

BAB III

METODOLOGI

3.1 Peralatan Pendukung

Dalam penelitian dibutuhkan beberapa alat dan bahan untuk mendukung berjalannya perancangan dan implementasi aplikasi.

3.1.1 Alat

Alat yang digunakan berupa perangkat keras dan perangkat lunak.

a. Perangkat Keras

1. Personal Computer (PC) atau Laptop.

64 bit architecture processor,
4 GB Random Access Memmory (RAM),
Sistem Operasi Windows 8.

2. Printer Dokumen untuk mencetak laporan.

b. Perangkat Lunak

1. Netbeans IDE 8.

2. MySQL Server 5.6.

c. Perangkat Keras Pengguna atau *Client*

1. Personal Computer (PC) atau Laptop.

32/64 bit architecture processor,
2 GB Random Access Memmory (RAM),
Sistem Operasi Windows XP/7/8.

2. Printer Dokumen untuk mencetak laporan.

d. Perangkat Lunak

1. Netbeans IDE 7/8.

2. MySQL Server 5.6

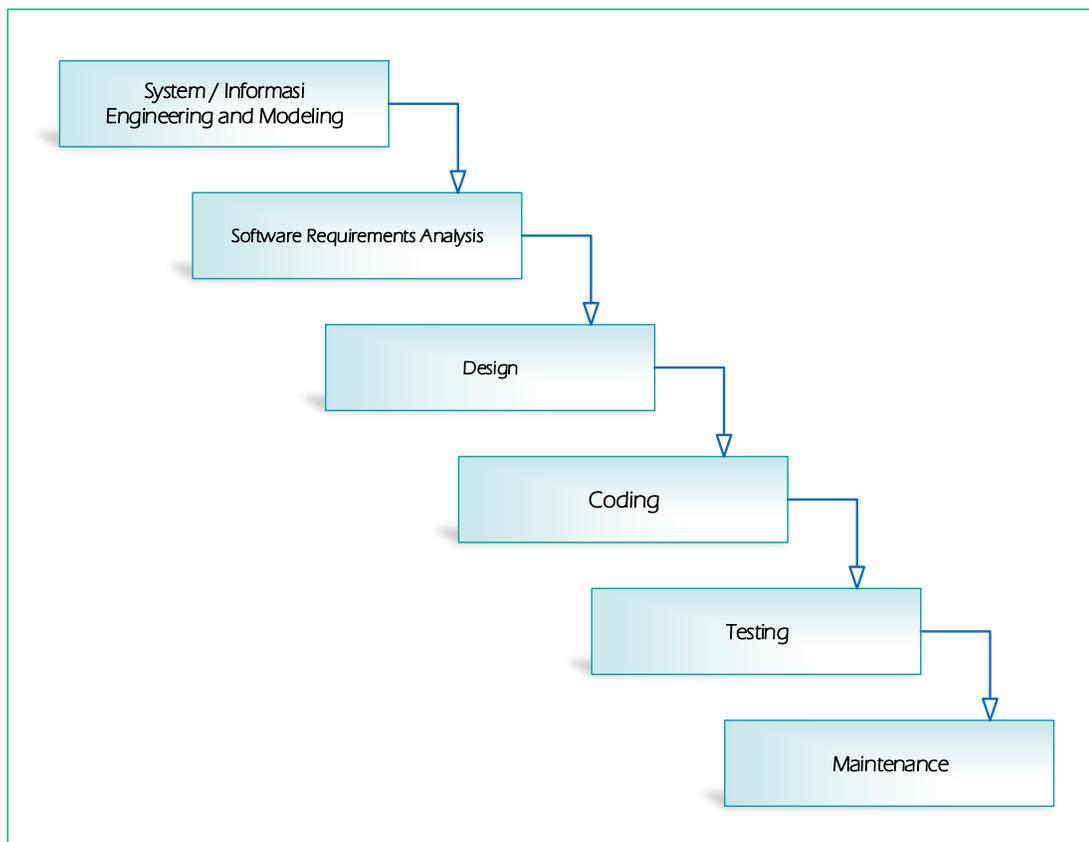
3.1.2 Bahan

Dalam penelitian bahan yang digunakan adalah:

1. Data yang diperoleh melalui studi literatur berdasarkan penelitian sebelumnya yang masih memiliki keterkaitan dengan aplikasi yang dikembangkan. Dari data yang diperoleh, maka didapatkan kebutuhan dari aplikasi pada saat pengembangan aplikasi.
2. Data anggota, data transaksi simpanan, data transaksi pinjaman dan angsuran perbulan dari anggota pada BMT Insan Madani Kalibayem.

3.2 Alur Penelitian

Dengan mengikuti pendapat Roger S. Pressman (2007), tahapan pada model SDLC *Waterfall* dalam pembuatan aplikasi dapat dilihat pada Gambar 3.1.



Gambar 3.1 Model SLDC (*Software Development Life Cycle*)

Berikut adalah penjeleasan tentang Gambar 3.1 :

a. *System or Information Engineering and Modeling*

Langkah pertama adalah mencari kebutuhan dari keseluruhan sistem yang diaplikasikan ke dalam bentuk *software*. Seperti komponen *hardware* yang dibutuhkan dan basis data yang digunakan.

b. *Software Requirement Analysis*

Proses pencarian kebutuhan diintensifkan dan difokuskan pada *software*, seperti bagaimana tampilan *user interface* dan kebutuhan fungsi-fungsi yang digunakan dalam aplikasi.

c. *System Design*

System Design merupakan proses yang digunakan untuk mengubah kebutuhan-kebutuhan sebelumnya menjadi representasi ke dalam bentuk “*blueprint*” *software* sebelum *coding* dimulai. Desain harus dapat mengimplementasikan kebutuhan yang telah disebutkan pada tahap sebelumnya.

d. *Coding*

Untuk dapat dimengerti oleh mesin yang dalam, maka desain aplikasi harus diubah bentuknya menjadi bentuk yang dapat dimengerti oleh mesin, yaitu ke dalam bahasa pemrograman melalui proses *coding*. Bahasa pemrograman yang digunakan adalah bahasa pemrograman Java.

