

**PENGEMBANGAN FITUR AKUNTANSI *SINGLE ENTRY* PADA MULTI  
KOPERASI MENGGUNAKAN *FRAMEWORK LARAVEL***

**( Skripsi )**

**Oleh**

**Ki Agus Muhammad Ghani**



**S1 ILMU KOMPUTER  
JURUSAN ILMU KOMPUTER  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
2019**

## **ABSTRACT**

### **THE DEVELOPMENT OF ACCOUNTING FEATURES SINGLE ENTRY ON MULTI COOPERATIVES USING LARAVEL FRAMEWORK**

**By**

**KI AGUS MUHAMMAD GHANI**

The organization needs accounting to record, store, and process of financial reports and information to help the decision-making practice. There are two-known recording methods in accounting, which are single and double entry. New developing cooperation that less experienced with the accounting procedure faces the difficulties of operating the double-entry method due to limited knowledge of accounting. Therefore many cooperations use single entry methods that simpler, easier, more practical, and less professional.

The Single Entry of Accounting Information System (AIS) on multi cooperatives provide the solution for the problem above. AIS is developed using Laravel 5.5 and Software as a Service (SaaS) model with a modest display and understandable. AIS has the accounting and reporting features that comply with accounting rules to help cooperatives draft the financial report.

**Keywords:** Accounting information system, Single entry, Cooperative, Laravel framework

## ABSTRAK

### PENGEMBANGAN FITUR AKUNTANSI *SINGLE ENTRY* PADA MULTI KOPERASI MENGGUNAKAN *FRAMEWORK LARAVEL*

Oleh

**KI AGUS MUHAMMAD GHANI**

Akuntansi dibutuhkan oleh organisasi untuk mencatat, menyimpan dan memproses data sehingga menghasilkan laporan keuangan dan informasi untuk membantu pengambilan keputusan. Ada dua metode dalam melakukan pencatatan data didalam akuntansi, yaitu *single entry* dan *double entry*. Bagi koperasi yang baru berkembang dan belum mengerti kaidah akuntansi yang mendalam akan kesulitan untuk mengoperasikan metode *double entry* karena keterbatasan ilmu akuntansi. Sehingga masih banyak koperasi yang menggunakan metode *single entry* karena sederhana, mudah, praktis, dan tidak terlalu memikirkan kaidah akuntansi secara profesional.

Sistem Informasi Akuntansi (SIA) *Single Entry* pada multi koperasi hadir sebagai solusi dari masalah diatas. SIA dikembangkan menggunakan Laravel 5.5 dan dengan model *Software as a Service* (SaaS) dengan tampilan yang sederhana dan mudah dipahami. SIA sudah memiliki fitur pembukuan dan pelaporan keuangan sesuai kaidah akuntansi sehingga akan lebih mudah digunakan dan dapat memudahkan koperasi dalam membuat laporan keuangan.

**Kata Kunci:** Sistem informasi akuntansi, *Single entry*, Koperasi, *Framework* Laravel

**PENGEMBANGAN FITUR AKUNTANSI *SINGLE ENTRY* PADA MULTI  
KOPERASI MENGGUNAKAN *FRAMEWORK* LARAVEL**

Oleh

**KI AGUS MUHAMMAD GHANI**

Skripsi

Sebagai Salah Satu Syarat untuk Memperoleh Gelar  
**SARJANA KOMPUTER**

Pada

**Jurusan Ilmu Komputer  
Fakultas Matematika dan Ilmu Pengetahuan Alam**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2019**

Judul Skripsi

: PENGEMBANGAN FITUR AKUNTANSI  
*SINGLE ENTRY* PADA MULTI KOPERASI  
MENGUNAKAN *FRAMEWORK* LARAVEL

Nama Mahasiswa

: Ki Agus Muhammad Ghani

Nomor Pokok Mahasiswa

: 1517051027

Jurusan

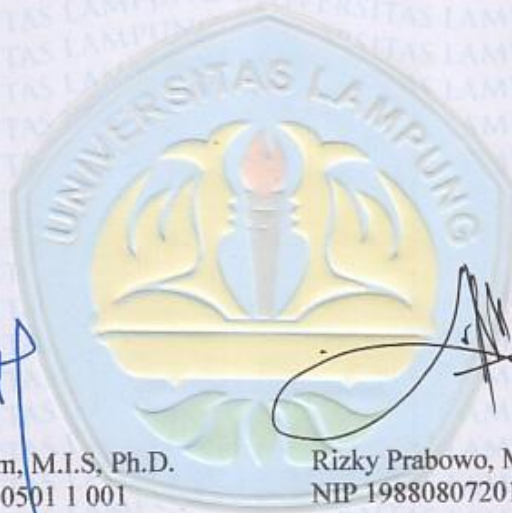
: Ilmu Komputer

Fakultas

: Matematika dan Ilmu Pengetahuan Alam

Menyetujui

1. Komisi Pembimbing



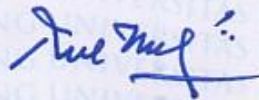


Tristiyanto, S.Kom, M.I.S, Ph.D.  
NIP 19810414 200501 1 001



Rizky Prabowo, M.Kom.  
NIP 19880807201903 1 011

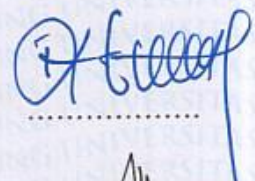
2. Ketua Jurusan Ilmu Komputer

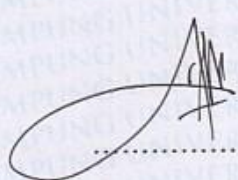


Dr. Ir. Kurnia Muludi, M.S.Sc.  
NIP 19640616 198902 1 001

**MENGESAHKAN**

1. Tim Penguji

Ketua Penguji : Tristiyanto, S.Kom, M.I.S, Ph.D. 

Sekretaris Penguji : Rizky Prabowo, M.Kom. 

Penguji Utama  
Bukan Pembimbing : Astria Hijriani, S.Kom., M.Kom. 



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



  
Drs. Suratman, M.Sc.  
NIP 19640604199003 1 002

Tanggal Lulus Ujian Skripsi : 15 November 2019

## PERNYATAAN

Saya yang bertanda tangan dibawah ini, menyatakan bahwa skripsi saya yang berjudul "Pengembangan Fitur Akuntansi *Single Entry* Pada Multi Koperasi Menggunakan *Framework* Laravel" merupakan hasil karya saya sendiri dan bukan karya orang lain. Semua tulisan yang tertuang di skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila dikemudian hari terbukti skripsi saya merupakan hasil penjiplakan atau dibuat orang lain, maka saya bersedia menerima sanksi berupa pencabutan gelar yang telah saya terima.

Bandar Lampung, 15 Nov 2019



Ki Agus Muhammad Ghani

NPM. 1517051027

## RIWAYAT HIDUP



Penulis dilahirkan pada 23 Febuari 1997 di Bandar Lampung, sebagai anak tunggal dengan Ayah bernama Kiagus Muhammad Azwar dan Ibu bernama Sundari.

Penulis menyelesaikan pendidikan Sekolah Dasar (SD) di SD N 2 Rawa Laut (Teladan) pada tahun 2009, menyelesaikan Sekolah Menengah Pertama (SMP) di SMP N 23 Bandar Lampung pada tahun 2012. Kemudian melanjutkan jenjang Sekolah Menengah Kejuruan (SMK) di SMK N 1 Bandar Lampung dan lulus pada tahun 2015.

Pada tahun 2015, penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung. Pada Bulan Februari 2018 penulis melakukan Kerja Praktik di Kantor Notaris dan PPAT Erista, S.H. Pada bulan Juli 2018 penulis melaksanakan Kuliah Kerja Nyata Kebangsaan di Desa Umbar, Kecamatan Kelumbayan, Kabupaten Tanggamus.



## **PERSEMBAHAN**

*Puji dan syukur saya ucapkan kepada Allah Subhanahu wa ta'ala atas segala nikmat dan karunia-Nya sehingga skripsi ini dapat diselesaikan.*

*Teruntuk Bapak dan Mamah yang sangat kucintai, kupersembahkan skripsi ini. Terimakasih untuk kasih sayang, kesabaran, perhatian, pengorbanan, usaha, dukungan moril maupun materi, dan do'a-do'a yang tiada henti serta untuk keluarga besar yang selalu mendukung.*

*Teruntuk sahabat dan teman-teman, terimakasih untuk canda, tawa, tangis, dan perjuangan yang telah kita lewati bersama dan terimakasih untuk setiap kenangan yang telah terukir selama ini*

*Keluarga Ilmu Komputer 2015*

*dan, Almamater yang kubanggakan*

**UNIVERSITAS LAMPUNG**

## **MOTTO**

*"Siapa yang menempuh jalan untuk mencari ilmu, maka Allah akan memudahkan baginya jalan menuju surga."*

(HR. Muslim)

*"Allah tidak membebani seseorang melainkan sesuai kesanggupannya"*

(Q.S Al-Baqarah: 286)

*"Hai orang-orang yang beriman, jadikanlah sabar dan shalat sebagai penolongmu, sesungguhnya Allah beserta orang-orang yang sabar."*

(Q.S. Al-Baqarah: 153)

## SANWACANA

Alhamdulillah, segala puji hanya milik Allah Subhanahu wa ta'ala yang telah melimpahkan rahmat, hidayah, kesehatan serta karunia-Nya sehingga penulis dapat menyelesaikan penulisan skripsi yang berjudul “Pengembangan Fitur Akuntansi *Single Entry* Pada Multi Koperasi Menggunakan *Framework* Laravel” dengan baik.

Terima kasih penulis ucapkan kepada semua pihak yang telah membantu dan berperan besar dalam penyusunan skripsi ini, seperti antara lain:

1. Kedua orang tua tercinta, Bapak dan Mama serta Keluarga yang selalu memberi doa, motivasi, dan kasih sayang yang tak terhingga.
2. Bapak Tristiyanto, S.Kom, M.I.S, Ph.D., sebagai pembimbing utama, yang telah membimbing serta memberikan kritik dan saran selama masa perkuliahan dan penyusunan skripsi sehingga penulis bisa sampai ditahap ini.
3. Bapak Rizky Prabowo, M.Kom., sebagai pembimbing II, yang telah membimbing serta memberikan kritik dan saran selama masa perkuliahan dan penyusunan skripsi sehingga penulis bisa sampai ditahap ini.
4. Ibu Astria Hijriani, S.Kom., M.Kom., sebagai pembahas utama, yang telah memberikan komentar dan masukan yang bermanfaat untuk perbaikan dalam penyusunan skripsi ini.

5. Bapak Febi Eka Febriansyah, S.T., sebagai pembimbing akademik yang telah membimbing, memotivasi serta memberikan, kritik dan saran selama masa perkuliahan.
6. Bapak Drs. Suratman, M.Sc., selaku Dekan FMIPA Universitas Lampung.
7. Bapak Dr. Ir. Kurnia Muludi, M.S.Sc., selaku Ketua Jurusan Ilmu Komputer FMIPA Universitas Lampung.
8. Bapak Didik Kurniawan, S.Si., M.T., selaku Sekretaris Jurusan Ilmu Komputer FMIPA Universitas Lampung.
9. Bapak dan Ibu Dosen Jurusan Ilmu Komputer yang telah memberikan ilmu dan pengalaman hidup selama penulis menjadi mahasiswa.
10. Ibu Ade Nora Maela yang telah membantu segala urusan administrasi penulis di Jurusan Ilmu Komputer.
11. Sahabat shalih yang selalu mengingatkan dan menyemangatkan kejalan yang benar.
12. Teman seperjuangan Abnormal Dito, Remy, Fikri, Jaka, Pido, Baim, Wawai, Shendy, Elin, Puri dkk yang telah saling menyemangati dan membantu penulis selama penelitian.
13. Sahabat nongkrong Yoga, Brok, Irfan, Alletta dan teman-teman sekolahku yang lain.
14. Keluarga besar Ilmu Komputer 2015 dan HIMAKOM yang telah memberikan kenangan dan pelajaran berorganisasi selama masa perkuliahan.
15. Almamater Tercinta, Universitas Lampung yang telah memberikan penulis kesempatan untuk menempuh pendidikan perkuliahan S1 dengan baik.

## DAFTAR ISI

DAFTAR ISI.....	xixiii
DAFTAR GAMBAR .....	xv
DAFTAR TABEL.....	xx
DAFTAR KODE .....	vii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	3
1.3 Batasan Masalah.....	3
1.4 Tujuan.....	4
1.5 Manfaat.....	4
BAB II TINJAUAN PUSTAKA .....	5
2.1 Sistem Informasi Akuntansi .....	5
2.2 Akuntansi .....	5
2.3 UML ( <i>Unified Modelling Language</i> ).....	9
2.4 <i>Framework</i> Laravel .....	14
2.5 <i>Database</i> .....	14
2.6 PHP.....	15
2.7 <i>Black Box Testing</i> .....	15
2.8 Proses Bisnis Koperasi .....	16
2.9 Metode Pengembangan Sistem .....	18
2.10 <i>Expert Judgement</i> .....	19
2.11 Penelitian Terdahulu .....	20
BAB III METODOLOGI PENELITIAN .....	21
3.1 Waktu dan Tempat Penelitian .....	21
3.2 Metodologi Penelitian .....	21
3.3 Analisis Sistem.....	26
3.4 Desain Sistem.....	28
3.5 Penulisan Kode Program .....	59

3.6	Pengujian Program .....	59
3.7	Jadwal Penelitian .....	59
BAB IV HASIL DAN PEMBAHASAN .....		61
4.1	Hasil .....	61
4.2	Implementasi Sistem .....	61
4.3	Pengujian Sistem .....	78
BAB V KESIMPULAN DAN SARAN .....		85
5.1	Kesimpulan.....	85
5.2	Saran.....	85
DAFTAR PUSTAKA .....		86

## DAFTAR GAMBAR

	Halaman
Gambar 2.1 Siklus Akuntansi .....	8
Gambar 2.2 Proses Bisnis Koperasi .....	17
Gambar 2.3 Metode <i>Waterfall</i> .....	18
Gambar 3.1 Kerangka Penelitian .....	23
Gambar 3.2 <i>Use Case Diagram</i> .....	27
Gambar 3.3 <i>Activity Diagram</i> Mengelola Rekening.....	28
Gambar 3.4 <i>Activity Diagram</i> Melihat Transaksi Sebelumnya .....	29
Gambar 3.5 <i>Activity Diagram</i> Melihat Akuntansi Mitra .....	30
Gambar 3.6 <i>Activity Diagram</i> Melihat Akuntansi Pinjaman .....	31
Gambar 3.7 <i>Activity Diagram</i> Melihat Transaksi Jual Beli .....	32
Gambar 3.8 <i>Activity Diagram</i> Melihat Transaksi Sewa Mitra .....	32
Gambar 3.9 <i>Activity Diagram</i> Melihat Simpanan Mitra.....	33
Gambar 3.10 <i>Activity Diagram</i> Mengelola Rekening.....	34
Gambar 3.11 <i>Activity Diagram</i> Mengelola Jurnal .....	35
Gambar 3.12 <i>Activity Diagram</i> Melihat Periode .....	36
Gambar 3.13 <i>Activity Diagram</i> Melihat Buku Besar .....	36
Gambar 3.14 <i>Activity Diagram</i> Melihat Laporan Sisa Hasil Usaha .....	37
Gambar 3.15 <i>Activity Diagram</i> Melihat Neraca .....	38
Gambar 3.16 <i>Activity Diagram</i> Melihat Perubahan Modal .....	39
Gambar 3.17 <i>Activity Diagram</i> Melihat Arus Kas.....	40

Gambar 3.18 <i>Activity Diagram</i> Melihat Simpanan.....	40
Gambar 3.19 <i>Activity Diagram</i> Melihat Pinjaman .....	41
Gambar 3.20 <i>Activity Diagram</i> Melihat Transaksi Jual Beli.....	42
Gambar 3.21 <i>Activity Diagram</i> Melihat Transaksi Sewa .....	43
Gambar 3.22 <i>Class Diagram</i> Sistem Informasi Akuntansi.....	44
Gambar 3.23 Rancangan <i>Interface</i> Rekening .....	46
Gambar 3.24 Rancangan <i>Interface Form</i> Rekening .....	46
Gambar 3.25 Rancangan <i>Interface</i> Periode .....	47
Gambar 3.26 Rancangan <i>Interface</i> Jurnal Umum <i>Single Entry</i> .....	47
Gambar 3.27 Rancangan <i>Interface</i> Tambah Data Jurnal Umum <i>Single Entry</i> .....	48
Gambar 3.28 Rancangan <i>Interface</i> Buku Besar.....	48
Gambar 3.29 Rancangan <i>Interface</i> Sisa Hasil Usaha .....	49
Gambar 3.30 Rancangan <i>Interface</i> Laporan Sisa Hasil Usaha .....	50
Gambar 3.31 Rancangan <i>Interface</i> Perubahan Modal .....	50
Gambar 3.32 Rancangan <i>Interface</i> Laporan Perubahan Modal.....	51
Gambar 3.33 Rancangan <i>Interface</i> Neraca .....	51
Gambar 3.34 Rancangan <i>Interface</i> Laporan Neraca.....	52
Gambar 3.35 Rancangan <i>Interface</i> Arus Kas.....	52
Gambar 3.36 Rancangan <i>Interface</i> Laporan Arus Kas .....	53
Gambar 3.37 Rancangan <i>Interface</i> Akuntansi Simpanan .....	53
Gambar 3.38 Rancangan <i>Interface</i> Akuntansi Pinjaman .....	54
Gambar 3.39 Rancangan <i>Interface</i> Daftar Transaksi Jual Beli.....	54
Gambar 3.40 Rancangan <i>Interface</i> Daftar Transaksi Sewa .....	55
Gambar 3.41 Rancangan <i>Interface</i> Lihat Rekening.....	55



Gambar 3.42 Rancangan <i>Interface</i> Lihat Transaksi Sebelumnya.....	56
Gambar 3.43 Rancangan <i>Interface</i> Lihat Akuntansi Mitra.....	56
Gambar 3.44 Rancangan <i>Interface</i> Lihat Akuntansi Simpanan Mitra.....	57
Gambar 3.45 Rancangan <i>Interface</i> Lihat Akuntansi Pinjaman Mitra.....	57
Gambar 3.46 Rancangan <i>Interface</i> Lihat Daftar Transaksi Jual Beli Mitra .....	58
Gambar 3.47 Rancangan <i>Interface</i> Lihat Transaksi Sewa Mitra .....	58
Gambar 3.48 Gantt Chart .....	60
Gambar 4. 1 Halaman <i>Dashboard</i> .....	61
Gambar 4. 2 Halaman Periode .....	62
Gambar 4. 3 Halaman <i>Form</i> Tambah Periode .....	62
Gambar 4. 4 Halaman Rekening .....	63
Gambar 4. 5 Halaman <i>Form</i> Tambah Rekening .....	64
Gambar 4. 6 Halaman Jurnal Umum .....	65
Gambar 4. 7 Halaman <i>Form</i> Tambah Jurnal Umum .....	66
Gambar 4. 8 Halaman Buku Besar .....	67
Gambar 4. 9 Halaman Sisa Hasil Usaha .....	68
Gambar 4. 10 Halaman Perubahan Modal .....	69
Gambar 4. 11 Halaman Neraca .....	70
Gambar 4. 12 Halaman Arus Kas .....	71
Gambar 4. 13 Halaman Akuntansi Simpanan.....	72
Gambar 4. 14 Halaman Akuntansi Pinjaman.....	73
Gambar 4. 15 Halaman Rekening Super Admin.....	74
Gambar 4. 16 Halaman Lihat Transaksi Sebelumnya.....	74
Gambar 4. 17 Halaman Lihat Akuntansi Mitra.....	75

Gambar 4.18 Halaman Lihat Akuntansi Simpanan Mitra.....	76
Gambar 4. 19 Halaman Lihat Akuntansi Pinjaman Mitra .....	77

## DAFTAR TABEL

	Halaman
Tabel 2.1 Tabel Penjelasan <i>Use Case Diagram</i> .....	10
Tabel 2.2 Tabel <i>Activity Diagram</i> .....	11
Tabel 2.3 Tabel <i>Sequence Diagram</i> .....	12
Tabel 2.4 Tabel <i>Class Diagram</i> .....	13
Tabel 3.1 Perbedaan <i>Single Entry</i> dan <i>Double Entry</i> .....	24
Tabel 4. 1 Pengujian pada Admin .....	78
Tabel 4. 2 Pengujian pada Super Admin .....	83

## DAFTAR KODE

	Halaman
Kode 4. 1 Potongan Kode Program Halaman <i>Dashboard</i> .....	62
Kode 4. 2 Potongan Kode Program Halaman Periode.....	62
Kode 4. 3 Potongan Kode Program Halaman <i>Form</i> Tambah Periode.....	63
Kode 4. 4 Potongan Kode Program Halaman Rekening.....	64
Kode 4. 5 Potongan Kode Program Halaman <i>Form</i> Tambah Rekening.....	65
Kode 4. 6 Potongan Kode Program Halaman Jurnal Umum .....	65
Kode 4. 7 Potongan Kode Program Halaman <i>Form</i> Tambah Jurnal Umum .....	66
Kode 4. 8 Potongan Kode Program Halaman Buku Besar .....	67
Kode 4. 9 Potongan Kode Program Halaman Sisa Hasil Usaha.....	68
Kode 4. 10 Potongan Kode Program Halaman Perubahan Modal.....	69
Kode 4. 11 Potongan Kode Program Halaman Neraca.....	70
Kode 4. 12 Potongan Kode Program Halaman Arus Kas .....	71
Kode 4. 13 Potongan Kode Program Halaman Akuntansi Simpanan .....	72
Kode 4. 14 Potongan Kode Program Halaman Akuntansi Pinjaman .....	73
Kode 4. 15 Potongan Kode Program Halaman Rekening Super Admin .....	74
Kode 4. 16 Potongan Kode Program Lihat Transaksi Sebelumnya.....	75
Kode 4. 17 Potongan Kode Program Lihat Akuntansi Mitra.....	75
Kode 4. 18 Potongan Kode Program Lihat Akuntansi Simpanan Mitra.....	76
Kode 4. 19 Potongan Kode Program Lihat Akuntansi Pinjaman Mitra .....	77

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Akuntansi dibutuhkan oleh instansi untuk mencatat, menyimpan dan memproses data sehingga menghasilkan informasi untuk membantu pengambilan keputusan (Romney dan Steinbart, 2016). Akuntansi juga membantu instansi untuk menyediakan informasi pelaporan keuangan dengan kualitas yang tinggi, karena secara tidak langsung akan mempengaruhi penyedia modal dan pemangku kepentingan lainnya dalam melakukan investasi, kredit dan pengambilan keputusan (Soyinka dkk, 2017) (IASB, 2008). Dengan adanya akuntansi, proses bisnis yang memiliki banyak transaksi keuangannya dapat dengan mudah diolah untuk dijadikan laporan keuangan. Penyesuaian dan pencatatan bisnis yang baik akan membantu meningkatkan efisiensi dan produktivitas bisnis untuk kinerja bisnis yang efektif dan berguna untuk meningkatkan peluang operasi bisnis agar mencapai kesuksesan (Rahamon dan Adejare, 2014).

