Sequence Diagram Menambah Data Pembelian  
 (Sumber : Hasil Analisis, 2015).



Gambar 4.4 Sequence Diagram Menambah Data Pembelian  
 (Sumber : Hasil Analisis, 2015).

**4.2.5.2 Sequence Diagram Menambah Data Penjualan**

Setelah menggambarkan alur kerja pada *usecase scenario* dan *activity diagram* maka dilanjutkan dengan menggambarkan kedalam *sequence diagram*. Berikut penggambaran *sequence diagram* Menambah Data Penjualan dapat dilihat pada gambar 4.5.



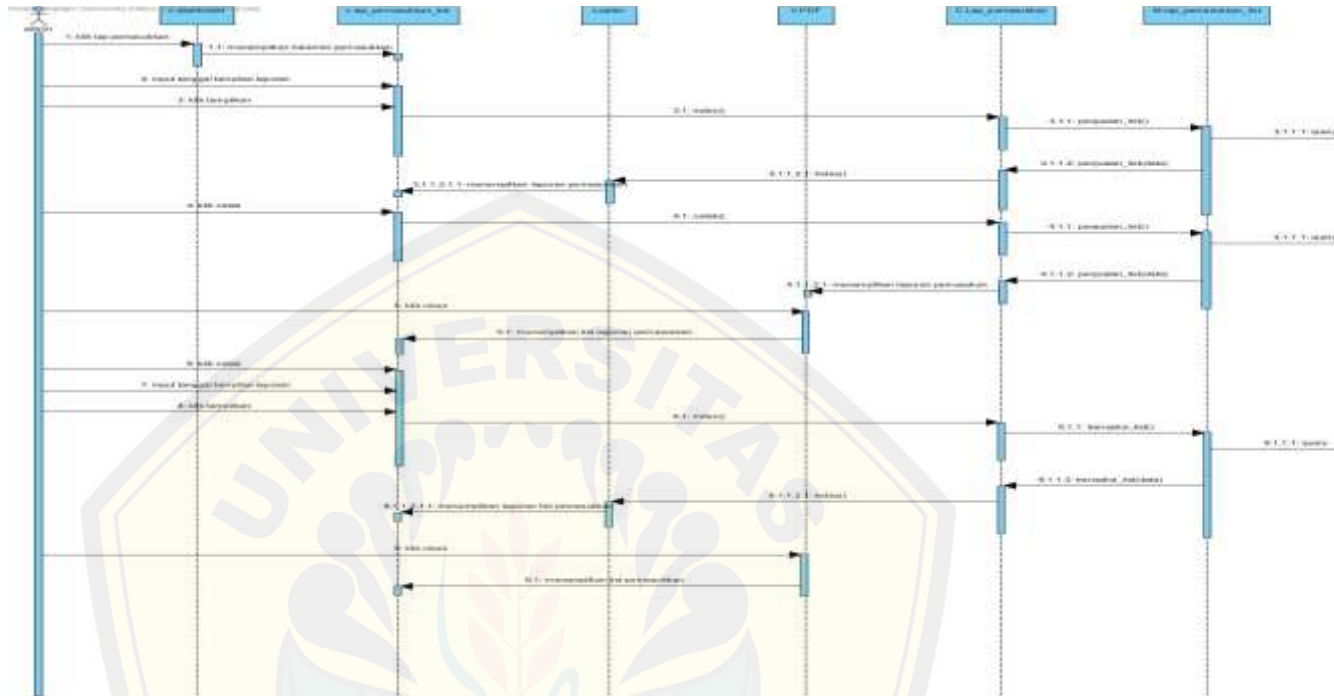
Gambar 4.5 Sequence Diagram Menambah Data Penjualan

(Sumber : Hasil Analisis, 2015).

**4.2.5.3 Sequence Diagram Menampilkan Laporan Pemasukan**

Setelah menggambarkan alur kerja pada *usecase scenario* dan *activity diagram* maka dilanjutkan dengan menggambarkan kedalam *sequence diagram*.

Berikut penggambaran *sequence diagram* Menampilkan Laporan Pemasukan dapat dilihat pada gambar 4.6.



Gambar 4.6 *Sequence Diagram* Menampilkan Laporan Pemasukan

(Sumber : Hasil Analisis, 2015).

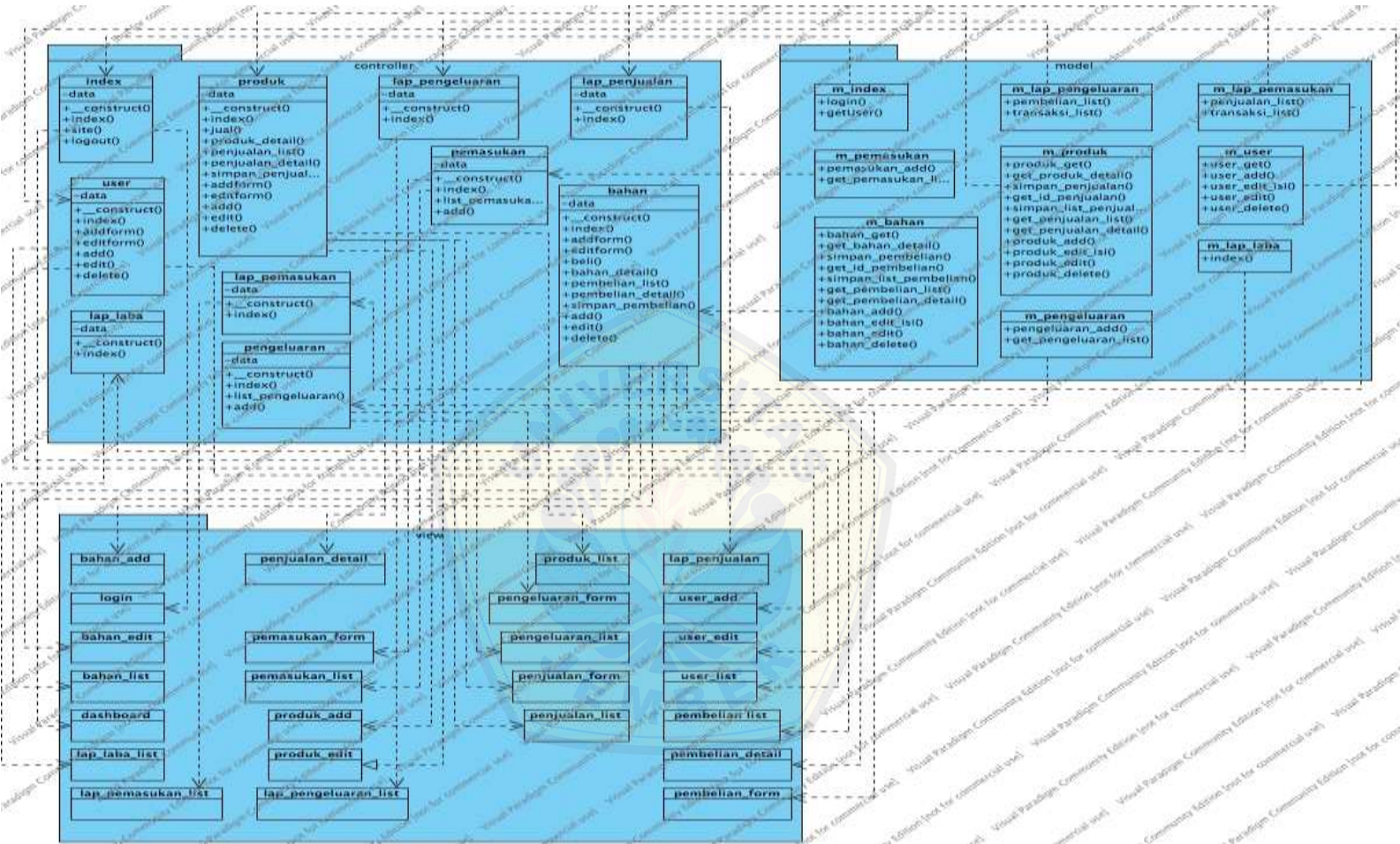
#### 4.2.5.4 *Sequence Diagram* Menampilkan Laporan Pengeluaran

Setelah menggambar alur kerja pada *usecase scenario* dan *activity diagram* maka dilanjutkan dengan menggambar kedalam *sequence diagram*. Berikut penggambaran *sequence diagram* Menampilkan Laporan Pengeluaran dapat dilihat pada gambar 4.7.







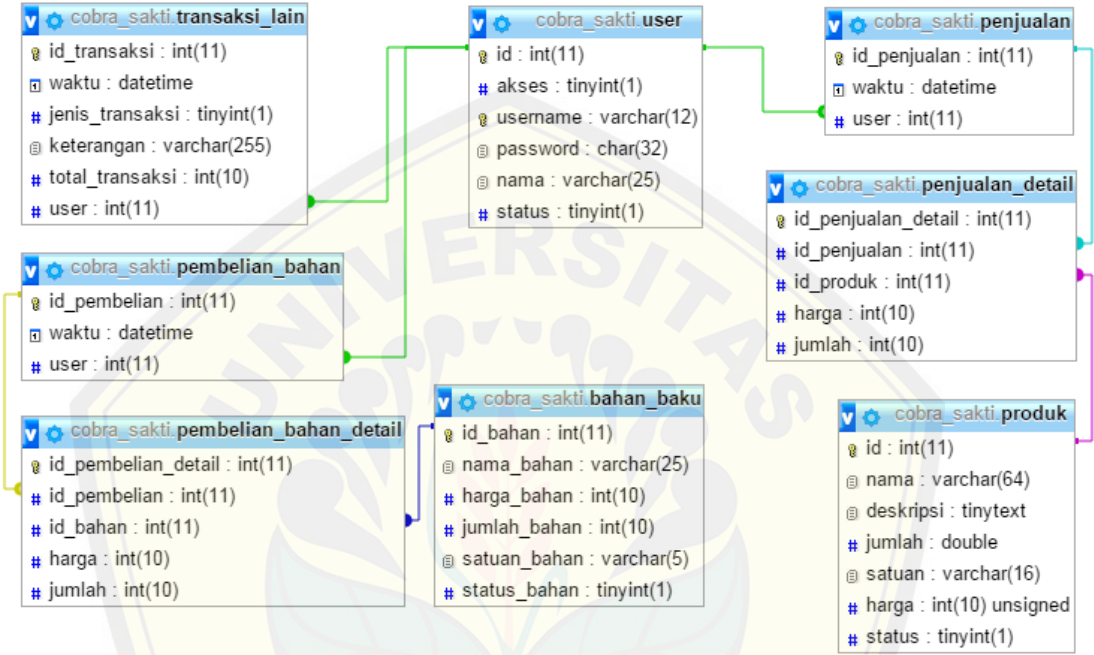


Gambar 4.8 Class Diagram Aplikasi Akuntansi

(Sumber : Hasil Analisis, 2015).

4.2.7 Entity Relational Diagram (ERD)

Menggambarkan hubungan antar entitas yang ada dalam sistem informasi pencarian hotel. Penggambaran entty relational diagram sistem informasi pencarian hotel dapat dilihat pada Gambar 4.9.



Gambar 4.9 Entity Relational Diagram Aplikasi Akuntansi

(Sumber : Hasil Analisis, 2015).

## BAB 5. Hasil dan Pembahasan

Bab ini menjelaskan pembahasan dari hasil penelitian pada bab sebelumnya. Bab Pembahasan menguraikan tentang pengkodean sistem, pengujian sistem dan Implementasi.

### 5.1 Pengkodean Sistem

Tahap pengkodean sistem ini menerjemahkan perancangan sistem yang sudah dibuat kedalam bahasa pemrograman dengan menggunakan bahasa pemrograman PHP. Aplikasi akuntansi UD Cobra Sakti ini dibuat dengan menggunakan metode pengembangan waterfall seperti yang sudah dijelaskan pada bab-bab sebelumnya.

Aplikasi akuntansi UD Cobra Sakti ini dalam pengkodean sistemnya terbagi menjadi tiga pattern yaitu view, controller dan model. Hasil dari pengkodean sistem ini hanya menjelaskan kode-kode program pada bagian inti dari setiap fitur yaitu pada bagian controller.

#### 5.1.1 Menambah Data Pembelian

Controller pada fungsi Memanajemen Data Pembelian berisi tentang *statement* untuk menambahkan halaman pembelian. Kode program menambahpembelian dapat dilihat pada tabel 5.1.

```
57 |  
58 |  
59 | function beli() {  
60 |     $this->data['kind'] = 'table_form';  
61 |     $this->data['page'] = 'pembelian';  
62 |     $this->data['page_title'] = 'Pembelian';  
63 |     $this->data['page_description'] = 'Halaman pembelian bahan baku';  
64 |     $this->data['breadorumb'] = array('', 'bahan/beli.Pembelian', 'Transaksi Pembelian');  
65 |  
66 |     $isi['box_title'] = 'Transaksi Pembelian';  
67 |     $isi['combo_bahan'] = $this->m_bahan->bahan_get();  
68 |     $this->data['content'] = $this->load->view('pembelian_form', $isi, TRUE);  
69 |     $this->load->view('template', $this->data);  
70 | }
```

Gambar 5.1 Kode *Controller* Menambah Data Pembelian

(Sumber : Hasil Analisis, 2015).

Model pada fungsi Menambah Data Pembelian berisi tentang *statement* untuk mengambil list bahan. Kode program menambah data pembelian dapat dilihat pada tabel 5.2.

```

7
8 function bahan_get() {
9     $query = $this->db->query
10         (
11             SELECT
12             *
13             FROM
14             bahan_baku
15             WHERE
16             status_bahan = 1
17             ORDER BY
18             id_bahan asc
19         );
20
21     return $query->result_array();
22 }
23

```

Gambar 5.2 Kode Model Menambah Data Pembelian

(Sumber : Hasil Analisis, 2015).

*Controller* pada fungsi Menambah Data Pembelian digunakan untuk menyimpan pembelian bahan. Kode program menambah data pembelian dapat dilihat pada tabel 5.3.

```

300
301 function simpan_pembelian() {
302     if ($this->bahan->simpan_pembelian()) {
303         $this->session->set_userdata('pesan_sistem', 'Selamat! Pembelian, SUKSES!');
304         $this->session->set_userdata('type_pesan', 'Sukses');
305         redirect('bahan/beli');
306     } else {
307         $this->session->set_userdata('pesan_sistem', 'Maaf! Pembelian, GAGAL!<br />Silahkan periksa dan coba kembali!');
308         $this->session->set_userdata('type_pesan', 'Gagal');
309         redirect('bahan/beli');
310     }
311 }
312

```

Gambar 5.3 Kode *Controller* Menambah Data Pembelian

(Sumber : Hasil Analisis, 2015).

Model pada fungsi Menambah Data Pembelian digunakan untuk menyimpan pembelian bahan. Kode program menambah data pembelian dapat dilihat pada tabel 5.4.



```
37
38 function simpan_pembelian() {
39     $query = $this->db->query
40         (
41             INSERT INTO pembelian_bahan
42             (
43                 id_pembelian
44                 , waktu
45                 , user
46             )
47             VALUES
48             (
49                 ' . "" . "" . '
50                 , NOW()
51                 , ' . $this->session->userdata('id_user') . '
52             )
53         ');
54
55     if ($query) {
56         if ($this->simpan_list_pembelian()) {
57             return TRUE;
58         } else {
59             return FALSE;
60         }
61     } else {
62         return FALSE;
63     }
64 }
65
```

Gambar 5.4 Kode Model Menambah Data Pembelian

(Sumber : Hasil Analisis, 2015).

Model pada fungsi Menambah Data Pembelian digunakan untuk mengambil id bahan yang dibeli. Kode program menambah data pembelian dapat dilihat pada tabel 5.5.

```
65
66 function get_id_pembelian() {
67     $query = $this->db->query
68         (
69             SELECT
70                 id_pembelian
71             FROM
72                 pembelian_bahan
73             ORDER BY
74                 id_pembelian desc
75             LIMIT 1
76         ');
77
78     return $query->row()->id_pembelian;
79 }
80
```

Gambar 5.5 Kode Model Menambah Data Pembelian

(Sumber : Hasil Analisis, 2015).

Model pada fungsi Menambah Data Pembelian digunakan untuk menyimpan list pembelian bahan. Kode program menambah data pembelian dapat dilihat pada tabel 5.6.



```
function simpan_list_pembelian() {
82     $this->db->trans_start();
83     $id_pembelian = $this->get_id_pembelian();
84
85     for ($i = 0; $i < count($_POST['id_bahan']); $i++) {
86         $id = $_POST['id_bahan'][$i];
87         $jumlah = $_POST['jumlah'][$i];
88         $harga = $_POST['harga'][$i];
89         //         $record = $this->cek_record_barang($id);
90
91         $this->db->query
92             (
93                 INSERT INTO pembelian_bahan_detail
94                 (
95                     id_pembelian_detail
96                     , id_pembelian
97                     , id_bahan
98                     , harga
99                     , jumlah
100                )
101                VALUES
102                (
103                    ' . "'" . "'" . "'" . '
104                    , ' . $id_pembelian . '
105                    , ' . $id . '
106                    , ' . $harga . '
107                    , ' . $jumlah . '
108                );
109            );
110         $this->db->query
111             (
112                 UPDATE bahan_baku SET
113                 jumlah_bahan = (jumlah_bahan + ' . $jumlah . ')
114                 WHERE
115                 id_bahan = ' . $id . ';
116             );
117         //
118     }
119 }
120
121 if ($this->db->trans_status()) {
122     $this->db->trans_commit();
123     return TRUE;
124 } else {
125     $this->db->trans_rollback();
126     return FALSE;
127 }
128 }
129 }
130 }
```

Gambar 5.6 Kode Model Menambah Data Pembelian

(Sumber : Hasil Analisis, 2015).



### 5.1.2 Menambah Data Penjualan

Controller pada fungsi Menambah Data Penjualan berisi tentang *statement* untuk menampilkan halaman penjualan. Kode program menambah data penjualan dapat dilihat pada tabel 5.7.

```
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
function jual() {  
    $this->data['kind'] = 'table_form';  
    $this->data['page'] = 'penjualan';  
    $this->data['page_title'] = 'Penjualan';  
    $this->data['page_description'] = 'Halaman penjualan produk';  
    $this->data['breadcrumb'] = array('', 'produk.Produk', 'Transaksi Penjualan');  
  
    $sisi['box_title'] = 'Transaksi Penjualan';  
    $sisi['combo_produk'] = $this->m_produk->produk_get();  
    $this->data['content'] = $this->load->view('penjualan_form', $sisi, TRUE);  
    $this->load->view('template', $this->data);  
}
```

Gambar 5.7 Kode *Controller* Menambah Data Penjualan

(Sumber : Hasil Analisis, 2015).

Model pada fungsi Menambah Data Penjualan digunakan untuk mengambil list produk. Kode program menambah data pembelian dapat dilihat pada tabel 5.8.

```
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
function produk_get() {  
    $query = $this->db->query  
        ('  
            SELECT  
                *  
            FROM  
                produk  
            WHERE  
                status = 1  
            ORDER BY  
                id asc  
        ');  
    return $query->result_array();  
}
```

Gambar 5.8 Kode Model Menambah Data Penjualan

(Sumber : Hasil Analisis, 2015).



```

66 function get_id_penjualan() {
67     $query = $this->db->query
68         (
69             SELECT
70                 id_penjualan
71             FROM
72                 penjualan
73             ORDER BY
74                 id_penjualan desc
75             LIMIT 1
76         );
77
78     return $query->row()->id_penjualan;
79 }
80

```

Gambar 5.11 Kode Model Menambah Data Penjualan

(Sumber : Hasil Analisis, 2015).

Model pada fungsi Menambah Data Penjualan digunakan untuk menyimpan list penjualan. Kode program menambah data pembelian dapat dilihat pada tabel 5.12.

```

81 function simpan_list_penjualan() {
82     $this->db->trans_start();
83     $sid_penjualan = $this->get_id_penjualan();
84
85     for ($i = 0; $i < count($_POST['id_produk']); $i++) {
86         $sid = $_POST['id_produk'][$i];
87         $jumlah = $_POST['jumlah'][$i];
88         $harga = $_POST['harga'][$i];
89         // $record = $this->cek_record_barang($sid);
90
91         $this->db->query
92             (
93                 INSERT INTO penjualan_detail
94                 (
95                     id_penjualan_detail
96                     , id_penjualan
97                     , id_produk
98                     , harga
99                     , jumlah
100                )
101                VALUES
102                (
103                    ' . "" . "" . '
104                    , ' . $sid_penjualan . '
105                    , ' . $sid . '
106                    , ' . $harga . '
107                    , ' . $jumlah . '
108                );
109            );

```



## 5.2.2 Pengujian *White box*

Pengujian *white box* dilakukan dengan cara pembuatan diagram alir dari *listing* program pengujian ini dilakukan. Pengujian ini akan diterapkan pada beberapa fitur yang dinilai dapat mewakili aplikasi akuntansi pada UD Cobra Sakti. Berikut fitur yang telah diuji.

