

## **BAB III**

### **METODOLOGI DAN PERANCANGAN SISTEM**

#### **3.1 Metode Pengumpulan Data**

Metode pengumpulan data merupakan suatu prosedur untuk memperoleh data yang diperlukan untuk penelitian. Metode pengumpulan data yang akan digunakan dalam penelitian ini, yaitu studi literatur dan studi lapangan. Dalam studi lapangan ini, mencakup dua tahapan yaitu observasi dan wawancara.

##### **3.1.1 Studi Literatur**

Pada tahap studi literatur ini, tahap pengumpulan data dan informasi dengan mempelajari berbagai data literatur yang didapatkan dari sumber seperti buku-buku, jurnal, situs di internet, dan sumber bacaan lain yang berkaitan dengan topik penelitian.

##### **3.1.2 Studi Lapangan**

Pada tahapan studi lapangan ini, dilakukan dengan cara melakukan kunjungan pada tempat di mana studi kasus didapatkan untuk melakukan pengumpulan data secara langsung. Hal ini meliputi:

###### **1. Observasi**

Penulis melakukan pengumpulan data secara langsung dengan melakukan observasi ke tempat penelitian, yaitu Koperasi Mandiri Sejahtera Politeknik Negeri Cilacap pada tanggal 12 April 2022. Hasil observasi yang didapatkan yaitu data dapat digunakan untuk mendukung berlangsungnya penelitian ini. Data yang didapatkan yaitu data anggota aktif dan pasif serta pengurus koperasi, data simpan pinjam dan data formulir permohonan anggota koperasi maupun peminjaman dana

###### **2. Wawancara**

Penulis melakukan wawancara pada tanggal 12 April 2022. Wawancara ini dilaksanakan dengan narasumber yaitu Bapak Ganjar Ndaru Ikhtiangung, S.E., M.M. selaku ketua koperasi dan Ibu Farendah Herwina, S.E. selaku bendahara koperasi. Hasil dari wawancara yang didapatkan penulis yaitu data dan informasi mengenai proses transaksi simpan pinjam, proses pendaftaran menjadi anggota dan pengajuan dana, serta permasalahan yang muncul dalam kedua proses tersebut.

Berdasarkan hasil wawancara dengan narasumber, disampaikan bahwa saat ini proses pendaftaran menjadi anggota dan permohonan peminjaman dana di koperasi masih menggunakan metode semi-manual. Hal ini menyebabkan banyaknya lembar formulir yang menumpuk dan

mengakibatkan kesulitan bagi bendahara dalam mencari dokumen asli dengan cepat. Selain itu, banyaknya kertas juga meningkatkan risiko kehilangan formulir tersebut. Masalah lainnya adalah akses terhadap informasi data transaksi simpan pinjam anggota yang tidak mudah dan cepat. Anggota koperasi tidak dapat melihat secara berkala jumlah simpanan dan pinjaman mereka, sehingga mereka harus menghubungi bendahara melalui chat atau telepon untuk menanyakan informasi tersebut. Situasi ini menyebabkan ketidakmerataan informasi mengenai simpan pinjam dalam koperasi, dan oleh karena itu, diperlukan pengembangan sistem agar proses transaksi menjadi lebih mudah dan informasi dapat diakses oleh seluruh anggota koperasi.

### **3.2 Metode Pengembangan Sistem**

Metode pengembangan sistem merupakan langkah–langkah metode yang dilakukan dalam penelitian. Metode Pengembangan Sistem yang digunakan dalam perancangan Sistem Informasi Simpan Pinjam Berbasis *Website* adalah metode *waterfall*. Metode tersebut dipilih karena mengikuti pendekatan yang terstruktur dan linear. Dengan metode ini, setiap tahap pengembangan dilakukan secara berurutan, mulai dari analisis kebutuhan hingga implementasi dan pengujian. Hal ini memungkinkan pengembangan sistem yang lebih terorganisir, mudah dipahami, dan meminimalkan risiko kesalahan karena setiap langkah sudah diuji dengan baik sebelum melanjutkan ke tahap berikutnya.

#### **3.2.1 Pengumpulan Kebutuhan**

Pada tahap pengumpulan kebutuhan ini dilakukan secara langsung ke tempat penelitian yaitu Koperasi Mandiri Sejahtera Politeknik Negeri Cilacap. Di tahap pengumpulan kebutuhan ini, dilakukan pengumpulan data yang menjadi kebutuhan dasar dari sistem yang akan dibuat yaitu data anggota dan pengurus koperasi, data anggota aktif dan pasif, data simpanan dan pinjaman dan data formulir permohonan menjadi anggota koperasi maupun peminjaman dana.

#### **1. Bahan Penelitian**

Bahan penelitian digunakan penulis untuk mengetahui data apa saja yang diperlukan untuk melakukan penelitian Sistem Informasi Simpan Pinjam Berbasis Website. Bahan penelitian didapatkan dari hasil wawancara dengan ketua dan bendahara Koperasi Mandiri Sejahtera.

### **A. Data Primer**

Data primer adalah data yang diperoleh secara langsung dari narasumber. Data primer yang dibutuhkan dalam Sistem Informasi Simpan Pinjam Berbasis Website sebagai berikut:

- 1) Proses Permohonan Menjadi Anggota Koperasi
- 2) Proses Pengajuan Permohonan Dana
- 3) Proses Transaksi Simpanan dan Pinjaman

### **B. Data Sekunder**

Data sekunder adalah data yang diakses oleh peneliti melalui data yang sudah ada dan mengolahnya untuk keperluan penelitian yang baru. Data sekunder yang dibutuhkan dalam proses pembangunan Sistem Informasi Simpan Pinjam Berbasis Website sebagai berikut:

- 1) Data Pengurus Koperasi
- 2) Data Anggota Koperasi
- 3) Data Simpanan dan Pinjaman
- 4) Data Laporan Aktif dan Pasif

## **2. Jalan Penelitian**

Penelitian ini berisi langkah-langkah yang akan diambil dalam pelaksanaan penelitian. Untuk mengembangkan Sistem Informasi Simpan Pinjam Berbasis Website, penulis menggunakan pemodelan *waterfall* agar lebih mudah dipahami. Berikut adalah tahap-tahap yang dilakukan:

### **1. Perencanaan:**

Pada tahap ini, dilakukan perancangan dan pengembangan Sistem Informasi Simpan Pinjam Berbasis Website berdasarkan komunikasi sebelumnya untuk menentukan kebutuhan fungsional dan kebutuhan non-fungsional, seperti kebutuhan perangkat lunak dan perangkat keras.

### **2. Analisis Kebutuhan Perangkat Lunak:**

Penulis melakukan analisis tentang proses pendaftaran anggota koperasi, proses pengajuan peminjaman dana, dan proses transaksi simpan pinjam di Koperasi Mandiri Sejahtera Politeknik Negeri Cilacap. Analisis ini didapatkan dari hasil observasi dan wawancara. Selanjutnya, dianalisis juga kebutuhan pengguna dan kebutuhan perangkat lunak yang akan digunakan dalam pengembangan Sistem Informasi Simpan Pinjam Berbasis Website.

3. Perancangan:  
Setelah menganalisis kebutuhan sistem, langkah selanjutnya adalah mendesain perangkat lunak dengan menggunakan beberapa diagram seperti *Flowchart*, *Use Case Diagram*, *Sequence diagram*, dan ERD.
4. Implementasi:  
Pada tahap ini, dilakukan pembuatan sistem *menggunakan Visual Studio Code* dengan bahasa PHP sebagai pengembang *website*. Penulis melakukan uji coba program untuk menemukan kesalahan dalam pengkodean, sehingga dapat diperbaiki dan meminimalisir kesalahan pada Sistem Informasi Simpan Pinjam Berbasis Website.
5. Pengujian:  
Pengujian program dilakukan setelah uji coba program sebagai tahap akhir dari pembuatan Sistem Informasi Simpan Pinjam Berbasis Website. Pengujian sistem menggunakan metode pengujian *blackbox*, yang difokuskan pada kebutuhan fungsional dari perangkat lunak. Metode *blackbox* dilakukan untuk memperoleh gambaran kesesuaian antara *input* dan *output*.
6. Pemeliharaan:  
Sistem Informasi Simpan Pinjam Berbasis Website dapat digunakan oleh anggota dan pengurus koperasi. Pemeliharaan mencakup koreksi dari *bug* yang ditemukan setelah melewati tahapan pengujian. Sistem akan diperbarui untuk mengatasi *bug* dan melakukan perubahan desain serta penambahan fitur sesuai permintaan pengguna.

### 3. Alat Penelitian

Dalam melaksanakan penelitian ini, penulis membutuhkan alat penelitian berupa perangkat keras dan perangkat lunak komputer. Kebutuhan perangkat keras dan perangkat lunak yang digunakan dalam pengembangan sistem ini adalah sebagai berikut:

#### A. Analisis Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak merupakan analisis kemampuan perangkat lunak dalam menjalankan sistem yang akan dibangun. Perangkat lunak yang dibutuhkan dalam perancangan sistem ini yaitu yang memiliki kemampuan untuk mengembangkan, menjalankan, serta mendokumentasikan sistem. Adapun kebutuhan perangkat lunak yang digunakan dapat dilihat pada **Tabel 3.1**

**Tabel 3. 1 Kebutuhan Perangkat Lunak**

No.	Nama Perangkat Lunak	Fungsi
1.	<i>Windows 11 Home Single Language, 64 bit</i>	Merupakan sistem operasi untuk menjalankan sistem informasi ini.
2.	<i>Microsoft Visual Studio Code 2023-07-27t20</i>	Merupakan perangkat lunak yang digunakan untuk peng- <i>input</i> -an kode program untuk membuat sistem.
3.	<i>MySQL version 7.4.25</i>	Merupakan DBMS ( <i>Database Management System</i> ).
4.	<i>draw.io</i>	Membuat UML ( <i>Unified Modeling Language</i> ).
5.	<i>yEd Graph Editor 3.22</i>	Membuat ERD ( <i>Entity Relationship Diagram</i> ).
6.	<i>Balsamiq Mockup 3.5.17</i>	Membuat <i>Mockup</i> Aplikasi.
7.	<i>Google Chrome 111.0.5 (64-bit)</i>	<i>Web Browser</i>

### B. Analisis Kebutuhan Perangkat Keras

Kebutuhan perangkat keras yang digunakan untuk perancangan Sistem Informasi Simpan Pinjam Berbasis Website adalah Perangkat keras dengan spesifikasi yang cukup untuk menjalankan perangkat lunak dapat dilihat pada **Tabel 3.2**.

**Tabel 3. 2 Kebutuhan Perangkat Keras**

No	Kebutuhan Perangkat Keras	Keterangan
1.	Prosesor Intel(R) Core(TM) i3-1005G1 CPU @ 1.20GHz 1.19 GHz	Sebagai inti komputer yang memproses instruksi dasar menggerakkan komputer.
2.	<i>Random Access Memory (RAM)</i> 8,00 GB	Tempat penyimpanan sementara.
3.	<i>Solid State Drive</i> 256 Gb	Tempat penyimpanan permanen

### C. Analisis Kebutuhan Pengguna

Setiap aktor yang terlibat dalam penggunaan sistem memiliki fungsi utama sesuai dengan kebutuhan masing-masing pengguna yang dapat dilihat pada **Tabel 3.3**

**Tabel 3.3 Kebutuhan Analisis Pengguna**

No	User	Hak Akses
1.	Pengawas Koperasi	<ol style="list-style-type: none"> <li><b>1. Mengelola data akun</b> <ol style="list-style-type: none"> <li>a. Menampilkan biodata pengawas</li> <li>b. Mengubah biodata pengawas</li> </ol> </li> <li><b>2. Mengelola permohonan dana</b> <ol style="list-style-type: none"> <li>a. Menampilkan detail pengajuan dana oleh ketua</li> <li>b. Memverifikasi permohonan dana</li> </ol> </li> <li><b>3. Mengelola data simpanan</b> <ol style="list-style-type: none"> <li>a. Menampilkan data simpanan pengawas</li> <li>b. Menampilkan data simpanan koperasi</li> <li>c. Menampilkan statistik simpanan pengawas</li> <li>d. Menambah simpanan sukarela</li> </ol> </li> <li><b>4. Mengelola data pinjaman</b> <ol style="list-style-type: none"> <li>a. Menampilkan data pinjaman pengawas</li> <li>b. Menampilkan data pinjaman koperasi</li> <li>c. Menampilkan statistik pinjaman koperasi</li> <li>d. Mengajukan permohonan dana</li> </ol> </li> <li><b>5. Mengelola data anggota koperasi</b> <ol style="list-style-type: none"> <li>a. Menampilkan data keaktifan anggota</li> <li>b. Menampilkan data pengurus koperasi</li> </ol> </li> <li><b>6. Mengelola data laporan koperasi</b> <ol style="list-style-type: none"> <li>a. Menampilkan data laporan koperasi</li> <li>b. Memverifikasi laporan koperasi</li> <li>c. Mencetak data laporan koperasi</li> </ol> </li> </ol>
2.	Atasan Langsung	<ol style="list-style-type: none"> <li><b>1. Mengelola data akun</b> <ol style="list-style-type: none"> <li>a. Mengubah biodata atasan</li> <li>b. Menampilkan biodata atasan</li> </ol> </li> <li><b>2. Mengelola permohonan anggota</b> <ol style="list-style-type: none"> <li>a. Menampilkan detail permohonan anggota</li> <li>b. Memverifikasi permohonan menjadi anggota</li> </ol> </li> <li><b>3. Mengelola data simpanan</b> <ol style="list-style-type: none"> <li>a. Menampilkan data simpanan atasan</li> </ol> </li> </ol>

		<ul style="list-style-type: none"> <li>b. Menampilkan data simpanan koperasi</li> <li>c. Menampilkan statistik simpanan koperasi</li> <li>d. Menambah simpanan sukarela</li> </ul> <p><b>4. Mengelola data pinjaman</b></p> <ul style="list-style-type: none"> <li>a. Menampilkan data pinjaman atasan</li> <li>b. Menampilkan data pinjaman koperasi</li> <li>c. Menampilkan statistik pinjaman koperasi</li> <li>d. Mengajukan permohonan dana</li> <li>e. Meemverifikasi permohonan dana</li> </ul> <p><b>5. Mengelola data anggota koperasi</b></p> <ul style="list-style-type: none"> <li>a. Menampilkan data anggota koperasi</li> <li>b. Menampilkan grafik anggota koperasi</li> <li>c. Menampilkan data keaktifan anggota</li> <li>d. Menampilkan data pengurus koperasi</li> </ul>
3.	Ketua Koperasi	<p><b>7. Mengelola data akun</b></p> <ul style="list-style-type: none"> <li>a. Menampilkan biodata ketua</li> <li>b. Mengubah biodata ketua</li> </ul> <p><b>8. Mengelola permohonan dana</b></p> <ul style="list-style-type: none"> <li>c. Menampilkan detail pengajuan dana</li> <li>d. Memverifikasi permohonan dana</li> </ul> <p><b>9. Mengelola data simpanan</b></p> <ul style="list-style-type: none"> <li>e. Menampilkan data simpanan ketua</li> <li>f. Menampilkan data simpanan koperasi</li> <li>g. Menampilkan statistik simpanan koperasi</li> <li>h. Menambah simpanan sukarela</li> </ul> <p><b>10. Mengelola data pinjaman</b></p> <ul style="list-style-type: none"> <li>e. Menampilkan data pinjaman ketua</li> <li>f. Menampilkan data pinjaman koperasi</li> <li>g. Menampilkan statistik pinjaman koperasi</li> <li>h. Mengajukan permohonan dana</li> </ul> <p><b>11. Mengelola data anggota koperasi</b></p> <ul style="list-style-type: none"> <li>a. Menampilkan data anggota koperasi</li> <li>b. Menampilkan grafik anggota koperasi</li> <li>c. Menampilkan data keaktifan anggota</li> <li>d. Menampilkan data pengurus koperasi</li> </ul> <p><b>12. Mengelola data laporan koperasi</b></p> <ul style="list-style-type: none"> <li>a. Menampilkan data laporan koperasi</li> <li>b. Memverifikasi laporan koperasi</li> </ul>

		c. Mencetak data laporan koperasi
4.	Bendahara	<ol style="list-style-type: none"> <li>1. <b>Mengelola data akun</b> <ol style="list-style-type: none"> <li>a. Menampilkan biodata bendahara</li> <li>b. Mengubah biodata bendahara</li> </ol> </li> <li>2. <b>Mengelola data permohonan anggota</b> <ol style="list-style-type: none"> <li>a. Menampilkan detail permohonan anggota</li> <li>b. Memverifikasi permohonan anggota</li> </ol> </li> <li>3. <b>Mengelola data simpanan</b> <ol style="list-style-type: none"> <li>a. Menampilkan data simpanan bendahara</li> <li>b. Menampilkan data simpanan koperasi</li> <li>c. Menampilkan statistik simpanan</li> <li>d. Memverifikasi simpanan sukarela</li> <li>e. Menambah simpanan wajib</li> <li>f. Menambah simpanan sukarela</li> </ol> </li> <li>4. <b>Mengelola data Pinjaman</b> <ol style="list-style-type: none"> <li>a. Menampilkan data pinjaman bendahara</li> <li>b. Menampilkan data pinjaman koperasi</li> <li>c. Menampilkan statistik pinjaman koperasi</li> <li>d. Mengajukan permohonan dana</li> </ol> </li> <li>5. <b>Mengelola data anggota koperasi</b> <ol style="list-style-type: none"> <li>a. Menampilkan data anggota koperasi</li> <li>b. Menampilkan grafik anggota</li> <li>c. Mengubah status anggota koperasi</li> <li>d. Menampilkan data keaktifan anggota koperasi</li> <li>e. Menampilkan data pengurus koperasi</li> </ol> </li> <li>6. <b>Mengelola data laporan koperasi</b> <ol style="list-style-type: none"> <li>a. Menampilkan data laporan koperasi</li> <li>b. Mengajukan data laporan koperasi</li> <li>f. Mencetak data laporan koperasi</li> </ol> </li> </ol>
5.	Anggota Koperasi	<ol style="list-style-type: none"> <li>1. <b>Mengelola data akun</b> <ol style="list-style-type: none"> <li>a. Mengubah biodata pribadi</li> <li>b. Menampilkan biodata pribadi</li> <li>c. Menampilkan formulir pendaftaran</li> <li>d. Mencetak formulir pendaftaran</li> </ol> </li> <li>2. <b>Mengelola data simpanan</b> <ol style="list-style-type: none"> <li>a. Menampilkan statistik simpanan pribadi</li> <li>b. Menampilkan riwayat penyimpanan</li> <li>c. Menambah simpanan sukarela</li> </ol> </li> </ol>

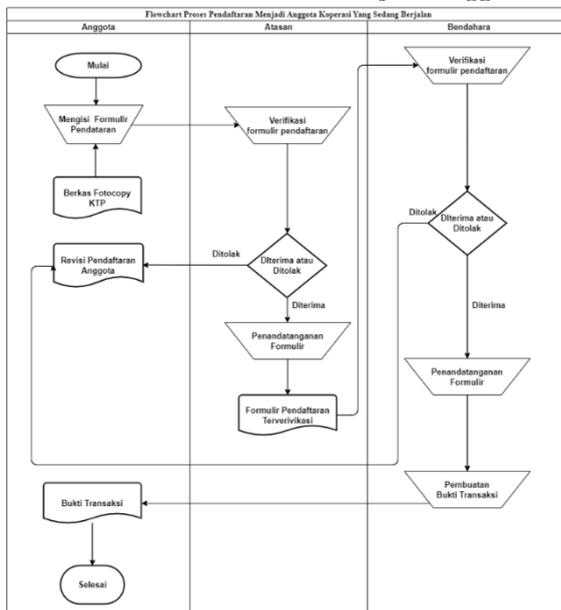
		<p><b>3. Mengelola data pinjaman</b></p> <ol style="list-style-type: none"> <li>a. Mengajukan permohonan dana</li> <li>b. Menampilkan statistik pinjaman pribadi</li> <li>c. Menampilkan riwayat peminjaman</li> <li>d. Mencetak formulir permohonan dana</li> </ol> <p><b>4. Mengelola data anggota</b></p> <ol style="list-style-type: none"> <li>a. Menampilkan data pengurus koperasi</li> <li>b. Menampilkan data keaktifan anggota</li> </ol>
--	--	---

### 3.2.2 Pemodelan Sistem

Setelah dilakukan analisa kebutuhan, maka pada tahap ini akan dibangun sebuah pemodelan sesuai dengan data yang didapatkan pada saat analisa kebutuhan. Pemodelan yang dibangun yaitu *flowchart* sistem yang sedang berjalan dan akan dikembangkan, *use case diagram*, *sequence diagram*, ERD, dan *mock-up* sistem.

