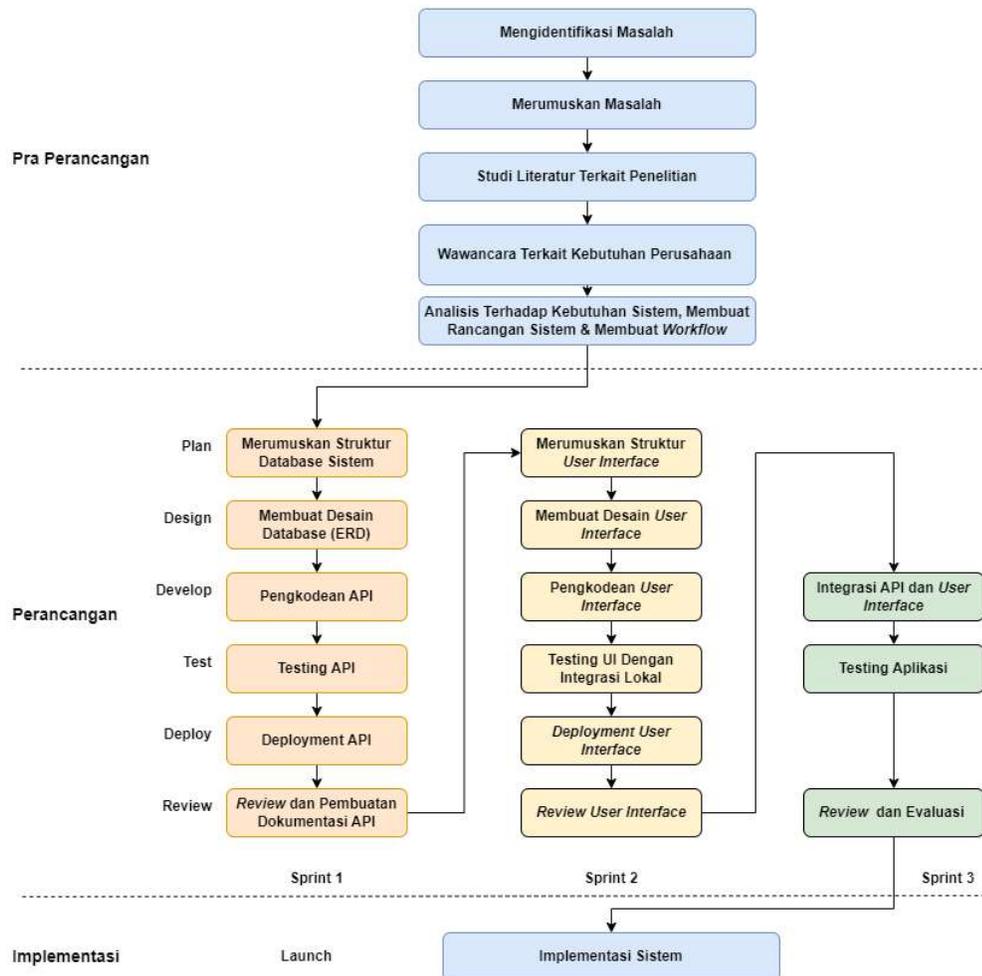


BAB III METODE PENELITIAN

3.1 Desain Penelitian

Desain penelitian merupakan elemen kunci untuk memastikan penelitian berjalan terarah menuju pencapaian tujuan. Berikut adalah dari desain penelitian ini:



Gambar 3.1 Desain Penelitian

Sumber : Data olahan peneliti (2023)

Dari desain penelitian di atas terlihat bahwa penelitian ini memiliki 3 tahapan yaitu:

1. Pra Perancangan

Tahap ini adalah tahap identifikasi dan analisis sebelum aplikasi mulai dibangun, adapun tahapan ini terdiri dari:

- a. Mengidentifikasi masalah, yaitu tahapan pemahaman dari permasalahan yang ditemukan pada objek penelitian, berdasarkan observasi yang dilakukan pada PT Segara Catur Perkasa terdapat masalah pada sistem pencatatan keuangan, dimana PT Segara Catur Perkasa masih melakukan pencatatan transaksi keuangan dengan aplikasi *spreadsheet* sehingga cukup rumit dan mengakibatkan keterlambatan penyusunan laporan keuangan, tidak dapat dikerjakan secara kolaborasi, sangat tergantung pada penyimpanan eksternal yang membahayakan keamanan data.
- b. Merumuskan masalah melibatkan proses menguraikan dan menentukan secara jelas masalah yang akan dipecahkan atau diteliti. Adapun rumusan masalah penelitian ini adalah bagaimana merancang *REST API* untuk sistem akuntansi berbasis *web* dengan menggunakan metode pengembangan *Agile Software Development*.
- c. Studi literatur terkait penelitian, yaitu meninjau dan melakukan analisis terhadap literatur atau karya tulis yang berkaitan dengan penelitian.
- d. Observasi dan wawancara, dilakukan untuk mendapatkan kebutuhan-kebutuhan *user* akan sebuah sistem untuk memecahkan permasalahan yang ada.

- e. Analisis kebutuhan sistem, merancang sistem dan membuat *workflow*. Pada tahap ini, merancang sistem yang akan dikembangkan serta membuat *workflow* (alur kerja) dari sistem tersebut berdasarkan analisis kebutuhan saat wawancara.

2. Perancangan

Perancangan merupakan proses dimana aplikasi mulai dikembangkan, secara garis besar proses ini memiliki 6 tahapan yaitu *plan*, *design*, *develop*, *test*, *deploy*, dan *review* yang terjadi selama 3 periode kerja (*sprint*), yaitu:

a. *Sprint* 1

Pada periode kerja ini pengembangan berfokus pada inti sistem atau desain dari *API* yang akan dibangun. Periode kerja ini dimulai dengan merumuskan struktur dari *database* yang dideskripsikan dalam bentuk ERD (*Entity Relationship Diagram*), kemudian dilanjutkan dengan pengkodean untuk membangun *API* berdasarkan desain sistem yang dibuat, setelah pengkodean selesai maka tahap berikutnya adalah uji coba (*testing*) *API* dengan menggunakan Postman.

Jika pada saat pengujian ditemukan *endpoint* yang tidak bekerja dengan baik maka dilakukan pengkodean ulang untuk memperbaiki *endpoint* tersebut (*debugging*), setelah memastikan semua *endpoint* dapat digunakan maka dapat dilanjutkan ke tahap berikutnya yaitu *deployment* agar *API* dapat digunakan melalui jaringan internet. Tahap terakhir dari *sprint* ini adalah

review dan membuat dokumentasi *API* agar pada *sprint* berikutnya penggunaan *API* menjadi lebih mudah.

b. *Sprint 2*

Periode kerja ini berfokus pada tampilan pengguna (*user interface*). Tahapannya dimulai dengan merumuskan struktur dari *user interface* itu sendiri, mulai dari tata letak, pemilihan *color palette*, hingga pengorganisasian menu-menu yang akan dibuat. Setelah perumusan selesai, desain UI dirancang menggunakan Mockflow untuk memudahkan dalam proses pengkodean *user interface*. Setelah semua bagian dari *user interface* selesai dibuat maka dilakukan uji coba integrasi secara lokal dengan data *dummy* untuk memastikan semua komponen UI dapat digunakan untuk mengirim *request* dan menampilkan *response* dari *API*. Jika semua komponen telah diuji maka dapat dilanjutkan ke tahap *deployment* dimana seluruh komponen UI akan dijalankan oleh Vercel supaya dapat digunakan secara *online*.

c. *Sprint 3*

Periode kerja ini berfokus pada integrasi antara *back-end* dan *front-end* yaitu menghubungkan *API* dan UI. Pada tahap ini *API* dan UI yang sudah di-*deploy* akan diuji pada *develop environment* secara *online*, tahap ini juga melibatkan *user* dan pemangku kepentingan yang lain untuk memastikan aplikasi telah sesuai dengan kebutuhan *user*. Setelah semua fungsionalitas diuji maka dapat dilanjutkan ke tahap *review* dan evaluasi untuk mendapatkan keputusan apakah sistem sudah layak diimplementasikan atau masih perlu perbaikan.

3. Implementasi

Implementasi merupakan tahapan akhir pada penelitian ini, yaitu menerapkan sistem tersebut untuk digunakan dalam pencatatan keuangan perusahaan.

3.2 Metode Pengumpulan Data

Menurut (Sugiyono 2018) teknik pengumpulan data merupakan langkah yang sangat krusial dalam suatu penelitian. Proses pengumpulan data dapat dilakukan melalui berbagai metode dan dari berbagai sumber. Terdapat 2 metode yang digunakan untuk mengumpulkan data pada penelitian ini, yaitu:

1. Wawancara

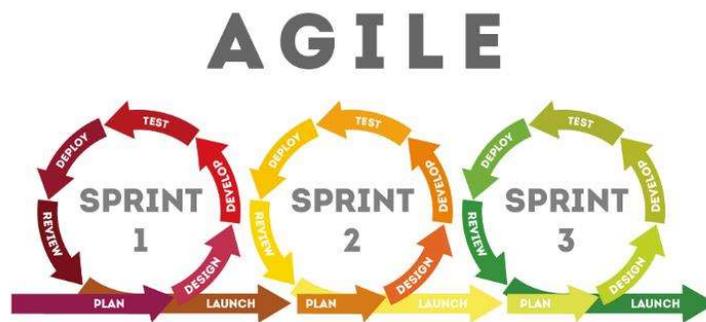
Wawancara dilakukan kepada 2 orang narasumber yang merupakan pegawai perusahaan dengan jabatan admin dan akuntan. Wawancara dilaksanakan untuk mendapatkan informasi terkait kebutuhan sistem, permasalahan yang dialami, dan beberapa informasi lain terkait dengan alur kerja sistem.

2. Observasi

Observasi dilakukan dengan mencermati secara langsung proses pencatatan keuangan dan dokumen terkait transaksi di perusahaan. Tujuannya adalah untuk menentukan rincian yang diperlukan dalam pengembangan sistem yang akan dibangun.

3.3 Metode Perancangan Sistem

Setelah mengumpulkan data dan melakukan analisis terhadap kebutuhan sistem maka langkah berikutnya adalah membuat rancangan sistem. Untuk memastikan bahwa proses perancangan berjalan sesuai tujuan, metode *Agile* digunakan, sebagaimana tergambar pada gambar berikut:



Gambar 3.2 *Agile Sprints*

Sumber : <https://www.radview.com/>

Pada gambar di atas terdapat 3 iterasi atau *sprint* yang mana setiap iterasinya memiliki 6 tahapan yaitu *plan*, *design*, *develop*, *test*, *deploy*, dan *review* dijelaskan setiap tahapan pada perancangan sistem, yaitu:

1. *Plan*

Plan adalah tahap awal pada setiap sprint dimana pada tahap ini dilakukan perencanaan atas apa saja yang menjadi target pada sprint tersebut serta apa saja yang dibutuhkan. Pada *sprint* pertama dilakukan perumusan kebutuhan dan struktur *database* atau model penyimpanan data yang akan digunakan diterapkan pada *API*, begitu pula dengan beberapa aspek lain yang diperlukan saat mengembangkan *API* seperti *middleware* yang diperlukan dan metode keamanan yang digunakan. Pada

plan di *sprint* kedua dilakukan perumusan dari struktur UI yang akan digunakan, mulai dari tata letak, pemilihan *color palette*, hingga pengorganisasian menu-menu fungsi yang akan dibuat, sementara itu tidak ada lagi *plan* di *sprint* ketiga karena *sprint* ketiga berfokus pada integrasi antara *API* dan UI.

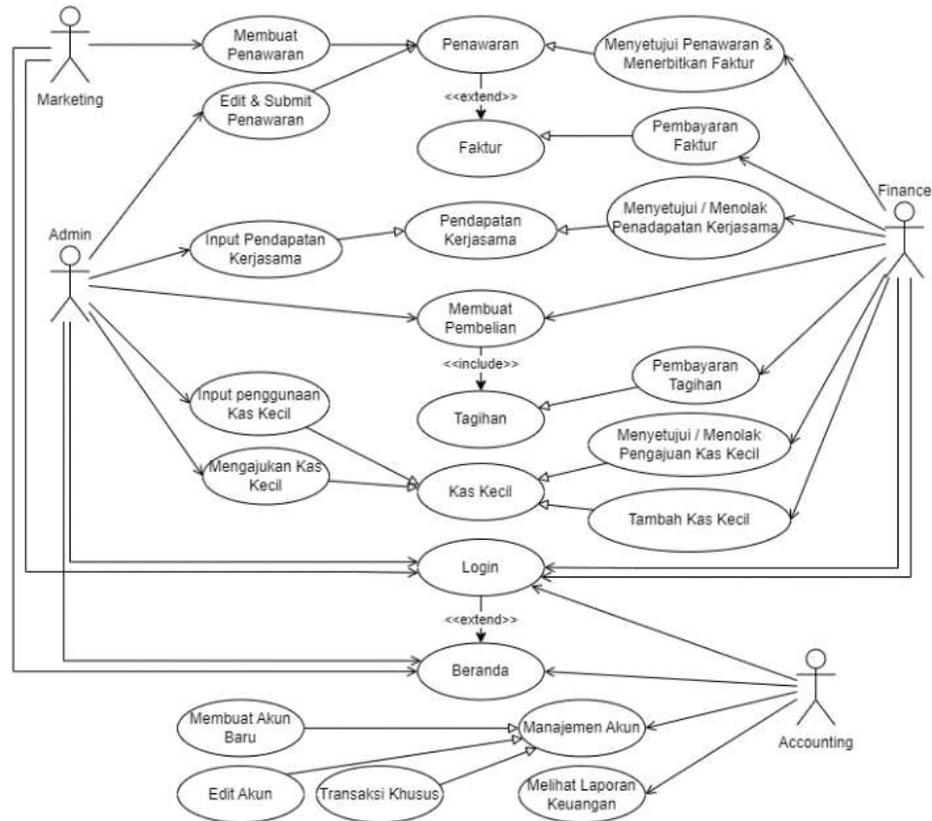
2. Design

Tahap *design* berfokus pada bagaimana memetakan atau mendeskripsikan rencana yang sudah dibuat pada tahap *plan* agar lebih mudah dipahami oleh anggota tim serta para *stakeholders*. Adapun rancangan yang dimodelkan meliputi analisis fungsional sistem, desain antarmuka dan ERD (*Entity Relationship Diagram*) atau rancangan *database*.

a. Analisis fungsional

Peneliti menggunakan UML (*Unified Modelling Language*) dalam memodelkan rancangan fungsional sistem agar lebih mudah dipahami, adapun rancangan fungsional tersebut dimodelkan menggunakan 4 diagram sebagai berikut:

1. Use case diagram



Gambar 3.3 Use case diagram

Sumber : Data olahan peneliti (2023)

Berdasarkan *use case diagram* di atas terdapat 4 aktor yang akan berinteraksi dengan sistem, berikut definisinya:

Tabel 3.1 Definisi aktor

No	Aktor	Definisi
1	<i>Marketing</i>	Aktor ini merepresentasikan staf pemasaran yang bertugas untuk membuat penawaran yang akan dikirimkan kepada calon pengguna jasa.

