

BAB IV

ANALISA DAN PERANCANGAN SISTEM

4.1 Analisa Permasalahan

Hasil observasi dan wawancara yang dilakukan di UMKM Sempol Ayam Ratih di Tegal menunjukkan bahwa pencatatan laporan keuangan dan pengelolaan presensi karyawan masih dilakukan secara manual dan belum terintegrasi dalam satu sistem. Hal ini menyebabkan berbagai masalah dalam pengelolaan data operasional, terutama dalam proses pencatatan, penyimpanan, pencarian, dan rekapitulasi data.

Pada proses presensi karyawan, pencatatan kehadiran masih memanfaatkan grup WhatsApp. Karyawan melakukan presensi dengan mengirim pesan beserta foto sebagai bukti kehadiran. Cara tersebut menimbulkan kendala karena data presensi bercampur dengan percakapan lain sehingga pencarian data dan proses rekapitulasi kehadiran menjadi lebih sulit dilakukan. Selain itu, penyimpanan foto presensi secara terus-menerus membutuhkan kapasitas penyimpanan yang cukup besar.

Permasalahan lain terdapat pada pencatatan laporan keuangan, admin masih menggunakan buku catatan untuk mencatat transaksi harian, kemudian data tersebut diperiksa kembali oleh *owner* sebelum dipindahkan ke *Microsoft Excel*. Proses yang dilakukan secara bertahap dan menggunakan media yang berbeda-beda ini membuat pengelolaan data menjadi lambat serta kurang efisien. Selain itu, penggunaan *Microsoft Excel* berisiko menimbulkan

kesalahan input, kerusakan formula, serta perbedaan data antar file yang digunakan.

Transaksi yang berasal dari mitra *franchise* maupun non-mitra juga masih dicatat secara manual, sehingga proses monitoring kondisi keuangan belum dapat dilakukan secara langsung dan terpusat. Akibatnya, pencarian data transaksi tertentu dan penyusunan laporan keuangan membutuhkan waktu lebih lama dibandingkan jika data tersimpan dalam sistem yang terintegrasi.

Berdasarkan hasil analisis permasalahan yang telah dilakukan, UMKM Sempol Ayam Ratih memerlukan sebuah sistem informasi yang mampu menggabungkan pengelolaan presensi karyawan dan laporan keuangan dalam satu aplikasi. Penerapan sistem ini diharapkan dapat mendukung penyimpanan data yang terintegrasi, mempermudah pengelolaan informasi, meningkatkan ketelitian dalam pencatatan, serta mempersingkat proses pemantauan dan penyusunan rekapitulasi data operasional. Untuk memenuhi kebutuhan tersebut, fitur presensi berbasis *geolocation* dan pengelolaan laporan keuangan dikembangkan menggunakan konsep *Progressive Web App* (PWA) agar sistem dapat diakses dengan mudah melalui berbagai perangkat, memiliki kinerja yang ringan, serta sesuai dengan kebutuhan operasional UMKM.

4.2 Analisa Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan sebagai dasar untuk menentukan berbagai kebutuhan yang harus dipenuhi dalam pengembangan sistem sesuai

dengan kondisi yang ditemukan di UMKM Sempol Ayam Ratih Tegal. Berdasarkan hasil analisis, sistem yang akan dibangun harus mampu mengelola presensi karyawan berbasis *geolocation* dan pencatatan laporan keuangan dalam satu *platform* yang terintegrasi.

4.2.1 Analisis Kebutuhan Fungsional

Kebutuhan fungsional merupakan spesifikasi mengenai fitur dan layanan yang harus tersedia pada sistem agar seluruh proses bisnis dapat berjalan sesuai dengan kebutuhan pengguna. Fungsi utama ini harus ada agar sistem dapat berjalan dengan baik dan sesuai dengan tujuan penelitian. Adapun kebutuhan fungsional pada sistem yang dikembangkan adalah sebagai berikut:

1. Sistem dapat menjalankan proses *Login* untuk setiap pengguna berdasarkan hak akses mereka..
2. Sistem dapat mengelola data *user* dan karyawan.
3. Sistem dapat melakukan presensi masuk dan presensi pulang.
4. Sistem dapat melakukan validasi lokasi presensi menggunakan teknologi *geolocation* berdasarkan radius maksimal ± 50 meter dari titik koordinat lokasi UMKM yang telah ditentukan.
5. Sistem dapat menampilkan data riwayat presensi pribadi untuk karyawan serta rekapitulasi presensi seluruh karyawan untuk admin.
6. Sistem dapat mengelola prestasi harian pribadi untuk karyawan

7. Sistem dapat mengelola pengajuan cuti karyawan (proses pengajuan oleh karyawan serta validasi persetujuan atau penolakan oleh admin).
8. Sistem dapat mengelola validasi lokasi UMKM dan radius.
9. Sistem dapat mengelola profil saya
10. Modul buku kas umum menyediakan fasilitas untuk mencatat transaksi pemasukan dan pengeluaran, memantau saldo kas, serta menampilkan rekapitulasi transaksi keuangan.
11. Sistem mendukung penyimpanan seluruh data operasional ke dalam basis data yang terintegrasi sehingga data dapat dikelola secara terpusat.
12. Sistem dapat melakukan proses *Logout* pengguna.

4.2.2 Analisis Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan kebutuhan yang berkaitan dengan kualitas sistem agar sistem dapat berjalan dengan baik dan nyaman digunakan oleh pengguna. Adapun kebutuhan non-fungsional pada sistem yang dikembangkan adalah sebagai berikut:

1. Keamanan, sistem menerapkan mekanisme *autentikasi* untuk memverifikasi identitas pengguna sehingga setiap pengguna hanya dapat mengakses fitur sesuai dengan hak akses yang dimilikinya. Selain itu, data pengguna, presensi, transaksi, dan laporan keuangan disimpan pada basis data yang terintegrasi guna menjaga keamanan dan integritas data.