### a) Listing Program Fitur *Login*

*Listing* program fitur *login* dapat dilihat pada gambar 5.13

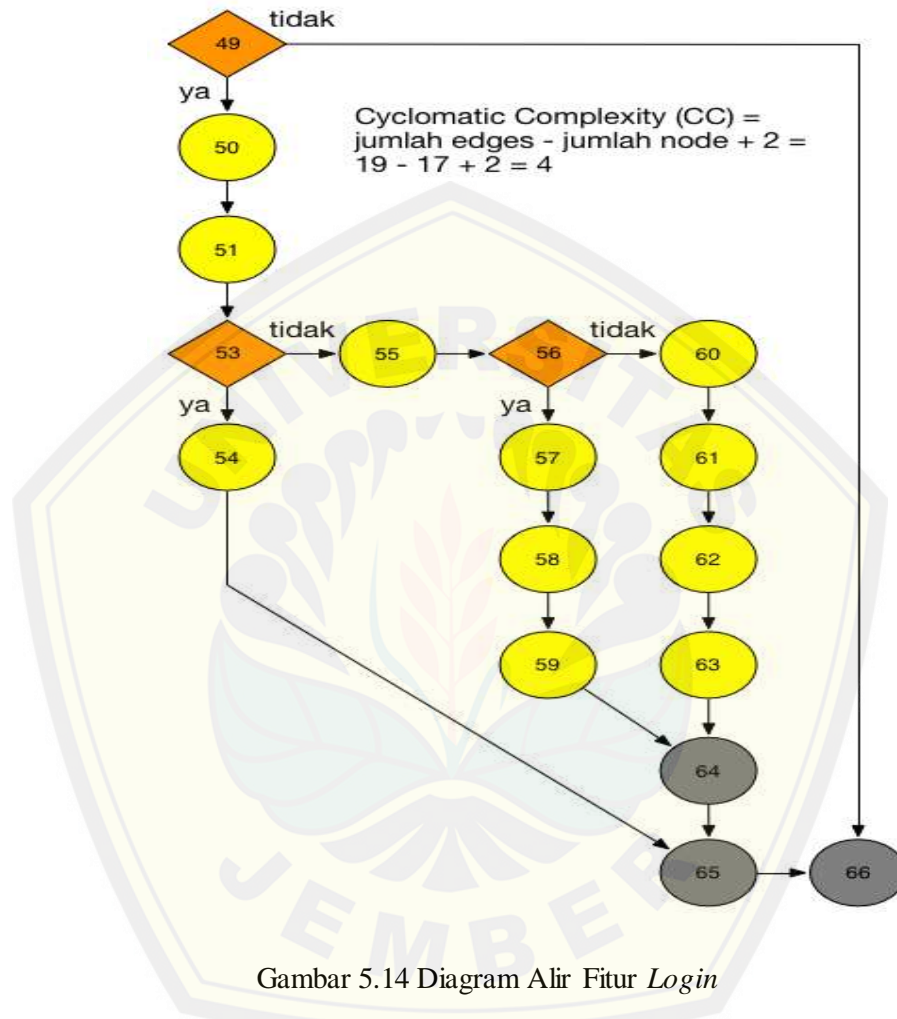
```
48 function site() {
49     if ($POST) {
50         $u = $POST['username']; //mengambil data dr inputan username
51         $p = $POST['password'];
52
53         if (empty($u) || empty($p)) { //cek apakah apakah masih ada yg belum terisi
54             redirect(); // Panggil controller index lg
55         } else { //atau data sudah diisi semua
56             if ($this->index->login($u, $p)) { //kalo datanya ada dan level lebih dari nol
57                 $this->session->set_userdata('pesan_sistem', 'Selamat! Anda berhasil login!');
58                 $this->session->set_userdata('tipe_pesan', 'Sukses');
59                 redirect(); //menampilkan dashboard
60             } else {
61                 $this->session->set_userdata('pesan_sistem', 'Username atau Password SALAH!<br />Silahkan per
62                 $this->session->set_userdata('tipe_pesan', 'Gagal');
63                 redirect(); //menampilkan login
64             }
65         }
66     }
67 }
68 }
```

Gambar 5.13 Listing Program Fitur *Login*

(Sumber : Hgasil Analisis, 2015).

b) Diagram Alir Fitur *Login*

Diagram alir dari fitur *login* bisa dilihat pada gambar 5.14 dibawah ini.



Gambar 5.14 Diagram Alir Fitur *Login*

(Sumber : Hasil Analisis, 2015).

c) Perhitungan *Cyclomatic Complexity* Fitur *Login*

Perhitungan diagram alir pada fitur *login* menggunakan *Cyclomatic Complexity* adalah sebagai berikut:

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

$$V(G) = 19 - 17 + 2$$

$$V(G) = 4$$

d) Pengujian jalur program fitur *Login*

Pengujian jalur jalur program fitur *login* berdasarkan perhitungan *Cyclomatic Complexity* adalah sebagai berikut :

Jalur 1 : 49-50-51-53-54-65-66

Jalur 2 : 49-50-51-53-55-56-57-58-59-64-65-66

Jalur 3 : 49-50-51-53-55-56-60-61-62-63-64-65-66

Jalur 4 : 49-66

e) Test case fitur *Login*

Tabel 5.1 Test Case Fitur Login

(Sumber : Hasil Analisis, 2015).

NO	JALUR	TEST CASE	HASIL	KESIMPULAN
1	49-50-51-53-54-65-66	Terdapat salah satu inputan yang belum diisi	Kembali halaman login	[√] Berhasil [ ] Gagal
2	49-50-51-53-55-56-57-58-59-64-65-66	Semua <i>field</i> sudah diisi dan <i>valid</i>	Halaman <i>dashboard</i>	[√] Berhasil [ ] Gagal
3	49-50-51-53-55-56-60-61-62-63-64-65-66	Semua <i>field</i> sudah diisi dan tidak <i>valid</i>	Kembali halaman <i>login</i>	[√] Berhasil [ ] Gagal
4	49-66	Tidak terdapat inputan	Menampilkan <i>blank pages</i>	[√] Berhasil [ ] Gagal

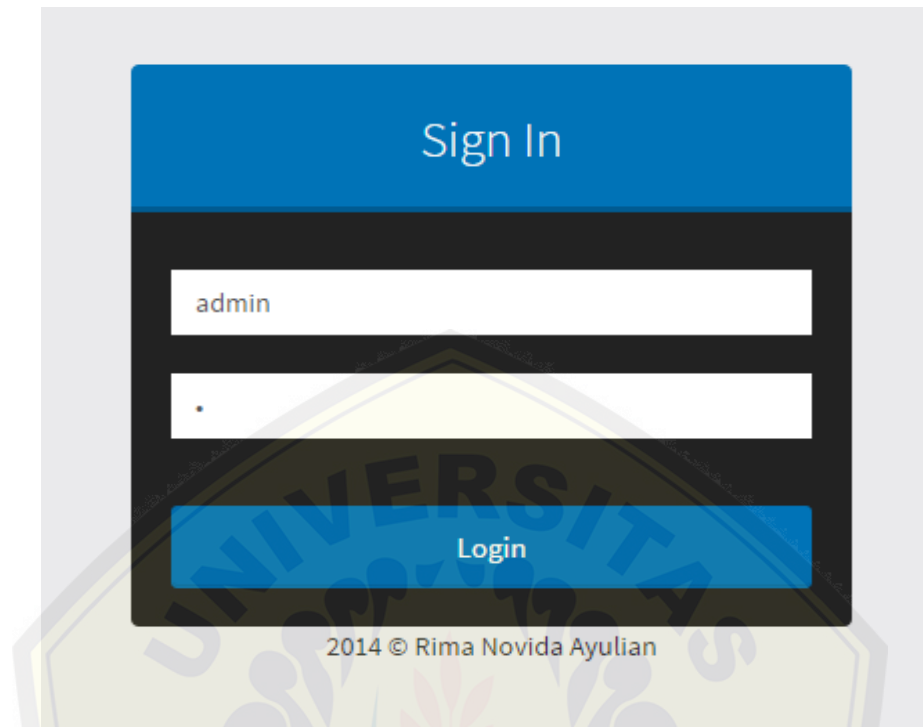


### 5.3 Hasil Implementasi Aplikasi

Pada tahap implementasi ini telah dibangun aplikasi akuntansi pada UD Cobra Sakti dengan melakukan pengkodean program yang dilakukan sesuai dengan diagram-diagram fungsi pada tahap perancangan sistem. Sistem memiliki 3 aktor untuk mengakses sistem melalui desktop, dan 3 aktor tersebut memiliki batasan akses sesuai level aktor tersebut. Pada tahapan ini akan dijelaskan beberapa tampilan program sesuai fitur dan jenis akses aktor, berikut tampilan program menurut fitur yang telah selesai dibangun:

#### 5.3.1 Fitur *Login*

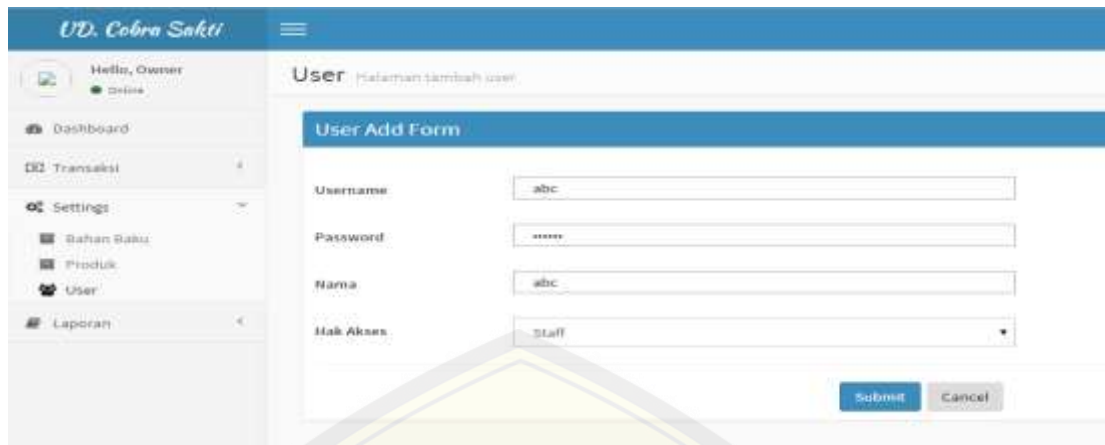
Fitur *login* yang merupakan tampilan awal dari aplikasi akuntansi pada UD Cobra Sakti berfungsi sebagai gerbang utama dari sistem. *Login* juga berfungsi sebagai penentu fitur-fitur apa yang akan didapat oleh aktor. *login* untuk admin dilakukan dengan memasukkan "admin" sebagai username, untuk kasir dapat melakukan *login* dengan memasukkan "kasir" sebagai username, dan untuk staff dapat melakukan *login* dengan memasukkan "staff" sebagai usernamenya. Untuk tampilan *login* seperti pada gambar 5.1 dibawah ini.



Gambar 5.15 Halaman *Login*  
(Sumber: Hasil Analisis, 2015).

### 5.3.2 Fitur Tambah Data User

Fitur tambah data user merupakan fitur untuk menambahkan data user kedalam basis data sistem. Data user ini berisi antara lain; *Username*, *password*, nama dan hak akses. Fitur tambah data user ini hanya bisa diakses oleh admin saja. Untuk tampilan *input* data member seperti gambar 5.16 dibawah ini.



The screenshot shows a web application interface for adding a user. The header includes the logo 'UD. Cobra Sakti' and a user profile 'Hello, Owner'. A sidebar menu on the left lists 'Dashboard', 'Transaksi', 'Settings', 'Bahan Baku', 'Produk', 'User', and 'Laporan'. The main content area is titled 'User' and 'User Add Form'. The form contains four fields: 'Username' (abc), 'Password' (masked with asterisks), 'Nama' (abc), and 'Hak Akses' (Staff). 'Submit' and 'Cancel' buttons are at the bottom right.

Gambar 5.16 Fitur Tambah Data *User*

(Sumber: Hasil Analisis, 2015).

Apabila admin tidak menginputkan data atau kurang satu data yang kosong, maka sistem akan memberikan atau menampilkan peringatan data kosong seperti gambar 5.17 dibawah ini.

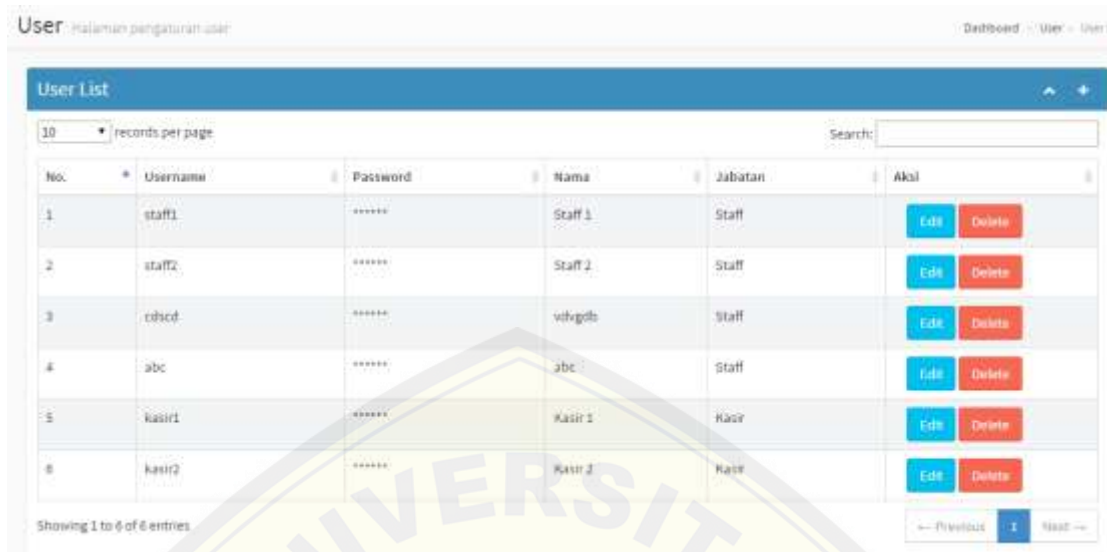


This screenshot shows the same 'User Add Form' as in Gambar 5.16, but with a validation error. The 'Nama' field is empty and highlighted in yellow. A yellow warning icon and the text 'Please fill out this field.' are displayed next to the field. The other fields (Username, Password, Hak Akses) are filled with 'abc', masked characters, and 'Staff' respectively. The 'Submit' and 'Cancel' buttons are still present at the bottom right.

Gambar 5.17 Peringatan Kolom Kosong Tambah Data *User*

(Sumber: Hasil Analisis, 2015).

Saat admin telah menginputkan seluruh data dan data telah valid tanpa ada kesalahan maka sistem akan menyimpan di basis data dan menampilkan pesan sukses seperti gambar 5.18 dibawah ini.



The screenshot displays a web interface for user management. At the top, it says 'User' and 'Halaman pengaturan user'. Below this is a 'User List' section with a search bar and a dropdown for 'records per page' set to 10. The table below has the following data:

No.	Username	Password	Nama	Jabatan	Aksi
1	staff1	*****	Staff 1	Staff	Edit Delete
2	staff2	*****	Staff 2	Staff	Edit Delete
3	rbhcd	*****	vbygdts	Staff	Edit Delete
4	abc	*****	abc	Staff	Edit Delete
5	kasir1	*****	Kasir 1	Kasir	Edit Delete
6	kasir2	*****	Kasir 2	Kasir	Edit Delete

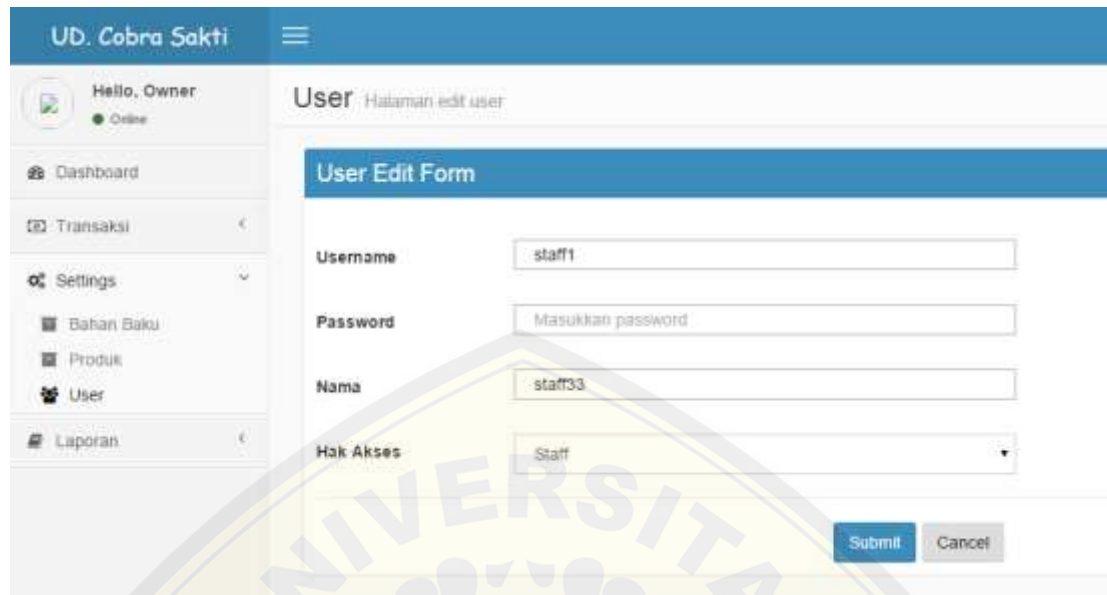
At the bottom, it shows 'Showing 1 to 6 of 6 entries' and pagination controls for 'Previous', '1', and 'Next'.

Gambar 5.18 Tambah Data *User* Sukses

(Sumber: Hasil Analisis, 2015).

### 5.3.3 Fitur *Edit Data User*

Fitur *edit* data member berfungsi untuk merubah data *user* yang telah ada di basis data. *Edit* data user hanya bisa diakses oleh admin, dan dilakukan dengan cara memilih tombol *edit* yang ada pada data member. Setelah memilih tombol *edit* maka sistem menampilkan halaman *update* member seperti gambar 5.19 dibawah ini.



The screenshot shows a web application interface for editing a user. The header is blue with the text 'UD. Cobra Sakti' and a menu icon. Below the header, there is a sidebar with navigation options: Dashboard, Transaksi, Settings, Bahan Baku, Produk, User, and Laporan. The main content area is titled 'User' and 'Halaman edit user'. The 'User Edit Form' contains the following fields: Username (staff1), Password (Masukkan password), Nama (staff33), and Hak Akses (Staff). There are 'Submit' and 'Cancel' buttons at the bottom right of the form.

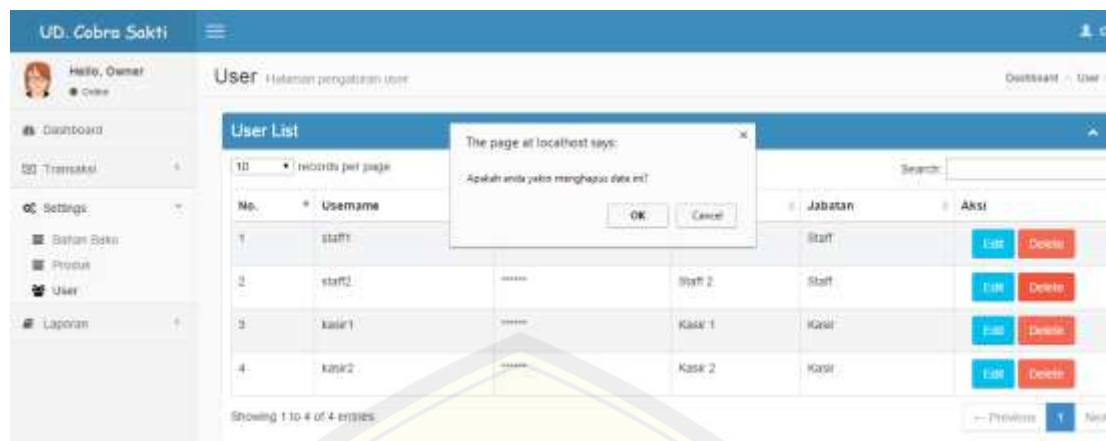
Gambar 5.19 Halaman *Edit User*

Sumber: Hasil Analisis, 2015).

Jika admin telah selesai merubah data member maka sistem akan menampilkan pada halaman data member sama seperti gambar 5.18 sebelumnya.

#### 5.3.4 Fitur Hapus Data User

Fitur hapus data user ini berfungsi untuk menghapus data user yang ada di basis data. Fitur ini hanya bisa diakses oleh admin, dan dilakukan dengan cara memilih tombol hapus pada data member. Setelah memilih tombol hapus maka sistem akan menampilkan pesan seperti gambar 5.20 dibawah ini.