Dalam proses bisnisnya, akuntansi memiliki dua metode dalam melakukan pencatatan data yaitu *single entry* dan *double entry* (Rahamon dan Adejare, 2014). *Single entry* adalah metode pencatatan transaksi yang dilakukan dengan pencatatan satu kali. Transaksi yang berakibat bertambahnya kas akan dicatat pada sisi penerimaan sedangkan transaksi yang berakibat berkurangnya kas akan dicatat pada sisi pengeluaran (Halim, 2010), sedangkan *double entry* merupakan

metode pencatatan transaksi dengan selalu melibatkan minimal dua akun yang setiap akunya berubah-ubah (Prasetyo, 2013).

Bagi koperasi yang baru berkembang dan belum mengerti kaidah akuntansi yang mendalam masih sangat kesulitan untuk mengoperasikan metode *double entry*, sehingga masih banyak koperasi yang menggunakan metode *single entry* karena sederhana, mudah, praktis, dan tidak terlalu memikirkan kaidah akuntansi secara profesional (Ruza dkk, 2015). Oleh karena itu, metode *single entry* masih menjadi pilihan pengelola koperasi terutama menengah kebawah dan baru yang proses bisnisnya masih sedikit dan tidak memahami kaidah akuntansi namun tetap dapat melakukan pembukuan keuangan. Dengan proses bisnis yang sedikit maka instansi dapat menggunakan sistem akuntansi yang sederhana.

Sistem informasi akuntansi adalah kumpulan data dan prosedur pengolahan yang menciptakan informasi yang diperlukan untuk penggunaannya (Nancy dkk, 2009). Sistem informasi akuntansi dapat digunakan koperasi untuk melakukan pencatatan dan semua pelaporan keuangan dengan cepat, baik, mudah dan dapat membantu dalam pengambilan keputusan dan kontrol terhadap koperasi. Sistem informasi akuntansi juga sangat membantu bagi instansi skala kecil maupun menengah karena akuntabilitas yang baik akan memudahkan instansi dalam dalam pengambilan keputusan dan melakukan proses bisnisnya.

Sistem informasi multi koperasi yang telah dibangun Iman (Iman, 2018) terdiri dari beberapa modul gabungan seperti penjualan, pinjaman, simpanan, sewa dan akuntansi yang hanya menggunakan metode akuntansi *double entry* yang dikembangkan oleh Rahmanda (Rahmanda, 2018). Namun, masih banyak

koperasi yang masih menggunakan metode *single entry* karena sederhana, mudah, praktis dan tidak memikirkan kaidah akuntansi secara profesional (Ne dkk, 2015). Oleh karena itu pengembangan fitur akuntansi dengan metode *single entry* perlu diterapkan untuk memenuhi kebutuhan koperasi yang menggunakan metode *single entry* dan membantu proses bisnis akuntansi mulai dari pengolahan data (Tungadi, 2013), pencatatan, hingga laporan keuangan di koperasi tersebut. Diharapkan dengan adanya pengembangan fitur akuntansi *single entry* pada sistem informasi multi koperasi dapat digunakan oleh seluruh kalangan koperasi di Indonesia.

## **1.2 Rumusan Masalah**

Adapun rumusan masalah pada penelitian ini sebagai berikut.

1. bagaimana langkah-langkah mengembangkan fitur akuntansi *single entry* pada sistem informasi akuntansi multi koperasi?,
2. apakah dengan menggunakan fitur akuntansi *single entry* yang *user friendly* dapat memudahkan koperasi?,
3. apakah dengan menggunakan fitur akuntansi *single entry* dapat digunakan untuk membantu mencetak laporan keuangan pada koperasi?.

## **1.3 Batasan Masalah**

Adapun batasan masalah pada penelitian ini sebagai berikut.

1. pembuatan sistem menggunakan *framework* Laravel,
2. sistem yang dibuat adalah sistem *single entry*,
3. pembuatan laporan keuangan koperasi yang meliputi laba/rugi, neraca, arus kas dan perubahan modal.

#### **1.4 Tujuan**

Adapun tujuan dari penelitian ini sebagai berikut.

1. untuk membangun fitur *single entry* yang dapat mencetak laporan keuangan meliputi laba/rugi, perubahan modal, neraca dan arus kas,
2. untuk mengetahui apakah fitur *single entry* dapat mencatat seluruh transaksi dengan baik,
3. untuk membangun sistem informasi akuntansi yang *user friendly* bagi koperasi yang tidak paham kaidah akuntansi yang mendalam.

#### **1.5 Manfaat**

Adapun manfaat dari penelitian ini sebagai berikut.

1. menghasilkan fitur *single entry* yang sesuai kebutuhan koperasi mulai dari koperasi yang kecil, berkembang dan belum memahami akuntansi,
2. meminimalisir kesalahan manusia dan memudahkan dalam pengoperasian,
3. menghasilkan laporan keuangan lengkap dari proses bisnis yang berjalan,
4. dapat dengan mudah dipahami oleh pengguna SIA multi koperasi.



## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Sistem Informasi Akuntansi**

Sistem informasi akuntansi adalah kumpulan data mentah dan disimpan (bersama-sama biasanya berfungsi sebagai *input*), metode pemrosesan (biasanya disebut " prosedur "), dan informasi (*output*) untuk tujuan akuntansi yang berguna (Nancy dkk, 2009).

Sistem informasi akuntansi (SIA) pada dasarnya merupakan integrasi dari berbagai sistem pengolahan transaksi (SPT) atau sub SIA. Karena setiap SPT memiliki siklus pengolahan transaksi maka SIA juga dapat dikatakan sebagai integrasi dari berbagai siklus pengolahan transaksi. Dalam setiap pengolahan transaksi yang dilakukannya, SPT atau sub SIA menggunakan berbagai komponen yang dimilikinya seperti *hardware*, *software*, *brainware*, prosedur, *database* dan jaringan komunikasi (Susanto, 2017).

#### **2.2 Akuntansi**

Akuntansi bisa dikatakan salah satu bagian terpenting dalam suatu instansi. Akuntansi menghasilkan informasi yang berguna bagi pihak-pihak yang terlibat untuk pengambilan keputusan. Akuntansi juga menyediakan informasi kuantitatif terutama yang bersifat keuangan.

Susanto (2017:65) menjelaskan bahwa transaksi bisnis atau akuntansi pada dasarnya merupakan penyeleksian semua aktivitas ekonomi kedalam aktivitas yang hanya berkaitan dengan suatu organisasi perusahaan, aktivitas ekonomi lainnya yang tidak berkaitan dikesampingkan. Transaksi bisnis tersebut kemudian dimasukan kedalam suatu formulir (kertas atau layar komputer) sehingga menjadi dokumen atau data untuk selanjutnya diolah menjadi informasi atau informasi akuntansi.

### 1. Sistem Pencatatan

Ada dua tipe dasar metode pembukuan: *single entry* dan *double entry* (Rahamon dan Adejare, 2014) yaitu sebagai berikut.

#### 1) *Single Entry*

Dalam sistem ini, pencatatan transaksi ekonomi dilakukan dengan mencatatnya satu kali. Transaksi yang berakibat bertambahnya kas akan dicatat pada sisi Penerimaan dan transaksi yang berakibat berkurangnya kas akan dicatat pada sisi Pengeluaran.

#### 2) *Double Entry*

Menurut sistem ini, pada dasarnya suatu transaksi ekonomi akan dicatat dua kali (*double* = berpasangan/ganda, *entry* = pencatatan). Pencatatan dengan sistem ini disebut dengan istilah menjurnal. Dalam pencatatan tersebut ada sisi Debit dan Kredit (Halim, 2010).

*Single entry* digunakan karena lebih mudah dipahami dan dioperasikan orang awam sedangkan *double entry* lebih auditabel dan biasa digunakan untuk instansi pemerintahan yang perlu keterbukaan namun untuk mengoperasikannya dibutuhkan pemahaman akuntansi yang lebih mendalam.

## 2. Buku Besar

Perkiraan merupakan catatan untuk menampung suatu transaksi yang dicatat lewat jurnal. Sekumpulan perkiraan biasa disebut dengan istilah buku besar atau *ledger*. Perkiraan-perkiraan tersebut biasanya diberi nomor dan dikelompokkan berdasarkan kemunculannya di neraca atau laba rugi. (Syarief, 2009).

## 3. Jurnal

Jurnal adalah pencatatan akuntansi yang pertama kali dilakukan. Pencatatan di dalam jurnal ini biasanya lebih lengkap dan lebih mendetail dengan melakukan pengurutan yang sesuai tanggal kejadian transaksi. Jurnal merupakan catatan akuntansi permanen yang pertama digunakan untuk mencatat transaksi keuangan (Romney dan Steinbart, 2016). Transaksi-transaksi keuangan yang terjadi di dalam suatu perusahaan perlu dicatat. Catatan untuk mencatat transaksi tersebut disebut jurnal. Jadi, jurnal merupakan media pencatatan sebelum transaksi tersebut dikelompokkan. (HD, 2009). Dengan adanya jurnal di SIA diharapkan tidak adanya transaksi yang tidak di catat.

## 4. Siklus Akuntansi

Siklus akuntansi adalah siklus keuangan organisasi, yang dimulai dengan data transaksi (misalnya, diambil di titik penjualan) dan berakhir dengan laporan keuangan periodiknya (Bagranoff dkk, 2009). Kemudian transaksi ini memiliki bukti yang sah sebagai dasar terjadinya transaksi kemudian berdasarkan data atau bukti ini, lalu diinput keproses pengolahan data

sehingga menghasilkan *output* berupa informasi laporan keuangan (Harahap, 2007). Siklus akuntansi secara sederhana dapat dilihat pada gambar 2.1.



Gambar 2.1 Siklus Akuntansi.

- Tahap pertama yaitu melakukan pencatatan transaksi.
- Tahap kedua yaitu pemilihan akun.
- Tahap ketiga yaitu pencatatan di jurnal.
- Tahap keempat yaitu melakukan penggolongan antara debit dan kredit di buku besar.
- Tahap keempat yaitu pelaporan mulai dari perubahan modal, laba/rugi, neraca dan arus kas.

## 5. Laporan

Laporan adalah *output* dari sistem yang disusun dengan sedemikian rupa sehingga menjadi urutan yang bermakna, yang kemudian digunakan oleh pegawai untuk mengendalikan aktivitas operasional, manajer untuk pengambilan keputusan dan mendesain strategi bisnis, dan bagian *investor* dan *kreditor* untuk melihat aktivitas bisnis di suatu organisasi (Romney dan Steinbart, 2016).

Pelaporan keuangan dengan kualitas yang tinggi sangat begitu penting karena secara tidak langsung akan mempengaruhi penyedia modal dan pemangku kepentingan lainnya dalam melakukan investasi, kredit dan pengambilan keputusan agar meningkatkan efisiensi pasar secara keseluruhan (Soyinka dkk, 2017).

### **2.3 UML (*Unified Modelling Language*)**


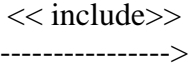
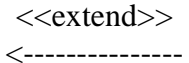



Pemodelan adalah perancangan aplikasi perangkat lunak yang digunakan sebelum pengkodean. Pemodelan adalah bagian penting dari proyek perangkat lunak yang memiliki skala yang besar, dan membantu untuk proyek-proyek menengah dan bahkan kecil juga. Dengan menggunakan pemodelan, mereka akan bertanggung jawab atas keberhasilan dari proyek pengembangan perangkat lunak dan dapat memastikan bahwa fungsi bisnis sudah lengkap dan benar, kebutuhan pengguna akhir terpenuhi, dan desain program mendukung persyaratan untuk skalabilitas, ketahanan, keamanan, perpanjangan, dan karakteristik lainnya, sebelum memasuki implementasi dalam pembuatan kode yang sulit dan mahal untuk dibuat (Object Management Group, 2005).

UML (*Unified Modelling Language*) juga merupakan metodologi yang paling marak digunakan untuk saat ini baik analisa maupun perancangan sistem dengan metodologi berorientasi objek dengan mengadaptasi maraknya penggunaan bahasa pemrograman berorientasi objek (OOP) yang merupakan gabungan dari beberapa metode seperti Booch dll (Nugroho, 2009).

### 2.3.1 *Use Case Diagram*

*Use case* adalah suatu aktivitas yang dilakukan oleh sistem berupa respon terhadap permintaan pengguna dan hubungan antar aktor di dalam penggunaan sistem tersebut (Satzinger dkk, 2012). *Use Case* menggambarkan fungsionalitas yang disediakan oleh sistem tersebut, dan menentukan persyaratan yang ditimbulkan sistem pada lingkungannya.



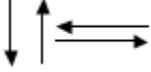



Tabel 2.1 Tabel Penjelasan *Use Case Diagram*.

NO	Gambar	Nama	Keterangan
1		<i>Actor</i>	Diartikan sebagai peran yang digunakan ketika berhubungan dengan <i>use case</i> .
2		<i>Include</i>	Diartikan bahwa <i>use case</i> sumber secara eksplisit.
3		<i>Extend</i>	Diartikan bahwa <i>use case</i> target untuk memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
4		<i>System</i>	Diartikan dengan paket yang menjadi pembatas dalam menampilkan <i>use case</i> .
5		<i>Association</i>	Penghubung antara objek satu dengan objek lainnya.
6		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor

### 2.3.1 *Activity Diagram*

*Workflow* memiliki sebuah tipe yang dinamakan *activity diagram*. *Activity diagram* digunakan untuk menggambarkan tentang aktivitas dari pengguna ketika melakukan kegiatan (Satzinger dkk, 2012).

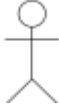

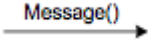



Tabel 2.2 Tabel *Activity Diagram*.

No	Gambar	Nama	Keterangan
1		<i>Activy</i>	Digunakan oleh masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2		<i>Action</i>	<i>State</i> dari sistem yang mencerminkan eksekusi dari suatu aksi.
3		<i>Line Connector</i>	Digunakan sebagai penghubung antara satu simbol dengan simbol lainnya.
4		<i>Desicion</i>	Digunakan untuk menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu.
5		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan diakhiri.
6		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.

### 2.3.2 *Sequence Diagram*

*Sequence diagram* adalah jenis diagram interaksi. *Sequence diagram* digunakan untuk menggambarkan aliran informasi ke dalam dan keluar dari sistem. Jadi, *sequence diagram* mendokumentasikan input dan output dan mengidentifikasi interaksi antara aktor dan sistem (Satzinger dkk, 2012).

Tabel 2.3 Tabel *Sequence Diagram*.






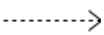
No	Gambar	Nama	Keterangan
1		<i>Actor</i>	Digunakan untuk menggambarkan user pengguna.
2		<i>Life Line</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi tentang aktifitas yang terjadi
4		<i>Boundary</i>	Digunakan untuk menggambarkan sebuah <i>form</i>
5		<i>Control Class</i>	Digunakan untuk menghubungkan <i>boundary</i> dengan tabel.
6		<i>Entity Class</i>	Digunakan untuk menggambarkan hubungan kegiatan yang akan dilakukan.

### 2.3.3 *Class Diagram*

*Class diagram* adalah gambaran struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat dalam membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi (Sukamto dan Shalahuddin, 2013).



Tabel 2.4 Tabel *Class Diagram*.

No	Gambar	Nama	Keterangan
1		<i>Class</i>	Sebuah class digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian. Bagian atas adalah bagian nama dari <i>class</i> . Bagian tengah mendefinisikan <i>property</i> /atribut class. Bagian akhir mendefinisikan <i>method-method</i> dari sebuah <i>class</i> .
2		<i>Association</i>	Sebuah asosiasi merupakan sebuah <i>relationship</i> paling umum antara 2 <i>class</i> .
3		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
4		<i>Relationship composition</i>	Jika sebuah <i>class</i> tidak bisa berdiri sendiri dan harus merupakan bagian dari <i>class</i> yang lain, maka class tersebut memiliki relasi <i>Composition</i> terhadap class tempat dia bergantung tersebut.
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.

## 2.4 Framework Laravel

Laravel merupakan *framework PHP* yang menekankan pada kesederhanaan dan fleksibilitas pada desainnya. Laravel dirilis dibawah lisensi MIT. Dengan terobosannya Laravel berusaha menjadi *framework* yang mudah digunakan oleh pengguna sehingga mengurangi beban-beban umum yang sering digunakan oleh sebagian besar proyek *web* seperti *authentication*, *routing*, *session* dan *caching* (Laravel, 2011).

*Framework* Laravel di buat oleh Taylor Otwel pada April 2011. Latar belakang dibuatnya Laravel karena Otwel tidak menemukan *framework* yang up-to-date dengan versi *PHP* (Aminuddin, 2015).

Adapun kelebihan dari Laravel adalah sebagai berikut:

1. *Expressive*

Seorang *programmer* diharapkan akan langsung mengetahui apa kegunaan dari sintaks Laravel meskipun belum pernah mempelajarinya apalagi menggunakannya.

2. *Simple*

Dengan adanya *Eloquent ORM* pada Laravel membuat ia begitu *simple*.

3. *Accessible*

Laravel Memiliki dokumentasi yang lengkap, sehingga *framework* ini menjadi lebih mudah untuk digunakan (Aminuddin, 2015).

## 2.5 Database

*Database* adalah kumpulan dari data terkait yang biasanya disimpan dalam komputer, file yang ditautkan dan dimanipulasi oleh paket perangkat lunak

husus yang disebut *database management systems* (DBMSs) (Nancy dkk, 2009).

*Database* bisa diartikan sebagai data yang tersimpan (walaupun sebenarnya tidak akan tersimpan untuk jangka waktu yang lama) didalam komputer (Susanto, 2017). Dengan adanya database lebih memudahkan untuk membangun sebuah SIA yang sudah di rancang sedemikian rupa dalam pengolahan datanya.

## **2.6 PHP**

PHP (singkatan rekursif untuk PHP: *Hypertext Preprocessor*) adalah bahasa *scripting* umum *open source* yang banyak digunakan yang sangat cocok untuk pengembangan *web* dan dapat dimasukkan ke dalam HTML (*The PHP Group*, 2017)

Hampir seluruh aplikasi yang berbasis *web* dapat dibuat dengan menggunakan PHP. Namun kekuatan utama PHP adalah konektivitas basis data dengan *web*. Dengan kemampuan ini akan pengguna mempunyai suatu sistem basis data yang dapat diakses dengan baik (Nugroho, 2009).

## **2.7 Black Box Testing**

Pengujian kotak hitam atau *black box testing* tidak bergantung pada pengetahuan terperinci tentang logika dari internal aplikasi yang di tes. Sebaliknya, mereka menganalisis bagan alur dan mewawancarai anggota yang memiliki pengetahuan di organisasi klien untuk memahami karakteristik fungsional aplikasi. Dengan pemahaman tentang apa yang seharusnya dilakukan oleh aplikasi, auditor menguji aplikasi dengan melakukan transaksi

*input* produksi yang diproses oleh aplikasi dengan hasil *output*-nya. Hasil *output* lalu dianalisis untuk memverifikasi apakah hasil aplikasi sesuai dengan persyaratan fungsionalnya (Hall, 2010).

Pengujian *black box* berusaha menemukan kesalahan dalam kategori yaitu:

1. Fungsi-fungsi yang salah atau hilang.
2. Kesalahan *interface*.
3. Kesalahan dalam struktur data atau akses *database* eksternal.
4. Kesalahan kinerja sistem.
5. Inisialisasi dan kesalahan terminasi.

Pengujian *black box* dilakukan selama tahap akhir pengujian karena *black box* memperhatikan struktur kontrol, maka perhatian berfokus pada fungsi akhir (Pressman, 2009).

## **2.8 Proses Bisnis Koperasi**

1. Mendaftar anggota koperasi

Untuk mendaftar menjadi anggota adalah dengan syarat membayar simpanan pokok dan membayar simpanan wajib tiap bulannya. Kemudian akan mendapat hasil dari sisa hasil usaha koperasi.

2. Meminjam Dana pada Koperasi

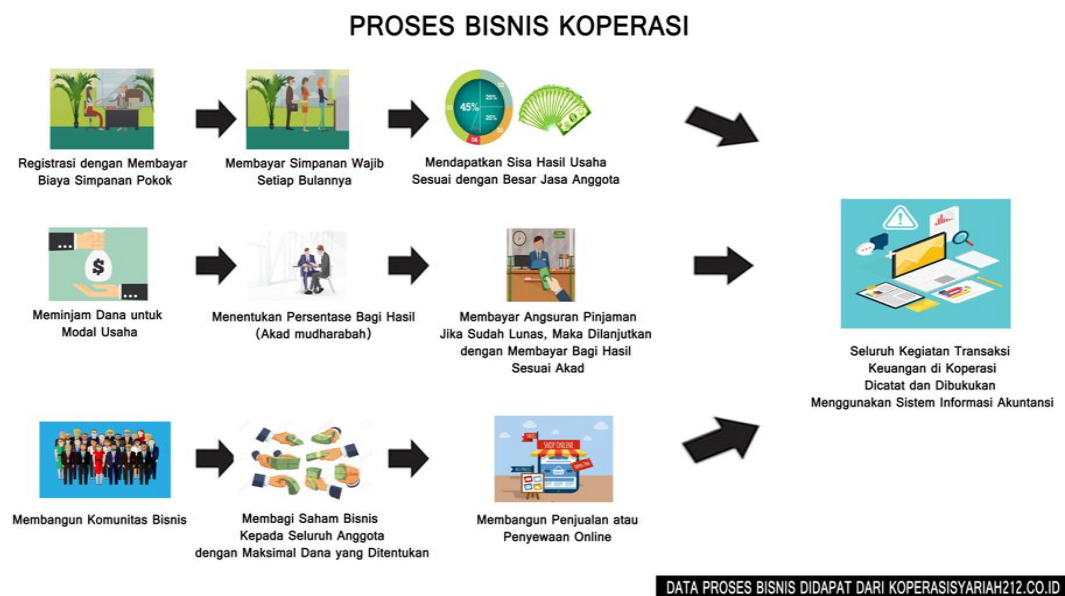
Untuk meminjam dana pada koperasi harus berdasarkan pada kebutuhannya, jika digunakan untuk kebutuhan pribadi maka tidak ada sistem bagi hasil dan jumlah dana yang dipinjamkan akan sama dengan jumlah dana yang dikembalikan. Bila dana tersebut digunakan untuk membuka usaha maka akan ada akad *mudharabah* yaitu akad bagi hasil dari keuntungan bisnis tersebut.

