BAB IV

ANALISIS DAN PERANCANGAN SISTEM

4.1 Analisis Sistem yang Sedang Berjalan

Analisis merupakan suatu kegiatan yang bertujuan untuk mempelajari serta mengevaluasi suatu bentuk permasalahan yang ada pada sebuah sistem. Dalam analisis sistem akan ditemukan masalah yang mungkin akan mempengaruhi kerja sistem. Agar sistem yang dirancang dapat berjalan sebagaimana mestinya, perlu dilakukan analisis terhadap kinerja sistem yang pada akhirnya bertujuan untuk pengembangan sistem.

Adapun sistem yang sedang berjalan pada Koperasi Simpan Pinjam Abadi Jaya sampai saat ini yaitu sistem informasi pengolahan data simpan pinjam yang berjalan pada koperasi tersebut masih dilakukan secara manual. Pengarsipan data anggota, pengarsipan data penyimpanan atau peminjaman, hingga pada proses pengarsipan data angsuran masih dilakukan dengan pencatatan manual dan proses penghitungan menggunakan kalkulator.

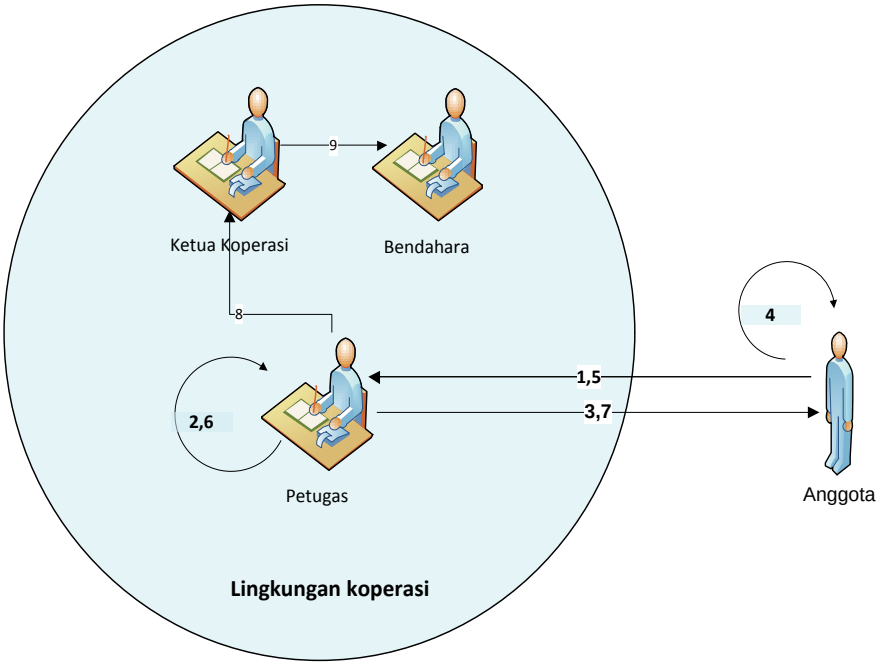
4.1.1 Analisis Prosedur yang Sedang Berjalan

Analisis prosedur merupakan proses analisa terhadap kegiatan-kegiatan yang dilakukan oleh Koperasi Simpan Pinjam Abadi Jaya dalam proses simpan-pinjam dan proses pengangsuran. Analisis terhadap prosedur yang sedang berjalan hendaknya perlu dilakukan terlebih dahulu sebelum dilakukannya perancangan terhadap sistem informasi yang akan diusulkan. Harus diketahui pula hal-hal yang

menjadi tujuan pemakai sehingga masalah tersebut dapat didefinisikan secara jelas.

4.1.1.1 Proses Bisnis Berjalan

Proses bisnis adalah suatu kumpulan aktivitas atau pekerjaan terstruktur yang saling terkait untuk menyelesaikan suatu masalah tertentu atau yang menghasilkan produk atau layanan. Proses bisnis penyimpanan yang sedang berjalan adalah sebagai berikut :



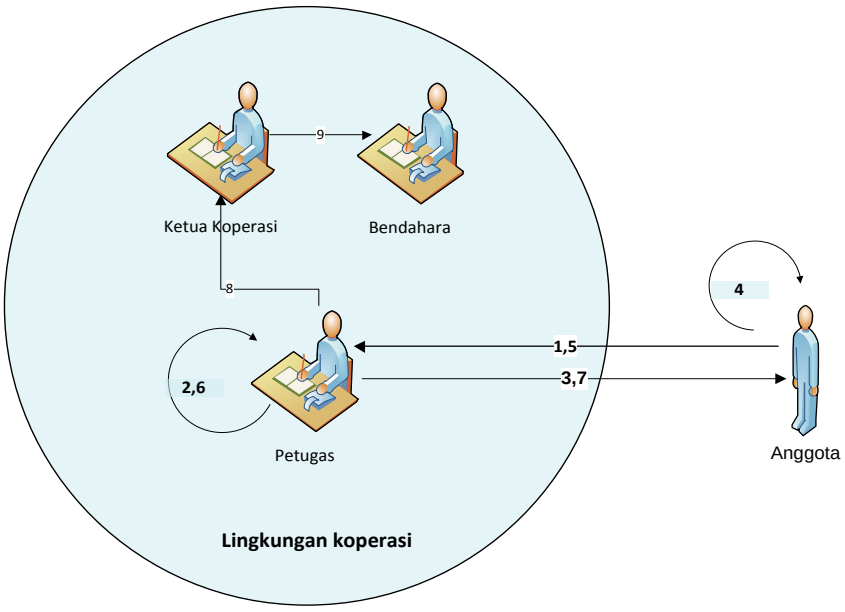
Gambar 4. 1 Proses bisnis penyimpanan yang sedang berjalan

Keterangan :

1. Anggota datang kepada petugas dan menunjukkan buku anggota miliknya.
2. Petugas memeriksa keaslian buku anggota milik anggota tersebut.
3. Jika buku tersebut asli, maka petugas memberikan formulir penyimpanan sebanyak 2 rangkap kepada anggota untuk diisi.

4. Anggota mengisi data penyimpanan pada formulir.
5. Anggota menyerahkan formulir pada petugas dan simpanan sesuai dengan yang tertulis pada formulir.
6. Petugas menyimpan 1 rangkap formulir untuk pengarsipan
7. Petugas menyerahkan salah 1 formulir kepada anggota sebagai tanda bukti.
8. Dari buku simpanan anggota, petugas dibuatkan laporan simpanan anggota 2 rangkap dan diserahkan kepada ketua koperasi simpan pinjam abadi jaya untuk divalidasi
9. Setelah divalidasi laporan akan diserahkan kepada bendahara untuk dijadikan arsip.

Proses bisnis pengambilan simpanan yang sedang berjalan adalah sebagai berikut:

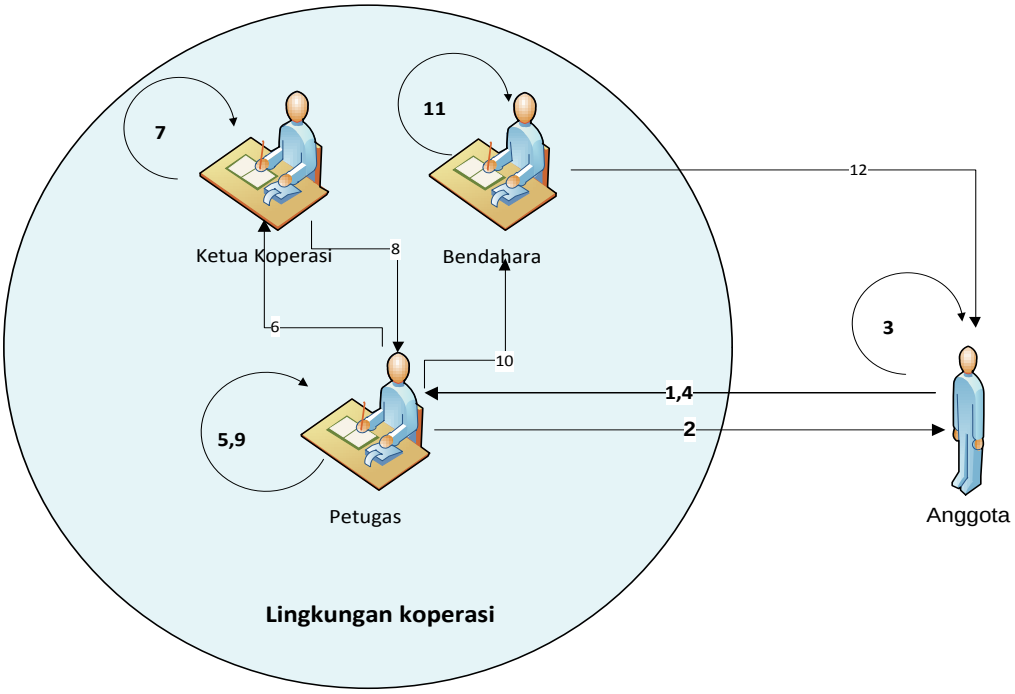


Gambar 4. 2 Proses bisnis pengambilan simpanan yang sedang berjalan

Keterangan :

1. Anggota datang kepada petugas dan menunjukkan buku anggota miliknya.
2. Petugas memeriksa keaslian buku anggota milik anggota tersebut dan memeriksa apakah anggota masih memiliki tunggakan atau tidak.
3. Jika buku tersebut asli dan anggota tidak memiliki tunggakan, maka petugas memberikan formulir pengambilan sebanyak 2 rangkap kepada anggota untuk diisi.
4. Anggota mengisi data pengambilan simpanan pada formulir.
5. Anggota menyerahkan formulir pada petugas.
6. Petugas menyimpan formulir data pengambilan simpanan sebagai bukti pengambilan
7. Petugas memberikan simpanan sesuai dengan formulir pengambilan.
8. Dari buku simpanan anggota, petugas dibuatkan laporan pengambilan simpanan anggota 2 rangkap dan diserahkan kepada ketua koperasi simpan pinjam abadi jaya untuk divalidasi
9. Setelah divalidasi laporan akan diserahkan kepada bendahara untuk dijadikan arsip.

Proses bisnis peminjaman yang sedang berjalan adalah sebagai berikut:



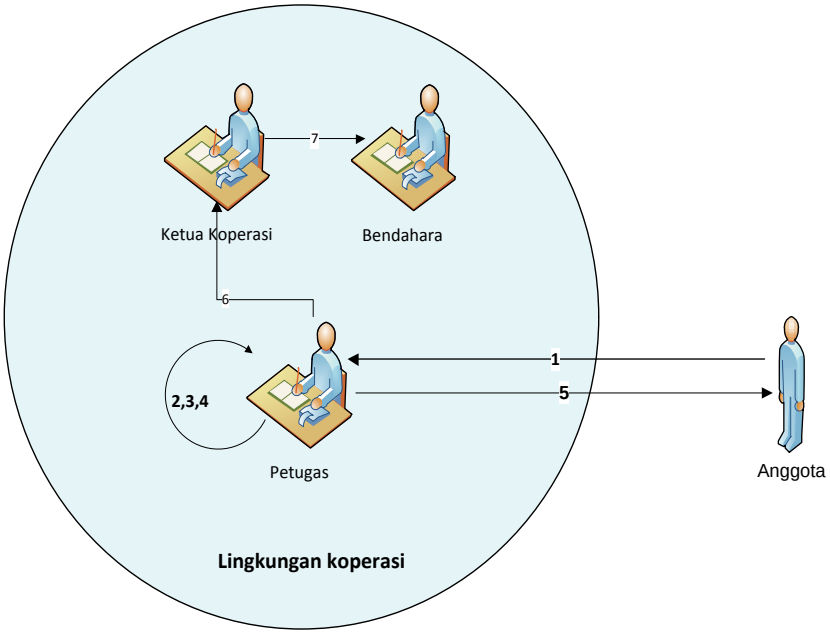
Gambar 4.3 Proses bisnis peminjaman yang sedang berjalan

Keterangan :

1. Anggota datang kepada petugas dan meminta formulir peminjaman.
2. Petugas menyerahkan formulir peminjaman.
3. Anggota mengisi data peminjaman pada formulir.
4. Anggota menyerahkan formulir peminjaman beserta buku keanggotaan pada petugas.
5. Petugas melakukan survey pemeriksaan jaminan
6. Petugas menyerahkan formulir pengajuan peminjaman pada ketua koperasi.
7. Ketua koperasi memeriksa dan menentukan persetujuan peminjaman.
8. Ketua menyerahkan data persetujuan pada petugas
9. Petugas mencatat pinjaman acc pada buku simpanan anggota

10. Petugas memberikan dokumen ke bagian bendahara
11. Dokumen di validasi oleh bendahara.
12. Bendahara menyerahkan 1 rangkap kepada anggota sebagai bukti peminjaman dan rangkap 2 dijadikan arsip oleh bagian bendahara

Proses bisnis pembayaran angsuran yang sedang berjalan adalah sebagai berikut:



Gambar 4.4 Proses bisnis pembayaran angsuran yang sedang berjalan

Keterangan :

1. Anggota datang kepada petugas dan menyerahkan buku anggota dan data angsuran serta angsuran.
2. Petugas memeriksa data pinjaman pada buku anggota.
3. Jika anggota benar memiliki pinjaman, maka petugas mencatat angsuran anggota pada buku angsuran.
4. Petugas membuat kwitansi angsuran.

5. Petugas menyerahkan kwitansi angsuran pada anggota.
6. Dari buku simpanan anggota,petugas dibuatkan laporanangsurandan denda anggota (jika ada) 2 rangkap dan diserahkan kepada ketua koperasi simpan pinjam abadi jaya untuk divalidasi
7. Setelah divalidasi laporan akan diserahkan kepada bendahara untuk dijadikan arsip.

Syarat dan ketentuan yang berlaku di Koperasi Simpan Pinjam Abadi Jaya adalah :

1. Syarat
   a. Syarat untuk menjadi anggota adalah harus menyerahkan photocopy KTP, photocopy Kartu Keluarga.
   b. Syarat jika anggota akan melakukan peminjaman adalah harus menyerahkan photocopy KTP, photocopy Kartu Keluarga, jaminan, dan juga harus memiliki jenis usaha yang jelas (terlihat jenis usahanya)
   c. Untuk jenis jaminan diantaranya : bpkb motor/mobil, sertifikat tanah, rumah, barang elektronik (Jika barang elektronik harus di sertakan kwitansi pembelian)
2. Ketentuan :
   a. Jika anggota pertama melakukan pinjaman, anggota tidak boleh meminjam lebih dari 1.000.000, anggota bisa menaikkan pinjamannya jika anggota telah melakukan pinjaman lebih dari 2 kali, dan batas maksimum pinjamannya 25.000.000 (pinjaman secara bertahap).

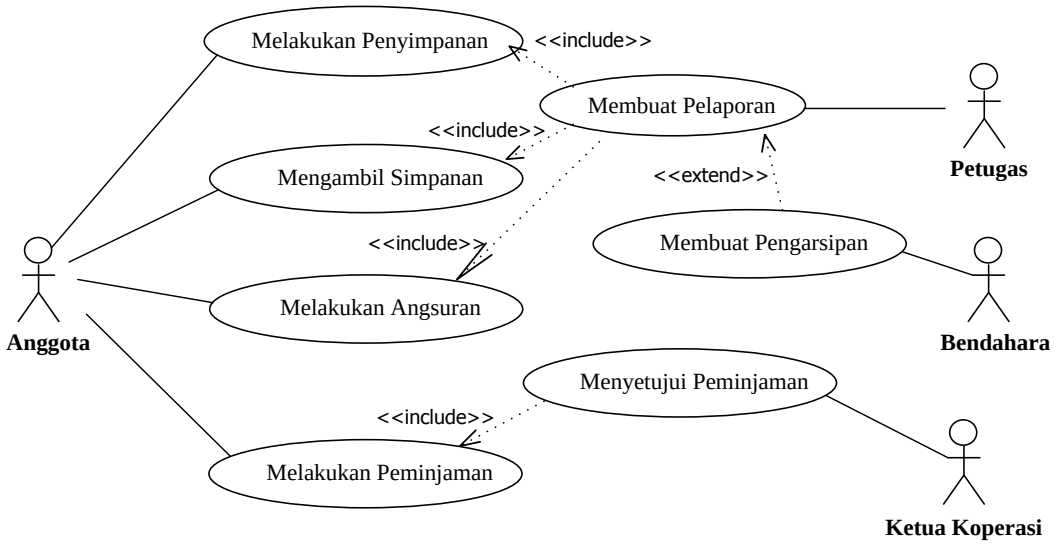
b. Ketika melakukan peminjaman , anggota akan dikenakan jasa sebesar 10% dari uang pinjaman, dan juga akan mendapat potongan 5% dari uang pinjamannya, 1.5% masuk ke simpanan wajib dan 3.5% sebagai adminstrasi
c. Anggota bisa mengambil uang simpanan ketika anggota tidak memiliki angsuran
d. Untuk ketentuan jaminan, jika anggota tidak melakukan pembayaran angsuran hingga 6 bulan, maka jaminan akan langsung diambil

Koperasi Abadi Jaya pu memiliki resort yaitu wilayah yang mereka tangani ada 6 daiantaranya adalah :

1. A – menangani wilayah ledeng
2. B – mengani wilayah Dayeuhkolot
3. C – menangani wilayah Soreang
4. D – menangani wilayah Pangalengan
5. E – menangani wilayah Caringin
6. F – menangani wilayah Cimahi