e. *Testing*

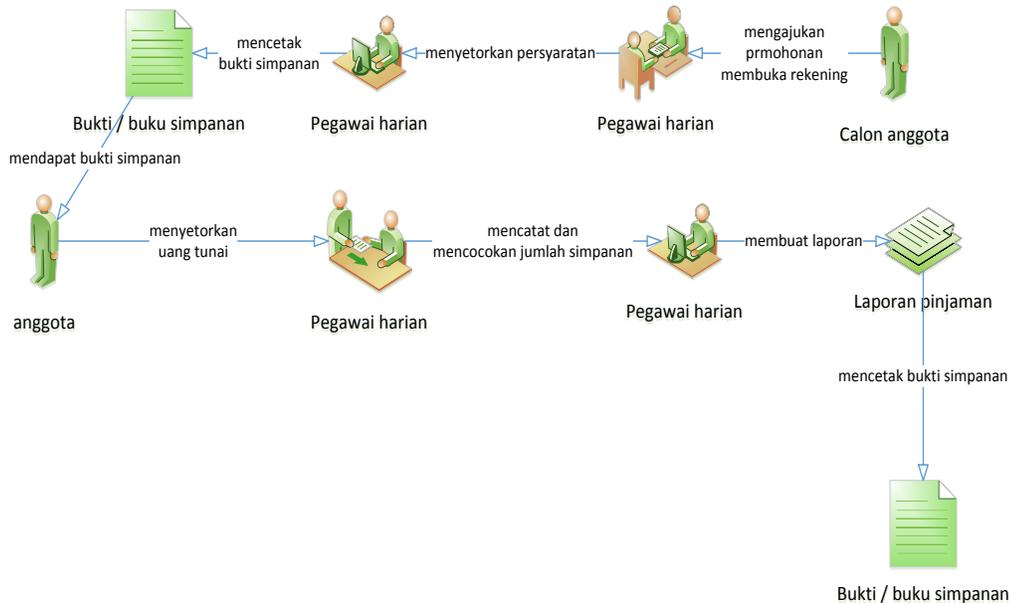
Dalam tahap testing dilakukan pengujian *software* yang sudah dibuat. Semua fungsi-fungsi *software* harus diuji coba agar *software* bebas dari *error* atau kesalahan dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya.

f. *Maintenance*

Pemeliharaan dalam aplikasi diperlukan, termasuk di dalamnya adalah pengembangan, karena *software* yang dibuat tidak selamanya hanya seperti itu. Ketika dijalankan mungkin saja masih ada *error* kecil yang tidak ditemukan sebelumnya atau ada penambahan fitur-fitur yang belum ada pada *software*.

3.3 Analisis Transaksi Simpanan

Gambaran transaksi simpanan yang sedang berjalan pada BMT Insan Madani dapat dilihat pada Gambar 3.2.



Gambar 3.2 Proses Transaksi Simpanan yang Sedang Berjalan

Berikut adalah penjelasan tentang Gambar 3.2 :

a. Anggota

Mengajukan permohonan pembukaan rekening dengan mengisi permohonan sesuai dengan kartu identitas untuk pembukaan rekening. Setelah terdaftar menjadi anggota dengan melakukan setoran Simpanan wajib, simpanan pokok dan simpanan sukarela.

b. Penyetoran

Pada setoran selanjutnya anggota mengisi dan menandatangani slip setoran yang berisikan tanggal, jenis simpanan, nomor rekening, nama pemilik dan jumlah nominal yang akan disimpan atau ditabung.

c. Pihak Manajemen

Kemudian pihak manajemen BMT menerima uang tunai dan mencocokkan jumlahnya dengan jumlah yang tertera pada slip setoran. Setelah itu teller dapat memvalidasi dengan mencocokkan saldo simpanan dengan arsip simpanan atau komputer data simpanan.

d. Laporan

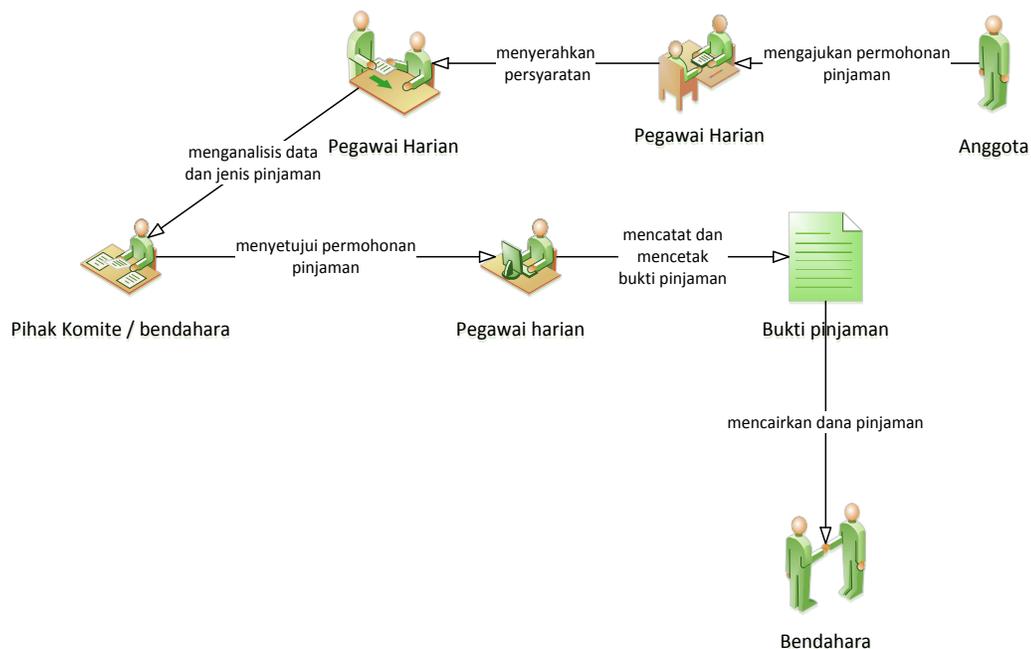
Pihak manajemen BMT dapat memvalidasi slip setoran pada printer dan mencetak setoran dan saldo terakhir pada buku simpanan. Kemudian slip setoran diserahkan kepada bagian pembukuan.

e. Bukti Simpanan

Rekening tabungan akan diberi setempel sesuai dengan jenis simpanan dan kemudian diserahkan kepada nasabah.

3.4 Analisis Transaksi Peminjaman

Gambaran transaksi pinjaman yang sedang berjalan pada BMT Insan Madani dapat dilihat pada Gambar 3.3.



Gambar 3.3 Proses Transaksi Pinjaman yang Sedang Berjalan

Berikut adalah penjelasan tentang Gambar 3.3 :

1. Pengajuan Pinjaman

Anggota mengajukan permohonan secara tertulis kepada BMT dengan mengisi formulir pengajuan. Proses ini dilakukan oleh pegawai harian BMT. Ini dilakukan setelah semua persyaratan formal dipenuhi, seperti yang menyangkut legalitas calon peminjam (data diri, dsb).

2. Analisis usulan pembiayaan

Sementara usulan pembiayaan diproses oleh pihak manajemen BMT (merupakan tugas dan wewenangnya), pihak manajemen BMT mengajukan permohonan analisis kredit, seperti penilaian kelayakan usaha, penilaian jaminan, permohonan informasi calon peminjam, dan analisis yuridis ke bagian administrasi pembiayaan dan hukum.

3. Persetujuan komite pembiayaan BMT

Bila seluruh proses oleh pegawai harian telah selesai dilakukan, dokumen yang berisi usulan pembiayaan tersebut diserahkan ke bagian administrasi pembiayaan untuk diperiksa kelengkapannya. Selanjutnya dimintakan persetujuan komite pembiayaan.