### 3. Membuka toko penjualan atau penyewaan

Untuk membuka suatu toko penjualan ataupun penyewaan maka toko harus dibuka dengan suatu komunitas yang memiliki anggota yang banyak dan modal yang didapatkan dari anggota komunitas tersebut sesuai saham yang akan ia ambil.

### 4. Pencatatan menggunakan SIA

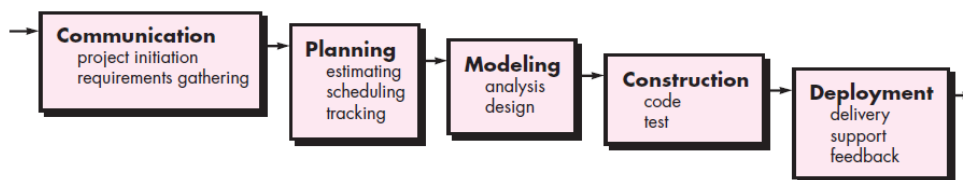
Ketika salah satu proses bisnis diatas dipilih, selanjutnya akan di catat keseluruhan nya oleh SIA. Pencatatan transaksi haruslah transparan kepada seluruh anggotanya karena kepercayaan anggota adalah kunci berjalannya koperasi tersebut. SIA dapat memudahkan proses pencatatan data dan laporan untuk mengurangi *human error* dalam pencatatan data karena data keuangan sangat sensitif jika terjadi kesalahan.



Gambar 2.2 Proses Bisnis Koperasi.

## 2.9 Metode Pengembangan Sistem

Metode *waterfall* sering juga disebut siklus hidup klasik (*classic life cycle*), kemajuan yang mengalir di sebagian besar satu arah ("ke bawah" seperti air terjun), dimulai dengan spesifikasi kebutuhan pengguna (*communication*) lalu berlanjut melalui tahapan-tahapan perencanaan (*planning*), permodelan (*modeling*), konstruksi (*construction*), serta penyerahan sistem ke para pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan (Pressman, 2012).



Gambar 2.3 Metode *Waterfall*.

### 1. *Communication*

Pada tahap ini komunikasi dilakukan untuk keselarasan pengembangan sistem agar sesuai kebutuhan pengguna.

### 2. *Planning*

Pada tahapan ini melakukan estimasi waktu dan penjadwalan pengerjaan sistem.

### 3. *Modeling*

Pada tahap ini dibuatkan *design* sistemnya. *Design* sistem membantu dalam menentukan perangkat keras dan membantu arsitektur sistem secara keseluruhan.

#### 4. *Construction*

Pada tahapan ini akan dilakukan pengkodean dan *testing* dari pemodelan tahap sebelumnya. Pengkodean dilakukan dengan menggunakan *framework* Laravel dan *testing* menggunakan *blackbox testing*.

#### 5. *Deployment*

Setelah dilakukannya *construction* maka selanjutnya adalah penyerahan sistem kepada pengguna.

### **2.10 *Expert Judgement***

*Expert Judgement* adalah suatu *tools* dan teknik yang digunakan dalam manajemen resiko guna menilai input dan proses yang tujuannya untuk mengembangkan *project charte* dalam artian adalah dianggap sebagai aset organisasi karena mampu memberikan masukan untuk merencanakan dan memperkirakan aktivitas yang penting dari proyek yang dibangun (PMI, 2013). *Expert Judgement* digunakan pada situasi yang memerlukan pertimbangan dari para ahli dengan menyelesaikan, memvalidasi, menafsirkan, dan mengintegrasikan data yang tersedia, menilai dampak perubahan yang terjadi pada organisasi, melakukan prediksi kejadian yang akan datang dan konsekuensi setiap keputusan, menentukan keadaan saat ini, menyediakan unsur-unsur yang dibutuhkan untuk proses pengambilan keputusan yang tepat.

### **2.11 Penelitian Terdahulu**

Iman (2018) meneliti Sistem Informasi Multi Koperasi dengan model *Software as a Service* (SaaS) sehingga harga sistem tidak terlalu mahal dan koperasi dapat menggunakan sistem tanpa perlu memikirkan perawatan sistem. Didalam multi koperasi terdapat beberapa modul dan salah satunya adalah fitur Akuntansi *Double Entry* yang dikembangkan oleh Rahmanda (2018). SIA multi koperasi sudah memiliki fitur pembukuan yang sesuai kaidah akuntansi sehingga akan lebih mudah digunakan bagi koperasi yang memiliki pemahaman akuntansi secara profesional atau memiliki staff ahli bidang akuntansi sehingga dapat memudahkan koperasi dalam membuat laporan keuangan.



## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1 Waktu dan Tempat Penelitian**

Penelitian dilakukan di Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung yang berada di Jalan Soemantri Bojonegoro No.1 Gedung Meneng, Bandar Lampung. Penelitian ini dimulai pada bulan Desember 2018 sampai dengan selesai.

#### **3.2 Metodologi Penelitian**

##### **3.2.1 Kerangka Penelitian**

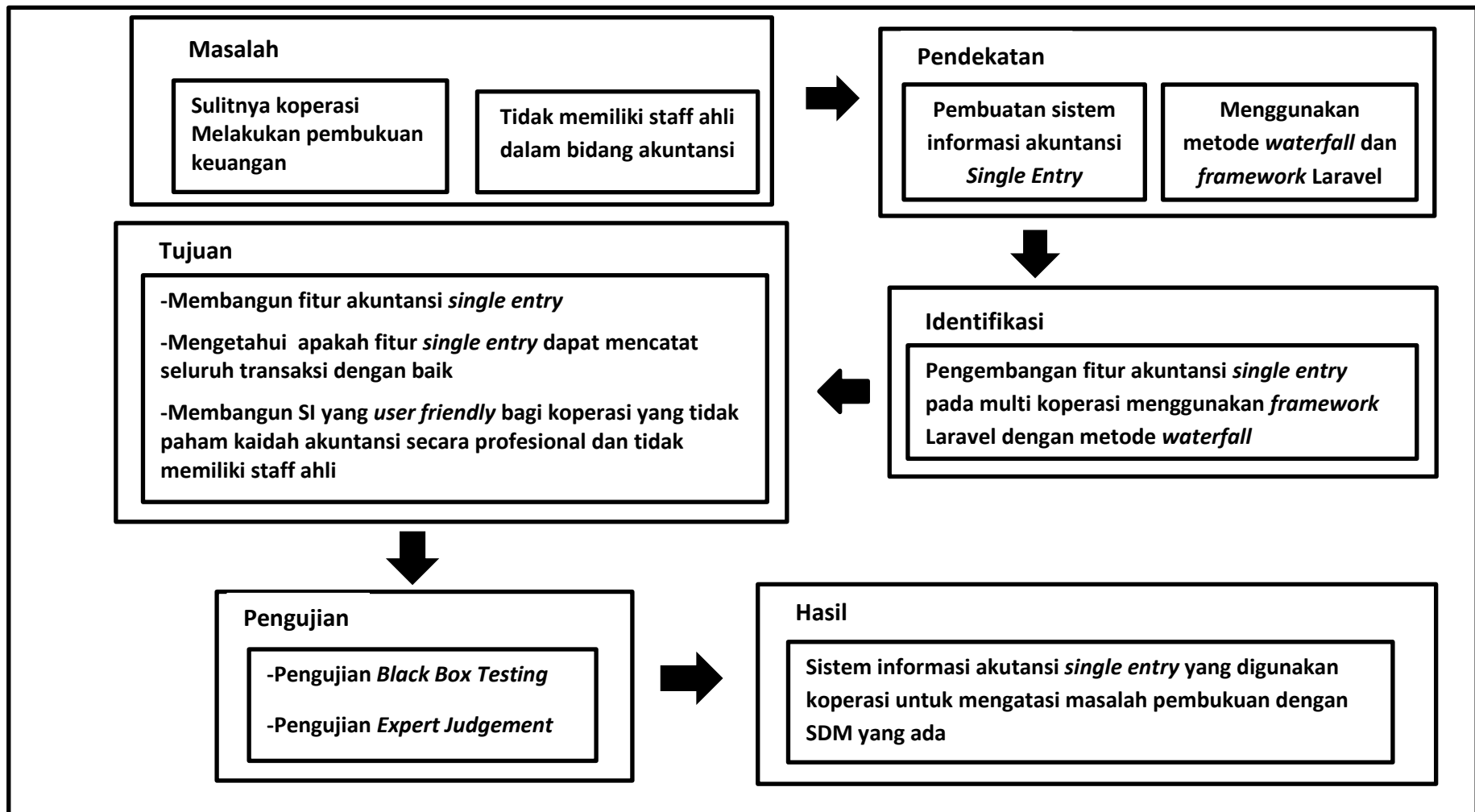
Penelitian berawal dari masalah sulitnya koperasi melakukan pembukuan keuangan dengan menggunakan metode *double entry* dan tidak memiliki staff ahli akuntansi yang mempunyai. Hal itu menyebabkan informasi menjadi sulit untuk didapatkan dan koperasi tidak bisa mengetahui jumlah laba ataupun saldo untuk koperasi tersebut.

Kemudian dilakukan pendekatan dengan membangun fitur *single entry* pada sistem informasi multi koperasi untuk memudahkan koperasi yang belum memahami kaidah akuntansi yang di dalamnya dapat digunakan untuk mencatat proses bisnis koperasi, mencetak laporan keuangan seperti laba rugi, neraca, perubahan modal dan arus kas. Penggunaan metode pengembangan sistem ini adalah *waterfall* dikarenakan metode ini sudah familiar. *Framework* laravel

digunakan karena dari segi keamanan laravel sudah baik dengan menggunakan *bcrypt* dan laravel memiliki banyak dokumentasi sehingga memudahkan peneliti dalam membangun sistem.

Selanjutnya melakukan identifikasi. Identifikasi yang dilakukan dengan menggabungkan beberapa pendekatan dan didapatkan pengembangan fitur akuntansi *single entry* menggunakan *framework* laravel dan metode pengembangan sistem *waterfall*. Dengan pengembangan *waterfall* sistem akan dikembangkan secara bertahap mulai dari analisis kebutuhan apa saja, desain sistem, penulisan kode program dan terakhir melakukan pengujian sistem sebelum digunakan koperasi.

Tujuan dari penelitian ini adalah dengan adanya fitur akuntansi *single entry* diharapkan dapat memudahkan koperasi yang tidak paham kaidah akuntansi yang mendalam dan membantu koperasi dalam melakukan seluruh pembukuan keuangan. Setelah sistem selesai dibuat, selanjutnya akan dilakukan pengujian sistem dengan menggunakan teknik *black box testing* yaitu dengan menguji fungsionalitas pada sistem sudah sesuai atau belum. Selain *black box testing*, sistem akan diuji dengan pengujian *expert judgement*. Setelah diuji keseluruhan maka sistem diharapkan dapat digunakan koperasi untuk membantu melakukan proses bisnis yang ada di koperasi.



Gambar 3.1 Kerangka Penelitian.

### 3.2.2 Metode Pencatatan Akuntansi

Tabel 3.1 Perbedaan *Single Entry* dan *Double Entry*.

N o.	Kategori	Contoh <i>Single Entry</i>		Contoh <i>Double Entry</i>			
1	Jumlah baris saat input pembukuan (Jurnal Umum): - <i>Single Entry</i> hanya memiliki satu baris inputan yang mempengaruhi kas masuk/ kas keluar.  - <i>Double Entry</i> Memiliki 2 baris saat penginputan sesuai dengan kaidah akuntansi debit/kredit.	Nama rekening	Jumlah	Nama rekening	Debit	Kredit	
		Pembelian Pena	2000	Pembelian Pena	2000	0	
				Kas	0	2000	
2	Penggolongan Debit Kredit (Buku Besar) - <i>Single Entry</i> Debit: Kas masuk Kredit: kas keluar  - <i>Double Entry</i> 1. Harta (+) Debit; (-) Kredit  2. Hutang (+) Kredit; (-) Debit  3. Modal (+) Kredit; (-) Debit  4. Pendapatan (+) Kredit; (-) Debit  5. Beban (+) Debit; (-) Kredit	Nama rekening	Debit	Kredit	Nama rekening	Debit	Kredit
		Pendapatan Toko	2000		Kas	2000	0
		Bayar Hutang	1000	Pendapatan Toko	0	2000	
				Bayar Hutang	1000	0	
				Kas	0	1000	
3	Laporan Arus Kas	Kas keluar mengambil data di jurnal umum yang ada di kredit dan kas masuk mengambil data di jurnal umum yang ada di debit.		Mengambil data jurnal umum yang berhubungan dengan kas yang biasanya pada akun pendapatan atau akun beban.			

Pada tabel 3.1 dijelaskan tentang perbedaan *single entry* dengan *double entry*. Pada jurnal umum *single entry* hanya ada satu masukan dan *double entry* dua masukan, kemudian pada buku besar *single entry* hanya membedakan antara debit dan kredit sedangkan *double entry* terdapat dua baris yang nilainya ditentukan dengan rumus tabel akuntansi dan yang terakhir pada laporan keuangan.

### 3.2.3 Metode Pengembangan Sistem

Metode pengembangan sistem pada penelitian ini menggunakan metode *waterfall*.

Adapun tahapan yang dilakukan adalah sebagai berikut:

1. *Communication*

Pada tahap ini komunikasi dilakukan untuk keselarasan pengembangan sistem agar sesuai kebutuhan pengguna.

2. *Planning*

Pada tahapan ini melakukan estimasi waktu dan penjadwalan dalam pengerjaan sistem.

3. *Modeling*

Pada tahap ini dibuatkan *design* sistemnya. *Design* sistem membantu dalam menentukan perangkat keras dan membantu arsitektur sistem secara keseluruhan.

4. *Construction*

Pada tahapan ini akan dilakukan pengkodean dan *testing* dari pemodelan tahap sebelumnya.

5. *Deployment*

Setelah dilakukannya *construction* maka selanjutnya adalah penyerahan sistem kepada pengguna.

### **3.2.4 Metode Pengumpulan Data**

Pada penelitian ini menggunakan dua metode pengumpulan data, yaitu:

1. Data Primer

Data primer adalah data yang didapatkan dengan cara mewawancarai dari pihak koperasi mengenai proses bisnis yang ada.

2. Data Sekunder

Data sekunder adalah data yang didapatkan dari jurnal, buku dan internet yang menyajikan informasi tentang proses bisnis *single entry*.

## **3.3 Analisis Sistem**

### **3.3.1 Analisis Masalah**

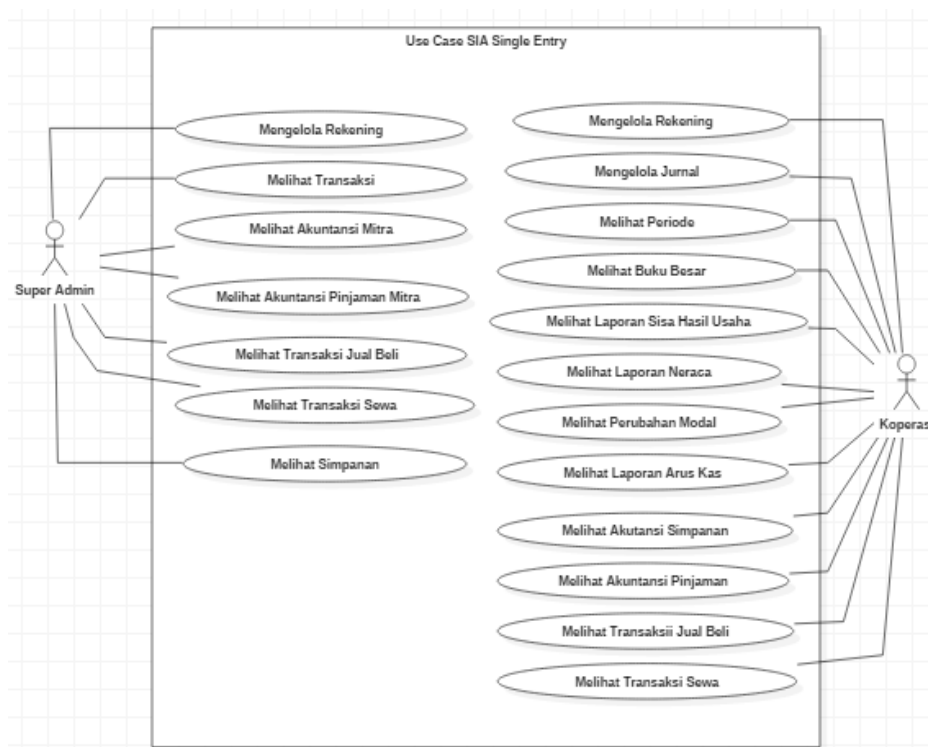
Koperasi di Indonesia sudah mulai ditinggalkan. Kesulitan pembukuan keuangan, transparansi keuangan dan laporan keuangan yang tidak jelas pada koperasi mengakibatkan koperasi sulit mengambil keputusan itulah yang menjadi alasan mengapa koperasi ditinggalkan.

Sistem informasi akuntansi *single entry* pada koperasi bisa menjadi solusi bagi koperasi dalam melakukan proses pembukuan keuangan. Pembuatan laporan keuangan juga dapat dengan mudah diselesaikan dengan menggunakan sistem tersebut. Diharapkan dengan adanya sistem akuntansi *single entry* dapat memudahkan koperasi dalam melakukan pencatatan hingga laporan keuangan.

### **3.3.2 Analisis Kebutuhan Sistem**

Dalam membangun sistem informasi akuntansi *single entry* yang diperlukan yaitu informasi dari proses bisnis yang berjalan di koperasi. Informasi didapatkan dari hasil wawancara dan analisis kepada pihak koperasi dan sistem sebelumnya.

Pada proses awal koperasi melakukan pencatatan pada jurnal umum dengan mengisi tanggal, keterangan, nama rekening, nomor rekening, debit dan kredit. Kemudian hasil dari pencatatan jurnal umum akan ditampilkan pada buku besar dengan jumlah debit dan kreditnya. Pada tahapan akhir koperasi melakukan pelaporan keuangan meliputi arus kas, sisa hasil usaha, neraca dan perubahan modal. Kebutuhan sistem telah digambarkan kedalam *use case diagram* seperti pada gambar 3.2.



Gambar 3.2 *Use Case Diagram*.

Pada gambar 3.2 terdapat dua level *user*, penjelasan level *user* adalah sebagai berikut.

#### 1. Super Admin

Super admin adalah pengembang dari SI yang dikelola oleh pengembang sistem multi koperasi. Super admin dapat melihat data seluruh anggota koperasi yang menggunakan sistem, tetapi tidak bisa mengubah data tersebut.

## 2. Admin koperasi

Admin adalah pemilik koperasi yang menggunakan SIA. Admin dapat mengelola data akuntansi dan mencetak seluruh laporan keuangan.

### 3.4 Desain Sistem

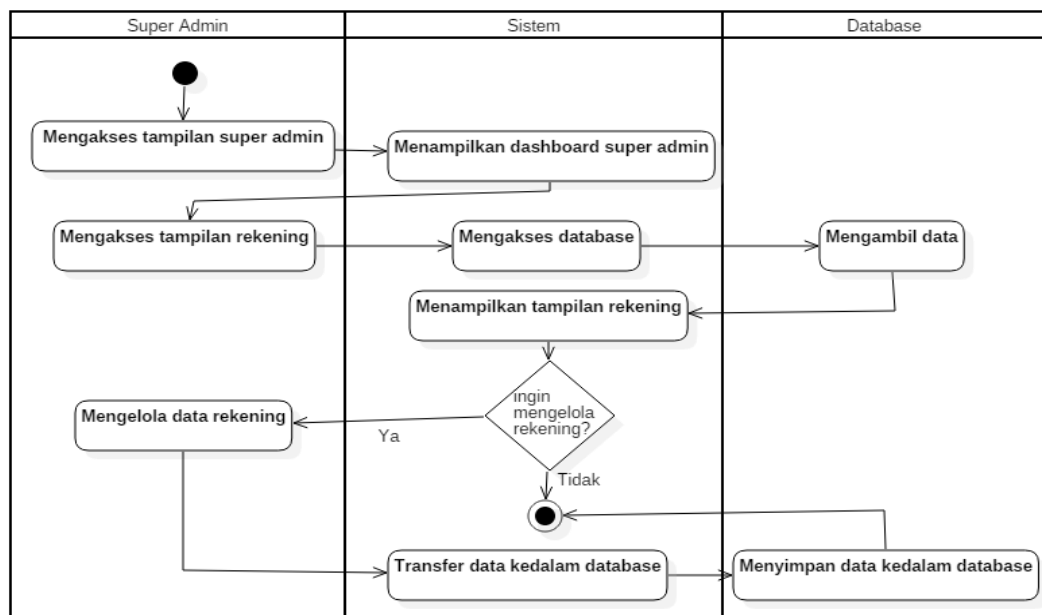
Desain sistem adalah tahapan selanjutnya dari metode *waterfall* setelah tahapan analisis. Desain sistem dibuat untuk memudahkan dalam proses pembangunan sistem. Desain juga menjadi gambaran bagaimana sistem akan berjalan, bagaimana tampilan antar muka dan bagaimana sistem akan menyelesaikan suatu masalah.