4.1.1.2 *Use Case* Diagram

Diagram *use case* bisnis memperlihatkan hubungan-hubungan antara aktor-aktor bisnis, *use case* bisnis, dan pekerja-pekerja bisnis untuk suatu organisasi atau perusahaan. Diagram ini memberikan model lengkap tentang apa yang dilakukan perusahaan, siapa yang ada di dalam perusahaan, serta siapa yang berada di luar perusahaan. Berikut adalah gambar model diagram use case bisnis yang berjalan :



Gambar 4.5 *Use case* sistem yang sedang berjalan

4.1.1.3 Skenario *Use Case*

Skenario *Use Case* digunakan untuk memudahkan dalam menganalisa skenario yang akan kita gunakan pada fase-fase selanjutnya dengan melakukan penilaian terhadap skenario tersebut. Adapun tahapan dari skenario *use case* pada sistem yang sedang berjalan adalah sebagai berikut :

4.1.1.3.1 Skenario u*se case* melakukan penyimpanan

Tabel 4. 1 Skenario *use case* melakukan penyimpanan

Identifikasi	
Judul	Melakukan penyimpanan
No. Use Case	UC-01
Deskripsi Use Case	Anggota melakukan penyimpanan ke koperasi
Aktor	Anggota
Trigger	Anggota datang untuk melakukan penyimpanan
Kondisi awal	Anggota sudah menyediakan buku anggota dan petugas siap melayani

Skenario Utama	
Aksi- Aktor	**Reaksi Sistem**
1. Anggota datang kepada petugas dan menunjukkan buku anggota miliknya	
	2. Petugas memeriksa keaslian buku
	3. Jika buku asli, petugas menyerahkan formulir penyimpanan
4 Anggota mengisi formulir dan menyerahkan formulir dan simpanan pada petugas	5 Petugas menyimpan formulir untuk membuat pelaporan

4.1.1.3.2 Skenario u*se case* mengambil simpanan

Tabel 4. 2 Skenario *use case* mengambil simpanan

Identifikasi	
Judul	Mengambil simpanan
No. Use Case	UC-02
Deskripsi Use Case	Anggota mengambil simpanan dari koperasi
Aktor	Anggota
Trigger	Anggota datang untuk mengambil simpanan
Kondisi awal	Anggota sudah menyediakan buku anggota dan petugas siap melayani
Skenario Utama	
Aksi- Aktor	**Reaksi Sistem**
1. Anggota datang kepada petugas dan menunjukkan buku anggota miliknya	
	2. Petugas memeriksa keaslian buku

	3. Jika buku asli, petugas menyerahkan formulir pengambilan simpanan
4 Anggota mengisi formulir dan menyerahkan formulir pada petugas	5 Petugas menyimpan formulir untuk membuat pelaporan
	6 Petugas menyerahkan jumlah pengambilan sesuai dengan formulir

4.1.1.3.3 Skenario u*se case* melakukan peminjaman

Tabel 4. 3 Skenario *use case* melakukan peminjaman

Identifikasi	
Judul	Melakukan peminjaman
No. Use Case	UC-03
Deskripsi Use Case	Anggota melakukan peminjaman kepada koperasi
Aktor	Anggota
Trigger	Anggota datang untuk melakukan peminjaman
Kondisi awal	Anggota sudah menyediakan buku anggota dan petugas siap melayani
Skenario Utama	
Aksi- Aktor	**Reaksi Sistem**
1. Anggota datang kepada petugas dan menunjukkan buku anggota miliknya	
	2. Petugas memeriksa keaslian buku dan data tunggakan

	3. Jika buku asli dan tidak ada tunggakan, petugas menyerahkan formulir peminjaman
4 Anggota mengisi formulir dan menyerahkan formulir pada petugas	5 Petugas menyimpan formulir untuk dilakukan persetujuan oleh kepala koperasi
	6 Jika formulir pengajuan disetujui, petugas menyerahkan pinjaman sesuai dengan formulir dan mencatat data peminjaman

4.1.1.3.4 Skenario u*se case* melakukan angsuran

Tabel 4.4 Skenario *use case* melakukan angsuran

Identifikasi	
Judul	Melakukan angsuran
No. Use Case	UC-04
Deskripsi Use Case	Anggota melakukan angsuran pinjaman kepada koperasi
Aktor	Anggota
Trigger	Anggota datang untuk melakukan pembayaran angsuran
Kondisi awal	Anggota sudah menyediakan buku anggota dan petugas siap melayani
Skenario Utama	
Aksi- Aktor	**Reaksi Sistem**
1. Anggota datang kepada petugas dan menunjukkan buku anggota miliknya	
	2. Petugas memeriksa keaslian buku dan data tunggakan

	3. Jika buku asli dan ada tunggakan, petugas membuat kwitansi angsuran
4 Membayar angsuran	5 Petugas menyimpan pelaporan untuk pengarsipan

4.1.1.3.5 Skenario u*se case* membuat pelaporan

Tabel 4. 5 Skenario *use case* membuat pelaporan

Identifikasi	
Judul	Melakukan pelaporan
No. Use Case	UC-05
Deskripsi Use Case	Petugas membuat data pelaporan
Aktor	Petugas
Trigger	Proses penyimpanan, mengambil simpanan, atau melakukan angsuran selesai
Kondisi awal	Data penyimpanan, pengambilan simpanan, atau angsuran ada
Skenario Utama	
Aksi- Aktor	**Reaksi Sistem**
1. Petugas mencatat data penyimpanan, pengambilan simpanan, atau angsuran	
	2. Data penyimpanan, pengambilan simpanan, atau angsuran tersimpan
3. Petugas menyerahkan data pelaporan pada ketua koperasi	4. Ketua koperasi menerima data pelaporan

4.1.1.3.6 Skenario u*se case* membuat pengarsipan

Tabel 4. 6 Skenario *use case* membuat pengarsipan

Identifikasi	
Judul	Melakukan pengarsipan
No. Use Case	UC-06
Deskripsi Use Case	Bendahara membuat data pengarsipan
Aktor	Bendahara
Trigger	Proses pelaporan selesai dan laporan telah divalidasi ketua koperasi
Kondisi awal	Data pelaporan tersedia dan bendahara siap melakukan pengarsipan
Skenario Utama	
Aksi- Aktor	**Reaksi Sistem**
	1. Ketua koperasi menyerahkan data pelaporan yang telah divalidasi kepada bendahara
2. Bendahara melakukan pengarsipan data pelaporan yang telah divalidasi	
	3. Data pelaporan yang telah divalidasi masuk ke pengarsipan

4.1.1.3.7 Skenario u*se case* menyetujui peminjaman

Tabel 4. 7 Skenario *use case* menyetujui peminjaman

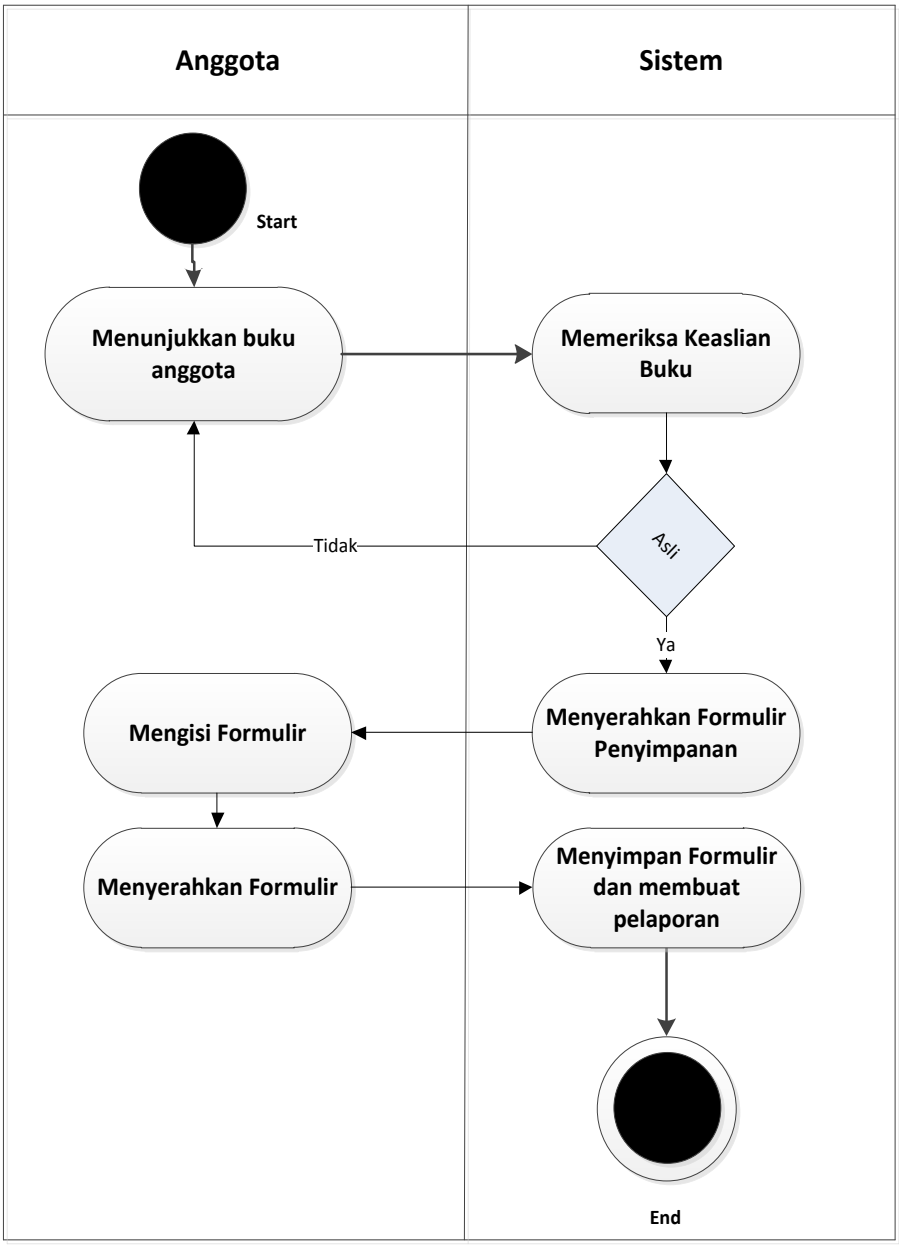
Identifikasi	
Judul	Menyetujui peminjaman
No. Use Case	UC-07
Deskripsi Use Case	Ketua Koperasi menyetujui data peminjaman
Aktor	Ketua Koperasi

Trigger	Data peminjaman baru yang belum disetujui diserahkan pada ketua koperasi
Kondisi awal	Ketua koperasi siap melayani
Skenario Utama	
Aksi- Aktor	**Reaksi Sistem**
	1. Petugas menyerahkan data permintaan peminjaman pada ketua koperasi
2. Ketua koperasi memeriksa data permintaan peminjaman dan memeriksa persyaratan peminjaman	
3. Jika semua syarat dipenuhi, data peminjaman disetujui dan diserahkan kembali pada petugas, jika tidak formulir dikembalikan pada anggota.	
	4. Data peminjaman yang disetujui tersimpan

4.1.1.4 *ActivityDiagram*

Activity diagram adalah diagram yang menggambarkan aliran fungsionalitas dari sistem. Berikut *activity diagram* proses penyimpanan pada sistem yang berjalan.

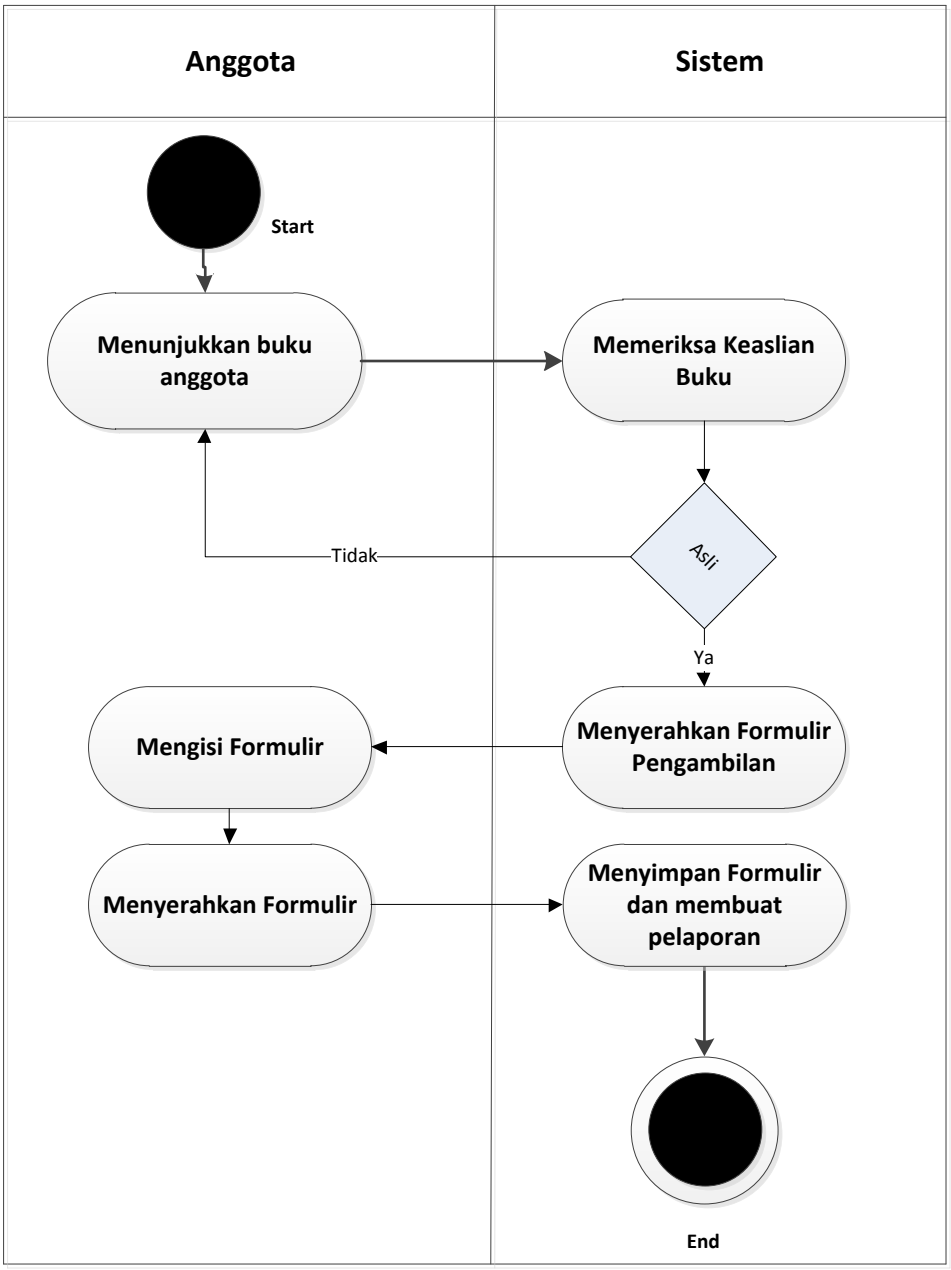
4.1.1.4.1 *Activity diagram* proses melakukan penyimpanan



Gambar 4. 6 *Activity diagram* proses melakukan penyimpanan

4.1.1.4.2 *Activity diagram* mengambil simpanan

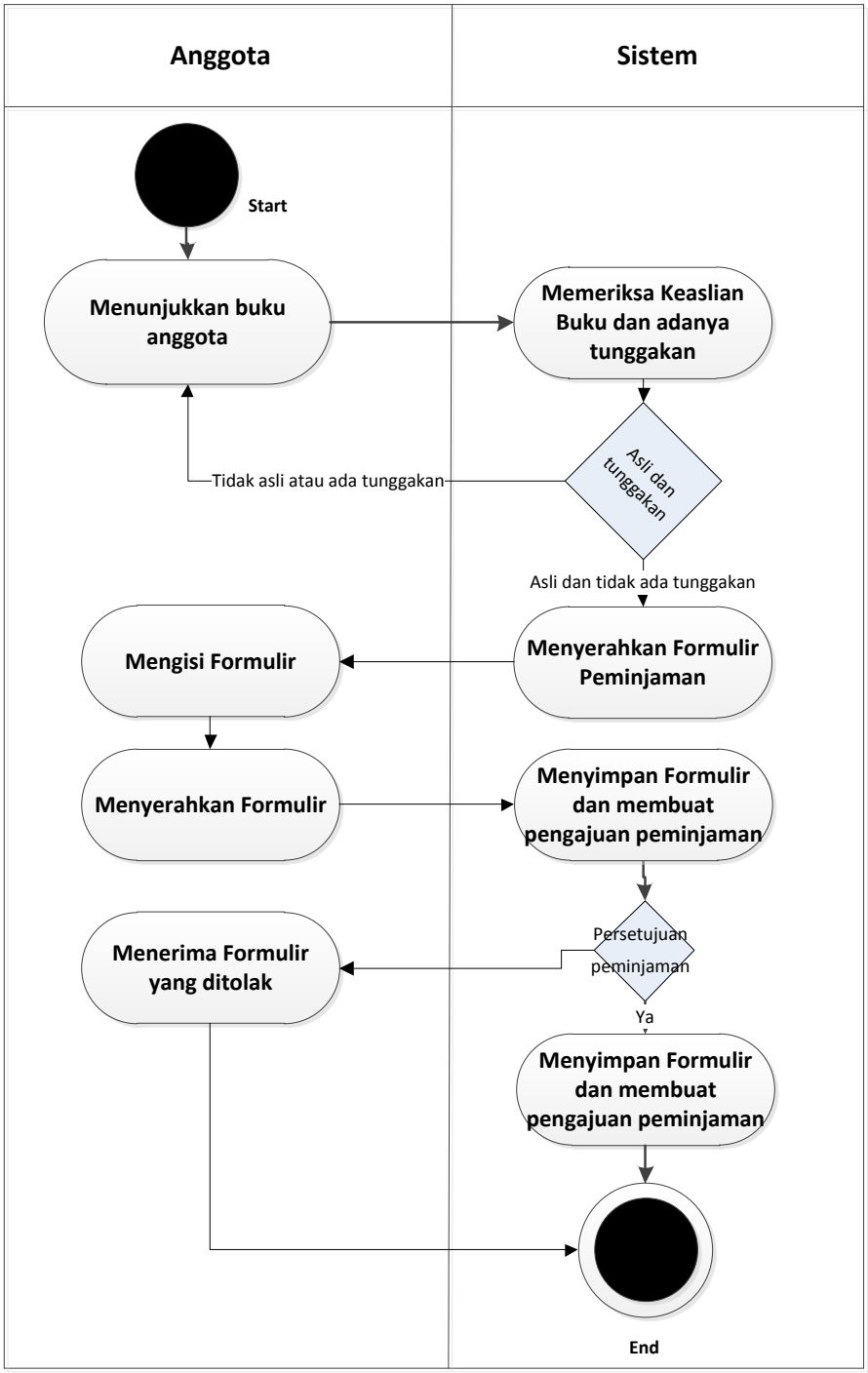
Activity diagram pada proses mengambil simpanan