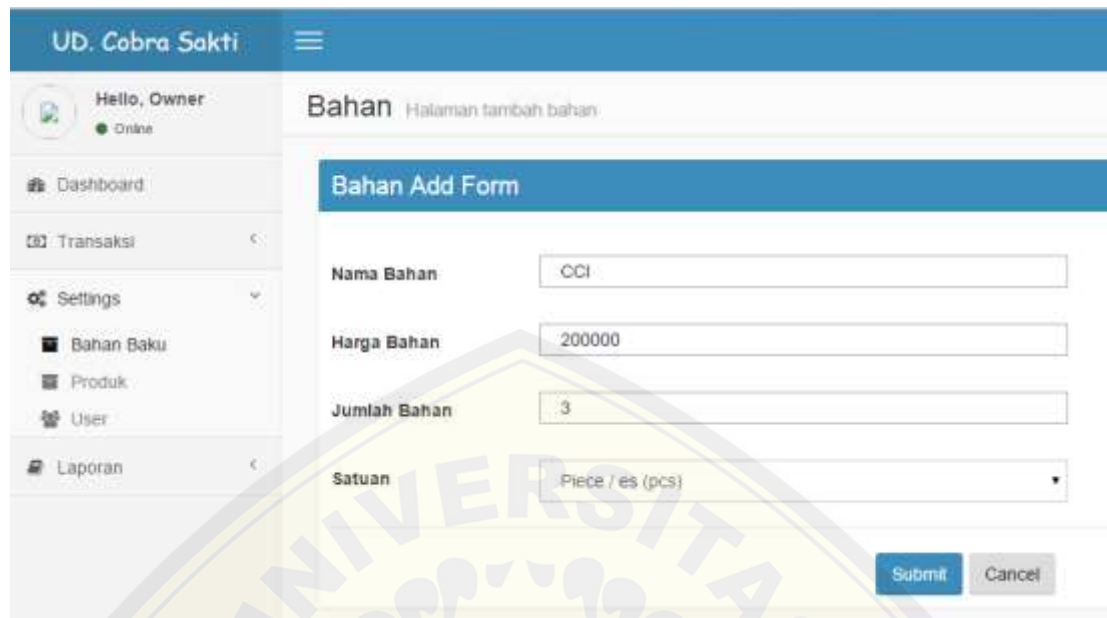
Gambar 5.20 Peringatan Hapus Data *User*

(Sumber: Hasil Analisis, 2015).

Setelah data berhasil dihapus maka sistem akan menampilkan pesan sukses pada halaman data user seperti gambar 5.18 sebelumnya

### 5.3.5 Fitur *Add Bahan Baku*

Fitur *add* bahan baku merupakan fitur untuk menambahkan bahan baku kedalam basis data sistem. Bahan baku ini berisi antara lain; nama bahan, harga bahan, jumlah bahan dan satuan. Fitur *add* bahan baku ini bisa diakses oleh admin dan staff. Untuk tampilan *add* bahan baku seperti gambar 5.21 dibawah ini.



UD. Cobra Sakti

Hello, Owner  
Online

Bahan Halaman tambah bahan

Bahan Add Form

Nama Bahan: CCI

Harga Bahan: 200000

Jumlah Bahan: 3

Satuan: Piece / es (pcs)

Submit Cancel

Gambar 5.21 Add Bahan Baku

(Sumber: Hasil Analisis, 2015).

Apabila admin dan staff tidak menginputkan data atau kurang satu data yang kosong, maka sistem akan memberikan atau menampilkan peringatan data kosong seperti gambar 5.22 dibawah ini.



UD. Cobra Sakti

Hello, Owner  
Offline

Bahan Halaman tambah bahan

Bahan Add Form

Nama Bahan: CCI

Harga Bahan: 200000

Jumlah Bahan: Masukkan jumlah bahan

Satuan: Piece / es (pcs) Please fill out this field.

Submit Cancel

Gambar 5.22 Peringatan Kolom Kosong Add Bahan Baku

(Sumber: Hasil Analisis, 2015).



Saat admin dan staff telah menginputkan seluruh data dan data telah *valid* tanpa ada kesalahan maka sistem akan menyimpan di basis data dan menampilkan *list* bahan baku seperti gambar 5.23 dibawah ini.



No.	Nama	Harga	Jumlah	Satuan	Aksi
1	Lem	7500	07	kg	<a href="#">Edit</a> <a href="#">Delete</a>
2	roll Cobra	50000	115	br	<a href="#">Edit</a> <a href="#">Delete</a>
3	Paku	100	1004	pcs	<a href="#">Edit</a> <a href="#">Delete</a>
4	ring D	20000	7	pcs	<a href="#">Edit</a> <a href="#">Delete</a>
5	OCI	200000	0	pcs	<a href="#">Edit</a> <a href="#">Delete</a>
6	OCI	200000	0	pcs	<a href="#">Edit</a> <a href="#">Delete</a>

Gambar 5.23 Add Bahan Baku Sukses

(Sumber: Hasil Analisis, 2015).

### 5.3.6 Fitur *Edit* Bahan Baku

Fitur *edit* bahan baku berfungsi untuk merubah bahan baku yang telah ada di basis data. *Edit* bahan baku bisa diakses oleh admin dan staff, dan dilakukan dengan cara memilih tombol *edit* yang ada pada bahan baku. Setelah memilih tombol *edit* maka sistem menampilkan halaman *update* bahan baku seperti gambar 5.24 dibawah ini.



The image shows a web application interface for editing material data. The header includes the company name 'UD. Cobra Sakti' and a user profile 'Hello, Owner' with an 'Online' status. A sidebar on the left contains navigation links: Dashboard, Transaksi, Settings, Bahan Baku, Produk, User, and Laporan. The main content area is titled 'Bahan' and 'Halaman edit bahan'. The 'Bahan Edit Form' contains the following fields: 'Nama Bahan' with the value 'CCI', 'Harga Bahan' with '205000', 'Jumlah Bahan' with '3', and 'Satuan' with a dropdown menu showing 'Piece / es (pcs)'. At the bottom right of the form are 'Submit' and 'Cancel' buttons.

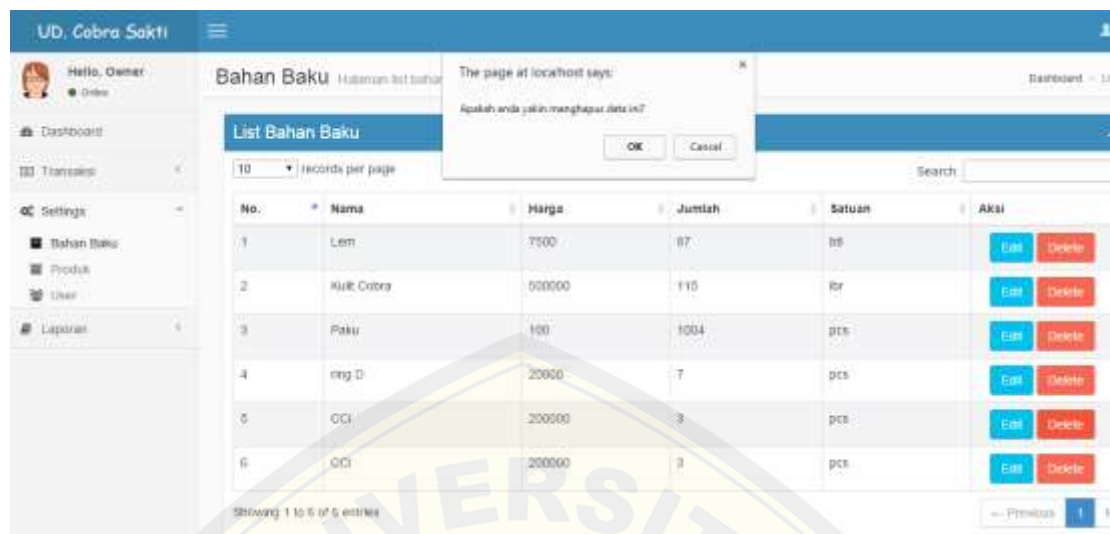
Gambar 5.24 Halaman *Edit* Bahan Baku

(Sumber: Hasil Analisis, 2015).

Jika admin telah selesai merubah data member maka sistem akan menampilkan pada halaman data member sama seperti gambar 5.23 sebelumnya.

### 5.3.7 Fitur Hapus Bahan Baku

Fitur hapus bahan baku ini berfungsi untuk menghapus bahan baku yang ada di basis data. Fitur ini bisa diakses oleh admin dan staff, dan dilakukan dengan cara memilih tombol hapus pada data member. Setelah memilih tombol hapus maka sistem akan menampilkan pesan seperti gambar 5.25 dibawah ini.



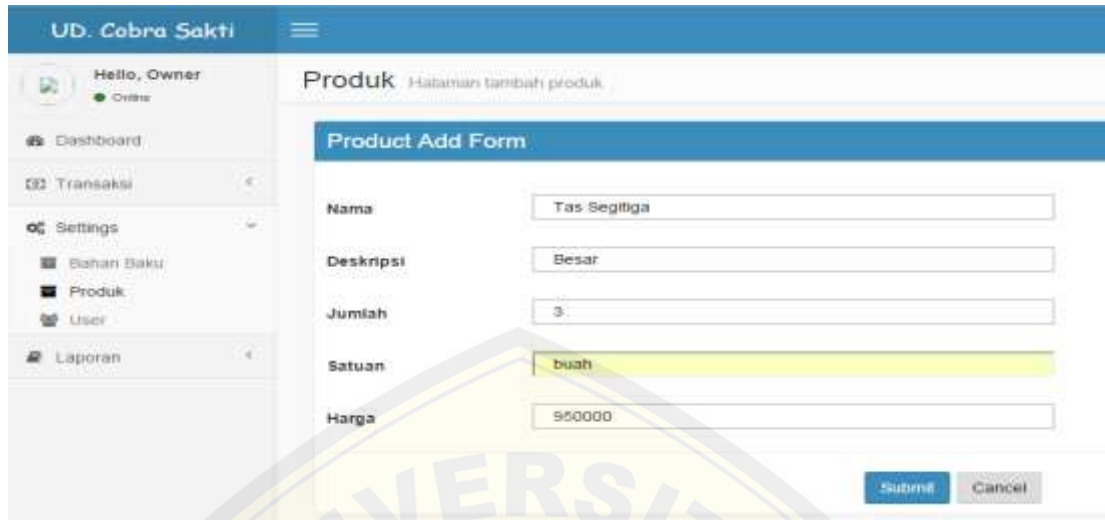
Gambar 5.25 Peringatan Hapus Bahan Baku

(Sumber: Hasil Analisis, 2015).

Setelah data berhasil dihapus maka sistem akan menampilkan pesan sukses pada halaman data *user* seperti gambar 5.23 sebelumnya

### 5.3.8 Fitur Add Produk

Fitur *add* produk merupakan fitur untuk menambahkan bahan baku kedalam basis data sistem. Bahan baku ini berisi antara lain; nama, deskripsi, jumlah, satuan dan harga. Fitur *add* produk ini bisa diakses oleh admin dan staff. Untuk tampilan *add* produk seperti gambar 5.26 dibawah ini.



UD. Cobra Sakti

Hello, Owner  
Online

Dashboard

Transaksi

Settings

Bahan Baku

Produk

User

Laporan

Produk Halaman tambah produk

Product Add Form

Nama: Tas Segitiga

Deskripsi: Besar

Jumlah: 3

Satuan: buah

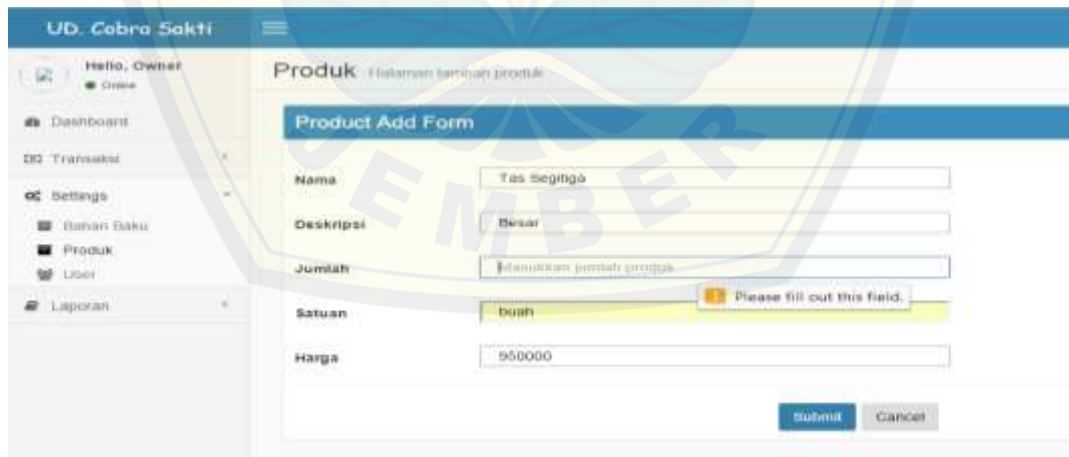
Harga: 950000

Submit Cancel

Gambar 5.26 Add Produk

(Sumber: Hasil Analisis, 2015).

Apabila admin dan staff tidak menginputkan data atau kurang satu data yang kosong, maka sistem akan memberikan atau menampilkan peringatan data kosong seperti gambar 5.27 dibawah ini.



UD. Cobra Sakti

Hello, Owner  
Online

Dashboard

Transaksi

Settings

Bahan Baku

Produk

User

Laporan

Produk Halaman tambah produk

Product Add Form

Nama: Tas Segitiga

Deskripsi: Besar

Jumlah: Menentukan jumlah produk

Satuan: buah

Harga: 950000

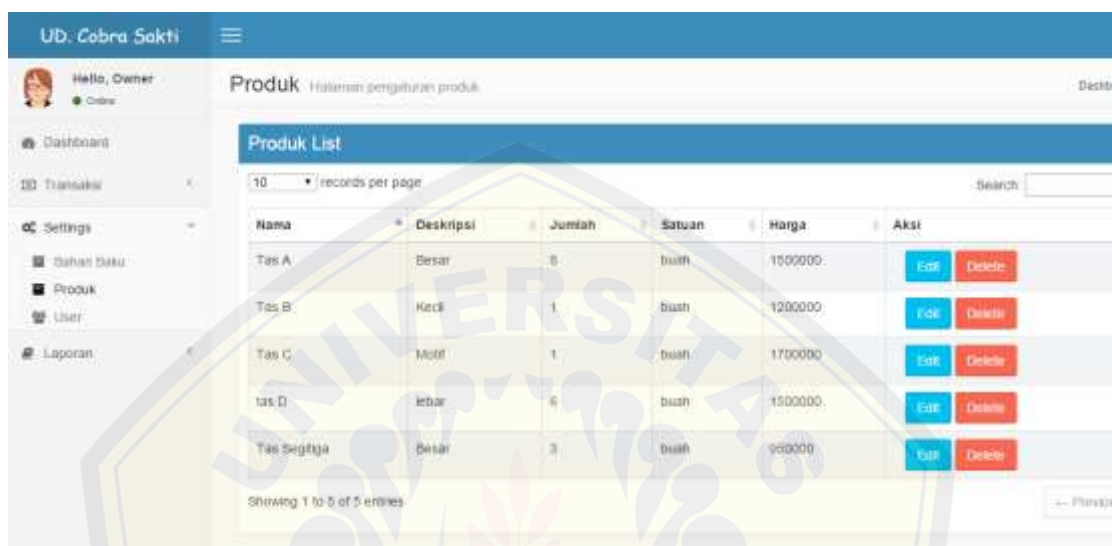
Please fill out this field.

Submit Cancel

Gambar 5.27 Peringatan Kolom Kosong Add Produk

(Sumber: Hasil Analisis, 2015).

Saat admin dan staff telah menginputkan seluruh data dan data telah *valid* tanpa ada kesalahan maka sistem akan menyimpan di basis data dan menampilkan *list* produk seperti gambar 5.28 dibawah ini.

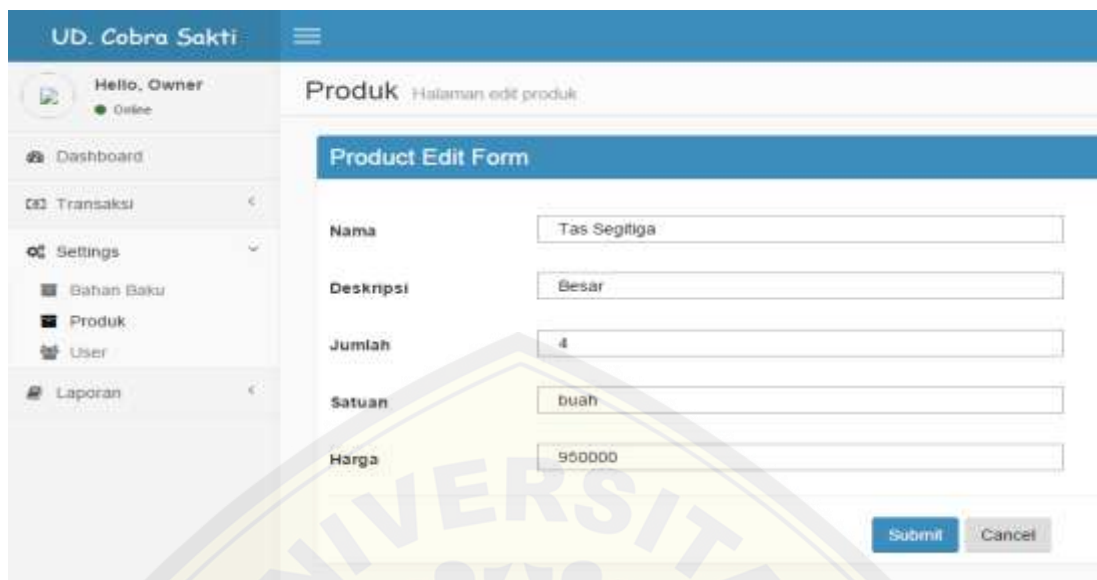


Nama	Deskripsi	Jumlah	Satuan	Harga	Aksi
Tas A	Besarnya	5	buah	1000000	<a href="#">Edit</a> <a href="#">Delete</a>
Tas B	Kecil	1	buah	1200000	<a href="#">Edit</a> <a href="#">Delete</a>
Tas C	Mottif	1	buah	1700000	<a href="#">Edit</a> <a href="#">Delete</a>
tas D	lebar	6	buah	1500000	<a href="#">Edit</a> <a href="#">Delete</a>
Tas Segitiga	Besar	3	buah	900000	<a href="#">Edit</a> <a href="#">Delete</a>

Gambar 5.28 Add Produk Sukses  
(Sumber: Hasil Analisis, 2015).

### 5.3.9 Fitur *Edit* Produk

Fitur *edit* prooduk berfungsi untuk merubah produkyang telah ada di basis data. *Edit* bahan baku bisa diakses oleh admin dan staff, dan dilakukan dengan cara memilih tombol *edit* yang ada pada bahan baku. Setelah memilih tombol *edit* maka sistem menampilkan halaman *update* produk seperti gambar 5.29 dibawah ini.



The screenshot shows a web application interface for editing a product. The header includes the company name 'UD. Cobra Sakti' and a user greeting 'Hello, Owner'. A sidebar menu on the left lists various system functions like Dashboard, Transaksi, Settings, and Laporan. The main content area is titled 'Produk' and 'Halaman edit produk'. Below this is a 'Product Edit Form' with several input fields: 'Nama' (Tas Segitiga), 'Deskripsi' (Besar), 'Jumlah' (4), 'Satuan' (buah), and 'Harga' (950000). At the bottom right of the form are 'Submit' and 'Cancel' buttons.

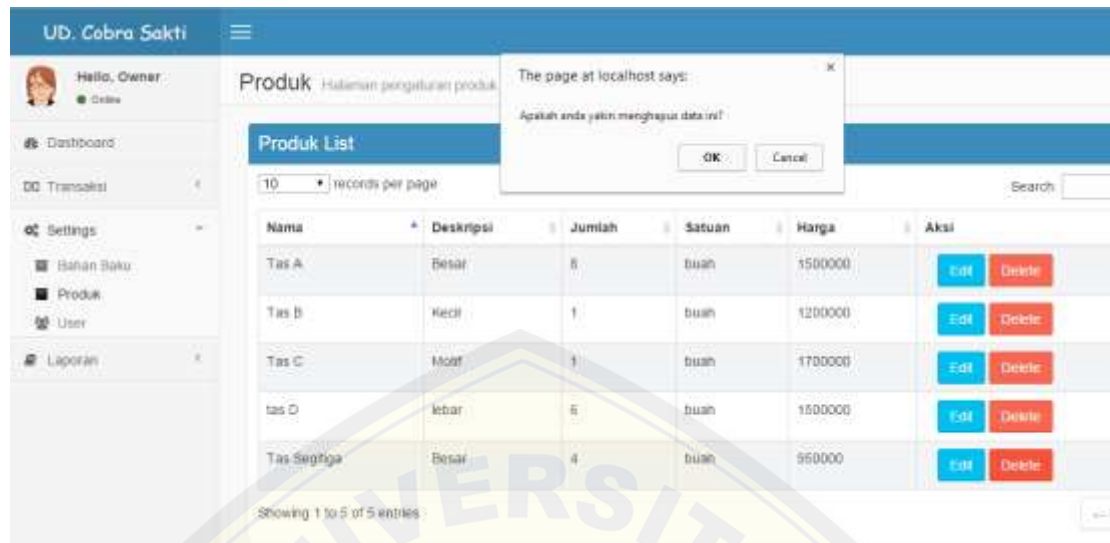
Gambar 5.29 Halaman *Edit* Produk

(Sumber: Hasil Analisis, 2015).