#### 3.4.1 Perancangan UML (*Unified Modelling Language*)

Perancangan sistem dilakukan dengan memodelkan masalah dalam bentuk diagram-diagram UML. Diagram-diagram UML tersebut adalah sebagai berikut.

##### 1. *Activity Diagram*

##### a. *Activity Diagram Super Admin Mengelola Rekening*

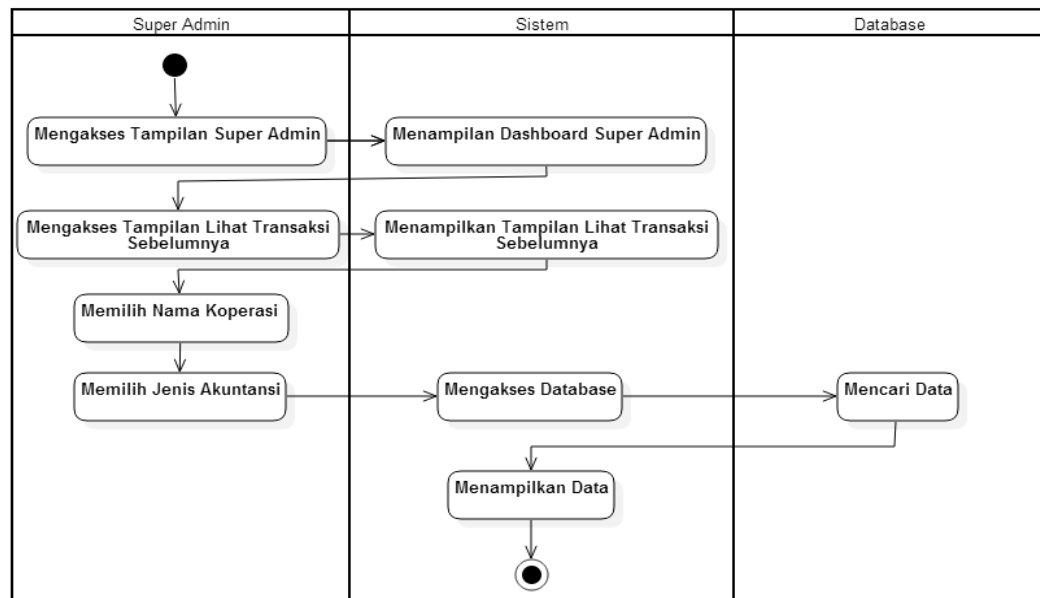


Gambar 3.3 *Activity Diagram* Mengelola Rekening.



Pada gambar 3.3 dapat dijelaskan bahwa super admin dapat melihat dan mengelola data rekening akuntansi dari mitra pengguna sistem. Pertama super admin mengakses tampilan super admin dengan login terlebih dahulu. Setelah sistem menampilkan *dashboard* lalu super admin melihat tampilan rekening. Lalu super admin memilih apakah ingin mengelola rekening atau tidak. Jika ya, maka sistem akan mengelola rekening dan disimpan di *database*. Jika tidak, maka proses selesai.

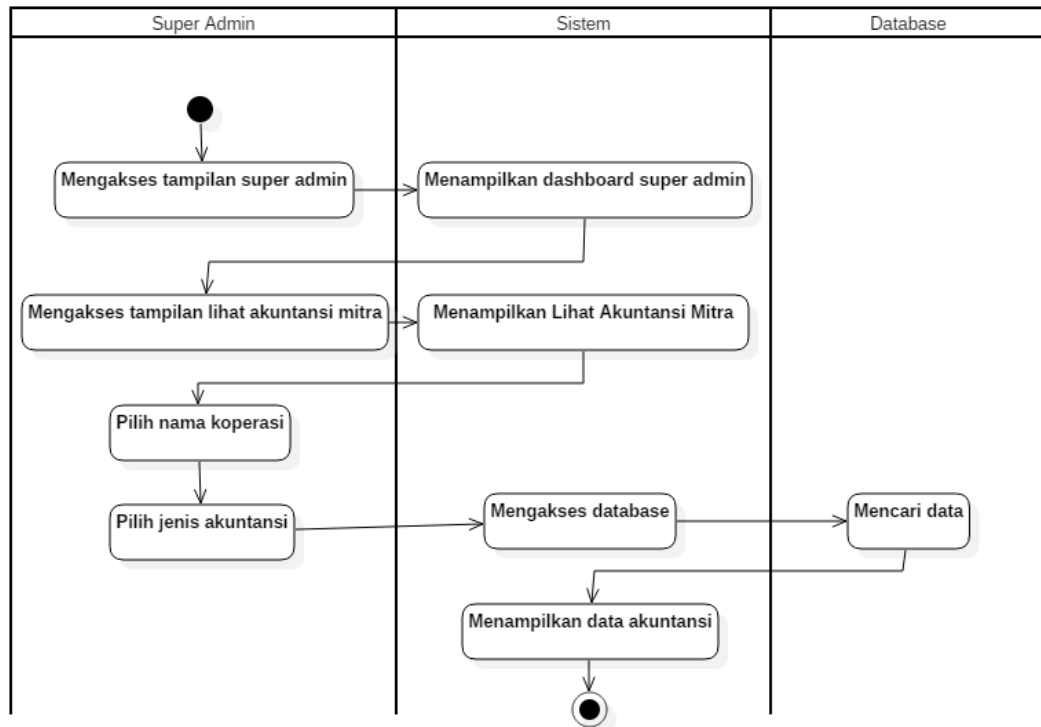
b. *Activity Diagram* Melihat Transaksi Sebelumnya



Gambar 3.4 *Activity Diagram* Melihat Transaksi Sebelumnya.

Pada gambar 3.4 dapat dijelaskan bahwa super admin dapat melihat data transaksi sebelumnya. Pertama super admin mengakses tampilan dengan login terlebih dahulu, lalu sistem menampilkan dashboard super admin. Kemudian super admin mengakses tampilan transaksi sebelumnya dengan memilih koperasi dan jenis akuntansinya. Setelah itu sistem akan mengakses database dan database mencari data dan sistem akan menampilkan data transaksi mitra.

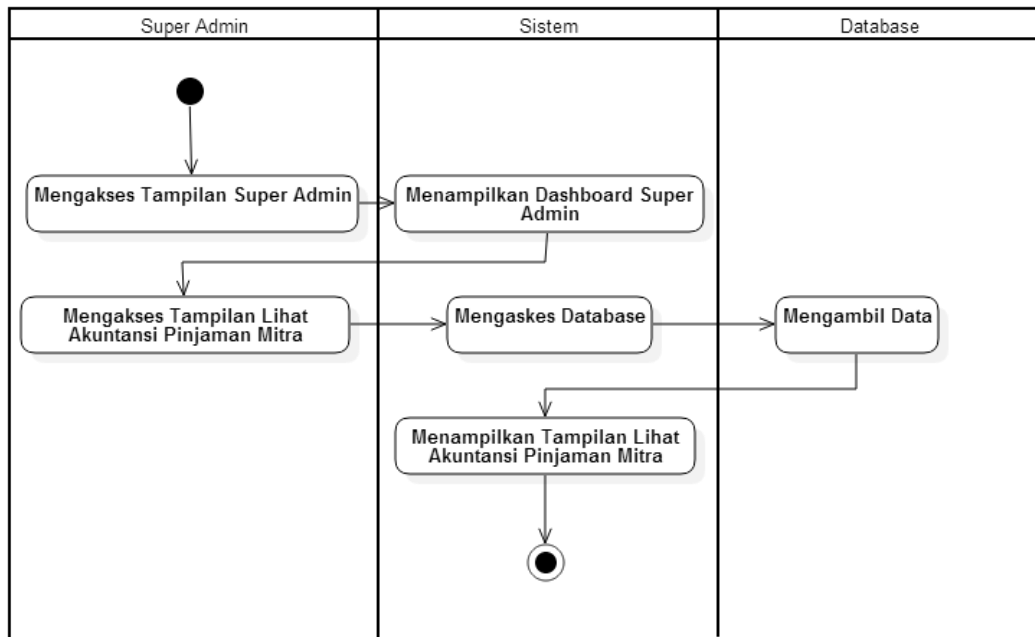
c. *Activity Diagram* Melihat Akuntansi Mitra



Gambar 3.5 *Activity Diagram* Melihat Akuntansi Mitra.

Pada gambar 3.5 dapat dijelaskan bahwa super admin dapat melihat akuntansi mitra. Pertama super admin mengakses tampilan super admin, lalu sistem menampilkan *dashboard* dan super admin mengakses tampilan melihat akuntansi mitra. Kemudian super admin memilih nama koperasi dan jenis akuntansi mitra. Sistem akan mengakses *database* dan menampilkannya pada sistem.

d. *Activity Diagram* Melihat Akuntansi Pinjaman Mitra

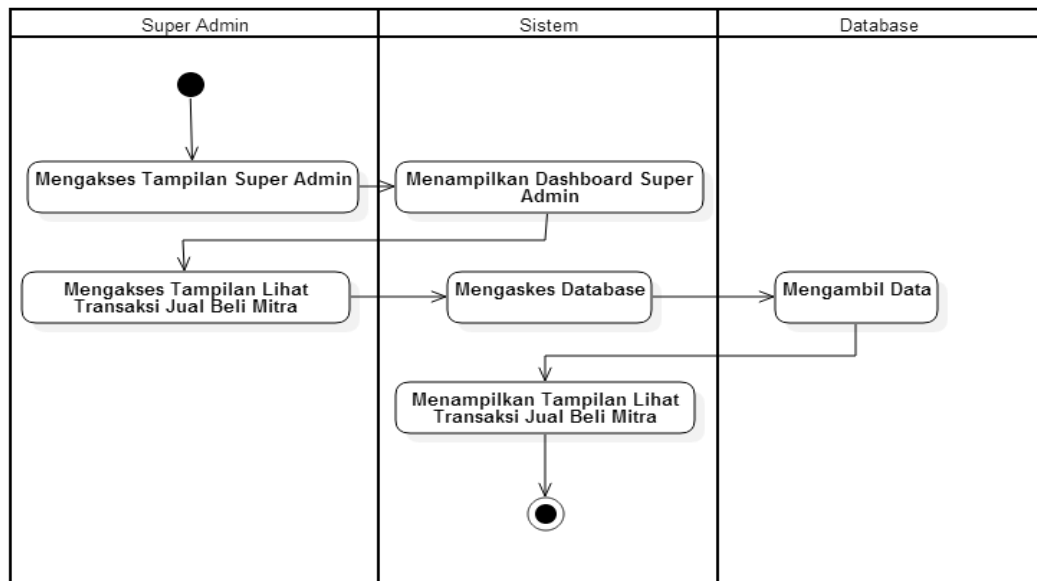


Gambar 3.6 *Activity Diagram* Melihat Akuntansi Pinjaman.

Pada gambar 3.6 dapat dijelaskan bahwa super admin dapat melihat akuntansi pinjaman mitra. Pertama super admin mengakses tampilan super admin dan sistem akan menampilkan *dashboard* super admin, kemudian super admin mengakses lihat akuntansi pinjaman mitra. Setelah itu sistem akan mengakses *database* dan mengambil data di *database* kemudian ditampilkan pada sistem.

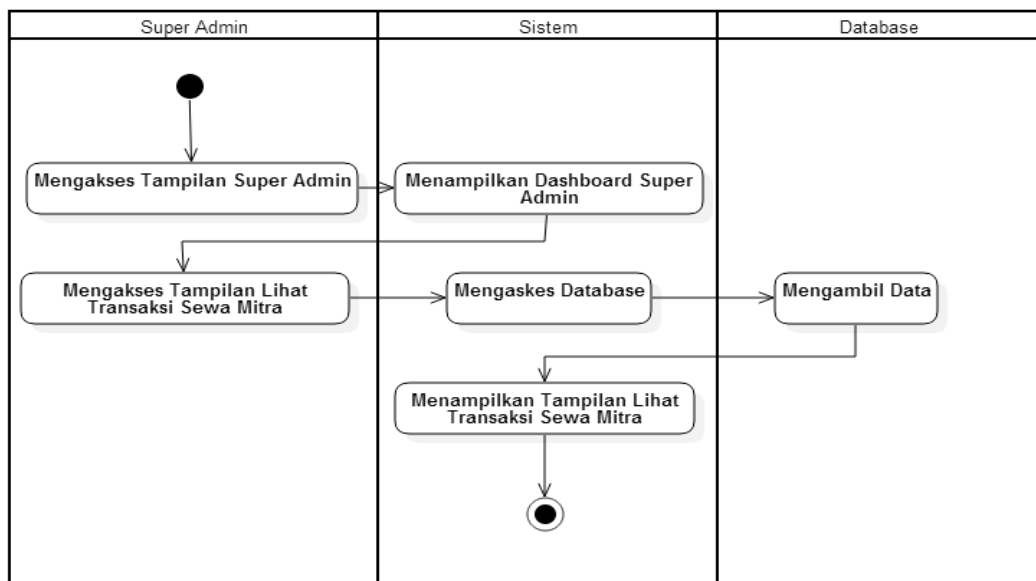
e. *Activity Diagram* Melihat Transaksi Jual Beli

Pada gambar 3.7 dapat dijelaskan bahwa super admin dapat melihat data transaksi jual beli mitra. Pertama super admin mengakses tampilan super admin dan sistem akan menampilkan *dashboard* super admin kemudian super admin mengakses lihat transaksi jual beli mitra. Setelah itu sistem akan mengakses *database* dan mengambil data di *database* kemudian ditampilkan pada sistem.



Gambar 3.7 Activity Diagram Melihat Transaksi Jual Beli.

f. Activity Diagram Melihat Transaksi Sewa Mitra

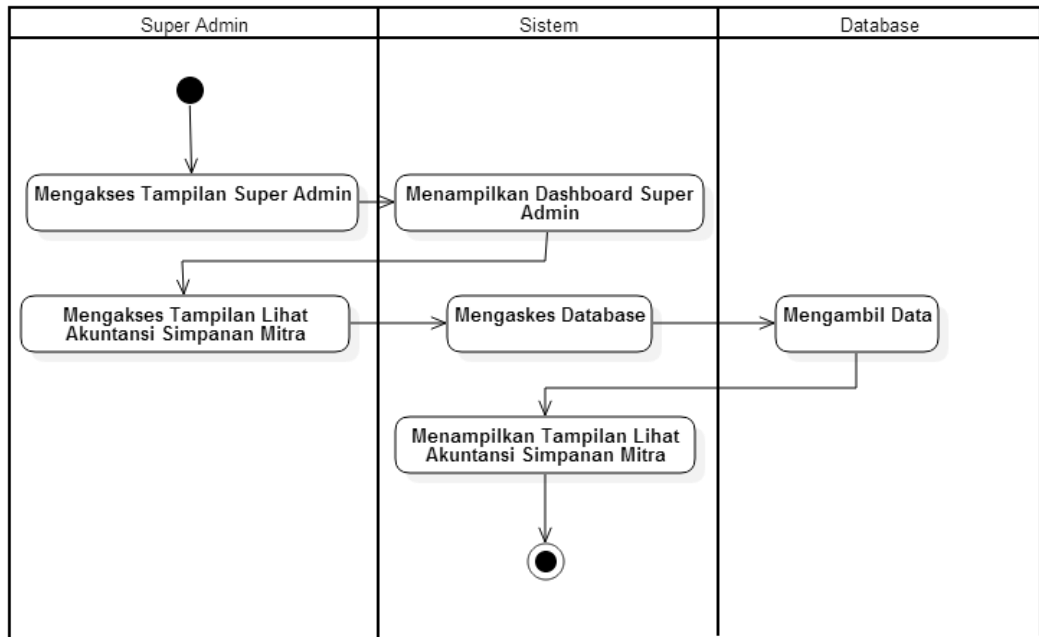


Gambar 3.8 Activity Diagram Melihat Transaksi Sewa Mitra.

Pada gambar 3.8 dapat dijelaskan bahwa super admin dapat melihat data transaksi sewa mitra. Pertama super admin mengakses tampilan super admin dan sistem akan menampilkan *dashboard* super admin kemudian super admin

mengakses lihat transaksi sewa mitra. Setelah itu sistem akan mengakses *database* dan mengambil data di *database* kemudian ditampilkan pada sistem.

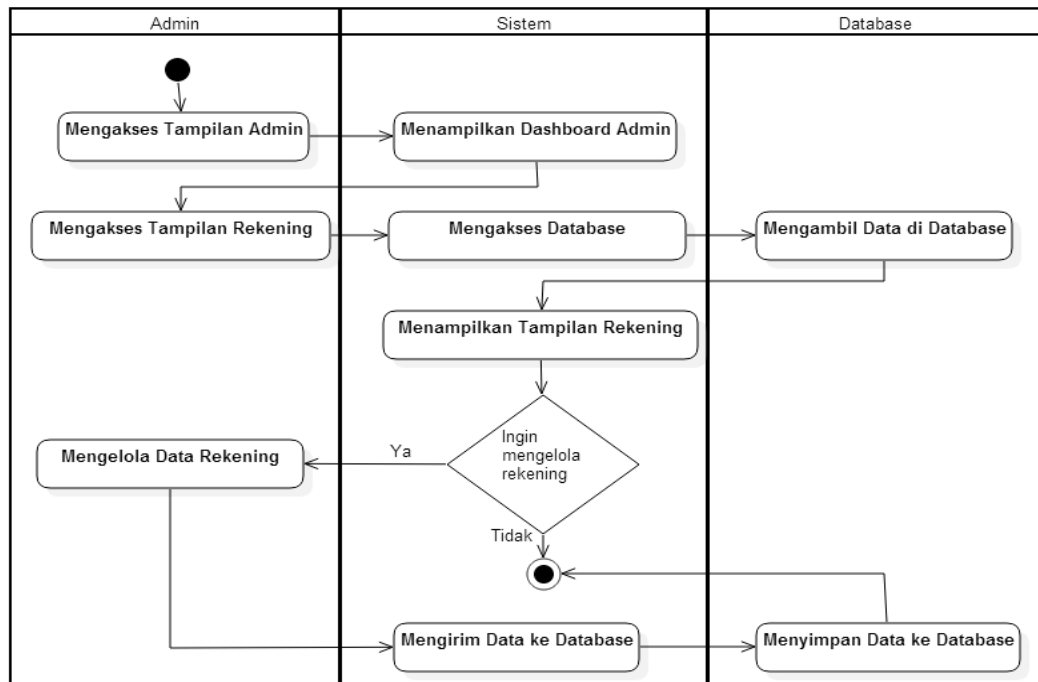
g. *Activity Diagram* Melihat Simpanan Mitra



Gambar 3.9 *Activity Diagram* Melihat Simpanan Mitra.

Pada gambar 3.9 dapat dijelaskan bahwa super admin dapat melihat data akuntansi simpanan mitra. Pertama super admin mengakses tampilan super admin dan sistem akan menampilkan *dashboard* super admin kemudian super admin mengakses lihat akuntansi simpanan mitra. Setelah itu sistem akan mengakses *database* dan mengambil data di *database* kemudian ditampilkan pada sistem.

#### h. Activity Diagram Mengelola Rekening



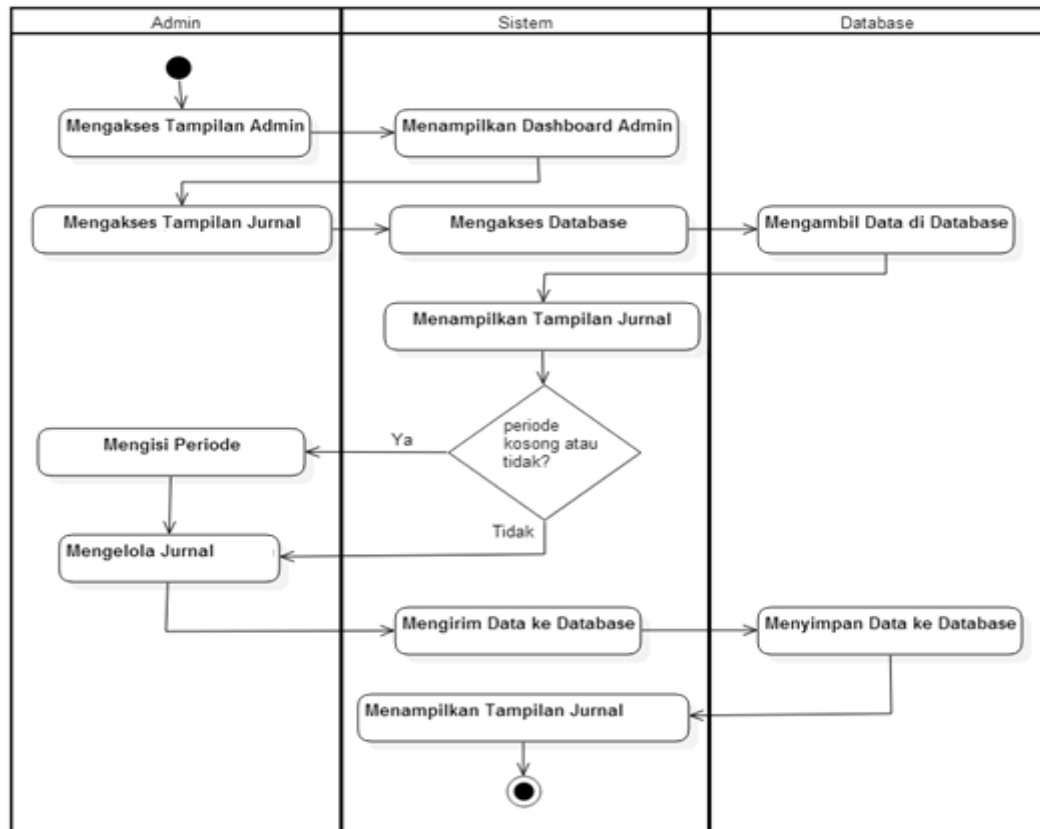
Gambar 3.10 Activity Diagram Mengelola Rekening.