Gambar 4. 7 *Activity diagram* mengambil simpanan

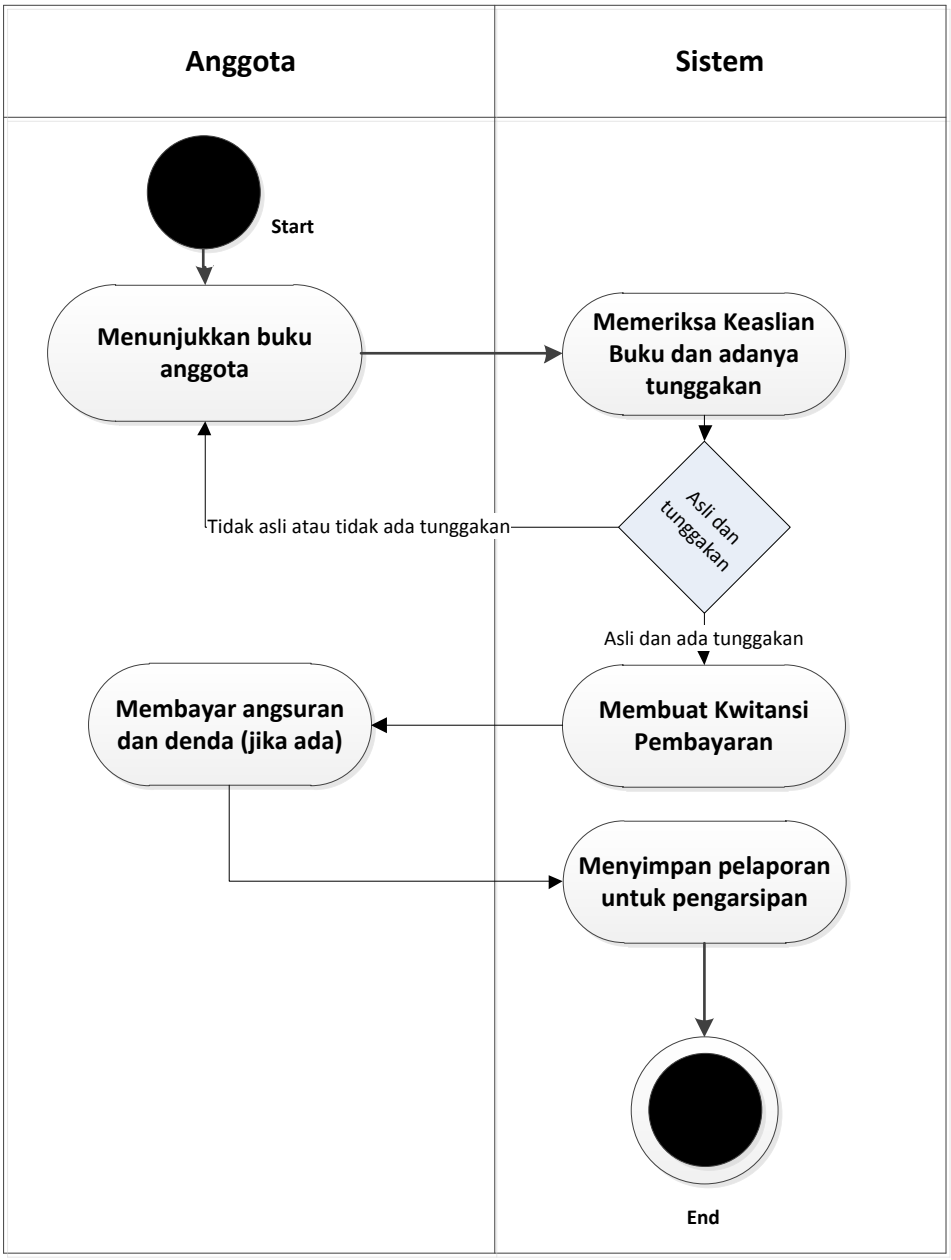
4.1.1.4.3 Activity diagram melakukan peminjaman

Activity diagram pada proses melakukan peminjaman



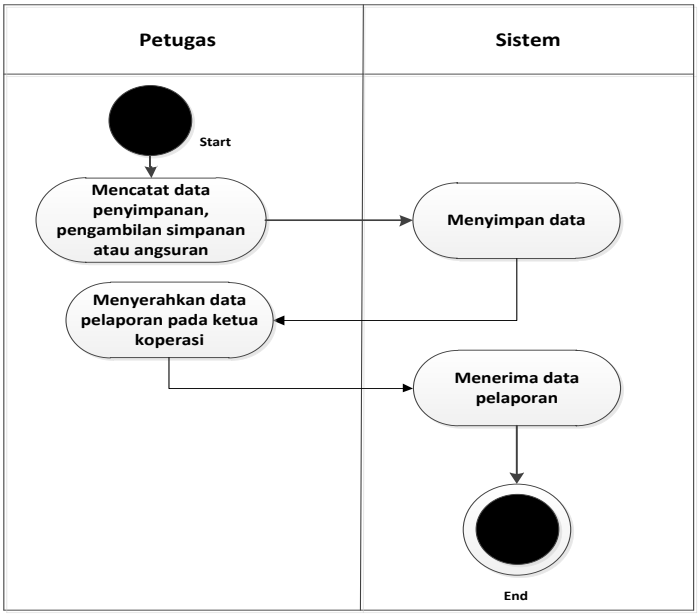
Gambar 4.8 *Activity diagram* melakukan peminjaman

4.1.1.4.4 Activity diagram melakukan angsuran



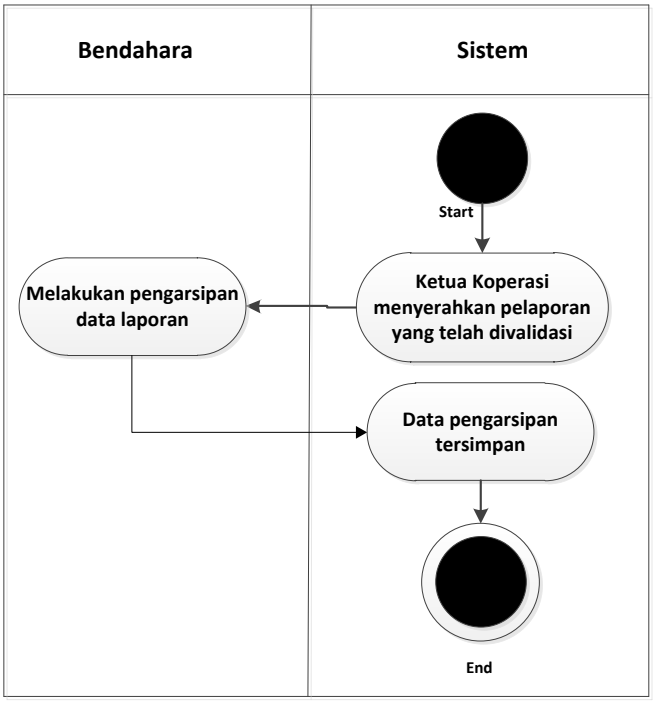
Gambar 4.9 *Activity diagram* melakukan angsuran

4.1.1.4.5 Activity diagram membuat pelaporan



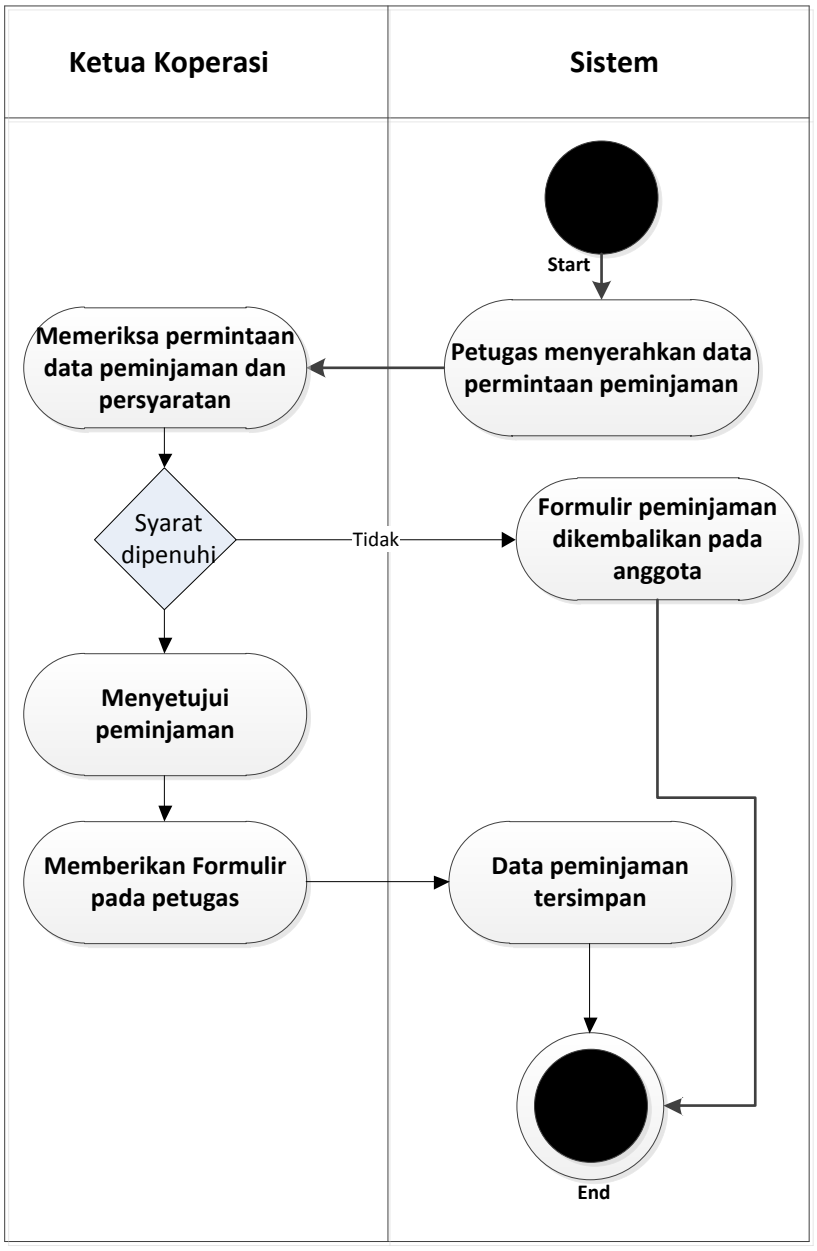
Gambar 4.10 *Activity diagram* membuat pelaporan

4.1.1.4.6 Activity diagram membuat pengarsipan



Gambar 4. 11 *Activity diagram* membuat pengarsipan

4.1.1.4.7 Activity Diagram menyetujui peminjaman



Gambar 4.12 *Activity diagram* menyetujui peminjaman

4.1.2 Evaluasi Proses Bisnis yang Sedang Berjalan

Setelah dilakukan penelitian pada proses bisnis berjalan di Koperasi Simpan Pinjam Abadi Jaya, terdapat beberapa kelemahan yang ditemukan pada proses yang sedang berjalan. Kelemahan-kelemahan tersebut adalah sebagai berikut:

Tabel 4.8 Evaluasi proses bisnis yang berjalan

No	Permasalahan	Penyelesaian
1.	Proses penyimpanan, pengambilan simpanan, pembayaran angsuran, maupun peminjaman masih memerlukan waktu yang cukup lama karena petugas harus berulang kali melakukan pemeriksaan buku anggota.	Merancang aplikasi yang memiliki dan mengelola *database* secara baik dan terintegrasi yang menyimpan data keanggotaan serta history anggota
2.	Proses peminjaman masih tergolong cukup lama karena setelah anggota mengisi formulir peminjaman petugas harus menemui ketua koperasi secara langsung untuk melakukan persetujuan.	Merancang dan membangun applikasi *client-server* yang terkoneksi pada *database* sehingga setiap ada permintaan peminjaman, ketua koperasi dapat langsung memeriksanya dan melakukan persetujuan
3.	Proses pelaporan yang dilakukan masih membutuhkan waktu yang cukup lama karena data yang ada masih berupa arsip, sehingga ada kemungkinan kehilangan data.	Merancang dan membangun aplikasi yang mengelola *database* dengan baik sehingga proses simpan-pinjam dapat dilakukan lebih cepat dan memberikan data yang akurat.

4.2 Perancangan Sistem

Perancangan sistem merupakan suatu kegiatan pengembangan prosedur dan proses yang sedang berjalan untuk menghasilkan suatu sistem yang baru, atau memperbaharui sistem yang ada untuk meningkatkan efektifitas kerja agar dapat memenuhi hasil yang digunakan dengan tujuan memanfaatkan teknologi dan fasilitas yang tersedia. Pada bagian ini penyusun akan memberikan usulan yang merupakan sistem informasi secara komputerisasi yang diharapkan akan membantu dan mempermudah pekerjaan.

4.2.1 Tujuan Perancangan Sistem

Perancangan sistem bertujuan untuk memberikan gambaran yang jelas dan rancang bangun yang sesuai dengan kebutuhan pemakai sistem itu sendiri. Perancangan sistem dilakukan apabila tahap analisis sistem telah dilakukan. Maka untuk selanjutnya seorang analis sistem merancang bagaimana membentuk sistem yang baru ataupun memperbaharui sistem yang lama. Tahap inilah yang dinamakan dengan istilah dari perancangan sistem.

Adapun tujuan perancangan sistem yang diusulkan yaitu untuk menghasilkan suatu rancangan sistem yang dapat membantu proses pengolahan data anggota dalam menangani proses simpan-pinjam dan pelaporan yang semuanya terintegrasi dengan satu *database* dan aplikasi tersebut dapat mencapai sasaran kebutuhan dan tujuannya yakni membantu koperasi dalam pengelolaan data.

4.2.2 Gambaran Umum Sistem yang Diusulkan

Gambaran umum tentang sistem yang diusulkan pada proses perancangan ini adalah suatu sistem yang merubah seluruh proses bisnis yang manual menjadi terkomputerisasi. Proses bisnis yang dimaksud adalah pengolahan data anggota, proses penyimpanan, proses pengambilan simpanan, pembayaran angsuran, proses peminjaman, persetujuan peminjaman, hingga proses pelaporan atau pengarsipan. Proses-proses ini ditangani oleh sistem informasi ini. Sistem ini diharapkan dapat membantu efektifitas dan optimalisasi waktu pada seluruh proses bisnis yang berjalan serta dapat membantu mengatasi permasalahan yang ada dan yang paling penting dapat menghasilkan informasi yang cepat, tepat dan akurat.

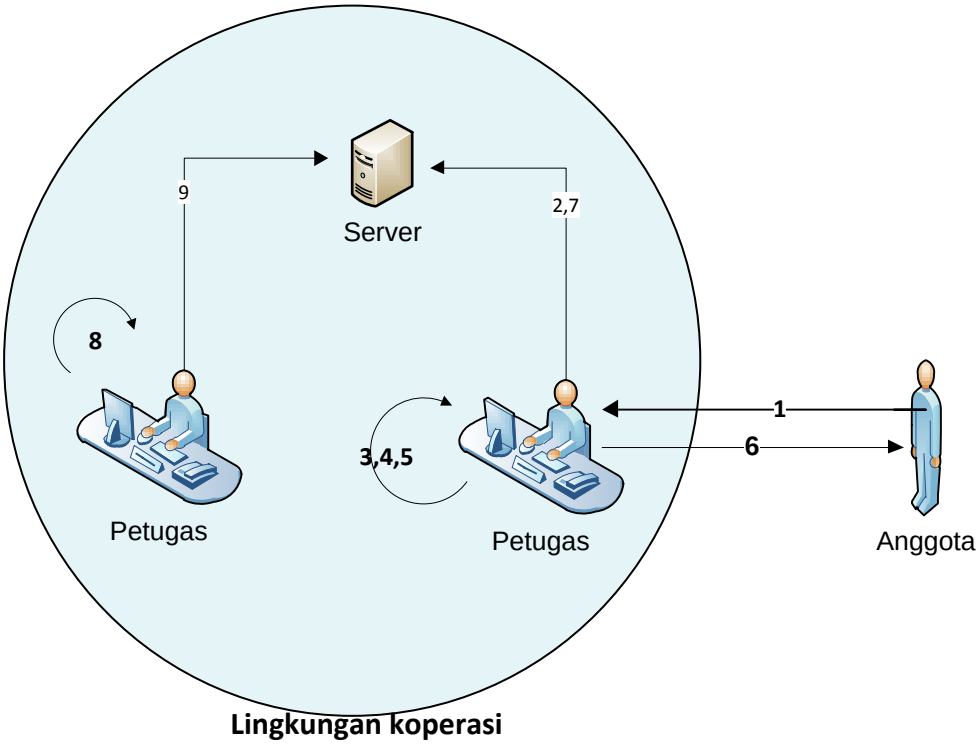
4.2.3 Perancangan Prosedur yang Diusulkan

Perancangan prosedur merupakan awal dari pembuatan sistem informasi yang akan dibuat, dimana dapat dilihat proses-proses apa saja yang nantinya diperlukan dalam pembuatan suatu sistem informasi. Sedangkan perancangan prosedur yang diusulkan merupakan tahap untuk memperbaiki atau meningkatkan efisiensi kerja. Tahap perancangan sistem yang digambarkan merupakan tahap perancangan untuk membangun suatu sistem dan mengkonfigurasikan komponen-komponen perangkat lunak dan perangkat keras sehingga menghasilkan sistem informasi yang baik. Sistem informasi yang dirancang tersebut menjadi satu komponen. Tahapan perancangan prosedur ini akan dijelaskan dengan

menggunakan pemodelan sistem informasi berorientasi objek dengan *Unified Modeling Language* (UML).