#### A. Analisis Sistem yang Sedang Berjalan

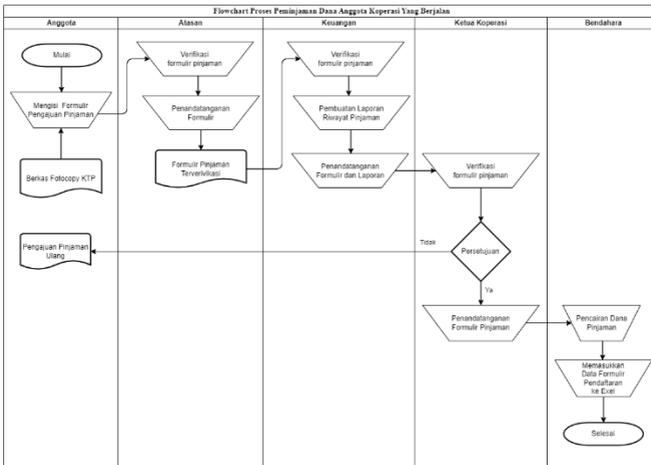
##### 1. Flowchart Proses Pendaftaran Menjadi Anggota Koperasi



**Gambar 3. 1** Flowchart Permohonan Menjadi Anggota Koperasi

Proses pendaftaran anggota koperasi dimulai dengan anggota datang langsung ke koperasi dan mengisi formulir dengan menggunakan pena yang berisi informasi pribadi seperti nomor pegawai, nama lengkap, tempat dan tanggal lahir, alamat KTP, alamat domisili, nomor telepon, nomor KTP, pekerjaan/jabatan, jenis kelamin, serta melampirkan satu *fotocopy* KTP. Setelah mengisi formulir, lembar formulir tersebut akan diserahkan ke atasan untuk diverifikasi. Setelah proses verifikasi selesai, atasan akan menandatangani lembar formulir tersebut kemudian diteruskan ke bendahara. Bendahara akan melakukan verifikasi data dan memberikan pilihan terkait jenis pembayaran yang akan dilakukan oleh anggota, anggota harus membayar uang administrasi, simpanan pokok, simpanan wajib, dan simpanan sukarela sesuai ketentuan yang ditetapkan oleh koperasi. Setelah anggota menentukan jenis pembayaran dan melakukan transaksi maka bendahara akan menandatangani formulir tersebut dan memberikan bukti transaksi kepada anggota.

**2. Flowchart Proses Peminjaman Dana Anggota Koperasi**



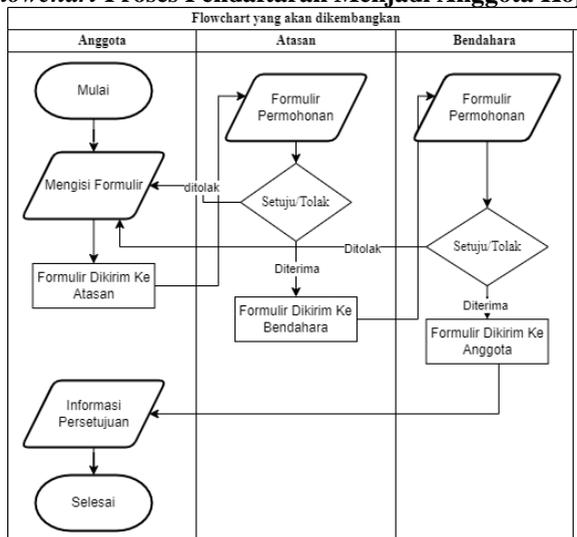
**Gambar 3. 2** Flowchart Proses Peminjaman Dana

Proses pengajuan dana dimulai dengan kedatangan anggota ke koperasi untuk mengisi formulir biodata diri kembali, termasuk nominal pinjaman, jangka waktu, dan alasan pengajuan dana. Setelah dilengkapi, formulir tersebut diserahkan untuk diverifikasi oleh atasan. Setelah verifikasi selesai, atasan menandatangani formulir dan dokumen tersebut

diserahkan ke bagian keuangan untuk diverifikasi lebih lanjut. Bagian keuangan membuat laporan tentang riwayat peminjaman pemohon dan menyampaikannya kepada ketua koperasi setelah ditandatangani. Ketua koperasi mempertimbangkan formulir berdasarkan laporan tersebut. Jika permohonan ditolak, pemohon diberitahu alasan dan diberikan solusi untuk mengajukan ulang. Jika disetujui, formulir ditandatangani dan diserahkan ke bendahara. Bendahara memasukkan data ke dalam *microsoft excel* dan melakukan proses pencairan dana agar dapat digunakan oleh pemohon.

## B. Sistem Yang Akan Dikembangkan

### 1. Flowchart Proses Pendaftaran Menjadi Anggota Koperasi



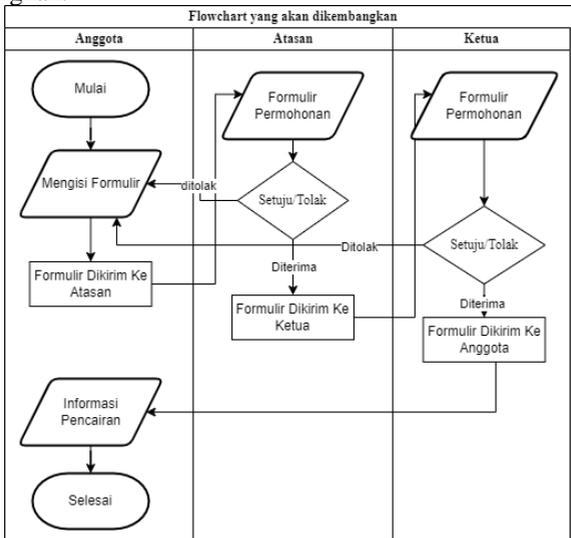
**Gambar 3. 3** Flowchart Proses Pendaftaran Anggota Koperasi

Proses pendaftaran anggota koperasi dimulai dengan mengisi nama lengkap dan nomor *whatsapp* yang aktif untuk menerima kode OTP. Setelah kode OTP dimasukkan, anggota akan diminta mengisi formulir pendaftaran yang mencakup informasi seperti nomor pegawai, nama lengkap, tempat dan tanggal lahir, alamat KTP dan domisili, nomor telepon, nomor KTP, pekerjaan/jabatan, jenis kelamin, nama dan nomor rekening, serta mengunggah foto KTP. Selanjutnya, anggota harus memenuhi persyaratan keanggotaan koperasi yang meliputi pembayaran

uang administrasi, simpanan pokok, simpanan wajib, dan simpanan sukarela. Pembayaran dapat dilakukan melalui tiga opsi, yaitu tunai, non-tunai, dan pemotongan pendapatan pada bulan berikutnya. Setelah semua formulir terisi lengkap, anggota dapat menekan *button* "ajukan" pada sistem untuk mengirimkan formulir. Kemudian, notifikasi *whatsapp* akan memberitahukan bahwa formulir telah diajukan. Formulir tersebut akan diverifikasi oleh atasan yang berhak untuk menyetujui atau menolak. Jika ditolak, anggota akan diminta untuk merevisi permohonan dan akan diberi informasi melalui *whatsapp* mengenai alasan penolakan yang ditampilkan pada *interface* permohonan. Jika formulir disetujui, atasan akan menekan *button* setuju, sebelum diteruskan ke bendahara untuk verifikasi. Jika bendahara menolak, proses yang sama akan berlaku seperti pada tahapan penolakan oleh atasan. Namun, jika disetujui, bendahara akan menekan *button* setuju. Setelah menjadi anggota koperasi, anggota tersebut dapat mengajukan permohonan untuk peminjaman dana.

2. **Flowchart Proses Pengajuan Peminjaman Dana**

Pada **Gambar 3.5** dijelaskan bagaimana alur perancangan proses pengajuan peminjaman dana pada Koperasi Mandiri Sejahtera yang akan dikembangkan.

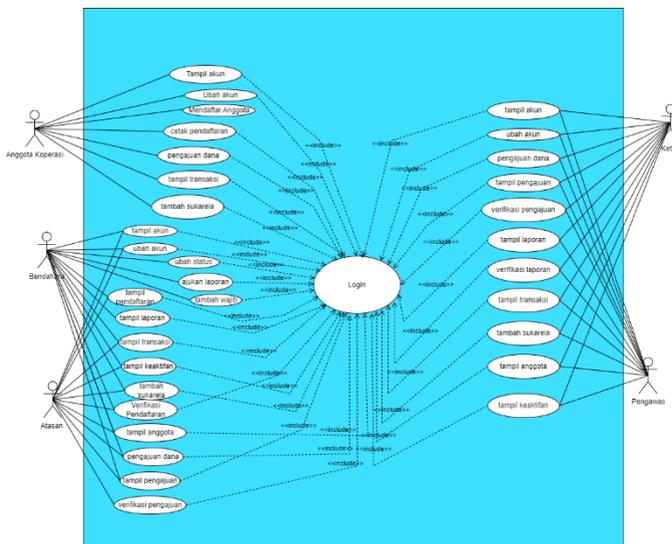


**Gambar 3. 4** Flowchart Proses Pengajuan Peminjaman Dana

Proses pengajuan dana dimulai dari anggota melengkapi formulir pada *form* pengajuan dana seperti nominal pinjaman, jangka waktu, dan alasan pengajuan dana. Jika semua data sudah dilengkapi, pilih *button* ajukan untuk mengajukan formulir. Kemudian notifikasi *whatsapp* akan muncul untuk memberikan informasi bahwa formulir telah diajukan. Formulir tersebut akan langsung diverifikasi oleh atasan yang berhak memilih keputusan diterima maupun ditolak. Jika ditolak, anggota akan diberitahu alasan dan solusinya yang diberitahukan melalui notifikasi *whatsapp* dan ditampilkan juga pada *interface* pengajuan. Jika disetujui, atasan akan memilih *button* setuju yang menerapkan fitur tanda tangan *digital* pada formulir kemudian diteruskan ke Ketua untuk memverifikasi kembali permohonan tersebut, proses validasi sama seperti atasan. Setelah proses pencairan berhasil, anggota akan menerima notifikasi *whatsapp* bahwa pencairan dana berhasil.

### C. Use Case Diagram

Berikut adalah *use case* diagram Sistem Informasi Simpan Pinjam Berbasis *Website* dijelaskan pada **Gambar 3.5**.



**Gambar 3.5** Use Case Diagram

Pada **Gambar 3.6** merupakan *use case* diagram Sistem Informasi Simpan Pinjam Berbasis *Website* di atas, terdapat 5 aktor yang memiliki

hak akses. Aktor pada sistem ini terdiri dari pengawas, atasan, ketua, bendahara, dan anggota. Setiap aktor memiliki hak akses yang berbeda-beda. Aktor-aktor tersebut diwajibkan masuk sebelum mengakses sistem tersebut.

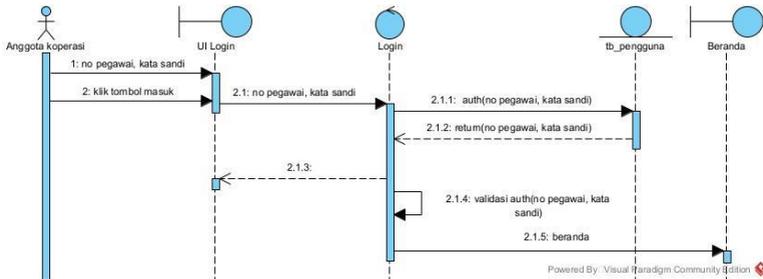
Berikut hak akses yang dapat dijalankan oleh aktor pada sistem, antara lain:

1. Pengawas Koperasi memiliki hak akses untuk *login*, melihat data anggota, simpan pinjam, grafik transaksi, laporan koperasi, dan melihat grafik anggota, memverifikasi permohonan peminjaman dana oleh ketua koperasi, mengajukan peminjaman dana serta memverifikasi laporan koperasi.
2. Atasan langsung memiliki hak akses untuk *login*, melihat data simpan pinjam, grafik transaksi, laporan koperasi, data dan melihat grafik anggota, memverifikasi permohonan menjadi anggota dan dana, serta mengajukan peminjaman dana.
3. Ketua koperasi memiliki hak akses untuk *login*, melihat data simpan pinjam, grafik transaksi, laporan koperasi, data dan melihat grafik anggota, memverifikasi permohonan peminjaman dana, serta memverifikasi laporan koperasi.
4. Bendahara memiliki hak akses untuk *login*, melihat data simpan pinjam, grafik transaksi, laporan koperasi, data dan melihat grafik anggota, mengubah data simpan pinjam, data anggota, laporan koperasi dan mengubah status anggota, memverifikasi permohonan menjadi anggota, mengajukan permohonan dana serta mengajukan laporan koperasi.
5. Anggota koperasi memiliki hak akses untuk *login*, melihat data simpanan dan pinjaman, melihat grafik transaksi, mengajukan permohonan dana.

#### **D. Sequence Diagram**

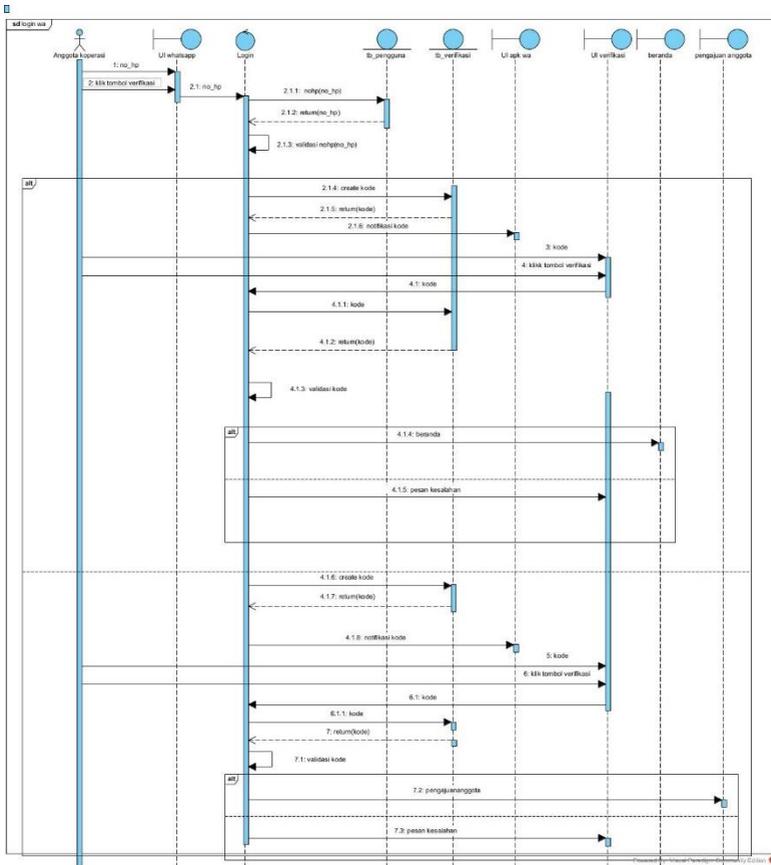
*Sequence* diagram menunjukkan perilaku dari objek di *use case diagram* dengan menggambarkan urutan langkah untuk menyelesaikan sebuah tugas dalam sebuah *timeline*.

## 1. Sequence Diagram Login



**Gambar 3.6** Sequence Diagram Login Nomor Pegawai

**Gambar 3.7** menjelaskan mengenai *sequence diagram* pada proses *login* yang dilakukan oleh aktor anggota koperasi yang meliputi anggota, bendahara, atasan, ketua dan pegawai. Proses *login* dimulai dari aktor menginputkan nomor pegawai dan kata sandi. Kemudian, UI masuk akan mengirimkan nomor pegawai dan kata sandi ke *controller login*. Dari UI masuk akan mengirimkan nomor pegawai dan kata sandi ke *controller login*, selanjutnya *controller login* akan mengambil data nomor pegawai dan kata sandi ke tabel *tb\_pengguna* dan *tb\_pengguna* akan memberikan *return* data pengguna ke *controller login*. Kemudian *controller login* melakukan verifikasi nomor pegawai dan kata sandi. Jika benar, maka aktor berhasil login dan masuk ke UI beranda, sedangkan jika salah, tampil notifikasi nomor pegawai atau kata sandi salah di UI masuk.

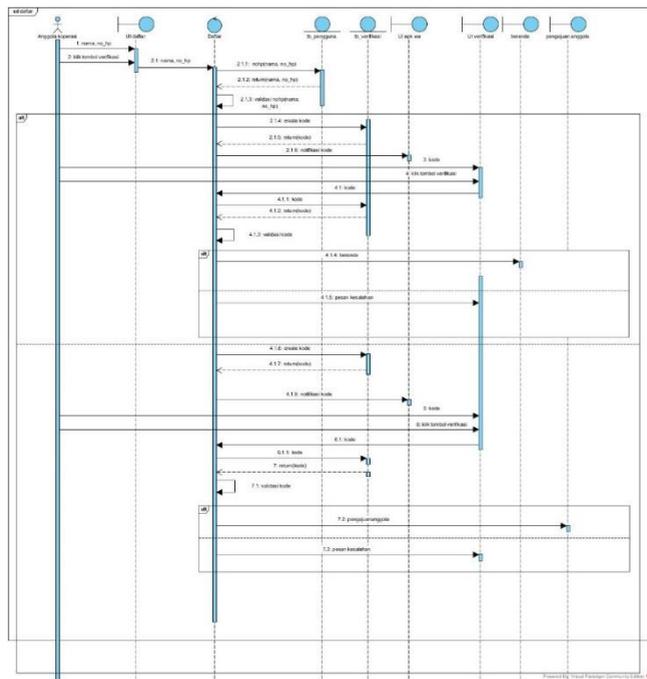


**Gambar 3.7** Sequence Diagram Login Whatsapp

**Gambar 3.8** menjelaskan mengenai *sequence diagram* pada proses login whatsapp yang dilakukan oleh anggota koperasi yang meliputi anggota, bendahara, atasan, ketua dan pengawas. Proses *login* dimulai dengan menekan tombol *whatsapp* pada UI masuk, maka sistem akan menampilkan UI *whatsapp* untuk menginputkan nomor *whatsapp* aktif, lalu UI *whatsapp* akan mengirimkan nomor *whatsapp* ke *controller Login* untuk dilakukan pemanggilan data nomor *whatsapp* pada *tb\_pengguna*, kemudian dilakukan validasi dari data yang ada pada tabel *tb\_pengguna*. Jika data sudah ada, maka *controller Login* membuat data

yang selanjutnya disimpan di `tb_verifikasi`, lalu dikirimkan ke UI aplikasi *whatsapp*. Selanjutnya, anggota koperasi memasukkan kode verifikasi ke UI verifikasi, kemudian *controller login* akan melakukan verifikasi terhadap kode tersebut. Jika kode benar maka akan masuk ke halaman beranda. Jika salah maka akan muncul pesan kesalahan pada UI verifikasi. Selanjutnya, jika validasi nomor whatsapp belum ada pada `tb_pengguna`, maka *controller Login* akan tetap membuat kode yang disimpan pada `tb_verifikasi` yang dikirim ke UI aplikasi *whatsapp*. Kemudian anggota koperasi memasukkan kode tersebut pada UI verifikasi, kemudian *controller login* akan melakukan verifikasi terhadap kode tersebut. Jika kode benar maka akan masuk ke halaman pengajuan anggota. Jika salah maka akan muncul pesan kesalahan pada UI verifikasi.