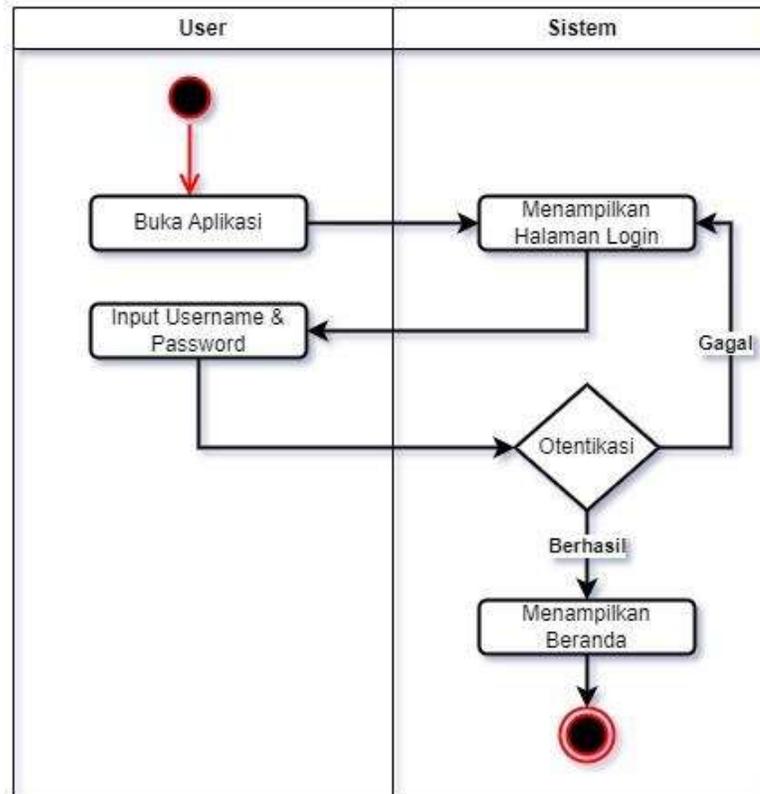
2	<i>Admin</i>	Aktor ini merepresentasikan staf administrasi yang bertugas untuk memproses penawaran yang telah dibuat oleh <i>Marketing</i> dan meneruskannya ke <i>Finance</i> , <i>Admin</i> juga dapat membuat pembelian, mengolah dan mengajukan kas kecil, serta menambahkan pendapatan kerjasama.
3	<i>Finance</i>	Aktor ini adalah representasi dari staf keuangan yang bertugas mengelola transaksi di perusahaan, <i>Finance</i> dapat menyetujui penawaran dan menerbitkan faktur, menambahkan penerimaan pembayaran faktur, menyetujui atau menolak pendapatan kerjasama, melakukan pembelian dan pembayaran tagihan, dan menyetujui atau menolak pengajuan kas kecil.
4	<i>Accounting</i>	Aktor ini adalah representasi dari seorang akuntan dalam perusahaan, tugasnya adalah melakukan pengawasan terhadap keuangan perusahaan dan melaporkannya kepada atasan, <i>Accounting</i> dapat membuat dan mengubah akun serta melihat laporan keuangan perusahaan.

Sumber : Data olahan peneliti (2023)

2. Activity diagram

Activity diagram adalah visualisasi dari alur atau urutan kerja dalam suatu sistem. Berikut adalah *activity diagram* sistem yang dibangun:

a. *Activity diagram Login*

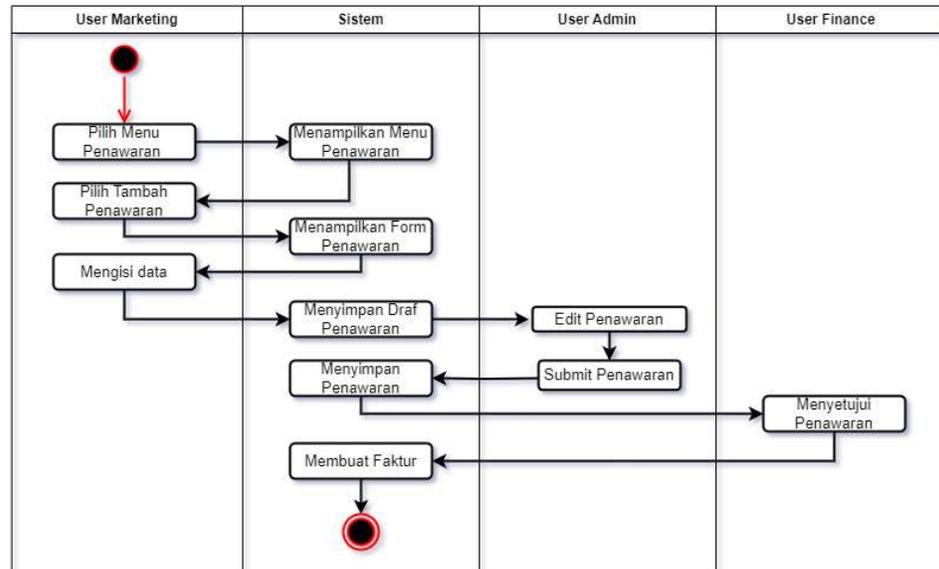


Gambar 3.4 *Activity diagram Login*

Sumber : Data olahan peneliti (2023)

Pertama-tama *user* diharuskan untuk *login* dengan cara membuka aplikasi kemudian mengisi *username* dan *password* untuk *login* dan mengakses sistem, setelah *user* mengirim *username* dan *password* sistem akan melakukan proses otentikasi untuk memastikan bahwa *user* memang memiliki hak untuk mengakses aplikasi, jika otentikasi berhasil maka *user* akan diteruskan ke beranda dan jika gagal maka *user* akan diminta untuk mengulang proses *login*.

b. *Activity diagram* Penawaran

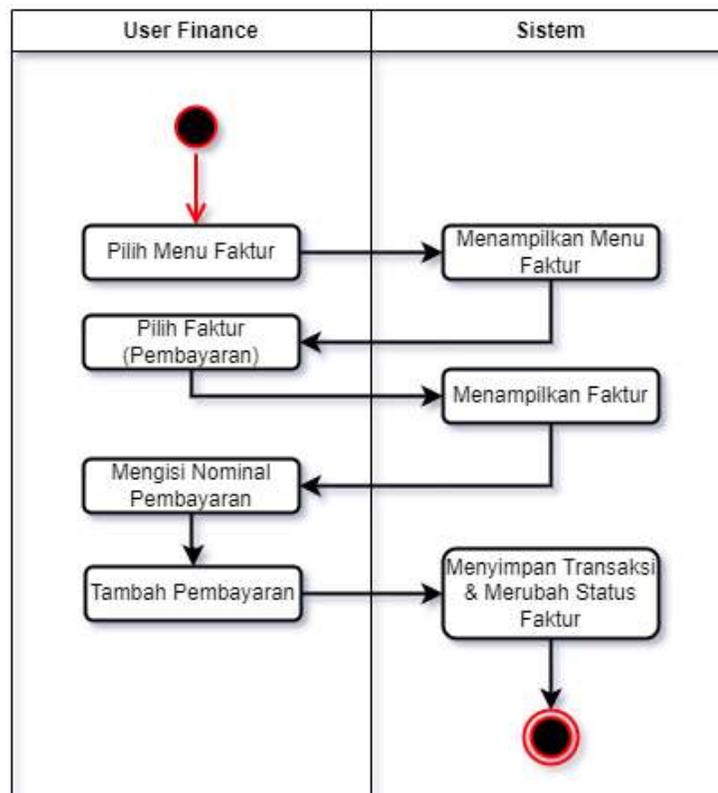


Gambar 3.5 *Activity diagram* Penawaran

Sumber : Data olahan peneliti (2023)

Marketing sudah dalam keadaan *login* dan berada di beranda, sementara *Admin* dan *Finance* berada pada tampilan penawaran. Langkah awal dalam pembuatan penawaran adalah *Marketing* memilih menu penawaran dan sistem akan memperlihatkan menu penawaran, kemudian *Marketing* dapat mengklik “Tambah Penawaran” dan sistem akan memunculkan halaman pengisian penawaran lalu *Marketing* mengisi form dan klik simpan, sistem akan menyimpan draf penawaran, draf tersebut akan diolah oleh *Admin* kemudian di-*submit* dan sistem akan kembali menyimpan form penawaran tersebut. Setelah form di-*submit*, *Finance* dapat menyetujui Penawaran tersebut yang kemudian akan diterbitkan faktur oleh sistem.

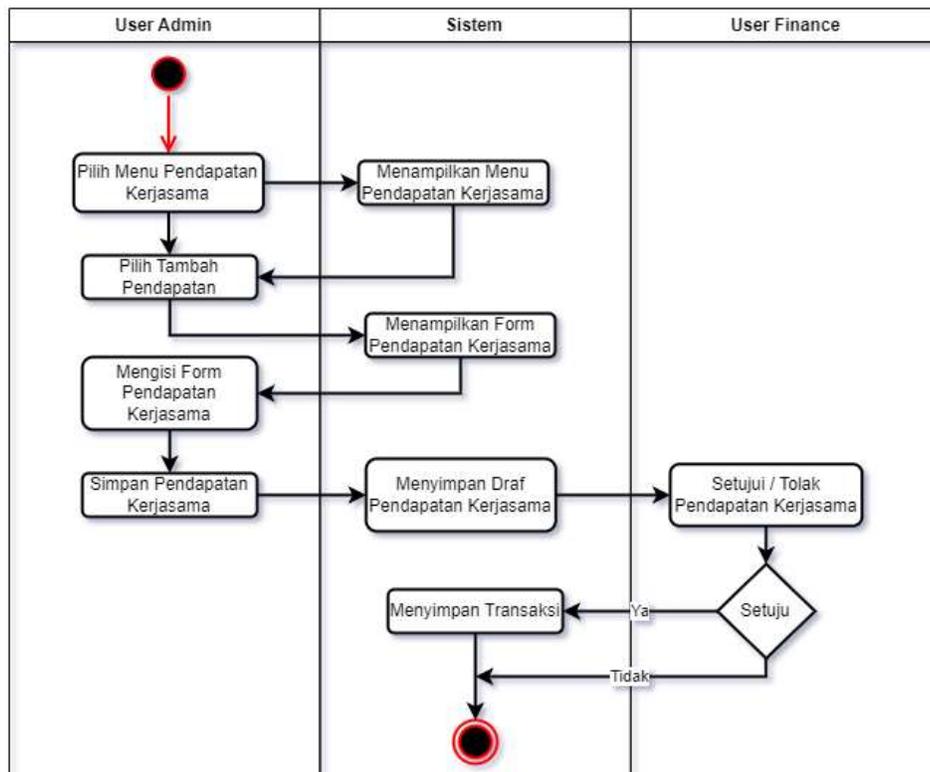
c. *Activity diagram* Pembayaran Faktur



Gambar 3.6 *Activity diagram* pembayaran Faktur

Sumber : Data olahan peneliti (2023)

Finance sudah *login* dan berada di beranda, kemudian memilih menu Faktur dan sistem akan memperlihatkan menu faktur yang berisi daftar faktur, *user* memilih faktur dengan klik “Pembayaran” dan sistem akan menyajikan rincian dari faktur yang tersebut, *user* mengisi nominal pembayaran kemudian klik “Tambah Pembayaran”

d. *Activity diagram* Pendapatan Kerja sama

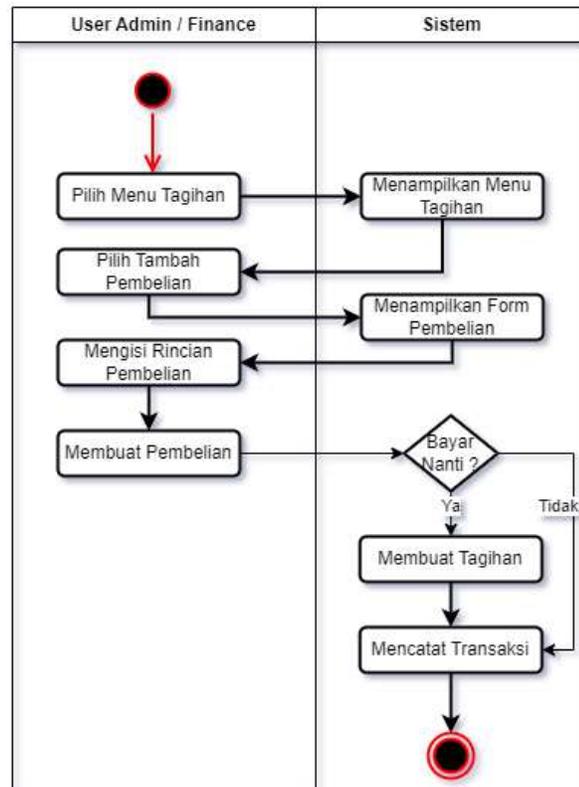
Gambar 3.7 *Activity diagram* Pendapatan Kerjasama

Sumber : Data olahan peneliti (2023)

Admin sudah *login* dan berada di beranda, sementara *Finance* berada di menu Pendapatan Kerjasama, mula-mula *Admin* memilih menu Pendapatan Kerjasama lalu sistem akan merespon dengan menampilkan menu Pendapatan Kerjasama dan daftar pendapatan kerjasama yang telah diajukan sebelumnya, kemudian *user* memilih “Tambah Pendapatan” lalu mengisi data dan klik “Simpan Pendapatan Kerjasama”, sistem akan menyimpan data draf sebagai pengajuan yang

kemudian akan diperiksa oleh *Finance* dan diproses (disetujui / ditolak), jika disetujui maka sistem akan memproses transaksinya.