4.2.3.1 Proses Bisnis Diusulkan

Proses bisnis penyimpanan yang diusulkan adalah sebagai berikut :



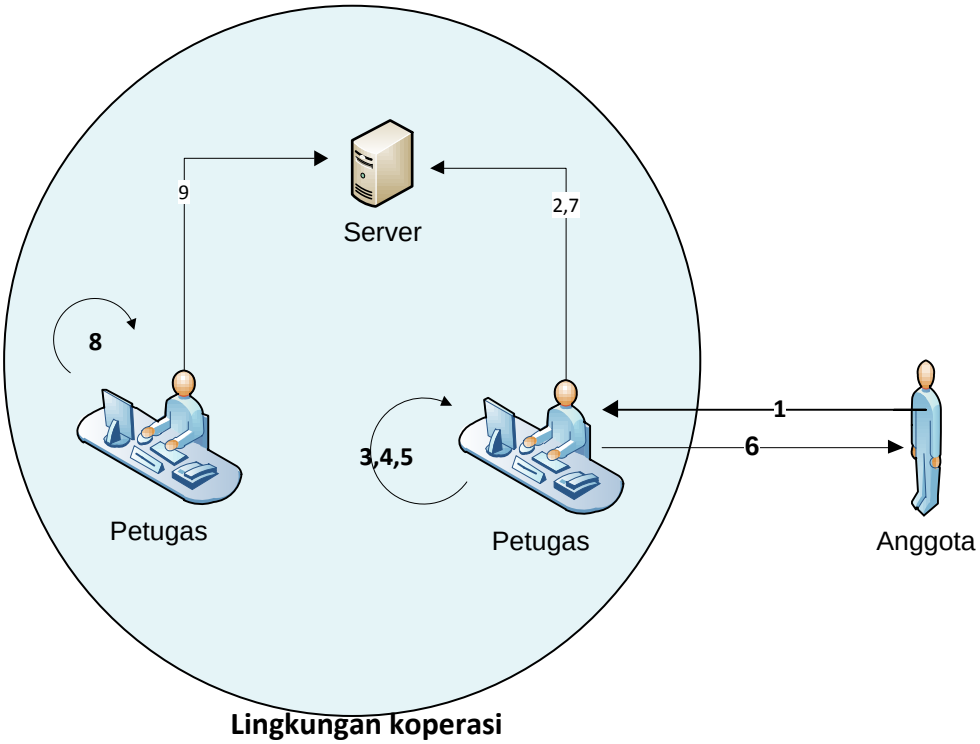
Gambar 4. 13 Proses bisnis penyimpanan yang diusulkan

Keterangan :

1. Anggota datang ke koperasi dengan membawa buku keanggotaan dan menyerahkan simpanannya
2. Petugas memeriksa data anggota pada sistem
3. Jika data anggota tidak ada, petugas dapat menambahkan data keanggotaan

4. Jika data anggota ada, petugas menambahkan data penyimpanan untuk anggota tersebut.
5. Petugas mencetak bukti penyimpanan
6. Petugas menyerahkan bukti penyimpanan pada anggota
7. Data penyimpanan tersimpan di database untuk kemudian divalidasi ketua koperasi
8. Ketua koperasi melakukan validasi peminjaman
9. Data peminjaman ter-update pada database

Proses bisnis pengambilan simpanan yang diusulkan adalah sebagai berikut:

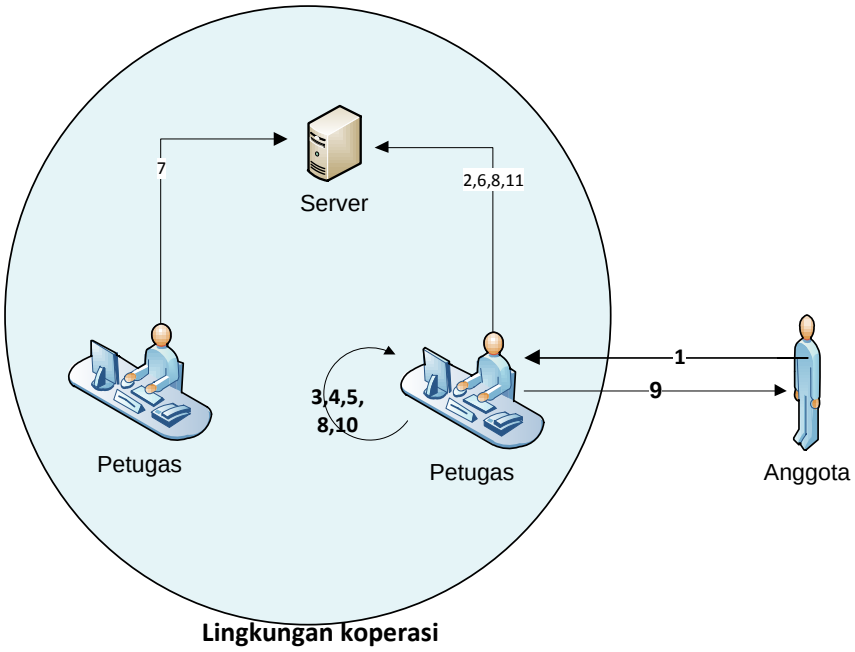


Gambar 4. 14 Proses bisnis pengambilan simpanan yang diusulkan

Keterangan :

1. Anggota datang ke koperasi dengan membawa buku keanggotaan dan meminta penarikan
2. Petugas memeriksa data anggota pada sistem
3. Jika data anggota tidak ada, petugas dapat menambahkan data keanggotaan
4. Jika data anggota ada dan anggota tidak sedang memiliki tunggakan, petugas menambahkan data pengambilan simpanan untuk anggota tersebut.
5. Petugas mencetak bukti pengambilan simpanan
6. Petugas menyerahkan bukti pengambilan pada anggota untuk ditandatangani
7. Data pengambilan simpanan tersimpan di database untuk kemudian divalidasi ketua koperasi
8. Ketua koperasi melakukan validasi pengambilan simpanan
9. Data peminjaman ter-update pada database

Proses bisnis peminjaman yang diusulkan adalah sebagai berikut:



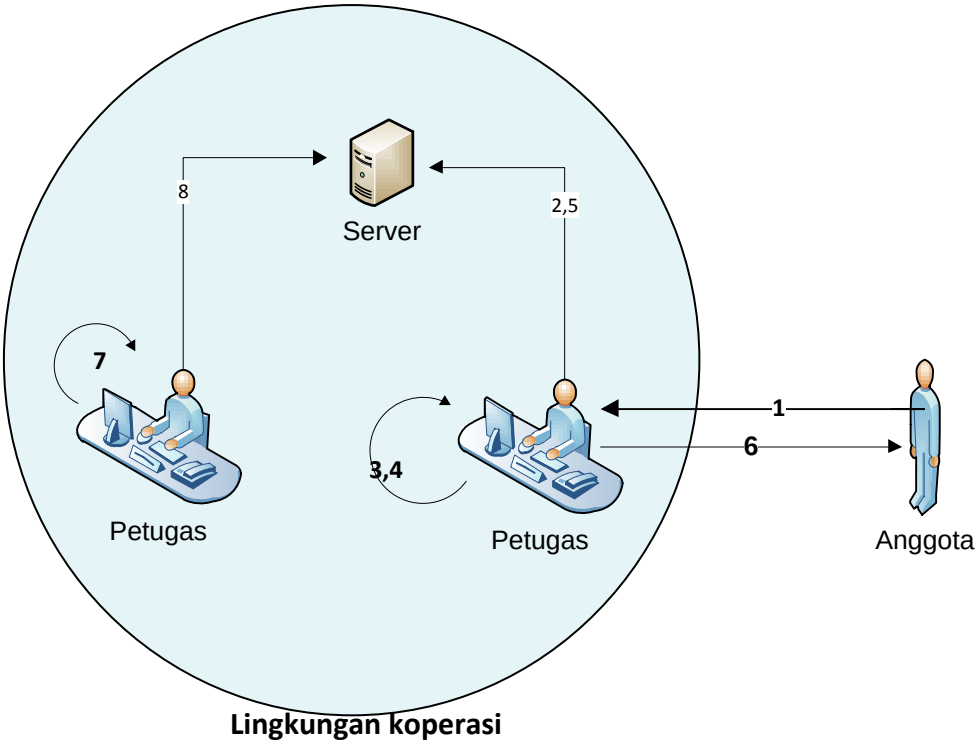
Gambar 4. 15 Proses bisnis peminjaman yang diusulkan

Keterangan :

1. Anggota datang ke koperasi untuk meminta pinjaman
2. Petugas memeriksa data anggota pada sistem
3. Jika data anggota tidak ada, petugas dapat menambahkan data keanggotaan
4. Petugas melakukan survey pemeriksaan jaminan (di luar sistem)
5. Jika jaminan ada, petugas mengisi permintaan peminjaman.
6. Data permintaan tersimpan di database dengan status pending dan belum diberikan pada anggota

7. Ketua koperasi memeriksa data permintaan peminjaman dengan status pending dan menentukan persetujuan (status approve atau reject dan belum diberikan pada anggota).
8. Petugas memeriksa data persetujuan yang pinjamannya belum diberikan pada anggota
9. Petugas menyerahkan pinjaman untuk data yang approve dan belum diberikan
10. Petugas mencetak bukti peminjaman kepada anggota serta meng-update status menjadi “sudah diberikan pada anggota”
11. Data peminjaman ter-update pada database

Proses bisnis pembayaran angsuran yang diusulkan adalah sebagai berikut:



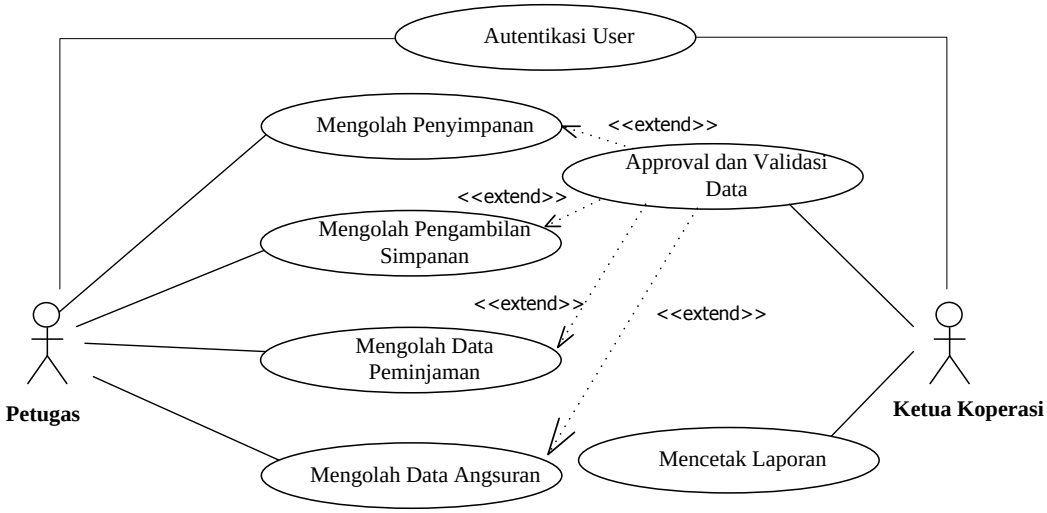
Gambar 4. 16 Proses bisnis pembayaran angsuran yang diusulkan

Keterangan :

1. Anggota datang ke koperasi untuk melakukan pengangsuran
2. Petugas memeriksa data anggota pada sistem
3. Jika anggota memiliki data pinjaman petugas mengisi data pengangsuran
4. Petugas mencetak kwitansi angsuran
5. Data angsuran tersimpan pada database dengan status belum divalidasi dan data pinjaman ter-update
6. Petugas menyerahkan kwitansi pada anggota
7. Ketua koperasi memeriksa data angsuran untuk divalidasi.
8. Data angsuran ter-update statusnya menjadi “valid”

4.2.3.2 *Use Case* Diagram

Diagram *use case* bisnis memperlihatkan hubungan-hubungan antara aktor-aktor bisnis, *use case* bisnis, dan pekerja-pekerja bisnis untuk suatu organisasi atau perusahaan. Diagram ini memberikan model lengkap tentang apa yang dilakukan perusahaan, siapa yang ada di dalam perusahaan, serta siapa yang berada di luar perusahaan. Berikut adalah gambar model diagram use case bisnis yang diusulkan :



Gambar 4. 17 *Use case* diagram yang diusulkan

4.2.3.3 Skenario *Use Case*

Skenario *Use Case* digunakan untuk memudahkan dalam menganalisa skenario yang akan kita gunakan pada fase-fase selanjutnya dengan melakukan penilaian terhadap skenario tersebut. Adapun tahapan dari skenario *use case* pada sistem yang diusulkan adalah sebagai berikut :

4.2.3.3.1 Skenario u*se case* autentikasi user

Tabel 4.9 Skenario *use case* autentikasi user

Identifikasi	
Judul	Autentikasi user
No. Use case	UC-01
Deskripsi use case	Aktor login ke sistem untuk menentukan hak aksesnya
Aktor	Petugas, Ketua Koperasi
Trigger	Aktor menekan tombol login
Kondisi awal	Aplikasi menampilkan sebuah form untuk login
Skenario Utama	
Aksi- aktor	**Reaksi sistem**
1. Aktor mengisi *field* kosong pada *form* login.	
2. Aktor menekan tombol login.	3. Sistem memeriksa keabsahan *user* berdasarkan *username* dan *password* pada *database*.
	4. Sistem mengkonfirmasi keabsahan user.
Kondisi akhir skenario utama	Sistem menampilkan halaman utama berdasarkan hak akses dari user
Skenario Alternatif – 1	
Aksi- aktor	**Reaksi sistem**
1. Aktor menekan tombol login tanpa mengisi *field-field* pada form.	2. *Form* login ditampilkan dan ada pesan kesalahan yang menyatakan *username* atau *password* harus diisi.
	3. *Form* utama tidak ditampilkan.
Kondisi akhir skenario alternatif -1	Aplikasi menampilkan *form login.*

4.2.3.3.2 Skenario u*se case* mengolah penyimpanan

Tabel 4.10 Skenario *use case* mengolah penyimpanan

Identifikasi	
Judul	Mengolah penyimpanan
No. Use case	UC-02
Deskripsi use case	Aktor menambah data penyimpanan
Aktor	Petugas
Trigger	Aktor memilih menekan tombol simpan
Kondisi awal	Aplikasi menampilkan *form* penyimpanan
Skenario Utama	
Aksi- aktor	**Reaksi sistem**
1. Aktor mengisi data penyimpanan yang baru	
2. Aktor men-*submit* data penyimpanan.	3. Sistem memeriksa apakah *field-field* pada *form* pengisian data telah diisi dan sesuai.
	4. Sistem menambahkan data penyimpananyang baru pada *database*
	5. *Form* data anggota yang baru melakukan penyimpanan tampil
Kondisi akhir skenario utama	Sistem menyimpan data transaksi penyimpanan yang baru pada *database*
Skenario Alternatif –1	
Aksi- aktor	**Reaksi sistem**

1. Aktor men-*submit* data dengan menekan sebuah *button* di mana ada *field-field* pada form yang seharusnya diisi tetapi dikosongkan atau ada data yang tidak sesuai.	2. *Form* penyimpanan ditampilkan dengan pesan bahwa ada data *field* yang harus diisi namun masih kosong, atau ada data yang tidak sesuai.
	3. Data penyimpanan baru tidak disimpan pada *database.*
Kondisi akhir skenario alternatif -1	*Form* penyimpanan ditampilkan

4.2.3.3.3 Skenario u*se case* mengolah data pengambilan simpanan

Tabel 4.11 Skenario *use case* mengolah data pengambilan simpanan

Identifikasi	
Judul	Mengolah data pengambilan simpanan
No. Use case	UC-03
Deskripsi use case	Aktor menambahkan data pengambilan simpanan.
Aktor	Petugas
Trigger	Aktor menekan tombol simpan.
Kondisi awal	Aplikasi menampilkan sebuah form untuk menambahkan data pengambilan simpanan
Skenario Utama	
Aksi- aktor	**Reaksi sistem**
1. Aktor mengisi *field* pada *form* pengambilan simpanan.	
2. Aktor menekan tombol simpan	3. Sistem memeriksa apakah *field-field* pada *form* pengisian data telah diisi dan sesuai.