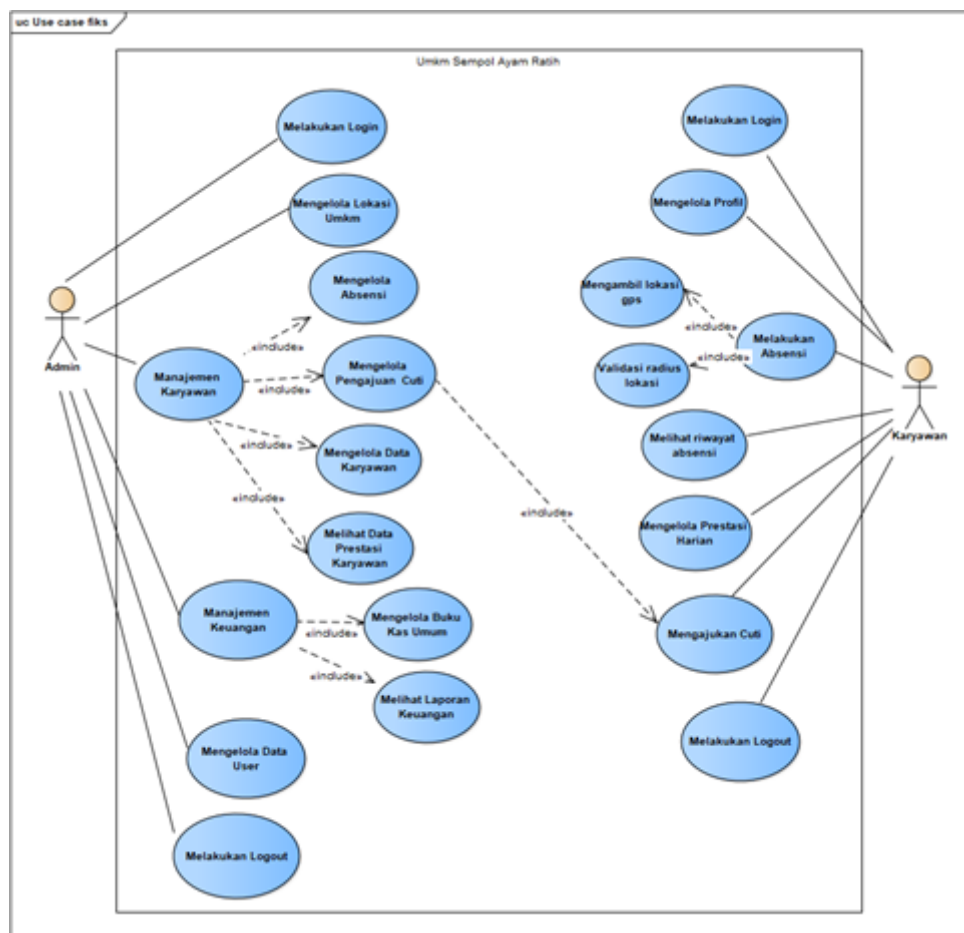
2. Sistem harus mudah digunakan oleh pengguna sehingga proses pengoperasian sistem dapat dilakukan tanpa kesulitan yang berarti.
3. Kompatibilitas, sistem harus dapat diakses melalui berbagai perangkat, baik *desktop* maupun *mobile*, selama menggunakan *browser* yang mendukung.
4. Sistem mampu menyimpan data operasional secara terpusat sehingga memudahkan proses pencarian dan pengelolaan data
5. Aksesibilitas, sistem dikembangkan menggunakan konsep *Progressive Web App* (PWA) sehingga dapat diakses melalui *browser* tanpa perlu melakukan instalasi aplikasi tambahan.

4.3 Perancangan Sistem

Perancangan Perancangan sistem merupakan tahapan yang dilakukan untuk menyusun gambaran mengenai cara kerja sistem sebelum proses implementasi dimulai. Pada penelitian ini, proses perancangan dilakukan dengan memanfaatkan *Unified Modeling Language* (UML) sebagai alat pemodelan sistem. Penggunaan UML membantu menggambarkan alur proses, fungsi, serta hubungan antarbagian dalam sistem secara lebih sistematis. Melalui tahap ini, rancangan sistem dapat menjadi acuan dalam proses pengembangan sehingga sistem yang dibangun sesuai dengan kebutuhan yang telah dianalisis.

4.3.1 Use case diagram

Use case diagram digunakan untuk menggambarkan hubungan antara aktor dengan fungsi-fungsi yang ada pada sistem. Pada penelitian ini, aktor yang berinteraksi dengan sistem terdiri atas admin dan karyawan.. Diagram use case ditunjukkan pada Gambar 4.1.



Gambar 4.1 Use Case Diagram

Berdasarkan diagram *use case* yang sudah dibuat, ada beberapa fungsi utama yang bisa diakses oleh aktor sesuai dengan hak

akses masing-masing. Berikut adalah deskripsi *use case* pada sistem seperti yang ditunjukkan pada Tabel 4.1.

Tabel 4.1 Deskripsi *Use Case*

No	Use Case	Actor	Deskripsi	Relasi
1.	<i>Login</i>	Admin, Karyawan.	Proses autentikasi pengguna untuk mendapatkan hak akses ke dalam sistem sesuai dengan level akun masing-masing.	
2.	Mengelola Profil Saya	Karyawan	Proses pengelolaan profil saya yang mencakup pembaruan data, seperti nama, foto profil, dan kata sandi.	
3.	Melakukan Presensi	Karyawan	Proses pencatatan kehadiran karyawan berupa presensi masuk dan presensi pulang yang dilakukan melalui sistem berbasis <i>geolocation</i>	<< <i>include</i> >> Mengambil lokasi GPS dan << <i>include</i> >> Validasi radius lokasi
4.	Melihat Riwayat Presensi	Karyawan	Menampilkan riwayat presensi pribadi milik karyawan	