e. *Activity diagram* Pembelian



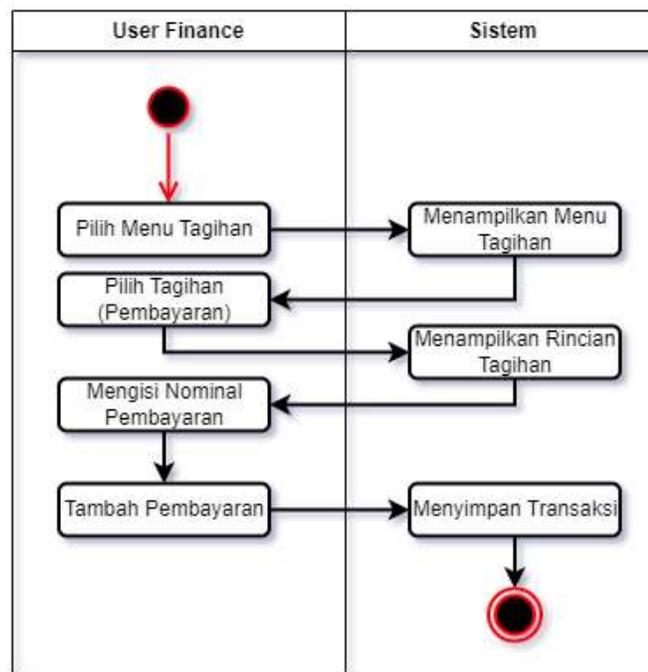
Gambar 3.8 *Activity diagram* Pembelian

Sumber : Data olahan peneliti (2023)

User Admin / Finance sudah dalam keadaan *login* dan berada di beranda, *user* memilih menu Tagihan kemudian sistem akan menunjukkan menu Tagihan, setelah itu *user* dapat mengklik “Tambah Pembelian” dan sistem akan memunculkan form pembelian, kemudian *user* mengisi rincian pembelian dan pilih “Buat Pembelian”, sistem akan memeriksa apakah *user* memilih bayar nanti atau

tidak pada form, jika iya maka sistem akan membut tagihan dan mencatat transaksi sebagai pembelian dengan hutang, jika tidak maka sistem akan mencatat transaksi pembelian.

f. *Activity diagram* Pembayaran Tagihan

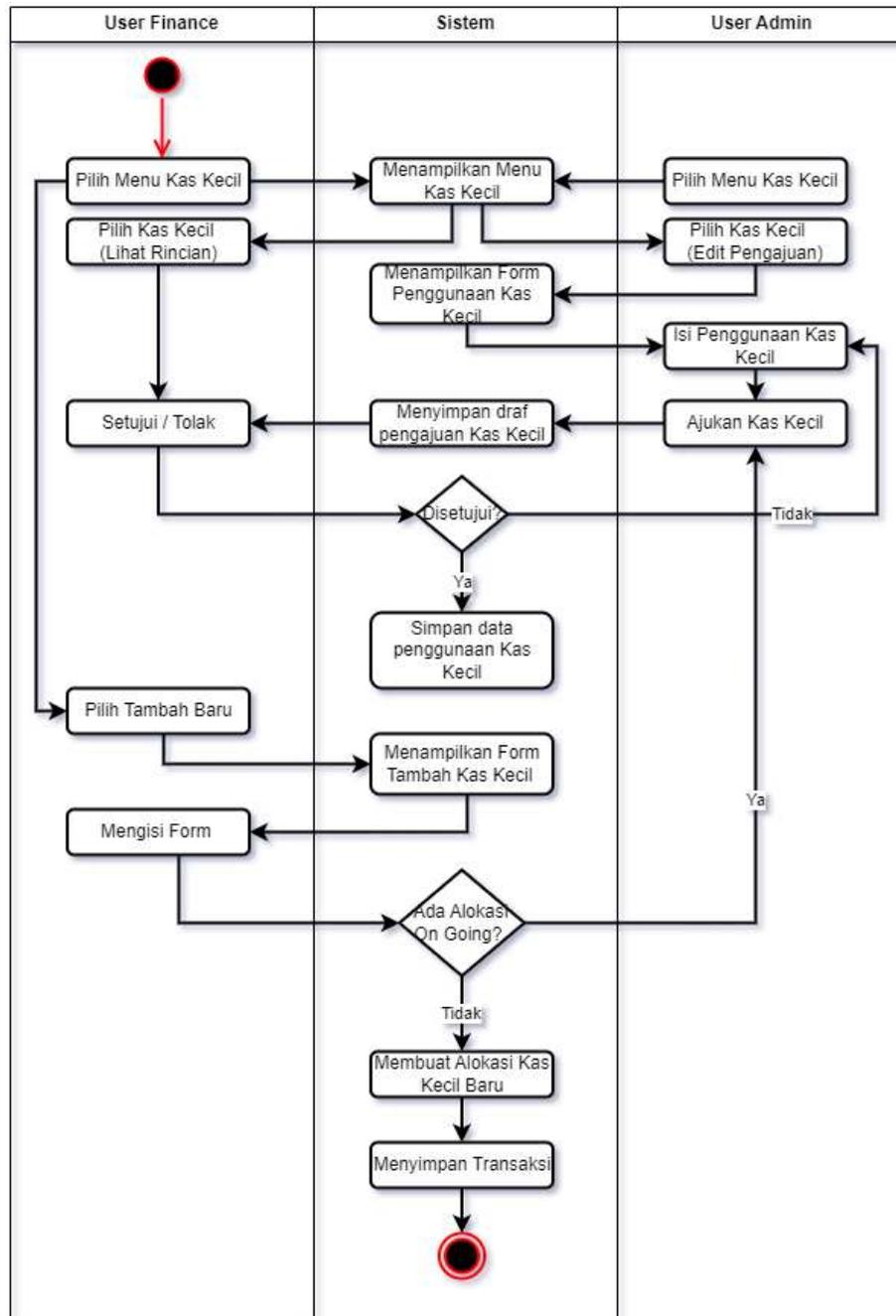


Gambar 3.9 *Activity diagram* Pendapatan Kerjasama

Sumber : Data olahan peneliti (2023)

Finance sudah *login* dan berada di beranda, kemudian memilih menu Tagihan dan sistem akan menunjukkan menu Tagihan, selanjutnya *Finance* memilih tagihan yang hendak dibayarkan dengan klik “Pembayaran”, sistem akan menampilkan rincian dari tagihan, lalu *user* diminta untuk mengisi nominal pembayaran dan klik “Tambah Pembayaran” maka sistem akan menyimpan transaksi tersebut.

g. Activity diagram Kas Kecil



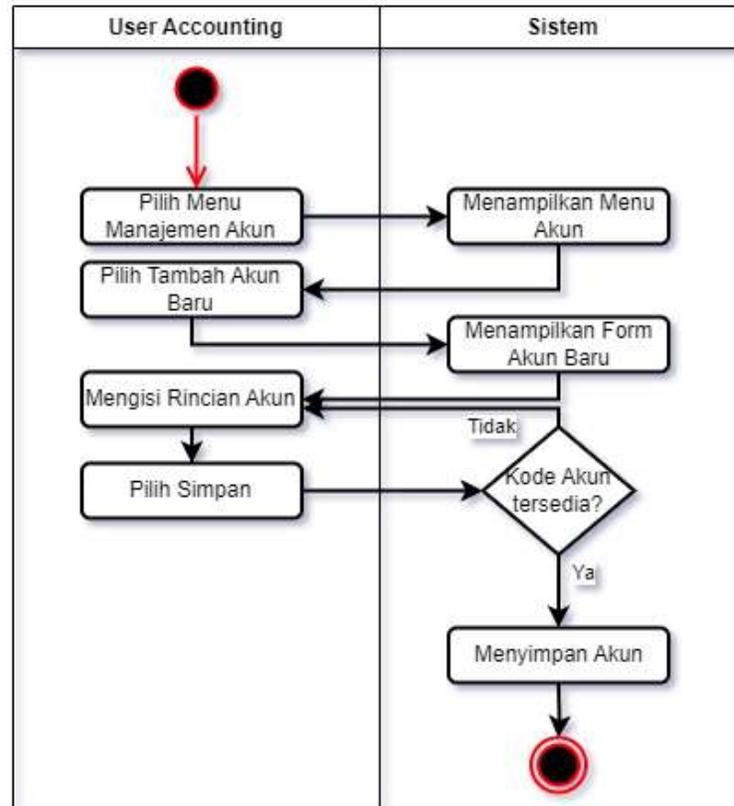
Gambar 3.10 Activity diagram Kas Kecil

Sumber : Data olahan peneliti (2023)

User Finance dan *Admin* sudah *login* dan berada di beranda, pada menu ini *Finance* dapat melakukan 2 hal yaitu menyetujui/menolak pengajuan serta mengisi ulang kas kecil, sedangkan *Admin* dapat melakukan pengisian penggunaan kas kecil dan melakukan pengajuan isi ulang kas kecil.

Pertama-tama *Admin* memilih menu Kas Kecil kemudian sistem akan menampilkan daftar kas kecil, *Admin* memilih kas kecil yang aktif dan lihat rincian untuk menambahkan penggunaan kas kecil, setelah kas kecil tinggal sedikit *Admin* melakukan pengajuan isi ulang dengan klik “Ajukan”, sistem akan merubah status kas kecil tersebut dan menunggu *Finance* untuk melakukan persetujuan,

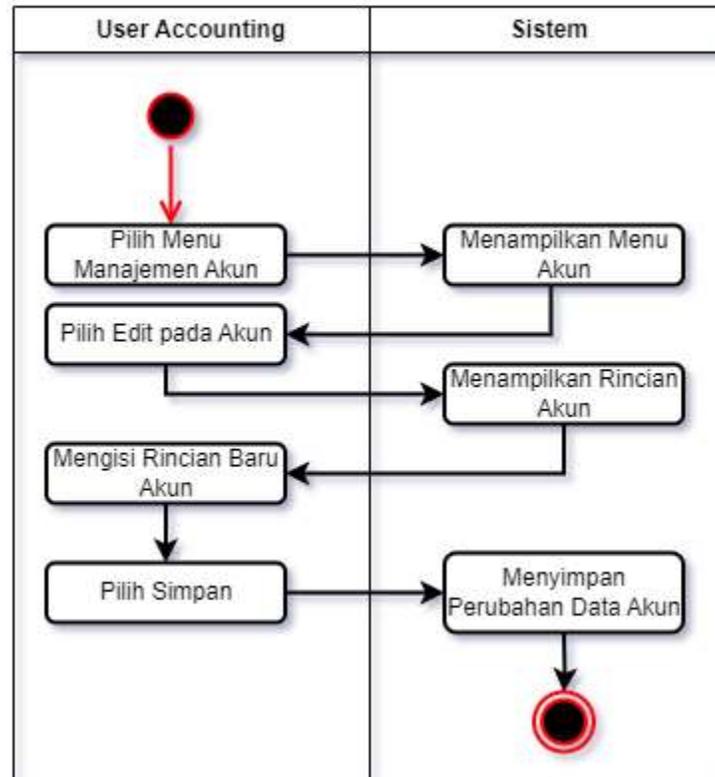
Setelah *Finance* menyetujui pengajuan kas kecil maka sistem akan merubah status kas kecil tersebut dan kini *Finance* bisa membuat alokasi kas kecil baru dengan pilih “Tambah Baru” di menu Kas Kecil, kemudian mengisi form penambahan kas kecil dan klik “Tambah Kas Kecil”, maka sistem akan membuat form penggunaan kas kecil baru dan menyimpan transaksi isi ulang kas kecil.

h. *Activity diagram* Buat Akun Baru

Gambar 3.11 *Activity diagram* Buat Akun Baru

Sumber : Data olahan peneliti (2023)

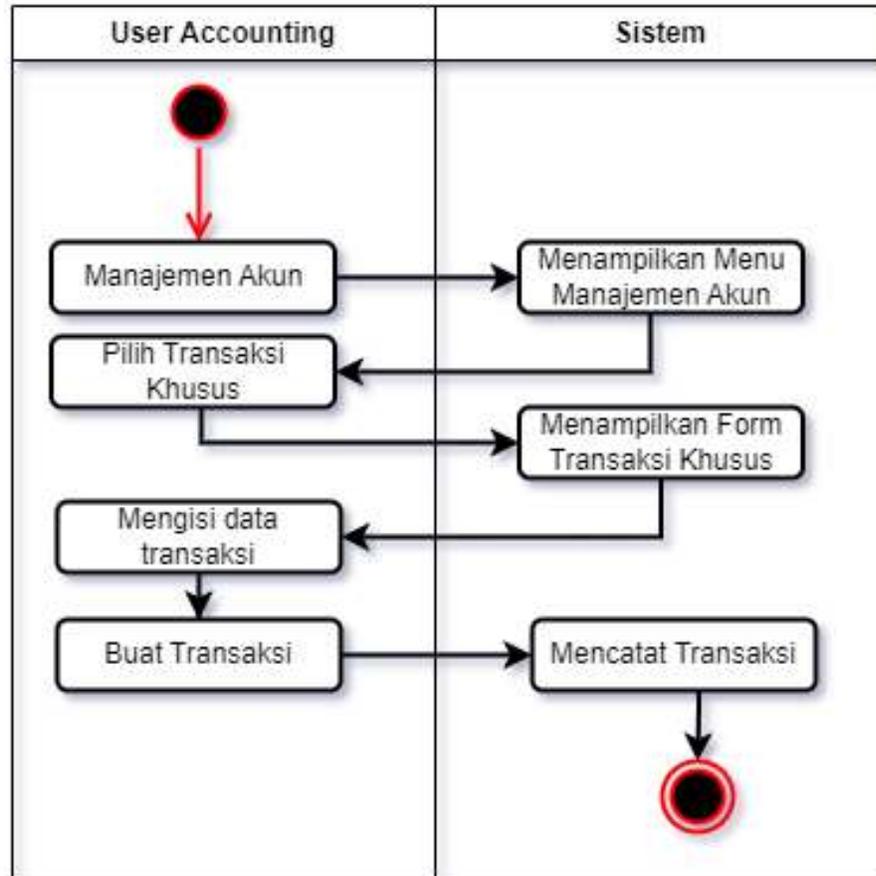
User Accounting sudah *login* dan berada di beranda, kemudian *user* klik Manajemen Akun dan sistem akan menunjukkan menu Akun, *user* memilih Tambah Akun Baru lalu sistem akan memunculkan form untuk menambahkan akun baru dan *user* diminta untuk mengisi rincian akun baru, setelah itu pilih simpan dan sistem akan memeriksa apakah kode akun yang dimasukkan tersedia, jika tidak maka *user* harus mengisi ulang rincian dengan kode akun berbeda, jika masih tersedia maka sistem akan menyimpan akun baru tersebut.