	4. Sistem menambahkan data pengambilan simpanan dan meng-update data simpanan pada database.
	5. Sistem menampilkan form history dari data anggota yang baru melakukan pengambilan simpanan.
Kondisi akhir skenario utama	Data saldo anggota berkurang pada database dan data pengambilan simpanan tersimpan pada database
Skenario Alternatif –1	
Aksi- aktor	**Reaksi sistem**
1. Aktor men-*submit* data dengan menekan sebuah *button* di mana ada *field-field* pada form yang seharusnya diisi tetapi dikosongkan atau ada data yang tidak sesuai.	2. *Form* data pengambilan simpanan ditampilkan dengan pesan bahwa ada data *field* yang harus diisi namun masih kosong, atau ada data yang tidak sesuai.
	3. Data pengambilan simpananbaru tidak disimpan pada *database.*
Kondisi akhir skenario alternatif -1	*Form* pengambilan simpanan ditampilkan

4.2.3.3.4 Skenario u*se case* mengolah data peminjaman

Tabel 4.12 Skenario *use case* mengolah data peminjaman

Identifikasi	
Judul	Mengolah data peminjaman
No. Use case	UC-04
Deskripsi use case	Aktor menambahkan data peminjaman.
Aktor	Petugas

Trigger	Aktor menekan tombol simpan.
Kondisi awal	Aplikasi menampilkan sebuah form untuk menambahkan data peminjaman
Skenario Utama	
Aksi- aktor	**Reaksi sistem**
1. Aktor mengisi *field* pada *form* peminjaman.	
2. Aktor menekan tombol simpan	3. Sistem memeriksa apakah *field-field* pada *form* pengisian data telah diisi dan sesuai.
	4. Sistem menambahkan data peminjaman pada database.
	5. Sistem menampilkan form history dari data anggota yang baru melakukan peminjaman.
Kondisi akhir skenario utama	Data peminjaman tersimpan pada database
Skenario Alternatif –1	
Aksi- aktor	**Reaksi sistem**
1. Aktor men-*submit* data dengan menekan sebuah *button* di mana ada *field-field* pada form yang seharusnya diisi tetapi dikosongkan atau ada data yang tidak sesuai.	2. *Form* data peminjaman ditampilkan dengan pesan bahwa ada data *field* yang harus diisi namun masih kosong, atau ada data yang tidak sesuai.
	3. Data peminjaman baru tidak disimpan pada *database.*
Kondisi akhir skenario alternatif -1	*Form* peminjaman ditampilkan

4.2.3.3.5 Skenario u*se case* mengolah data angsuran

Tabel 4.13 Skenario *use case* mengolah data angsuran

Identifikasi	
Judul	Mengolah data angsuran
No. Use case	UC-05
Deskripsi use case	Aktor menambahkan data angsuran.
Aktor	Petugas
Trigger	Aktor menekan tombol simpan.
Kondisi awal	Aplikasi menampilkan sebuah form untuk menambahkan data angsuran
Skenario Utama	
Aksi- aktor	**Reaksi sistem**
1. Aktor mengisi *field* pada *form* angsuran.	
2. Aktor menekan tombol simpan	3. Sistem memeriksa apakah *field-field* pada *form* pengisian data telah diisi dan sesuai.
	4. Sistem menambahkan data angsuran pada database.
	5. Sistem menampilkan form history dari data anggota yang baru melakukan angsuran.
Kondisi akhir skenario utama	Data angsuran tersimpan pada database dan data jumlah pinjaman berkurang sesuai dengan angsuran
Skenario Alternatif –1	
Aksi- aktor	**Reaksi sistem**

1. Aktor men-*submit* data dengan menekan sebuah *button* di mana ada *field-field* pada form yang seharusnya diisi tetapi dikosongkan atau ada data yang tidak sesuai.	2. *Form* data angsuran ditampilkan dengan pesan bahwa ada data *field* yang harus diisi namun masih kosong, atau ada data yang tidak sesuai.
	3. Data agsuran baru tidak disimpan pada *database.*
Kondisi akhir skenario alternatif -1	*Form* angsuran ditampilkan

4.2.3.3.6 Skenario u*se case* Approval dan validasi data

Tabel 4.14 Skenario *use case* approval dan validasi data

Identifikasi	
Judul	Approval dan validasi data
No. Use case	UC-06
Deskripsi use case	Aktor melakukan approval atau validasi data.
Aktor	Ketua Koperasi
Trigger	Aktor menekan tombol approve.
Kondisi awal	Data penyimpanan, pengambilan simpanan, peminjaman atau angsuran ada yang belum di approve atau divalidkan
Skenario Utama	
Aksi- aktor	**Reaksi sistem**
1. Memilih data yang akan diapprove atau divalidkan dari list data yang belum diapprove atau divalidkan.	

2. Aktor menekan tombol approve	3. Sistem memeriksa apakah *field-field* pada *form* pengisian data telah diisi dan sesuai.
	4. Sistem meng-update status transaksi pada database menjadi valid atau approved.
	5. Sistem menampilkan form dengan table berisi list data yang belum diapprove atau divalidkan.
Kondisi akhir skenario utama	Data transaksi diupdate pada database dan form dengan table berisi list data yang belum diapprove atau divalidkan tampil

4.2.3.3.7 Skenario u*se case* mencetak laporan

Tabel 4.15 Skenario *use case* mencetak laporan

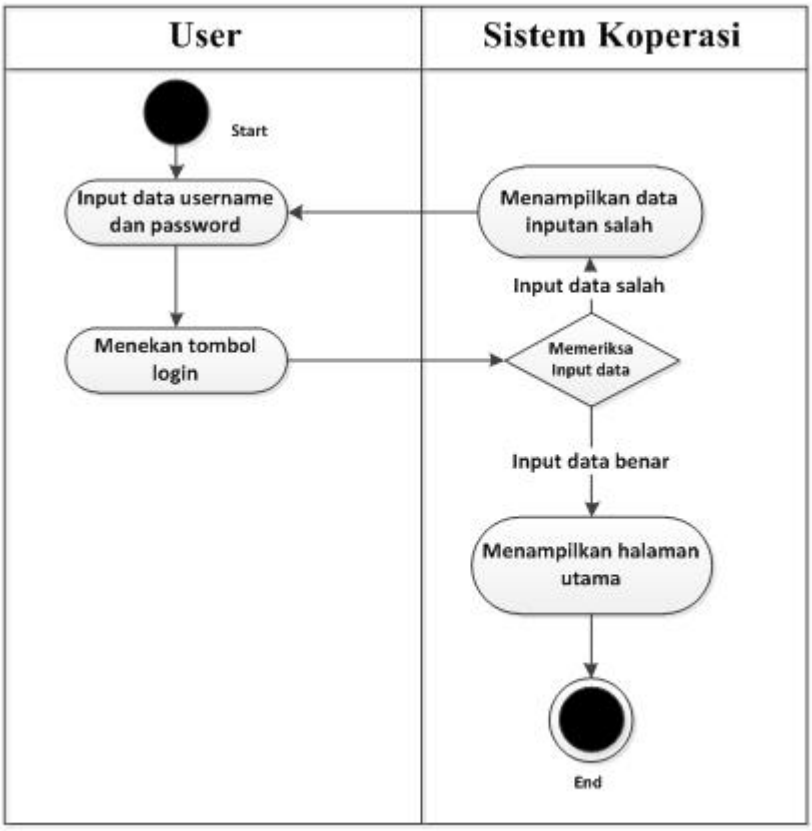
Identifikasi	
Judul	Mencetak laporan
No. Use case	UC-07
Deskripsi use case	Aktor mencetak laporan yang terjadi pada range waktu yang diinginkan.
Aktor	Ketua koperasi.
Trigger	Aktor memilih menu untuk mencetak laporan.
Kondisi awal	Aplikasi menampilkan form untuk menentukan periode dan jenis laporan yang diinginkan.
Skenario Utama	
Aksi- aktor	**Reaksi sistem**

1. Aktor mengisi periode waktu dan jenis laporan yang diinginkan untuk dicetak.	
2. Aktor menekan tombol untuk mencetak laporan	
	3. Sistem mencetak laporan berdasarkan sesuai dengan jenis dan periode laporan yang dipilih.
Kondisi akhir skenario utama	Laporan dicetak
Skenario Alternatif –1	
Aksi- aktor	**Reaksi sistem**
1. Aktor memilih periode yang tidak memiliki data samasekali.	
2. Aktor menekan tombol untuk mencetak laporan	
	3. Sistem menampilkan pesan bahwa data pada periode tersebut kosong.
Kondisi akhir skenario alternatif -1	Sistem menampilkan *form* untuk melakukan pengisian periode laporan yang diinginkan.

4.2.3.4 *ActivityDiagram*

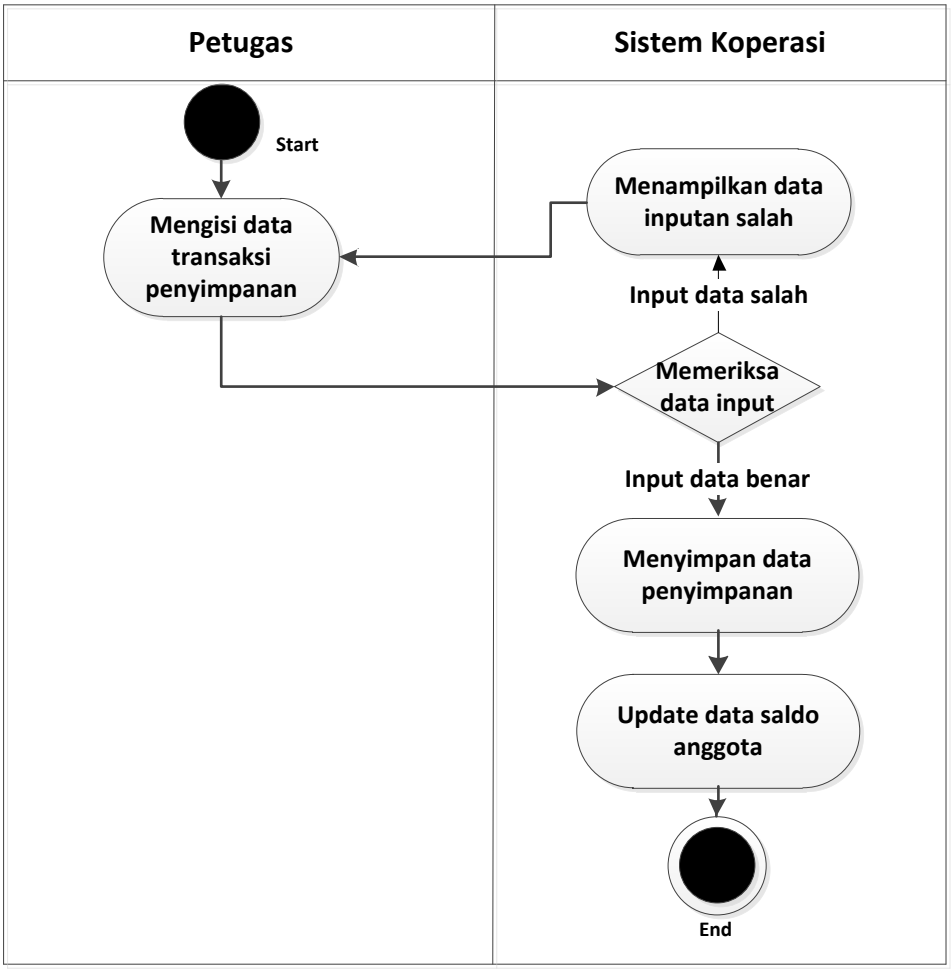
Activity diagram (diagram aktivitas) adalah diagram yang menggambarkan aliran fungsionalitas dari sistem. Berikut *activity diagram* proses perbaikan pada sistem yang diusulkan :

4.2.3.4.1 *Activity Diagram* autentikasi user



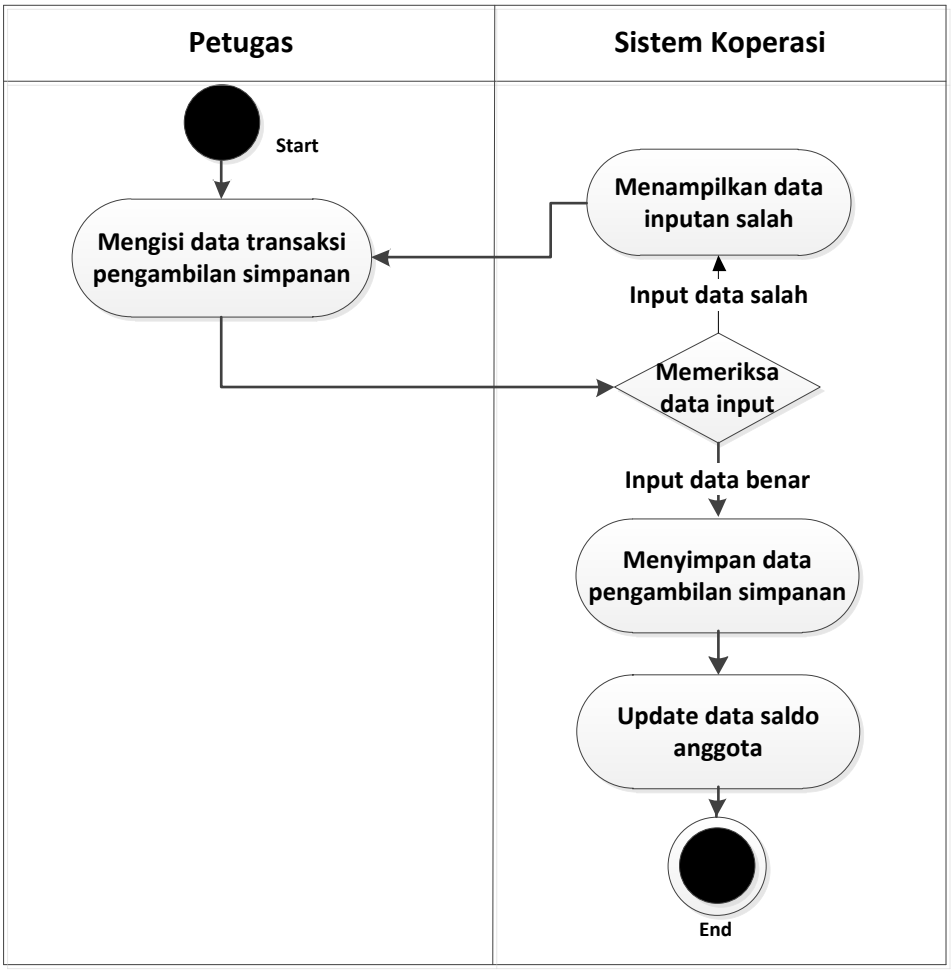
Gambar 4. 18 *Activity diagram* autentikasi *user*

4.2.3.4.2 *Activity Diagram* mengolah Penyimpanan



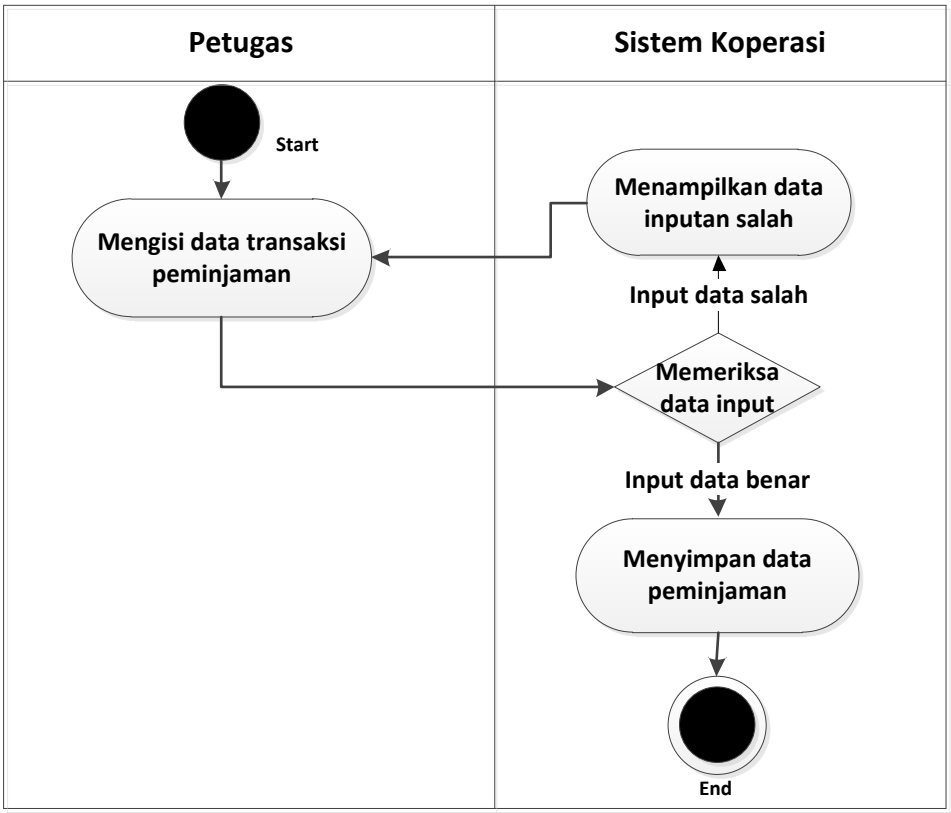
Gambar 4.19 *Activity diagram* mengolah penyimpanan