No	Use Case	Actor	Deskripsi	Relasi
5.	Mengelola Prestasi Harian	Karyawan	Proses pencatatan dan pengelolaan data prestasi harian yang telah dicapai oleh karyawan seperti jumlah adonan, jam lembur.	<<extend>> Melakukan presensi
6.	Mengajukan Cuti	Karyawan	Proses pengiriman permohonan cuti oleh karyawan kepada admin untuk mendapatkan persetujuan	
7.	Mengelola Lokasi UMKM	Admin	Proses pengelolaan data lokasi UMKM dan mengatur radius lokasi yang digunakan untuk <i>geolocation</i>	
8.	Mengelola Manajemen Karyawan	Admin	Mengelola data karyawan yang mencakup informasi karyawan, rekap presensi, pengajuan cuti, dan prestasi karyawan	<<include>> Mengelola Data Karyawan, <<include>> Melihat Rekap Presensi, <<include>> Mengelola Pengajuan Cuti, <<include>> Melihat Data Prestasi Karyawan

No	Use Case	Actor	Deskripsi	Relasi
9.	Melihat Rekap Presensi	Admin	Proses menampilkan keseluruhan data presensi karyawan	
10.	Mengelola Pengajuan Cuti	Admin	Proses verifikasi data pengajuan cuti karyawan untuk menentukan persetujuan atau penolakan cuti	<<include>> Mengajukan cuti
11.	Mengelola Data Karyawan	Admin	Mengelola data karyawan melalui penambahan, perubahan, dan penghapusan data.	
12.	Melihat Data Prestasi Karyawan	Admin	Proses menampilkan data prestasi harian yang telah diinput oleh karyawan.	
13.	Manajemen Keuangan	Admin	Proses pengelolaan data keuangan yang meliputi Buku Kas Umum dan laporan keuangan.	<<include>> Mengelola Buku Kas Umum dan <<include>> Melihat Laporan Keuangan
14.	Mengelola Buku Kas Umum	Admin	Proses pencatatan dan pengelolaan data pemasukan, pengeluaran, serta saldo kas pada laporan keuangan	

No	Use Case	Actor	Deskripsi	Relasi
15.	Melihat Laporan Keuangan	Admin	Proses menampilkan laporan keuangan berdasarkan data Buku Kas Umum yang telah tersimpan.	
16.	Mengelola Data <i>User</i>	Admin	Mengelola data akun <i>user</i> melalui penambahan, perubahan, dan penghapusan sesuai dengan hak akses yang dimiliki.	
17.	<i>Logout</i>	Admin, Karyawan	Melakukan <i>Logout</i> guna mengakhiri sesi penggunaan sistem secara aman. penggunaan aplikasi secara aman.	

Pada *use case* diagram terdapat beberapa relasi <<*include*>> dan <<*extend*>> antar *use case*, yaitu sebagai berikut:.

1. Melakukan presensi memiliki relasi <<*include*>> terhadap *use case* mengambil lokasi GPS dan validasi radius lokasi. Relasi ini menunjukkan bahwa kedua proses tersebut bersifat wajib. Saat karyawan melakukan presensi, sistem akan secara otomatis mengambil koordinat lokasi pengguna dan memvalidasi apakah

lokasi tersebut berada dalam radius yang diizinkan sebelum data presensi disimpan.

2. Manajemen keuangan menggunakan relasi `<<include>>` terhadap use case mengelola buku kas umum dan melihat laporan keuangan. Relasi ini menggambarkan bahwa kedua *use case* tersebut termasuk dalam ruang lingkup pengelolaan data keuangan pada sistem.
3. Manajemen karyawan memiliki relasi `<<include>>` terhadap *use case* mengelola data karyawan, melihat rekap presensi, mengelola pengajuan cuti, dan melihat data prestasi karyawan. Relasi ini menunjukkan bahwa *use case* manajemen karyawan berfungsi sebagai pengelompokan beberapa fitur yang berkaitan dengan pengelolaan data karyawan sehingga memudahkan pengorganisasian fungsi pada sistem.
4. Mengelola pengajuan cuti memiliki relasi `<<include>>` terhadap mengajukan cuti. Relasi ini menunjukkan bahwa admin tidak dapat melakukan proses verifikasi, persetujuan, atau penolakan cuti sebelum ada pengajuan cuti yang dikirim oleh karyawan. Admin harus mengakses data pengajuan tersebut terlebih dahulu.

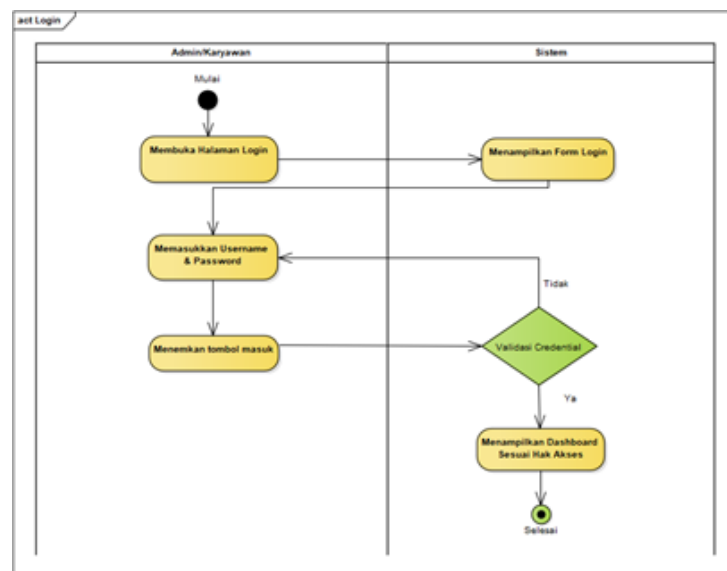
4.3.2 *Activity Diagram*

Activity Diagram digunakan untuk menggambarkan alur aktivitas atau proses kerja sistem dari awal hingga akhir. Melalui

diagram ini, alur interaksi antara pengguna dan sistem pada setiap tahapan proses dapat dipahami secara lebih jelas.

1. *Activity Diagram Login*

Activity Diagram Login menjelaskan alur yang dilalui pengguna saat melakukan proses masuk ke dalam sistem. Proses diawali dengan pengisian *username* dan *password*, kemudian sistem melakukan validasi terhadap data yang dimasukkan sebelum memberikan hak akses kepada pengguna. *Activity Diagram Login* ditunjukkan pada Gambar 4.2.