i. *Activity diagram* Ubah Akun

Gambar 3.12 *Activity diagram* Ubah Akun

Sumber : Data olahan peneliti (2023)

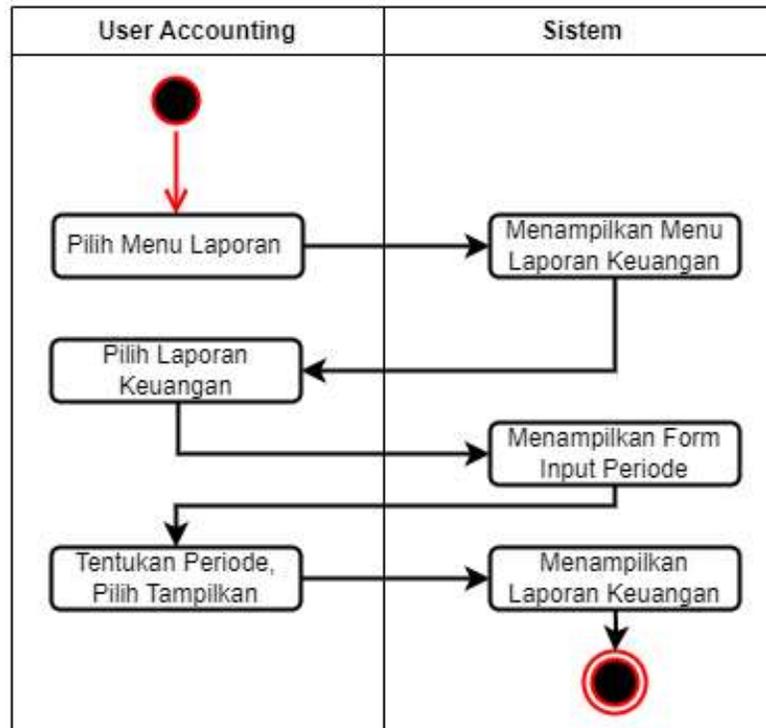
User Accounting sudah dalam keadaan *login* dan berada di beranda, kemudian *user* klik menu Manajemen Akun dan sistem akan memperlihatkan Menu Akun, *user* dapat memilih akun yang akan diubah dengan mengklik “*Edit*” akun tersebut dan sistem akan menyajikan rincian akun tersebut, *user* merubah data yang diinginkan pada rincian akun lalu klik simpan dan sistem akan menyimpan perubahan data pada akun tersebut.

j. *Activity diagram* Transaksi Khusus

Gambar 3.13 *Activity diagram* Transaksi Khusus

Sumber : Data olahan peneliti (2023)

User Accounting sudah *login* dan berada di beranda, kemudian *user* memilih manajemen akun dan sistem akan memperlihatkan menu manajemen akun, pada menu manajemen akun *user* klik transaksi khusus, sistem akan memunculkan form transaksi khusus yang harus di isi oleh *user*, setelah selesai *user* dapat mengklik “Buat Transaksi” dan sistem akan menyimpan transaksi tersebut.

k. *Activity diagram* Laporan Keuangan

Gambar 3.14 *Activity diagram* Laporan Keuangan

Sumber : Data olahan peneliti (2023)

User Accounting sudah *login* dan berada di beranda, kemudian *user* klik menu laporan dan memilih laporan yang ingin dibuka dan sistem akan meminta *user* untuk memasukkan periode yang diinginkan lalu *user* memasukkan periode dan klik tampilkan maka sistem akan memperlihatkan laporan keuangan.

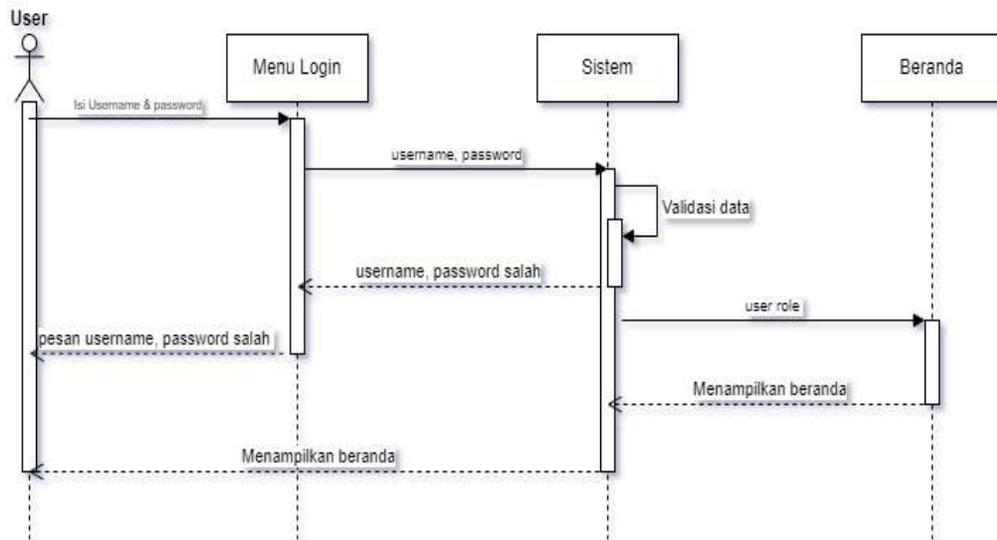
3. *Sequence diagram*

Sequence diagram adalah visualisasi dalam bentuk grafis yang mengilustrasikan perilaku objek dalam suatu *use case* dengan menyajikan detail

waktu hidup objek dan urutan pesan yang dikirim dan diterima oleh objek tersebut.

Berikut adalah *sequence diagram* dari sistem yang dibangun:

a. *Sequence diagram* Login

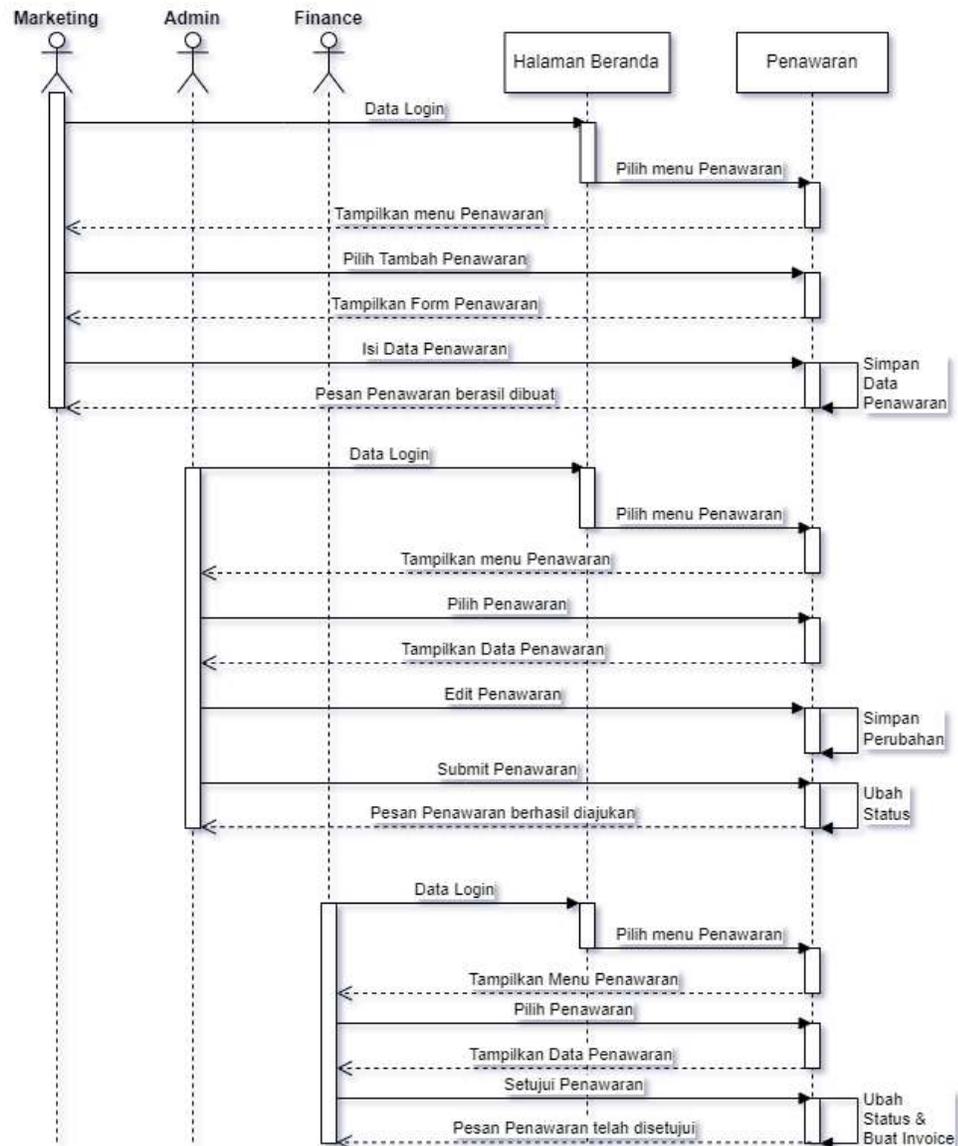


Gambar 3.15 *Sequence diagram* Login

Sumber : Data olahan peneliti (2023)

Ketika *user* membuka aplikasi, sistem akan mengarahkan *user* ke halaman *login* kemudian *user* mengisi *username* dan *password*, setelah klik *login* maka sistem memvalidasi data yang dikirimkan *user*, jika tidak sesuai maka sistem akan mengirimkan pesan *username* atau *password* salah dan *user* diminta untuk mengulang proses *login*, jika sesuai maka *user* akan diteruskan ke beranda dengan role yang telah ditentukan pada akun pengguna.

b. *Sequence diagram* Penawaran



Gambar 3.16 *Sequence diagram* Penawaran

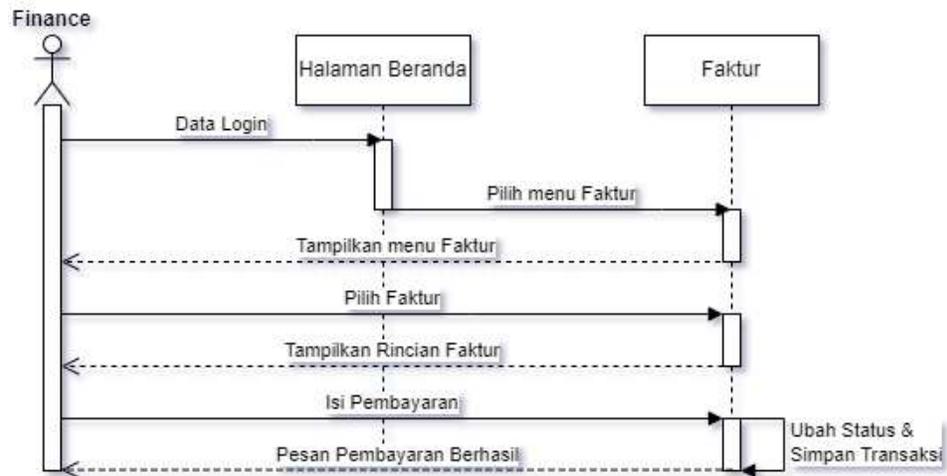
Sumber : Data olahan peneliti (2023)

Setelah melakukan *login*, *user Marketing* klik menu penawaran dan sistem akan menyajikan menu penawaran, untuk menambahkan penawaran baru *user* dapat mengklik tambah penawaran dan mengisi data-data penawaran pada form penawaran yang ditampilkan sistem, untuk menyimpan data yang telah diisikan pada form, *user* dapat mengklik Buat Penawaran dan sistem akan menyimpan data penawaran lalu mengirimkan pesan penawaran berhasil dibuat kepada *Marketing*.

Selanjutnya *user Admin* yang sudah *login* dapat memilih menu Penawaran lalu memilih penawaran dengan mengklik *edit*, sistem akan memunculkan form penawaran dan *user* dapat menambahkan item yang diperlukan, setelah selesai mengisi rincian penawaran maka *user* dapat mengklik “Simpan” untuk menyimpan data penawaran dan mengajukan penawaran ke *user Finance*.

Setelah penawaran diajukan oleh *user Admin* maka *user Finance* dapat melakukan persetujuan atau penolakan dengan memilih menu penawaran lalu memilih penawaran dengan mengklik “Lihat Penawaran” lalu mengklik “Setujui” atau “Tolak” pada *pop-up* rincian penawaran yang ditampilkan sistem, sistem akan mengubah status penawaran dan membuat *invoice* lalu mengembalikan pesan berhasil menyimpan menyetujui penawaran.

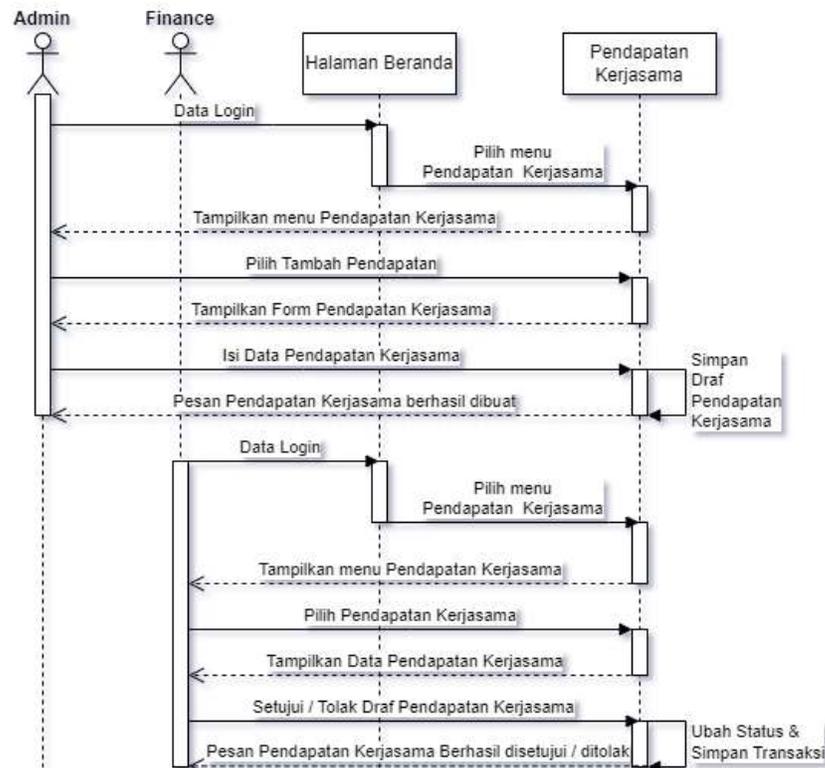
c. *Sequence diagram* Pembayaran Faktur



Gambar 3.17 *Sequence diagram* Pembayaran Faktur

Sumber : Data olahan peneliti (2023)

User Finance telah melakukan *login* dan diteruskan ke beranda, kemudian memilih menu faktur dan sistem akan menyajikan daftar faktur pada menu tersebut, *user* memilih faktur yang ingin dibayar dengan klik “Pembayaran” dan sistem akan menyajikan rincian faktur tersebut, kemudian *user* memasukkan nominal pembayaran dan sistem akan menyimpan transaksi, merubah status faktur, dan mengirim pesan bahwa pembayaran berhasil.