4.2.3.4.3 *Activity Diagram* mengolah pengambilan simpanan



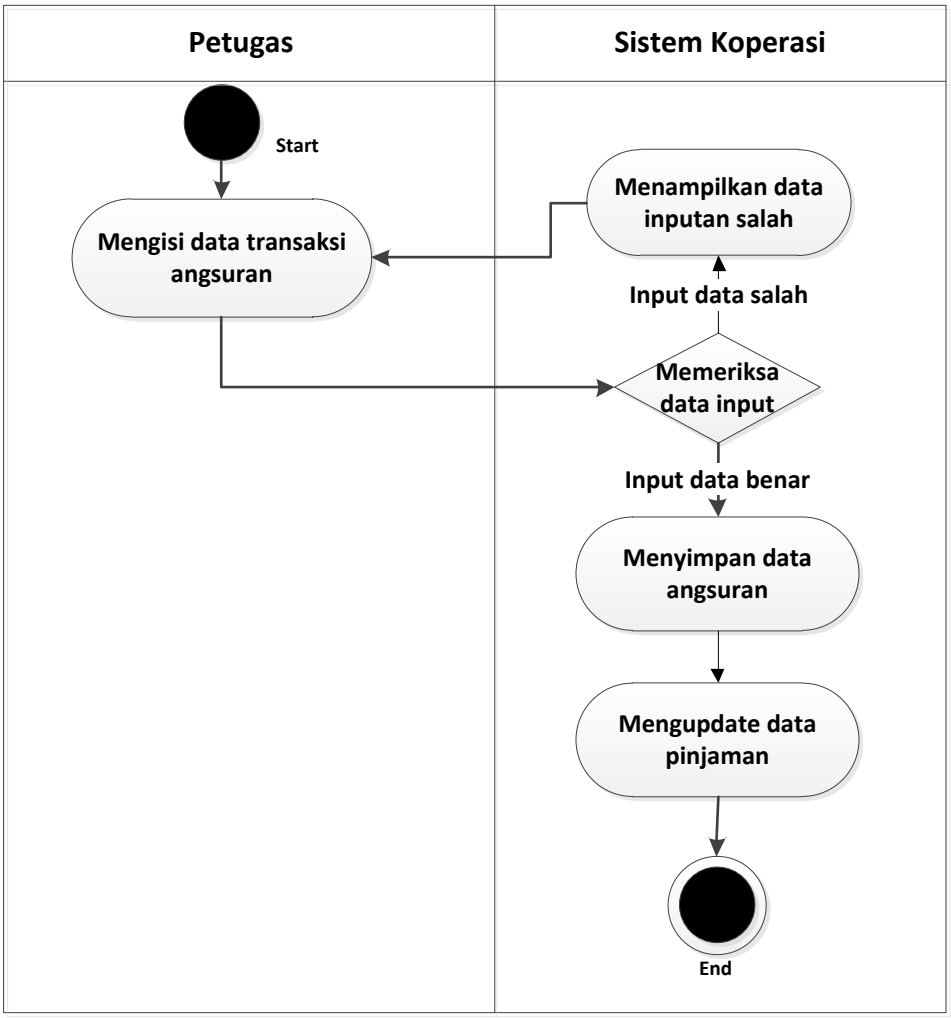
Gambar 4. 20 *Activity diagram* pengambilan simpanan

4.2.3.4.4 *Activity Diagram* mengolah peminjaman



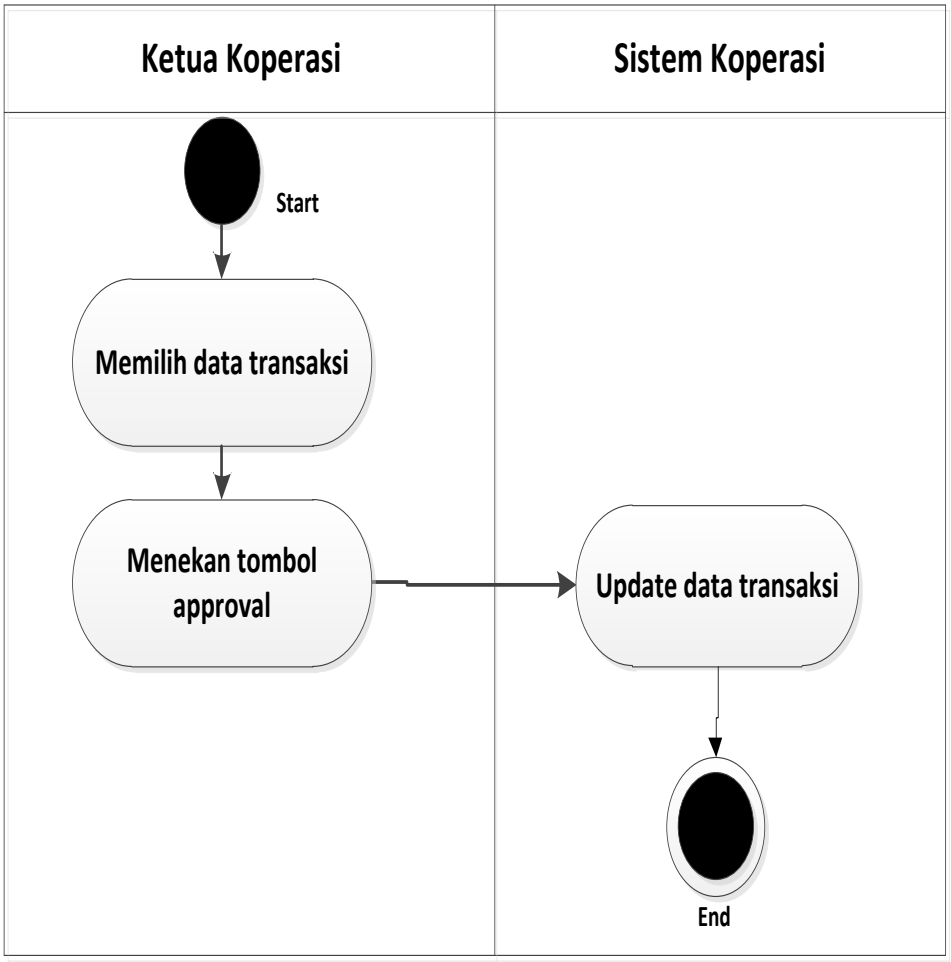
Gambar 4. 21 *Activity diagram* mengolah peminjaman

4.2.3.4.5 *Activity Diagram* mengolah angsuran



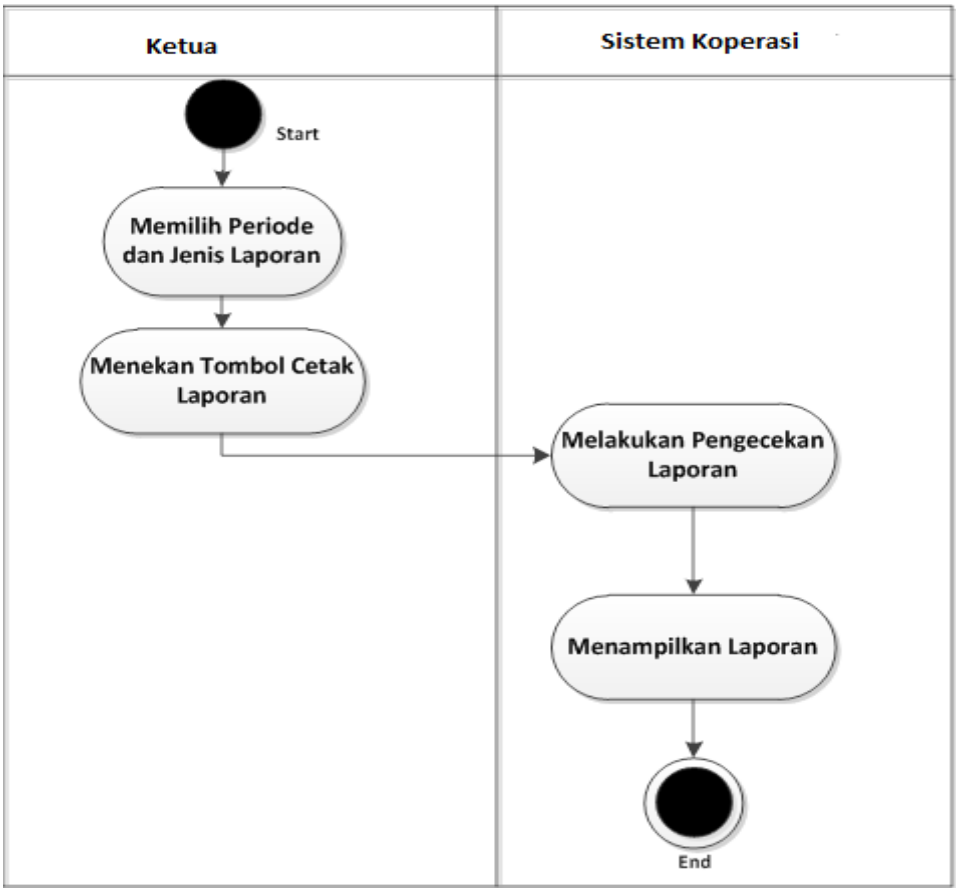
Gambar 4. 22 *Activity diagram* mengolah angsuran

4.2.3.4.6 *Activity Diagram* approval dan validasi data



Gambar 4. 23 *Activity diagram* approval dan validasi data

4.2.3.4.7 *Activity Diagram* mengolah pelaporan

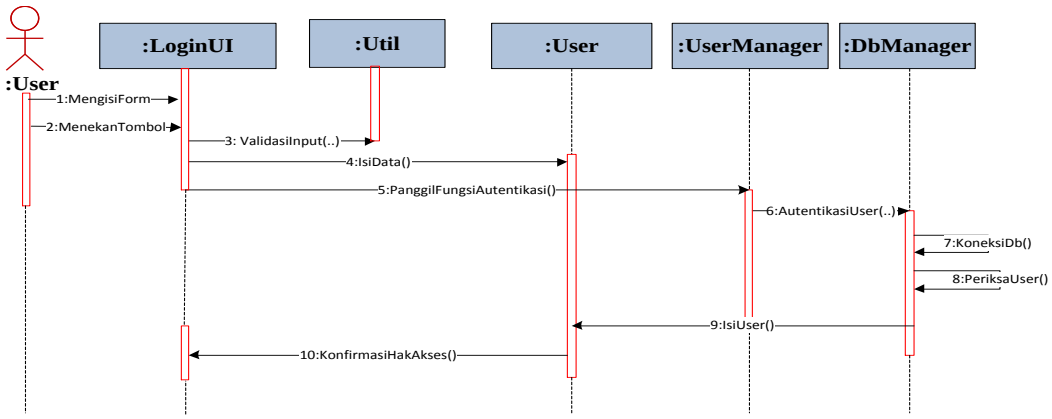


Gambar 4. 24 *Activity diagram* mengolah pelaporan

4.2.3.5 SequenceDiagram

Sequence Diagram digunakan untuk menggambarkan interaksi antar objek dalam waktu yang berurutan. Tetapi pada dasarnya *sequence diagram* digunakan dalam lapisan abstraksi model objek. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antar object, juga interaksi antar objek, dan menunjukkan sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. Komponen utama *sequence diagram* terdiri atas objek yang dituliskan dengan kotak segiempat bernama, pesan diwakili oleh garis dengan tanda panah, dan waktu yang ditunjukkan dengan proses vertikal. Berikut adalah *sequence diagram* yang diusulkan :

4.2.3.5.1 *Sequence Diagram* autentikasi user



Gambar 4. 25 *Sequence diagram* Autentikasi User

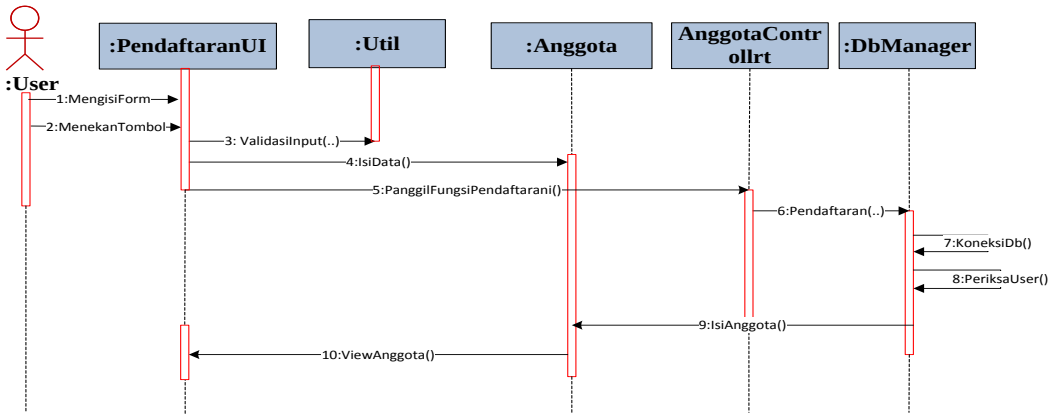
Keterangan:

Pada Gambar 4.16 dapat dilihat bahwa ada 5 kelas yang saling berinteraksi, yakni:

1. LoginUI
2. Util
3. User
4. UserManager

5. DbManager

4.2.3.5.2 *Sequence Diagram* Pendaftaran Anggota



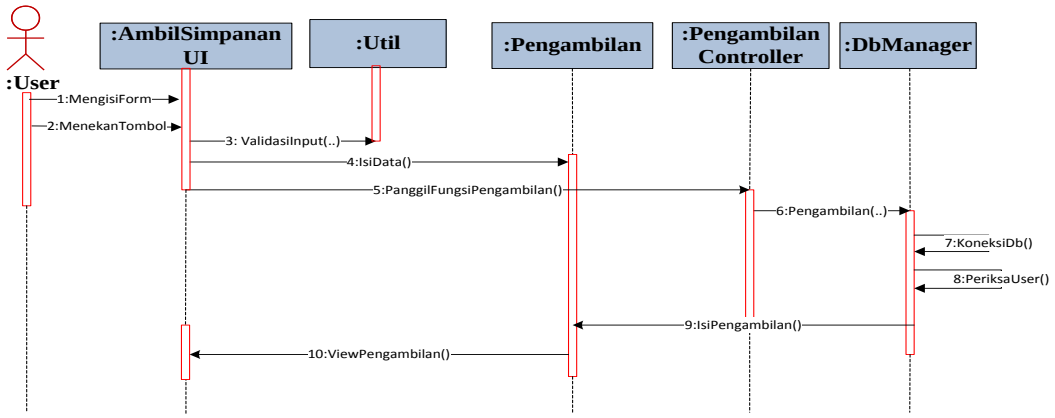
Gambar 4. 26 *Sequence diagram* Pendaftaran Anggota

Keterangan:

Pada Gambar 4.26 dapat dilihat bahwa ada 5 kelas yang saling berinteraksi, yakni:

1. PendaftaranUI
2. Util
3. Anggota
4. AnggotaManager
5. DbManager

4.2.3.5.3 *Sequence Diagram* Pengambilan Simpanan



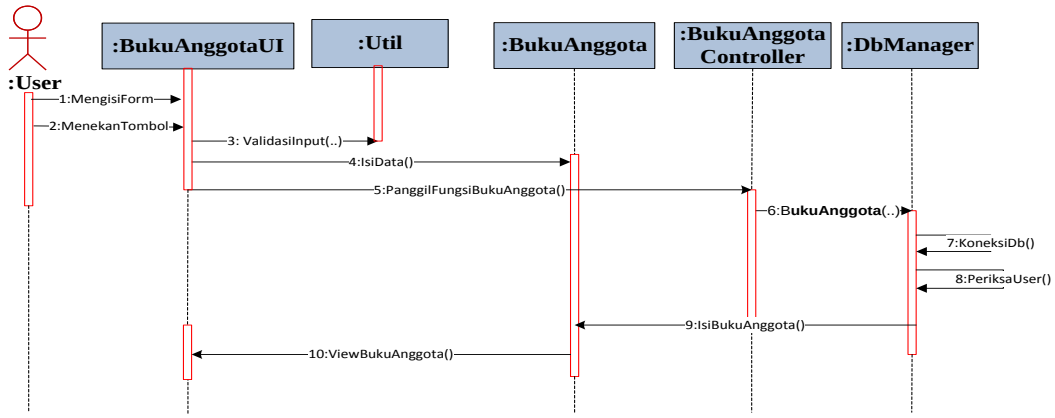
Gambar 4. 27 *Sequence diagram* Pengambilan simpanan

Keterangan:

Pada Gambar 4.27 dapat dilihat bahwa ada 5 kelas yang saling berinteraksi, yakni:

1. AmbilSimpananUI
2. Util
3. Pengambilan
4. PengambilanManager
5. DbManager

4.2.3.5.4 *Sequence Diagram* Buku Anggota



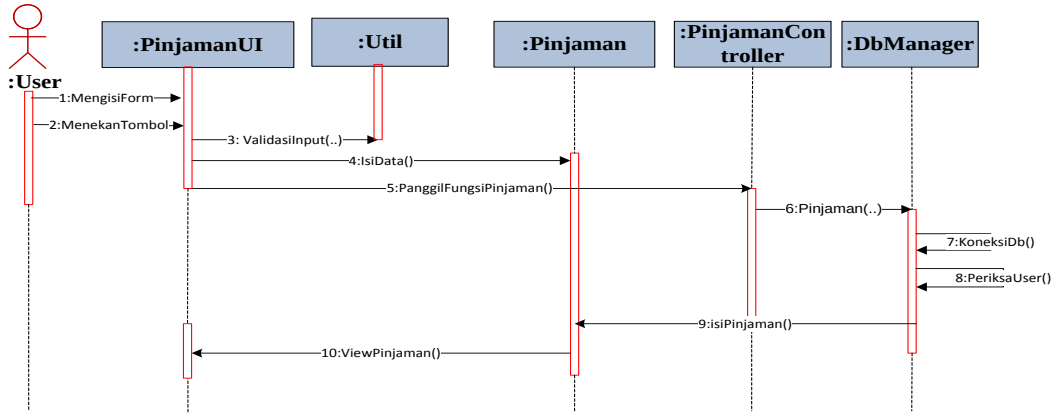
Gambar 4. 188 *Sequence diagram* Buku Anggota