Jika admin dan staff telah selesai merubah data member maka sistem akan menampilkan halaman produk *list* sama seperti gambar 5.28 sebelumnya.

### 5.3.10 Fitur Hapus Produk

Fitur hapus produk ini berfungsi untuk menghapus produk yang ada di basis data. Fitur ini bisa diakses oleh admin dan staff, dan dilakukan dengan cara memilih tombol hapus pada produk. Setelah memilih tombol hapus maka sistem akan menampilkan pesan seperti gambar 5.30 dibawah ini.



Gambar 5.30 Peringatan Hapus Produk

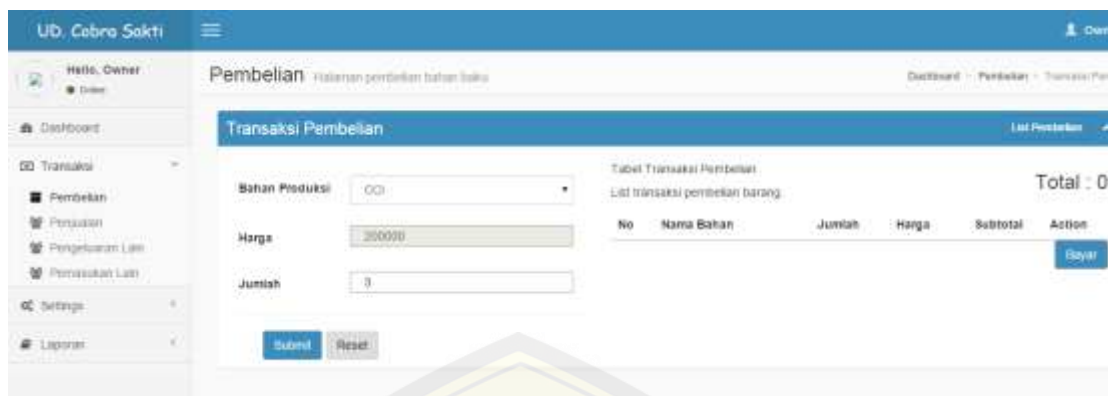
(Sumber: Hasil Analisis, 2015).

Setelah data berhasil dihapus maka sistem akan menampilkan pesan sukses pada halaman data user seperti gambar 5.28 sebelumnya

### 5.3.11 Fitur Add Tansaksi Pembelian

Fitur *add* transaksi pembelian merupakan fitur untuk menambahkan transaksi pembelian kedalam basis data sistem. Transaksi pembelian ini berisi antara lain; bahan produksi, harga dan jumlah. Fitur add transaksi pembelian ini bisa diakses oleh admin dan staff. Untuk tampilan *add* transaksi pembelian seperti gambar 5.31 dibawah ini.





UD. Cobra Sakti

Hello, Owner

Pembelian Halaman pembelian bahan baku

Dashboard : Pembelian - Transaksi Pembelian

Transaksi Pembelian

Bahan Produksi: CCI

Harga: 250000

Jumlah: 0

Total: 0

No	Nama Bahan	Jumlah	Harga	Subtotal	Action
----	------------	--------	-------	----------	--------

Submit Reset

Gambar 5.31 Add Transaksi Pembelian

(Sumber: Hasil Analisis, 2015).

Apabila admin dan staff tidak menginputkan data atau kurang satu data yang kosong, maka sistem akan memberikan atau menampilkan peringatan data kosong seperti gambar 5.32 dibawah ini.



UD. Cobra Sakti

Hello, Owner

Pembelian Halaman pembelian

Dashboard : Pembelian - Transaksi Pembelian

Transaksi Pembelian

Bahan Produksi: CCI

Harga: 250000

Jumlah: Masukkan jumlah pembelian

The page at localhost says:  
Jumlah Bahan Tidak Boleh Kosong!

OK

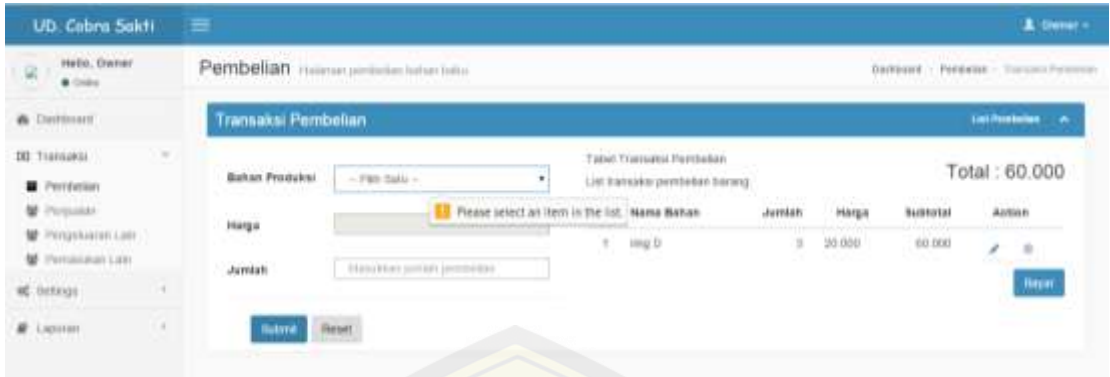
No	Nama Bahan	Jumlah	Harga
----	------------	--------	-------

Submit Reset

Gambar 5.32 Peringatan Kolom Kosong Add Transaksi Pembelian

(Sumber: Hasil Analisis, 2015).

Saat admin dan staff telah menginputkan seluruh data dan data telah *valid* tanpa ada kesalahan maka sistem akan menyimpan di basis data dan menampilkan *list* transaksi pembelian seperti gambar 5.33 dibawah ini.



Gambar 5.33 *List* Transaksi Pembelian

(Sumber: Hasil Analisis, 2015).

Dan setelah masuk ke list transaksi selanjutnya admin atau staff menekan tombol bayar. Secara otomatis transaksi pembelian akan menampilkan *detail list* transaksi pembelian seperti gambar 5.34 di bawah ini.



Gambar 5.34 *Detail List* Transaksi Pembelian

(Sumber: Hasil Analisis, 2015).

### 5.3.12 Fitur *Edit* Tansaksi Pembelian

Fitur *edit* transaksi pembelian berfungsi untuk merubah transaksi yang telah ada di basis data. *Edit* transaksi pembelian bisa diakses oleh admin dan staff, dan dilakukan dengan cara memilih tombol *edit* yang ada pada tabel pembelian . Setelah memilih tombol *edit* maka sistem menampilkan halaman *update* pembelian seperti gambar 5.35 dibawah ini.

No	Nama Bahan	Jumlah	Harga	Subtotal	Action
1	plank pembungkus	2.000	300.000	600.000.000	<a href="#">Edit</a>

Gambar 5.35 Halaman *Edit* Pembelian

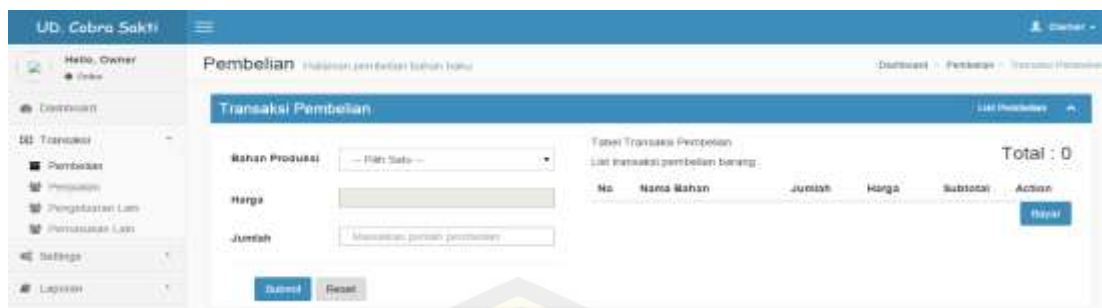
(Sumber: Hasil Analisis, 2015).

Jika admin dan staff telah selesai merubah transaksi pembelian maka sistem akan menampilkan halaman transaksi pembelian sama seperti gambar 5.33 sebelumnya.

### 5.3.13 Fitur Hapus Transaksi Pembelian

Fitur hapus transaksi pembelian ini berfungsi untuk menghapus transaksi yang ada di basis data. Fitur ini bisa diakses oleh admin dan staff, dan dilakukan dengan cara memilih tombol hapus pada transaksi pembelian. Setelah memilih tombol hapus

maka sistem akan menampilkan seperti gambar 5.36 dibawah ini.

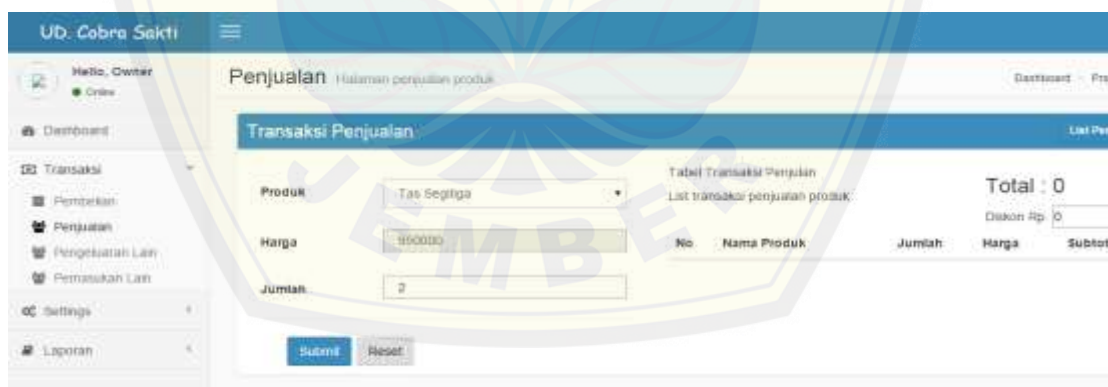


Gambar 5.36 Hapus Transaksi Pembelian

(Sumber: Hasil Analisis, 2015).

#### 5.3.14 Fitur Add Tansaksi Penjualan

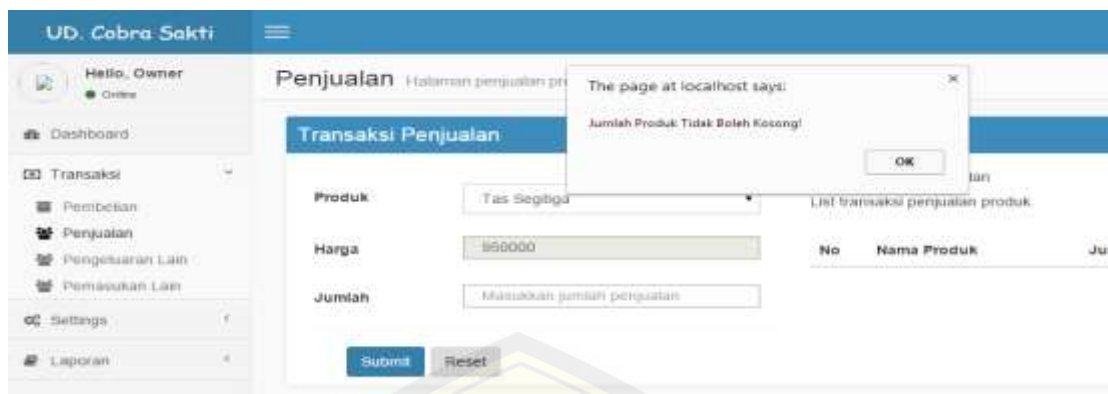
Fitur *add* transaksi penjualan merupakan fitur untuk menambahkan transaksi penjualan kedalam basis data sistem. Transaksi penjualan ini berisi antara lain; produk, harga dan jumlah. Fitur *add* transaksi pembelian ini bisa diakses oleh admin dan kasir. Untuk tampilan *add* transaksi penjualan seperti gambar 5.37 dibawah ini.



Gambar 5.37 Add Transaksi Penjualan

(Sumber: Hasil Analisis, 2015).

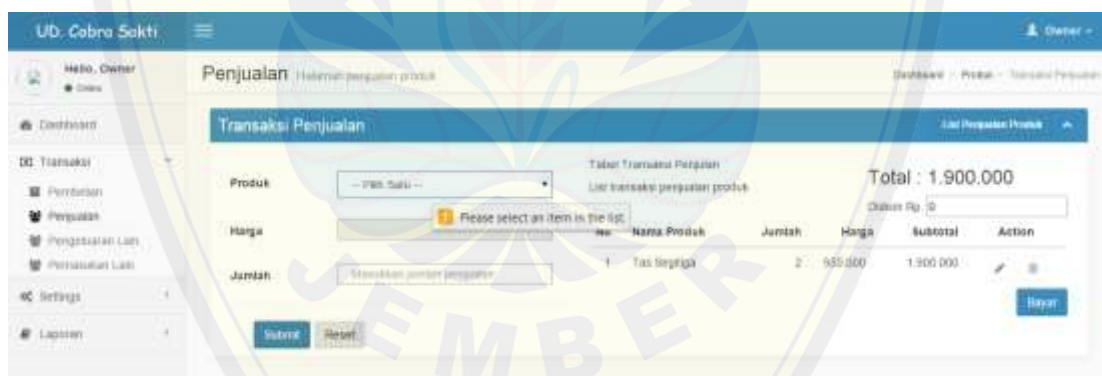
Apabila admin dan kasir tidak menginputkan data atau kurang satu data yang kosong, maka sistem akan memberikan atau menampilkan peringatan data kosong seperti gambar 5.38 dibawah ini.



Gambar 5.38 Peringatan Kolom Kosong Add Transaksi Penjualan

(Sumber: Hasil Analisis, 2015).

Saat admin dan kasir telah menginputkan seluruh data dan data telah *valid* tanpa ada kesalahan maka sistem akan menyimpan di basis data dan menampilkan *list* transaksi penjualan seperti gambar 5.39 dibawah ini.



Gambar 5.39 List Transaksi Penjualan

(Sumber: Hasil Analisis, 2015).

Dan setelah masuk ke *list* transaksi selanjutnya admin atau kasir menekan tombol bayar. Secara otomatis transaksi penjualan akan menampilkan *detail list* transaksi penjualan seperti gambar 5.40 di bawah ini.



Gambar 5.40 *Detail List* Transaksi Penjualan

(Sumber: Hasil Analisis, 2015).

### 5.3.15 Fitur *Edit* Tansaksi Penjualan

Fitur *edit* transaksi penjualan berfungsi untuk merubah transaksi yang telah ada di basis data. *Edit* transaksi penjualan bisa diakses oleh admin dan kasir, dan dilakukan dengan cara memilih tombol *edit* yang ada pada tabel penjualan . Setelah memilih tombol edit maka sistem menampilkan halaman *update* penjualan seperti gambar 5.41 dibawah ini.



Gambar 5.41 Halaman *Edit* Penjualan

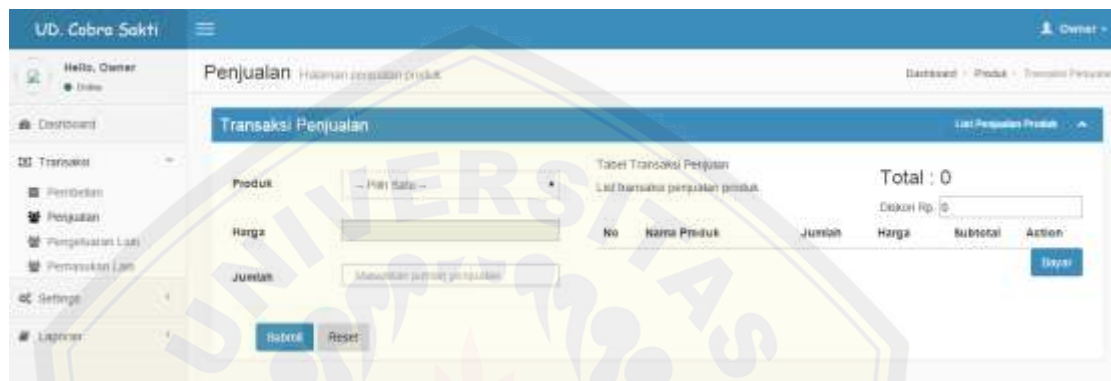
(Sumber: Hasil Analisis, 2015).

Jika admin dan kasir telah selesai merubah transaksi penjualan maka sistem akan menampilkan halaman transaksi penjualan sama seperti gambar 5.39 sebelumnya.



### 5.3.16 Fitur Hapus Transaksi Penjualan

Fitur hapus transaksi penjualan ini berfungsi untuk menghapus transaksi yang ada di basis data. Fitur ini bisa diakses oleh admin dan kasir, dan dilakukan dengan cara memilih tombol hapus pada transaksi penjualan. Setelah memilih tombol hapus maka sistem akan menampilkan pesan seperti gambar 5.42 dibawah ini.




Gambar 5.42 Hapus Transaksi Pembelian

(Sumber: Hasil Analisis, 2015).

### 5.3.17 Fitur Add Tansaksi Pemasukan lain

Fitur *add* transaksi pemasukan lain merupakan fitur untuk menambahkan transaksi pemasukan lain kedalam basis data sistem. Transaksi pemasukan lain ini berisi antara lain; keterangan pemasukan dan jumlah pemasukan. Fitur *add* transaksi pemasukan lain ini bisa diakses oleh admin dan kasir. Untuk tampilan *add* transaksi pemasukan lain seperti gambar 5.43 dibawah ini.





UD. Cobra Sakti

Hello, Owner  
Online

Pemasukan Lain - lain Halaman tambah pemasukan.

Pemasukan Add Form

Keterangan Pemasukan: diskon

Jumlah Pemasukan: 150000

Submit Cancel

Gambar 5.43 Add Transaksi Pemasukan Lain

(Sumber: Hasil Analisis, 2015).

### 5.3.18 Fitur Add Tansaksi Pengeluaran lain

Fitur *add* transaksi pengeluaranlain merupakan fitur untuk menambahkan transaksi pengeluaran lain kedalam basis data sistem. Transaksi pengeluaran lain ini berisi antara lain; keterangan pengeluaran dan jumlah pengeluaran. Fitur *add* transaksi pengeluaran lain ini bisa diakses oleh admin dan kasir. Untuk tampilan *add* transaksi pengeluaran lain seperti gambar 5.44 dibawah ini.



UD. Cobra Sakti

Hello, Owner  
Online

Pemasukan Lain - lain Halaman tambah pemasukan.

Pemasukan Add Form

Keterangan Pemasukan: gaji karyawan

Jumlah Pemasukan: 50000000

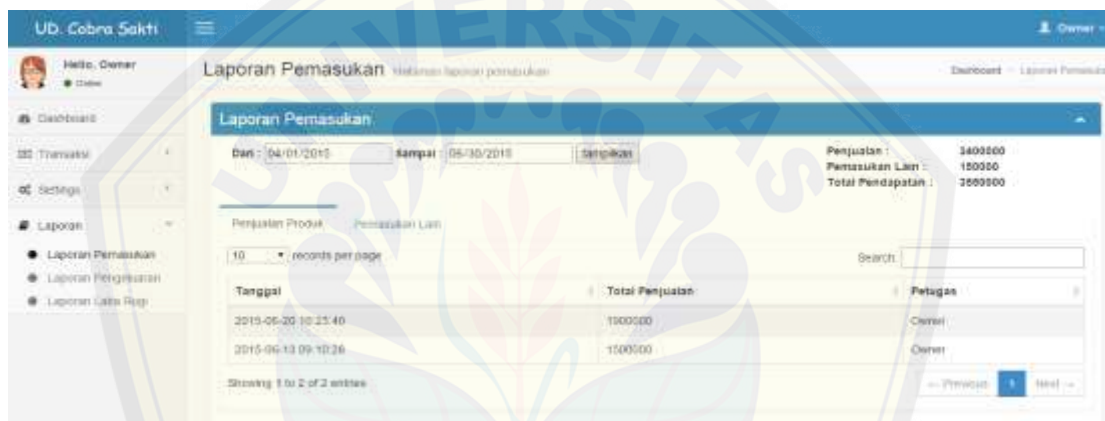
Submit Cancel

Gambar 5.44 Add Transaksi Pengeluaran Lain

(Sumber: Hasil Analisis, 2015).