Gambar 4.2 *Activity Diagram Login*

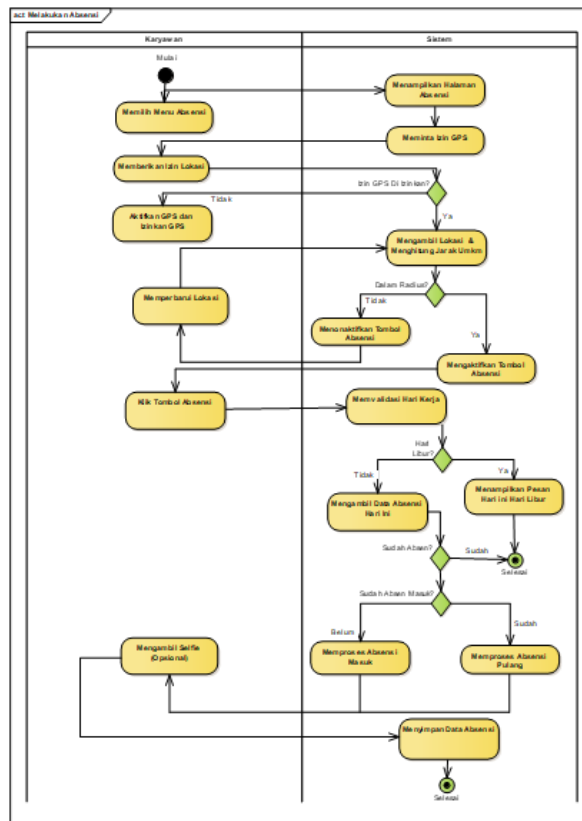
Alur proses *Login* pada sistem melalui *Activity Diagram* yang ditampilkan pada Gambar 4.2 menunjukkan urutan proses *Login* yang dilakukan oleh pengguna hingga sistem memberikan akses sesuai hak pengguna. Adapun alur proses *Login* dijelaskan sebagai berikut:

1. Pengguna membuka halaman *Login* pada sistem.
2. Sistem menampilkan *form Login* kepada pengguna.
3. Pengguna memasukkan *username* dan *password* pada form *Login*.
4. Pengguna menekan tombol *Login* untuk mengirim data *Login* ke sistem.
5. Setelah pengguna memasukkan *username* dan *password*, sistem menerima data tersebut untuk dilakukan proses verifikasi.
6. Sistem melakukan validasi data *Login* berdasarkan data pengguna yang tersimpan pada basis data.
7. Jika *username* dan *password* sesuai, sistem akan menampilkan *dashboard* sesuai hak akses pengguna.
8. Jika *username* dan *password* tidak sesuai, sistem akan menampilkan pesan gagal *Login* kepada pengguna.
9. Pengguna dapat kembali memasukkan data *Login* yang benar untuk mengakses sistem.
10. Proses *Login* selesai.

2. Activity Diagram Melakukan Presensi Masuk dan Pulang

Activity Diagram melakukan presensi menggambarkan alur proses presensi karyawan berbasis *geolocation* mulai dari pengambilan lokasi pengguna hingga penyimpanan data presensi ke dalam

sistem. *Activity Diagram* melakukan presensi ditunjukkan pada Gambar 4.3.



Gambar 4.3 *Activity Diagram* Melakukan Presensi

Gambar 4.3 *Activity Diagram* melakukan presensi menunjukkan urutan proses presensi yang dilakukan oleh karyawan dengan memanfaatkan lokasi GPS pengguna. Adapun alur proses melakukan presensi dijelaskan sebagai berikut:

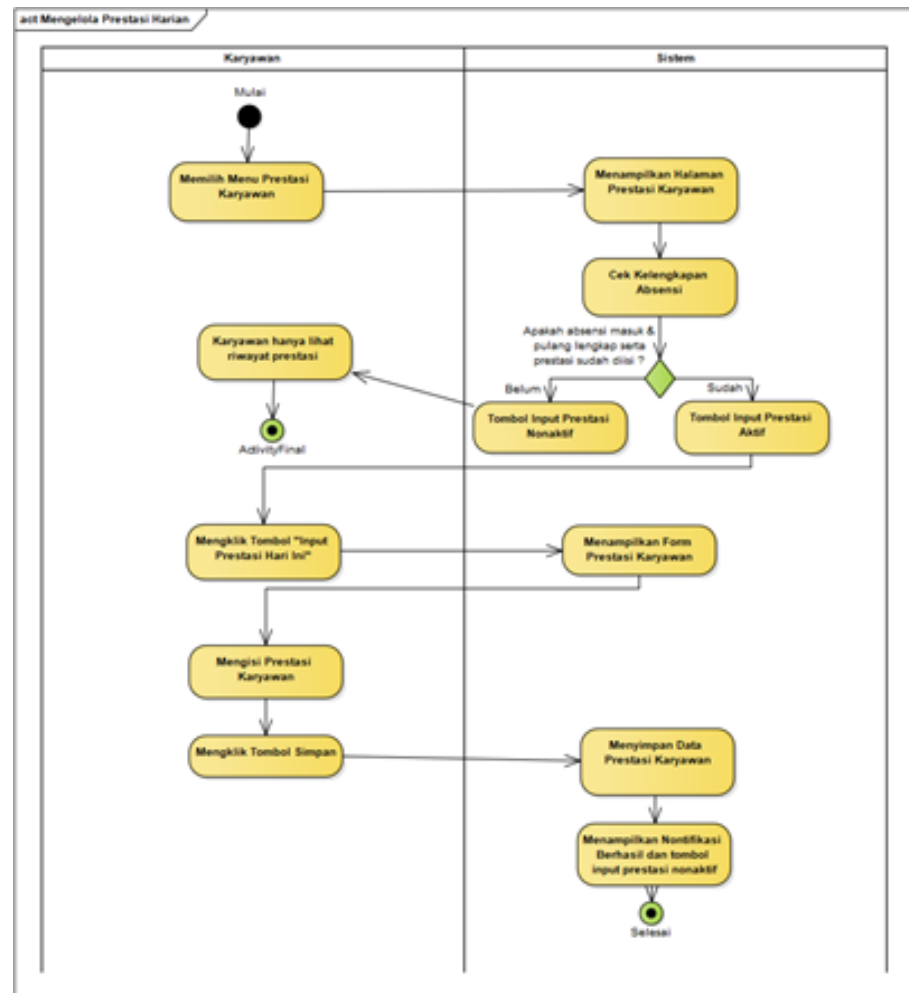
1. Karyawan memilih menu presensi pada sistem.
2. Sistem menampilkan halaman presensi kepada karyawan.
3. Sistem meminta izin akses GPS atau lokasi perangkat pengguna.
4. Karyawan memberikan izin akses lokasi kepada sistem.