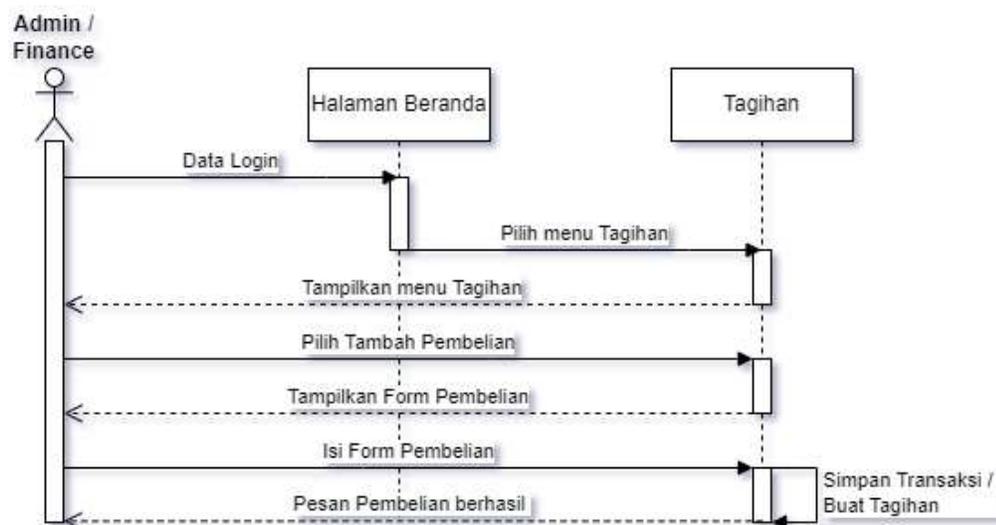
d. *Sequence diagram* Pendapatan Kerjasama**Gambar 3.18** *Sequence diagram* Pendapatan Kerjasama

Sumber : Data olahan peneliti (2023)

User telah login dan berada di beranda, user Admin membuat draf pendapatan kerjasama dengan memilih menu pendapatan kerjasama lalu sistem akan memperlihatkan menu pendapatan kerjasama, pada menu pendapatan kerjasama Admin klik Tambah Pendapatan kerjasama dan sistem akan menampilkan form pendapatan kerjasama, Admin mengisi rincian pendapatan kerjasama dan klik simpan maka sistem akan menyimpan draf pendapatan kerjasama dan mengirimkan pesan pendapatan kerjasama berhasil dibuat.

Draf pendapatan kerjasama yang telah dibuat oleh *Admin* kemudian akan diperiksa oleh *Finance* dengan memilih menu pendapatan kerjasama kemudian memilih pendapatan kerjasama pada menu pendapatan kerjasama yang ditampilkan sistem, setelah klik pilih pendapatan kerjasama maka sistem akan menyajikan rincian pendapatan kerjasama, jika tidak sesuai dengan laporan keuangan yang masuk maka pendapatan *Finance* dapat menolak dengan klik tolak dan sistem, jika sesuai maka *user Finance* dapat menyetujui dengan klik setuju maka sistem akan membuat mengubah status, membuat transaksi, dan mengirim pesan pendapatan kerjasama berhasil disetujui.

e. *Sequence diagram* Pembelian

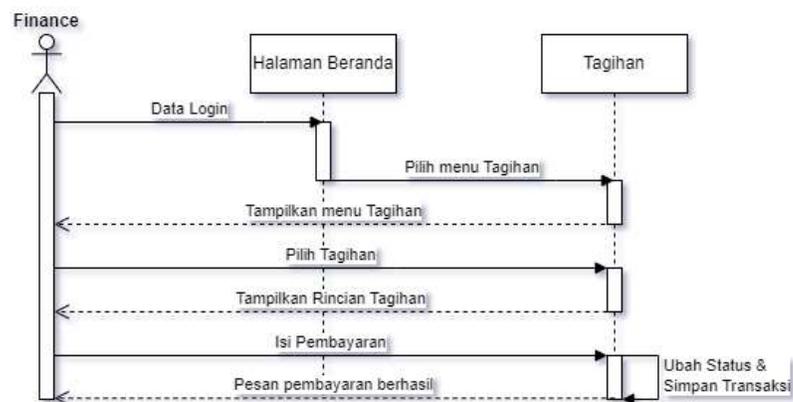


Gambar 3.19 *Sequence diagram* Pembelian

Sumber : Data olahan peneliti (2023)

User Admin atau *Finance* telah melakukan *login* dan diteruskan ke beranda, kemudian *user* mengklik menu Tagihan dan sistem akan menampilkan menu tagihan, kemudian *user* memilih tambah pembelian dan sistem akan merubah tampilan ke form pembelian, *user* diminta mengisi form pembelian dan klik Buat Pembelian maka sistem akan menyimpan pembelian dan membuat transaksi / membuat tagihan lalu mengirimkan pesan pembelian berhasil kepada *user*.

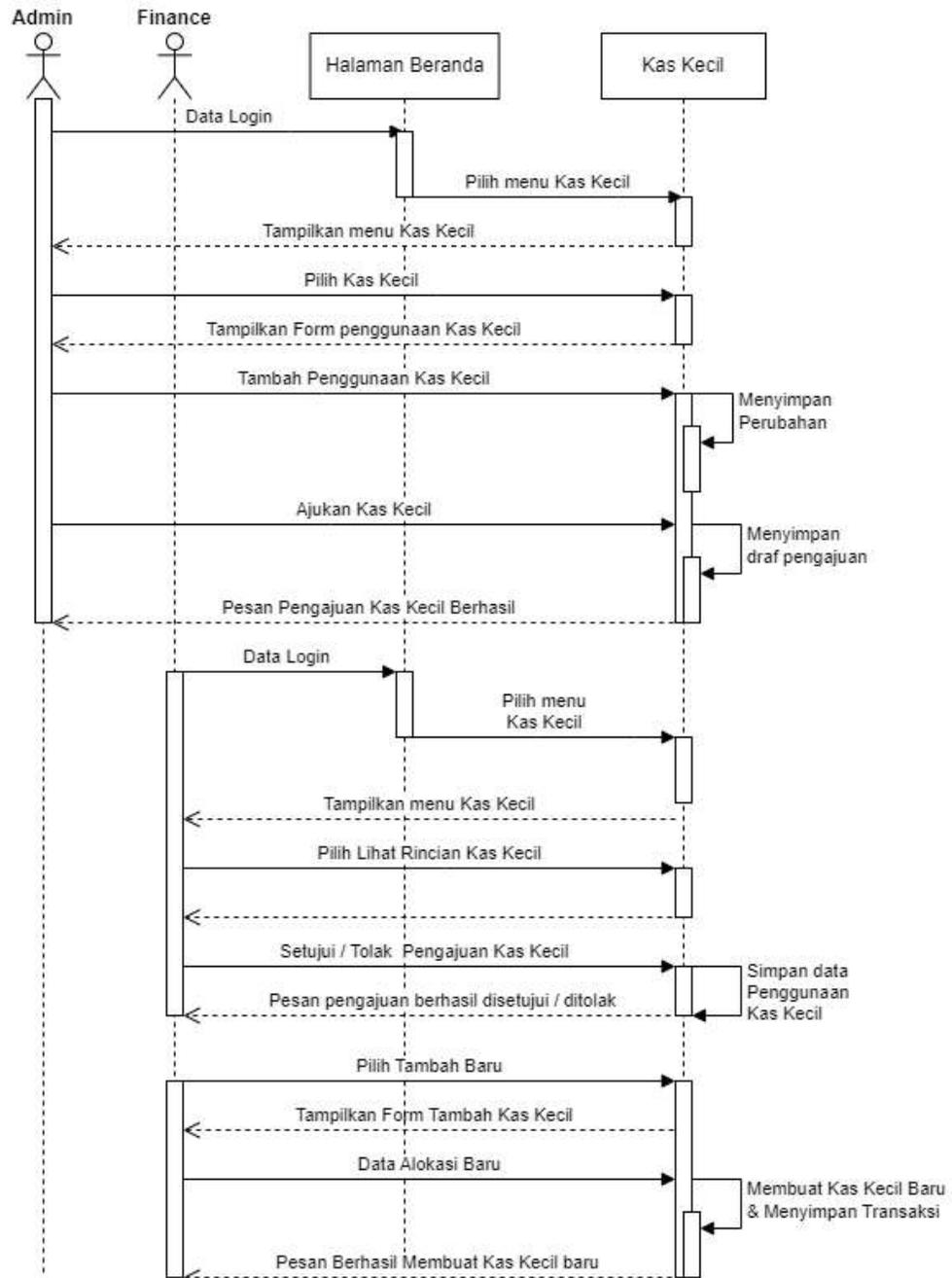
f. *Sequence diagram* Pembayaran Tagihan



Gambar 3.20 *Sequence diagram* Pembayaran Tagihan

Sumber : Data olahan penleiti (2023)

User Finance telah melakukan *login* dan diteruskan ke beranda, kemudian *user* mengklik menu Tagihan dan sistem akan menyajikan daftar tagihan pada menu tersebut, kemudian *user* klik pembayaran pada tagihan yang akan dibayarkan dan sistem akan menyajikan rincian pembelian, kemudian *user* mengisi nominal pembayaran dan klik tambah pembayaran maka sistem akan menyimpan transaksi dan mengubah status tagihan serta mengirimkan pesan pembayaran berhasil.

g. *Sequence diagram* Kas KecilGambar 3.21 *Sequence diagram* Kas Kecil

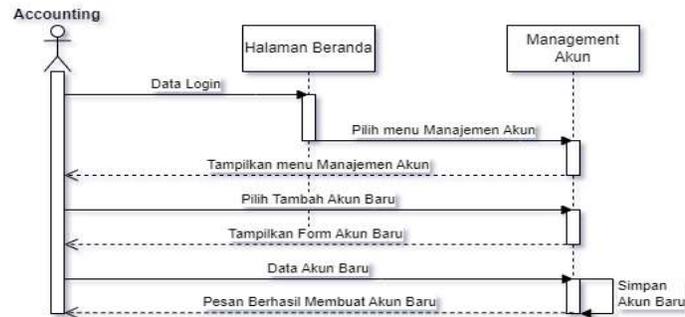
Sumber : Data olahan peneliti (2023)

User Admin dan *Finance* telah *login* dan diteruskan ke beranda, untuk menambahkan penggunaan kas kecil maka *Admin* memilih menu Kas Kecil lalu sistem akan menunjukkan menu Kas Kecil, kemudian *Admin* memilih kas kecil yang akan digunakan dan sistem akan menyajikan form rincian kas kecil yang dipilih, pada form tersebut *Admin* dapat menambahkan penggunaan kas kecil dan sistem akan menyimpan perubahan pada kas kecil tersebut. Setelah penggunaan kas kecil selesai maka *Admin* dapat melakukan pengajuan dengan klik ajukan, sistem akan merubah status kas kecil dan mengirimkan pesan pengajuan kas kecil berhasil.

Kas kecil yang telah diajukan akan diperiksa oleh *user Finance* dengan memilih menu Kas Kecil kemudian lihat rincian kas kecil, sistem akan menyajikan rincian penggunaan kas kecil dan sisa saldo, kemudian *Finance* dapat melakukan persetujuan atau penolakan pada pengajuan tersebut, jika ditolak maka sistem akan mengembalikan kepada status semula sebelum diajukan dan jika disetujui maka sistem akan merubah ke status selesai dan mengirimkan pesan pengajuan berhasil disetujui / ditolak.

Jika tidak ada form kas kecil yang masih aktif maka *user Finance* dapat melakukan tambah alokasi baru dengan pilih tambah baru pada menu kas kecil kemudian sistem akan memunculkan form pengisian kas kecil dan *Finance* diminta untuk mengisi data alokasi yang baru, setelah selesai maka sistem akan menyimpan transaksi dan membuat kas kecil baru serta mengirimkan pesan berhasil membuat kas kecil baru.

h. *Sequence diagram* Buat Akun Baru

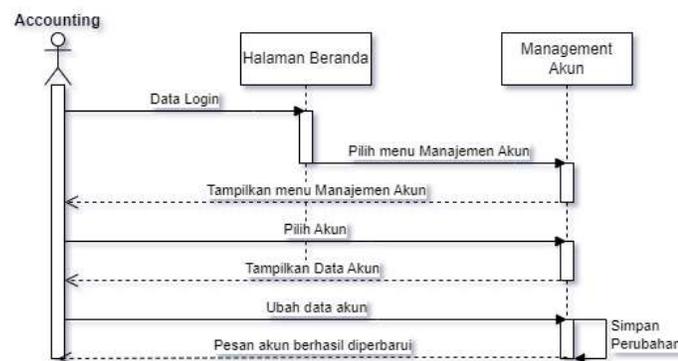


Gambar 3.22 *Sequence diagram* Buat Akun Baru

Sumber : Data olahan peneliti (2023)

User Accounting sudah *login* dan diteruskan ke beranda, kemudian *user* mengklik menu Manajemen Akun dan sistem akan memunculkan daftar akun perusahaan pada menu tersebut, lalu *user* pilih Tambah Akun Baru dan sistem akan menunjukkan form Akun baru, *user* mengisi data akun baru dan klik Buat Akun maka sistem akan menyimpan akun tersebut dan mengirimkan pesan berhasil membuat akun baru.

i. *Sequence diagram* Ubah Akun

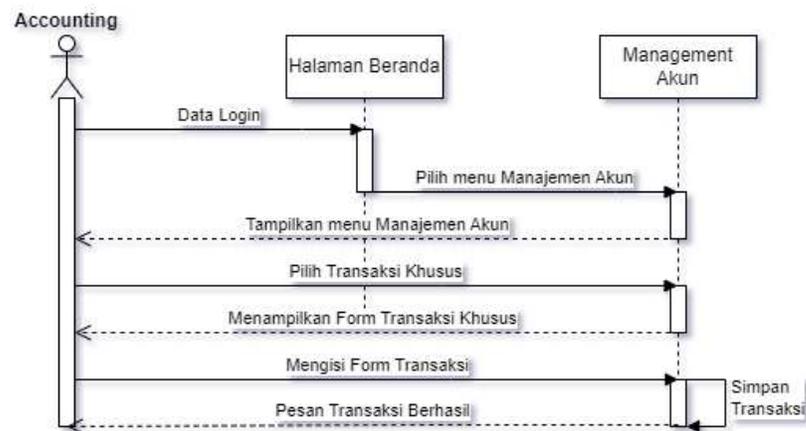


Gambar 3.23 *Sequence diagram* Ubah Akun

Sumber : Data olahan peneliti (2023)

User Accounting sudah *login* dan diteruskan ke beranda kemudian memilih menu Manajemen Akun dan sistem akan memperlihatkan menu Manajemen Akun, lalu *user* memilih akun yang ingin diubah dan sistem akan menyajikan rincian akun yang dipilih, kemudian *user* merubah data akun dan simpan, maka sistem akan menyimpan perubahan dan mengirimkan pesan akun berhasil diperbarui.