### 5.3.19 Fitur *view* Laporan Pemasukan

Fitur *view* laporan pemasukan ini berfungsi untuk menampilkan setiap transaksi pemasukan yang telah tersimpan di basis data. Fitur ini bisa diakses oleh admin saja, dengan cara memilih menu laporan pemasukan kemudian memasukkan tanggal dari dan sampaitanggal berapa laporan yang akan dilihat. Setelah selesai memasukkan tanggal dan menekan tombol tampilkan maka sistem akan menampilkan halaman laporan pemasukan yang didapat dari penjualan produk seperti gambar 5.45 dan laporan pemasukan lain seperti gambar 5.46 dibawah ini.



Gambar 5.45 *View* Laporan Pemasukan

(Sumber: Hasil Analisis, 2015).

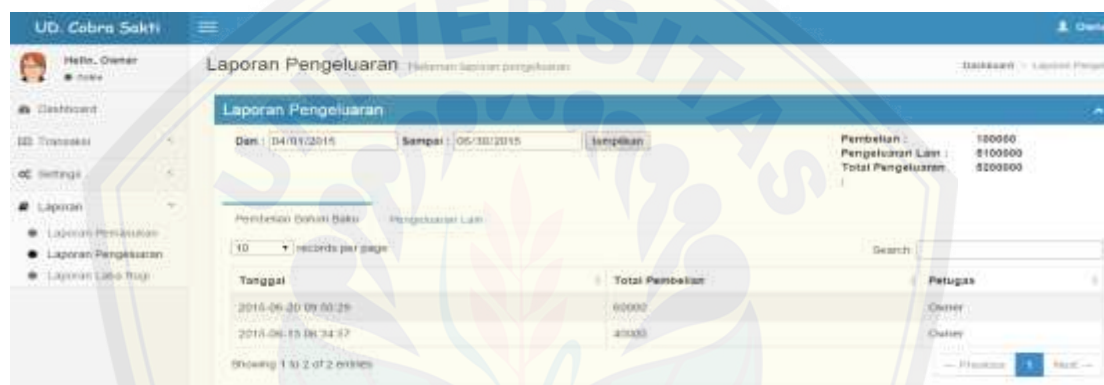


Gambar 5.46 *View* Laporan Pemasukan Lain

(Sumber: Hasil Analisis, 2015).

### 5.3.20 Fitur View Laporan Pengeluaran

Fitur *view* laporan pengeluaran ini berfungsi untuk menampilkan setiap transaksi pengeluaran yang telah tersimpan di basis data. Fitur ini bisa diakses oleh admin saja, dengan cara memilih menu laporan pengeluaran kemudian memasukkan tanggal dari dan sampai tanggal berapa laporan yang akan dilihat. Setelah selesai memasukkan tanggal dan menekan tombol tampilkan maka sistem akan menampilkan halaman laporan pengeluaran yang didapat dari pembelian bahan baku seperti gambar 5.47 dan laporan pengeluaran lain seperti gambar 5.48 dibawah ini.



The screenshot shows a web application interface for 'UD. Cobra Sakti'. The main content area is titled 'Laporan Pengeluaran' and displays a table of transactions. The table has columns for 'Tanggal' (Date), 'Total Pembelian' (Total Purchase), and 'Petugas' (Staff). The data shown is as follows:

Tanggal	Total Pembelian	Petugas
2015-06-30 09:50:29	60000	Owner
2015-06-15 08:34:57	40000	Owner

Summary statistics on the right: Pembelian: 100000, Pengeluaran Lain: 0, Total Pengeluaran: 100000. The interface also includes a sidebar with navigation options like 'Dashboard', 'Transaksi', 'Settings', and 'Laporan'.

Gambar 5.47 View Laporan Pengeluaran

(Sumber: Hasil Analisis, 2015).



The screenshot shows the same web application interface, but the table displays transactions for 'Pengeluaran Lain'. The table has columns for 'Tanggal' (Date), 'Keterangan' (Description), 'Total Pengeluaran' (Total Expense), and 'Petugas' (Staff). The data shown is as follows:

Tanggal	Keterangan	Total Pengeluaran	Petugas
2015-06-20 10:45:43	gaji karyawan	500000	Owner
2015-06-19 23:36:34	gaji	100000	Owner

Summary statistics on the right: Pembelian: 100000, Pengeluaran Lain: 600000, Total Pengeluaran: 700000. The interface also includes a sidebar with navigation options like 'Dashboard', 'Transaksi', 'Settings', and 'Laporan'.

Gambar 5.48 View Laporan Pengeluaran Lain

(Sumber: Hasil Analisis, 2015).

### 5.3.21 Fitur View Laporan LabaRugi

Fitur *view* laporan labarugi ini berfungsi untuk menampilkan hasil labarugi setiap transaksi penjualan dan pembelian yang telah tersimpan di basis data. Fitur ini bisa diakses oleh admin saja, dengan cara memilih menu laporan labarugi kemudian memasukkan tanggal dari dan sampai tanggal berapa laporan labarugi yang akan dilihat. Setelah selesai memasukkan tanggal dan menekan tombol tampilkan maka sistem akan menampilkan halaman laporan penjualan produk yang didapat dari hasil total penjualan dan pembelian seperti gambar 5.49 dan laporan pemasukan lain didapat dari hasil total pemasukan dan pengeluaran seperti gambar 5.50 dibawah ini.

The screenshot shows a web interface for 'Ub, Cobra Sakti' with a user profile 'Harto, Owner'. The main content is the 'Laporan Laba Rugi' page. It includes a sidebar with navigation options like 'Dashboard', 'Transaksi', 'Setings', and 'Laporan'. The report itself has a date range filter set to 'Dari: 04-01-2015' and 'Sampai: 06-30-2015'. Summary statistics are shown as follows:

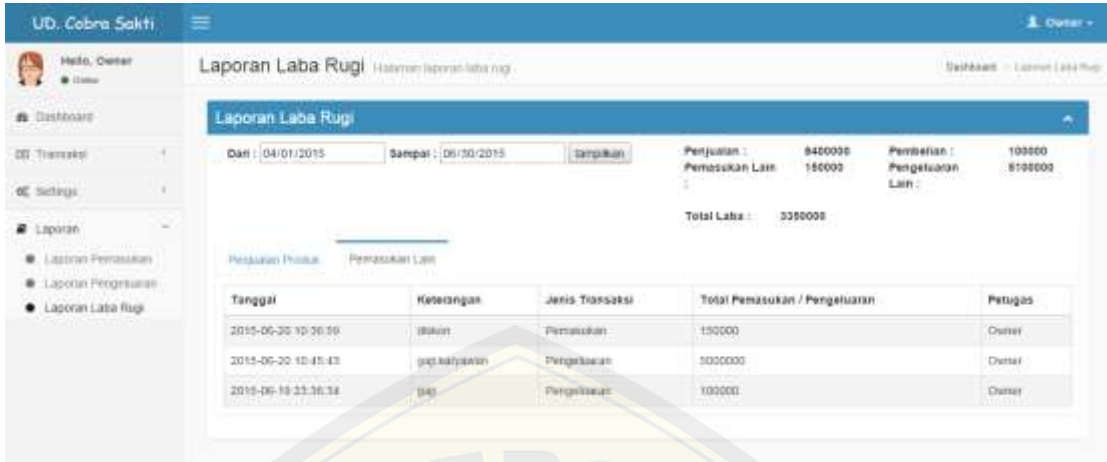
Penjualan :	840000	Pembelian :	100000
Pemasukan Lain :	160000	Pengeluaran Lain :	610000
Total Laba :		330000	

Below the summary, there is a table of transactions with columns for 'Tanggal', 'Jenis Transaksi', 'Total Penjualan / Pembelian', and 'Petugas'. The data is as follows:

Tanggal	Jenis Transaksi	Total Penjualan / Pembelian	Petugas
2015-06-12 08:34:57	40000	Pembelian	Owner
2015-06-13 09:10:26	1000000	Penjualan	Owner
2015-06-20 09:50:29	60000	Pembelian	Owner
2015-06-20 10:23:40	1900000	Penjualan	Owner
2015-06-20 11:09:02	1400000	Pembelian	Owner
2015-06-20 14:10:58	3600000	Penjualan	Owner

Gambar 5.49 View Laporan Laba Rugi Penjualan Pembelian

(Sumber: Hasil Analisis, 2015).



Gambar 5.50 View Laporan Laba Rugi Pemasukan Pengeluaran Lain  
(Sumber: Hasil Analisis, 2015).

## BAB 6. Penutup

Bab penutup merupakan bagian akhir dari penulisan tugas akhir ini. Pada bab ini menjelaskan mengenai kesimpulan dan saran dari hasil penelitian selama ini. Kesimpulan dibuat berdasarkan apa yang didapat dari semua tahap penelitian sedangkan saran merupakan apa yang diharapkan untuk penelitian selanjutnya.

### 6.1 Kesimpulan

Kesimpulan dari hasil penelitian yang telah dilakukan adalah sebagai berikut:

1. Perancangan dan Pembuatan Aplikasi Akuntansi UD Cobra Sakti Berbasis Web ini mampu menghasilkan laporan keuangan bulanan seperti laporan pemasukan, laporan pengeluaran dan laporan labarugi.
2. Perancangan dan Pembuatan Aplikasi Akuntansi UD Cobra Sakti Berbasis Web ini dapat mempercepat kinerja petugas bagian pengelolaan transaksi pemasukan dan pengeluaran pada UD Cobra Sakti.
3. Aplikasi akuntansi berbasis web memberikan keuntungan bagi UD Cobra Sakti, karena informasi pendapatan secara cepat dapat diketahui sehingga dapat lebih mudah dan cepat dalam menentukan sebuah keputusan yang berhubungan dengan pendapatan keuangan.

### 6.2 Saran

Pengembangan lebih lanjut dapat dilakukan dengan menambah lebih detail pada proses akuntansi. Penelitian masih hanya terfokus pada hasil laba rugi saja. Diharapkan penelitian selanjutnya bisa lebih melengkapi di tahapan-tahapan akuntansinya..



## DAFTAR PUSTAKA

- Hanum, Y. (2003). *Rekayasa Perangkat Lunak*. Jakarta: Erlangga.
- Ir. Kuswandi, M. (2008). *Pencatatan Keuangan Usaha Dagang*. Jakarta: PT. Gramedia.
- Irnama Sari, D. (2011). *Pembuatan Sistem Informasi Keuangan Berbasis Web pada UD. Saputra Jaya*. Yogyakarta.
- Kusrini, & Koniyo, A. (2007). *Membangun Sistem Informasi Akuntansi dengan Visual Basic & Microsoft SQL Server*. Yogyakarta: Andi.
- maretha, e. d. (2008). *Akuntansi Dasar I*. Semarang: Grasindo.
- Muawanah, U., & Poernawati, F. (2008). *Konsep Dasar Akuntansi dan Pelaporan Keuangan*. Jakarta: Direktorat Pembinaan Sekolah Menengah Kejuruan.
- Prasetyo, E. (2008). *Pemograman Web PHP dan MySQL*. Yogyakarta: Graha Ilmi.
- Santoso, H. (2000). *Membuat Multiaplikasi Menggunakan Visual Basic 6*. elex media komputindo.



## LAMPIRAN

### Lampiran A : Usecase Scenario

1. Usecase scenario “Memajemen Data User”

<b>ID</b>	CB-01	
<b>Name</b>	Memajemen Data User	
<b>Participating Actor</b>	Admin	
<b>Entry Condition</b>	Admin ingin menginput data User	
<b>Exit Condition</b>	Admin selesai menginput data User	
<b>Skenario Utama</b>		
<b>“Input Data User”</b>		
<b>Admin</b>	<b>Sistem</b>	
1. Memilih Menu User		
	2. Menampilkan list User	
3. Klik +		
	4. Menampilkan halaman add User	
5. Mengisi akses, username, password, nama		
6. Klik Submit		
	7. View alert sukses	
	8. Menampilkan list user	
<b>Skenario Alternatif</b>		

<b>“Pengisian Form User Baru yang Tidak Benar”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Pengisian form tambah User masih belum lengkap	
	7. view alert gagal “Please fill out this field”
	8. menampilkan halaman add form
<b>Skenario Alternatif “Memilih Tombol Cancel”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Klik Cancel	
	7. Kembali ke halaman list User dan menampilkan tabel data User, tombol Edit dan Delete, dan detail
<b>Skenario Utama “Edit Data User”</b>	
<b>Admin</b>	<b>Sistem</b>
1. Memilih Menu User	
	2. Menampilkan list user
2. Klik tombol Edit	
	3. Menampilkan form edit User
4. Mengedit akses, username, password, nama	
5. Klik Submit	

	6. View alert sukses
	7. Menampilkan list user
<b>Skenario Alternatif</b>	
<b>“Pengisian Form Edit User yang Tidak Benar”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Pengisian form edit User masih belum lengkap	
	7. View alert gagal “Please fill out this field”
	8. Menampilkan form edit user
<b>Skenario Alternatif</b>	
<b>“Memilih Tombol Cancel”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Klik Cancel	
	7. Kembali ke halaman list User dan menampilkan tabel data User, tombol Edit dan Delete.
<b>Skenario Utama</b>	
<b>“Menghapus Data User”</b>	
<b>Admin</b>	<b>Sistem</b>
1. Memilih Menu User.	
	2. Menampilkan list user
3. Klik tombol Delete	
	4. Menampilkan alert hapus

5. Klik ok	
	6. Menampilkan alert sukses
	7. Menampilkan list user
<b>Skenario Alternatif</b>	
<b>“Memilih Tombol Cancel”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Klik Cancel	
	7. Menampilkan list user

2. Usecase scenario “Memajemen Data Produk”

<b>ID</b>	CB-01
<b>Name</b>	Memanejemen data produk
<b>Participating Actor</b>	Admin
<b>Entry Condition</b>	Admin ingin menginput data Produk
<b>Exit Condition</b>	Admin selesai menginput data Produk
<b>Skenario Utama</b>	
<b>“Input Data Produk”</b>	
<b>Admin</b>	<b>Sistem</b>
1. Memilih Menu Produk	
	2. Menampilkan list produk
3. Klik +	
	4. Menampilkan halaman add Produk

5. Mengisi nama, deskripsi, jumlah, satuan, harga	
6. Klik Submit	
	7. View alert sukses
	8. Menampilkan list produk
<b>Skenario Alternatif</b>	
<b>“Pengisian Form Produk Baru yang Tidak Benar”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Pengisian form tambah Produk masih belum lengkap	
	7. View alert gagal “Please fill out this field”
	8. Menampilkan List Produk
<b>Skenario Alternatif</b>	
<b>“Memilih Tombol Cancel”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Klik Cancel	
	7. Kembali ke halaman list Produk dan menampilkan tabel data Produk, tombol Edit dan Delete, dan detail
<b>Skenario Utama</b>	
<b>“Edit Data Produk”</b>	
<b>Admin</b>	<b>Sistem</b>
1. Memilih Menu Produk	
	2. Menampilkan list produk

3. Klik tombol Edit	
	4. Menampilkan form edit Produk
5. Mengedit nama, deskripsi, jumlah, satuan, harga	
6. Klik Submit	
	7. View alert sukses
	8. Menampilkan list produk
<b>Skenario Alternatif</b>	
<b>“Pengisian Form Edit Produk yang Tidak Benar”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Pengisian form edit Produk masih belum lengkap	
	7. View alert gagal “Please fill out this field”
	8. Menampilkan form edit produk
<b>Skenario Alternatif</b>	
<b>“Memilih Tombol Cancel”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Klik Cancel	
	7. Kembali ke halaman list Produk dan menampilkan tabel data Produk, tombol Edit dan Delete.
<b>Skenario Utama</b>	
<b>“Menghapus Data Produk”</b>	
<b>Admin</b>	<b>Sistem</b>

1. Memilih Menu Produk.	
	2. Menampilkan list produk
3. Klik tombol Delete	
	4. Menampilkan alert hapus
5. Klik ok	
	6. Menampilkan alert sukses
	7. Menampilkan list Produk
<b>Skenario Alternatif</b>	
<b>“Memilih Tombol Cancel”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Klik Cancel	
	7. Menampilkan list produk

3. Usecase scenario “Memanajemen Data Bahan Baku”

<b>ID</b>	CB-01
<b>Name</b>	Memanajemen Data Bahan Baku
<b>Participating Actor</b>	Admin
<b>Entry Condition</b>	Admin ingin menginput data Bahan Baku
<b>Exit Condition</b>	Admin selesai menginput data Bahan Baku
<b>Skenario Utama</b>	
<b>“Input Data Bahan Baku”</b>	
<b>Admin</b>	<b>Sistem</b>



1. Klik Menu Bahan Baku	
	4. Menampilkan list bahan baku
5. Klik +	
	6. Menampilkan halaman add form
7. Mengisi nama, harga, jumlah, satuan	
8. Klik Submit	
	9. View alert sukses
	10. Menampilkan list bahan baku
<b>Skenario Alternatif</b>	
<b>“Pengisian Form Bahan Baku Baru yang Tidak Benar”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Pengisian form tambah bahan baku masih belum lengkap	
	7. Menampilkan Warning Message (view alert gagal) “Please fill out this field”
	8. Menampilkan halaman add form
<b>Skenario Alternatif</b>	
<b>“Memilih Tombol Cancel”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Klik Cancel	
	7. Kembali ke halaman list bahan baku dan menampilkan tabel data bahan baku, tombol Edit dan Delete, dan

	detail
<b>Skenario Utama</b> <b>“Edit Data Bahan Baku”</b>	
<b>Admin</b>	<b>Sistem</b>
1. Memilih Menu bahan baku	
	2. Menampilkan list bahan baku
3. Klik Edit	
	4. Menampilkan form edit bahan baku
5. Mengedit nama, harga, jumlah, satuan	
6. Klik Submit	
	7. Menampilkan alert sukses
	8. Menampilkan list bahan baku
<b>Skenario Alternatif</b> <b>“Pengisian Form Edit Bahan Baku yang Tidak Benar”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Pengisian form edit bahan baku masih belum lengkap	
	7. Menampilkan alert gagal “Please fill out this field”
	8. Menampilkan list bahan baku
<b>Skenario Alternatif</b> <b>“Memilih Tombol Cancel”</b>	
<b>Admin</b>	<b>Sistem</b>

6. Klik Cancel	
	7. Kembali ke halaman list bahan baku dan menampilkan tabel data bahan baku, tombol Edit dan Delete.
<b>Skenario Utama</b>	
<b>“Delete Data Bahan Baku”</b>	
<b>Admin</b>	<b>Sistem</b>
1. Memilih Menu bahan baku.	
	2. Menampilkan list bahan baku
3. Klik tombol Delete	
	4. Menampilkan alert hapus
5. Klik ok	
	6. Menampilkan alert sukses
	7. Menampilkan list bahan baku
<b>Skenario Alternatif</b>	
<b>“Memilih Tombol Cancel”</b>	
<b>Admin</b>	<b>Sistem</b>
6. Klik Cancel	
	7. Menampilkan list bahan baku

## 4. Scenario Diagram “Menambah Pengeluaran Lain”

<b>ID</b>	CB-01	
<b>Name</b>	Menambah Pengeluaran Lain	
<b>Participating Actor</b>	Admin	
<b>Entry Condition</b>	Admin ingin menginput data Pengeluaran Lain	
<b>Exit Condition</b>	Admin selesai menginput data Pengeluaran Lain	
<b>Skenario Utama</b>		
<b>“Input Data Pengeluaran Lain”</b>		
<b>Admin</b>	<b>Sistem</b>	
1. Memilih Menu Pengeluaran Lain		
	2. Menampilkan form input pengeluaran lain	
3. Input Jumlah		
4. Klik Submit		
	5. Menampilkan form input pengeluaran lain	

## 5. Scenario Diagram “Menambah Pemasukan Lain”

<b>ID</b>	CB-01	
<b>Name</b>	Menambah Pemasukan Lain	
<b>Participating Actor</b>	Admin	
<b>Entry Condition</b>	Admin ingin menginput data Pemasukan Lain	
<b>Exit Condition</b>	Admin selesai menginput data Pemasukan Lain	
<b>Skenario Utama</b> “Input Data Pemasukan Lain”		
<b>Admin</b>	<b>Sistem</b>	
1. Memilih Menu Pemasukan Lain		
	2. Menampilkan form input pemasukan lain	
3. Input Jumlah		
4. Klik Submit		
	5. Menampilkan form input pemasukan lain	

## 6.Scenario Diagram “Menampilkan Laporan Pengeluaran”

<b>ID</b>	CB-01	
<b>Name</b>	Menampilkan Laporan Pengeluaran	
<b>Participating Actor</b>	Admin	
<b>Entry Condition</b>	Admin ingin melihat laporan pengeluaran dan laporan pengeluaran lain	
<b>Exit Condition</b>	Admin selesai melihat laporan pengeluaran dan laporan pengeluaran lain	
<b>Skenario Utama</b>		
<b>“Menampilkan Laporan Pengeluaran”</b>		
<b>Admin</b>	<b>Sistem</b>	
1. Klik laporan pengeluaran		
	2. Menampilkan halaman laporan pengeluaran	
3. Input tanggal tampilan laporan		
4. Klik pembelian bahan baku		
5. Klik tampilkan		
	6. Menampilkan laporan sesuai tanggal	
7. Klik cetak		
	8. Menampilkan halaman pdf	
9. Klik pengeluaran lain		
10. Klik tampilkan		
	11. Menampilkan laporan pengeluaran lain sesuai tanggal	