5. Jika GPS belum aktif, karyawan diminta untuk mengaktifkan GPS dan memberikan izin lokasi.
6. Sistem mengambil lokasi pengguna dan menghitung jarak lokasi pengguna dengan lokasi UMKM.
7. Sistem melakukan validasi radius lokasi presensi.
8. Jika lokasi pengguna berada di luar radius yang ditentukan, maka tombol presensi akan dinonaktifkan.
9. Karyawan dapat memperbarui lokasi untuk mendapatkan lokasi terbaru.
10. Jika lokasi pengguna berada dalam radius yang ditentukan, maka sistem akan mengaktifkan tombol presensi.
11. Karyawan menekan tombol presensi untuk melakukan proses presensi.
12. Sistem melakukan validasi hari kerja untuk menentukan apakah hari tersebut merupakan hari kerja atau hari libur.
13. Jika hari tersebut merupakan hari libur, maka sistem akan menampilkan pesan bahwa presensi tidak dapat dilakukan.
14. Jika hari tersebut merupakan hari kerja, maka sistem akan mengambil data presensi karyawan pada hari tersebut.
15. Sistem memeriksa status presensi karyawan apakah sudah melakukan presensi masuk atau belum.
16. Jika karyawan belum melakukan presensi masuk, maka sistem akan memproses presensi masuk.

17. Jika karyawan sudah melakukan presensi masuk, maka sistem akan memproses presensi pulang.
18. Karyawan dapat melakukan pengambilan selfie sebagai dokumentasi presensi.
19. Sistem menyimpan data presensi ke dalam basis data.
20. Rangkaian proses presensi selesai.

3. *Activity* Diagram Mengelola Prestasi Harian

Activity diagram pengelolaan prestasi harian menunjukkan alur kerja karyawan untuk memasukkan prestasi harian. Diagram ini menunjukkan proses mulai dari memilih menu prestasi, pengecekan kelengkapan presensi, pengisian data prestasi, hingga penyimpanan data oleh sistem. *Activity* diagram mengelola prestasi harian ditunjukkan pada gambar 4.4.



Gambar 4.4 *Activity* Diagram Mengelola Prestasi Harian

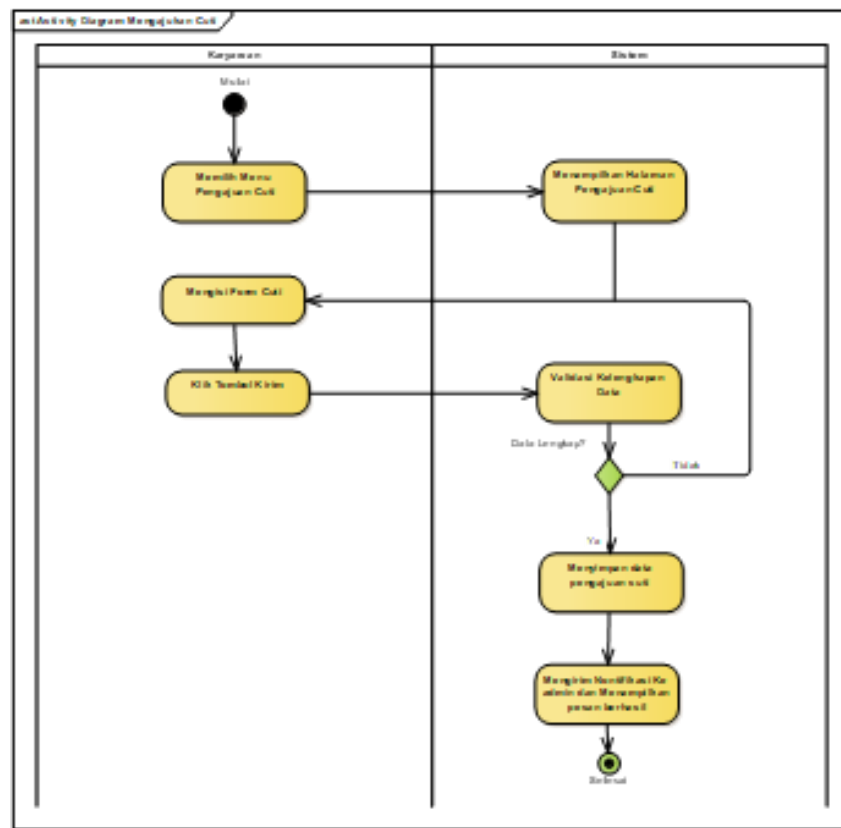
Gambar 4.4 *Activity* Diagram pengajuan cuti menunjukkan urutan proses pengajuan cuti yang dilakukan oleh karyawan melalui sistem. Adapun alur proses pengajuan cuti dijelaskan sebagai berikut:

1. Karyawan memilih menu prestasi karyawan.
2. Sistem menampilkan halaman prestasi karyawan.
3. Sistem melakukan pengecekan apakah presensi masuk dan presensi pulang telah lengkap serta memastikan prestasi hari tersebut belum pernah diinput.