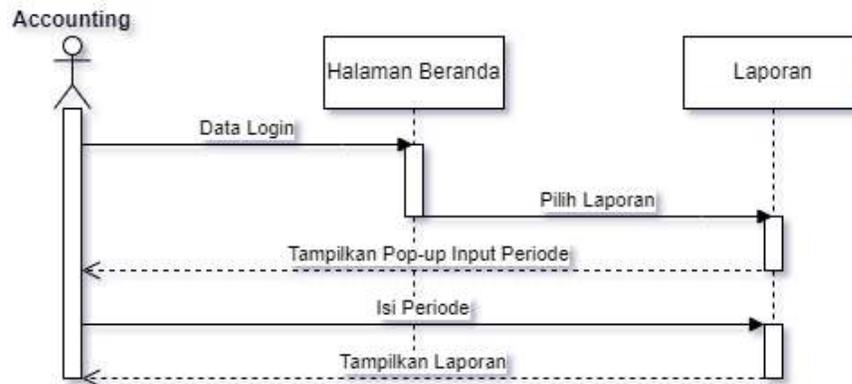
j. *Sequence diagram* Transaksi Khusus



Gambar 3.24 *Sequence diagram* Transaksi Khusus

Sumber : Data olahan peneliti (2023)

User Accounting sudah *login* dan diteruskan ke beranda, kemudian *user* mengklik menu Manajemen Akun dan sistem akan memperlihatkan Menu Manajemen Akun, kemudian *user* mengklik Transaksi Khusus lalu sistem akan memunculkan form Transaksi khusus, setelah itu *user* mengisi data transaksi dan klik “Simpan” dan sistem akan mencatat transaksi.

k. *Sequence diagram* Laporan**Gambar 3.25** *Sequence diagram* Laporan

Sumber : Data olahan peneliti(2023)

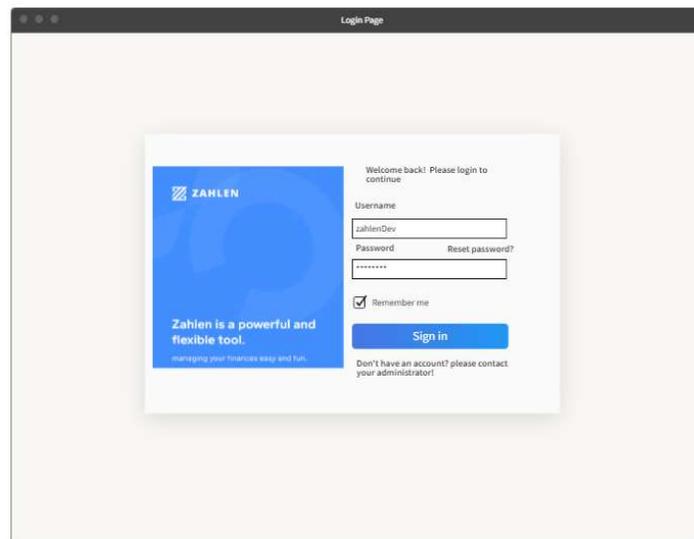
User Accounting sudah *login* dan diteruskan ke beranda, kemudian *user* mengklik menu laporan dan laporan yang diinginkan kemudian sistem akan meminta *user* memasukkan periode, setelah mengisi periode *user* klik “Tampilkan” kemudian sistem akan menunjukkan laporan keuangan.

merupakan pewarisan dari *class User*, sementara itu garis dengan warna menunjukkan hubungan *user* dengan *role*-nya terhadap *class* lain dan garis ungu menunjukkan hubungan antara *class* yang tidak berhubungan langsung dengan *user*.

b. Desain UI

Tahap desain pada sprint kedua adalah membuat rancangan *layout* dari UI yang dibuat di *website* Mockflow sebagai berikut:

1. *Login Page*

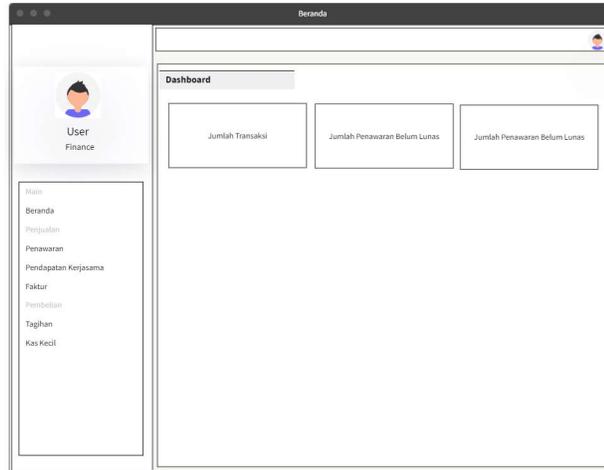


Gambar 3.27 Rancangan *Login Page*

Sumber : Data olahan peneliti (2023)

Pada halaman *login*, *user* akan diminta untuk mengisi *username* dan *password*.

2. Beranda

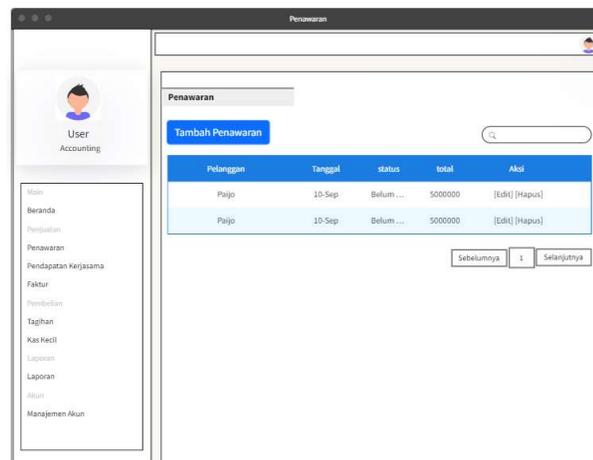


Gambar 3.28 Rancangan Beranda

Sumber : Data olahan peneliti (2023)

Halaman beranda merupakan halaman terusan dari *login* dan tidak perlu melakukan *input* apapun. Adapun gambar di atas merupakan rancangan beranda *user Finance*.

3. Penawaran



Gambar 3.29 Rancangan Penawaran

Sumber : Data olahan peneliti (2023)

Menu Penawaran memperlihatkan daftar penawaran dan statusnya, user tidak perlu memasukkan apapun pada halaman ini.

4. Tambah Penawaran

The screenshot shows a web application interface for adding a bid. On the left is a sidebar with a user profile for 'User Marketing' and a menu with options like 'Beranda', 'Penjualan', 'Penawaran', 'Pendapatan Kerjasama', 'Faktur', 'Pembelian', 'Tagihan', 'Kas Kecil', 'Laporan', 'Akun', and 'Manajemen Akun'. The main content area is titled 'Tambah Penawaran' and contains a form with the following fields:

- Customer: PT. Kurnia Samudra
- Jenis Jasa / Biaya: 004 - Jasa Tunda
- Jumlah: 1
- Harga Total: 120000000
- Deskripsi: Jasa Tunda MT Fumika Masuk

Below the form is a table with the following data:

No	Kode	Deskripsi	Jumlah	Harga Total	Aksi
1	003	Pandu MT Fumika	2	4500000	[Hapus]

The total amount is displayed as 'Total: 4500000'. There are three buttons: 'Kembali' (yellow), 'Tambah' (blue), and 'Simpan' (green).

Gambar 3.30 Rancangan Tambah Penawaran

Sumber : Data olahan peneliti (2023)

Pada form ini *user* diminta untuk memasukkan nama *customer*, kode akun yang akan ditambahkan dalam biaya penawaran, jumlah jasa yang digunakan, harga total dari biaya tersebut dan deskripsi dari jasa biaya tersebut.

5. Persetujuan Penawaran

Detail Penawaran x

#Ref	Detail Penawaran	Status		
Pemohon	Diserahkan Oleh			
<input type="text" value="Marketing"/>	<input type="text" value="1-12003 - Kas Kecil Umum"/>			
Tanggal Pengajuan:	Tanggal Jatuh tempo			
<input type="text" value="01-01-2023"/>	<input type="text" value="31-01-2023"/>			
Total : 56.000.000				
Rincian				
No	Kode Akun	Keterangan	Jumlah	Total
1	40001	Jasa Tunda MT. KIRKARA	1	56.000.000

Gambar 3.31 Rancangan Persetujuan Penawaran

Sumber : Data olahan peneliti (2023)

Pada tampilan ini *user* tidak perlu memasukkan apapun namun dapat memilih untuk menyetujui atau menolak penawaran, adapun tampilan ini hanya dimiliki oleh *user* dengan *role Finance*.

6. Pendapatan Kerjasama

Keterangan	Tanggal Pengajuan	Status	Total	Aksi
Pandu November	02-Nov	Pengajuan	400000000	[Lihat Detail][Edit][Hapus]
Pandu Oktober	10-Okt	Disetujui	1500000000	[Lihat Detail]

Gambar 3.32 Rancangan Pendapatan Kerjasama

Sumber : Data olahan peneliti (2023)

Pada menu ini akan ditampilkan daftar dari pengajuan pendapatan kerjasama dan *user* tidak perlu memasukkan apapun, adapun rancangan di atas merupakan sudut pandang dari *user* Admin.

7. Tambah Pendapatan Kerjasama

The screenshot shows a web application interface for adding cooperative income. The page title is "Tambah Pendapatan Kerjasama". On the left, there is a sidebar menu with the following items: Main, Beranda, Penjualan, Penawaran, **Pendapatan Kerjasama** (highlighted), Pembelian, Tagihan, and Kas Kecil. Above the menu, the user profile is displayed as "User Admin" with a profile icon. The main content area is titled "Tambah Pendapatan" and includes a "Kembali" button in the top right corner. The form contains the following fields:

- Keterangan:** A text area containing "Pendapatan Tunda Periode 1-30 November" and "Jumlah Kegiatan 57".
- Kode:** A text input field containing "004 - Jasa Tunda".
- Harga Total:** A text input field containing "120000000".

At the bottom center of the form is a green "Simpan" button.

Gambar 3.33 Rancangan Tambah Pendapatan Kerjasama

Sumber : Data olahan peneliti (2023)

Pada halaman ini *user* dengan *role Admin* diminta untuk memasukkan keterangan, kode akun jasa dan harga total atau total pendapatan dari jasa tersebut.

8. Persetujuan Pendapatan Kerjasama



The image shows a web form titled "Detail Pendapatan" with a close button (x) in the top right corner. The form contains the following fields:

- #Ref
- Kode akun:
40001 - Jasa Tunda
- Total:
Rp. 63.000.000
- Keterangan:
Pendapatan Jasa tunda bulan desember

At the bottom of the form, there are two buttons: a red "Tolak" button and a green "Setujui" button.

Gambar 3.34 Rancangan Persetujuan Pendapatan Kerjasama

Sumber : Data olahan peneliti (2023)

Tampilan ini merupakan tampilan form rincian dari pendapatan kerjasama dan hanya dapat diakses oleh *user* dengan *role Finance*, *user* dapat memilih untuk menyetujui atau menolak dan tidak perlu memasukkan apapun.

9. Faktur

Pelanggan	Ref	Tgl Terbit	Tgl Jatuh Tempo	Tgl Pelunasan	Aksi
PT. Samudra Tiga	WO-067	06-Nov-2023	31-Des-2023	-	[Pembayaran]
PT. Samudra Dua	WO-066	25-Okt-2023	25-Nov-2023	-	[Pembayaran]
PT. Samudra Lines	WO-065	10-Okt-2023	10-Nov-2023	08-Nov-2023	[Detail]

Gambar 3.35 Rancangan Faktur

Sumber : Data olahan peneliti (2023)

Halaman ini berisi daftar faktur yang telah diterbitkan dan tidak perlu memasukkan apapun, akses ke halaman ini hanya dimiliki oleh *user* dengan *role Finance*.