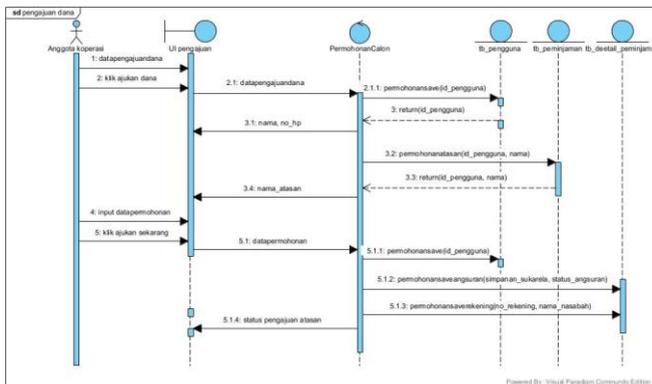
## 2. Sequence Diagram Pendaftaran Menjadi Anggota Koperasi



Gambar 3. 8 Sequence Diagram Pendaftaran Anggota Koperasi

**Gambar 3.9** menjelaskan mengenai *sequence diagram* pada proses pendaftaran menjadi anggota koperasi yang dilakukan oleh anggota. Proses pendaftaran dimulai dengan memilih *link* buat akun pada UI masuk, maka sistem akan menampilkan UI Daftar untuk menginputkan nama dan nomor *whatsapp* aktif, lalu UI Daftar akan mengirimkan nama dan nomor *whatsapp* ke *controller* Daftar untuk dilakukan pemanggilan data nama dan nomor *whatsapp* pada *tb\_pengguna*, kemudian dilakukan validasi dari data yang ada pada tabel *tb\_pengguna*. Jika data sudah ada, maka *controller* Daftar membuat data yang selanjutnya disimpan di *tb\_verifikasi*, lalu dikirimkan ke UI *apk whatsapp*. Selanjutnya, anggota koperasi memasukkan kode verifikasi ke UI verifikasi, kemudian *controller* Daftar akan melakukan verifikasi kode tersebut. Jika kode benar maka akan masuk ke halaman beranda anggota. Jika salah maka akan muncul pesan kesalahan pada UI verifikasi. Selanjutnya, jika validasi nama dan nomor *whatsapp* belum ada pada *tb\_pengguna*, maka *controller* Daftar membuat kode yang disimpan *tb\_verifikasi* yang dikirim ke UI *apk whatsapp*. Kemudian anggota memasukkan kode pada UI verifikasi, kemudian *controller* Daftar melakukan verifikasi kode. Jika kode benar maka masuk ke pengajuan anggota. Jika salah maka muncul pesan kesalahan pada UI verifikasi.

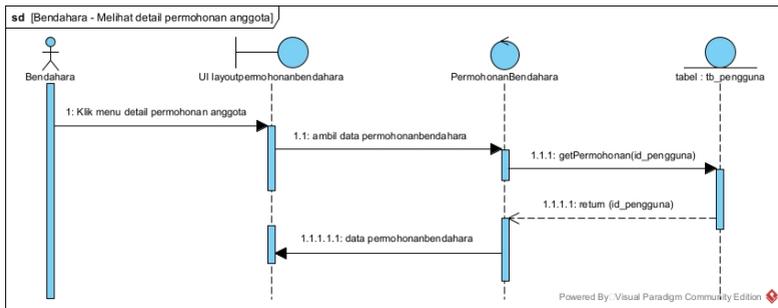
### 3. Sequence Diagram Mengajukan Permohonan Menjadi Anggota



**Gambar 3. 9** Sequence Diagram Permohonan Menjadi Anggota

**Gambar 3.10** menjelaskan mengenai *sequence diagram* yang dilakukan oleh anggota koperasi (pegawai), anggota koperasi memanggil data permohonan pada UI pengajuan anggota, UI pengajuan anggota mengirimkan data permohonan ke *controller*. Selanjutnya *controller* PermohonanCalon akan menjalankan fungsi permohonan save dengan parameter id pengguna ke tabel pengguna. Tabel pengguna akan memberikan data permohonan berdasarkan id pengguna ke permohonan *controller*. permohonan calon akan menampilkan nama dan no hp ke UI pengajuan anggota, selanjutnya permohonan calon menjalankan fungsi permohonan atasan dengan parameter id\_pengguna ke tabel atasan. Tabel atasan akan memberikan data permohonan atasan berdasar id pengguna, selanjutnya permohonan calon menampilkan nama atasan. Anggota koperasi kemudian memasukkan data permohonan ke UI pengajuan anggota, yang dikirimkan ke *controller* PermohonanCalon. *Controller* PermohonanCalon akan menjalankan fungsi permohonan save untuk menyimpan data pada tb\_pengguna, lalu menjalankan fungsi permohonan angsuran dan permohonan save rekening yang akan disimpan pada tb\_permohonan. Setelah data berhasil disimpan, maka *controller* PermohonanCalon akan menampilkan status pengajuan pada UI pengajuan anggota

#### 4. *Sequence Diagram* Menampilkan Detail Permohonan Anggota

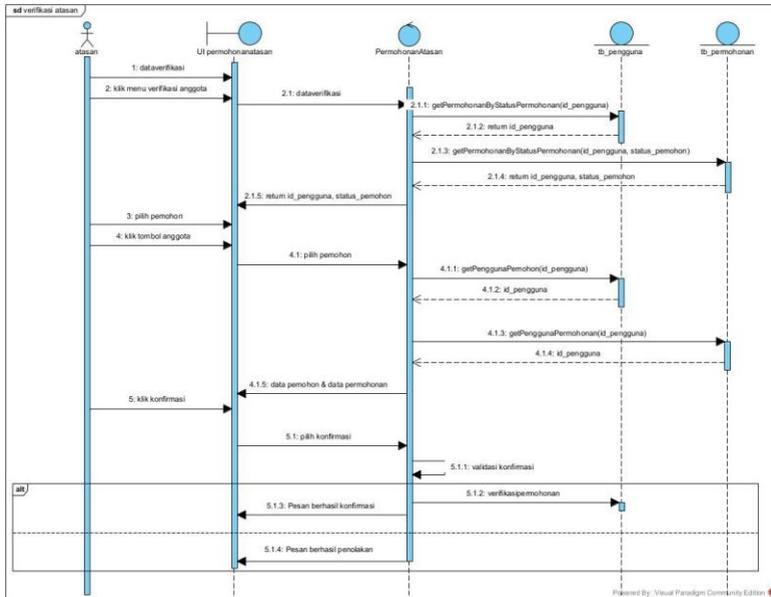


**Gambar 3. 10** *Sequence Diagram* Tampil Permohonan Anggota

**Gambar 3.11** menjelaskan *sequence diagram* untuk menampilkan detail permohonan anggota yang dilakukan oleh atasan dan bendahara. Aktor melakukan proses lihat data detail permohonan anggota yang diambil dari tabel data tb\_pengguna. Kemudian *controller*

PermohonanBendahara akan mengambil data tersebut dari tabel `tb_pengguna`. Lalu mengirimkan kembali data yang telah ditemukan ke *controller* PermohonanBendahara dan diteruskan ke UI.

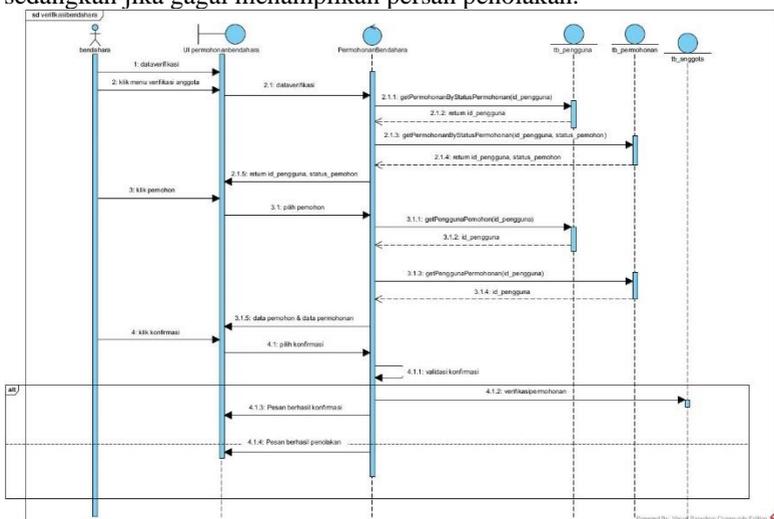
## 5. Sequence Diagram Memverifikasi Permohonan Menjadi Anggota



**Gambar 3. 11** Sequence Diagram Memverifikasi Anggota oleh Atasan

**Gambar 3.12** menjelaskan mengenai verifikasi permohonan anggota yang dilakukan oleh atasan. Atasan memanggil data verifikasi ke UI permohonanatasan, UI permohonan atasan akan meminta data verifikasi ke *controller* permohonan atasan, selanjutnya *controler* akan memanggil fungsi `getPermohonanByStatusPermohonan` berdasarkan `id_pengguna` ke `tb_pengguna`, dari `tb_pengguna` memberikan data yang diminta ke *controller*, selanjutnya *controller* memanggil fungsi `getPermohonanByStatusPermohonan` berdasarkan `id_pengguna` dan `status_permohon` ke `tb_permohonan` dan `tb_permohonan` akan memberikan data yang diminta ke *controller*. `PermohonanAtasan` *controller* menampilkan `id` pengguna dan status permohonan.

memilih pemohon di UI permohonan atasan selanjutnya akan mengambil data melalui *controller* permohonan atasan, selanjutnya *controller* memanggil fungsi `getPeggunaPemohon` berdasar `id_pegawai` ke `tb_pegguna`. `tb_pegguna` akan memberikan data berdasarkan `id_pegguna` ke *controller*. dari *controller* menampilkan data pemohon dan permohonan ke UI `Ppermohonanatasan`. Atasan memilih konfirmasi di UI `Permohonanatasan`, selanjutnya akan mengambil data konfirmasi melalui *controller* permohonan atasan, selanjutnya *controller* melakukan cek validasi. Jika benar akan melakukan verifikasi permohonan ke `tb_pegguna` dan menampilkan pesan berhasil di UI permohonan atasan, sedangkan jika gagal menampilkan pesan penolakan.

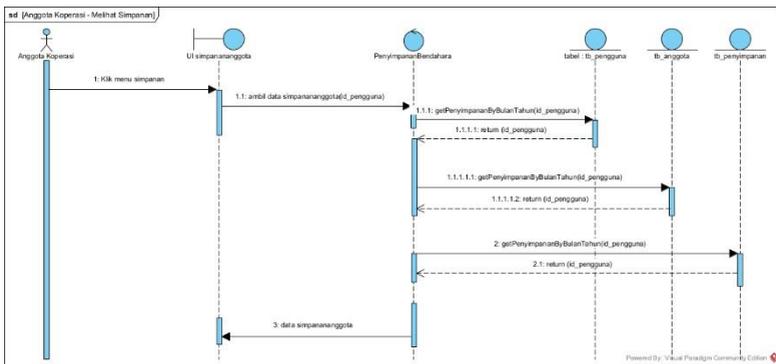


**Gambar 3. 12** Sequence Diagram Memverifikasi Permohonan Menjadi Anggota Oleh Bendahara

**Gambar 3.13** menjelaskan mengenai verifikasi permohonan anggota yang dilakukan oleh bendahara. Bendahara memanggil data verifikasi ke UI permohonan bendahara, UI permohonan bendahara akan meminta data verifikasi ke *controller* permohonan bendahara, selanjutnya *controller* akan memanggil fungsi `getPermohonanByStatusPermohonan` berdasarkan `id_pegawai` ke `tb_pegguna`, dari `tb_pegguna` memberikan data yang diminta ke *controller*, selanjutnya *controller* memanggil fungsi `getPermohonanByStatusPermohonan` berdasarkan `id_pegawai` dan

status\_pemohon ke tb\_permohonan dan tb\_permohonan akan memberikan data yang diminta ke *controller*. PermohonanBendahara *controller* menampilkan id pengguna dan status pemohon. Bendahara memilih pemohon di UI permohonan bendahara selanjutnya akan mengambil data melalui *controller* permohonan bendahara, selanjutnya *controller* memanggil fungsi `getPenggunaPemohon` berdasar id\_pegawai ke tb\_pengguna. tb\_pengguna akan memberikan data berdasarkan id\_pengguna ke *controller*. dari *controller* menampilkan data pemohon dan permohonan ke UI permohonanbendahara. Bendahara memilih konfirmasi di UI Permohonanbendahara, selanjutnya akan mengambil data konfirmasi melalui *controller* permohonan bendahara, selanjutnya *controller* melakukan cek validasi. Jika benar akan melakukan verifikasi permohonan ke tb\_anggota dan menampilkan pesan berhasil di UI permohonan bendahara, sedangkan jika gagal menampilkan pesan penolakan.

## 6. Sequence Diagram Menampilkan Data Simpanan

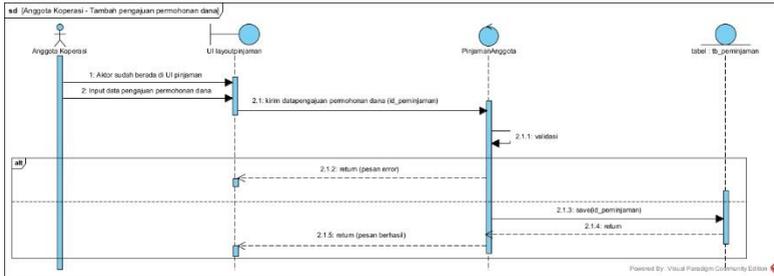


**Gambar 3. 13** Sequence Diagram Menampilkan Data Simpanan

**Gambar 3.14** menjelaskan mengenai *sequence diagram* menampilkan data simpanan yang dilakukan oleh aktor anggota koperasi (anggota, bendahara, atasan, pengawas dan ketua). Aktor melakukan proses tampil data simpanan yang diambil dari tabel data tb\_pengguna, tb\_anggota, tb\_simpanan. Kemudian *controller* PenyimpananBendahara akan mengambil data tersebut dari tabel tb\_pengguna, tb\_anggota, tb\_simpanan. Lalu mengirimkan kembali

data yang telah ditemukan ke *controller* PenyimpananBendahara dan diteruskan ke UI.

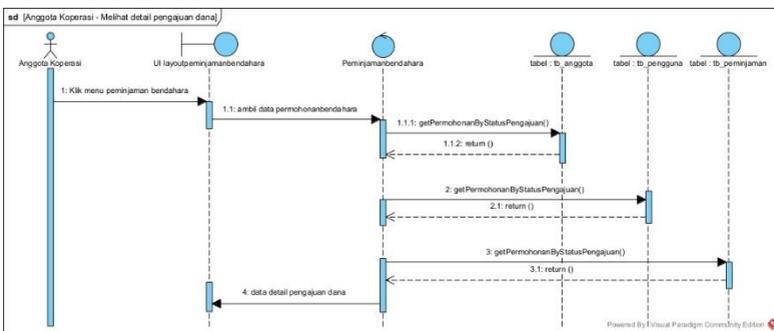
## 7. Sequence Diagram Mengajukan Permohonan Dana



**Gambar 3. 14** Sequence Diagram Mengajukan Permohonan Dana

**Gambar 3.15** menjelaskan mengenai *sequence diagram* mengajukan atau tambah pengajuan permohonan dana yang dilakukan oleh anggota koperasi (anggota, bendahara, atasan, pengawas dan ketua), pada antamuka layoutpinjaman (UI), aktor berada di UI layoutpinjaman kemudian memasukkan data yang nantinya akan dikirimkan ke *controller* PinjamanAnggota. Kemudian data yang dimasukkan akan dilakukan *validation*. Jika tidak berhasil, maka akan muncul pesan gagal. Jika tervalidasi maka muncul pesan berhasil, lalu data tersimpan pada tabel `tb_peminjaman`.

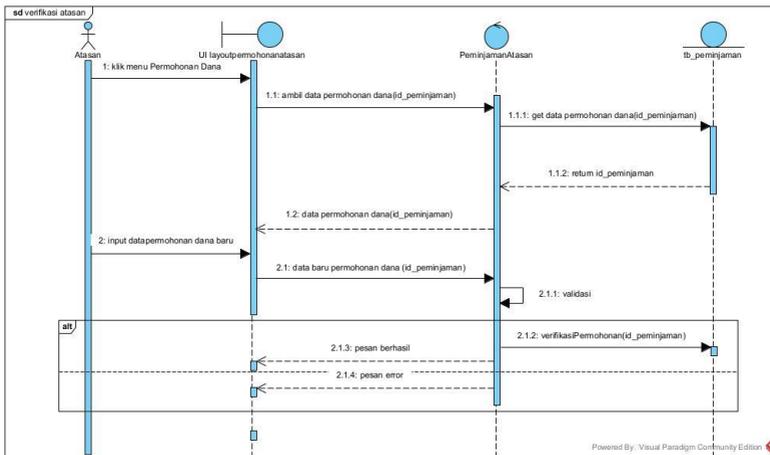
## 8. Sequence Diagram Menampilkan Detail Permohonan Dana



**Gambar 3. 15** Sequence Diagram Tampil Detail Permohonan Dana

**Gambar 3.16** menjelaskan *sequence diagram* untuk melihat data detail pengajuan dana yang dilakukan oleh aktor anggota koperasi (bendahara, dan ketua). Aktor melakukan proses lihat data detail pengajuan dana yang diambil dari tabel data tb\_anggota, tb\_pengguna, tb\_peminjaman. Kemudian *controller* PenyimpananBendahara akan mengambil data tersebut dari tabel tb\_anggota, tb\_pengguna, tb\_peminjaman. Lalu mengirimkan kembali data yang telah ditemukan ke *controller* PenyimpananBendahara dan diteruskan ke UI.