4. Apabila persyaratan belum terpenuhi, tombol input prestasi berada dalam kondisi tidak aktif sehingga karyawan hanya dapat melihat riwayat prestasi.
5. Apabila seluruh persyaratan telah terpenuhi, tombol input prestasi hari ini diaktifkan.
6. Karyawan menekan tombol input prestasi hari ini.
7. Sistem menampilkan formulir pengisian prestasi harian.
8. Karyawan mengisi data prestasi yang telah dicapai.
9. Karyawan menekan tombol Simpan.
10. Sistem menyimpan data prestasi ke dalam basis data.
11. Sistem menampilkan notifikasi bahwa data prestasi berhasil disimpan serta menonaktifkan kembali tombol input prestasi untuk hari tersebut.
12. Proses selesai.

4. Activity Diagram Pengajuan Cuti

Activity Diagram pengajuan cuti menggambarkan alur proses pengajuan cuti yang dilakukan oleh karyawan mulai dari pengisian *form* pengajuan hingga data pengajuan cuti berhasil disimpan ke dalam sistem. *Activity* Diagram pengajuan cuti ditunjukkan pada Gambar 4.5.



Gambar 4.5 *Activity Diagram* Pengajuan Cuti

Activity Diagram pengajuan cuti menunjukkan urutan proses pengajuan cuti yang dilakukan oleh karyawan melalui sistem.

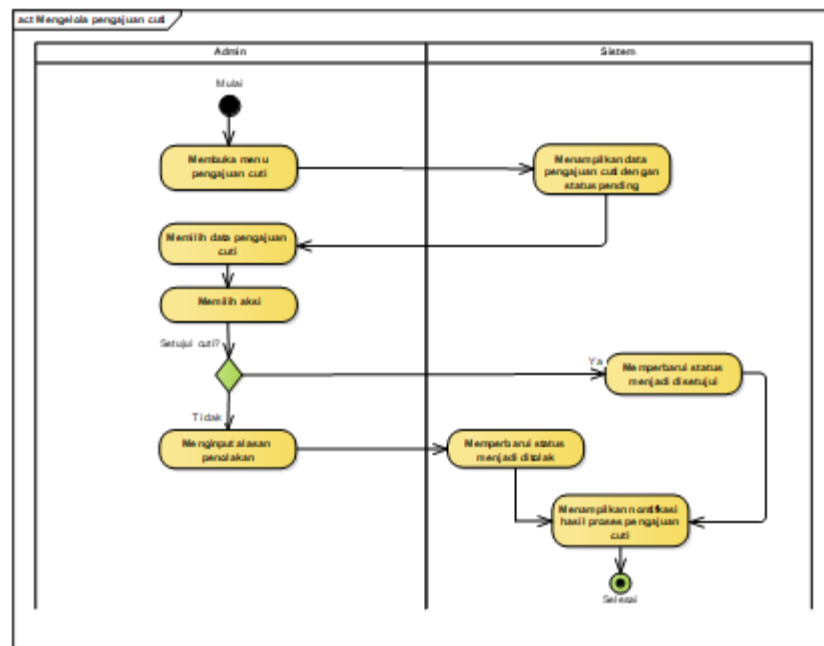
Adapun alur proses pengajuan cuti dijelaskan sebagai berikut:

1. Karyawan memilih menu pengajuan cuti pada sistem.
2. Sistem menampilkan halaman pengajuan cuti kepada karyawan.
3. Karyawan mengisi form pengajuan cuti sesuai data yang dibutuhkan.
4. Karyawan menekan tombol kirim untuk mengirim data pengajuan cuti.
5. Sistem melakukan validasi kelengkapan data pengajuan cuti.

6. Jika data pengajuan cuti tidak lengkap, maka proses pengajuan cuti tidak dapat dilanjutkan.
7. Jika data pengajuan cuti lengkap, maka sistem akan menyimpan data pengajuan cuti ke dalam basis data.
8. Sistem mengirim notifikasi pengajuan cuti kepada admin.
9. Sistem menampilkan pesan bahwa pengajuan cuti berhasil dikirim.
10. Proses pengajuan cuti selesai.

5. *Activity* Diagram Mengelola Pengajuan Cuti

Activity Diagram mengelola pengajuan cuti menggambarkan alur proses pengelolaan data pengajuan cuti yang dilakukan oleh admin mulai dari melihat data pengajuan cuti hingga memberikan keputusan persetujuan atau penolakan pengajuan cuti. *Activity* Diagram mengelola pengajuan cuti ditunjukkan pada Gambar 4.6.



Gambar 4.6 *Activity* Diagram Mengelola Pengajuan Cuti

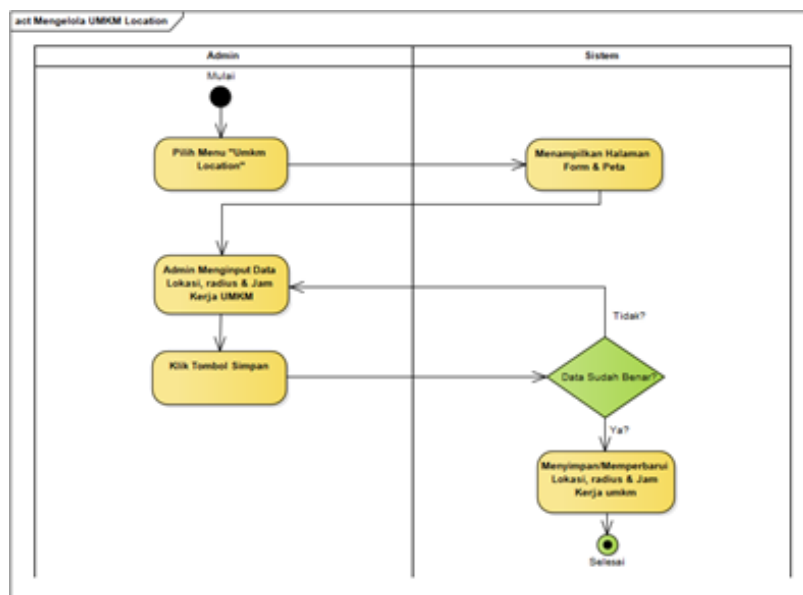
Activity Diagram mengelola pengajuan cuti menunjukkan urutan proses pengelolaan pengajuan cuti yang dilakukan oleh admin pada sistem. Adapun alur proses mengelola pengajuan cuti dijelaskan sebagai berikut:

1. Admin membuka menu pengajuan cuti pada sistem.
2. Sistem menampilkan data pengajuan cuti dengan status *pending*.
3. Admin memilih data pengajuan cuti yang akan diproses.
4. Admin memilih aksi pengajuan cuti berupa persetujuan atau penolakan pengajuan cuti.
5. Jika pengajuan cuti disetujui, maka sistem akan memperbarui status pengajuan menjadi disetujui.