10. Pembayaran Faktur

Faktur #NO23

Detail Pembayaran Faktur Ref: WO-079 Kembali

Customer: PT, Lintas Ocean Tanggal Jatuh Tempo: 13-Dec-2023

Rincian

No	Kode	Deskripsi	Jumlah	Harga Total
1	003	Pandua MT Fumika	2	4500000

Total : 4500000

Histori Pembayaran

Pembayaran

Sisa Tagihan Rp. 450.000

Total Dibayar: 4500000 Dibayar ke: 100001 - Kas (Rp) Tanggal Transaksi: 5-Nov-2023

Note: Pendapatan Tunda Periode 1-30 November
Jumlah Kegiatan 57

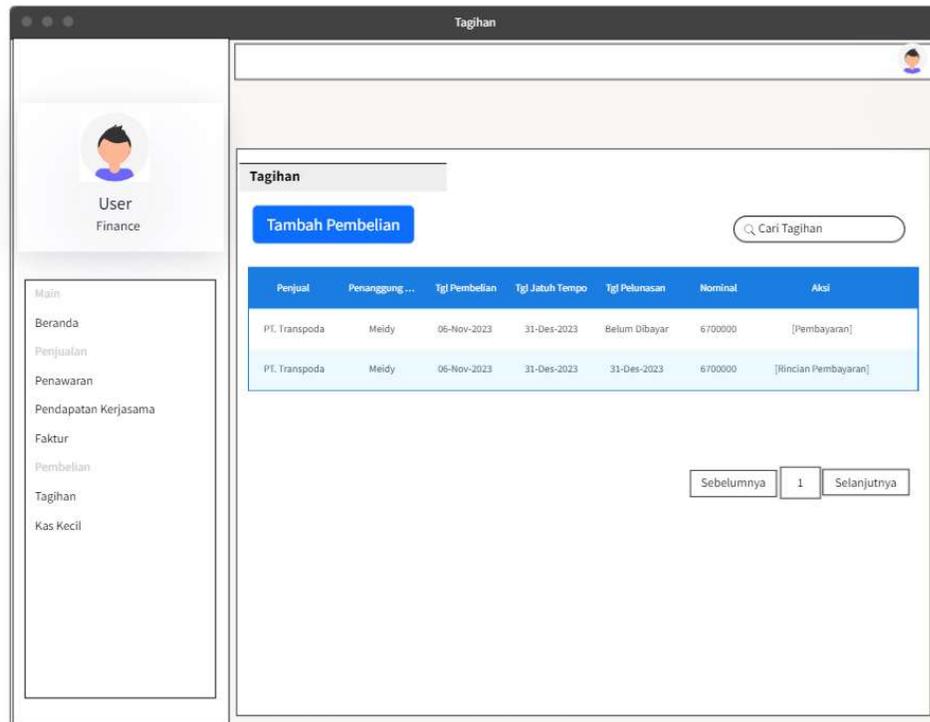
Tambah Pembayaran

Gambar 3.36 Rancangan Tambah Pembayaran Faktur

Sumber : Data olahan peneliti (2023)

Pada halaman ini *user* diminta untuk mengisi Total dibayar, dibayar ke (akun tujuan pembayaran), tanggal transaksi, dan *note* atau catatan bila ada.

11. Tagihan



Gambar 3.37 Rancangan Tagihan

Sumber : Data olahan peneliti (2023)

Pada halaman ini *user* dapat melihat daftar pembelian yang telah dibuat dan tidak perlu memasukkan apapun, ada pun rancangan di atas merupakan sudut pandang dari *user* dengan *role Finance* namun halaman ini juga dapat diakses oleh *user* dengan *role Admin*.

12. Tambah Pembelian

Tambah Tagihan / Pembelian

Kembali

Tambah Pembelian

Kode Akun: 67000 - Boat Service Nominal: 250000 Jumlah: 2

Deskripsi: Boot Service MT. Fumika In & Out

Reset Tambah

Detail

No	Kode	Deskripsi	Jumlah	Nominal	Total	Aksi
1	67000	Boot Service MT Fumika In & Out	2	250000	500000	[Hapus]

Total : 500000

Selanjutnya

Gambar 3.38 Rancangan Tambah Pembelian

Sumber : Data olahan peneliti (2023)

Pada halaman ini *user* diminta untuk mengisi kode akun, nominal, jumlah dan deskripsi dari pembelian atau pembuatan tagihan kemudian menambahkannya dengan mengklik tombol tambah, setelah selesai mengisi rincian pembelian, *user* diminta untuk mengklik selanjutnya yang mana sistem akan menunjukkan form pengisian data pembelian dengan rancangan sebagai berikut:

The image shows a mobile application form for creating a purchase. At the top left, the title 'Pembelian' is displayed. To the right is a toggle switch labeled 'Cash', which is currently turned on. Below the title, there are three input fields: 'Penjual' containing 'PT. TIRTA ASRI', 'Dibayar dari' containing '10010 - KAS BESAR', and 'Tanggal Transaksi' containing '01-01-2023'. At the bottom, there are two buttons: a grey 'Batal' button and a blue 'Buat Pembelian' button.

Gambar 3.39 Rancangan Keterangan Pembelian

Sumber : Data olahan peneliti (2023)

Pada form ini *user* dapat memilih apakah akan membuat pembelian kontan atau menambahkan tagihan dengan menggeser *toggle button*, jika memilih *cash* maka *user* diminta untuk memasukkan nama penjual, akun sumber pembayar dan tanggal transaksi, namun jika memilih tidak *cash* maka *user* akan diminta untuk mengisi nama penjual, tanggal transaksi dan tanggal jatuh tempo.

13. Pembayaran Tagihan

Detail Pembayaran Tagihan #213213

Penjual

Tanggal Jatuh Tempo

Rincian

No	Kode	Deskripsi	Jumlah	Nominal	Total
1	67000	Boot Service MT Fumika In & Out	2	250000	500000

Total : 500000

Histori Pembayaran

Referensi	Tanggal Pembayaran	Tagihan Awal Pembayaran	Nominal Pembayaran
#213213	17 Agustus	5000000	450000

Pembayaran

Dibayar dari

Tanggal Transaksi

Notice

Total Dibayar

Sisa Tagihan Rp. 450.000

Tambah Pembayaran

Gambar 3.40 Rancangan Pembayaran Tagihan

Sumber : Data olahan peneliti (2023)

Pembayaran hanya dapat dilakukan oleh *user* dengan *role Finance* dengan mengklik “Pembayaran” pada tagihan di menu Tagihan, pada tampilan ini *user* diminta untuk mengisi nominal pembayaran, akun sumber pembayaran, tanggal transaksi dan *note* atau catatan bila ada.

14. Kas Kecil

The screenshot shows a web application interface for 'Kas Kecil'. On the left is a sidebar menu with options: Main, Beranda, Pengajuan, Penawaran, Pendapatan Kerjasama, Faktur, Pembelian, Tagihan, and Kas Kecil. The main content area has a 'Kas Kecil' header with a 'Tambah Baru' button. Below is a table with columns: Deskripsi, Saldo Awal, Total Digunakan..., Saldo Akhir, Status, Tanggal, and Aksi. The table contains two rows of data for October and September. At the bottom right, there are navigation buttons: 'Sebelumnya', '1', and 'Selanjutnya'.

Deskripsi	Saldo Awal	Total Digunakan...	Saldo Akhir	Status	Tanggal	Aksi
Periode Oktober	500000	360000	120000	On Going	-	[Edit][Pengajuan]
Periode September	500000	480000	200000	Di Tutup	15 Okt-2023	[Lihat Rincian]

Gambar 3.41 Rancangan Kas Kecil

Sumber : Data olahan peneliti (2023)

Halaman ini berisi daftar form kas kecil yang aktif maupun yang sudah selesai atau ditutup, *user* tidak perlu mengisi apapun pada halaman ini.

15. Form Kas Kecil

The screenshot shows a web application interface for 'Kas Kecil' in form mode. The sidebar menu is different, including: Main, Beranda, Pengajuan, Penawaran, Pendapatan Kerjasama, Faktur, Pembelian, Tagihan, Kas Kecil, Laporan, Akun, and Manajemen Akun. The main content area has a 'Kas Kecil' header with 'Periode Oktober' and 'Status' dropdowns, and a 'Kembali' button. Below are input fields for 'No Bukti' (KRSND-82142), 'Tanggal Transaksi' (07-Nov-2023), 'Harga' (1200000), and 'Deskripsi' (FORM 2AJ PANDU @20 PACK). A 'Tambah' button is present. Below the form is a table with columns: No, Deskripsi, Harga, No Bukti, Tanggal Transaksi, User, and Aksi. The table contains two rows of data. At the bottom right, there are summary values: 'Total : 275000' and 'Sisa Saldo : 4725000', with an 'Ajukan' button.

No	Deskripsi	Harga	No Bukti	Tanggal Transaksi	User	Aksi
1	Air Minum Sanford @5 Galon	70000	BMRT-7213	02-04-2023	Melyd	[Hapus]
2	Kertas HVS A4 @1 Box	205000	BLS-7792	05-04-2023	Maa	[Hapus]

Gambar 3.42 Rancangan Form Kas Kecil

Sumber : Data olahan peneliti (2023)

Pada halaman ini *user* dapat menambahkan transaksi kas kecil dengan mengisi no bukti transaksi jika ada, tanggal transaksi, harga atau nominal transaksi, dan deskripsi.

16. Rincian Pengajuan Kas Kecil

Rincian Kas Kecil
X

#PC-23192
Periode Oktober
Status

Kode Akun

Diajukan Oleh:

Saldo Awal : Rp. 5.000.000 Jumlah Pemakaian: Rp. 4.870.000 Saldo Akhir : Rp. 130.000

Rincian

No	No Bukti	Deskripsi	Harga	Tanggal Transaksi	Penan...
1	IMRT-7213	Air Minum Sanford @5 Galon	70000	02-Okt-2023	Meidy
2	IMRT-7213	Air Minum Sanford @5 Galon	70000	02-Okt-2023	Meidy
3	IMRT-7213	Air Minum Sanford @5 Galon	70000	02-Okt-2023	Meidy

Tolak

Setujui

Gambar 3.43 Rancangan Rincian Kas Kecil

Sumber : Data olahan peneliti (2023)

Pada halamana ini *user* dapat melihat rincian dari penggunaan kas kecil dan melakukan persetujuan atau penolakan terhadap pengajuan kas kecil.

17. Tambah Kas Kecil

Tambah Kas Kecil
Status

Deskripsi

Dibayar dari:

Nominal

Akun Tujuan:

Notice : lorem ipsum ..
Sisa Saldo : Rp. 300.000

Saldo Awal	Saldo Ditambahkan	Saldo akhir
Rp. 300.000	Rp. 4.700.000	Rp. 5.000.000

Gambar 3.44 Rancangan Tambah Kas Kecil

Sumber : Data olahan peneliti (2023)

Pada tampilan ini *user* mengisi deskripsi dari alokasi kas kecil yang baru, nominal penambahan, akun sumber penambahan kas kecil, dan akun kas kecil tujuan, hanya *user* dengan *role Finance* yang dapat mengakses tampilan ini dengan tombol Tambah Baru pada menu Kas Kecil.

18. Laporan Laba Rugi

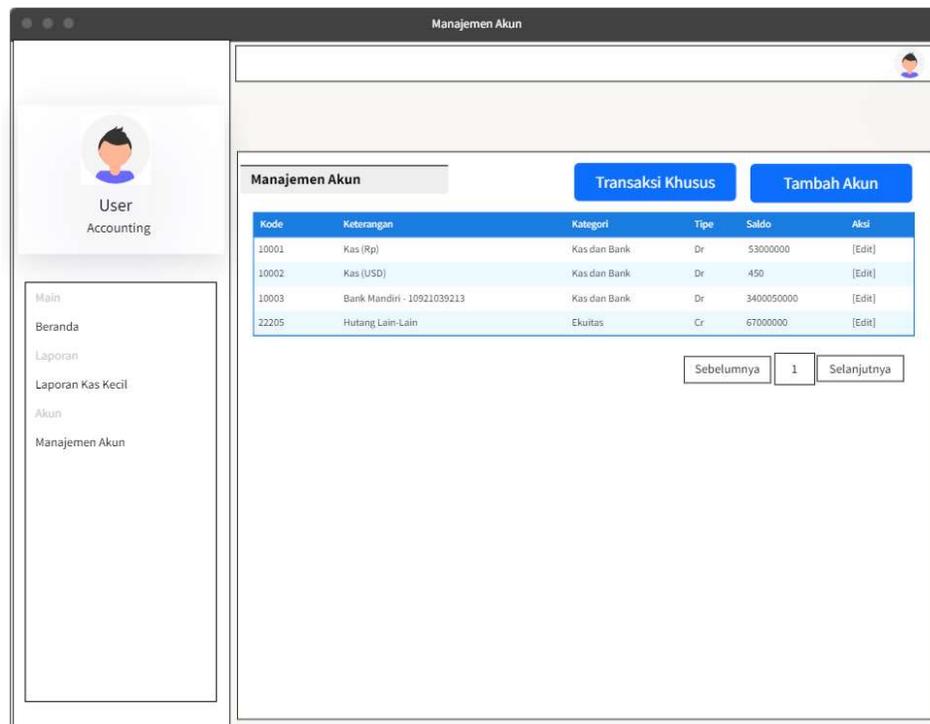
Laporan Laba Rugi	
PT. SEGARA CATUR PERKASA	
Periode Oktober	
Pendapatan	
Pendapatan Jasa Pandu	2.468.213.000
Pendapatan Jasa Tunda	4.184.641.000
Total Pendapatan	6.652.854.000
Biaya Pokok Jasa	
Biaya Overhead Jasa	332.667.335
Biaya PNBP Pandu Tunda	1.330.570.800
Biaya PNBP Koneksi Data VTS	12.000.000
Biaya Operasional lain	23.000.000
Total Beban Usaha	(1.663.238.135)
Total Laba Kotor	4.989.615.865
Biaya Lain-lain	
Biaya Administratif dan umum	13.800.000
Biaya Gaji	652.500.000
Biaya BPJS	26.400.000
Total biaya lain-lain	692.700.000
Laba Bersih sebelum pajak	4.296.915.865
Pajak	240.700.000
Laba / (Rugi) Bersih	4.056.215.865

Gambar 3.45 Rancangan Laporan Laba Rugi

Sumber : Data olahan peneliti (2023)

Tampilan ini merupakan rancangan laporan laba rugi perusahaan yang hanya dapat diakses oleh user *Accounting* setelah *user* menentukan periode transaksi.

19. Manajemen Akun



Gambar 3.46 Rancangan Manajemen Akun

Sumber : Data olahan peneliti (2023)

Pada halaman ini *user Accounting* dapat melihat daftar akun yang dimiliki, dari tampilan ini juga *user* dapat memilih untuk membuat akun baru, membuat transaksi baru maupun merubah akun yang sudah ada.

20. Tambah Akun Baru

The screenshot shows a web application interface for adding a new account. The page title is "Akun Baru". On the left, there is a sidebar menu with the following items: Main, Beranda, Penjualan, Penawaran, Pendapatan Kerjasama, Faktur, Pembelian, Tagihan, Kas Kecil, Laporan, Laporan, Akun, and Manajemen Akun. The user profile is "User Accounting". The main form area is titled "Tambah Akun Baru" and contains the following fields and controls:

- Nama Akun:** Text input field containing "Hutang Lain-lain".
- Kode Induk:** Text input field containing "4".
- Kode:** Text input field containing "45302".
- Kategori:** Dropdown menu with "Kategori Akun" selected. A toggle switch for "Kategori Baru" is present.
- Tipe Akun:** Dropdown menu with "Db / Cr" selected.
- Saldo Awal:** Text input field containing "0".