4. Pengikatan pembiayaan

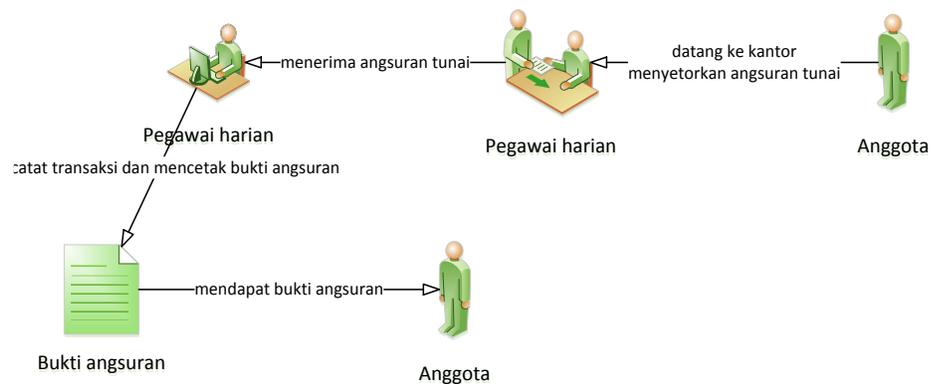
Setelah usulan pembiayaan tersebut mendapat persetujuan dari komite pembiayaan, tahap selanjutnya adalah mempersiapkan pengikatan pembiayaan (akad pembiayaan). Sebelum dilakukan pengikatan, semua dokumen asli dan dokumen jaminan harus telah diterima.

5. Pencairan dana

Setelah dilakukan pengikatan pembiayaan, proses pencairan dana dapat dilakukan, dengan terlebih dahulu dilakukan verifikasi tanda tangan calon peminjam.

3.5 Analisis Transaksi Angsuran

Gambaran transaksi angsuran yang sedang berjalan pada BMT Insan Madani dapat dilihat pada Gambar 3.4.



Gambar 3.4 Proses Transaksi Angsuran

Berikut adalah penjelasan tentang Gambar 3.4:

1. Anggota

Anggota datang ke kantor BMT membawa slip bukti pinjaman dan uang tunai.

2. Pegawai Harian

Pegawai harian menerima slip dan nomor anggota kemudian mencatat transaksi pembayaran angsuran, besar angsuran, dan tanggal pembayaran angsuran.

3. Bukti angsuran

Pegawai harian menerima uang angsuran dari anggota dan mencatat transaksi pada slip bukti angsuran yang kemudian dibawa kembali oleh anggota

3.6 Analisis Kebutuhan Sistem

Berdasarkan Gambar 3.2, Gambar 3.3 dan Gambar 3.4, dapat disimpulkan bahwa kebutuhan untuk sistem pada BMT Insan Madani Kalibayem meliputi:

1. Sistem aplikasi simpanan: Data anggota dan data nominal simpanan diperlukan oleh pengguna dalam melakukan pengelolaan data anggota dan transaksi simpanan.
2. Sistem aplikasi pinjaman: Data anggota, nominal dan tanggal permohonan diperlukan oleh pengguna dalam melakukan pengelolaan data pinjaman dan angsuran.
3. Laporan: Laporan simpanan, pinjaman dan angsuran.

Kebutuhan fungsional sistem yang diperlukan yaitu fungsi yang harus ada dalam sistem yang dikembangkan. Fungsi-fungsi itu antara lain:

1. Sistem dapat digunakan untuk proses pengolahan data barang anggota baru atau update data anggota dan pengolahan data simpanan, pinjaman dan angsuran.
2. Fasilitas *login* bagi pegawai harian agar dapat mengakses sistem.
3. Pegawai harian menggunakan sistem untuk melakukan pengelolaan data anggota, data simpanan, data pinjaman, transaksi pinjaman, dan transaksi angsuran.

3.7 Analisis Kebutuhan Perangkat Lunak

3.7.1 Kebutuhan Proses

Adapun kebutuhan proses dalam mengembangkan aplikasi yaitu :

1. Proses transaksi simpanan, pinjaman dan angsuran.
2. Proses mencatat data anggota baru.
3. Proses menghitung setiap transaksi simpan, pinjam dan angsuran.

3.7.2 Kebutuhan Input

Kebutuhan *input* yang diperlukan dalam aplikasi yaitu:

1. Data anggota.
2. Data transaksi simpanan, pinjaman dan angsuran

3.7.3 Kebutuhan Output

Keluaran yang diharapkan dari aplikasi yaitu berupa laporan bukti simpanan anggota, laporan bukti pinjaman anggota dan laporan transaksi angsuran yang dilakukan oleh anggota peminjam sehingga dapat diketahui siapa saja anggota yang melakukan transaksi berdasarkan tanggal dan total nominal yg masuk dan dipinjam.

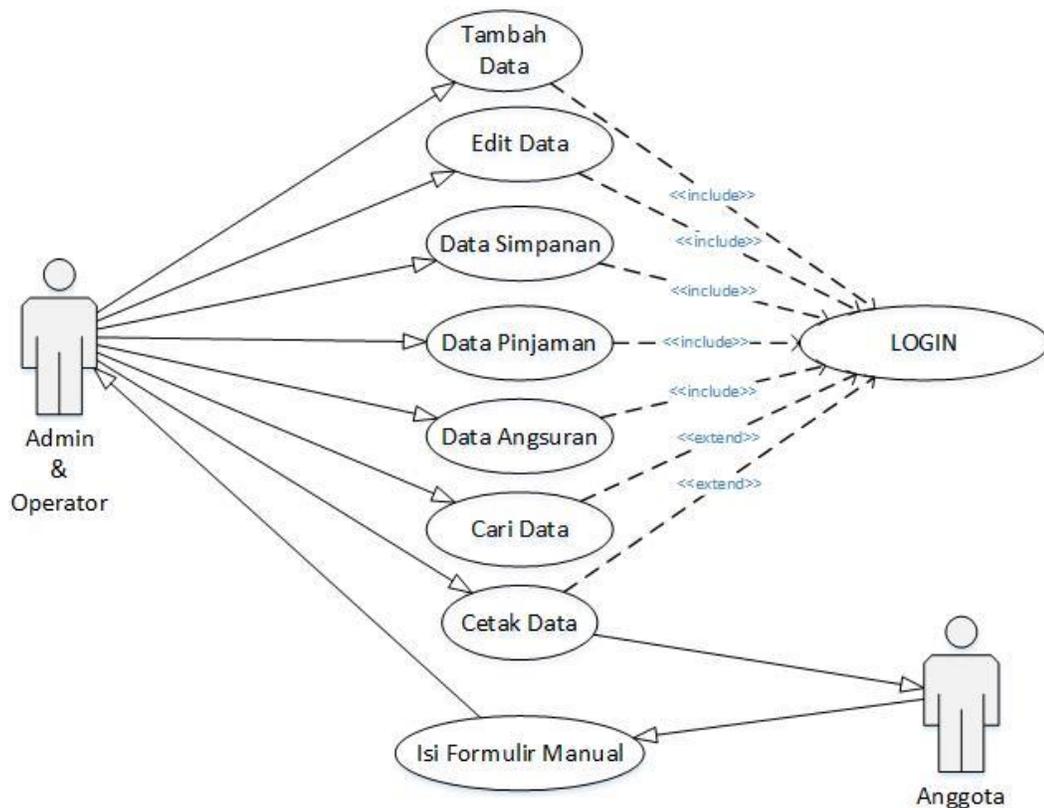
3.8 Rancangan Perangkat Lunak

3.8.1 Rancangan Proses

Metode perancangan yang digunakan dalam aplikasi adalah *United Markup Language* (UML). Seperti yang sudah dijelaskan sebelumnya, model UML yang dipakai dalam pengembangan aplikasi yaitu model *Use Case Diagram*, *Activity Diagram*, dan *Class Diagram*.