12. Klik cetak	
	13. Menampilkan halaman pdf
14. Klik close	
	15. Menampilkan halaman laporan pengeluaran

7. Scenario Diagram “Menampilkan Laporan Pemasukan”

<b>ID</b>	CB-01	
<b>Name</b>	Menampilkan Laporan Pemasukan	
<b>Participating Actor</b>	Admin	
<b>Entry Condition</b>	Admin ingin melihat laporan pemasukan dan laporan pemasukan lain	
<b>Exit Condition</b>	Admin selesai melihat laporan pemasukan dan laporan pemasukan lain	
<b>Skenario Utama</b>		
<b>“Menampilkan Laporan Pemasukan”</b>		
<b>Admin</b>	<b>Sistem</b>	
1. Klik laporan pemasukan		
	2. Menampilkan halaman laporan pemasukan	
3. Input tanggal tampilan laporan		
4. Klik produk penjualan		
5. Klik tampilkan		
	6. Menampilkan laporan sesuai tanggal	
7. Klik cetak		



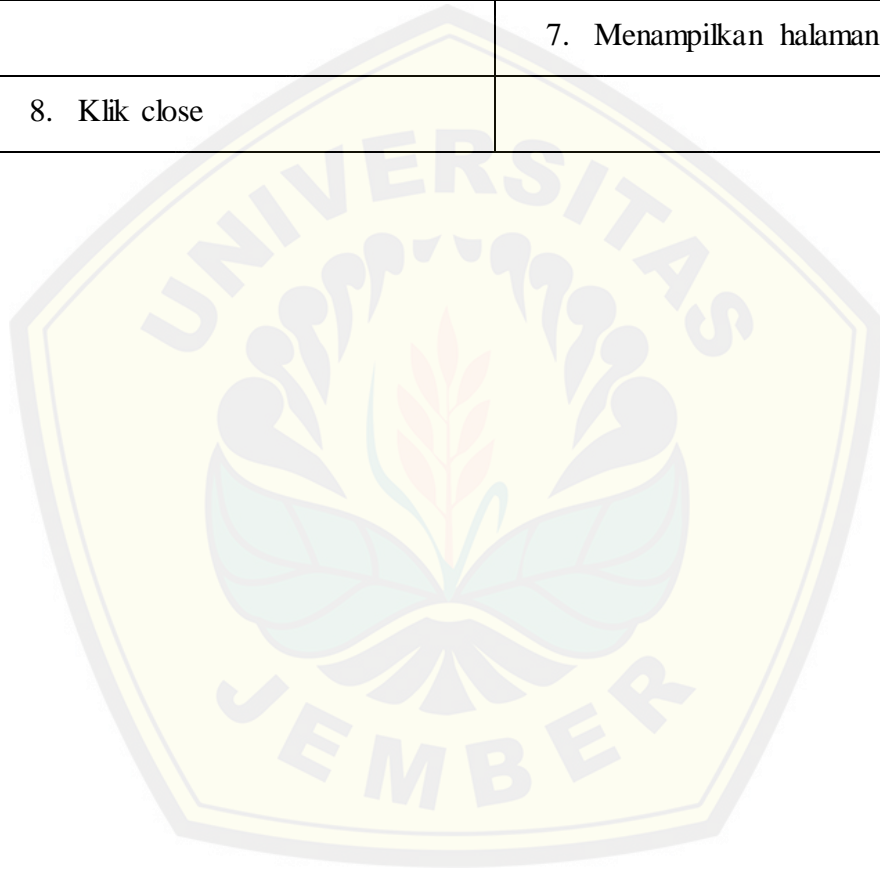
	8. Menampilkan halaman pdf
9. Klik pemasukan lain	
10. Klik tampilkan	
	11. Menampilkan laporan pemasukan lain sesuai tanggal
12. Klik cetak	
	13. Menampilkan halaman pdf
14. Klik close	
	15. Menampilkan halaman laporan pemasukan

8.Scenario Diagram “Menampilkan Laporan LabaRugi”

<b>ID</b>	CB-01
<b>Name</b>	Menampilkan Laporan Laba Rugi
<b>Participating Actor</b>	Admin
<b>Entry Condition</b>	Admin ingin melihat laporan laba rugi
<b>Exit Condition</b>	Admin selesai melihat laporan laba rugi

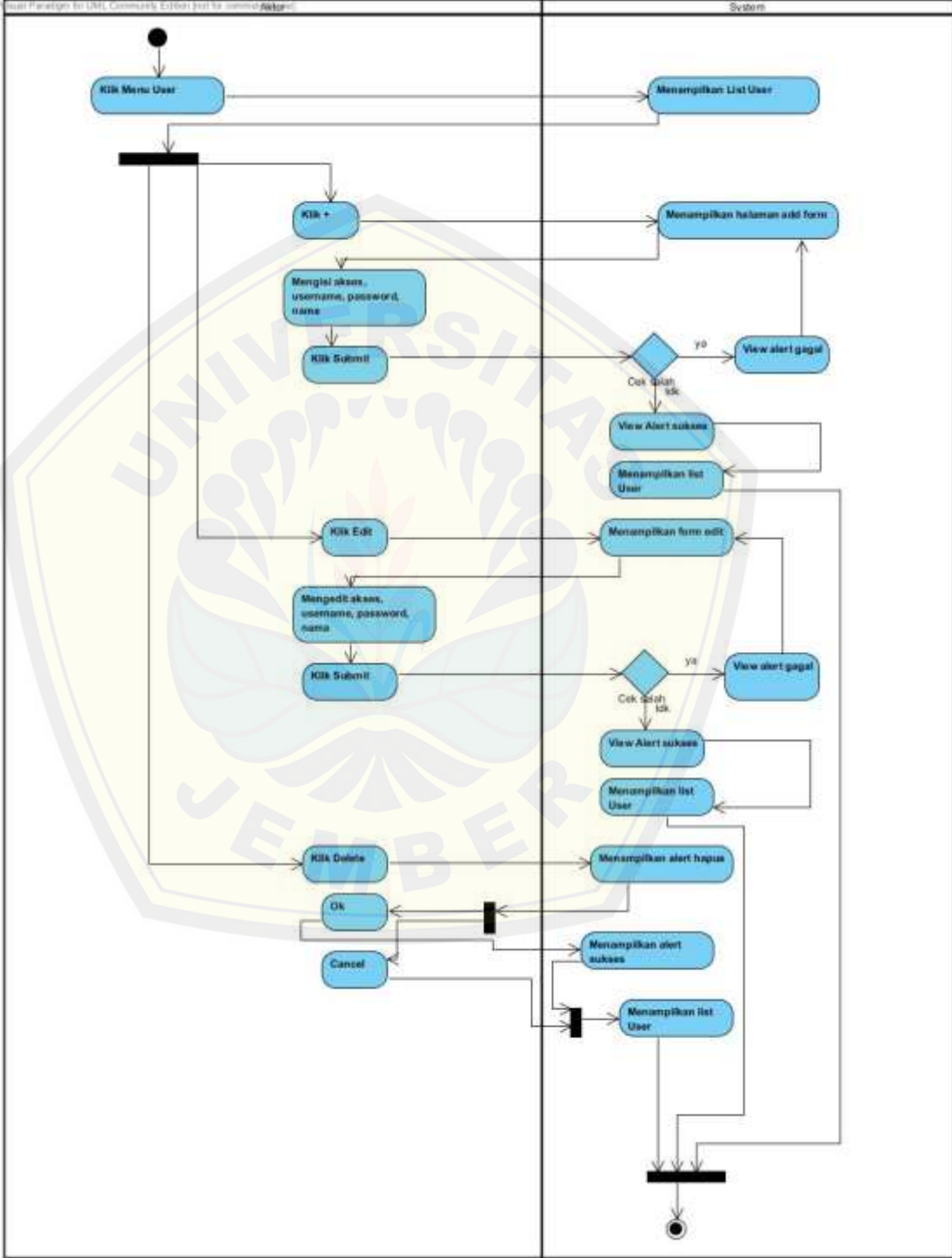
<b>Skenario Utama</b>	
<b>“Menampilkan Laporan Laba Rugi”</b>	
<b>Admin</b>	<b>Sistem</b>
1. Klik laporan laba rugi	
	2. Menampilkan halaman laporan laba rugi

3. Input tanggal tampilan laporan	
4. Klik tampilkan	
	5. Menampilkan laporan sesuai tanggal
6. Klik cetak	
	7. Menampilkan halaman pdf
8. Klik close	

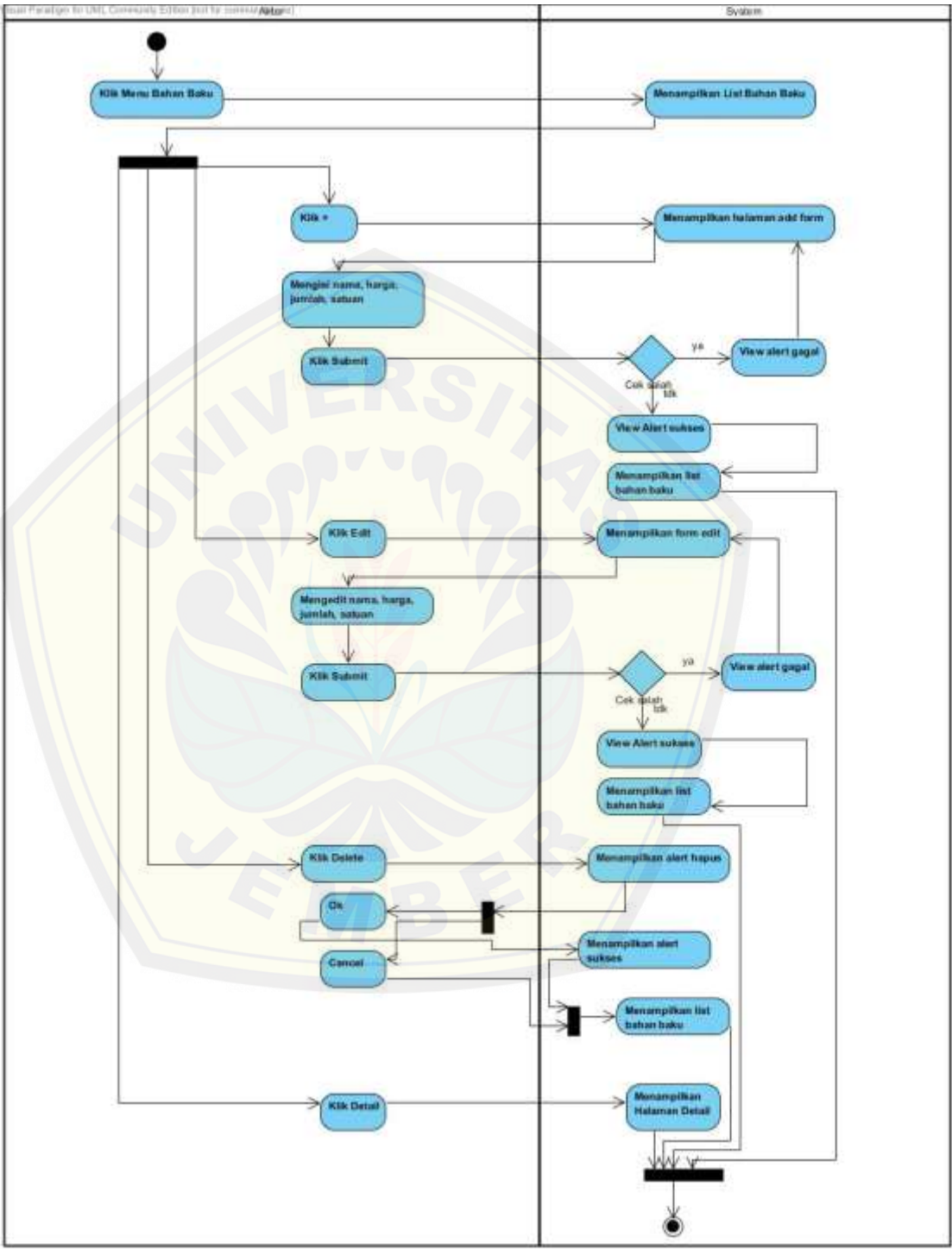


Lampiran B : Activity Diagram

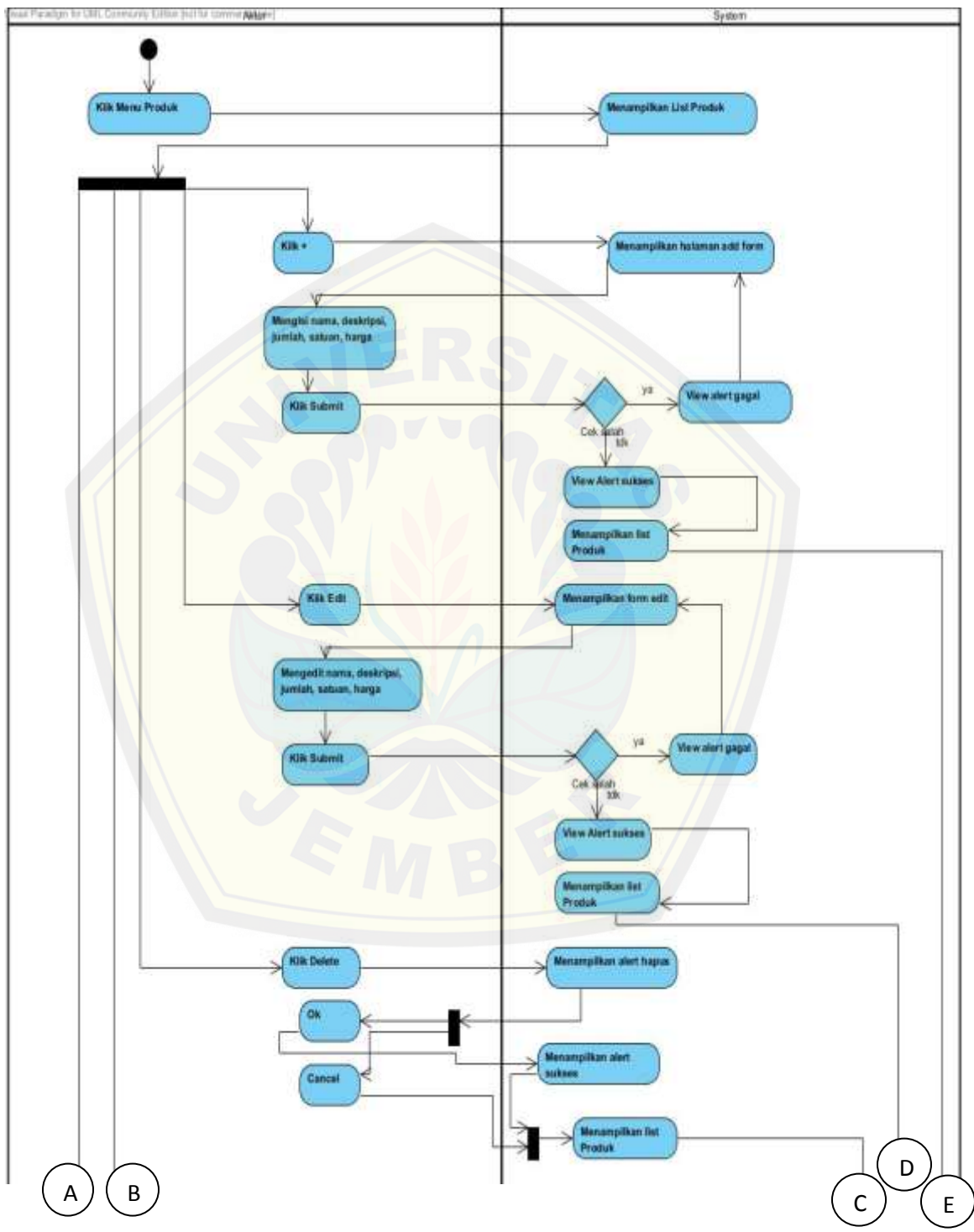
1. Activity Diagram “Memajemen Data User”