There are two buttons: "Kembali" (yellow) and "Buat Akun" (blue).

Gambar 3.47 Rancangan Tambah Akun

Sumber : Data olahan peneliti (2023)

Pada halaman ini *user* mengisi kode akun beserta kode induknya, kategori akun, nama akun, saldo awal, dan tipe akun.

21. Ubah Akun

The screenshot shows a web application interface for editing an account. On the left, a sidebar menu lists various accounting functions, with 'Manajemen Akun' (Account Management) selected. The main content area is titled 'Ubah Akun' (Change Account) and contains several input fields: 'Nama Akun' (Account Name) set to 'Hutang Lain-lain', 'Kode Induk' (Parent Code) set to '4', 'Kode' (Code) set to '45302', 'Kategori' (Category) set to 'Kategori Akun', 'Tipe Akun' (Account Type) set to 'Db / Cr', and 'Saldo Awal' (Initial Balance) set to '0'. A 'Kembali' (Back) button is located at the top right, and a 'Simpan' (Save) button is at the bottom right.

Gambar 3.48 Rancangan Ubah Akun

Sumber : Data olahan peneliti (2023)

Pada halaman ini *user* dapat mengubah data akun jika terdapat kesalahan pengisian data atau ada perubahan terhadap data seperti kategori akun, nama akun, kode induk dan tipe akun, hal ini hanya dapat dilakukan oleh *user Accounting* karena mengubah saldo awal tidak dapat dilakukan sembarangan.

22. Transaksi Khusus

Transaksi Khusus
Batal

Nominal

Tanggal Transaksi

Kode Akun

Debet / Kredit

Deskripsi

No	Kode Akun	Deskripsi	Tanggal Trans,...	Debit	Kredit
1	1-10001 - Kas (Rp)	Setor Bank BNI	02-02-2023	-	17000000
2	1-12002 - Bank BNI	Setor Bank BNI	02-02-2023	17000000	-

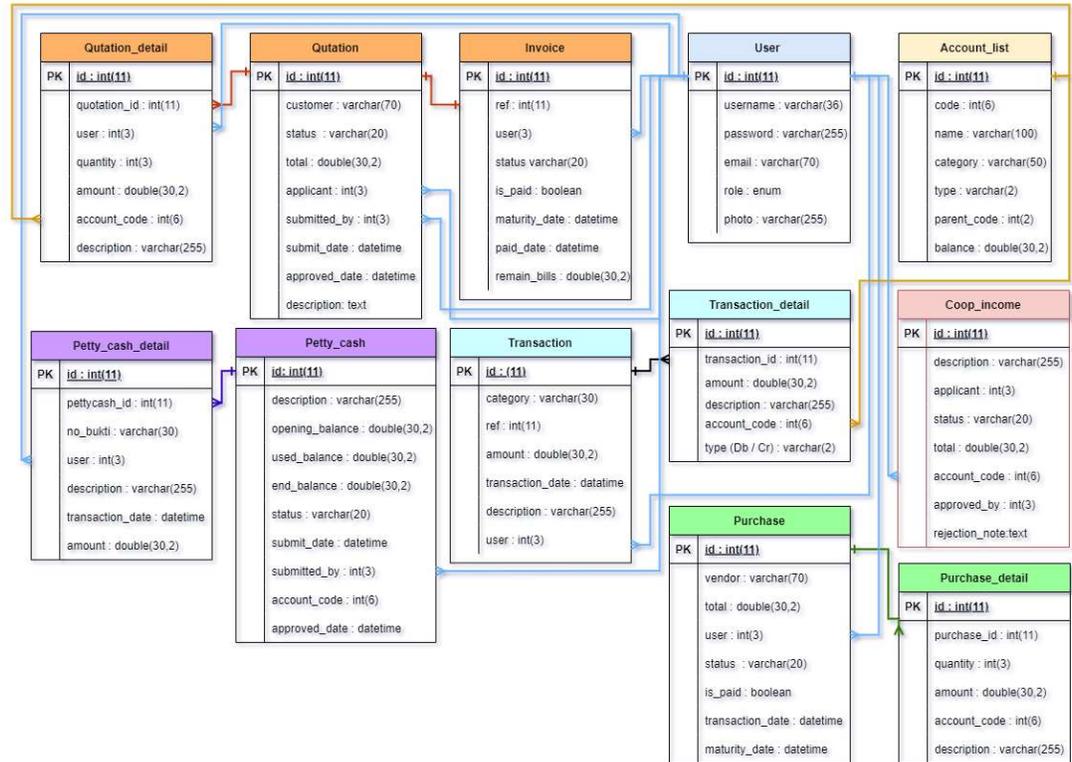
Buat Transaksi

Gambar 3.49 Rancangan Transaksi Khusus

Sumber : Data olahan peneliti (2023)

Pada tampilan ini *user* mengisi deskripsi dari transaksi, tanggal transaksi kemudian rincian transaksi yang berupa kode akun ,deskripsi, dan nominal debet atau kredit, menu ini juga merupakan alternatif dari merubah saldo awal pada menu ubah akun karena lebih aman.

c. Desain Database



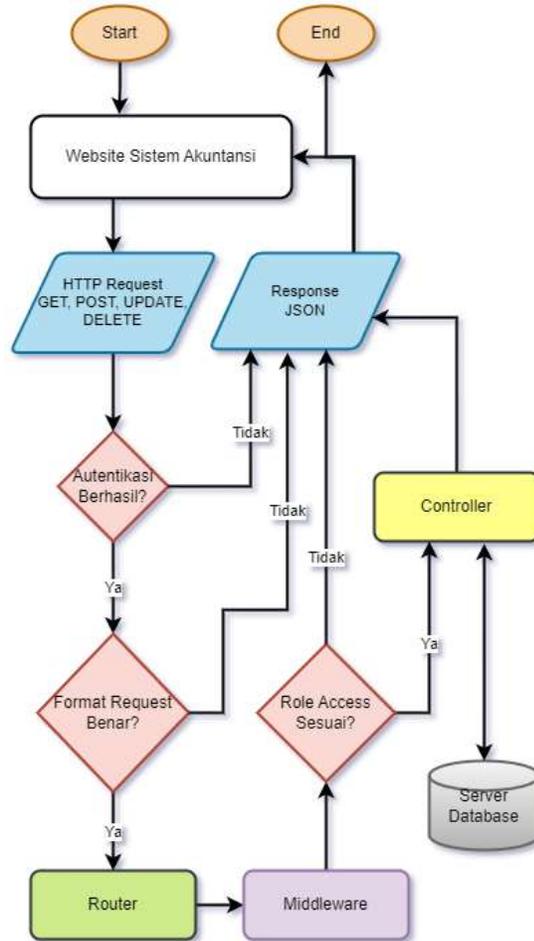
Gambar 3.50 Tabel Database

Sumber : Data olahan peneliti (2023)

Gambar di atas merupakan tabel struktur *database* dari sistem akuntansi yang dibangun, *database* dibuat dengan *Object-Relational Mapping (ORM)* untuk Node.js yaitu *Sequelize* sehingga pengembang dapat menggunakan *DBMS* seperti *Postgres*, *MySQL* dan banyak *database* relasional lainnya. Selain memiliki banyak pilihan *database* yang bisa digunakan, penggunaan *Sequelize* juga memudahkan ketika berpindah *database*. Adapun pada sistem ini terdapat 12 tabel atau entitas yaitu :

1. *User* : Tabel yang menyimpan data *user* selaku pengguna sistem.
2. *Account_list* : Tabel yang menyimpan data akun-akun akuntansi yang digunakan oleh perusahaan.
3. *Quotation* : Tabel yang menyimpan data penawaran.
4. *Quotation_detail* : Tabel yang menyimpan data rincian penawaran.
5. *Invoice* : Tabel yang menyimpan data faktur.
6. *Coop_income* : Tabel yang menyimpan data pendapatan kerjasama.
7. *Transaction* : Tabel yang menyimpan data transaksi.
8. *Transaction_detail* : Tabel yang menyimpan data rincian transaksi.
9. *Petty_cash* : Table yang berisi data kas kecil.
10. *Petty_cash_detail* : Tabel yang berisi data rincian kas kecil.
11. *Purchase* : Tabel yang berisi data pembelian dan tagihan.
12. *Purchase_detail* : Tabel yang berisi data rincian pembelian dan tagihan.

Warna pada tabel menunjukkan bahwa tabel tersebut berkaitan erat atau menggambarkan hubungan tabel utama dan tabel rincian, adapun warna garis menunjukkan relasi yang dimiliki suatu entitas.

d. *Flowchart REST API*

Gambar 3.51 *Flowchart REST API*

Sumber : Data olahan peneliti (2023)

Gambar di atas merupakan *flowchart REST API* sistem akuntansi yang dibangun, dapat dilihat bahwa untuk dapat menggunakan *endpoint*, *user* setidaknya harus melalui dua kali verifikasi yaitu verifikasi akun dan verifikasi *role*. Pada verifikasi akun, *API* akan memeriksa apakah *username* dan *password* yang

dikirimkan valid, jika tidak sesuai maka *API* akan mengirimkan respon *login* gagal, jika berhasil maka *API* akan memeriksa apakah format *request* yang dikirimkan sudah benar, jika salah maka sistem akan mengirimkan respon *error* dengan kode sesuai jenis *error*, jika berhasil maka *API* akan memverifikasi apakah *role user* berhak untuk mengakses *endpoint* tersebut, jika tidak maka sistem akan mengirimkan respon *error user* tidak memiliki akses melalui *endpoint* tersebut.

3. *Develop*

Tahap ini adalah tahap membangun atau pengkodean, pada *sprint* pertama tentu saja pengkodean dilakukan untuk membangun *API* dengan desain yang telah dibuat sebelumnya, lalu pada *sprint* kedua pengkodean dilakukan untuk membangun UI yang akan menjadi penghubung antara *user* dan *API*. Kemudian pada *sprint* ketiga dilakukan integrasi atau penggabungan kode *API* dan UI yang sudah di-*deploy* sebelumnya.

4. *Test*

Tahap *test* menjadi sangat krusial karena menentukan apakah sistem dapat beroperasi secara efektif atau tidak. Pada *sprint* pertama pengujian dilakukan terhadap setiap *endpoint* yang telah dibuat untuk memastikan bahwa *endpoint* tersebut dapat diakses dan memberikan *response* yang sesuai dengan *request* atau status yang telah diberikan, pengujian ini memanfaatkan Postman agar dapat dilakukan lebih cepat dan terstruktur. Pada *sprint* kedua pengujian difokuskan pada fungsionalitas dari UI yang telah dibuat, pengujian ini dilakukan dengan integrasi dengan *API* yang dijalankan secara lokal dengan data *dummy*. Pada *sprint* ketiga tahap pengujian dilakukan dengan mencoba secara langsung aplikasi yang telah

selesai dibangun, pengujian ini meliputi pengujian fungsionalitas aplikasi, kecocokan desain UI, dan kesesuaian sistem dengan *workflow* yang direncanakan.

5. *Deploy*

Tahap *deploy* adalah tahap implementasi dari *API* dan UI yang telah diuji sebelumnya. Pada *sprint* pertama *deployment* dilakukan terhadap *API* agar dapat digunakan secara *online*, sedangkan pada *sprint* kedua *deployment* dilakukan pada UI. Berbeda dari kedua *sprint* sebelumnya, *deployment* pada *sprint* ketiga justru merupakan implementasi dari aplikasi yang telah lulus pengujian, pada tahap ini aplikasi di-*deploy* ke *production environment* dan siap untuk digunakan.

6. *Review*

Tahap ini merupakan tahap evaluasi dimana setelah *API* dan *UI* diuji mungkin saja masih terdapat hal yang masih perlu diperbaiki atau ada perubahan yang diminta oleh *user* atau *stakeholders* lainnya sehingga dilakukan pengkodean ulang.

7. *Launch*

Tahap ini merupakan final dari pengembangan aplikasi dimana pada tahap ini semua aspek telah diuji dan memenuhi kebutuhan keinginan pengguna, mencapai tahap ini berarti sistem secara resmi siap untuk digunakan.

3.4 Metode Pengujian Sistem

Metode pengujian merupakan konsep-konsep teoritis tertentu yang digunakan untuk menjelaskan atau mendukung penelitian, pengujian dilakukan

kepada pengguna untuk meminimalkan kesalahan-kesalahan yang terjadi pada sistem yang telah dirancang (Hutabri 2019).

Dalam penelitian ini, metode pengujian black box diterapkan untuk menguji sistem yang telah diintegrasikan dengan *API*. Metode pengujian *black box* adalah pengujian yang berfokus pada evaluasi fungsionalitas dan perilaku eksternal sistem *software* tanpa mencermati rincian implementasinya secara internal. Dalam pengujian ini, penguji tidak perlu memiliki pengetahuan detail tentang struktur kode sumber atau logika internal sistem. Sebaliknya, fokusnya adalah pada memastikan bahwa sistem berfungsi sesuai dengan spesifikasi fungsionalnya, sehingga pengguna akhir dapat menggunakan produk atau aplikasi tersebut dengan efektif.

3.5 Lokasi dan Jadwal Penelitian

Lokasi penelitian yang dipilih peneliti dan jadwal penelitian yang ditetapkan peneliti adalah sebagai berikut:

1. Lokasi Penelitian

Penelitian ini dilakukan di kantor PT Segara Catur Perkasa dengan alamat Jl. Yos Sudarso, Komplek Bintang Industrial Park 1 Lot 27-28, Batu Ampar, Batam, Kepulauan Riau.

2. Jadwal Penelitian

Jadwal Penelitian adalah rincian waktu kegiatan yang dilakukan oleh peneliti mulai awal hingga akhir penelitian. Adapun jadwal pada penelitian ini yaitu:

Tabel 3.2 Jadwal Penelitian

Kegiatan	Tahun 2023												Tahun 2024			
	Sep		Okt				Nov				Des		Jan			
Pengajuan Judul Skripsi	■	■														
Penyusunan Bab I, Wawancara Kebutuhan Sistem dan Pembuatan Rancangan Sistem			■	■	■											
Penyusunan Bab II, Perancangan dan Pembuatan API					■	■	■	■								
Penyusunan Bab III, Perancangan dan Pembuatan <i>User Interface</i> , Integrasi API dengan <i>User Interface</i>									■	■	■	■	■			
<i>Testing</i> dan Implementasi API Serta Penyusunan Bab IV dan Bab V													■	■	■	■

Sumber : Data olahan peneliti (2023)