Keterangan:

Pada Gambar 4.28 dapat dilihat bahwa ada 5 kelas yang saling berinteraksi, yakni:

1. BukuAnggotaUI
2. Util
3. BukuAnggota
4. BukuAnggotaManager
5. DbManager

4.2.3.5.5 *Sequence Diagram* Pinjaman



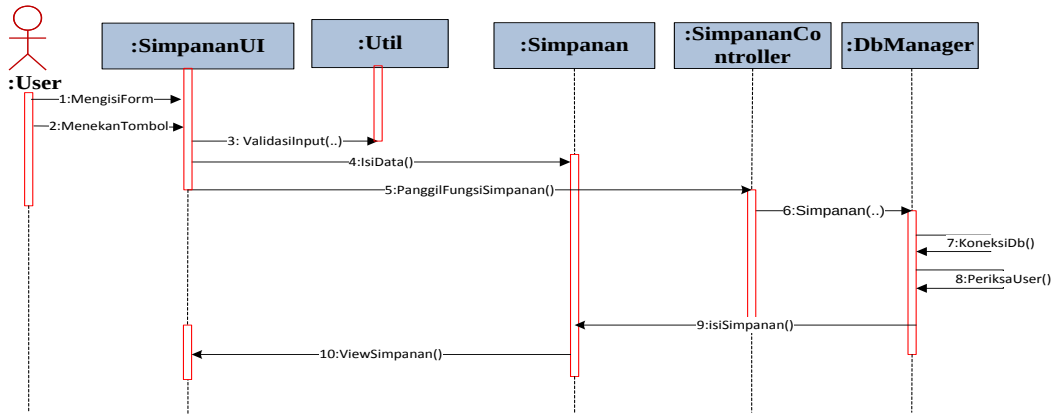
Gambar 4. 29 *Sequence diagram* Pinjaman

Keterangan:

Pada Gambar 4.29 dapat dilihat bahwa ada 5 kelas yang saling berinteraksi, yakni:

1. BukuAnggotaUI
2. Util
3. BukuAnggota
4. BukuAnggotaManager
5. DbManager

4.2.3.5.6 *Sequence Diagram* Simpanan



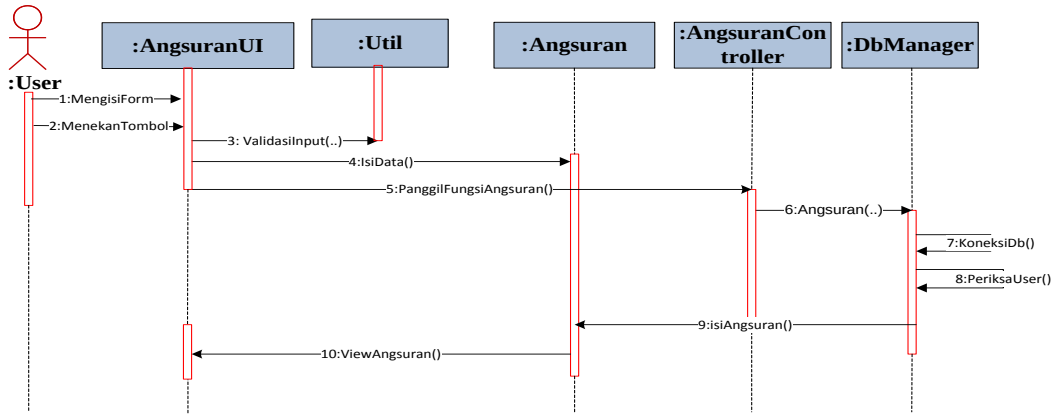
Gambar 4. 19 *Sequence diagram* Simpanan

Keterangan:

Pada Gambar 4.30 dapat dilihat bahwa ada 5 kelas yang saling berinteraksi, yakni:

1. SimpananUI
2. Util
3. Simpanan
4. SimpananManager
5. DbManager

4.2.3.5.7 *Sequence Diagram* Angsuran



Gambar 4.31 *Sequence diagram* Angsuran

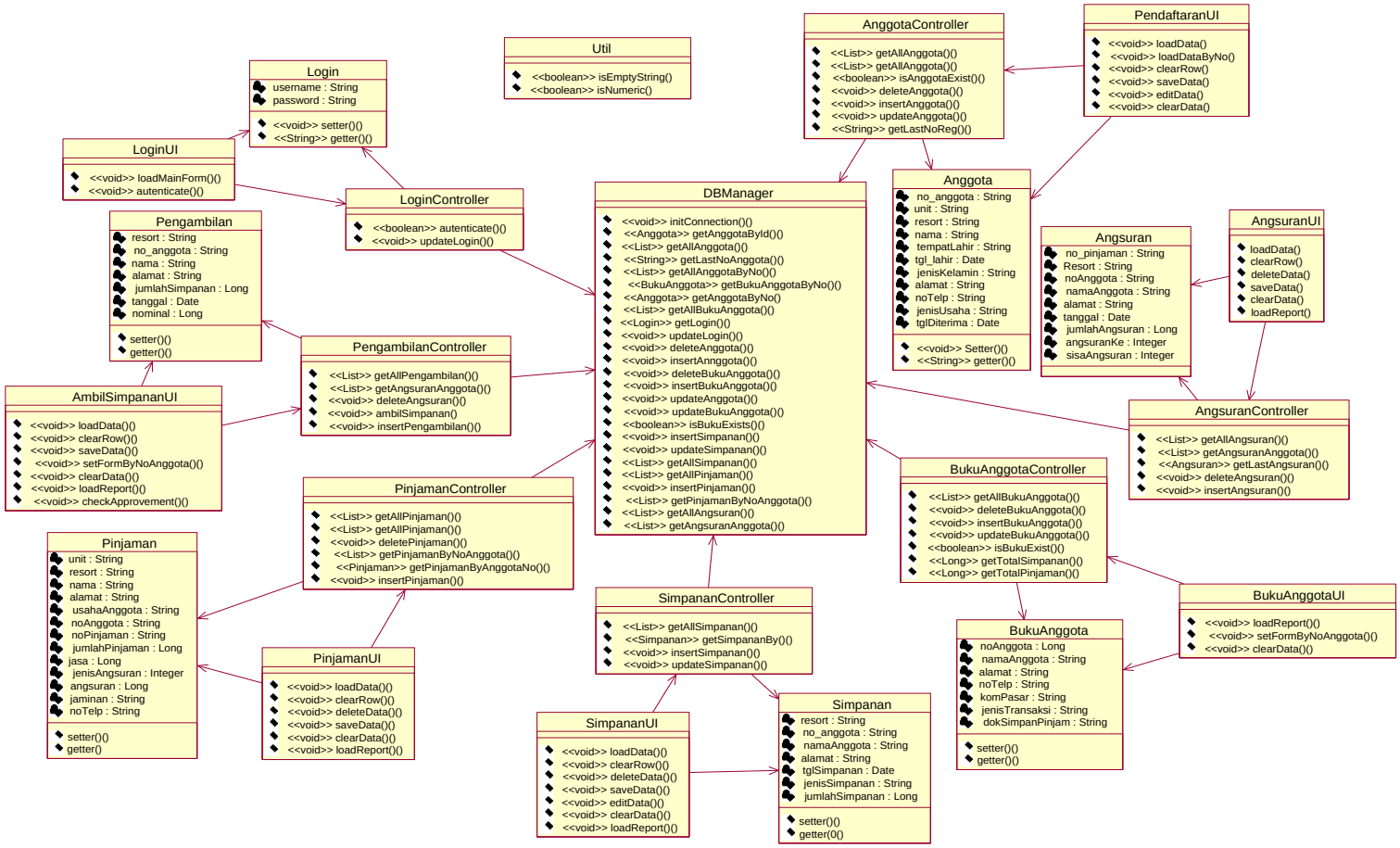
Keterangan:

Pada Gambar 4.31 dapat dilihat bahwa ada 5 kelas yang saling berinteraksi, yakni:

1. SimpananUI
2. Util
3. Simpanan
4. SimpananManager
5. DbManager

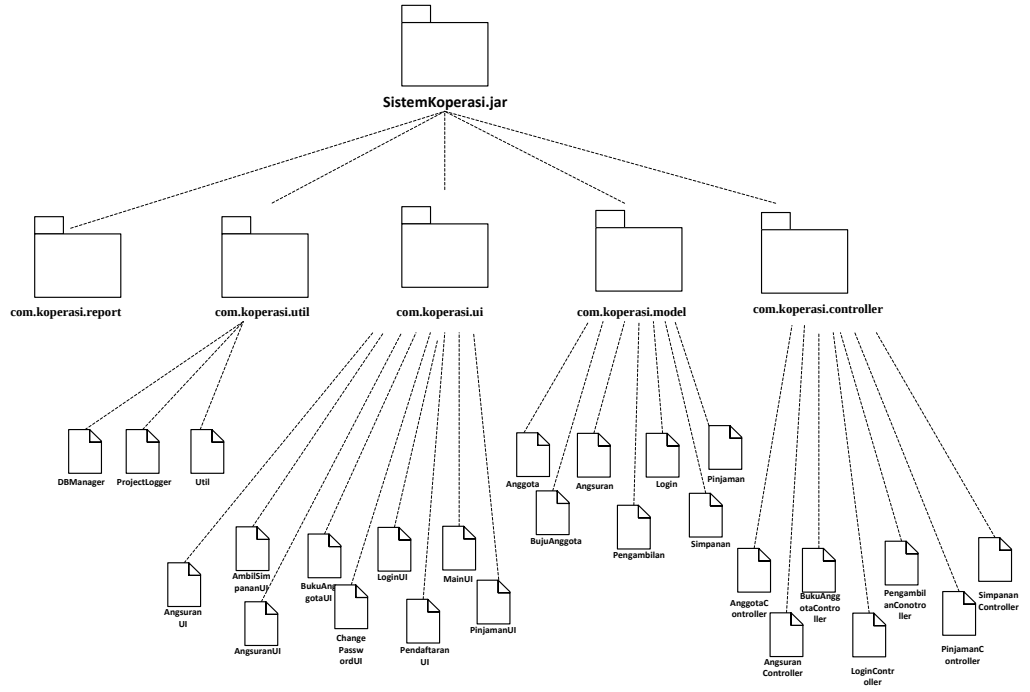
4.2.3.6 ClassDiagram

Menggambarkan struktur statis *class* di dalam sistem. *Class* merepresentasikan sesuatu yang ditangani oleh sistem. Dengan melihat karakteristik sistem pemasaran produk dari bagian penjualan beserta proses-proses yang terjadi, maka dapat dibuat *class diagram* Berikut *class diagram* yang diusulkan pada sistem informasi Koperasi Simpan Pinjam Abadi Jaya:



Gambar 4. 32 *Class Diagram*

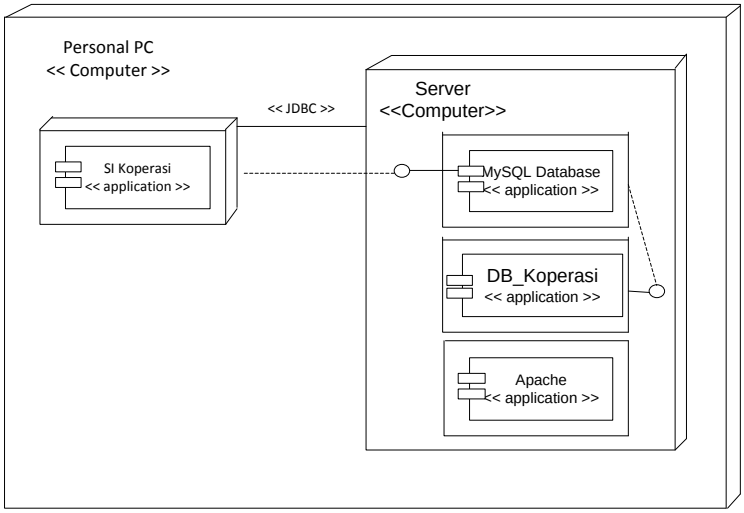
4.2.3.7 Component Diagram



Gambar 4. 33 *Component Diagram*

4.2.3.8 Deployment Diagram

Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan. Diagram ini memuat simpul-simpul beserta komponen-komponen yang ada didalamnya. *Deployment diagram* berhubungan dengan diagram komponen dimana *deployment diagram* memuat satu atau lebih komponen-komponen.



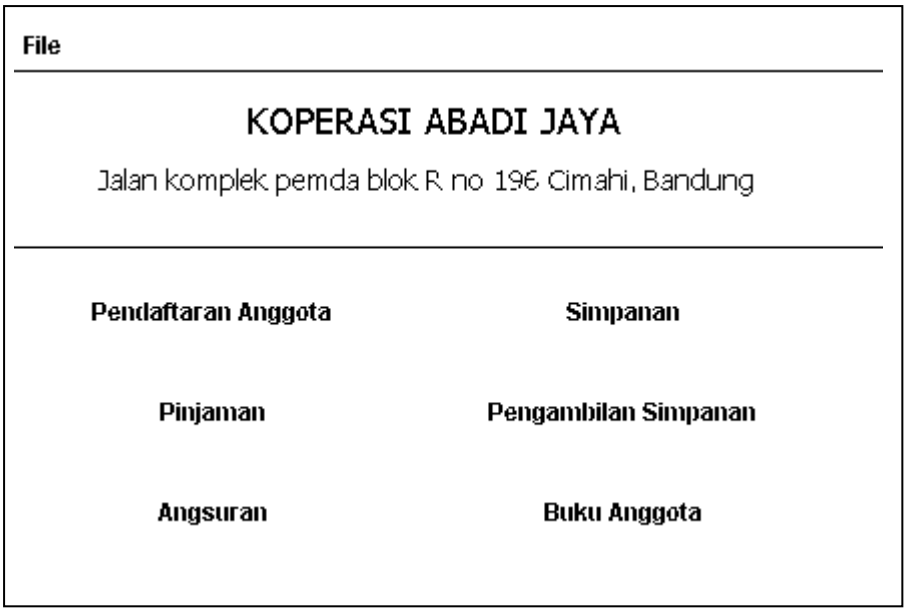
Gambar 4. 204 *Deployment diagram* sistem yang diusulkan

4.2.4 Perancangan Antar Muka

Perancangan antar muka merupakan perancangan yang dibuat sebelum program aplikasi dibuat, perancangan antar muka pada sistem informasi Koperasi Simpan Pinjam Abadi Jayaadalah sebagai berikut:

4.2.4.1 Perancangan Struktur Menu

Perancangan menu dibuat sebagai alat antar muka dengan pengguna untuk memudahkan pengoperasian perangkat lunak. Berikut rancangan menu perangkat lunak ini :



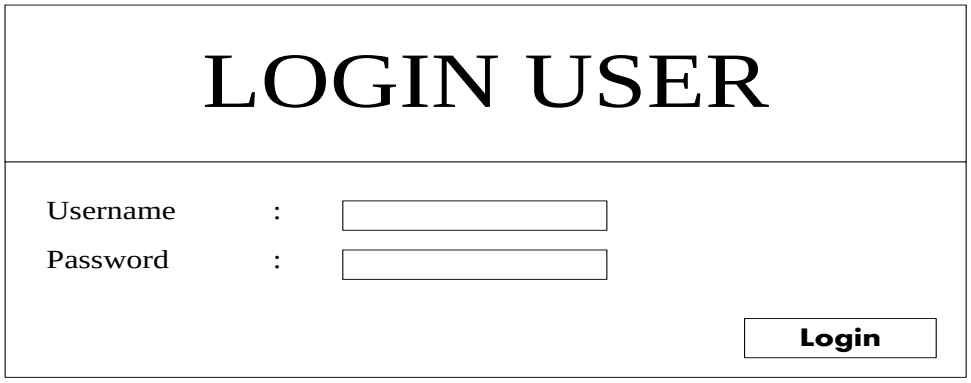
Gambar 4.35 Perancangan Struktur Menu

4.2.4.2 Perancangan Input

Perancangan input merupakan perancangan tampilan yang akan digunakan guna memasukkan data pada sistem untuk kemudian diproses. Dalam perancangan input ini, data yang dimasukkan akan mempengaruhi hasil yang ditampilkan. Adapun perancangan-perancangan input yang ada dalam perancangan ini adalah:

1. Rancangan Login/Absensi

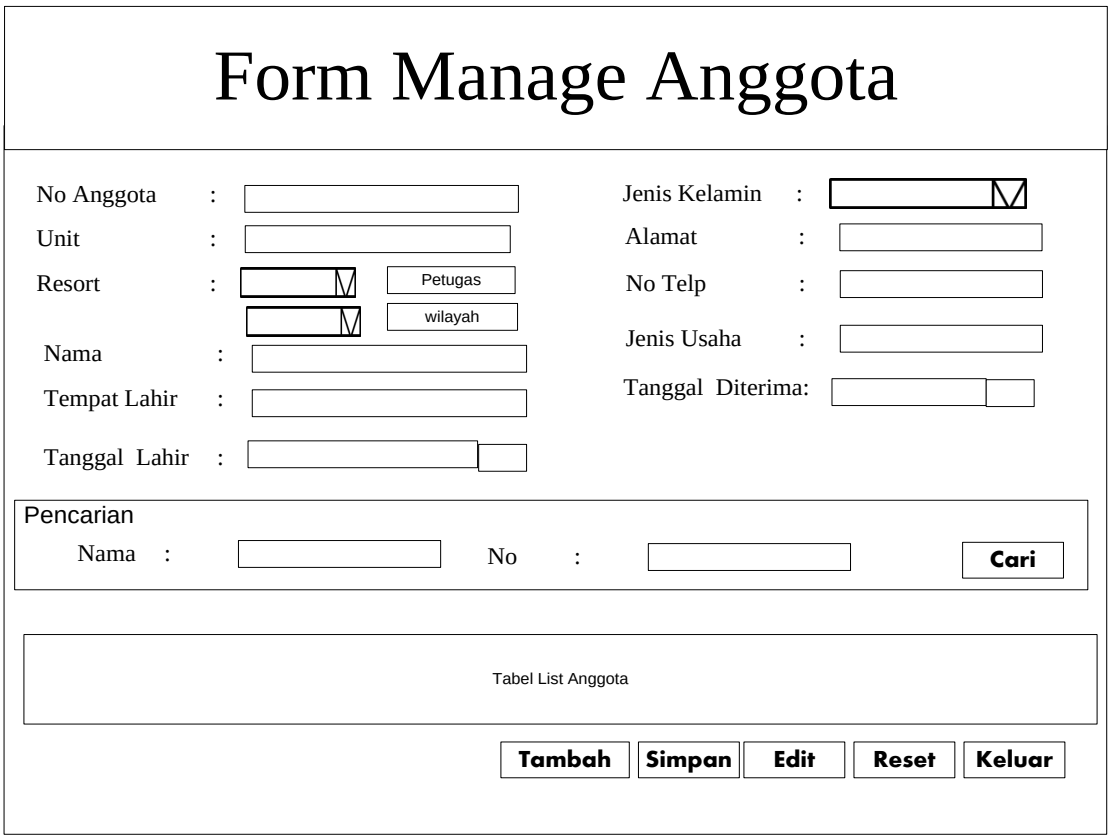
   Tampilan yang dirancang ini berfungsi sebagai tampilan untuk melakukan login.