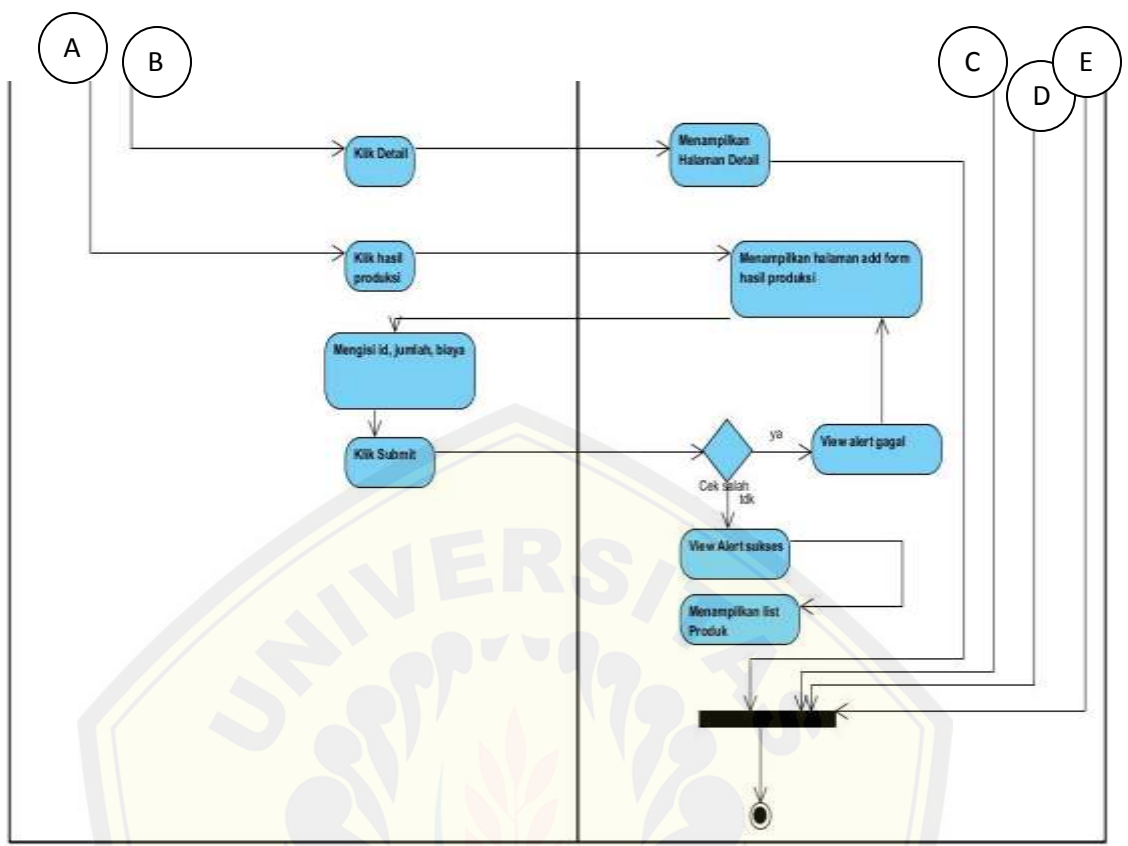


2. Activity Diagram “Memanajemen Data Bahan Baku”



3. Activity Diagram "Memanajemen Data Produk"

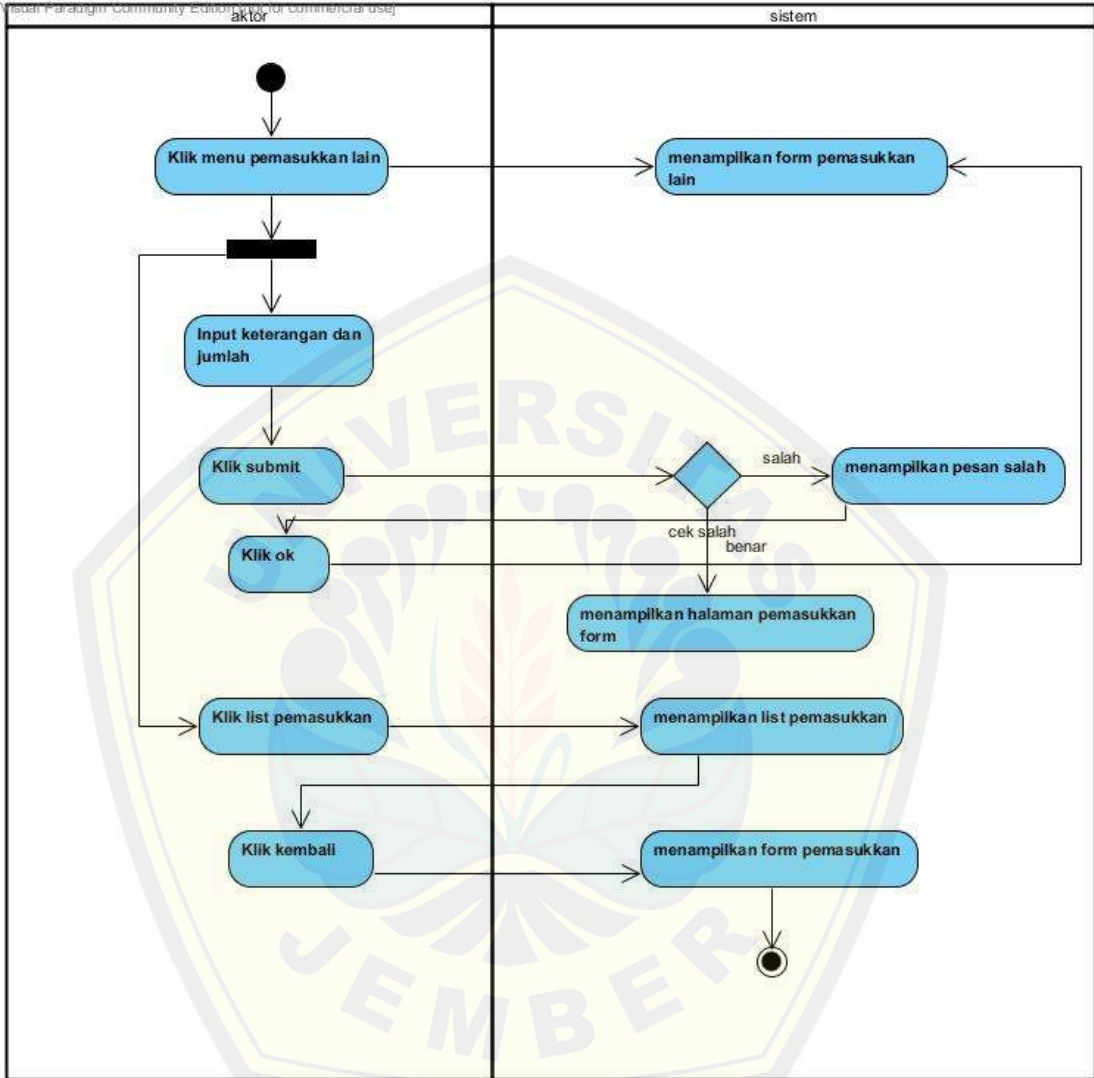




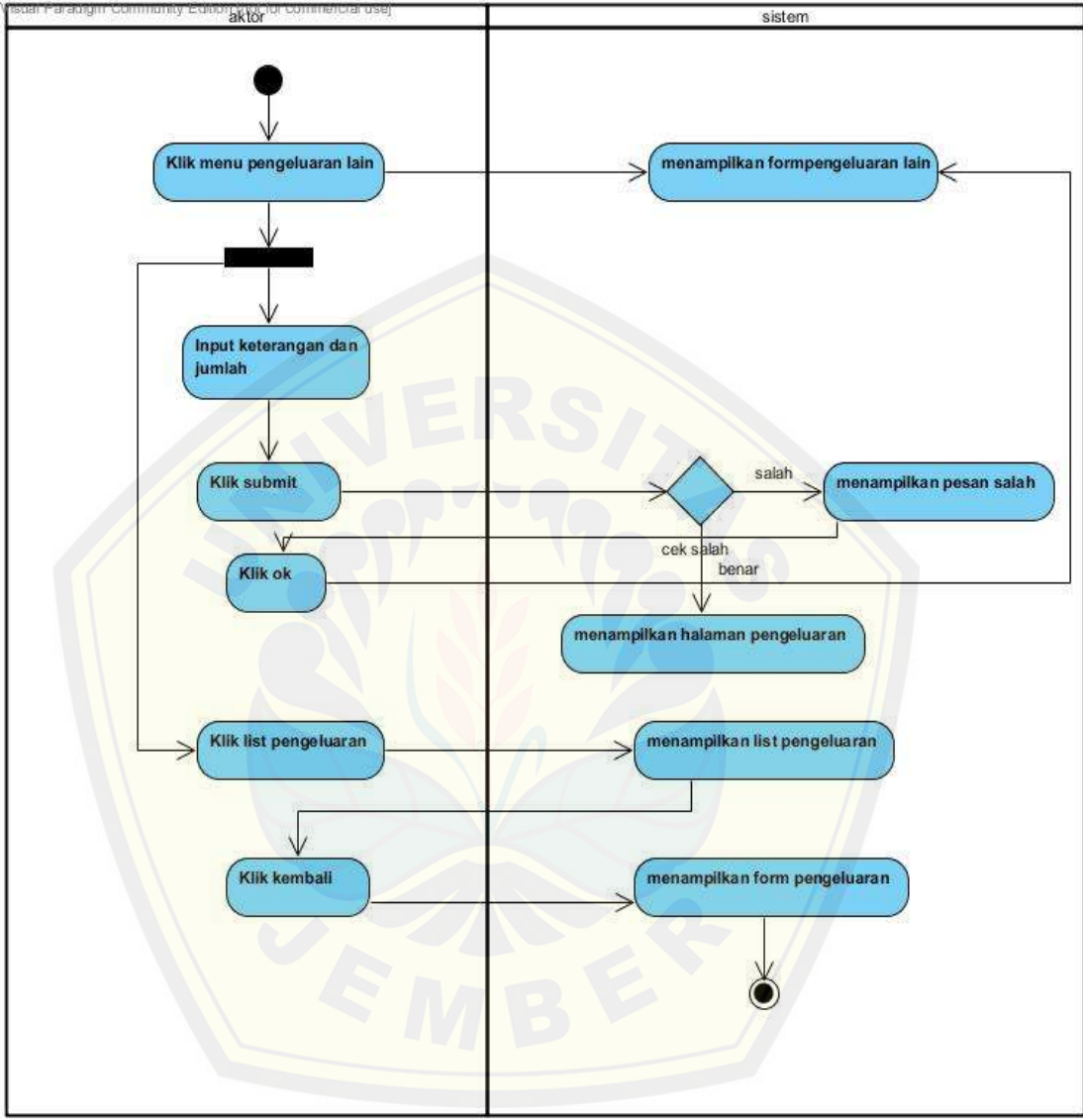




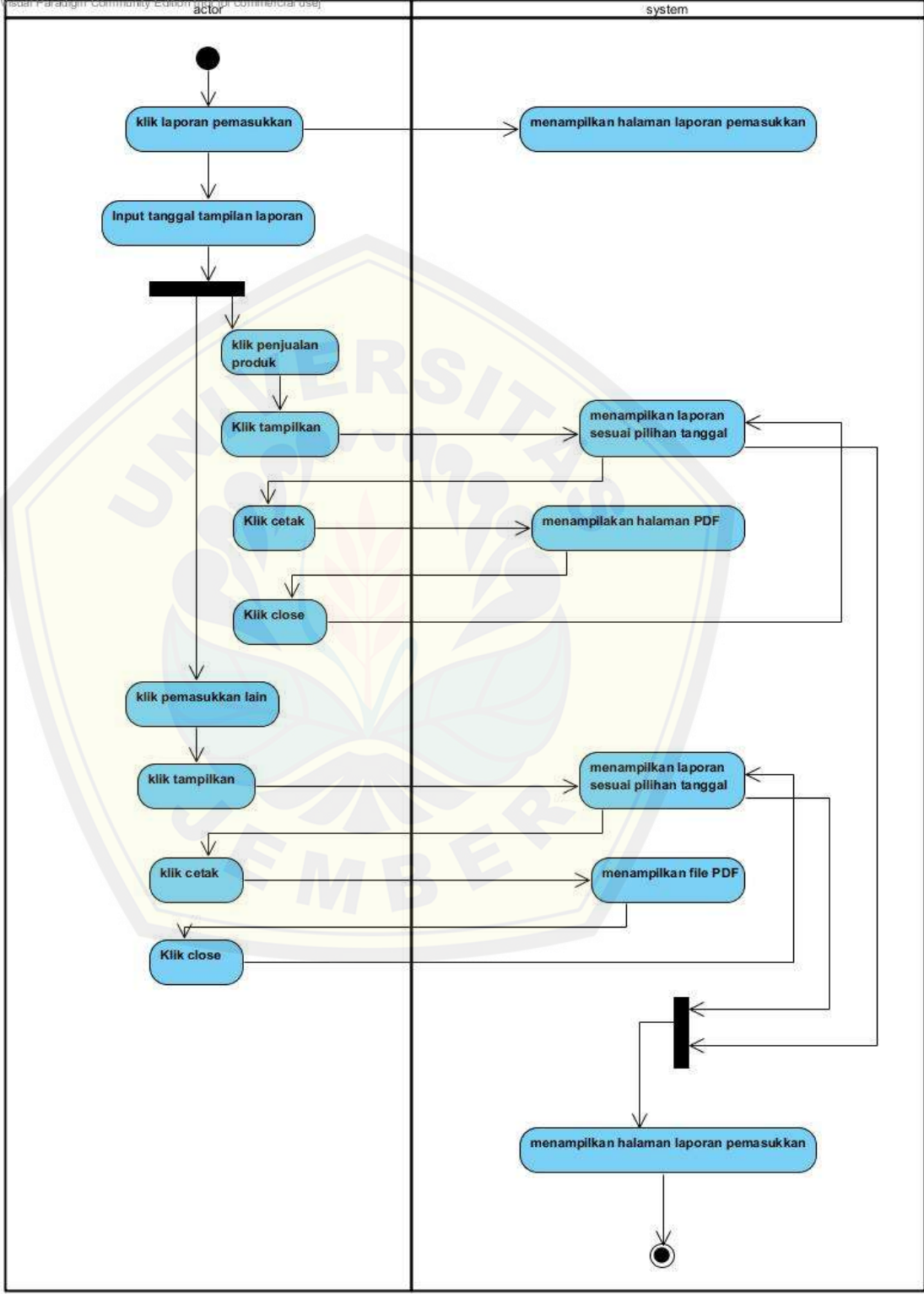
4. Activity Diagram “Menambah Pemasukan Lain”



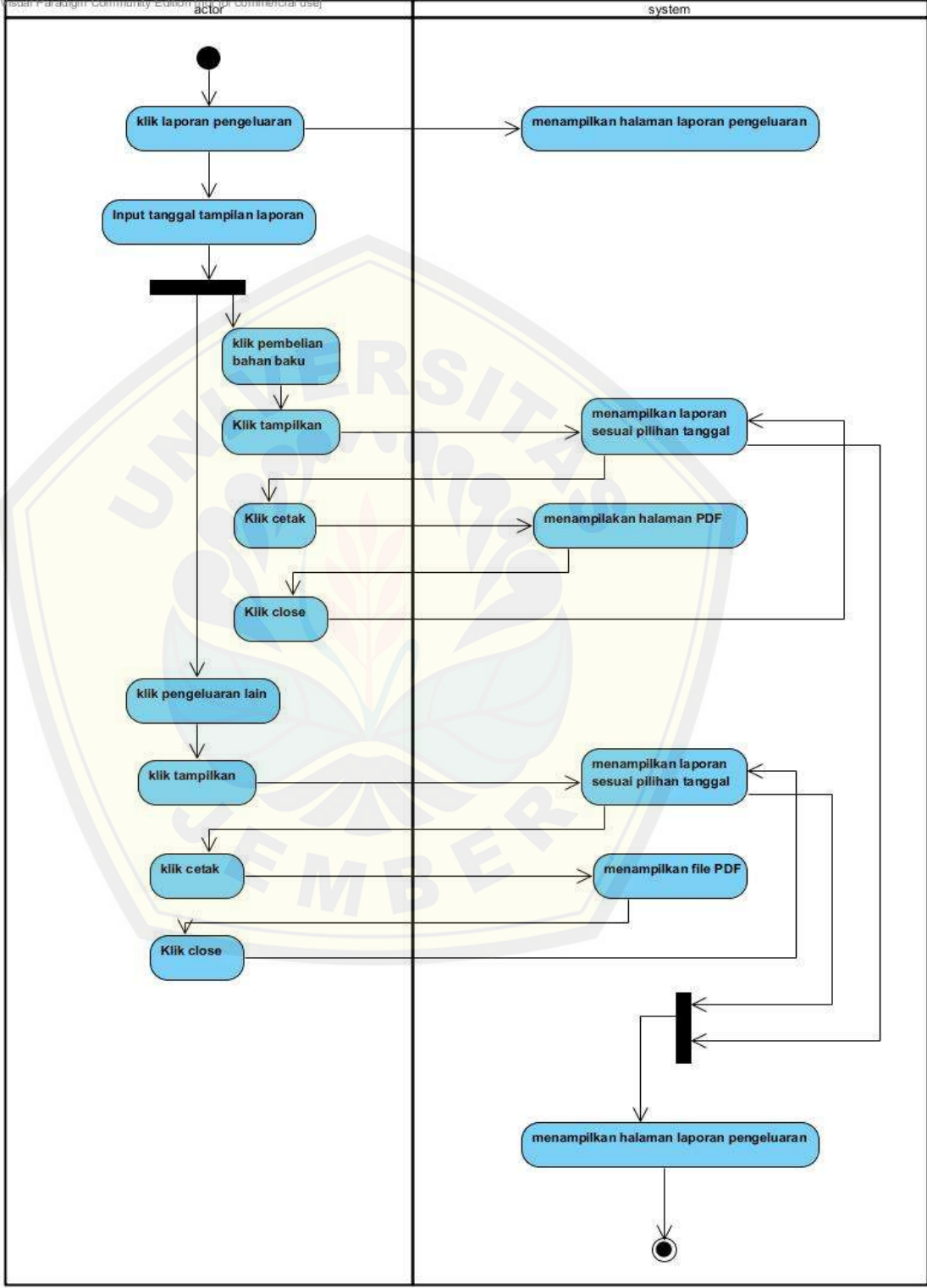
5. Activity Diagram “Menambah Pengeluaran Lain”



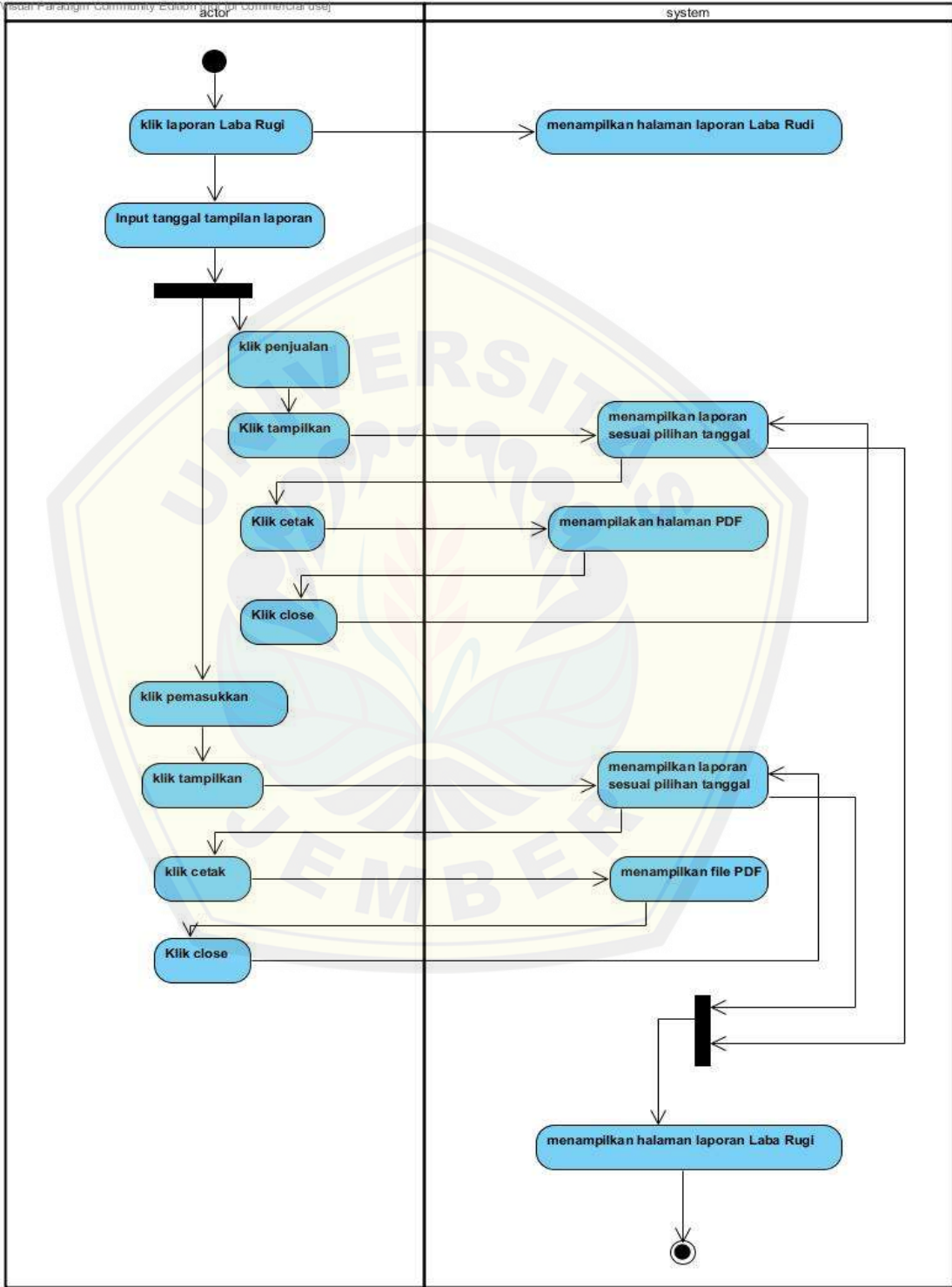
6. Activity Diagram "Menampilkan Laporan Pemasukan"