Pada gambar 3.10 dapat dijelaskan bahwa admin dapat mengelola rekening. Pertama admin mengakses tampilan admin dan sistem akan menampilkan *dashboard* admin kemudian admin mengakses tampilan rekening. Setelah itu sistem akan mengakses *database* dan mengambil data di *database* kemudian ditampilkan pada sistem. Kemudian admin dapat memilih ingin mengelola rekening atau tidak. Jika ya, maka admin akan mengelola rekening lalu disimpan di *database* dan selesai. Jika tidak, maka selesai.

#### i. Activity Diagram Mengelola Jurnal

Pada gambar 3.11 dapat dijelaskan bahwa admin dapat mengelola jurnal. Pertama admin mengakses tampilan admin dan sistem akan menampilkan *dashboard* admin kemudian admin mengakses tampilan mengelola jurnal. Kemudian sistem mengakses *database* dan mengambil data yang ditampilkan

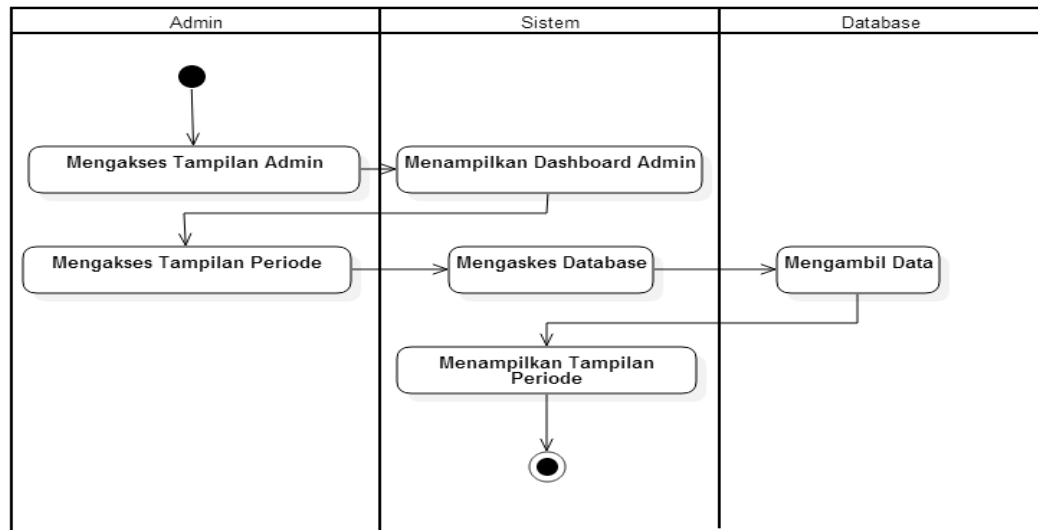
di sistem. Jika periode kosong, maka admin mengisi periode terlebih dahulu. Jika tidak maka admin dapat langsung mengelola jurnal. Kemudian sistem akan mengakses database dan disimpan pada database kemudian sistem menampilkan kembali tampilan jurnal dan selesai.



Gambar 3.11 Activity Diagram Mengelola Jurnal.

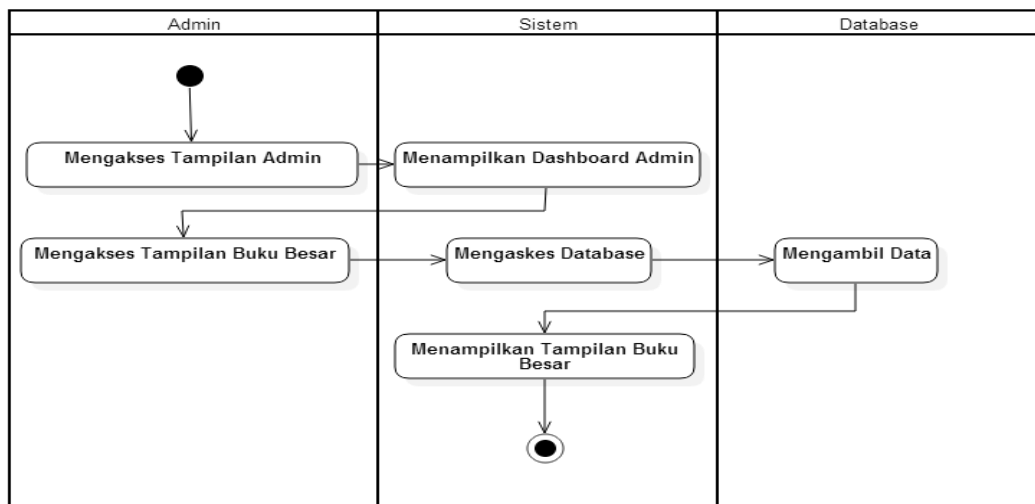
j. Activity Diagram Melihat Periode

Pada gambar 3.12 dapat dijelaskan bahwa admin dapat melihat periode. Pertama admin mengakses tampilan admin dan sistem akan menampilkan *dashboard* admin, kemudian admin mengakses periode. Setelah itu sistem akan mengakses *database* dan mengambil data di *database* kemudian ditampilkan pada sistem.



Gambar 3.12 *Activity Diagram* Melihat Periode.

k. *Activity Diagram* Melihat Buku Besar

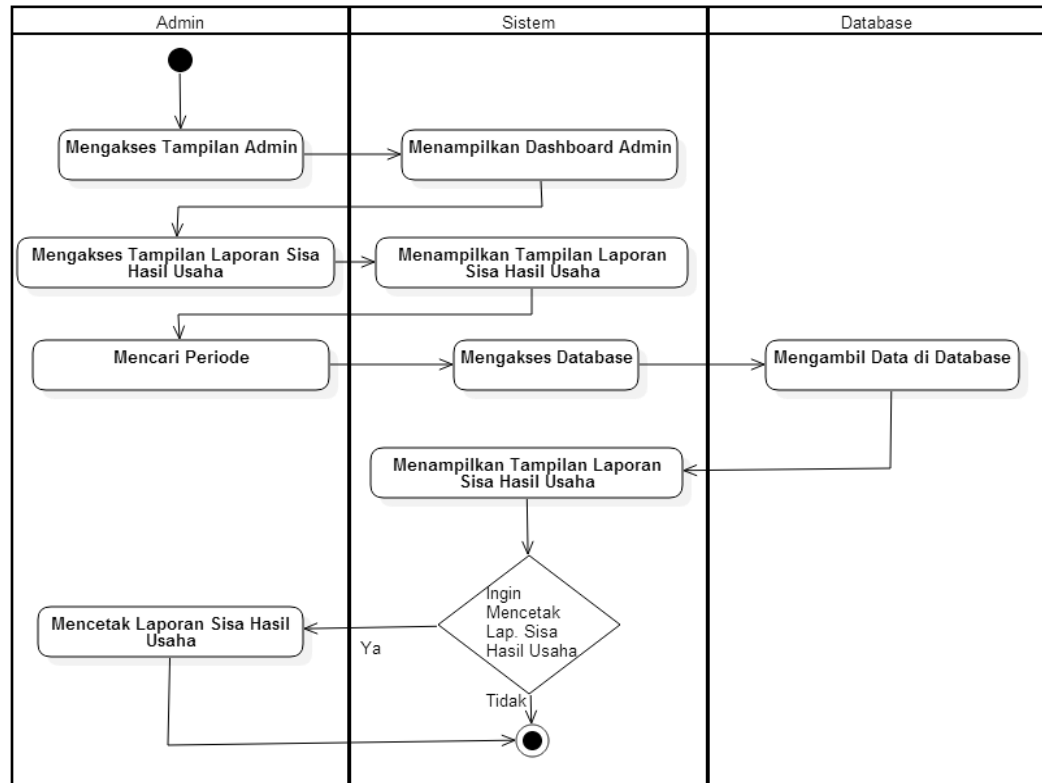


Gambar 3.13 *Activity Diagram* Melihat Buku Besar.

Pada gambar 3.13 dapat dijelaskan bahwa admin dapat melihat buku besar. Pertama admin mengakses tampilan admin dan sistem akan menampilkan *dashboard* admin, kemudian admin mengakses buku besar. Setelah itu sistem akan mengakses *database* dan mengambil data di *database* kemudian ditampilkan pada sistem.



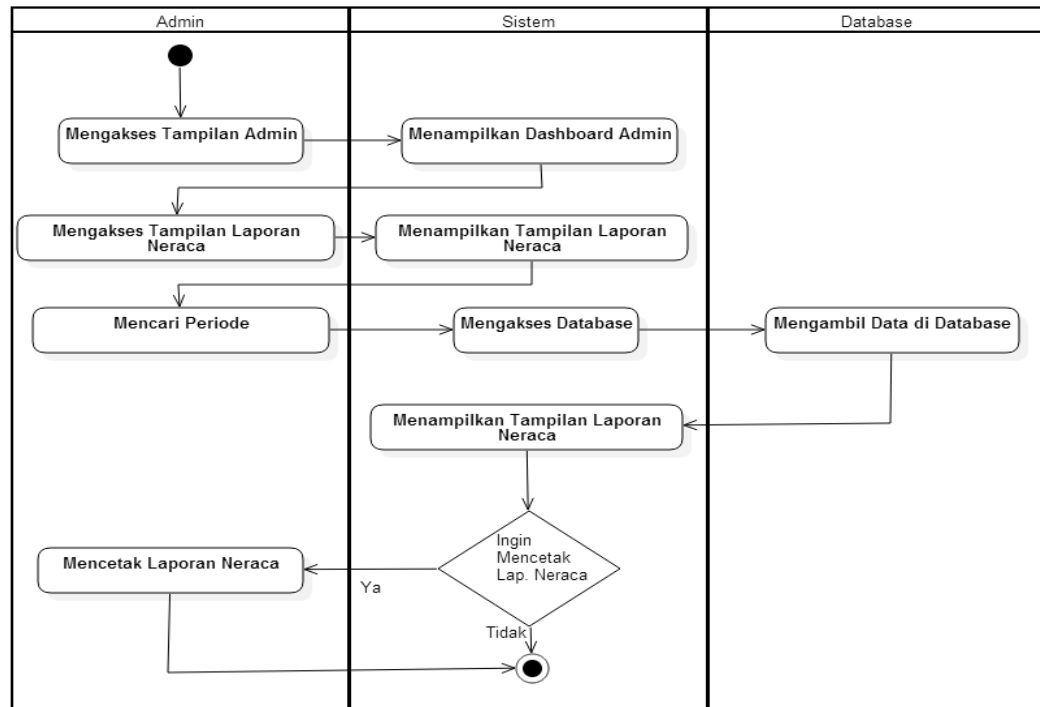
### 1. Activity Diagram Melihat Laporan Sisa Hasil Usaha



Gambar 3.14 Activity Diagram Melihat Laporan Sisa Hasil Usaha.

Pada gambar 3.14 dapat dijelaskan bahwa admin dapat melihat dan mencetak laporan sisa hasil usaha. Pertama admin mengakses tampilan admin dan sistem akan menampilkan *dashboard* admin, kemudian admin mengakses laporan sisa hasil usaha dan sistem menampilkan sisa hasil usaha. Lalu admin mencari periode dan sistem akan mengakses *database* untuk mengambil data di *database* dan menampilkan di sistem, kemudian admin memilih ingin mencetak laporan sisa hasil usaha atau tidak.

m. *Activity Diagram* Melihat Neraca



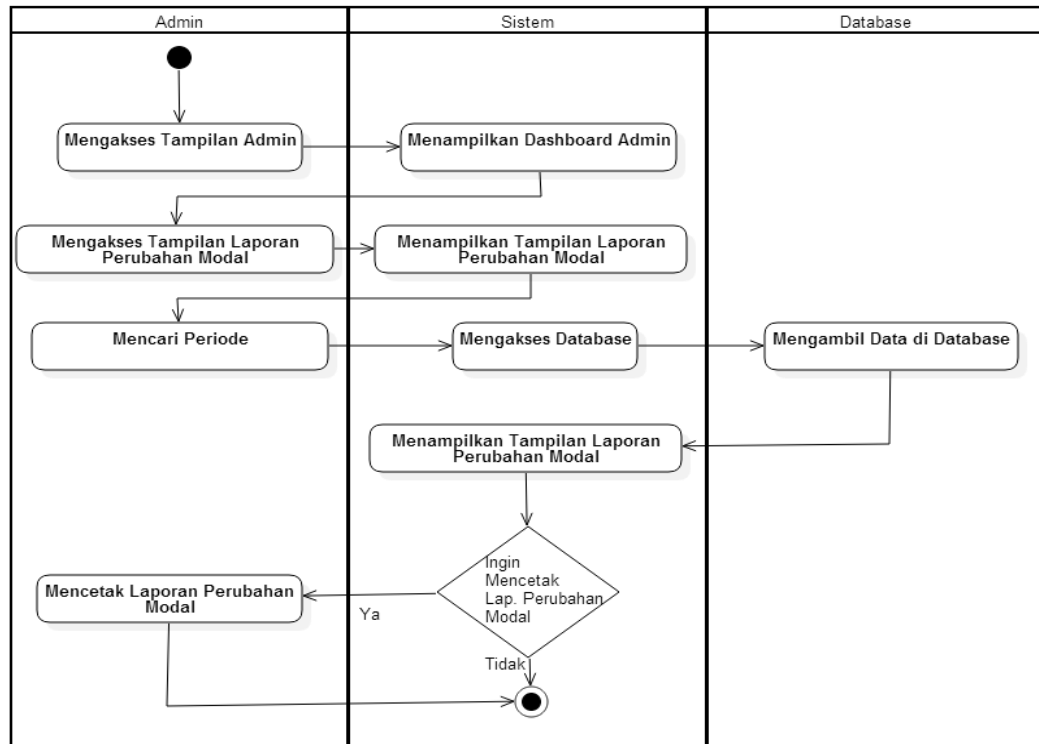
Gambar 3.15 *Activity Diagram* Melihat Neraca.

Pada gambar 3.15 dapat dijelaskan bahwa admin dapat melihat dan mencetak laporan neraca. Pertama admin mengakses tampilan admin dan sistem akan menampilkan *dashboard* admin, kemudian admin mengakses neraca dan sistem menampilkan neraca. Lalu admin memilih periode dan sistem akan mengakses *database* untuk mengambil data di *database* dan menampilkan di sistem, kemudian admin memilih ingin mencetak laporan neraca atau tidak.

n. *Activity Diagram* Melihat Perubahan Modal

Pada gambar 3.16 dapat dijelaskan bahwa admin dapat melihat dan mencetak laporan perubahan modal. Pertama admin mengakses tampilan admin dan sistem akan menampilkan *dashboard* admin, kemudian admin mengakses perubahan modal dan sistem menampilkan perubahan modal. Lalu admin memilih periode dan sistem akan mengakses *database* untuk mengambil data

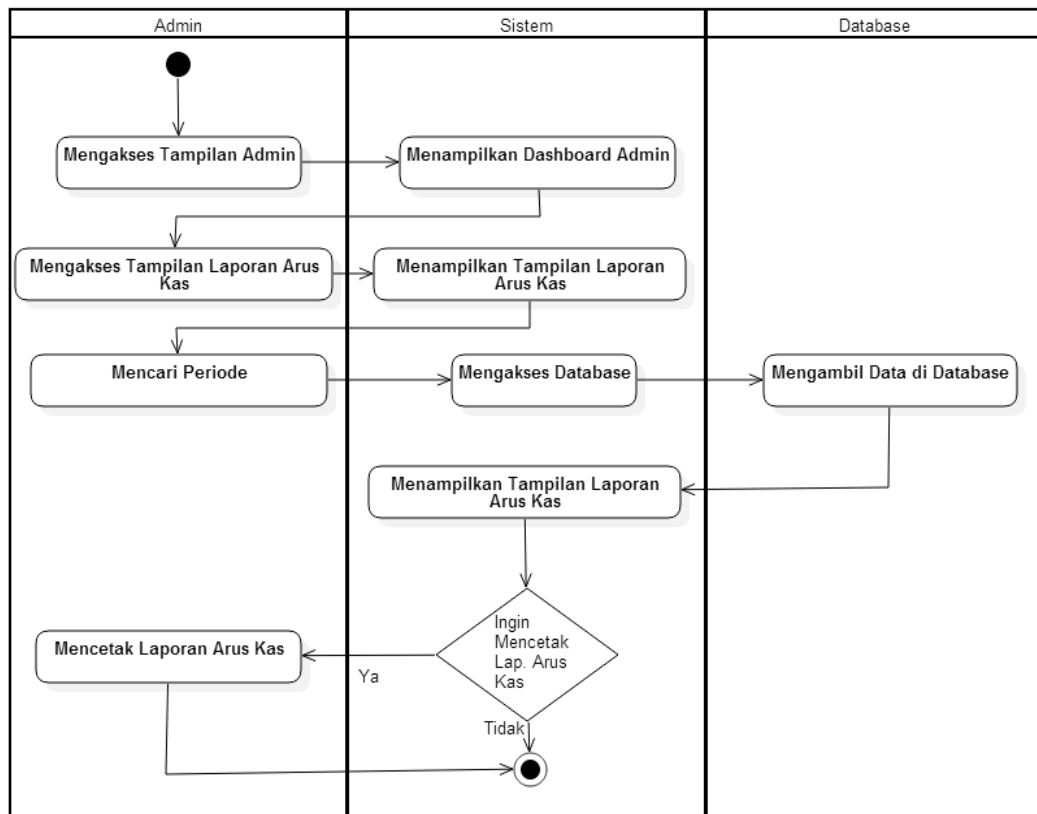
di *database* dan menampilkan di sistem, kemudian admin memilih ingin mencetak laporan perubahan modal atau tidak.



Gambar 3.16 *Activity Diagram* Melihat Perubahan Modal.

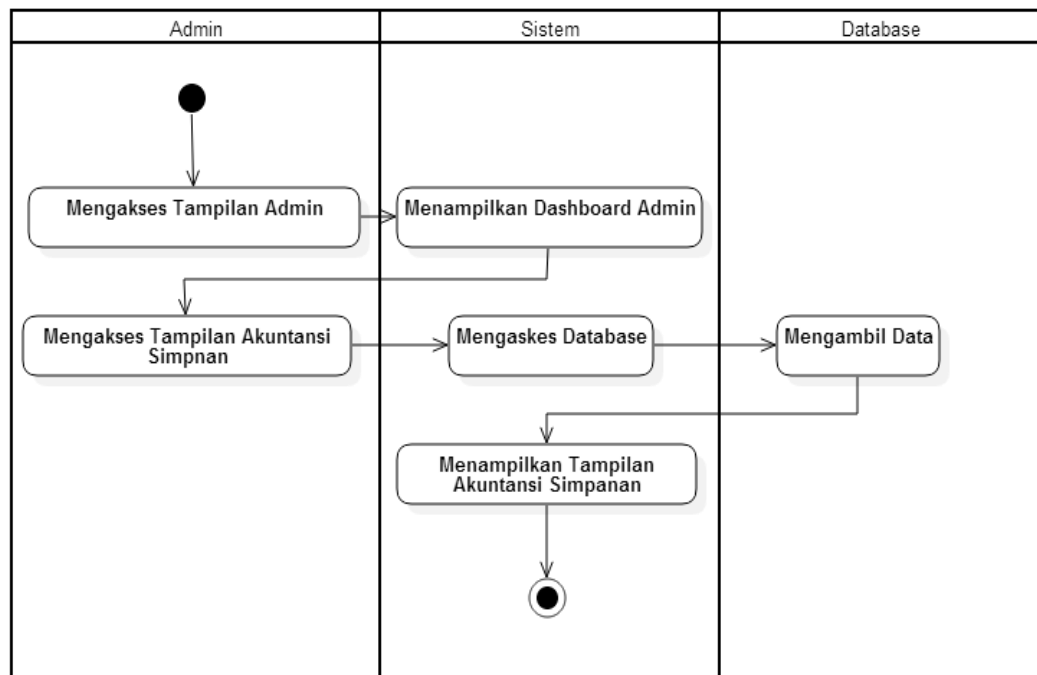
o. *Activity Diagram* Melihat Arus Kas

Pada gambar 3.17 dapat dijelaskan bahwa admin dapat melihat dan mencetak laporan arus kas. Pertama admin mengakses tampilan admin dan sistem akan menampilkan *dashboard* admin, kemudian admin mengakses arus kas dan sistem menampilkan arus kas. Lalu admin memilih periode dan sistem akan mengakses *database* untuk mengambil data di *database* dan menampilkan di sistem, kemudian admin memilih ingin mencetak laporan arus kas atau tidak.



Gambar 3.17 Activity Diagram Melihat Arus Kas.

p. Activity Diagram Melihat Akuntansi Simpanan

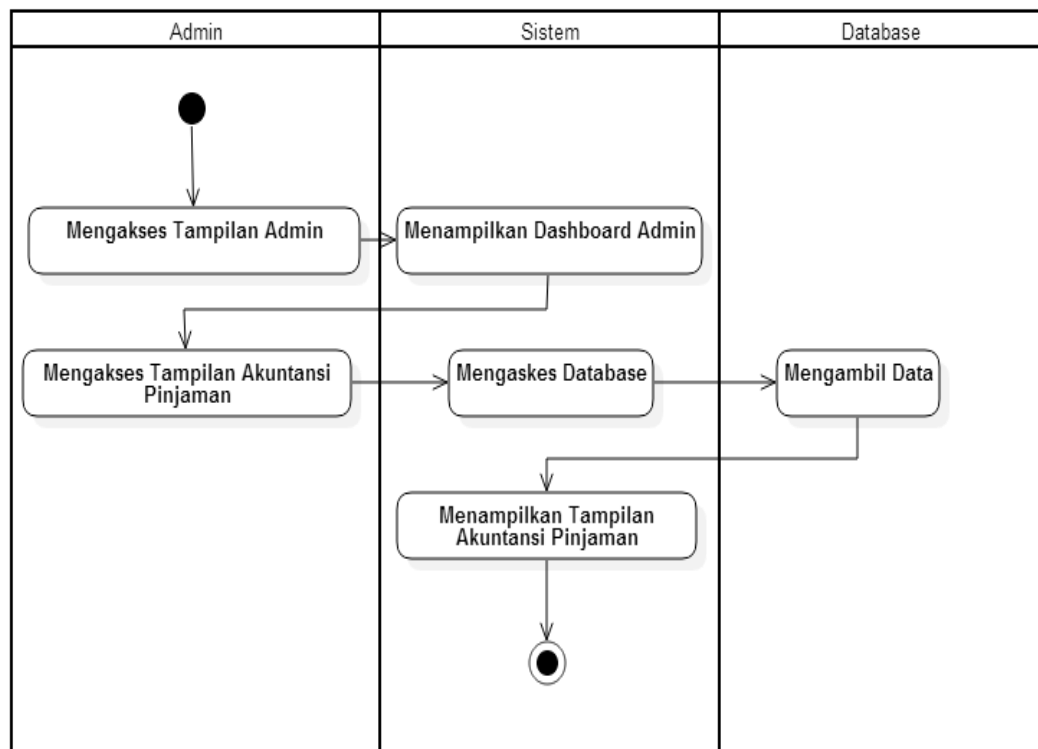


Gambar 3.18 Activity Diagram Melihat Simpanan.