Gambar 4.36 Rancangan Login

2. Rancangan Pendaftaran Anggota

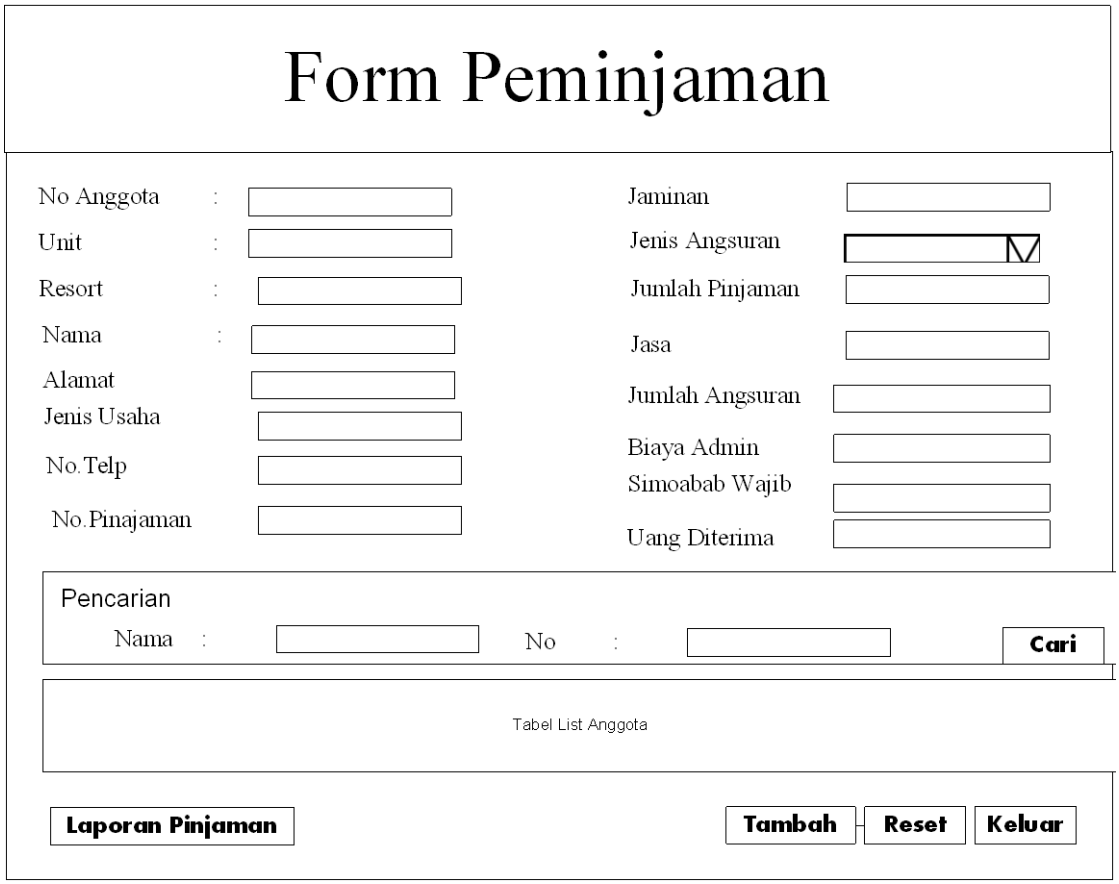
Tampilan yang dirancang ini berfungsi sebagai tampilan untuk melakukan pencarian, menambah dan mengubah data anggota.



Gambar 4. 37 Rancangan Pendaftaran anggota

3. Rancangan tampilan pinjaman

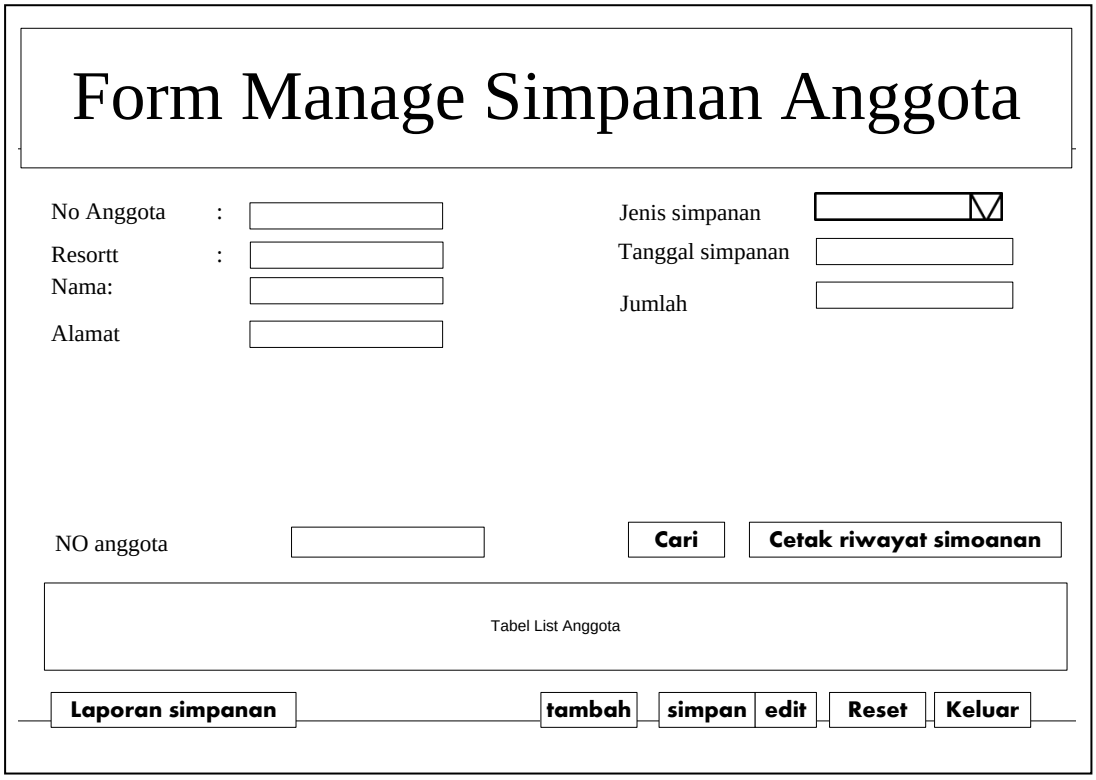
   Tampilan yang dirancang ini berfungsi sebagai tampilan untuk melakukan peminjaman



Gambar 4. 38 Rancangan Pinjaman

4. Rancangan tampilan simpanan

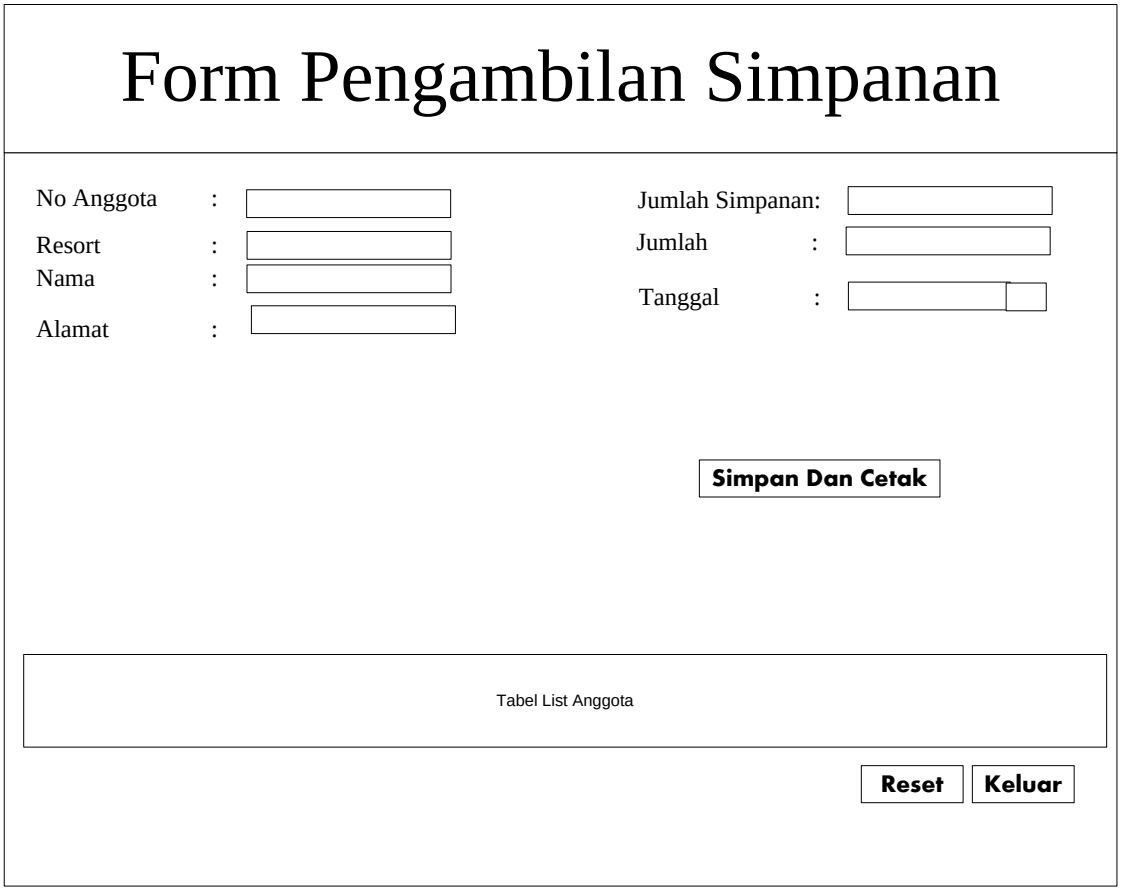
Tampilan yang dirancang ini berfungsi sebagai tampilan untuk simpanan anggota



Gambar 4. 39 Rancangan tampilan simpanan

5. Tampilan Pengambilan Simpanan

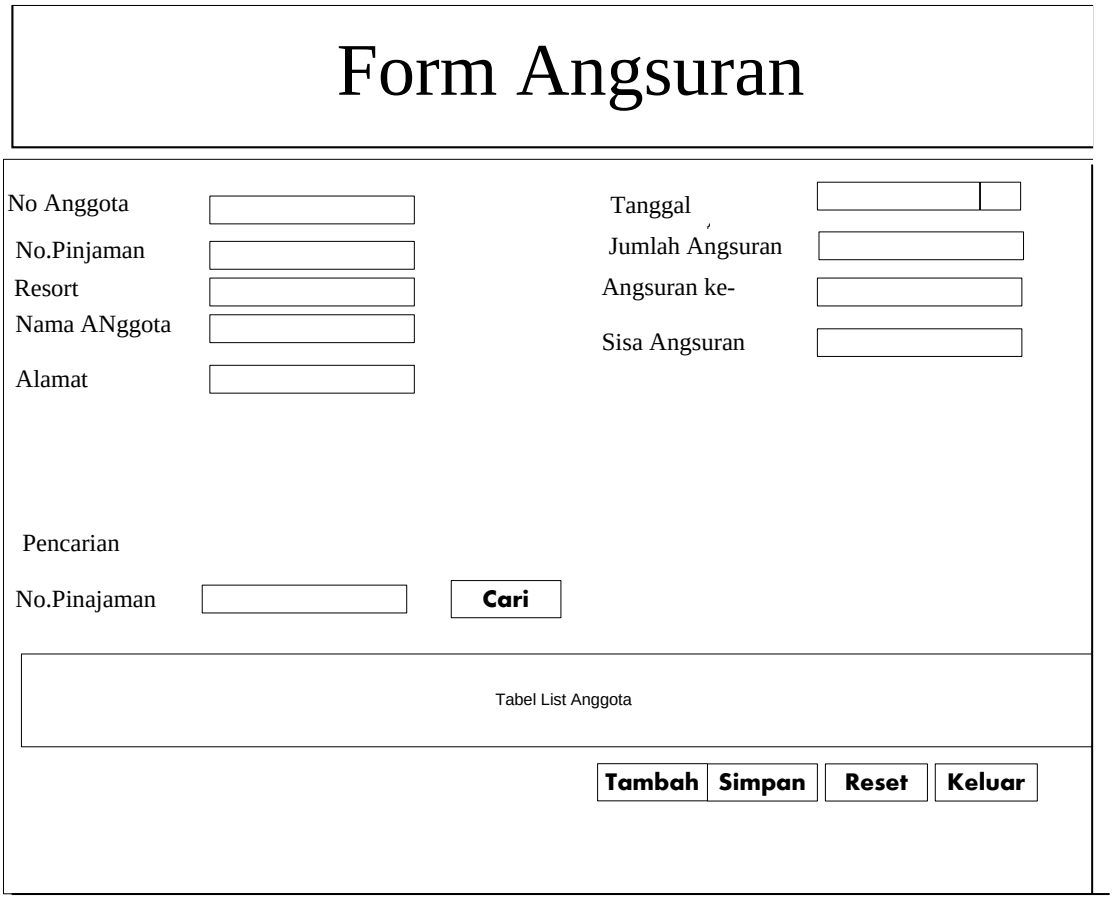
Tampilan yang dirancang ini berfungsi sebagai tampilan untuk pengambilan simpanan anggota



Gambar 4. 40 Rancangan tampilan pengambilan simpanan

6. Tampilan Angsuran

Tampilan ini berfungsi untuk melakukan angsuran dari pinjaman anngota



Gambar 4. 41 Rancangan tampilan angsuran

7. Tampilan untuk mencetak buku anggota

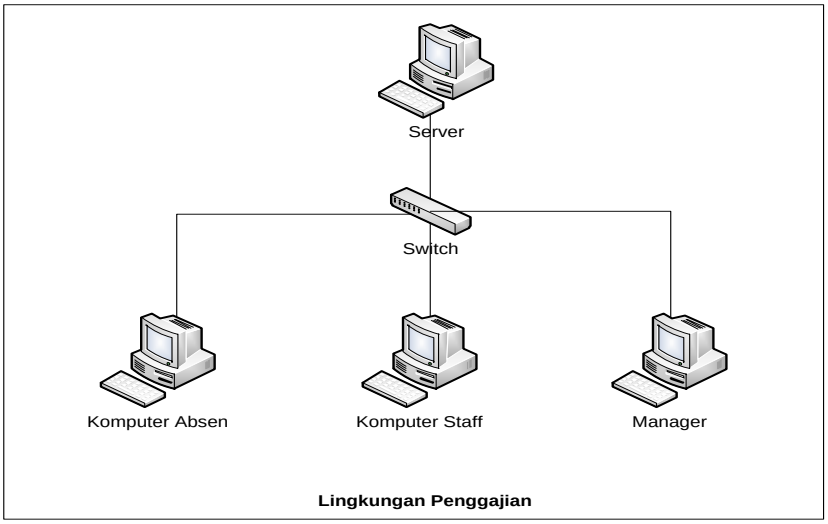
Tampilan ini berfungsi untuk mencetak buku anggota



Gambar 4. 42 Rancangan mencetak buku anggota

4.2.5 Perancangan Arsitektur Jaringan

Arsitektur jaringan yang digunakan adalah sistem *client/server* mempunyai dua komponen utama yaitu komputer *server* dan komputer *client*. *Server* merupakan komputer induk yang melakukan pemrosesan terbanyak untuk memenuhi permintaan-permintaan dari komputer client dan bertindak sebagai *server database* yang menyimpan data. *Client* yaitu komputer yang melakukan pengiriman permintaan-permintaan data pada *server* kemudian menampilkan data tersebut pada *interface* aplikasi yang dimilikinya.



Gambar 4.43 Arsitektur jaringan yang diusulkan

Gambar 4.43 menunjukkan arsitektur jaringan. Disini penulis menggunakan topologi star dengan alasan jika salah satu komputer *client* rusak, komputer yang lain masih dapat diakses dan kontrol menagement lebih mudah karena semuanya terpusat ke server.