3.8.1.1 Use Case Diagram

Gambaran *Use Case Diagram* yang digunakan dalam aplikasi dapat dilihat pada Gambar 3.5 :



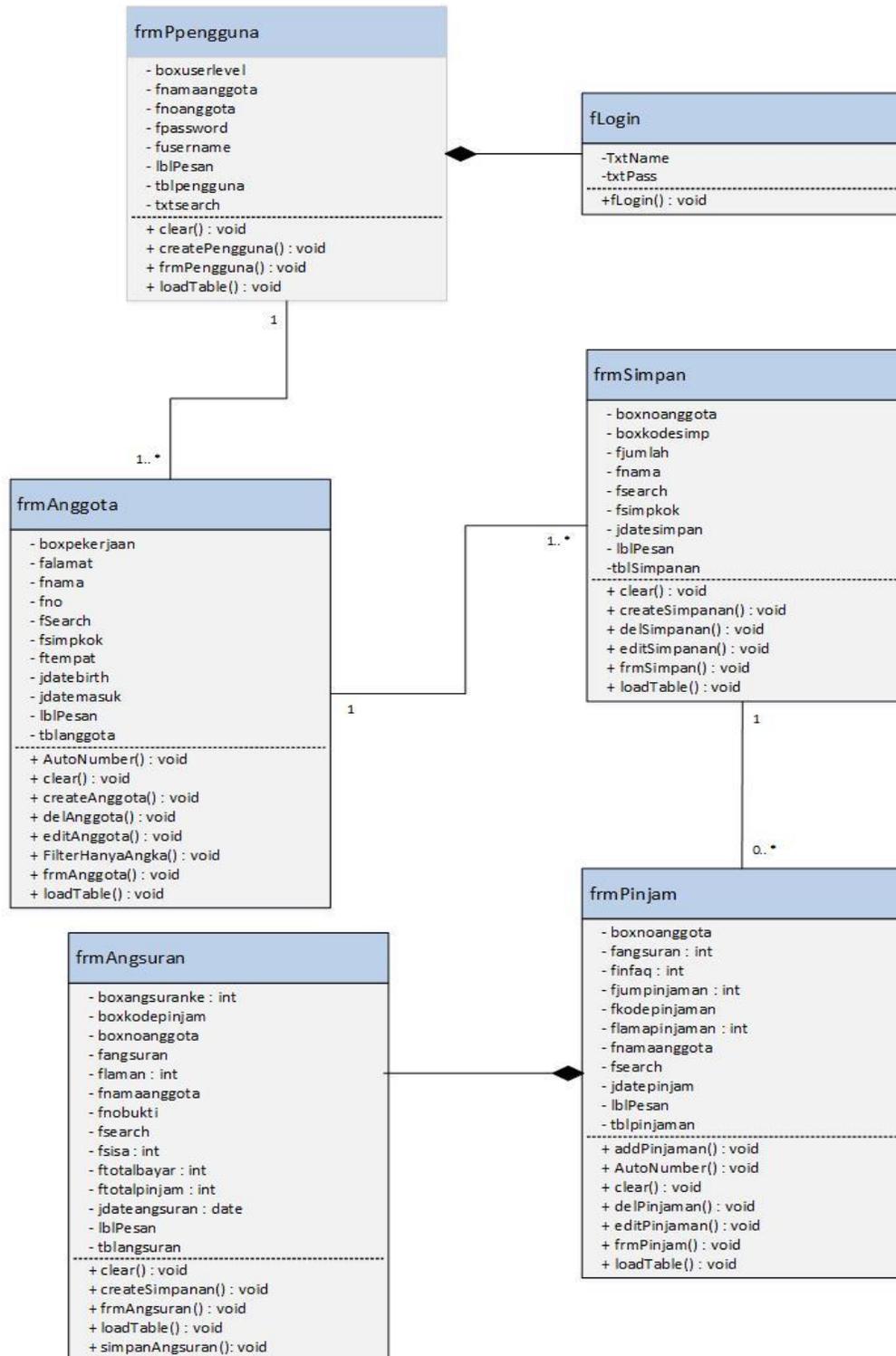
Gambar 3.5 Use Case Diagram

Berikut penjelasan tentang Gambar 3.5:

1. Terdapat 2 aktor pada *use case diagram* yakni admin (pegawai harian) dan anggota.
2. Admin (pegawai harian) berhubungan langsung dengan *input* data anggota yang meliputi *update* data *member*, *input* data simpanan, pinjaman dan angsuran sehingga admin dapat melakukan pengelolaan data.
3. Untuk melakukan transaksi, maka anggota melakukan pengisian data formulir secara manual yang kemudian disetorkan kepada pegawai harian.
4. Admin(pegawai harian) melakukan penyetoran transaksi anggota kepada bendahara.

3.8.1.2 Class Diagram

Gambaran *Class Diagram* yang digunakan dalam aplikasi dapat dilihat pada Gambar 3.6 :



Gambar 3.6 Class Diagram

Berikut adalah penjelasan fungsi-fungsi *Class Diagram* pada Gambar 3.6:

1. frmPengguna
Berfungsi untuk memasukan dan menampilkan data pengguna yang dapat menggunakan aplikasi.
2. fLogin
Berfungsi untuk memvalidasi *username* dan *password* yang terdapat pada kelas frmPengguna.
3. frmAnggota
Berfungsi untuk memasukan dan menampilkan data anggota yang diinputkan oleh pengguna.
4. frmSimpan
Berfungsi untuk memasukan dan menampilkan data simpanan dari anggota yang diinputkan oleh pengguna.
5. frmPinjam
Berfungsi untuk memasukan dan menampilkan data pinjaman dari anggota, juga dapat menghitung jumlah pinjaman dan angsuran yang harus dibayarkan oleh anggota.
6. frmAngsuran
Berfungsi untuk memasukan dan menampilkan data angsuran dari anggota, juga dapat menghitung sisa pinjaman yang belum dibayarkan oleh anggota peminjam.

Berikut adalah penjelasan *Class Diagram* pada Gambar 3.6 :

1. Semua kelas pada diagram digunakan untuk inisialisasi ke dalam *table model* dan basis data aplikasi.
2. Kelas fLogin memiliki *composition* dengan kelas frmPengguna. Maximal satu anggota harus memiliki satu akses *login*.
3. Kelas frmPengguna memiliki asosiasi dengan kelas frmAnggota. Pengguna boleh melakukan input data anggota lebih dari satu kali.