Pada gambar 3.18 dapat dijelaskan bahwa admin dapat melihat simpanan. Pertama admin mengakses tampilan admin dan sistem akan menampilkan *dashboard* admin, kemudian admin mengakses tampilan akuntansi simpanan. Lalu sistem mengakses *database* untuk mengambil data di *database* dan menampilkan akuntansi simpanan pada sistem.

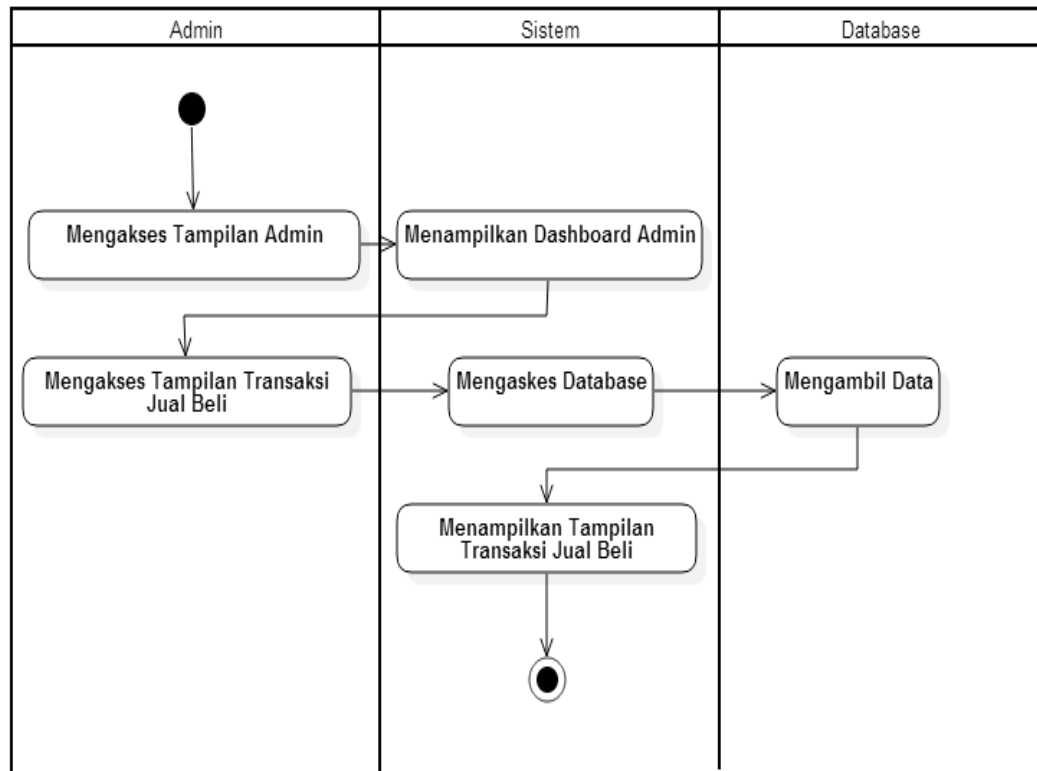
q. *Activity Diagram* Melihat Akuntansi Pinjaman

Pada gambar 3.19 dapat dijelaskan bahwa admin dapat melihat pinjaman. Pertama admin mengakses tampilan admin dan sistem akan menampilkan *dashboard* admin, kemudian admin mengakses tampilan akuntansi pinjaman. Lalu sistem mengakses *database* untuk mengambil data di *database* dan menampilkan akuntansi pinjaman pada sistem.



Gambar 3.19 *Activity Diagram* Melihat Pinjaman.

## r. Activity Diagram Melihat Transaksi Jual Beli



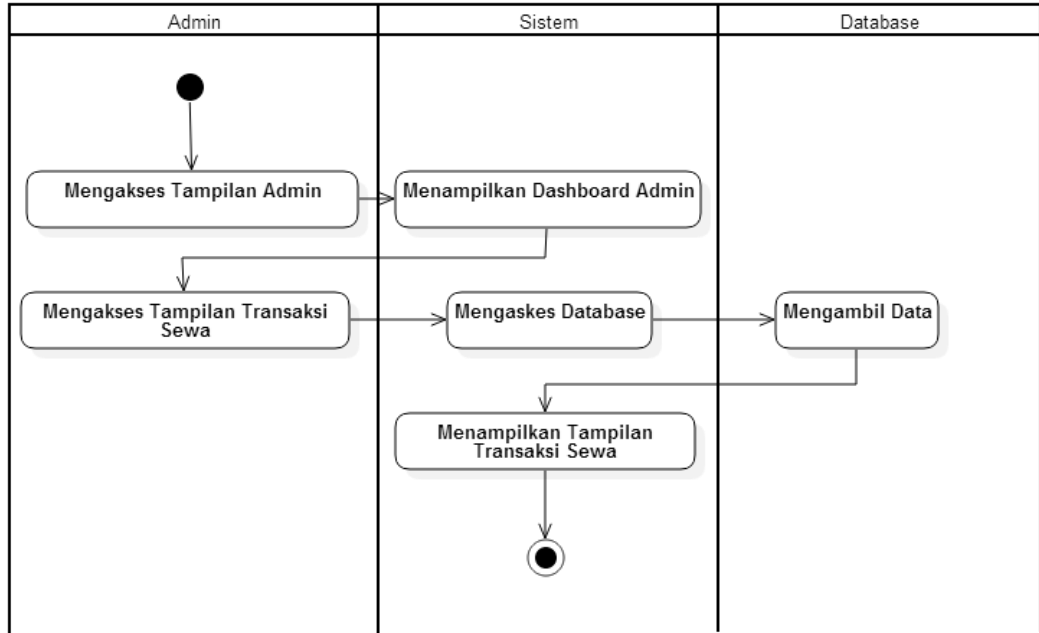
Gambar 3.20 Activity Diagram Melihat Transaksi Jual Beli.

Pada gambar 3.20 dapat dijelaskan bahwa admin dapat melihat pinjaman koperasi. Pertama admin mengakses tampilan admin dan sistem akan menampilkan *dashboard* admin, kemudian admin mengakses tampilan akuntansi jual beli. Lalu sistem mengakses *database* untuk mengambil data di *database* dan sistem menampilkan akuntansi jual beli pada sistem.

## s. Activity Diagram Melihat Transaksi Sewa

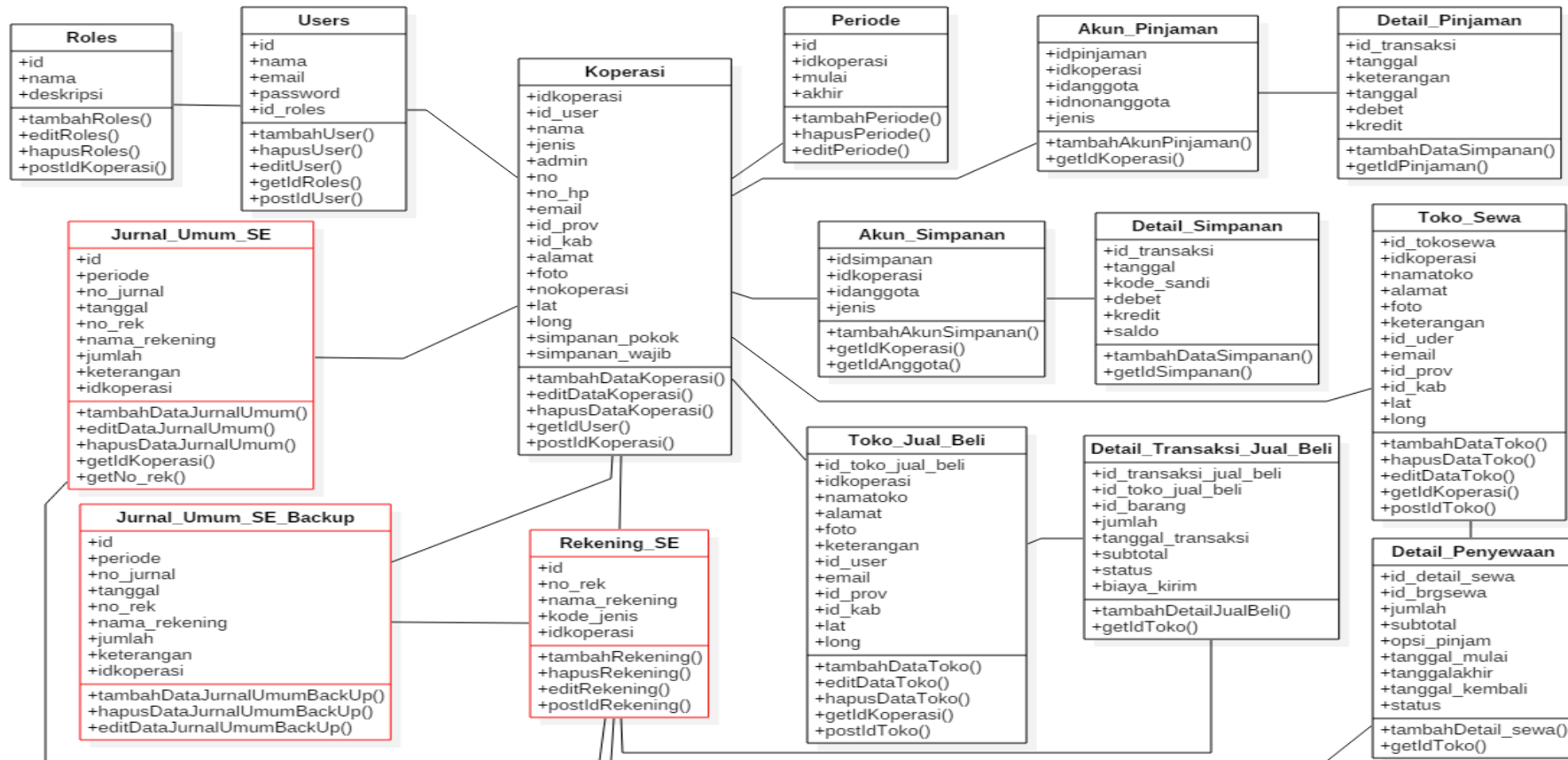
Pada gambar 3.21 dapat dijelaskan bahwa admin dapat melihat transaksi sewa. Pertama admin mengakses tampilan admin dan sistem akan menampilkan *dashboard* admin, kemudian admin mengakses tampilan

transaksi sewa. Lalu sistem mengakses *database* untuk mengambil data di *database* dan menampilkan transaksi sewa pada sistem.



Gambar 3.21 *Activity Diagram* Melihat Transaksi Sewa.

## 2. Class Diagram



Gambar 3.22 Class Diagram Sistem Informasi Akuntansi.



Pada gambar 3.22 dapat dijelaskan bahwa terdapat *class diagram* yang akan dibuat menjadi sistem informasi akuntansi, kelas-kelas saling berkaitan dengan modul lain seperti Jual beli, Simpan pinjam, Penyewaan dan Koperasi. Sedangkan pada *class* yang berwarna merah adalah *class* tambahan baru dalam tahap pengembangan fitur akuntansi *single entry*.

Pada tabel rekening\_SE terdapat (id, no rek, nama rek, kode jenis dan idkoperasi) yang dimana nantinya akan berhubungan dengan tabel koperasi dan tabel jurnal\_umum\_SE yang berisi (id, periode, no jurnal, tanggal, no rek, nama rek, jumlah, ket dan idkoperasi) yang berhubungan juga dengan tabel koperasi dan tabel jurnal\_umum\_SE\_backup yang berisi (id, periode, no jurnal, tanggal, no rek, nama rek, jumlah, ket dan idkoperasi) yang juga berhubungan dengan tabel koperasi. Data yang diolah oleh sistem ini didapatkan dari pencatatan proses bisnis koperasi yang menggunakan sistem informasi akuntansi.

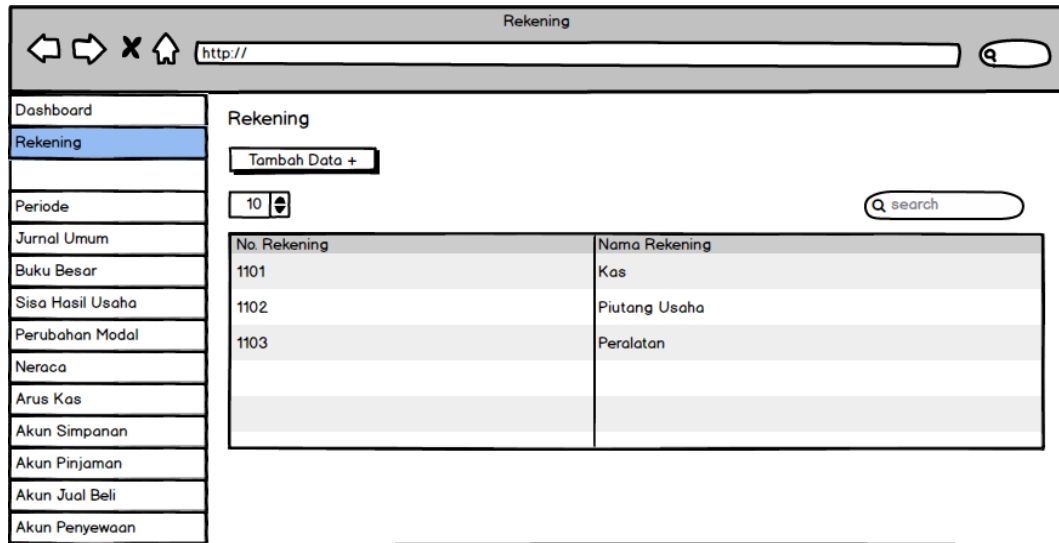
### 3.4.2 Perancangan Antarmuka

Perancangan antarmuka merupakan tahapan penggambaran bagaimana sebuah tampilan (*interface*) dari sistem dibuat.

#### A. Rancangan *Interface* pada Halaman Admin

##### 1. Rancangan *Interface* Rekening

Gambar 3.23 adalah rancangan dari tampilan rekening yang menampilkan nomor rekening, nama rekening, dan jenis akun pada koperasi. Admin juga dapat menambahkan data rekening dan mencari rekening.



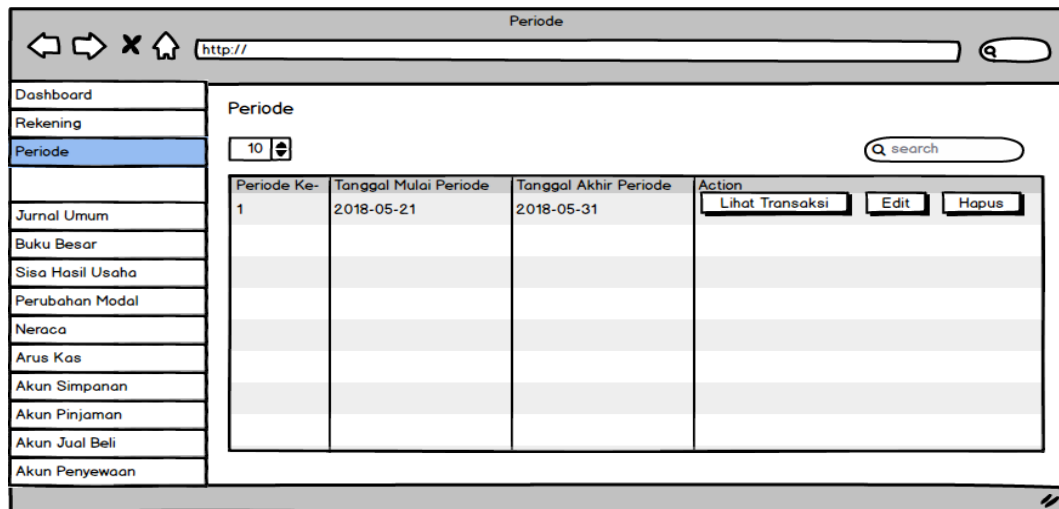
Gambar 3.23 Rancangan *Interface* Rekening.

## 2. Rancangan *Interface Form* Rekening

Gambar 3.24 adalah tampilan *form* tambah rekening yang berguna untuk menambahkan data-data rekening yang akan diperlukan koperasi.

Gambar 3.24 Rancangan *Interface Form* Rekening.

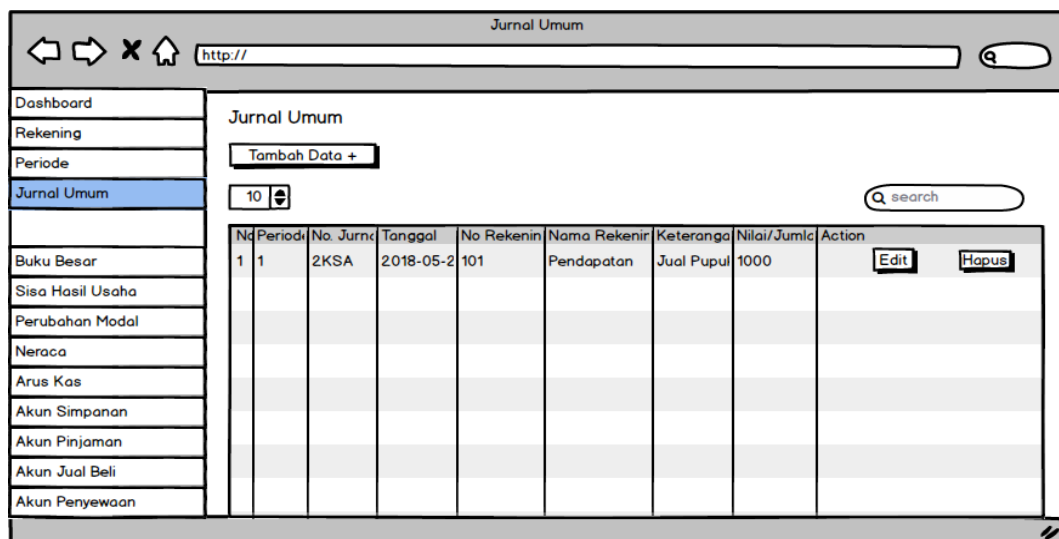
### 3. Rancangan *Interface* Periode



Gambar 3.25 Rancangan *Interface* Periode.

Pada Gambar 3.25 berguna untuk melihat data transaksi, edit dan hapus dalam suatu periode akuntansi tertentu dengan mengklik pada bagian kanan.

### 4. Rancangan *Interface* Jurnal Umum *Single Entry*



Gambar 3.26 Rancangan *Interface* Jurnal Umum *Single Entry*.

Pada Gambar 3.26 sistem menampilkan data transaksi harian yang dilakukan oleh koperasi tersebut dan juga dapat diedit dan hapus data pada jurnal umum. Pengguna juga dapat menambahkan data jurnal umum dengan klik pada bagian tambah data.

## 5. Rancangan *Interface Form* Tambah Data Jurnal Umum *Single Entry*

Gambar 3.27 Rancangan *Interface* Tambah Data Jurnal Umum *Single Entry*.

Pada Gambar 3.27 adalah rancangan tampilan dari jurnal umum *single entry* yang hanya ada satu kolom inputan untuk menambahkan data-data jurnal umum, yaitu debit (kas masuk) dan kredit (kas keluar). Sebelum menambahkan data jurnal umum, periksa periodenya terlebih dahulu. Jika terdapat peringatan periode kosong maka harus mengisi periode terlebih dahulu setelah itu dapat mengisi jurnal umum.

## 6. Rancangan *Interface* Buku Besar

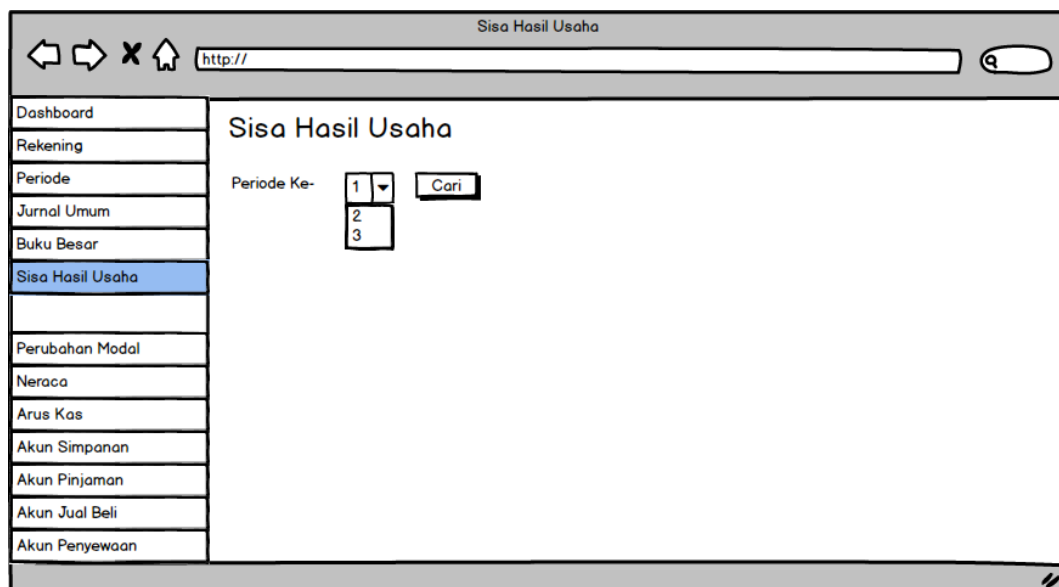
No. Jurnal	Tanggal	Keterangan	No. Rekening	Nama Rekening	Debet (Kas Masuk)	Kredit (Kas Keluar)
2KSA-1-1	1 Februari 2019	Jual Pupuk	1101	Pendapatan	1.000.000	
2KSA-1-2	4 Februari 2019	Bayar Bank	1301	Hutang		1500.000

Gambar 3.28 Rancangan *Interface* Buku Besar.