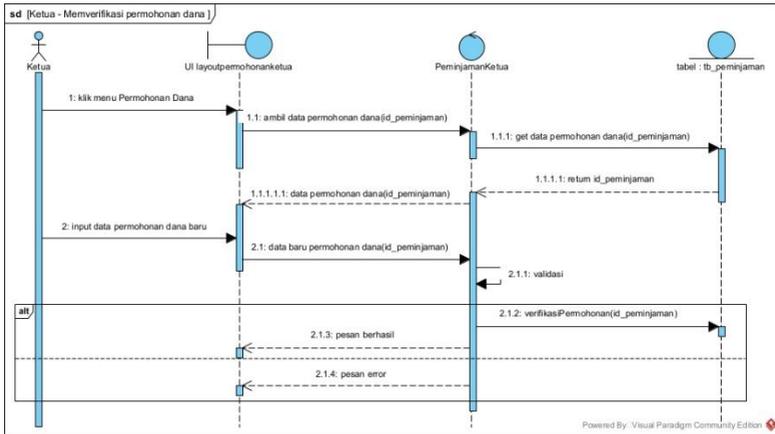
**9. Sequence Diagram Memverifikasi Permohonan Dana**



**Gambar 3. 16** Sequence Diagram Memverifikasi Permohonan Dana Oleh Atasan

**Gambar 3.17** menjelaskan mengenai *sequence diagram* verifikasi permohonan dana yang dilakukan oleh aktor atasan. Aktor memilih verifikasi permohonan pada UI, kemudian memilih salah satu data dan mengirimkan id\_peminjaman kepada *controller* PeminjamanAtasan untuk mendapatkan data, berdasarkan data yang telah dikirimkan. Data hasil pencarian akan dikembalikan kepada *controller* PeminjamanAtasan, lalu selanjutnya akan ditampilkan pada UI. Aktor memasukan data, setelah itu data masuk kedalam *controller* PeminjamanAtasan untuk dilakukan *validation*. Jika tervalidasi maka data yang diinputkan akan masuk ke tabel tb\_peminjaman dan muncul

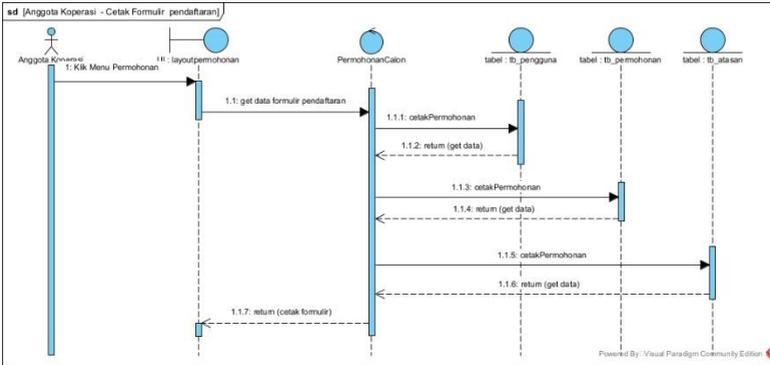
pesan berhasil. Jika terdapat kesalahan maka data tidak disimpan dan akan muncul pesan *error*.



**Gambar 3. 17** Sequence Diagram Memverifikasi Permohonan Dana Oleh Ketua

**Gambar 3.17** menjelaskan mengenai *sequence diagram* verifikasi permohonan dana yang dilakukan oleh aktor ketua. Aktor memilih verifikasi permohonan pada UI, kemudian memilih salah satu data dan mengirimkan `id_peminjaman` kepada *controller* PeminjamanKetua untuk mendapatkan data, berdasarkan data yang telah dikirimkan. Data hasil pencarian akan dikembalikan kepada *controller* PeminjamanKetua, lalu selanjutnya akan ditampilkan pada UI. Aktor memasukkan data, setelah itu data masuk kedalam *controller* PeminjamanKetua untuk dilakukan *validation*. Jika tervalidasi maka data yang diinputkan akan masuk ke tabel `tb_peminjaman` dan muncul pesan berhasil. Jika terdapat kesalahan maka data tidak disimpan dan akan muncul pesan *error*.

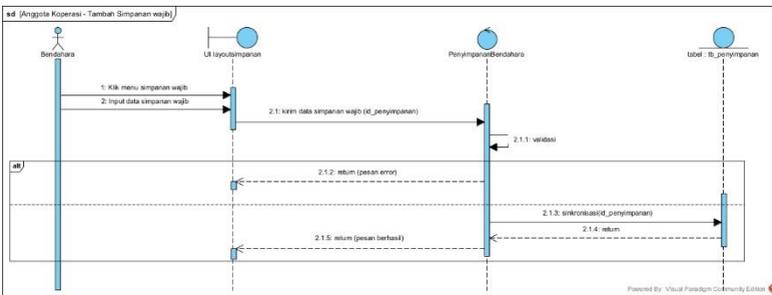
### 10. Sequence Diagram Mencetak Formulir Pendaftaran



Gambar 3. 18 Sequence Diagram Mencetak Formulir Pendaftaran

Gambar 3.18 menjelaskan mengenai *sequence diagram* cetak formulir yang dilakukan oleh aktor anggota koperasi. Aktor memilih menu Laporan pada UI kemudian klik tombol *export: pdf*. Kemudian *controller* LaporanPemeriksa melakukan proses cetak *formular* pendaftaran yang diambil dari tabel data *tb\_pengguna*, *tb\_permohonan*, *tb\_atasan*. Lalu mengirimkan kembali data yang telah ditemukan ke *controller* PermohonanCalon dan diteruskan ke UI dan mencetak data formulir.

### 11. Sequence Diagram Menambah Data Simpanan Wajib

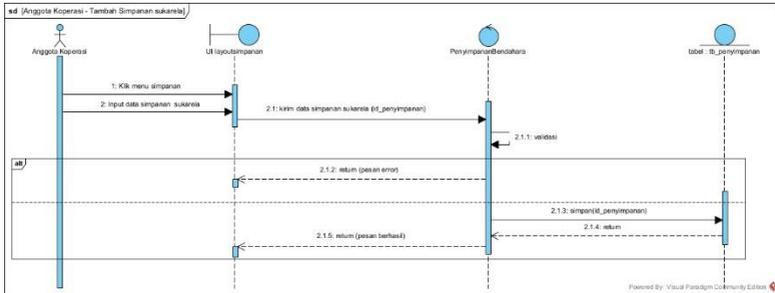


Gambar 3. 19 Sequence Diagram Menambah Data Simpanan Wajib

Gambar 3.19 menjelaskan mengenai tambah simpanan wajib yang dilakukan oleh bendahara pada antamuka pengguna (UI), aktor berada di UI layoutsimpanan kemudian memasukkan data yang nantinya

akan dikirimkan ke *controller* PenyimpananBendahara. Kemudian data yang dimasukkan akan dilakukan *validation*. Jika tidak berhasil, maka akan muncul pesan gagal. Jika tervalidasi maka muncul pesan berhasil, lalu data tersimpan pada tabel `tb_penyimpanan`.

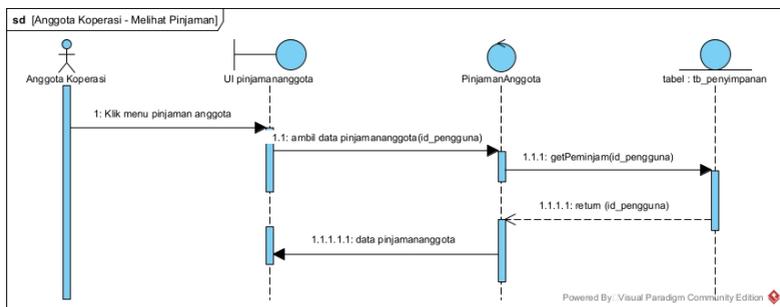
## 12. Sequence Diagram Menambah Data Simpanan Sukarela



**Gambar 3. 20** Sequence Diagram Menambah Data Simpanan Sukarela

**Gambar 3.20** menjelaskan mengenai tambah simpanan sukarela yang dilakukan oleh aktor anggota koperasi (anggota, bendahara, atasan, pengawas dan ketua) pada antamuka pengguna (UI), aktor berada di UI layoutsimpanan kemudian memasukkan data yang nantinya akan dikirimkan ke *controller* PenyimpananBendahara. Kemudian data yang dimasukkan akan dilakukan *validation*. Jika tidak berhasil, maka akan muncul pesan gagal. Jika tervalidasi maka muncul pesan berhasil, lalu data tersimpan pada tabel `tb_penyimpanan`.

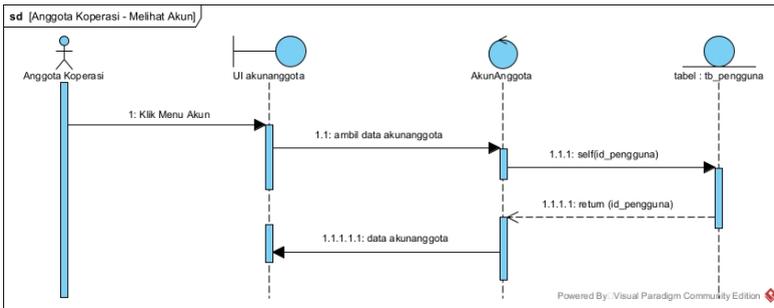
## 13. Sequence Diagram Menampilkan Data Pinjaman



**Gambar 3. 21** Sequence Diagram Menampilkan Data Pinjaman

**Gambar 3.21** menjelaskan mengenai *sequence diagram* menampilkan data pinjaman yang dilakukan oleh aktor anggota koperasi (anggota, bendahara, atasan, pengawas dan ketua). Aktor melakukan proses lihat data pinjaman yang diambil dari tabel data tb\_peminjaman. Kemudian *controller* PinjamanAnggota akan mengambil data tersebut dari tabel tb\_peminjaman. Lalu mengirimkan kembali data yang telah ditemukan ke *controller* PinjamanAnggota dan diteruskan ke UI.

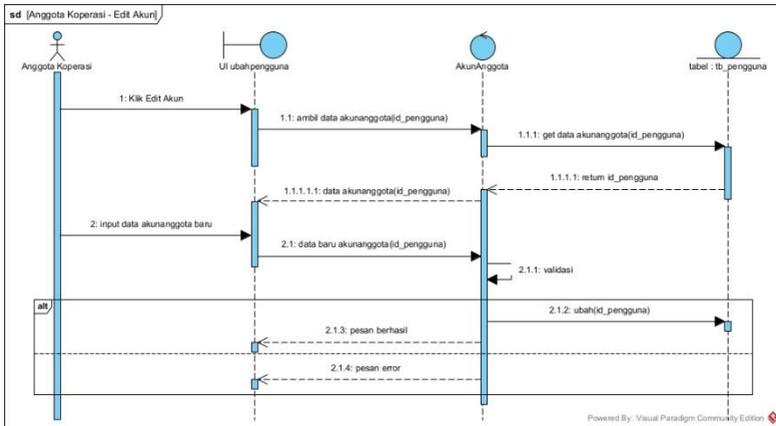
#### 14. *Sequence Diagram* Menampilkan Data Akun



**Gambar 3. 22** *Sequence Diagram* Menampilkan Data Akun

**Gambar 3.22** menjelaskan mengenai *sequence diagram* menampilkan data akun yang dilakukan oleh aktor anggota koperasi (anggota, bendahara, atasan, pengawas dan ketua). Aktor melakukan proses lihat data akun yang diambil dari tabel data tb\_pengguna. Kemudian *controller* AkunAnggota akan mengambil data tersebut dari tabel tb\_pengguna. Lalu mengirimkan kembali data yang telah ditemukan ke *controller* AkunAnggota dan diteruskan ke UI.

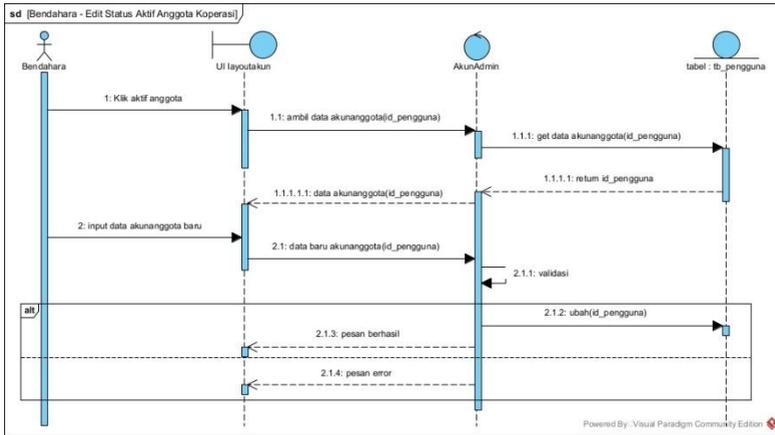
## 15. Sequence Diagram Mengubah Data Anggota Koperasi



**Gambar 3. 23** Sequence Diagram Mengubah Data Anggota Koperasi

**Gambar 3.23** menjelaskan mengenai edit Akun yang dilakukan oleh aktor anggota koperasi (anggota, bendahara, atasan, pengawas dan ketua). Aktor memilih akun pada UI, kemudian memilih salah satu data dan mengirimkan `id_pengguna` kepada `tb_pengguna` untuk mendapatkan data, berdasarkan data yang telah dikirimkan. Data hasil pencarian akan dikembalikan kepada *controller* AkunAnggota, lalu selanjutnya akan ditampilkan pada UI. Aktor memasukkan data baru, setelah itu data masuk kedalam *controller* AkunAnggota untuk dilakukan *validation*. Jika tervalidasi maka data yang diinputkan akan masuk ke tabel `tb_pengguna` dan muncul pesan berhasil. Jika terdapat kesalahan maka data tidak disimpan dan akan muncul pesan *error*.

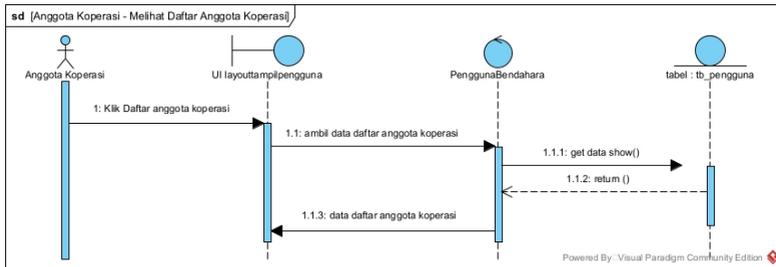
### 16. Sequence Diagram Menonaktifkan Status Anggota Koperasi



**Gambar 3. 24** Sequence Diagram Mengubah Status Anggota Koperasi

**Gambar 3.24** menjelaskan mengenai UI ubah status aktif anggota koperasi yang dilakukan oleh aktor Bendahara. Aktor memilih akun pada UI, kemudian memilih salah satu data dan mengirimkan id\_pengguna kepada *controller* AkunBendahara untuk mendapatkan data, berdasarkan data yang telah dikirimkan. Data hasil pencarian akan dikembalikan kepada *controller* AkunBendahara, lalu selanjutnya akan ditampilkan pada UI. Aktor memasukkan data baru, setelah itu data masuk kedalam *controller* AkunBendahara untuk dilakukan *validation*. Jika tervalidasi maka data yang diinputkan akan masuk ke tabel tb\_pengguna dan muncul pesan berhasil. Jika terdapat kesalahan maka data tidak disimpan dan akan muncul pesan *error*.

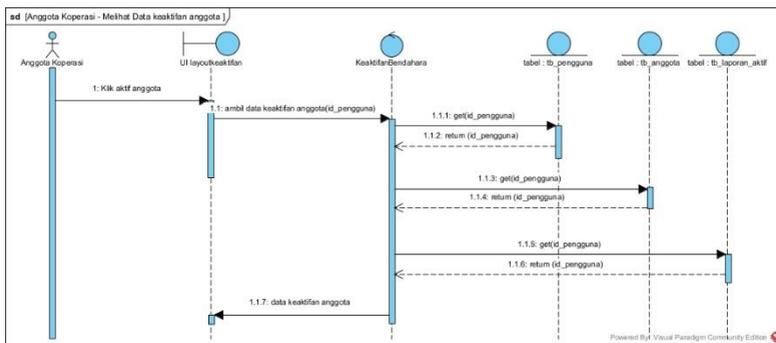
## 17. Sequence Diagram Menampilkan Data Anggota Koperasi



**Gambar 3. 25** Sequence Diagram Menampilkan Data Anggota Koperasi

**Gambar 3.25** menjelaskan *sequence diagram* untuk menampilkan data anggota koperasi yang dilakukan oleh aktor anggota koperasi (bendahara, pengawas dan ketua). Aktor melakukan tampil daftar anggota koperasi yang diambil dari tabel data `tb_pengguna`. Kemudian *controller* `PenggunaBendahara` akan mengambil data tersebut dari tabel `tb_pengguna`. Lalu mengirimkan kembali data yang telah ditemukan ke *controller* `PenggunaBendahara` dan diteruskan ke UI.

## 18. Sequence Diagram Menampilkan Data Keaktifan Anggota

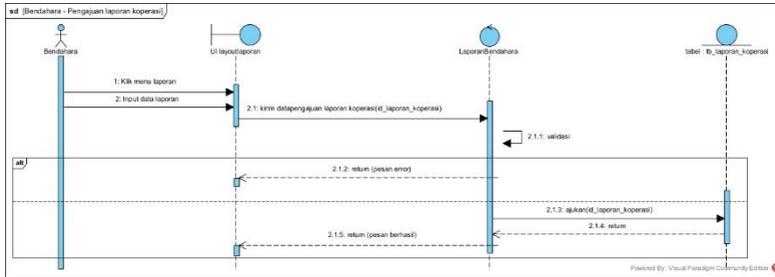


**Gambar 3. 26** Sequence Diagram Menampilkan Data Keaktifan Anggota

**Gambar 3.26** menjelaskan *sequence diagram* untuk menampilkan data keaktifan yang dilakukan oleh anggota koperasi (anggota, bendahara, atasan, pengawas, dan ketua). Proses tampil data

diambil dari `tb_pengguna`, `tb_anggota` dan `tb_laporan_aktif`. Kemudian *controller* `KeaktifanBendahara`, akan mengambil data tersebut dari tabel `tb_laporan_aktif`. Lalu mengirimkan kembali data yang telah ditemukan ke *controller* `KeaktifanBendahara` dan diteruskan ke UI.

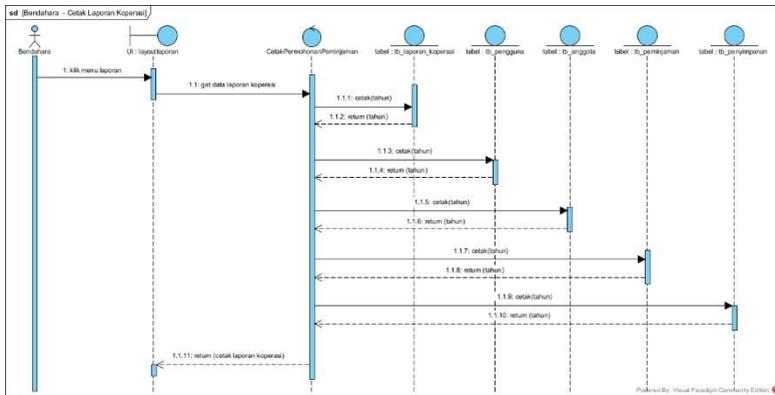
### 19. Sequence Diagram Mengajukan Laporan Koperasi



Gambar 3. 27 Sequence Diagram Mengajukan Laporan Koperasi

Gambar 3.27 menjelaskan mengenai tambah pengajuan laporan koperasi yang dilakukan oleh aktor bendahara pada antamuka pengguna (UI), aktor berada di UI layoutlaporan kemudian memasukkan data yang nantinya akan dikirimkan ke *controller* `LaporanBendahara`. Kemudian data yang dimasukkan akan dilakukan *validation*. Jika tidak berhasil, maka akan muncul pesan gagal. Jika tervalidasi maka muncul pesan berhasil, lalu data tersimpan pada tabel `tb_laporan_koperasi`.

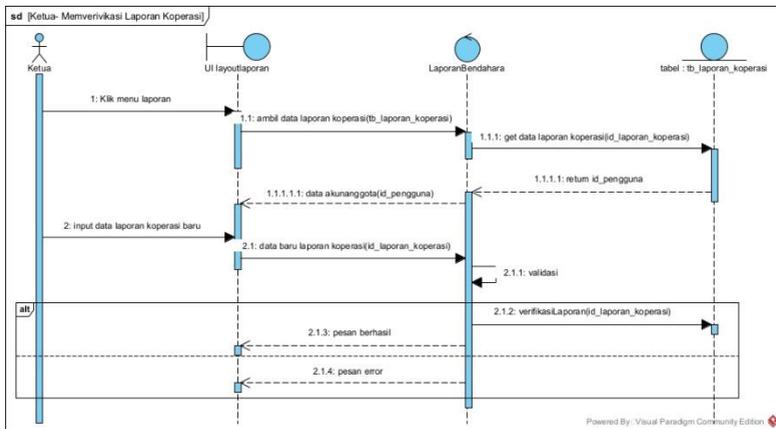
### 20. Sequence Diagram Mencetak Laporan Koperasi



### Gambar 3. 28 Sequence Diagram Mencetak Laporan Koperasi

Gambar 3.28 menjelaskan mengenai cetak laporan yang dilakukan oleh aktor bendahara. Aktor memilih menu Laporan pada UI kemudian klik tombol cetak. Kemudian *controller* LaporanBendahara melakukan proses lihat data laporan yang diambil dari tabel data *tb\_laporan\_koperasi*, *tb\_pengguna*, *tb\_anggota*, *tb\_peminjaman*, *tb\_penyimpanan*. Lalu mengirimkan kembali data yang telah ditemukan ke *controller* LaporanBendahara dan diteruskan ke UI dan mencetak data laporan.