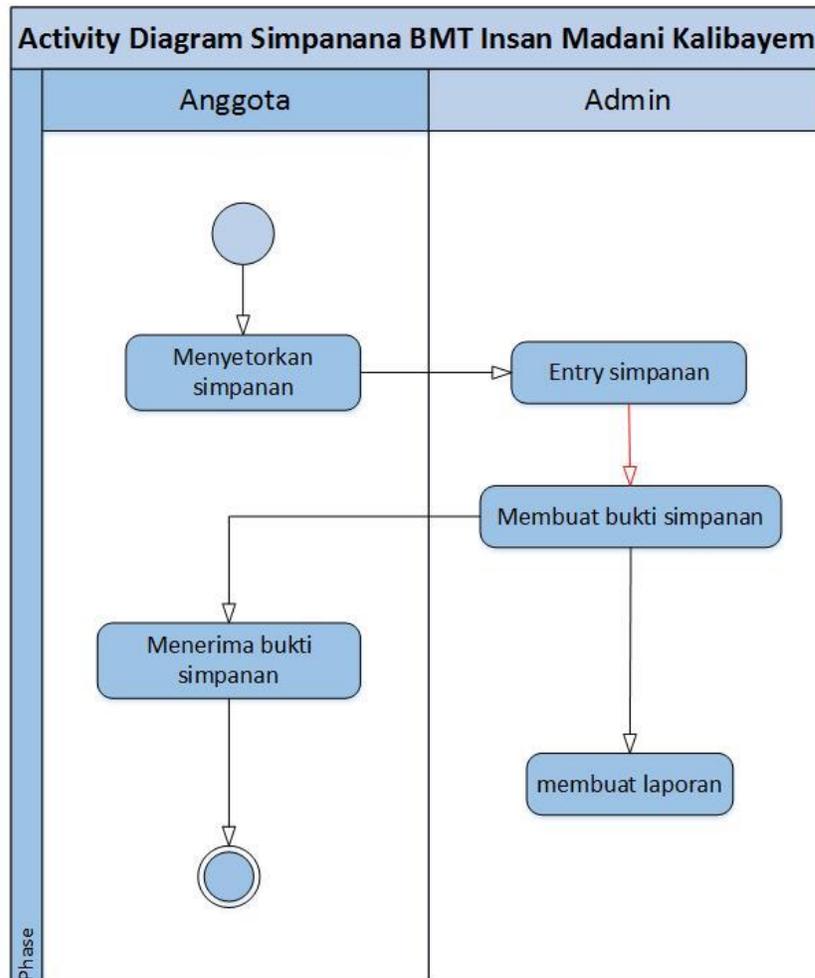
4. Kelas frmAnggota memiliki asosiasi dengan kelas frmSimpan. Anggota boleh melakukan simpanan lebih dari satu kali.
5. Kelas frmSimpan memiliki asosiasi dengan kelas frmPinjam. Anggota yang telah melakukan simpanan boleh tidak melakukan pinjaman atau melakukan pinjaman.
6. Kelas frmPinjam memiliki *composition* dengan kelas frmAngsuran, artinya kelas frmAngsuran merupakan bagian dari kelas frmPinjam. Kelas frmAngsuran tidak dapat berdiri sendiri apabila kelas frmPinjam tidak ada.

3.8.1.3 Activity Diagram

Berdasarkan use case yang telah dibuat sebelumnya maka dapat diperoleh *activity* diagram berdasarkan aktor yang terlibat dalam usecase diagram. *Activity diagram* dalam aplikasi dibagi menjadi dua bagian yaitu *activity* diagram simpanan dan *activity* diagram pinjaman.

a. *Activity Diagram* Simpanan

Gambaran *Activity Diagram* yang digunakan dalam aplikasi dapat dilihat pada Gambar 3.7.



Gambar 3.7 *Activity Diagram* Simpanan Aplikasi

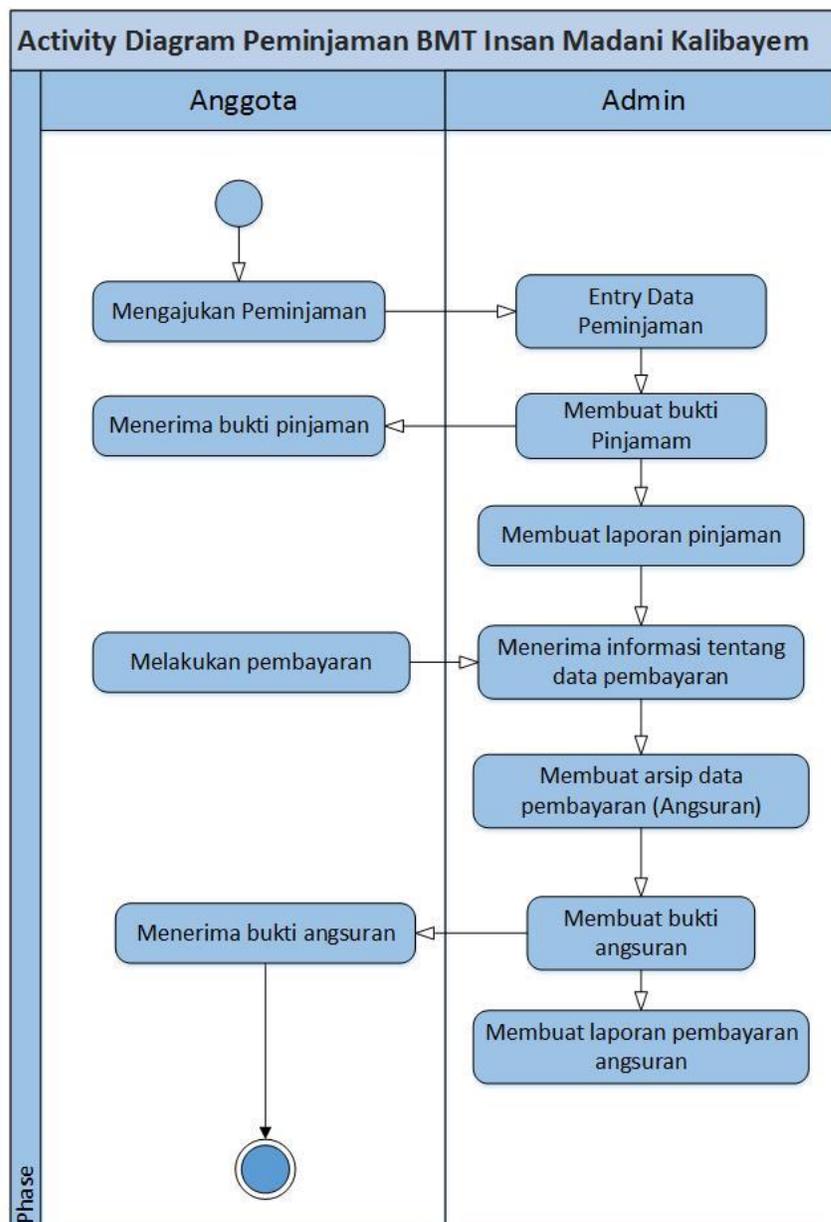
Berikut adalah penjelasan tentang Gambar 3.7:

1. Terdapat alur dari kegiatan transaksi simpanan, yaitu dimulai dari anggota yang membawa bukti nomor anggota dan menyetorkan nominal simpanan kepada admin (pegawai harian).
2. Admin melakukan *entry* data simpanan anggota dan menghitung total simpanan anggota.
3. Anggota mendapatkan bukti simpanan pada slip atau buku bukti simpanan yang terdapat jumlah total simpanan.