7. Activity Diagram “Menampilkan Laporan Pengeluaran”



8. Activity Diagram “Menampilkan Laporan LabaRugi”

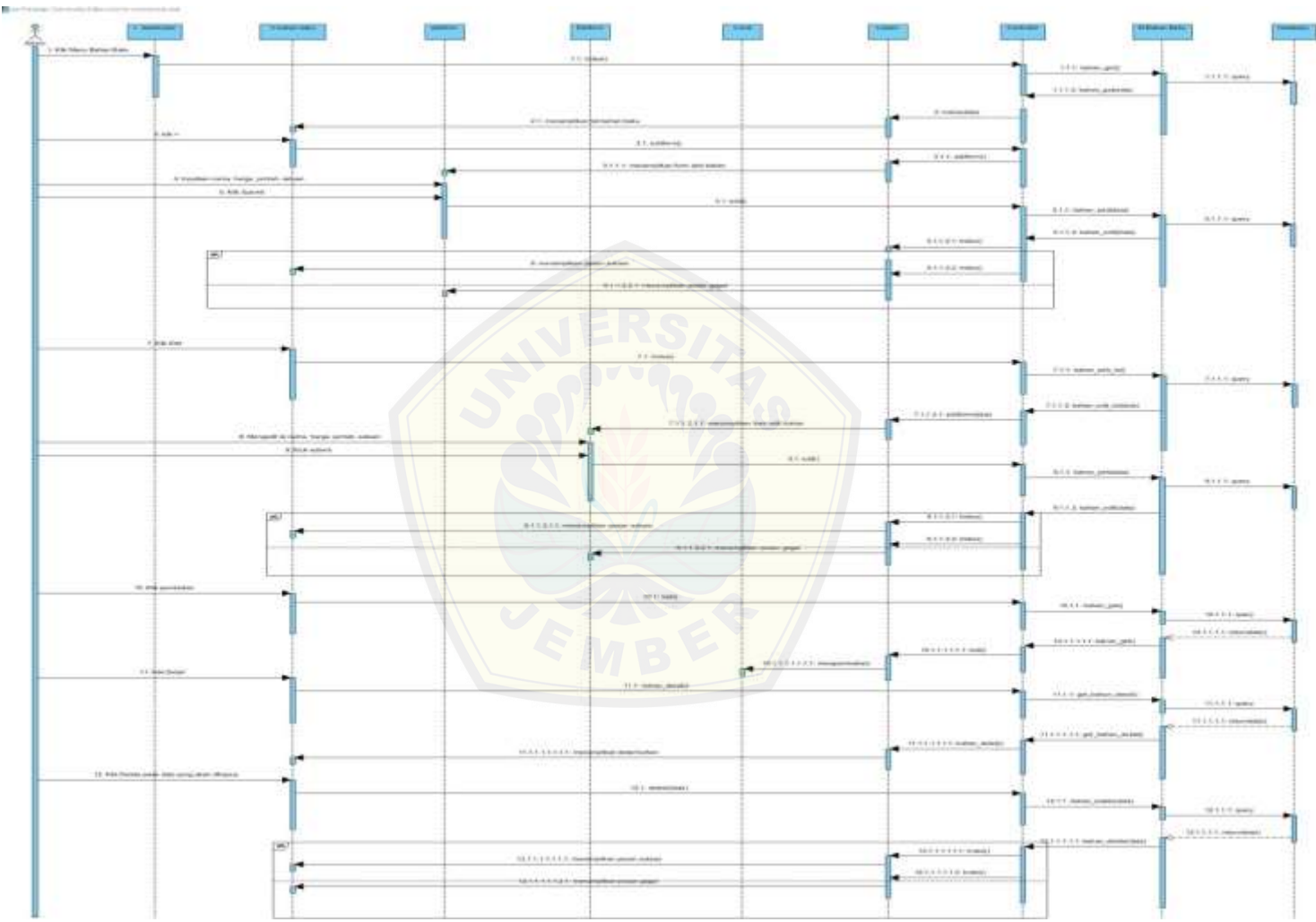




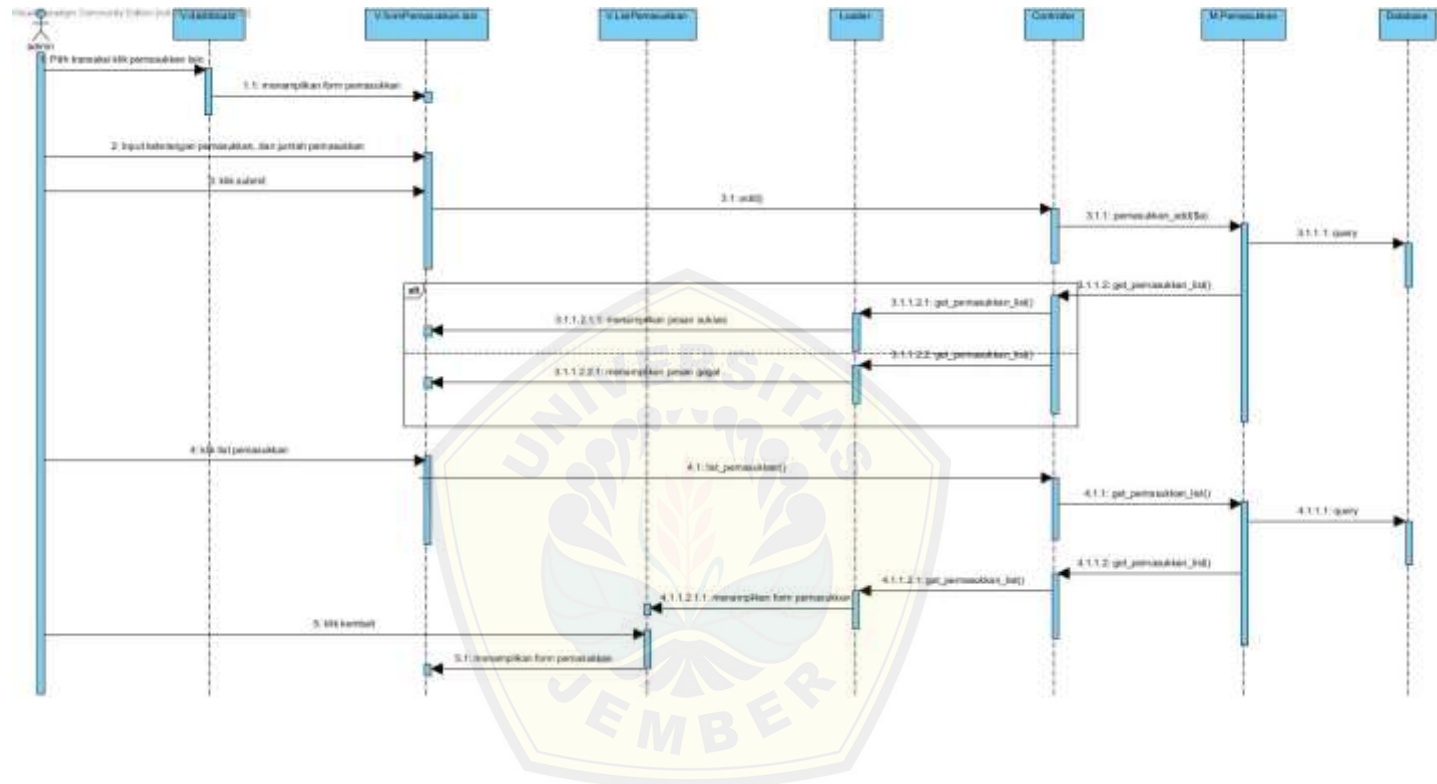




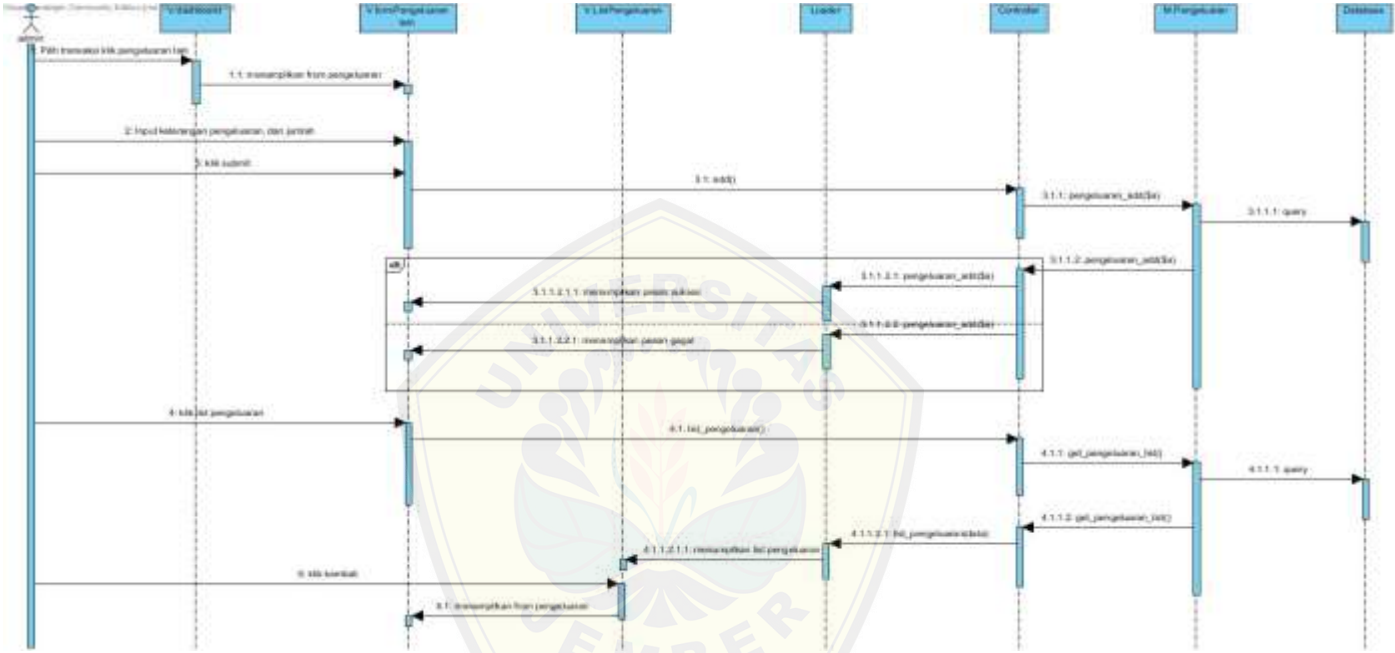
3. Sequence Diagram “Memajemen Data Bahan Baku”



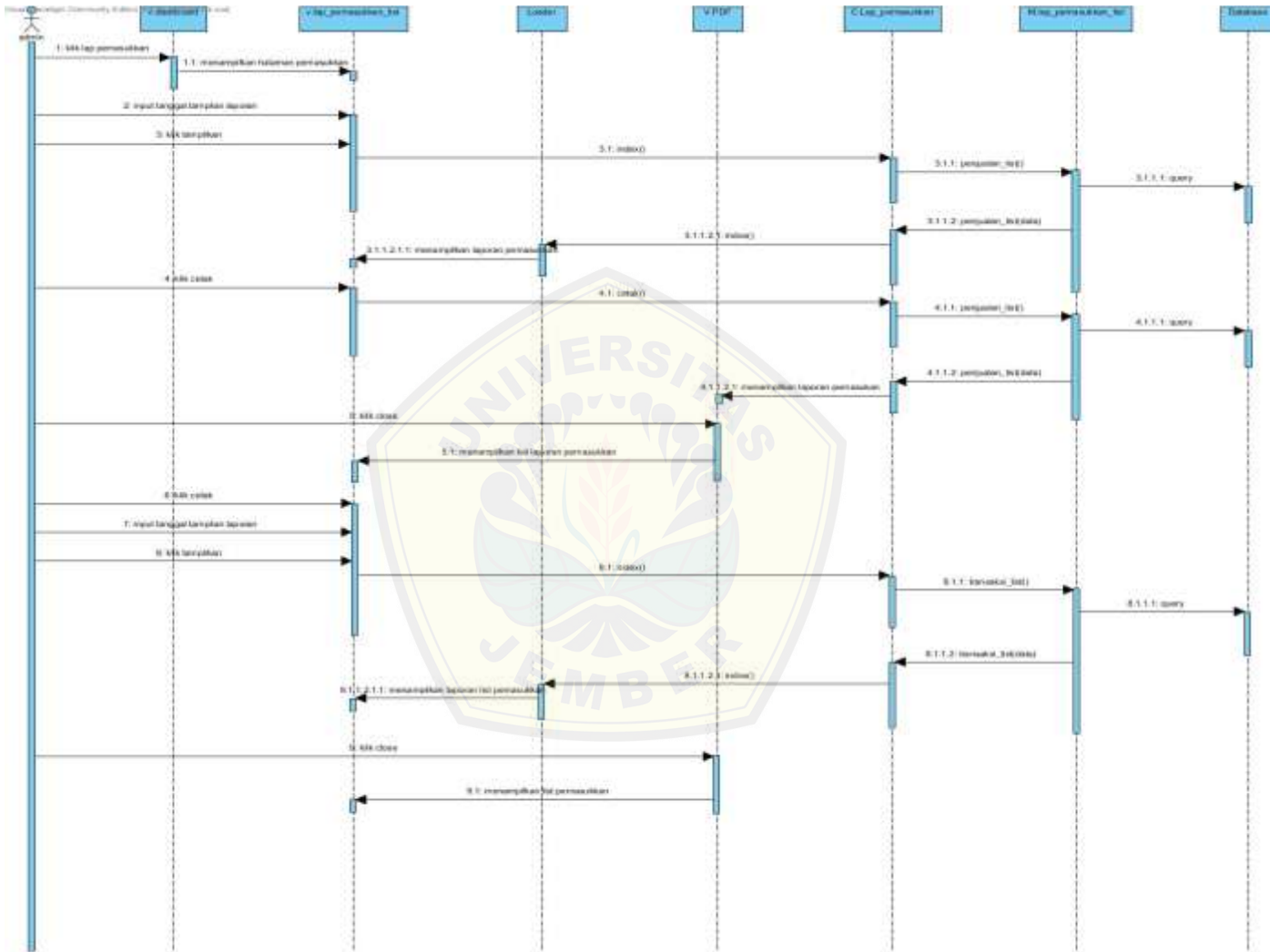
4. Sequence Diagram “Menambah Pemasukan Lain”



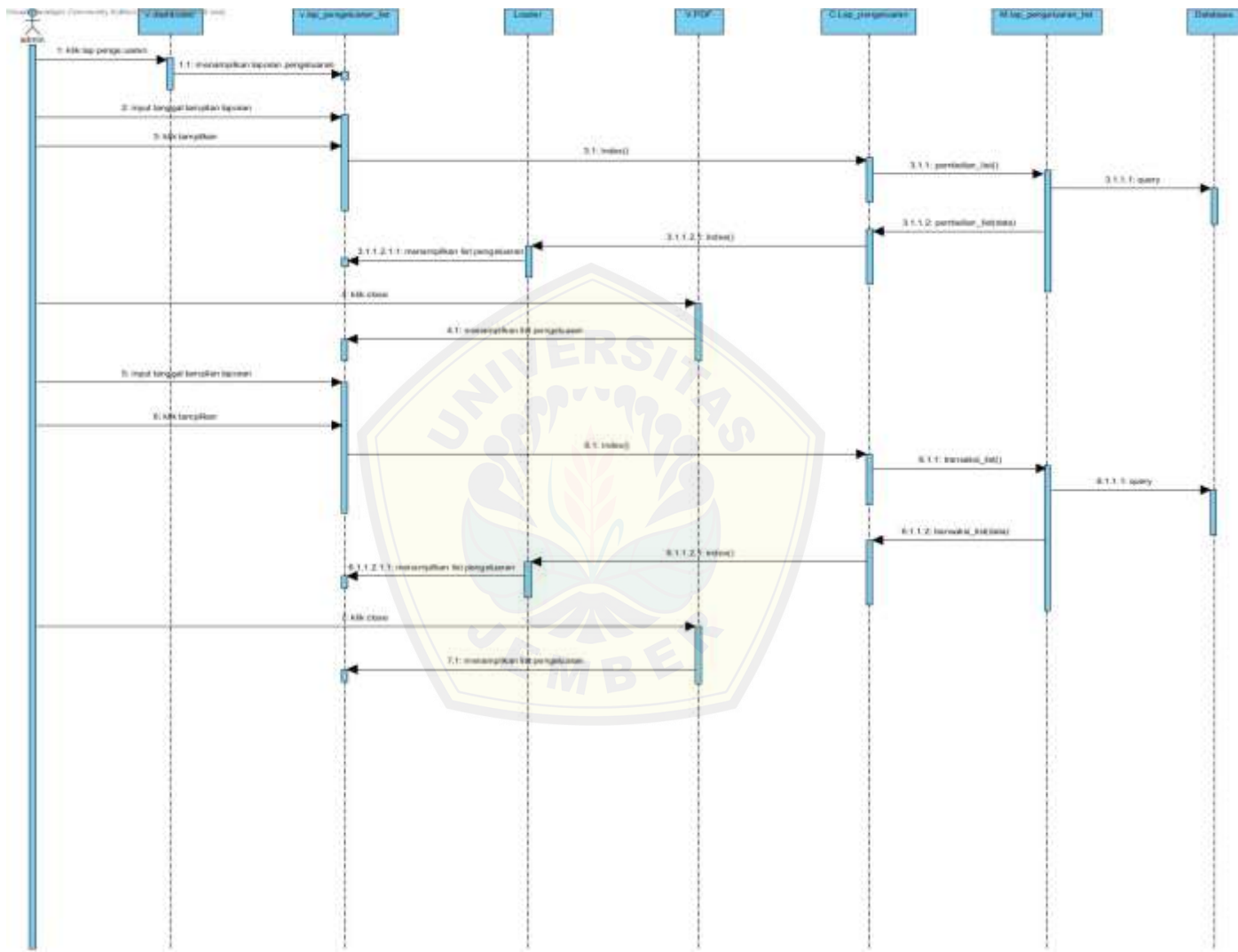
5. Sequence Diagram “Menambah Pengeluaran Lain”



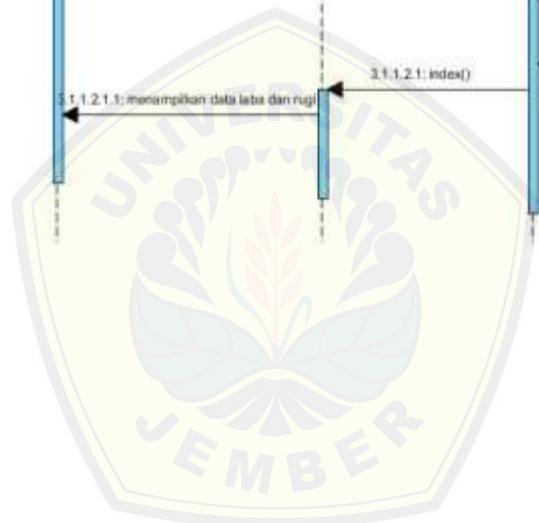
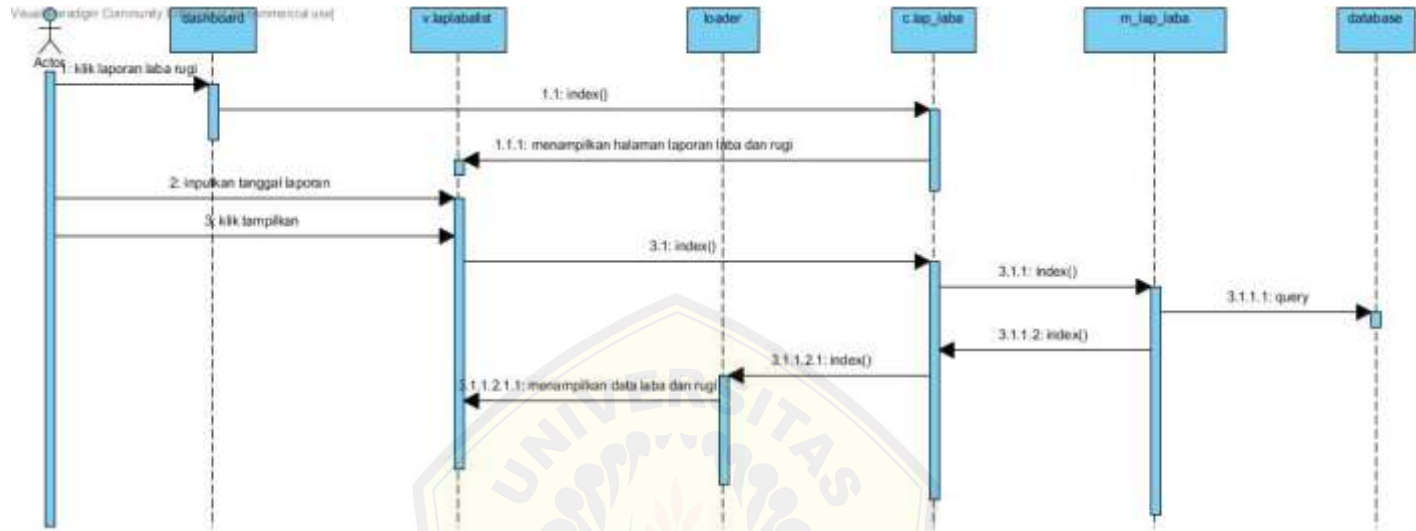
6. Sequence Diagram “Menampilkan Laporan Pemasukan”



7. Sequence Diagram "Menampilkan Laporan Pengeluaran"





8. Sequence Diagram "Menampilkan Laporan Laba Rugi"





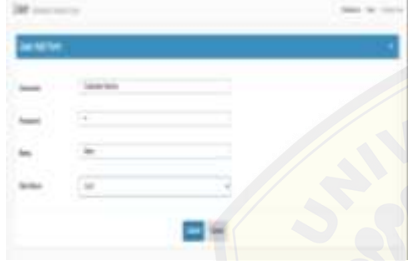





Lampiran D : Pengujian *Blackbox*



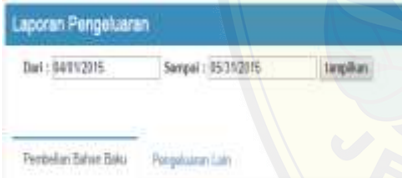

TESTING PROGRAM MENGGUNAKAN BLACK BOX TESTING  
PERANCANGAN DAN PEMBUATAN

APLIKASI AKUNTANSI UD. COBRA SAKTI BERBASIS WEB

NO	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Uji	Kesimpulan
1.	Pada fitur bahan baku system dapat melakukan add bahan baku, edit bahan baku, delete bahan baku		Sistem mampu melakukan add, edit, dan delete pada data bahan baku		Valid/Sesuai



<p>2.</p>	<p>Pada fitur produk system dapat melakukan add produk, edit produk, delete produk</p>		<p>Sistem mampu melakukan add, edit, dan delete pada data produk</p>		<p><b>Valid/Sesuai</b></p>
<p>3.</p>	<p>Pada fitur user system dapat melakukan add user, edit user, delete user</p>		<p>Sistem mampu melakukan add, edit, dan delete pada data user</p>		<p><b>Valid/Sesuai</b></p>
<p>4.</p>	<p>Pada menu pembelian system dapat menjalankan transaksi pembelian</p>		<p>Sistem dapat menjalankan transaksi pembelian</p>		<p><b>Valid/Sesuai</b></p>
<p>5.</p>	<p>Pada menu penjualan system dapat menjalankan transaksi penjualan</p>		<p>Sistem dapat menjalankan transaksi penjualan dengan memasukkan</p>		<p><b>Valid/Sesuai</b></p>

	dengan menghitung discount yang dikehendaki		discount		
6.	Pada menu laporan pemasukan system dapat menampilkan laporan pemasukan sesuai tanggal yang diinputkan		Sistem dapat menampilkan laporan pemasukan yang diminta sesuai tanggal		<b>Valid/Sesuai</b>
7.	Pada menu laporan pemasukan system dapat menampilkan laporan pemasukan sesuai tanggal yang diinputkan		Sistem dapat melakukan perubahan sesuai button yang dipilih		<b>Valid/Sesuai</b>