### 21. Sequence Diagram Memverifikasi Laporan Koperasi

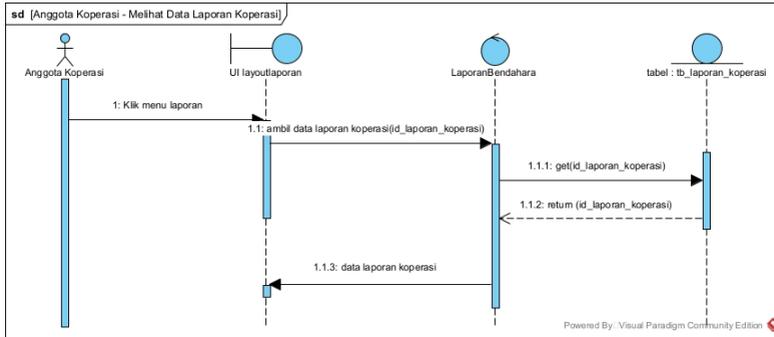


### Gambar 3. 29 Sequence Diagram Memverifikasi Laporan Koperasi

Gambar tersebut menjelaskan mengenai *Sequence Diagram* Verifikasi laporan koperasi yang dilakukan oleh aktor ketua. Aktor memilih menu laporan pada UI, kemudian memilih salah satu data dan mengirimkan *id\_laporan\_koperasi* kepada *controller* LaporanBendahara untuk mendapatkan data, berdasarkan data yang telah dikirimkan. Data hasil pencarian akan dikembalikan kepada *controller* LaporanBendahara, yang selanjutnya akan ditampilkan pada UI. Aktor memasukkan data baru, setelah itu data masuk kedalam *controller* LaporanBendahara untuk dilakukan *validation*. Jika tervalidasi maka data yang diinputkan akan masuk ke tabel *tb\_laporan\_koperasi* dan muncul pesan berhasil. Jika

terdapat kesalahan maka data tidak disimpan dan akan muncul pesan *error*.

## 22. *Sequence Diagram* Menampilkan Laporan Koperasi



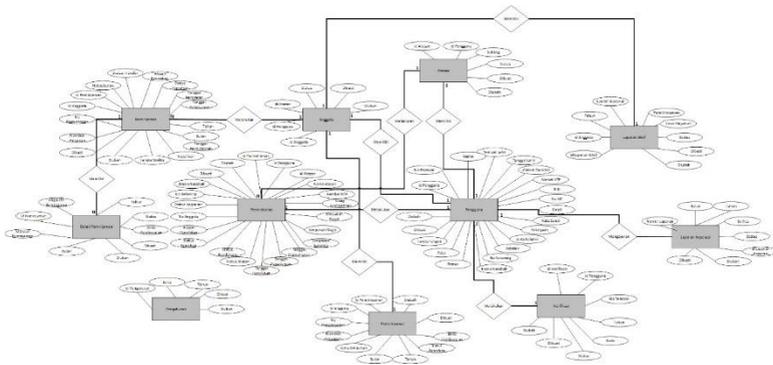
**Gambar 3. 30** *Sequence Diagram* Menampilkan Laporan Koperasi

**Gambar 3.29** menjelaskan *sequence diagram* untuk melihat data laporan koperasi yang dilakukan oleh aktor anggota koperasi (bendahara, pengawas dan ketua). Aktor melakukan proses lihat data laporan koperasi yang diambil dari tabel `tb_laporan_koperasi`. Kemudian *controller* LaporanBendahara akan mengambil data tersebut dari `tb_laporan_koperasi`. Lalu mengirimkan kembali data yang telah ditemukan ke *controller* LaporanBendahara dan diteruskan ke UI.

## E. *Class Diagram*

*Class diagram* digunakan untuk mendeskripsikan atau menggambarkan struktur dan deskripsi *class* serta hubungan antar *class*. *Class* terdiri dari nama *class*, atribut dan operasi atau metode. *Class diagram* dari Sistem Informasi Simpan Pinjam Berbasis *Website* adalah sebagai berikut:





**Gambar 3. 32** ERD Sistem Informasi Simpan Pinjam Berbasis *Website*

Pada ERD sistem di atas dijelaskan bahwa setiap *user* memiliki role untuk menentukan hak akses. Lalu Bendahara disini memiliki tanggung jawab yang cukup banyak dalam pengelolaan informasi.

**G. Struktur Tabel**

Struktur tabel adalah perancangan pada *database* yang digunakan dalam sistem yang akan dikembangkan. Berikut struktur tersebut :

**1. Tabel Anggota**

- Nama Tabel : *tb\_anggota*
- Fungsi : untuk mendapatkan *id\_anggota* dan *id\_atasan*
- Primary Key* : *id\_anggota*
- Foreign Key* : *id\_pengguna* dan *id\_atasan*

**Tabel 3. 4** Struktur Tabel Anggota

No	Nama Field	Tipe Data	Keterangan
1	<i>id_anggota</i>	<i>varchar(40)</i>	<i>Primary key</i>
2	<i>id_pengguna</i>	<i>varchar(40)</i>	<i>Foreign Key</i>
3	<i>id_atasan</i>	<i>varchar(40)</i>	
4	<i>status</i>	<i>enum(1)</i>	
5	<i>dibuat</i>	<i>varchar(25)</i>	
6	<i>diubah</i>	<i>varchar(25)</i>	

**2. Tabel Atasan**

- Nama Tabel : *tb\_atasan*
- Fungsi : menyimpan *id\_atasan* untuk menampilkan atasan dari *tb\_pengguna*

*Primary Key* : id\_atasan

*Foreign Key* : id\_pengguna

**Tabel 3. 5** Struktur Tabel Atasan

No	Nama Field	Tipe Data	Keterangan
1	id_atasan	varchar(40)	Primary key
2	id_pengguna	varchar(40)	Foreign Key
3	subbag	varchar(40)	
4	status	enum(1)	
5	dibuat	varchar(25)	
6	diubah	varchar(25)	

### 3. Tabel Detail Peminjaman

Nama Tabel : tb\_detail\_peminjaman

Fungsi : menyimpan data pinjaman yang dibayarkan perbulan

*Primary Key* : id\_detail\_peminjaman

*Foreign Key* : id\_peminjaman

**Tabel 3. 6** Struktur Tabel Detail Peminjaman

No	Nama Field	Tipe Data	Keterangan
1	id_detail_peminjaman	varchar(40)	Primary Key
2	id_peminjaman	varchar(40)	Foreign Key
3	angsuran peminjaman	varchar(25)	
4	bulan	varchar(10)	
5	tahun	varchar(10)	
6	status	enum(1)	
7	jenis_pembayaran	varchar(1)	
8	dibuat	varchar(25)	
9	diubah	varchar(25)	

### 4. Nama Tabel : tb\_laporan\_aktif

Fungsi : merekap jumlah data total simpanan dan pinjaman

*Primary Key* : id\_laporan\_aktif

*Foreign Key* : id\_anggota

**Tabel 3. 7** Struktur Tabel Laporan Aktif

No	Nama Field	Tipe data	Keterangan
1	id_laporan_aktif	varchar(40)	Primary Key
2	id_anggota	varchar(40)	Foreign Key
3	tahun	varchar(6)	
4	jumlah_nominal	varchar(25)	

5	total_simpanan	<i>varchar(50)</i>	
6	total_pinjaman	<i>varchar(50)</i>	
7	status	<i>enum(1)</i>	
8	dibuat	<i>varchar(25)</i>	
9	diubah	<i>varchar(25)</i>	

### 5. Tabel Laporan Koperasi

Nama Tabel : *tb\_laporan\_koperasi*

Fungsi : untuk menampung data laporan akhir koperasi

*Primary Key* : *id\_laporan\_koperasi*

*Foreign Key* : -

**Tabel 3. 8** Struktur Tabel Laporan Koperasi

No	Nama Field	Tipe data	Keterangan
1	<i>id_laporan_koperasi</i>	<i>varchar(40)</i>	<i>Primary Key</i>
2	<i>nomor_laporan</i>	<i>varchar(10)</i>	
3	<i>nomor_laporan</i>	<i>varchar(100)</i>	
4	<i>bulan</i>	<i>varchar(5)</i>	
5	<i>tahun</i>	<i>varchar(7)</i>	
6	<i>berkas</i>	<i>varchar(40)</i>	
7	<i>status</i>	<i>enum(1)</i>	
8	<i>dibuat</i>	<i>varchar(25)</i>	
9	<i>diubah</i>	<i>varchar(25)</i>	

### 6. Tabel Peminjaman

Nama Tabel : *tb\_peminjaman*

Fungsi : menyimpan data peminjaman dana

*Primary Key* : *id\_peminjaman*

*Foreign Key* : *id\_anggota*

**Tabel 3. 9** Struktur Tabel Peminjaman

No	Nama Field	Tipe Data	Keterangan
1	<i>id_peminjaman</i>	<i>varchar(40)</i>	<i>Primary Key</i>
2	<i>id_anggota</i>	<i>varchar(40)</i>	<i>Foreign Key</i>
3	<i>no_permohonan</i>	<i>varchar(25)</i>	
4	<i>nominal_pinjaman</i>	<i>varchar(25)</i>	
5	<i>jangka_waktu</i>	<i>varchar(10)</i>	
6	<i>keperluan</i>	<i>text</i>	
7	<i>tanggal_permohonan</i>	<i>varchar(25)</i>	
8	<i>bulan</i>	<i>varchar(10)</i>	

9	tahun	<i>varchar(10)</i>	
10	tanggal_persetujuan	<i>varchar(25)</i>	
11	tanggal_penolakan	<i>varchar(25)</i>	
12	status_pemohon	<i>enum(1)</i>	
13	alasan_penolakan	<i>text</i>	
14	status_transfer	<i>enum(1)</i>	
15	status_lunas	<i>enum(1)</i>	
16	dibuat	<i>varchar(25)</i>	
17	diubah	<i>varchar(25)</i>	

### 7. Tabel Pengaturan

Nama Tabel : *tb\_pengaturan*

Fungsi : menyimpan ketentuan data simpanan wajib koperasi

Primary Key : *id\_pengaturan*

Foreign Key : -

**Tabel 3. 10** Struktur Tabel Pengaturan

No	Nama Field	Tipe data	Keterangan
1	<i>id_pengaturan</i>	<i>varchar(40)</i>	<i>Primary Key</i>
2	<i>jenis</i>	<i>varchar(20)</i>	
3	<i>tahun</i>	<i>varchar(25)</i>	
4	<i>dibuat</i>	<i>varchar(25)</i>	
5	<i>diubah</i>	<i>varchar(25)</i>	

### 8. Tabel Pengguna

Nama Tabel : *tb\_pengguna*

Fungsi : menyimpan informasi biodata milik pengguna

Primary Key : *id\_pengguna*

Foreign Key : -

**Tabel 3. 11** Struktur Tabel Pengguna

No	Nama Field	Tipe Data	Keterangan
1	<i>id_pengguna</i>	<i>varchar(40)</i>	<i>Primary Key</i>
2	<i>no_pegawai</i>	<i>varchar(25)</i>	
3	<i>nama</i>	<i>varchar(40)</i>	
4	<i>tempat_lahir</i>	<i>varchar(50)</i>	
5	<i>tanggal_lahir</i>	<i>varchar(25)</i>	
6	<i>alamat_domisili</i>	<i>text</i>	
7	<i>alamat_ktp</i>	<i>text</i>	
8	<i>nik</i>	<i>varchar(16)</i>	

9	no_hp	<i>varchar(15)</i>	
10	email	<i>varchar(100)</i>	
11	kata_sandi	<i>varchar(200)</i>	
12	pekerjaan	<i>varchar(35)</i>	
13	jenis_kelamin	<i>enum(1)</i>	
14	jabatan	<i>varchar(35)</i>	
15	no_rekening	<i>varchar(20)</i>	
16	nama_nasabah	<i>varchar(50)</i>	
17	status	<i>enum(1)</i>	
18	foto	<i>varchar(40)</i>	
19	tanda_tangan	<i>varchar(40)</i>	
20	dibuat	<i>varchar(25)</i>	
21	diubah	<i>varchar(25)</i>	

### 9. Tabel Penyimpanan

Nama Tabel : tb\_penyimpanan

Fungsi : menyimpan semua data simpanan

Primary Key : id\_penyimpanan

Foreign Key : id\_anggota

**Tabel 3. 12** Struktur Tabel Penyimpanan

No	Nama Field	Tipe Data	Keterangan
1	id_penyimpanan	<i>varchar(40)</i>	<i>Primary Key</i>
2	id_anggota	<i>varchar(40)</i>	<i>Foreign Key</i>
3	no_penyimpanan	<i>varchar(25)</i>	
4	nominal_simpanan	<i>varchar(25)</i>	
5	jenis_simpanan	<i>varchar(10)</i>	
6	tanggal_simpanan	<i>varchar(25)</i>	
7	bulan	<i>varchar(2)</i>	
8	tahun	<i>varchar(5)</i>	
9	status_pemohon	<i>enum(1)</i>	
10	jenis_pembayaran	<i>varchar(1)</i>	
11	dibuat	<i>varchar(25)</i>	
12	diubah	<i>varchar(25)</i>	

### 10. Tabel Permohonan

Nama Tabel : tb\_permohonan

Fungsi : menyimpan data permohonan anggota

Primary Key : id\_permohonan

Foreign Key : id\_pengguna

**Tabel 3. 13** Struktur Tabel Permohonan

No.	Nama Field	Tipe Data	Keterangan
1	id_permohonan	<i>varchar(40)</i>	<i>Primary Key</i>
2	id_pengguna	<i>varchar(40)</i>	<i>Foreign Key</i>
3	id_atasan	<i>varchar(40)</i>	
4	nama_atasan	<i>varchar(100)</i>	
5	gambar_ktp	<i>varchar(40)</i>	
6	uang_administrasi	<i>varchar(20)</i>	
7	simpanan_pokok	<i>varchar(20)</i>	
8	simpanan_wajib	<i>varchar(20)</i>	
9	simpanan_sukarela	<i>varchar(20)</i>	
10	tanggal_permohonan	<i>varchar(25)</i>	
11	tanggal_persetujuan	<i>varchar(25)</i>	
12	tanggal_penolakan	<i>varchar(25)</i>	
13	status_atasan	<i>enum(1)</i>	
14	status_bendahara	<i>enum(1)</i>	
15	status_pemohon	<i>enum(1)</i>	
16	alasan_penolakan	<i>text</i>	
17	no_anggota	<i>varchar(50)</i>	
18	status_angsuran	<i>enum(1)</i>	
19	no_rekening	<i>varchar(20)</i>	
20	nama_nasabah	<i>varchar(50)</i>	
21	dibuat	<i>varchar(25)</i>	
22	diubah	<i>varchar(25)</i>	

### 11. Tabel Verifikasi

Nama Tabel : tb\_verifikasi

Fungsi : memverifikasi no *whatsapp* dan menyimpan kode

Primary Key : id\_verifikasi

Foreign Key : id\_pengguna

**Tabel 3. 14** Struktur Tabel Verifikasi

No	Nama Field	Tipe data	Keterangan
1	id_verifikasi	<i>varchar(40)</i>	<i>Primary Key</i>
2	id_pengguna	<i>varchar(40)</i>	<i>Foreign Key</i>
3	no_telepon	<i>varchar(20)</i>	
4	token	<i>text</i>	
5	kode	<i>varchar(15)</i>	

6	status	<i>enum(1)</i>	
7	dibuat	<i>varchar(25)</i>	
8	diubah	<i>varchar(25)</i>	

## H. Rancangan Antarmuka

Perancangan antarmuka pengguna merupakan komponen krusial dalam sistem karena tampilan antarmuka (*interface*) adalah hal pertama yang dilihat saat sistem berjalan. Tahap ini akan menjelaskan proses perancangan tampilan antarmuka (*interface*) dari Sistem Informasi Simpan Pinjam Berbasis Website sebagai berikut:

### 1. Rancangan Antarmuka Proses *Login* Pengguna



**Gambar 3. 33** Perancangan Antarmuka Halaman *Login* Pengguna

**Gambar 3.33** Terdapat beberapa *label* serta *button whatsapp* untuk *login* menggunakan nomor *whatsapp*, kemudian *text input* untuk *input* data nomor pegawai dan kata sandi, serta *button* masuk untuk menuju ke halaman beranda. Selain itu terdapat *link* "Lupa Kata Sandi" yang membantu pengguna untuk mengakses halaman beranda jika pengguna lupa kata sandi. Terdapat *checkbox* "Ingat Kata Sandi" yang berfungsi untuk memudahkan *login* pengguna tanpa menginputkan ulang nomor pegawai dan kata sandi. Terdapat *link* untuk membuat akun dan mendaftar menjadi anggota koperasi. Rancangan antarmuka saat *button whatsapp* ditekan dapat dilihat pada gambar berikut.

https://koperasi.pnc.ac.id

## Masuk Dengan Whatsapp

No Telepon Whatsapp

087740737154

Verifikasi

Login dengan kata sandi?

**Gambar 3.34** Rancangan Antarmuka *Login Menggunakan Whatsapp*

**Gambar 3.34** Terdapat *label* dengan *text input* untuk memasukan nomor telepon *whatsapp*, *button* verifikasi untuk memverifikasi data yang sudah diinputkan serta *button back* jika pengguna ingin *login* menggunakan nomor pegawai dan kata sandi. Rancangan antarmuka jika *button* verifikasi ditekan dapat dilihat pada gambar 4.7.

https://koperasi.pnc.ac.id

## VERIFIKASI KODE

Kode verifikasi telah dikirim ke nomor telepon aplikasi whatsapp anda, silahkan cek pesan dari sistem.

Kode OTP

Verifikasi

Butuh bantuan? [Hubungi kami!](#)

**Gambar 3. 35** Rancangan Antarmuka Halaman Verifikasi Kode OTP

**Gambar 3.35** Terdapat *textinput* untuk memasukkan kode *one time password* yang diterima, kemudian *button* verifikasi untuk masuk ke

halaman beranda. Selain itu Terdapat link untuk memudahkan pengguna jika ada pertanyaan terkait verifikasi kode OTP.



**Gambar 3. 36** Rancangan Antarmuka Halaman Lupa Kata Sandi

**Gambar 3.36** Terdapat *text*, *label* dengan *textinput* untuk memasukkan nomor telepon *whatsapp* serta *button* verifikasi yang mengarah pada halaman verifikasi kode OTP untuk memasukan kode dan menekan *button* verifikasi yang dapat dilihat pada gambar 4.7 untuk masuk ke halaman beranda.

## 2. Rancangan Antarmuka Proses Pendaftaran Anggota



**Gambar 3. 37** Rancangan Antarmuka Pendaftaran Anggota Koperasi

**Gambar 3.37** Terdapat *text* dengan *text input* untuk memasukan data nama dan nomor telepon *whatsapp*, kemudian *button* verifikasi untuk masuk ke halaman verifikasi kode OTP untuk memasukan kode dan menekan *button* verifikasi untuk masuk ke halaman formulir permohonan menjadi anggota koperasi.