4. Admin membuat laporan simpanan anggota yang kemudian disetorkan kepada pihak bendahara.

b. *Activity Diagram* Pinjaman

Gambaran *Activity Diagram* yang digunakan dalam aplikasi dapat dilihat pada Gambar 3.8:



Gambar 3.8. *Activity Diagram* Pinjaman Aplikasi

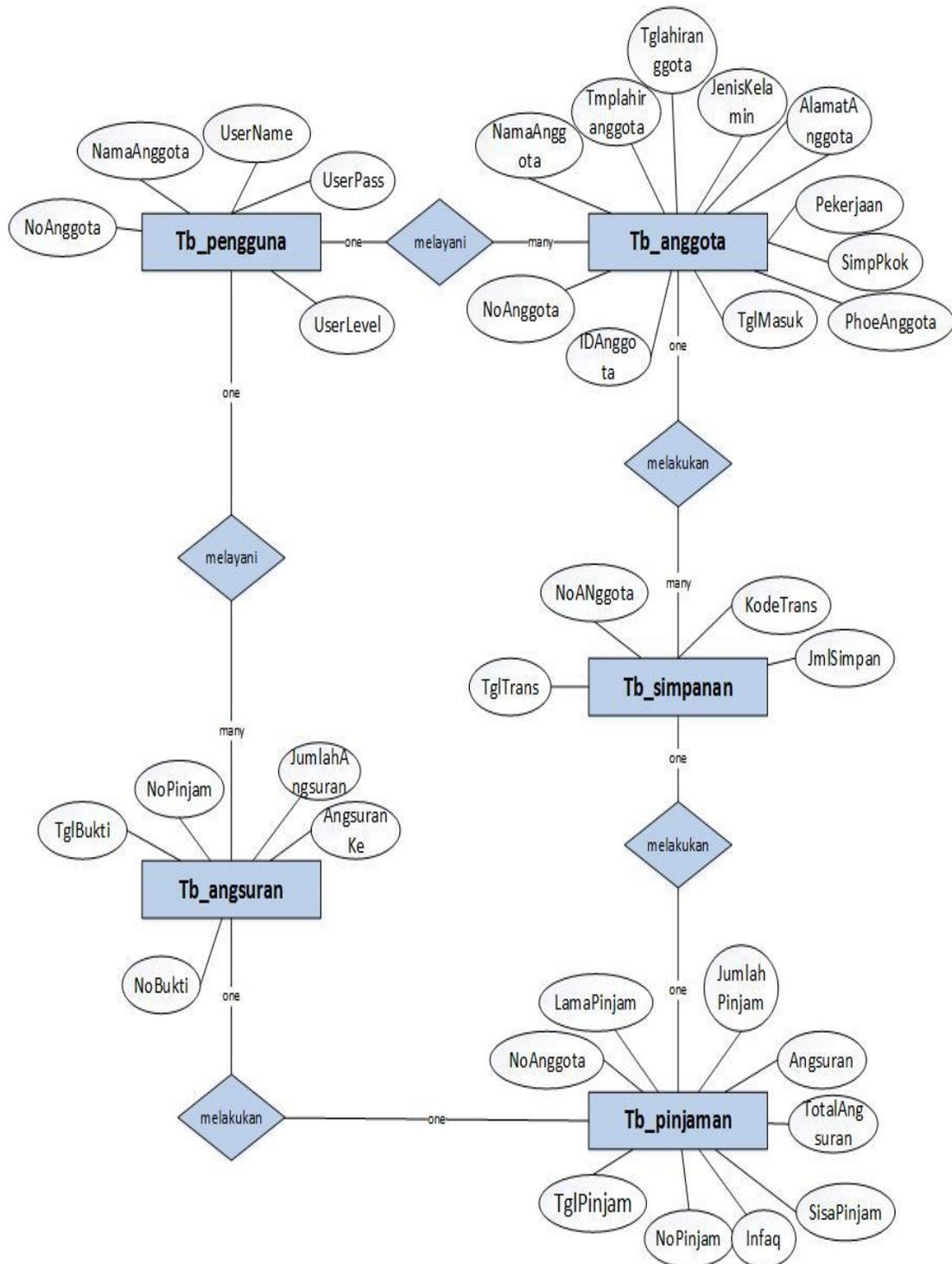
Berikut adalah penjelasan tentang Gambar 3.8:

1. Terdapat alur dari kegiatan peminjaman, yaitu dimulai dari anggota mengajukan permohonan pinjaman.
2. Jika permohonan telah memenuhi syarat dan disetujui maka admin memeriksa nomor anggota dan *entry* data nominal pinjaman.
3. Admin membuat arsip dan slip bukti pinjaman dengan nomor pinjaman yang kemudian akan digunakan anggota pada saat melakukan pengembalian pinjaman secara berangsur.
4. Anggota membawa bukti pinjaman dan melakukan angsuran pinjaman setiap bulan sesuai dengan ketentuan lamanya angsuran.
5. Anggota menyerahkan nominal sesuai ketentuan angsuran dan mendapatkan slip bukti angsuran dan mendapatkan informasi sisa pinjaman.
6. Admin membuat laporan transaksi pinjaman dan angsuran yang kemudian disetorkan kepada pihak bendahara.

3.8.2 Rancangan Basis Data

Untuk pengembangan aplikasi pemodelan basidata yang digunakan adalah ER Diagram. ER Diagram merupakan diagram yang digunakan untuk merancang basis data dan memperlihatkan relasi antar entitas atau objek dengan atributnya.

Tujuan ER Diagram dalam pengembangan aplikasi adalah dapat memberikan gambaran umum tentang sistem yang dikembangkan sehingga memudahkan dalam merancang basis data. Gambaran ER Diagram yang digunakan dalam aplikasi dapat dilihat pada Gambar 3.9.



Gambar 3.9 Entity Relatinship Diagram (ERD)

Berikut penjelasan tentang Gambar 3.9 :

Terdapat 5 entitas yang masing-masing entitas memiliki atribut dan relasi.