Pada Gambar 3.28 adalah rancangan tampilan dari buku besar untuk menampilkan seluruh data transaksi yang telah dipindahkan dari jurnal umum dengan debit sebagai (kas masuk) dan kredit (kas keluar) dan dapat melihat data transaksi sesuai dengan kelompok akun yang dipilih.

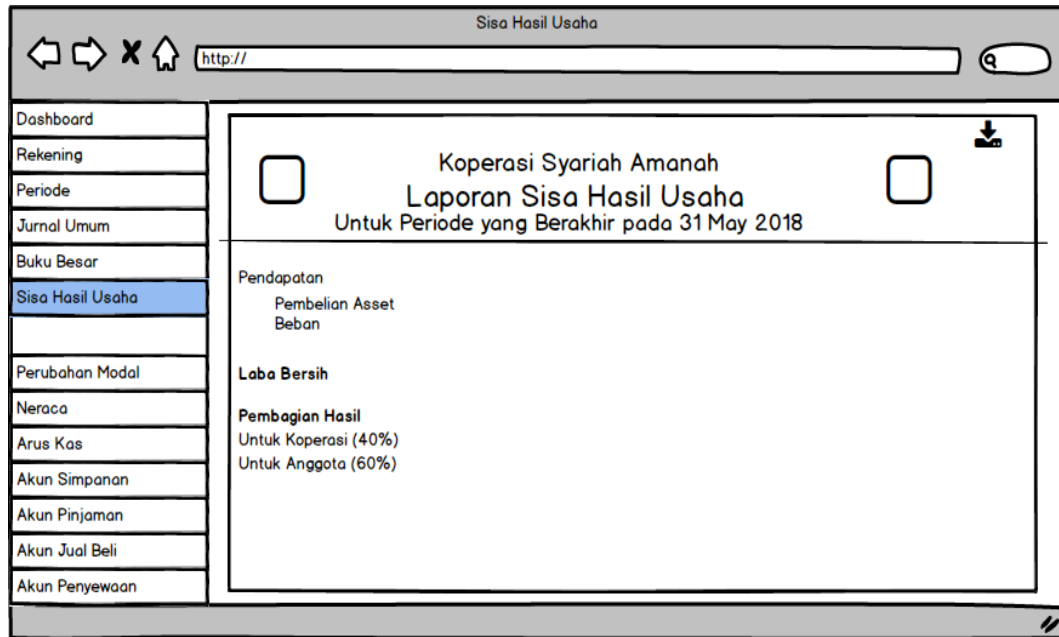
## 7. Rancangan *Interface* Laporan Sisa Hasil Usaha

Pada gambar 3.29 adalah rancangan untuk sisa hasil usaha dengan debit sebagai (kas masuk) dan kredit (kas keluar). Sebelum menampilkan laporan SHU, pilih periode yang ingin ditampilkan terlebih dahulu.



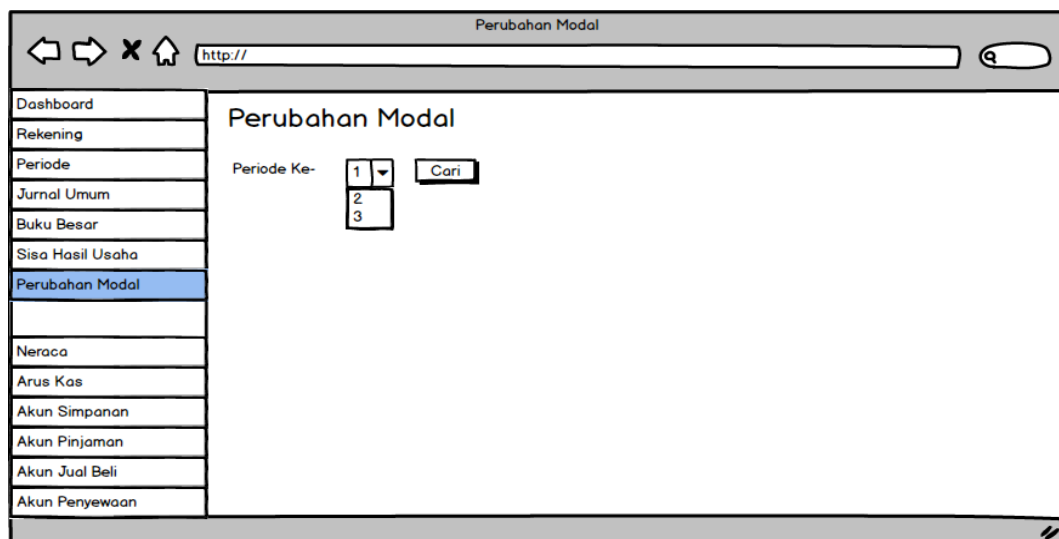
Gambar 3.29 Rancangan *Interface* Sisa Hasil Usaha.

Pada Gambar 3.30 akan menampilkan laporan sisa hasil usaha. Sebelumnya, harus memilih laporan sisa hasil usaha pada periode berapa yang ingin dilihat dan dicetak.



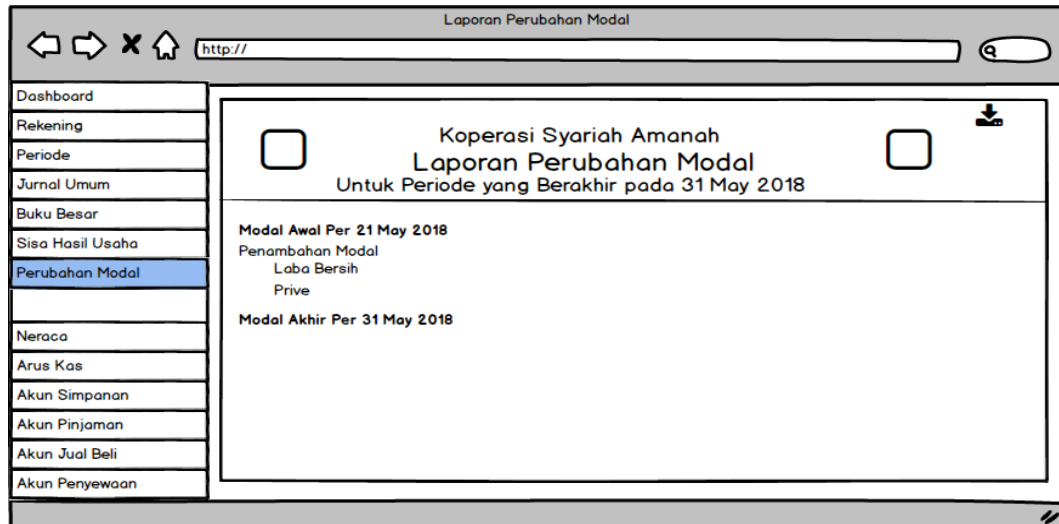
Gambar 3.30 Rancangan *Interface* Laporan Sisa Hasil Usaha.

## 8. Rancangan *Interface* Laporan Perubahan Modal



Gambar 3.31 Rancangan *Interface* Perubahan Modal.

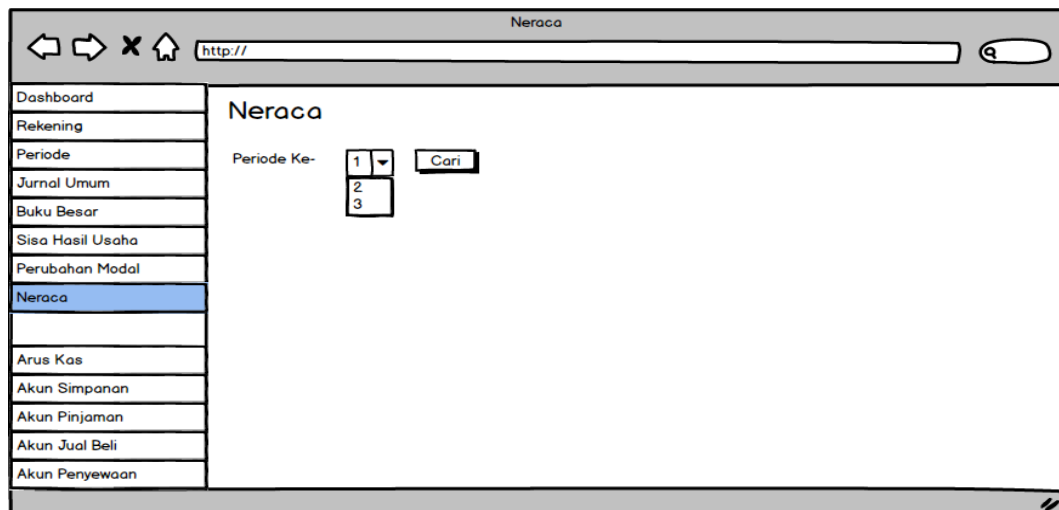
Pada gambar 3.31 koperasi memilih terlebih dahulu periode berapa yang akan dilihat perubahan modalnya, lalu klik cari data.



Gambar 3.32 Rancangan *Interface* Laporan Perubahan Modal.

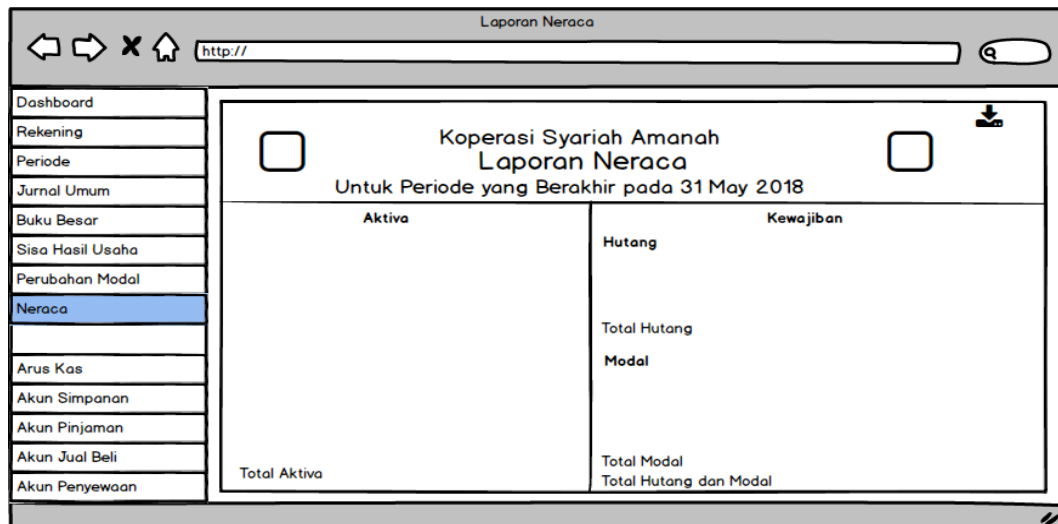
Pada Gambar 3.32 akan menampilkan laporan perubahan modal. Sebelumnya, harus memilih laporan perubahan modal pada periode berapa yang ingin dilihat dan dicetak, maka sistem akan otomatis mencetak file pdf.

## 9. Rancangan *Interface* Laporan Neraca



Gambar 3.33 Rancangan *Interface* Neraca.

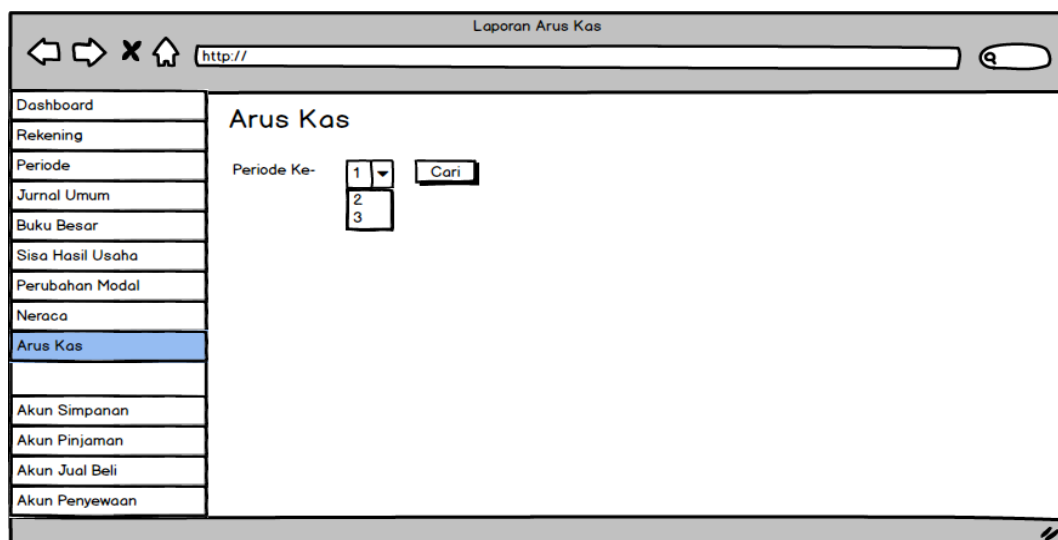
Pada gambar 3.33 adalah rancangan tampilan untuk neraca. Sebelum menampilkan laporan neraca, pilih periode yang ingin ditampilkan terlebih dahulu.



Gambar 3.34 Rancangan *Interface* Laporan Neraca.

Pada Gambar 3.34 menampilkan laporan neraca. Sebelumnya, harus memilih laporan neraca pada periode berapa yang ingin dilihat dan dicetak, maka sistem akan otomatis mencetak file pdf.

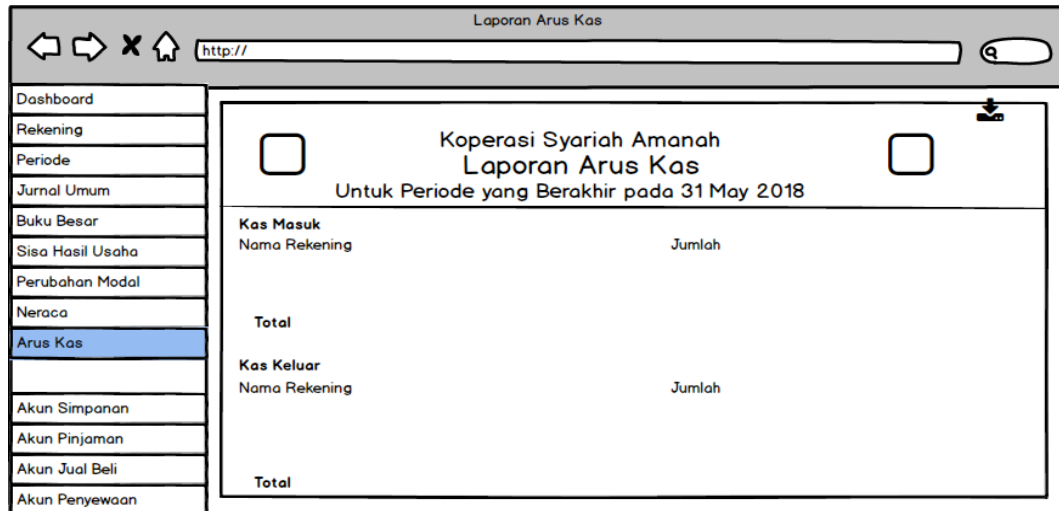
## 10. Rancangan *Interface* Laporan Arus Kas



Gambar 3.35 Rancangan *Interface* Arus Kas.

Pada gambar 3.35 adalah rancangan tampilan untuk arus kas. Sebelum menampilkan laporan arus kas, pilih periode yang ingin ditampilkan terlebih dahulu.

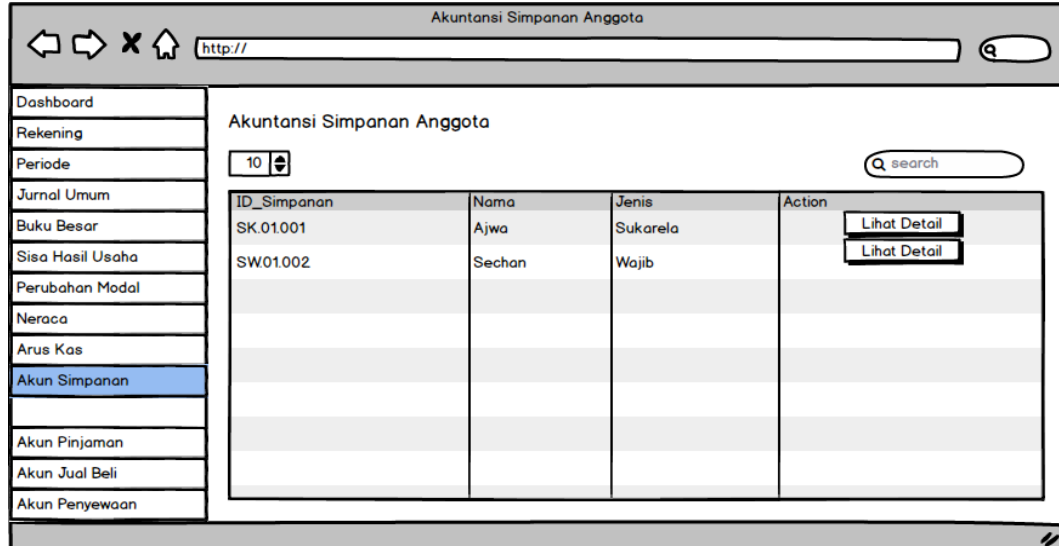




Gambar 3.36 Rancangan *Interface* Laporan Arus Kas.

Pada Gambar 3.36 menampilkan laporan arus kas. Sebelumnya, harus memilih arus kas pada periode berapa yang ingin dilihat dan dicetak.

## 11. Rancangan *Interface* Akuntansi Simpanan



Gambar 3.37 Rancangan *Interface* Akuntansi Simpanan.

Pada Gambar 3.37 tampilan yang menampilkan data anggota yang telah melakukan simpanan dan dapat melihat detail simpanan pelanggan.

## 12. Rancangan *Interface* Akuntansi Pinjaman

No. Transaksi	Nama	Jumlah	Angsuran	Jenis	Sisa Pinjaman	Action
PP.01.001	Fawwaz	10.000.000	12	Keperluan Pribadi	9.500.000	

Gambar 3.38 Rancangan *Interface* Akuntansi Pinjaman.

Pada Gambar 3.38 tampilan yang menampilkan data anggota yang telah melakukan pinjaman.

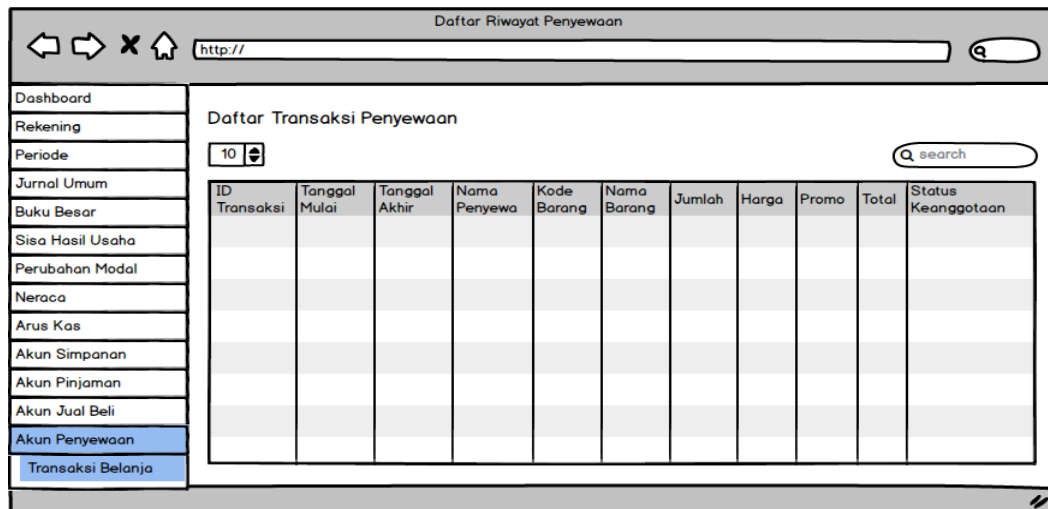
## 13. Rancangan *Interface* Daftar Transaksi Jual Beli

ID Transaksi	Tanggal	Nama Pembeli	Kode Barang	Nama Barang	Jumlah	Harga	Promo	Total	Status Keanggotaan

Gambar 3.39 Rancangan *Interface* Daftar Transaksi Jual Beli.

Pada Gambar 3.39 menampilkan data-data transaksi jual beli yang telah dilakukan oleh koperasi tersebut.

#### 14. Rancangan *Interface* Daftar Transaksi Sewa

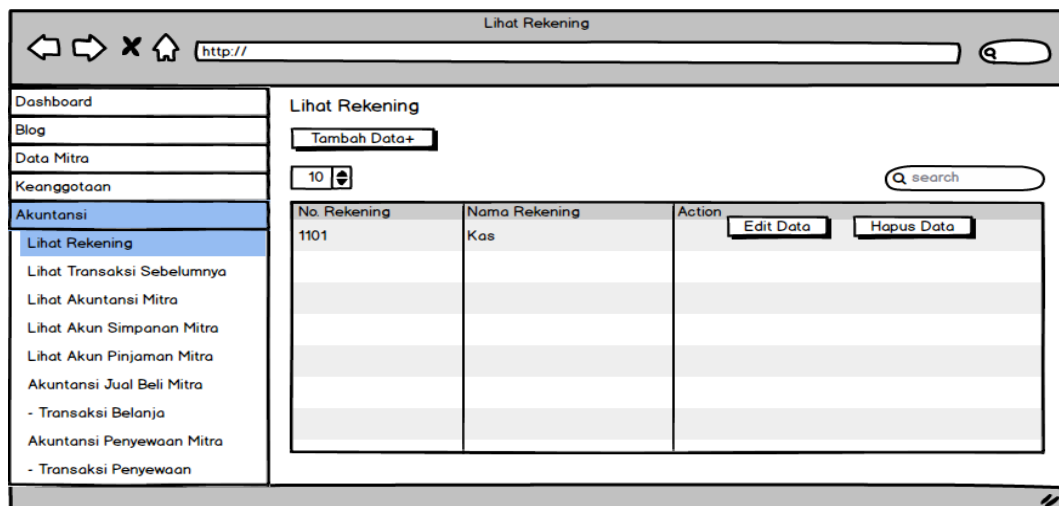


Gambar 3.40 Rancangan *Interface* Daftar Transaksi Sewa.