The image shows a web browser window with the URL <http://koperasi.pnc.ac.id>. The page content is as follows:

- Page Title: Formulir Pendaftaran Menjadi Anggota
- Main Title: Formulir Permohonan Menjadi Anggota Koperasi
- Subtitle: Koperasi Serba Usaha Mandiri Sejahtera Politeknik Negeri Cilacap
- Form Fields:
  - Profile picture icon
  - Nama Calon
  - Langkah 1: Yang bertanda tangan dibawah ini,
    - No Pegawai
    - Nama Lengkap
    - Tempat Lahir

**Gambar 3. 38** Rancangan Antarmuka Formulir Permohonan Anggota

**Gambar 3.38** Terdapat *text*, dan beberapa *label text input* untuk memasukan biodata diri seperti nomor pegawai, nama lengkap, jenis kelamin, tempat dan tanggal lahir, alamat KTP, domisili, nomor telepon, nomor KTP, pekerjaan/jabatan serta foto KTP asli, kemudian terdapat label penyetoran pertama anggota baru yang terdiri atas uang administrasi, simpanan pokok, simpanan wajib dan simpanan sukarela serta informasi nomor rekening BNI pegawai. Terdapat *button* ajukan untuk mengirimkan permohonan dan mengarah pada status permohonan pada gambar berikut.

**Gambar 3. 39** Rancangan Antarmuka Status Permohonan Anggota

**Gambar 3.39** Terdapat *icon*, *text area* serta *label* maupun tanggal pengiriman permohonan. Halaman status ini berisi informasi status validasi yang dilakukan oleh atasan langsung dan bendahara. Jika permohonan tersebut disetujui, maka anggota akan diarahkan ke halaman beranda.

### 3. Rancangan Antarmuka Proses Validasi Permohonan Menjadi Anggota Koperasi

**Gambar 3. 40** Rancangan Antarmuka Validasi Permohonan Anggota

**Gambar 3.40** terdapat data permohonan yang ditampilkan pada *form* validasi. Selain itu terdapat *button* tunda, tolak dan setuju yang

dipergunakan oleh atasan dan bendahara untuk memvalidasi permohonan menjadi anggota koperasi.

#### 4. Rancangan Antarmuka Proses Permohonan Peminjaman Dana

The screenshot shows a web browser window with the URL `https://koperasi.pnc.ac.id`. The page header includes a hamburger menu icon, the text "KOPERASI-MANTRA", and a "profil" button. The main content area contains a form with the following elements:

- Nominal Pinjaman:** A text input field.
- Jangka (Bulan):** A text input field.
- Untuk Keperluan:** A large text area for describing the purpose of the loan.
- Nomor Rekening:** A text input field containing the value "072849719625".
- Rp. 6.000.000:** A text input field showing the loan amount.
- Ajukan Sekarang:** A blue button to submit the application.

**Gambar 3. 41** Rancangan Antarmuka Halaman Pengajuan Peminjaman

**Gambar 3.41** Terdapat beberapa *label* dengan *text input* yang harus dilengkapi oleh anggota seperti nominal, jangka waktu, serta alasan peminjaman. Kemudian terdapat *button* ajukan untuk mengirimkan permohonan tersebut. Setelah itu anggota akan dialihkan ke halaman status peminjaman dana yang dapat ditampilkan pada gambar 4.21.

The screenshot shows the same web browser window, but the page content has changed to display the loan status. The header and navigation elements remain the same. The main content area includes:

- Nominal Pinjaman:** A text input field.
- Jangka (Bulan):** A text input field.
- Untuk Keperluan:** A large text area.
- Nomor Rekening:** A text input field containing "072849719625".
- Status masih draft:** A black button indicating the current status.
- Ajukan ke atasan:** A blue button to submit the application to the supervisor.
- STATUS PENGAJUAN:** A section header for the application status.
- Sudah diverifikasi oleh Atasan:** A black button indicating the application has been verified.
- Selamat! Pengajuan anda diterima:** A blue button indicating the application has been accepted.
- Kirim Notifikasi Ke Bendahara:** A grey button to notify the treasurer.
- Payment Schedule Table:**

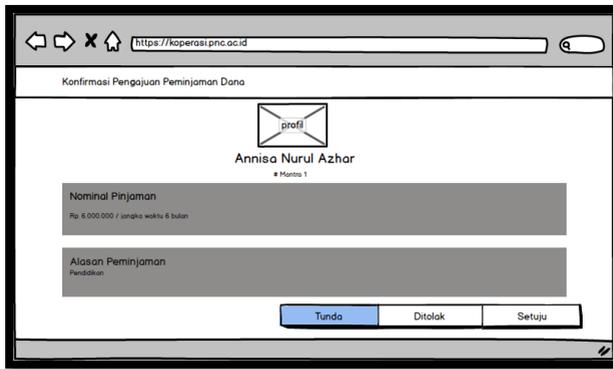
Bulan ke 1	Rp. 1.100.000
Bulan ke 2	Rp. 1.100.000
Bulan ke 3	Rp. 1.100.000

**Gambar 3. 42** Rancangan Antarmuka Status Peminjaman Dana.

**Gambar 3.42** Terdapat *icon*, *text*, *text area* serta *label* maupun tanggal pengiriman permohonan. Halaman status ini berisi informasi

status validasi yang dilakukan oleh ketua koperasi. Jika permohonan disetujui, maka bendahara akan memproses pencairan dana kepada anggota tersebut.

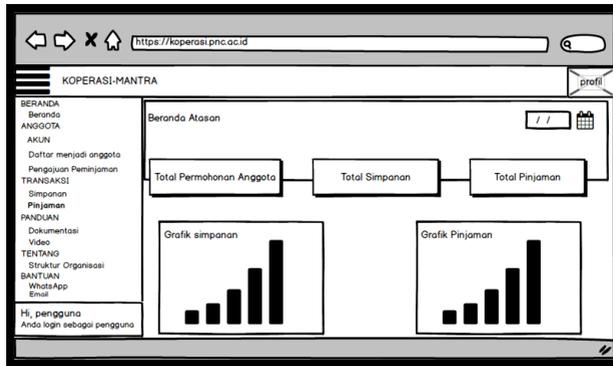
## 5. Rancangan Antarmuka Proses Validasi Peminjaman Dana



**Gambar 3. 43** Rancangan Antarmuka Validasi Oleh Ketua Koperasi

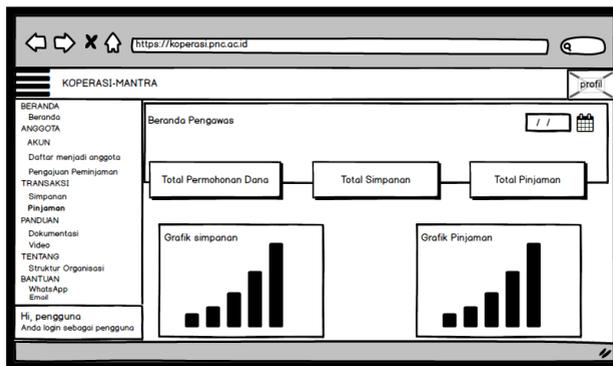
**Gambar 3.43** terdapat data permohonan yang ditampilkan pada *form* validasi. Selain itu terdapat *button* tunda, tolak dan setuju yang dipergunakan oleh ketua untuk memvalidasi permohonan menjadi anggota koperasi.

## 6. Rancangan Antarmuka Halaman Menu Beranda Pengguna



**Gambar 3. 44** Rancangan Antarmuka Halaman Beranda Atas

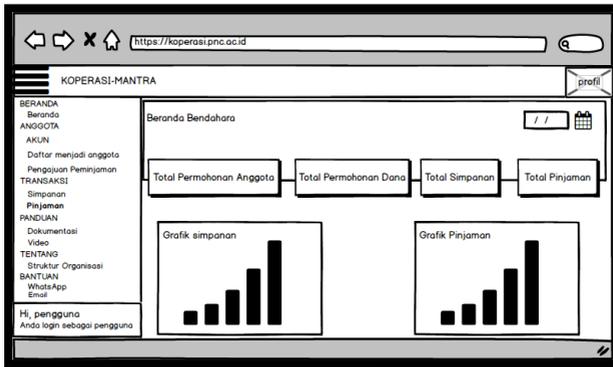
**Gambar 3.44** Terdapat statistik data jumlah permohonan anggota, jumlah simpanan, pinjaman. *Sidebar* menu memuat beranda, data akun, transaksi pribadi dan keseluruhan, verifikasi permohonan, data seluruh anggota, struktur organisasi, dan bantuan *whatsapp*. Selain itu terdapat grafik simpanan dan pinjaman serta informasi pengguna yang sedang *login* dan profil pengguna yang terdapat tombol *logout* ketika ditekan.



**Gambar 3. 45** Rancangan Antarmuka Halaman Menu Beranda Pengawas dan Ketua Koperasi

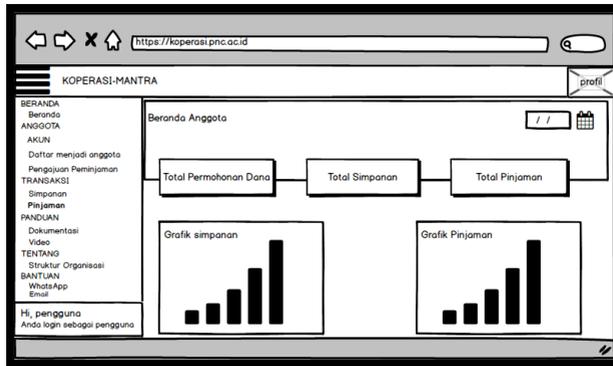
**Gambar 3.45** Terdapat statistik data jumlah permohonan dana, jumlah simpanan, pinjaman. *Sidebar* menu memuat beranda, data akun,

transaksi pribadi dan keseluruhan, verifikasi permohonan, data seluruh anggota, struktur organisasi, dan bantuan *whatsapp*. Selain itu terdapat grafik simpanan dan pinjaman serta informasi pengguna yang sedang *login* dan profil pengguna yang terdapat tombol *logout* ketika ditekan.



**Gambar 3. 46** Rancangan Antarmuka Halaman Beranda Bendahara

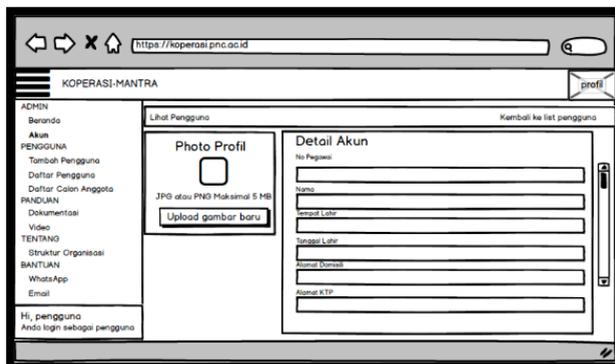
**Gambar 3.46** Terdapat statistik data jumlah permohonan anggota maupun dana, jumlah simpanan, pinjaman. *Sidebar* menu memuat beranda, data akun, transaksi pribadi dan keseluruhan, verifikasi permohonan, data seluruh anggota, struktur organisasi, dan bantuan *whatsapp*. Selain itu terdapat grafik simpanan dan pinjaman serta informasi pengguna yang sedang *login* dan profil pengguna yang terdapat tombol *logout* ketika ditekan.



**Gambar 3. 47** Rancangan Antarmuka Halaman Menu Beranda Anggota Koperasi

**Gambar 3.47** Terdapat statistik data total permohonan dana yang sudah diajukan, jumlah simpanan, pinjaman. *Sidebar* menu memuat beranda, data akun, transaksi pribadi, data keaktifan, struktur organisasi, dan bantuan *whatsapp*. Selain itu terdapat grafik simpanan dan pinjaman serta informasi pengguna yang sedang *login* dan profil pengguna yang terdapat tombol *logout* ketika ditekan.

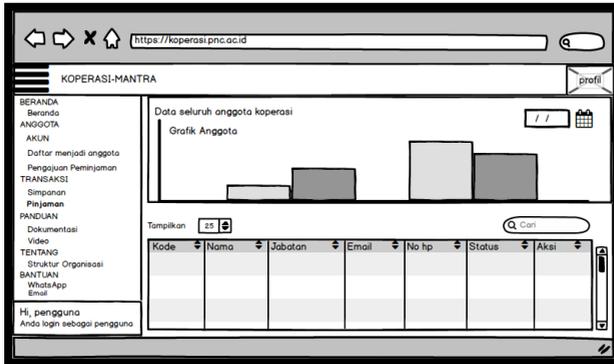
## 7. Rancangan Antarmuka Halaman Menu Akun Pengguna



**Gambar 3. 48** Rancangan Antarmuka Halaman Menu Akun Pengguna

**Gambar 3.48** Terdapat *image* untuk *photo* profil, *text*, *text area* serta *label* dengan *text input* no pegawai, nama, tempat lahir, tanggal lahir, alamat domisili, alamat KTP, NIK, no hp, email, pekerjaan, jenis kelamin, dan jabatan. Selain untuk melihat detail akun, menu ini dapat digunakan untuk memperbarui data diri yang disimpan menggunakan *button* simpan pembaharuan.

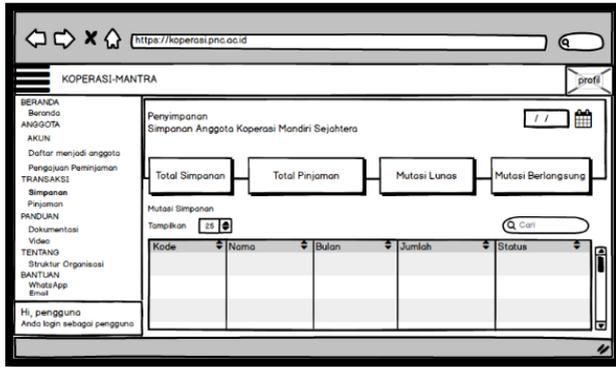
## 8. Rancangan Antarmuka Halaman Menu Daftar Pengguna



**Gambar 3. 49** Rancangan Antarmuka Halaman Menu daftar pengguna

**Gambar 3.49** Terdapat grafik, *text*, *label*, *num.stepper* untuk menampilkan jumlah data yang diinginkan, *search box* untuk menampilkan data yang diinginkan, *scroll bar* untuk menggulir semua data, *data grid* untuk informasi biodata diri, serta *link bar* untuk memilih kembali dan selanjutnya. Di menu ini bendahara juga bisa untuk mengedit data pengguna.

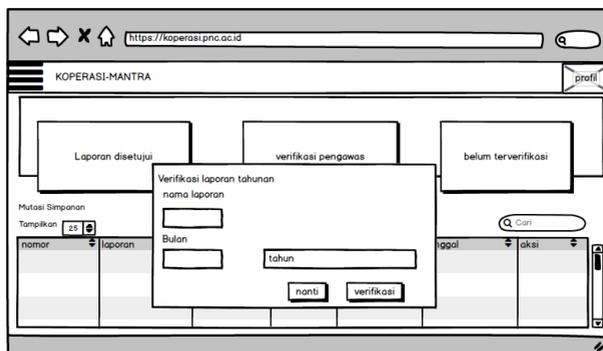
## 9. Rancangan Antarmuka Transaksi Simpan Pinjam



**Gambar 3. 50** Rancangan Antarmuka Transaksi Simpan Pinjam Pengguna

**Gambar 3.50** Menampilkan *button* total simpanan, total pinjaman, mutasi lunas, dan mutasi berlangsung. Pada halaman ini pengguna bisa melihat semua jenis transaksi simpan pinjam secara satu persatu dengan menekan *button* transaksi masing-masing untuk menampilkan detail data yang diinginkan. Selain itu terdapat tombol cari dan tampilkan data untuk mempermudah pengguna dalam mencari data dengan cepat.

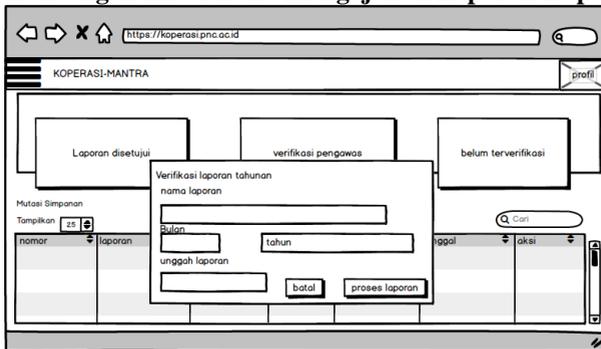
## 10. Rancangan Antarmuka Verifikasi Laporan Koperasi



**Gambar 3. 51** Rancangan Antarmuka Verifikasi Laporan Koperasi

**Gambar 3.51** Menampilkan *button* verifikasi, laporan disetujui verifikasi pengawas, belum terverifikasi. Terdapat *label* nama laporan, bulan dan tahun laporan koperasi. Kemudian terdapat *button* verifikasi untuk menyetujui laporan koperasi.

## 11. Rancangan Antarmuka Mengajukan Laporan Koperasi



**Gambar 3. 52** Rancangan Antarmuka Mengajukan Laporan Koperasi

**Gambar 3.52** Menampilkan *button* proses laporan, laporan disetujui verifikasi pengawas, belum terverifikasi. Terdapat *label* nama laporan, bulan, unggah laporan dan tahun laporan koperasi. Kemudian terdapat *button* proses laporan untuk mengajukan laporan koperasi.

## 12. Rancangan Antarmuka Keaktifan Anggota

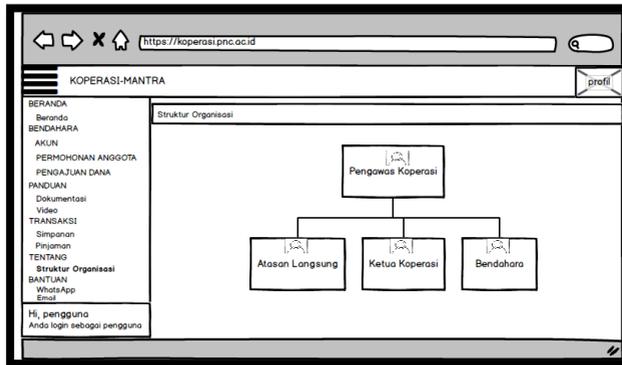
Anggota Aktif				Anggota Pasif			
No <sup>h</sup>	Nama	Total Simpanan	Total Pinjaman	No <sup>h</sup>	Nama	Total Simpanan	Total Pinjaman
1	Annisa	Rp. 1500.000	Rp. 6.000.000	1	Nisa	Rp. 600.000	Rp. 0
2	Nurul	Rp. 1000.000	Rp. 5.000.000	2	Nuri	Rp. 600.000	Rp. 0
3	Azhar	Rp. 800.000	Rp. 4.000.000	3	Azka	Rp. 600.000	Rp. 0

**Gambar 3. 53** Rancangan Antarmuka Halaman Keaktifan Anggota

**Gambar 3.53** Menampilkan *label* dan juga *data grid* mengenai data anggota aktif dan pasif yang dilihat berdasarkan aktivitas atau total

transaksi simpanan dan pinjaman yang dilakukan oleh anggota koperasi. Semakin banyak transaksi simpanan dan pinjaman yang dilakukan, maka semakin aktif status keanggotaan koperasi tersebut yang akan memberikan keuntungan tersendiri bagi anggota koperasi.

### 13. Rancangan Antarmuka Menu Pengurus Koperasi



**Gambar 3. 54** Rancangan Antarmuka Menu Pengurus Koperasi

**Gambar 3.54** Menampilkan nama, jabatan dan photo profil pengurus koperasi pada saat ini, seperti pengawas, ketua, atasan dan juga bendahara yang terdapat label dan photo profil masing-masing pengurus.