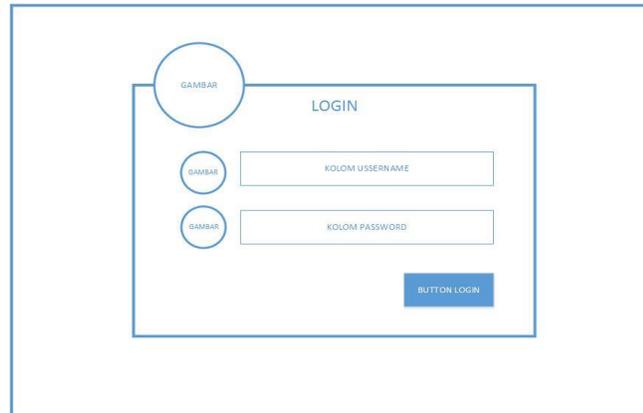
1. Entitas *tb_pengguna* memiliki relasi *one-to-many* dengan *tb_anggota*, artinya admin dapat melakukan banyak pelayanan transaksi langsung dengan anggota.
2. Entitas *tb_anggota* memiliki relasi *one-to-many* dengan *tb_simpanan*, artinya anggota dapat melakukan beberapa jenis simpanan dalam satu kali transaksi.
3. Entitas *tb_simpanan* memiliki relasi *one-to-one* dengan *tb_pinjaman*, artinya anggota yang telah melakukan transaksi simpanan dapat melakukan pinjaman.
4. Entitas *tb_pinjaman* memiliki relasi *one-to-many* dengan *tb_angsuran*, artinya anggota yang telah melakukan pinjaman dapat melakukan banyak angsuran.
5. Entitas *tb_angsuran* memiliki relasi *many-to-one* dengan *tb_pengguna*, artinya semua transaksi angsuran hanya dapat dilayani oleh admin.

3.8.3 Rancangan Antarmuka (User Interface)

User Interface sangat penting dalam suatu aplikasi karena merupakan bagian dari perangkat lunak yang menjadi sarana komunikasi antar user dengan sistem serta dapat membantu user dalam melakukan aktivitasnya.

3.8.3.1 Rancangan Antarmuka Halaman Login

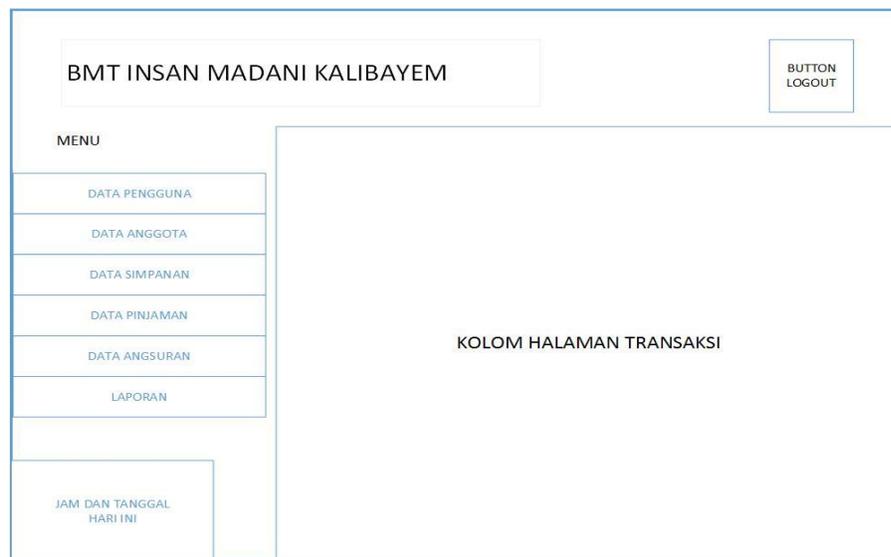
Rancangan antarmuka halaman *login* digunakan oleh pihak manajemen BMT untuk dapat masuk ke halaman utama aplikasi. Gambaran rancangan antarmuka menu login dapat dilihat pada Gambar 3.10 :



Gambar 3.10 Rancangan Antarmuka Halaman *Login*

3.8.3.2 Rancangan Antarmuka Halaman Utama

Rancangan antarmuka halaman utama merupakan tampilan utama dari aplikasi yang dapat dibuka oleh pihak manajemen. Gambaran rancangan antarmuka halaman utama dapat dilihat pada Gambar 3.11.



Gambar 3.11 Rancangan Antarmuka Halaman Utama

Pada Gambar 3.11 terdapat menu-menu yang dapat dibuka sesuai dengan pengguna. Jika pengguna sudah berhasil login, semua menu akan aktif. Fungsi setiap menu dapat dilihat pada Tabel 3.1.

Tabel 3.1 Penjelasan Antarmuka Halaman Utama

NO	Menu	Keterangan
1	<i>Logout</i>	Menu <i>logout</i> berfungsi untuk keluar dari aplikasi dan kembali ke halaman <i>login</i>
2	Data Pengguna	Menu data pengguna berfungsi untuk menampilkan halaman data pengguna.
3	Data Anggota	Menu data anggota berfungsi untuk menampilkan halaman data anggota.
4	Data Simpanan	Menu data simpanan berfungsi untuk menampilkan halaman data simpanan.
5	Data Pinjaman	Menu data pinjaman berfungsi untuk menampilkan halaman data pinjaman.
6	Data Angsuran	Menu data angsuran berfungsi untuk menampilkan halaman data angsuran.
7	laporan	Menu data laporan berfungsi untuk menampilkan halaman data laporan.

3.8.3.3 Rancangan Antarmuka Halaman Data Pengguna

Rancangan antarmuka halaman data pengguna dan ubah data *user* merupakan halaman yang berfungsi untuk memasukkan dan mengubah data *user* dimulai dari ID *user*, nama lengkap *user*, *username*, password, dan level. Gambaran rancangan antarmuka halaman tambah dan ubah data *user* dapat dilihat pada Gambar 3.12.

DATA PENGGUNA

Kode Pegawai

Nama Pegawai

Username

Password

Level

Button Tambah Button Simpan Button Refresh

CARI

NoPegawai	Username	UserLevel

Gambar 3.12 Rancangan Antarmuka Halaman Data Pengguna

3.8.3.4 Rancangan Antarmuka Halaman Data Anggota

Rancangan antarmuka halaman data anggota merupakan tampilan yang menampilkan seluruh data anggota serta dapat melakukan penambahan data anggota dimulai dari penomoran anggota otomatis oleh aplikasi, nama dan data diri anggota, simpanan pokok yang harus disetorkan oleh anggota dan tanggal masuk menjadi anggota BMT, serta dapat melakukan edit data dan menghapus data anggota. Gambaran rancangan antarmuka halaman data anggota dapat dilihat pada Gambar 3.13.

DATA ANGGOTA

No Anggota No Telepon

Nama Anggota Pekerjaan

Tempat, Tgl Lahir Simp. Pokok

Jenis Kelamin Tgl Masuk

Alamat

Button Tambah Button Edit Button Hapus Button Refresh

CARI

NoAnggota	NamaAnggota	TmplLahirAnggota	TglLahirAnggota	JenisKelamin	AlamatAnggota	PhoneAnggota	SimpPokok	TglMasuk	Pekerjaan

Gambar 3.13 Rancangan Antarmuka Halaman Data Anggota

3.8.3.5 Rancangan Antarmuka Halaman Data Simpanan

Rancangan antarmuka halaman data simpanan merupakan tampilan yang menampilkan seluruh data simpanan anggota serta dapat melakukan pengelolaan transaksi simpanan anggota dan dapat mencatat simpanan pokok, simpanan wajib dan simpanan sukarela serta dapat mengedit dan menghapus data simpanan. Gambaran rancangan antarmuka halaman data simpanan dapat dilihat pada Gambar 3.14.