Pada Gambar 3.40 menampilkan data-data transaksi sewa yang telah dilakukan oleh koperasi tersebut.

### B. Rancangan *Interface* pada Halaman Super Admin

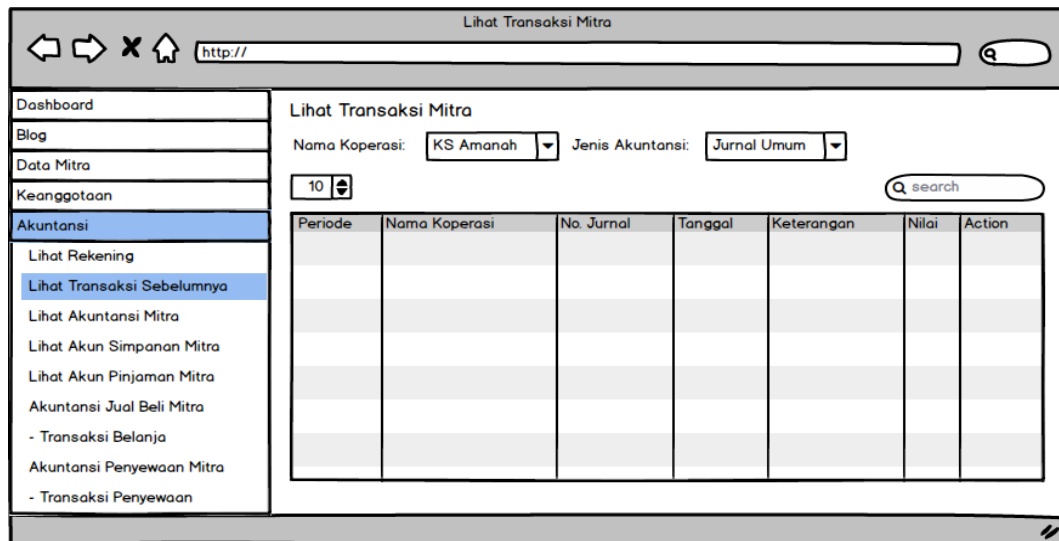
#### 1. Rancangan *Interface* Lihat Rekening



Gambar 3.41 Rancangan *Interface* Lihat Rekening.

Pada Gambar 3.41 adalah rancangan dari tampilan rekening yang menampilkan data-data nomor rekening, nama rekening, dan jenis akun pada koperasi tersebut. Super admin juga dapat mengedit dan menghapus data rekening.

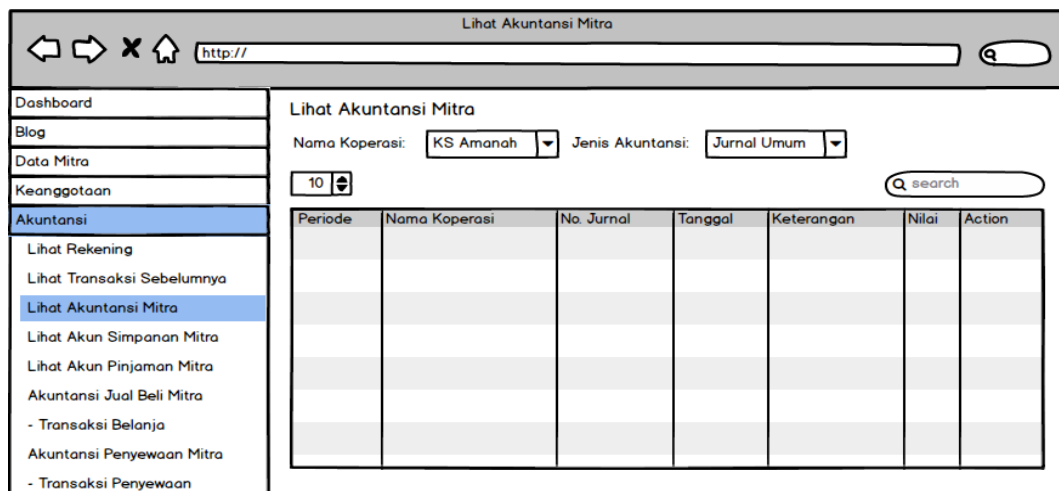
## 2. Rancangan *Interface* Lihat Transaksi Sebelumnya



Gambar 3.42 Rancangan *Interface* Lihat Transaksi Sebelumnya.

Pada Gambar 3.42 adalah rancangan dari tampilan lihat transaksi mitra berguna untuk menampilkan transaksi-transaksi yang telah dilakukan selama koperasi beroperasi dengan memilih nama koperasi dan jenis akuntansinya.

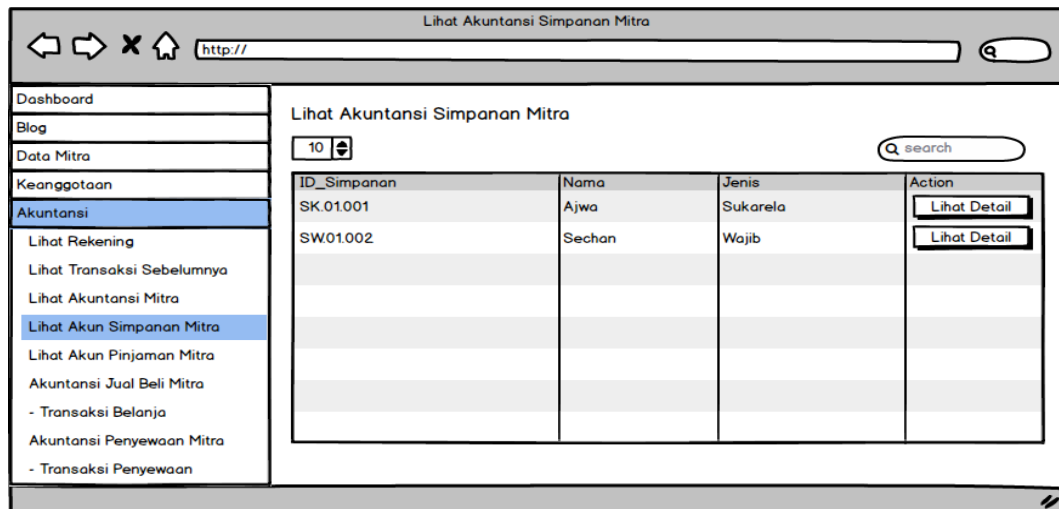
## 3. Rancangan *Interface* Lihat Akuntansi Mitra



Gambar 3.43 Rancangan *Interface* Lihat Akuntansi Mitra.

Pada Gambar 3.43 adalah rancangan dari tampilan lihat akuntansi mitra berguna untuk menampilkan seluruh akuntansi yang dilakukan oleh koperasi mitra seperti, jurnal umum, buku besar, dan lain-lain.

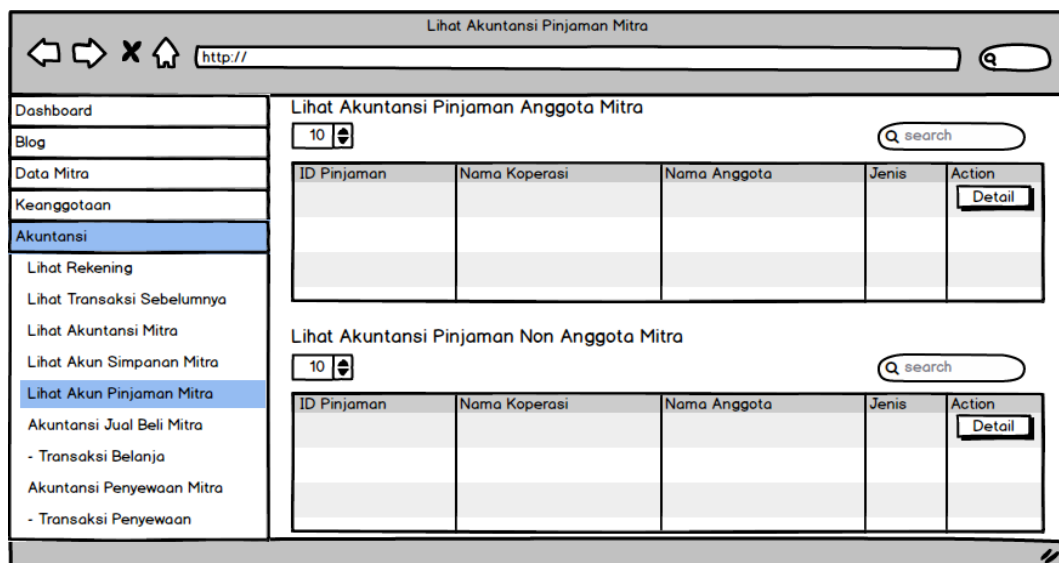
#### 4. Rancangan *Interface* Lihat Akuntansi Simpanan Mitra



Gambar 3.44 Rancangan *Interface* Lihat Akuntansi Simpanan Mitra.

Pada Gambar 3.44 untuk melihat siapa saja yang melakukan simpanan pada koperasi mitra dan dapat melihat detail simpanan mitra tersebut.

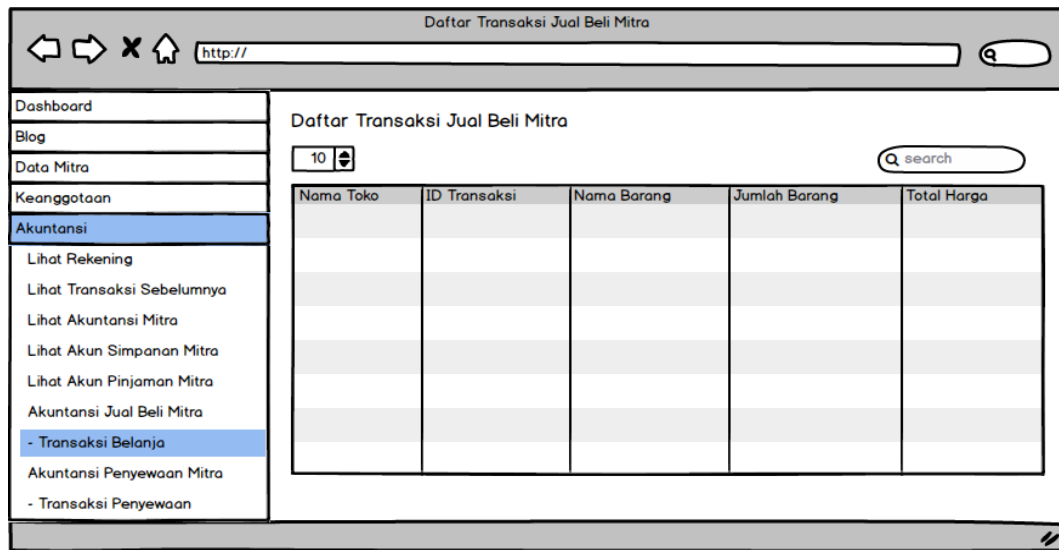
#### 5. Rancangan *Interface* Lihat Akuntansi Pinjaman Mitra



Gambar 3.45 Rancangan *Interface* Lihat Akuntansi Pinjaman Mitra.

Pada Gambar 3.45 berguna untuk melihat siapa saja yang melakukan pinjaman pada koperasi mitra dan dapat melihat detail pinjaman mitra koperasi.

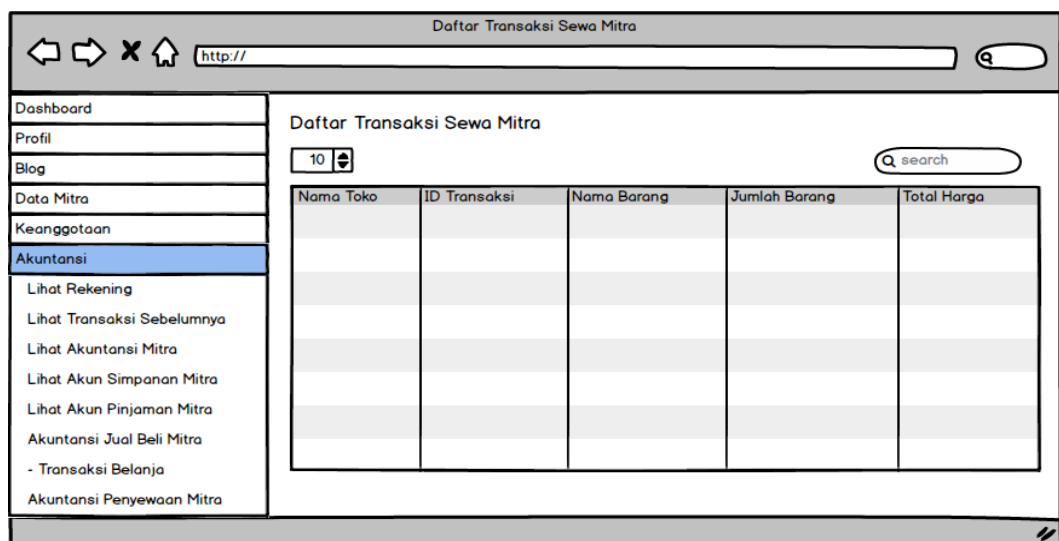
## 6. Rancangan *Interface* Lihat Transaksi Jual Beli Mitra



Gambar 3.46 Rancangan *Interface* Lihat Daftar Transaksi Jual Beli Mitra.

Pada Gambar 3.46 berguna untuk melihat daftar transaksi jual beli.

## 7. Rancangan *Interface* Lihat Transaksi Sewa Mitra



Gambar 3.47 Rancangan *Interface* Lihat Transaksi Sewa Mitra.

Pada Gambar 3.47 berguna untuk melihat transaksi sewa mitra koperasi.

### **3.5 Penulisan Kode Program**

Penulisan kode program menggunakan bahasa pemrograman PHP dan *Framework* Laravel 5.5.

### **3.6 Pengujian Program**

Pengujian sistem akan dilakukan *black box testing* dan *expert judgment*.

#### **3.6.1 Black Box Testing**

*Black box testing* akan dilakukan dalam pengujian Sistem Informasi Akuntansi *Single Entry*. Pengujian yang dilakukan terhadap fungsi-fungsi dari setiap komponen yang ada pada sistem.

#### **3.6.2 Expert Judgment**

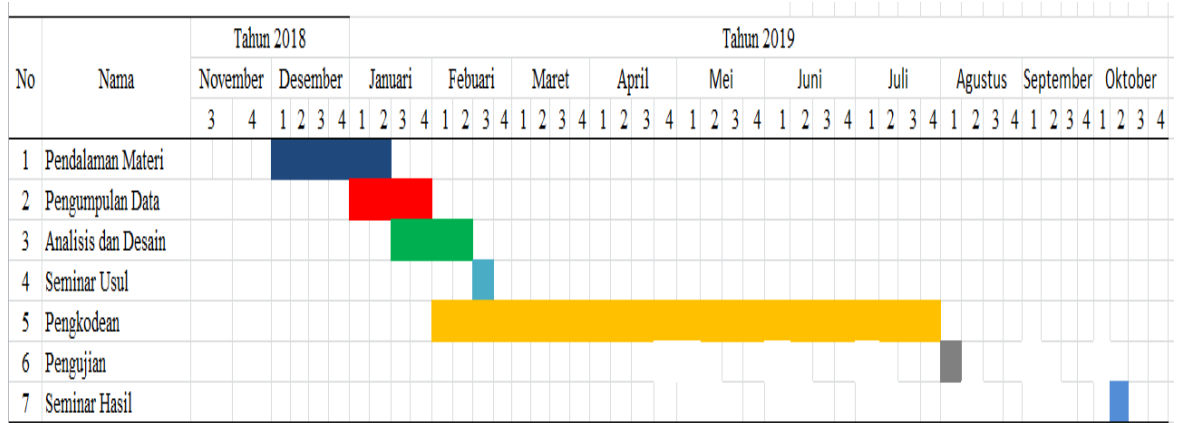
Pengujian *expert judgement* yang dilakukan adalah dengan melakukan presentasi SIA yang akan dibangun kepada pihak koperasi. Koperasi yang menjadi sampel adalah Koperasi Syariah Bina Insan Mandiri yang menjadi penguji dan *expert judgement* pada penelitian sebelumnya.

### **3.7 Penyerahan Sistem**

Setelah dilakukannya tahapan testing kemudian sistem dipublikasi dan diserahkan kepada pihak koperasi untuk digunakan.

### **3.8 Jadwal Penelitian**

Pada gambar 3.49 penelitian ini akan dilakukan sesuai jadwal yang sudah diatur dalam bentuk gantt chart. Dimulai pada bulan Desember 2018 dan berakhir pada bulan Oktober 2019.



Gambar 3.49 Gantt Chart.



## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Dari hasil penelitian yang dilakukan dapat disimpulkan sebagai berikut.

1. Sistem Informasi Akuntansi *Single Entry* telah selesai dibangun. Sistem ini dikembangkan menggunakan *framework* Laravel 5.5 dan *database server* *MySQL*. Sistem ini telah berhasil diuji dalam pencatatan proses bisnis koperasi mulai dari jurnal umum hingga laporan keuangan dalam bentuk pdf.
2. Sistem informasi akuntansi telah berhasil mencatat peroses bisnis koperasi seperti pembukuan keuangan. Fitur jurnal umum akan mencatat seluruh transaksi koperasi yang diisi oleh admin dan fitur buku besar yang menampilkan seluruh data transaksi beserta jumlah debit dan kredit.
3. Sisem informasi akuntansi telah berhasil membuat laporan keuangan berupa arus kas, perubahan modal, neraca dan sisa hasil usaha.

#### **5.2 Saran**

Berdasarkan analisis, perancangan dan implementasi sistem yang dilakukan, maka ada beberapa saran yang perlu diperhatikan dalam mengembangkan sistem ini adalah sebagai berikut.

1. Dibuatkan versi Android atau iOS sehingga lebih fleksibel dan dapat mempermudah koperasi dalam melakukan proses bisnisnya.

## DAFTAR PUSTAKA

- Abdul-Rahamon, O. A., dan Adejare, A. T. (2014). The analysis of the impact of accounting records keeping on the performance of the small scale enterprises. *International Journal of Academic Research in Business and Social Sciences*, 4(1), 1–17. <https://doi.org/10.6007/IJARBSS/v4-i1/506>
- Aminuddin, A. (2015). *Cara Efektif Belajar Framework Laravel*. Yogyakarta: Lokomedia.
- Halim, A. (2010). *Manajemen Keuangan Daerah Edisi Revisi*. Yogyakarta: UPP AMP YPKN.
- Hall, J. A. (2010). *Accounting Information Systems* (17th ed.). United States of America: South Westren Cengage Learning.
- Harahap, S. S. (2007). *Teori Akuntansi*. Jakarta: Raja Grafindo.
- HD, S. (2009). *Modul Akuntansi Dasar* (Free Edition). Surakarta: Sebelas Maret University.
- IASB. (2008). *Exposure Draft of Improved Conceptual Framework for Financial Reporting*. International Accounting Standard Board. London: International Accounting Standards Board.
- Iman, F. M. (2018). Sistem Informasi Multi Koperasi Menggunakan Framework Laravel, 6(2), 43–53.
- Laravel Is A Trademark Of Taylor Otwell. (n.d.). Introduction - Laravel - The PHP Framework For Web Artisans. <https://laravel.com/docs/4.2/introduction>

- Nancy A. Bagranoff, Mark G. Simkin, C. S. N. (2009). *Accounting Information Systems* (11th ed.). United States of America: JOHN WILEY dan SONS, INC.
- Ne, I. A. Ā., Połaka, G., dan Ruža, O. (2015). Calculation Of Financial Indicators In A Single-Entry Accounting System *Iluta*, 1(1), 17–27.
- Nugroho, A. (2009). *Rekayasa Perangkat Lunak Menggunakan UML dan Java*. Bandung: Informatika Bandung.
- Object Management Group. (2005). What is UML | Unified Modeling Language. Retrieved from <http://www.uml.org/what-is-uml.htm>
- PMI. (2013). *A Guide to the Project Management Body of Knowledge - Fifth Edition*. Project Management Institute (Vol. 5). Pennsylvania: Project Management Institute, Inc. <https://doi.org/10.1002/pmj.20125>
- Prasetyo, W. (2013). Membongkar Akuntansi Double Entry Systems. *Jurnal Akuntansi Multiparadigma*, 4(2), 308–321.
- Pressman, R. S. (2009). *Web Engineering A Practitioner's Approach*. Icw (1st ed., Vol. 3140). New York: McGraw-Hill. <https://doi.org/10.1007/b99180>
- Pressman, R. S. (2012). *Software Engineering A Practitioner's Approach 7th Edition*. New York: McGraw-Hill. <https://doi.org/10.1126/science.195.4283.1199>
- Rahmanda, R. A. (2018). Sistem Informasi Akuntansi pada Multi Koperasi Menggunakan Framework Laravel, 6(2), 54–63.
- Romney, M. B., dan Steinbart, P. J. (2016). *Accounting Information Systems*. New York: McGraw-Hill.
- Satzinger, J. W., Jackson, R., dan Burd, S. D. (2012). *Systems Analysis and*

- Design in a Changing World*. Cengage Learning: ISBN 978-1-305-11720-4.
- Soyinka, K. A., Fagbayimu, M. O., Adegoye, E., dan Ogunmola, J. O. (2017).  
Decision Usefulness and Financial Reporting : The General Public  
Perspective, 7(4), 160–168. <https://doi.org/10.6007/IJARAFMS/v7-i4/3470>
- Sukanto, R. A., dan Shalahuddin, M. (2013). *Rekayasa Perangkat Lunak*.  
Bandung: Informatika.
- Susanto, A. (2017). *Sistem Informasi Akuntansi: Pemahaman Konsep Secara  
Terpadu*. Bandung: Lingga Jaya.
- The PHP Group. (2017). PHP: What is PHP? - Manual.  
<https://doi.org/10.1016/j.physb.2007.10.009>
- Tungadi, A. L. (2013). *Sistem informasi akuntansi pada koperasi Universitas  
Atma Jaya Makassar*. Makasar: Universitas Atma Jaya.