### 3.3 Metode Pengujian Sistem

#### 3.3.1 Skenario Pengujian

Pengujian dengan metode *blackbox* yang memastikan sebuah bentuk keluaran sebagai syarat lulus pengujian. Berikut ini adalah skenario pengujian sistem informasi yang akan dikembangkan dan digunakan sebagai pengujian :

#### 1. Skenario Pengujian Proses Pendaftaran Anggota Koperasi

**Tabel 3. 15** Skenario Pengujian Proses Pendaftaran Anggota Koperasi

Identifikasi	
Nama Proses	Pendaftaran Menjadi Anggota Koperasi

Tujuan	Memasukan data_pengguna pada tb_pengguna
Aktor	Pegawai
<b>Deskripsi</b>	
Skenario pengujian proses pendaftaran menjadi anggota koperasi bertujuan untuk mendaftarkan pegawai sebagai anggota koperasi dan menampilkan <i>form</i> untuk mengisi formulir pendaftaran anggota	
<b>Kondisi Awal</b>	
data_pengguna belum tersimpan pada tb_pengguna	
<b>Skenario Utama</b>	
Aksi Aktor	Reaksi Sistem
1. Aktor memasukan data_pengguna	2. Sistem menjalankan fungsi no_hp untuk menyimpan data_pengguna ke dalam tb_pengguna
<b>Kondisi Akhir</b>	
Berhasil	Gagal
Data_ pengguna berhasil ditambahkan ke tb_pengguna	Sistem menampilkan pesan kesalahan

## 2. Skenario Pengujian Proses *Login* Nomor Pegawai

**Tabel 3. 16** Skenario Pengujian Proses *Login* Nomor Pegawai

<b>Identifikasi</b>	
Nama Proses	<i>Login</i> dengan nomor pegawai

Tujuan	Melakukan validasi data_pengguna
Aktor	Anggota Koperasi
<b>Deskripsi</b>	
Skenario pengujian proses <i>login</i> dengan nomor pegawai dan kata sandi bertujuan untuk anggota koperasi dapat menampilkan halaman beranda dan melakukan transaksi simpan pinjam	
<b>Kondisi Awal</b>	
data_pengguna tersimpan pada tb_pengguna	
<b>Skenario Utama</b>	
Aksi Aktor	Reaksi Sistem
1. Aktor memasukan data_pengguna	2. Sistem menjalankan fungsi <i>auth</i> untuk mengambil data_pengguna pada tb_pengguna berdasarkan no pegawai dan kata sandi 3. Sistem melakukan validasi data_pengguna
<b>Kondisi Akhir</b>	
Berhasil	Gagal
Sistem menampilkan halaman beranda	1. Sistem menampilkan pesan kesalahan 2. Sistem tetap berada di halaman <i>login</i> dan menghapus data yang diinputkan

**Tabel 3. 17** Skenario Pengujian Proses *Login Whatsapp*

Identifikasi	
Nama Proses	<i>Login dengan whatsapp</i>
Tujuan	Melakukan validasi data no_hp
Aktor	Anggota Koperasi
Deskripsi	
Skenario pengujian proses <i>login</i> dengan nomor <i>whatsapp</i> bertujuan untuk anggota koperasi dapat menampilkan halaman beranda dan melakukan transaksi simpan pinjam	
Kondisi Awal	
Data no_hp tersimpan pada tb_pengguna	
Skenario Utama	
Aksi Aktor	Reaksi Sistem
1. Aktor memasukan data_pengguna	2. Sistem menjalankan fungsi nohp untuk mengambil data no_hp pada tb_pengguna berdasarkan no_hp 3. Sistem melakukan validasi data no_hp
Kondisi Akhir	
Berhasil	Gagal
Sistem menampilkan halaman beranda	Sistem menampilkan pesan kesalahan

### 3. Skenario Pengujian Proses Mengajukan Formulir Anggota

**Tabel 3. 18** Skenario Pengujian Proses Mengajukan Formulir Anggota

Identifikasi	
Nama Proses	Mengajukan Formulir Menjadi Anggota
Tujuan	Memasukan data_permohonan pada tb_pengguna dan tb_permohonan
Aktor	Anggota Koperasi
Deskripsi	
Skenario pengujian proses mengajukan formulir menjadi anggota bertujuan untuk anggota koperasi dapat melakukan pengajuan menjadi anggota koperasi dengan memasukkan data_permohonan	
Kondisi Awal	
data_permohonan belum tersimpan pada tb_pengguna dan tb_permohonan	
Skenario Utama	
Aksi Aktor	Reaksi Sistem
1. Aktor memasukan data_permohonan	2. Sistem menjalankan fungsi permohonanSave untuk menyimpan data permohonan pada tb_pengguna dan tb_permohonan
Kondisi Akhir	
Berhasil	Gagal
Sistem menampilkan halaman pengajuan anggota	Sistem menampilkan pesan kesalahan

**4. Skenario Pengujian Proses Tampil Permohonan Anggota**  
**Tabel 3. 19** Skenario Pengujian Proses Tampil Permohonan Anggota

Identifikasi	
Nama Proses	Menampilkan Detail Permohonan Menjadi Anggota
Tujuan	Menampilkan data_permohonan pada verifikasi permohonan
Aktor	Atasan dan Bendahara
Deskripsi	
Skenario pengujian proses menampilkan detail permohonan menjadi anggota bertujuan untuk atasan dan bendahara dapat menampilkan detail permohonan menjadi anggota	
Kondisi Awal	
data_permohonan tersimpan pada tb_pengguna dan tb_permohonan	
Skenario Utama	
Aksi Aktor	Reaksi Sistem
1. Aktor memilih tombol verifikasi permohonan	2. Sistem menjalankan fungsi <code>getPeggunaPermohon</code> dan <code>getPeggunaPermohonan</code> untuk mengambil data_permohonan pada <code>tb_pengguna</code> dan <code>tb_permohonan</code>
Kondisi Akhir	
Berhasil	Gagal
Sistem menampilkan detail permohonan anggota	Sistem menampilkan pesan kesalahan

## 5. Skenario Pengujian Proses Memverifikasi Anggota

**Tabel 3. 20** Skenario Pengujian Proses Memverifikasi Permohonan

Identifikasi	
Nama Proses	Memverifikasi Permohonan Anggota
Tujuan	Melakukan validasi dan verifikasi data_permohonan pada tb_permohonan
Aktor	Atasan dan Bendahara
Deskripsi	
Skenario pengujian proses memverifikasi permohonan anggota bertujuan untuk atasan dan bendahara koperasi dapat melakukan validasi dan verifikasi terhadap permohonan yang diajukan oleh anggota	
Kondisi Awal	
data_permohonan tersimpan pada tb_pengguna dan tb_permohonan	
Skenario Utama	
Aksi Aktor	Reaksi Sistem
1. Aktor memilih button konfirmasi persetujuan atau konfirmasi penolakan	2 Sistem menjalankan fungsi verifikasi permohonan atau fungsi permohonan Ditolak untuk memverifikasi data_permohonan pada tb_permohonan
Kondisi Akhir	
Berhasil	Gagal

Sistem menampilkan total permohonan yang telah diverifikasi	Sistem menampilkan total permohonan yang belum terverifikasi
---	--

## 6. Skenario Pengujian Proses Mengajukan Permohonan Dana

**Tabel 3. 21** Skenario Pengujian Proses Mengajukan Permohonan Dana

Identifikasi	
Nama Proses	Mengajukan Permohonan Dana
Tujuan	Memasukan data_pengajuan pada tb_peminjaman
Aktor	Anggota Koperasi
Deskripsi	
Skenario pengujian proses mengajukan permohonan dana bertujuan untuk anggota koperasi dapat memasukkan data untuk mengajukan peminjaman dana	
Kondisi Awal	
data_pengajuan belum tersimpan pada tb_peminjaman	
Skenario Utama	
Aksi Aktor	Reaksi Sistem
1. Aktor memasukan data_pengajuan	2. Sistem menjalankan fungsi save untuk menambahkan data_pengajuan pada tb_peminjaman
Kondisi Akhir	
Berhasil	Gagal

Sistem menampilkan halaman pengajuan dana	Sistem menampilkan pesan kesalahan
---	------------------------------------

## 7. Skenario Pengujian Proses Tampil Permohonan Dana

**Tabel 3. 22** Skenario Pengujian Proses Tampil Permohonan Dana

Identifikasi	
Nama Proses	Menampilkan Detail Permohonan Dana
Tujuan	Menampilkan data_peminjaman pada halaman verifikasi permohonan dana
Aktor	Ketua, pengawas, dan bendahara
Deskripsi	
Skenario pengujian proses menampilkan detail permohonan dana bertujuan untuk ketua, pengawas, dan bendahara dapat menampilkan detail permohonan dana	
Kondisi Awal	
data_permohonan tersimpan pada tb_peminjaman	
Skenario Utama	
Aksi Aktor	Reaksi Sistem
1. Aktor memilih tombol verifikasi permohonan dana	2 Sistem menjalankan fungsi getPermohonanByStatusPengajuan untuk mengambil data_permohonan pada tb_peminjaman
Kondisi Akhir	

Berhasil	Gagal
Sistem menampilkan detail permohonan dana	Sistem menampilkan pesan kesalahan

## 8. Skenario Pengujian Proses Memverifikasi Dana

**Tabel 3. 23** Skenario Pengujian Proses Memverifikasi Dana

<b>Identifikasi</b>	
Nama Proses	Memverifikasi Permohonan Dana
Tujuan	Melakukan validasi dan verifikasi data_permohonan_dana pada tb_peminjaman
Aktor	Ketua atau Pengawas
<b>Deskripsi</b>	
Skenario pengujian proses memverifikasi permohonan dana bertujuan untuk ketua maupun pengawas koperasi dapat melakukan validasi dan verifikasi terhadap permohonan dana yang diajukan oleh anggota	
<b>Kondisi Awal</b>	
data_permohonan_dana tersimpan pada tb_peminjaman	
<b>Skenario Utama</b>	
Aksi Aktor	Reaksi Sistem
1. Aktor memilih button konfirmasi persetujuan atau konfirmasi penolakan	2. Sistem menjalankan fungsi verifikasiPermohonan atau fungsi verifikasiPenolakan untuk memverifikasi data_permohonan_dana pada tb_peminjaman

<b>Kondisi Akhir</b>	
Berhasil	Gagal
Sistem menampilkan total permohonan dana yang telah diverifikasi	Sistem menampilkan total permohonan dana yang belum terverifikasi

### 9. Skenario Pengujian Proses Mengajukan Laporan Koperasi

**Tabel 3. 24** Skenario Pengujian Proses Mengajukan Laporan Koperasi

<b>Identifikasi</b>	
Nama Proses	Mengajukan Laporan Koperasi
Tujuan	Memasukan data_laporan pada tb_laporan_koperasi
Aktor	Bendahara
<b>Deskripsi</b>	
Skenario pengujian proses mengajukan laporan koperasi bertujuan untuk bendahara dapat memasukkan data untuk mengajukan laporan koperasi	
<b>Kondisi Awal</b>	
data_laporan_koperasi belum tersimpan pada tb_laporan_koperasi	
<b>Skenario Utama</b>	
Aksi Aktor	Reaksi Sistem

1. Aktor memasukan data_laporan_koperasi	2. Sistem menjalankan fungsi save untuk menambahkan data_laporan_koperasi pada tb_laporan_koperasi
<b>Kondisi Akhir</b>	
Berhasil	Gagal
Sistem menampilkan halaman pengajuan laporan koperasi	Sistem menampilkan pesan kesalahan

**10. Skenario Pengujian Proses Menampilkan Detail Laporan**  
**Tabel 3. 25 Skenario Pengujian Proses Menampilkan Detail Laporan**

<b>Identifikasi</b>	
Nama Proses	Menampilkan Detail Laporan Koperasi
Tujuan	Menampilkan data_laporan pada halaman verifikasi laporan
Aktor	Ketua dan Pengawas
<b>Deskripsi</b>	
Skenario pengujian proses menampilkan detail laporan koperasi bertujuan untuk ketua maupun pengawas dapat menampilkan detail laporan koperasi	
<b>Kondisi Awal</b>	
data_laporan_koperasi tersimpan pada tb_laporan_koperasi	
<b>Skenario Utama</b>	

Aksi Aktor	Reaksi Sistem
1. Aktor memilih tombol verifikasi laporan koperasi	2 Sistem menjalankan fungsi get untuk mengambil data_laporan pada tb_laporan_koperasi
<b>Kondisi Akhir</b>	
Berhasil	Gagal
Sistem menampilkan detail laporan koperasi	Sistem menampilkan pesan kesalahan

### 11. Skenario Pengujian Proses Memverifikasi Laporan

**Tabel 3. 26** Skenario Pengujian Proses Memverifikasi Laporan

<b>Identifikasi</b>	
Nama Proses	Memverifikasi Laporan Koperasi
Tujuan	Melakukan validasi dan verifikasi data_laporan_koperasi pada tb_laporan_koperasi
Aktor	Ketua atau Pengawas
<b>Deskripsi</b>	
Skenario pengujian proses memverifikasi laporan koperasi bertujuan untuk ketua maupun pengawas koperasi dapat melakukan validasi dan verifikasi terhadap laporan koperasi yang diajukan oleh bendahara	
<b>Kondisi Awal</b>	
data_laporan_koperasi tersimpan pada tb_laporan_koperasi	
<b>Skenario Utama</b>	

Aksi Aktor	Reaksi Sistem
1. Aktor memilih button konfirmasi persetujuan atau konfirmasi penolakan	2 Sistem menjalankan fungsi verifikasi laporan untuk memverifikasi data_laporan_koperasi pada tb_laporan_koperasi
<b>Kondisi Akhir</b>	
Berhasil	Gagal
Sistem menampilkan laporan koperasi yang telah diverifikasi	Sistem menampilkan laporan koperasi yang belum terverifikasi

## 12. Skenario Pengujian Proses Menampilkan Data Simpanan

**Tabel 3. 27** Skenario Pengujian Proses Menampilkan Data Simpanan

<b>Identifikasi</b>	
Nama Proses	Menampilkan Data Simpanan
Tujuan	Menampilkan data_simpanan pada halaman simpanan
Aktor	Anggota koperasi
<b>Deskripsi</b>	
Skenario pengujian proses menampilkan data simpanan bertujuan untuk anggota koperasi dapat menampilkan data simpanan	
<b>Kondisi Awal</b>	
data_simpanan tersimpan pada tb_penyimpanan	
<b>Skenario Utama</b>	

Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu simpanan	2 Sistem menjalankan fungsi getPenyimpananByBulanTahun untuk mengambil data_simpanan pada tb_penyimpanan
<b>Kondisi Akhir</b>	
Berhasil	Gagal
Sistem menampilkan data simpanan	Sistem menampilkan pesan kesalahan

### 13. Skenario Pengujian Proses Menampilkan Data Pinjaman

**Tabel 3. 28** Skenario Pengujian Proses Menampilkan Data Pinjaman

<b>Identifikasi</b>	
Nama Proses	Menampilkan Data Pinjaman
Tujuan	Menampilkan data_pinjaman pada halaman pinjaman
Aktor	Anggota koperasi
<b>Deskripsi</b>	
Skenario pengujian proses menampilkan data pinjaman bertujuan untuk anggota koperasi dapat menampilkan data pinjaman	
<b>Kondisi Awal</b>	
data_ pinjaman tersimpan pada tb_peminjaman	
<b>Skenario Utama</b>	

Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu pinjaman	2 Sistem menjalankan fungsi getPeminjam untuk mengambil data_ pinjaman pada tb_peminjaman
<b>Kondisi Akhir</b>	
Berhasil	Gagal
Sistem menampilkan data pinjaman	Sistem menampilkan pesan kesalahan

#### 14. Skenario Pengujian Proses Menampilkan Anggota

**Tabel 3. 29** Skenario Pengujian Proses Menampilkan Anggota

<b>Identifikasi</b>	
Nama Proses	Menampilkan Data Anggota Koperasi
Tujuan	Menampilkan data_anggota pada halaman daftar anggota
Aktor	Ketua, bendahara, dan atasan
<b>Deskripsi</b>	
Skenario pengujian proses menampilkan daftar anggota bertujuan untuk ketua, bendahara, dan atasan dapat menampilkan daftar anggota koperasi	
<b>Kondisi Awal</b>	
data_anggota tersimpan pada tb_pengguna	

<b>Skenario Utama</b>	
Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu daftar anggota	2. Sistem menjalankan fungsi <code>getpengguna</code> untuk mengambil <code>data_angota</code> pada <code>tb_pengguna</code>
<b>Kondisi Akhir</b>	
Berhasil	Gagal
Sistem menampilkan data anggota	Sistem menampilkan pesan kesalahan

**15. Skenario Pengujian Proses Menampilkan Data Keaktifan**  
**Tabel 3. 30** Skenario Pengujian Proses Menampilkan Data Keaktifan

<b>Identifikasi</b>	
Nama Proses	Menampilkan Data Keaktifan
Tujuan	Menampilkan <code>data_keaktifan</code> pada halaman keaktifan anggota
Aktor	Anggota Koperasi
<b>Deskripsi</b>	
Skenario pengujian proses menampilkan data keaktifan anggota bertujuan untuk anggota koperasi dapat menampilkan data keaktifan anggota	
<b>Kondisi Awal</b>	
data_keaktifan tersimpan pada <code>tb_laporan_aktif</code>	

Skenario Utama	
Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu keaktifan anggota	2 Sistem menjalankan fungsi get untuk mengambil data_keaktifan pada tb_laporan_aktif
Kondisi Akhir	
Berhasil	Gagal
Sistem menampilkan data keaktifan anggota	Sistem menampilkan pesan kesalahan

### 16. Skenario Pengujian Proses Menambah Simpanan Wajib

**Tabel 3. 31** Skenario Pengujian Proses Menambah Simpanan Wajib

Identifikasi	
Nama Proses	Proses menambah data simpanan wajib
Tujuan	Menambah semua data_simpananwajib anggota pada tb_penyimpanan
Aktor	Bendahara
Deskripsi	
Skenario pengujian proses menambah simpanan wajib anggota koperasi bertujuan untuk bendahara dapat menambahkan simpanan wajib anggota koperasi	
Kondisi Awal	

data_simpananwajib belum tersimpan pada tb_penyimpanan	
<b>Skenario Utama</b>	
Aksi Aktor	Reaksi Sistem
1. Aktor memasukkan data_simpanan wajib	2. Sistem menjalankan fungsi sinkronisasi untuk menambahkan data_simpananwajib pada tb_penyimpanan.
<b>Kondisi Akhir</b>	
Berhasil	Gagal
Data_simpananwajib berhasil ditambahkan ke tb_penyimpanan	Sistem menampilkan pesan kesalahan

### 17. Skenario Pengujian Proses Menambah Simpanan Sukarela

**Tabel 3. 32** Skenario Pengujian Proses Menambah Simpanan Sukarela

<b>Identifikasi</b>	
Nama Proses	Proses menambah simpanan sukarela
Tujuan	Memasukkan data_simpanansukarela pada tb_penyimpanan
Aktor	Anggota koperasi
<b>Deskripsi</b>	
Skenario pengujian proses menambah simpanan sukarela bertujuan untuk anggota koperasi dapat menambahkan simpanan sukarela	

anggota koperasi	
<b>Kondisi Awal</b>	
data_simpanansukarela belum tersimpan pada tb_penyimpanan	
<b>Skenario Utama</b>	
Aksi Aktor	Reaksi Sistem
1. Aktor memasukkan data_simpanan sukarela	2. Sistem menjalankan fungsi simpan untuk menambahkan data_simpanansukarela pada tb_penyimpanan.
<b>Kondisi Akhir</b>	
Berhasil	Gagal
Data_simpanansukarela berhasil ditambahkan ke dalam tb_penyimpanan	Sistem menampilkan pesan kesalahan

### 18. Skenario Pengujian Proses Mengubah Data Anggota

**Tabel 3. 33** Skenario Pengujian Proses Mengubah Data Anggota

<b>Identifikasi</b>	
Nama Proses	Proses mengubah data anggota koperasi
Tujuan	Mengubah data_anggota pada tb_pengguna
Aktor	Anggota koperasi
<b>Deskripsi</b>	

Skenario pengujian proses mengubah data anggota bertujuan untuk anggota koperasi dapat mengubah data anggota sesuai dengan yang dibutuhkan	
<b>Kondisi Awal</b>	
data_anggota tersimpan pada tb_pengguna	
<b>Skenario Utama</b>	
Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu akun anggota, lalu memilih tombol simpan perubahan jika data pengguna telah diubah	2. Sistem menjalankan fungsi ubah untuk mengubah data pada tb_pengguna
<b>Kondisi Akhir</b>	
Berhasil	Gagal
data_anggota berhasil diubah	Sistem menampilkan pesan kesalahan

### 19. Skenario Pengujian Proses Mengubah Status Anggota

**Tabel 3. 34** Skenario Pengujian Proses Mengubah Status Anggota

<b>Identifikasi</b>	
Nama Proses	Proses mengubah status anggota
Tujuan	Mengubah data_status_anggota pada tb_pengguna
Aktor	Bendahara
<b>Deskripsi</b>	

Skenario pengujian proses mengubah status anggota bertujuan untuk bendahara dapat mengubah status data anggota jika anggota tersebut mengundurkan diri dari koperasi	
<b>Kondisi Awal</b>	
data_anggota tersimpan pada tb_pengguna	
<b>Skenario Utama</b>	
Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu akun anggota, lalu memilih tombol simpan perubahan jika status anggota telah diubah	2. Sistem menjalankan fungsi ubah untuk mengubah data pada tb_pengguna
<b>Kondisi Akhir</b>	
Berhasil	Gagal
data_status_anggota berhasil diubah	Sistem menampilkan pesan kesalahan

**3.4 Analisis kuisiонер**

Pengujian dengan menggunakan kuisiонер berfungsi untuk mengetahui penilaian pengguna terhadap sistem yang sudah dibangun. Hasil dari pengujian kuisiонер ini, dapat menjadi masukan bagi penulis untuk ke depannya..