NoPinjam	NoTrans	TglTrans	Wajib	Sukarela

Gambar 3.14 Rancangan Antarmuka Halaman Data Simpanan

3.8.3.6 Rancangan Antarmuka Halaman Data Pinjaman

Rancangan antarmuka halaman data pinjaman merupakan tampilan yang menampilkan seluruh data pinjaman anggota serta dapat melakukan pengelolaan transaksi pinjaman anggota, dapat mencatat jumlah pinjaman, *infaq* dan lamanya angsuran yang kemudian sistem secara otomatis menampilkan jumlah angsuran setiap bulan, serta dapat mengedit dan menghapus data pinjaman. Gambaran rancangan antarmuka halaman data pinjaman dapat dilihat pada Gambar 3.15:

DATA PINJAMAN

No Anggota Pilih no anggota Jumlah Pinjaman

Nama Anggota Lama Pinjaman Bulan

Kode Pinjaman Infaq

Tgl Pinjaman Angsuran

NoPinjam	NoAnggota	TglPinjam	LamaPinjam	Infaq	JumlahPinjam	Angsuran	TotalAngsuran	SisaAngsuran

Gambar 3.15 Rancangan Antarmuka Halaman Data Pinjaman

3.9.3.7 Rancangan Antarmuka Halaman Data Angsuran

Rancangan antarmuka halaman data angsuran merupakan tampilan yang menampilkan seluruh data angsuran anggota dan sisa angsurannya serta dapat melakukan pengelolaan transaksi simpanan anggota, dapat mencatat, mengedit dan menghapus data simpanan. Gambaran rancangan antarmuka halaman data angsuran dapat dilihat pada Gambar 3.16:

DATA ANGSURAN

Kode Pinjam No Bukti

Tgl Pinjam Tgl Angsur

No Anggota Total Pinjam

Nama Anggota Total Bayar

Lama Pinjaman Angsuran ke

Angsuran Sisa

NoBukti	TglBukti	NoPinjam	JumlahAngsuran	AngsuranKe

Gambar 3.16 Rancangan Antarmuka Halaman Data Angsuran

3.8.3.8 Rancangan Antarmuka Halaman Laporan

Rancangan antarmuka halaman laporan merupakan halaman yang berfungsi untuk menampilkan pilihan transaksi yang akan ditampilkan laporannya berdasarkan bulan dan tahun. Gambaran rancangan antarmuka halaman laporan dapat dilihat pada Gambar 3.17.



Gambar 3.17 Rancangan Antarmuka Halaman Laporan

3.8.3.9 Rancangan Antarmuka Halaman Laporan Pilih Bulan dan Tahun

Rancangan antarmuka halaman laporan pilih bulan dan tahun berfungsi untuk memilih periode laporan transaksi yang akan ditampilkan dan dicetak. *Button* Tampilkan berfungsi untuk menampilkan data ketika kolom bulan dan tahun sudah terisi. Gambaran rancangan antarmuka halaman laporan pilih bulan dan tahun dapat dilihat pada Gambar 3.18.



Gambar 3.18 Rancangan Antarmuka Halaman Laporan Pilih Bulan dan Tahun

3.8.3.10 Rancangan Antarmuka Halaman Laporan Transaksi Perbulan

Rancangan antarmuka halaman laporan transaksi perbulan merupakan halaman yang berfungsi untuk menampilkan laporan bulanan dari transaksi simpanan, pinjaman dan angsuran. Gambaran rancangan antarmuka halaman laporan transaksi perbulan dapat dilihat pada Gambar 3.19.



Gambar 3.19 Rancangan Antarmuka Halaman Laporan Transaksi Perbulan

3.9 Pengembangan Sistem

Dalam aplikasi, terdapat dua user yaitu operator (pegawai harian) dan admin, namun belum ada perbedaan hak akses diantara kedua *user* tersebut dan karena untuk sementara hanya pihak manajemen saja yang dapat menggunakan aplikasi yang berperan penting dalam pengolahan data anggota, transaksi simpanan, transaksi pinjaman dan transaksi angsuran. Berikut beberapa prosedur yang dilakukan dalam pengembangan sistem:

1. Menyiapkan perangkat keras dan perangkat lunak pendukung yang dibutuhkan dan menyiapkan kebutuhan sistem dalam aplikasi.
2. Merancang *User Interface* aplikasi sehingga diharapkan dapat memudahkan pihak manajemen dalam menggunakan aplikasi.
3. Membuat basis data sesuai dengan kebutuhan aplikasi.
4. Membuat sistem ke dalam bahasa pemrograman melalui proses *coding*.
5. Membuat sistem yang dapat menampilkan dan mencetak laporan setiap transaksi.

3.10 Metode Pengujian

Pengujian perangkat lunak merupakan suatu kegiatan yang dilakukan untuk memperoleh informasi serta mengevaluasi mengenai kualitas dari produk atau layanan yang sedang diuji. Tujuan pengujian dalam pengembangan aplikasi adalah untuk dapat memenuhi kebutuhan yang diperlukan oleh pihak manajemen dengan mendasari pada rancangan dan pengembangan perangkat lunak.

Metode pengujian yang dipakai dalam pengembangan aplikasi adalah *black box testing*. *Black box testing* atau tes fungsional adalah pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak yang dikembangkan.

Pengujian program dalam pengembangan aplikasi dilakukan oleh pengembang dan pihak manajemen yang terlibat sebagai dengan memberikan *input* (data pengguna, data anggota, data transaksi simpanan, data transaksi pinjaman dan data transaksi angsuran) tertentu dan melihat hasil (laporan) yang didapatkan dari input. Beberapa hal yang diujikan dalam sistem aplikasi yaitu apakah sesuai dengan tujuan awal dikembangkan sistem, tujuan pengembangan sistem adalah:

1. Aplikasi dapat memberikan informasi jumlah simpanan, pinjaman dan angsuran yang harus dibayar dengan berapa kali angsuran. Sistem juga dapat mencatat transaksi simpanan dan pinjaman yang sedang berlangsung.
2. Sistem secara otomatis dapat menghitung dan menampilkan total angsuran dan sisa pinjaman anggota. Sistem juga dapat mencatat transaksi angsuran yang sedang berlangsung.
3. Aplikasi dapat menampilkan dan mencetak bukti transaksi simpanan, pinjaman dan angsuran setiap anggota.
4. Aplikasi dapat menampilkan dan mencetak laporan bulanan yang diperlukan seperti laporan data anggota, laporan transaksi simpanan, laporan transaksi pinjaman, dan laporan angsuran.