**1. Perancangan Kuisiонер untuk Ketua**

Berikut ini adalah perancangan untuk mengetahui penilaian pengguna dari Koperasi Mandiri Sejahtera pada Sistem Informasi Simpan Pinjam Berbasis *Website*.

**Tabel 3. 35** Perancangan Kuisiонер untuk Ketua

<b>A. Learnability</b>		<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>
1	Tampilan untuk proses mengajukan permohonan dana mudah dipahami				

2	Tampilan untuk proses memverifikasi permohonan dana mudah dipahami				
3	Tampilan untuk proses memverifikasi laporan koperasi mudah dipahami				
4	Tampilan untuk proses menambah simpanan sukarela mudah dipahami				
5	Tampilan untuk proses menampilkan data transaksi mudah dipahami				
6	Tampilan untuk menampilkan data keaktifan anggota mudah dipahami				
7	Tampilan untuk menampilkan data anggota koperasi mudah dipahami				
<b>A. Efficiency</b>					
8	Saat mengajukan permohonan dana, informasi dapat ditampilkan dengan cepat dan tepat				
9	Saat memverifikasi permohonan dana, informasi dapat ditampilkan dengan cepat dan tepat				
10	Saat memverifikasi laporan koperasi, informasi dapat ditampilkan dengan cepat dan tepat				
11	Saat menambahkan simpanan sukarela, informasi dapat ditampilkan dengan cepat dan tepat				
12	Saat menampilkan data transaksi, informasi dapat ditampilkan dengan cepat dan tepat				
13	Saat menampilkan data keaktifan anggota, informasi dapat ditampilkan dengan cepat dan tepat				
14	Saat menampilkan data anggota koperasi, informasi dapat ditampilkan dengan cepat dan tepat				
<b>B. Memorability</b>					
15	Anda sebagai ketua dapat dengan mudah mengingat cara mengajukan permohonan dana				

16	Anda sebagai ketua dapat dengan mudah mengingat cara memverifikasi permohonan dana				
17	Anda sebagai ketua dapat dengan mudah mengingat cara memverifikasi laporan koperasi				
18	Anda sebagai ketua dapat dengan mudah mengingat cara menambahkan simpanan sukarela				
19	Anda sebagai ketua dapat dengan mudah mengingat cara menampilkan data transaksi				
20	Anda sebagai ketua dapat dengan mudah mengingat cara menampilkan data keaktifan anggota				
21	Anda sebagai ketua dapat dengan mudah mengingat cara menampilkan data anggota koperasi				
<b>C. Error</b>					
22	Pesan kesalahan selalu muncul ketika anda melakukan kesalahan saat menggunakan sistem				
23	Apakah pesan kesalahan error sesuai dengan konten?				
<b>D. Satisfaction</b>					
24	Teks informasi dapat anda baca dengan mudah				
25	Bahasa yang digunakan dalam sistem mudah untuk dipahami				
26	Desain warna dan tata letak dari sistem nyaman untuk dilihat				
27	Informasi yang ditampilkan sesuai dengan kebutuhan dan ekspektasi anda sebagai pengguna				
28	Kedepannya anda lebih memilih menggunakan sistem ini untuk melakukan peminjaman dana, memverifikasi permohonan, melihat				

	transaksi simpan pinjam dan melihat data anggota dan keaktifan anggota				
--	--	--	--	--	--

## 2. Perancangan Kuesioner untuk Pengawas

Berikut ini adalah untuk mengetahui penilaian pengguna dari Koperasi Mandiri Sejahtera pada Sistem Informasi Simpan Pinjam Berbasis *Website*.

**Tabel 3. 36** Perancangan Kuesioner untuk Pengawas

<b>A. Learnability</b>		<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>
1	Tampilan untuk proses mengajukan permohonan dana mudah dipahami				
2	Tampilan untuk proses memverifikasi permohonan dana mudah dipahami				
3	Tampilan untuk proses memverifikasi laporan koperasi mudah dipahami				
4	Tampilan untuk proses menambah simpanan sukarela mudah dipahami				
5	Tampilan untuk proses menampilkan data transaksi mudah dipahami				
6	Tampilan untuk menampilkan data keaktifan anggota mudah dipahami				
7	Tampilan untuk menampilkan data anggota koperasi mudah dipahami				
<b>B. Efficiency</b>					
8	Saat mengajukan permohonan dana, informasi dapat ditampilkan dengan cepat dan tepat				
9	Saat memverifikasi permohonan dana, informasi dapat ditampilkan dengan cepat dan tepat				
10	Saat memverifikasi laporan koperasi, informasi dapat ditampilkan dengan cepat dan tepat				
11	Saat menambahkan simpanan sukarela, informasi dapat ditampilkan dengan cepat dan tepat				

12	Saat menampilkan data transaksi, informasi dapat ditampilkan dengan cepat dan tepat				
13	Saat menampilkan data keaktifan anggota, informasi dapat ditampilkan dengan cepat dan tepat				
14	Saat menampilkan data anggota koperasi, informasi dapat ditampilkan dengan cepat dan tepat				
<b>C. Memorability</b>					
15	Anda sebagai pengawas dapat dengan mudah mengingat cara mengajukan permohonan dana				
16	Anda sebagai pengawas dapat dengan mudah mengingat cara memverifikasi permohonan dana				
17	Anda sebagai pengawas dapat dengan mudah mengingat cara memverifikasi laporan koperasi				
18	Anda sebagai pengawas dapat dengan mudah mengingat cara menambahkan simpanan sukarela				
19	Anda sebagai pengawas dapat dengan mudah mengingat cara menampilkan data transaksi				
20	Anda sebagai pengawas dapat dengan mudah mengingat cara menampilkan data keaktifan anggota				
21	Anda sebagai pengawas dapat dengan mudah mengingat cara menampilkan data anggota koperasi				
<b>D. Error</b>					
22	Pesan kesalahan selalu muncul ketika anda melakukan kesalahan saat menggunakan sistem				
23	Apakah pesan kesalahan error sesuai dengan konten?				
<b>E. Satisfaction</b>					

24	Teks informasi dapat anda baca dengan mudah				
25	Bahasa yang digunakan dalam sistem mudah untuk dipahami				
26	Desain warna dan tata letak dari sistem nyaman untuk dilihat				
27	Informasi yang ditampilkan sesuai dengan kebutuhan dan ekspektasi anda sebagai pengguna				
28	Kedepannya anda lebih memilih menggunakan sistem ini untuk melakukan peminjaman dana, memverifikasi permohonan, melihat transaksi simpan pinjam dan melihat data anggota dan keaktifan anggota				

### 3. Perancangan Kuesioner untuk Atasan Langsung

Berikut ini adalah perancangan untuk mengetahui penilaian pengguna dari Koperasi Mandiri Sejahtera pada Sistem Informasi Simpan Pinjam Berbasis *Website*.

**Tabel 3. 37** Perancangan Kuesioner untuk Atasan Langsung

<b>A. Learnability</b>		<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>
1	Tampilan untuk proses mengajukan permohonan dana mudah dipahami				
2	Tampilan untuk proses memverifikasi permohonan anggota mudah dipahami				
3	Tampilan untuk proses menampilkan laporan koperasi mudah dipahami				
4	Tampilan untuk proses menambah simpanan sukarela mudah dipahami				
6	Tampilan untuk proses menampilkan data transaksi mudah dipahami				
7	Tampilan untuk menampilkan data keaktifan anggota mudah dipahami				
<b>B. Efficiency</b>					
8	Saat mengajukan permohonan dana, informasi dapat ditampilkan dengan cepat dan tepat				

9	Saat memverifikasi permohonan anggota, informasi dapat ditampilkan dengan cepat dan tepat				
10	Saat menampilkan laporan koperasi, informasi dapat ditampilkan dengan cepat dan tepat				
11	Saat menambahkan simpanan sukarela, informasi dapat ditampilkan dengan cepat dan tepat				
12	Saat menampilkan data transaksi, informasi dapat ditampilkan dengan cepat dan tepat				
13	Saat menampilkan data keaktifan anggota, informasi dapat ditampilkan dengan cepat dan tepat				
<b>C. Memorability</b>					
14	Anda sebagai atasan dapat dengan mudah mengingat cara mengajukan permohonan dana				
15	Anda sebagai atasan dapat dengan mudah mengingat cara memverifikasi anggota koperasi				
16	Anda sebagai atasan dapat dengan mudah mengingat cara memverifikasi laporan koperasi				
17	Anda sebagai atasan dapat dengan mudah mengingat cara menambahkan simpanan sukarela				
18	Anda sebagai atasan dapat dengan mudah mengingat cara menampilkan data transaksi				
19	Anda sebagai atasan dapat dengan mudah mengingat cara menampilkan data keaktifan anggota				
<b>D. Error</b>					
20	Pesan kesalahan selalu muncul ketika anda melakukan kesalahan saat menggunakan sistem				

21	Apakah pesan kesalahan error sesuai dengan konten?				
<b>E. Satisfaction</b>					
22	Teks informasi dapat anda baca dengan mudah				
23	Bahasa yang digunakan dalam sistem mudah untuk dipahami				
24	Desain warna dan tata letak dari sistem nyaman untuk dilihat				
25	Informasi yang ditampilkan sesuai dengan kebutuhan dan ekspektasi anda sebagai pengguna				
26	Kedepannya anda lebih memilih menggunakan sistem ini untuk melakukan peminjaman dana, memverifikasi permohonan, melihat transaksi simpan pinjam dan melihat data anggota dan keaktifan anggota				

#### 4. Perancangan Kuesioner untuk Bendahara

Berikut ini adalah perancangan untuk mengetahui penilaian pengguna dari Koperasi Mandiri Sejahtera pada Sistem Informasi Simpan Pinjam Berbasis *Website*.

**Tabel 3. 38** Perancangan Kuesioner untuk Bendahara

<b>B. Learnability</b>		<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>
1	Tampilan untuk proses mengajukan permohonan dana mudah dipahami				
2	Tampilan untuk proses memverifikasi permohonan anggota mudah dipahami				
3	Tampilan untuk proses mengajukan laporan koperasi mudah dipahami				
4	Tampilan untuk proses menambah simpanan sukarela mudah dipahami				
5	Tampilan untuk proses menambah simpanan wajib mudah dipahami				
6	Tampilan untuk proses menampilkan data transaksi mudah dipahami				

7	Tampilan untuk menampilkan data keaktifan anggota mudah dipahami				
<b>E. Efficiency</b>					
8	Saat mengajukan permohonan dana, informasi dapat ditampilkan dengan cepat dan tepat				
9	Saat memverifikasi permohonan anggota, informasi dapat ditampilkan dengan cepat dan tepat				
10	Saat mengajukan laporan koperasi, informasi dapat ditampilkan dengan cepat dan tepat				
11	Saat menambahkan simpanan sukarela, informasi dapat ditampilkan dengan cepat dan tepat				
12	Saat menambahkan simpanan wajib, informasi dapat ditampilkan dengan cepat dan tepat				
13	Saat menampilkan data transaksi, informasi dapat ditampilkan dengan cepat dan tepat				
14	Saat menampilkan data keaktifan anggota, informasi dapat ditampilkan dengan cepat dan tepat				
<b>F. Memorability</b>					
15	Anda sebagai bendahara dapat dengan mudah mengingat cara mengajukan permohonan dana				
16	Anda sebagai bendahara dapat dengan mudah mengingat cara memverifikasi anggota koperasi				
17	Anda sebagai bendahara dapat dengan mudah mengingat cara memverifikasi laporan koperasi				
18	Anda sebagai bendahara dapat dengan mudah mengingat cara menambahkan simpanan sukarela				

19	Anda sebagai bendahara dapat dengan mudah mengingat cara menambahkan simpanan wajib				
20	Anda sebagai bendahara dapat dengan mudah mengingat cara menampilkan data transaksi				
21	Anda sebagai bendahara dapat dengan mudah mengingat cara menampilkan data keaktifan anggota				
<b>G. Error</b>					
22	Pesan kesalahan selalu muncul ketika anda melakukan kesalahan saat menggunakan sistem				
23	Apakah pesan kesalahan error sesuai dengan konten?				
<b>H. Satisfaction</b>					
24	Teks informasi dapat anda baca dengan mudah				
25	Bahasa yang digunakan dalam sistem mudah untuk dipahami				
26	Desain warna dan tata letak dari sistem nyaman untuk dilihat				
27	Informasi yang ditampilkan sesuai dengan kebutuhan dan ekspektasi anda sebagai pengguna				
28	Kedepannya anda lebih memilih menggunakan sistem ini untuk melakukan peminjaman dana, memverifikasi permohonan, melihat transaksi simpan pinjam dan melihat data anggota dan keaktifan anggota				

### 5. Perancangan Kuesioner untuk Anggota

Berikut adalah perancangan kuesioner anggota untuk mengetahui penilaian pengguna dari Koperasi Mandiri Sejahtera pada Sistem Informasi Simpan Pinjam Berbasis *Website*.

**Tabel 3. 39** Perancangan Kuesioner untuk Anggota

<b>A. Learnability</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>
------------------------	----------	----------	----------	----------

1	Tampilan untuk proses mengajukan permohonan dana mudah dipahami				
2	Tampilan untuk proses menampilkan data permohonan anggota mudah dipahami				
3	Tampilan untuk proses menambah simpanan sukarela mudah dipahami				
4	Tampilan untuk proses menampilkan data transaksi mudah dipahami				
5	Tampilan untuk menampilkan data keaktifan anggota mudah dipahami				
<b>B. Efficiency</b>					
6	Saat mengajukan permohonan dana, informasi dapat ditampilkan dengan cepat dan tepat				
7	Saat menampilkan data permohonan anggota, informasi dapat ditampilkan dengan cepat dan tepat				
8	Saat menambahkan simpanan sukarela, informasi dapat ditampilkan dengan cepat dan tepat				
9	Saat menampilkan data transaksi, informasi dapat ditampilkan dengan cepat dan tepat				
10	Saat menampilkan data keaktifan anggota, informasi dapat ditampilkan dengan cepat dan tepat				
<b>C. Memorability</b>					
11	Anda sebagai anggota dapat dengan mudah mengingat cara mengajukan permohonan dana				
12	Anda sebagai anggota dapat dengan mudah mengingat cara menampilkan data permohonan anggota				
13	Anda sebagai anggota dapat dengan mudah mengingat cara menambahkan simpanan sukarela				
14	Anda sebagai anggota dapat dengan mudah mengingat cara menampilkan data transaksi				

15	Anda sebagai anggota dapat dengan mudah mengingat cara menampilkan data keaktifan anggota				
<b>D. Error</b>					
16	Pesan kesalahan selalu muncul ketika anda melakukan kesalahan saat menggunakan sistem				
17	Apakah pesan kesalahan error sesuai dengan konten?				
<b>E. Satisfaction</b>					
18	Teks informasi dapat anda baca dengan mudah				
19	Bahasa yang digunakan dalam sistem mudah untuk dipahami				
20	Desain warna dan tata letak dari sistem nyaman untuk dilihat				
21	Informasi yang ditampilkan sesuai dengan kebutuhan dan ekspektasi anda sebagai pengguna				
22	Kedepannya anda lebih memilih menggunakan sistem ini untuk melakukan peminjaman dana, menampilkan permohonan, melihat transaksi simpan pinjam dan melihat data anggota dan keaktifan anggota				

## 6. Perancangan Kuesioner untuk Pegawai

Berikut ini adalah perancangan kuesioner untuk mengetahui penilaian pengguna dari Koperasi Mandiri Sejahtera pada Sistem Informasi Simpan Pinjam Berbasis *Website*.

**Tabel 3. 40** Perancangan Kuesioner untuk Pegawai

<b>A. Learnability</b>		<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>
1	Tampilan untuk proses daftar/buat akun mudah dipahami				
2	Tampilan untuk proses mengajukan formulir menjadi anggota koperasi mudah dipahami				
3	Tampilan untuk proses menampilkan data permohonan anggota mudah dipahami				
<b>B. Efficiency</b>					

4	Saat menampilkan proses daftar/buat akun, informasi dapat ditampilkan dengan cepat dan tepat				
5	Saat mengajukan formulir permohonan anggota, informasi dapat ditampilkan dengan cepat dan tepat				
6	Saat menampilkan data permohonan anggota, informasi dapat ditampilkan dengan cepat dan tepat				
<b>C. Memorability</b>					
7	Anda sebagai pegawai/calon anggota koperasi dapat dengan mudah mengingat cara daftar/buat akun anggota				
8	Anda sebagai pegawai/calon anggota koperasi dapat dengan mudah mengingat cara mengajukan formulir permohonan anggota				
9	Anda sebagai pegawai/calon anggota koperasi dapat dengan mudah mengingat cara menampilkan data permohonan anggota				
<b>D. Error</b>					
10	Pesan kesalahan selalu muncul ketika anda melakukan kesalahan saat menggunakan sistem				
11	Apakah error sesuai dengan konten?				
<b>E. Satisfaction</b>					
12	Teks informasi dapat anda baca dengan mudah				
13	Bahasa yang digunakan dalam sistem mudah untuk dipahami				
14	Desain warna dan tata letak dari sistem nyaman untuk dilihat				
15	Informasi yang ditampilkan sesuai dengan kebutuhan dan ekspektasi anda sebagai pengguna				
16	Kedepannya anda lebih memilih menggunakan sistem ini untuk melakukan permohonan menjadi anggota koperasi				