6. Jika pengajuan cuti ditolak, maka admin diminta untuk menginput alasan penolakan pengajuan cuti.
7. Sistem memperbarui status pengajuan cuti menjadi ditolak.
8. Sistem menampilkan notifikasi hasil proses pengajuan cuti kepada pengguna.
9. Proses pengelolaan pengajuan cuti selesai.

6. *Activity Diagram* Mengelola Lokasi UMKM

Activity Diagram mengelola lokasi UMKM menggambarkan alur proses pengelolaan data lokasi dan radius *geofencing* yang dilakukan oleh admin. Diagram ini menunjukkan urutan aktivitas mulai dari pemilihan menu hingga penyimpanan atau pembaruan data lokasi UMKM ke dalam basis data. *Activity Diagram* mengelola lokasi UMKM ditunjukkan pada Gambar 4.7.



Gambar 4.7 *Activity Diagram* Mengelola Lokasi UMKM

Activity Diagram mengelola lokasi UMKM menunjukkan urutan proses mengelola lokasi UMKM yang dilakukan oleh admin melalui sistem. Adapun alur proses mengelola lokasi UMKM dijelaskan sebagai berikut:

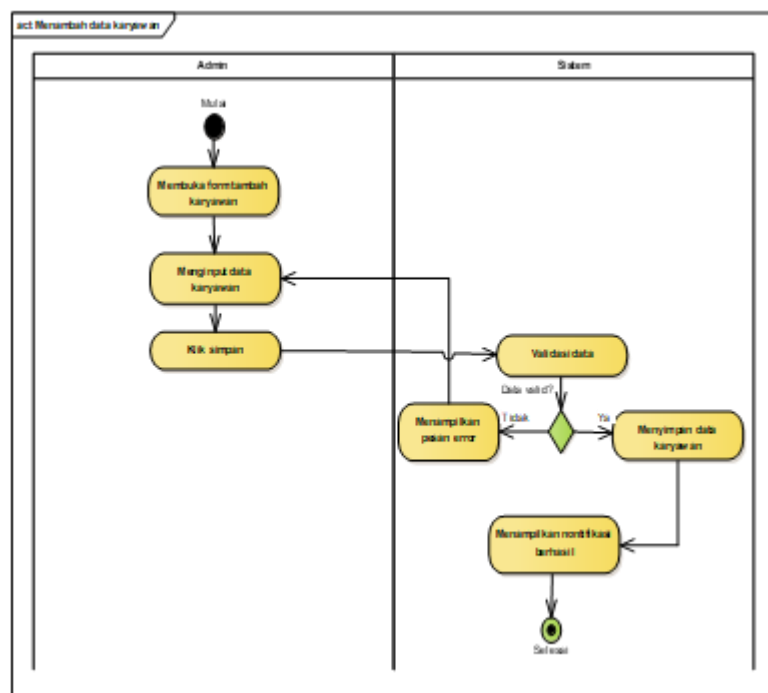
1. Admin memilih menu “UMKM *Location*” pada sistem.
2. Sistem menampilkan halaman form dan peta lokasi UMKM.
3. Admin menginput data lokasi (koordinat), radius, dan jam kerja UMKM.
4. Admin mengklik tombol Simpan..
5. Sistem melakukan validasi terhadap kelengkapan dan kebenaran data yang diinput.
6. Jika data belum benar atau belum lengkap, sistem akan menampilkan notifikasi dan admin dapat memperbaiki data tersebut.
7. Jika data sudah benar, sistem akan menyimpan atau memperbarui data lokasi, radius, serta jam kerja UMKM ke dalam basis data..
8. Proses pengelolaan lokasi UMKM selesai.

7. *Activity* Diagram Mengelola Data Karyawan

Activity Diagram mengelola data karyawan menjelaskan alur pengelolaan data karyawan yang dilakukan oleh admin. Proses tersebut meliputi penambahan, perubahan, dan penghapusan data karyawan.

a. Menambah Data Karyawan

Proses penambahan data karyawan oleh admin dijelaskan dalam *Activity Diagram*. Diagram tersebut memperlihatkan setiap tahapan mulai dari pengisian data hingga penyimpanan ke dalam sistem. *Activity Diagram* menambah data karyawan ditunjukkan pada Gambar 4.8.



Gambar 4.8 *Activity Diagram* Menambah Data Karyawan

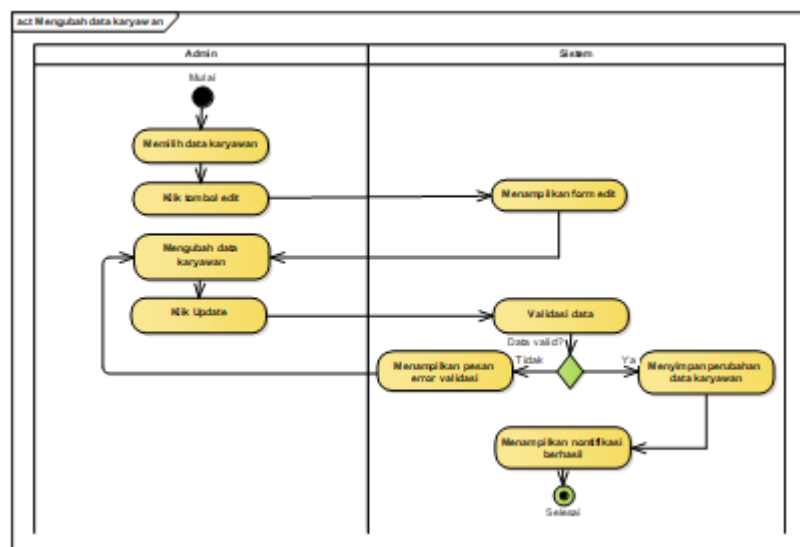
Alur penambahan data karyawan oleh admin dijelaskan melalui *Activity Diagram*. Proses tersebut dimulai dari pengisian formulir data karyawan hingga informasi berhasil tersimpan di dalam sistem. Alur prosesnya adalah sebagai berikut:

1. Admin membuka *form* tambah karyawan.

2. Admin menginput data karyawan yang diperlukan.
3. Admin mengklik tombol Simpan.
4. Sistem melakukan validasi data.
5. Jika data belum lengkap atau belum benar, sistem menampilkan pesan error.
6. Apabila seluruh data telah memenuhi validasi, sistem merekam informasi karyawan ke basis data dan memberikan pemberitahuan bahwa proses penambahan data berhasil.

b. Mengubah Data Karyawan

Activity Diagram ini menjelaskan tahapan pembaruan data karyawan oleh admin, dimulai dari memilih data yang akan diubah sampai proses penyimpanan perubahan selesai dilakukan. *Activity* Diagram ini dapat dilihat pada Gambar 4.9.



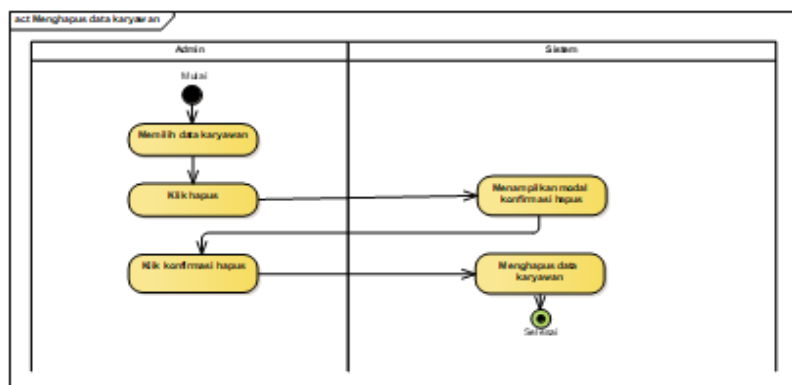
Gambar 4.9 *Activity* Diagram Mengubah Data Karyawan

Activity Diagram mengubah data karyawan menunjukkan urutan aktivitas pengeditan data oleh admin. Alur prosesnya adalah sebagai berikut:

1. Admin memilih data karyawan yang akan diubah.
2. Admin mengklik tombol Edit.
3. Sistem menampilkan *form* edit.
4. Admin mengubah data karyawan sesuai kebutuhan.
5. Admin mengklik tombol *Update*.
6. Sistem melakukan validasi data.
7. Jika data belum benar, sistem menampilkan pesan error validasi.
8. Jika data sudah benar, sistem menyimpan perubahan data karyawan dan menampilkan notifikasi berhasil.

c. Menghapus Data Karyawan

Activity Diagram menghapus data karyawan menggambarkan proses penghapusan data karyawan. *Activity* Diagram ini dapat dilihat pada Gambar 4.10.



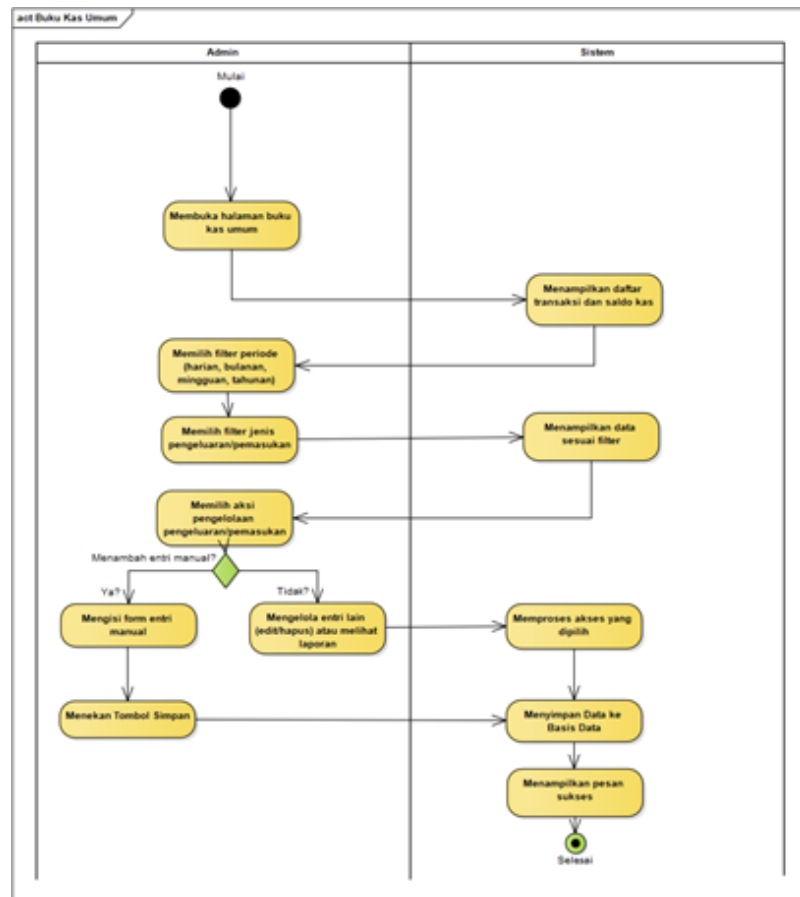
Gambar 4.10 *Activity* Diagram Menghapus Data Karyawan

Activity Diagram menghapus data karyawan menunjukkan urutan aktivitas penghapusan data oleh admin. Alur prosesnya adalah sebagai berikut:

1. Admin memilih data karyawan yang akan dihapus.
2. Admin mengklik tombol Hapus.
3. Sistem menampilkan modal konfirmasi hapus.
4. Admin mengklik konfirmasi hapus.
5. Sistem menghapus data karyawan dari basis data.
6. Proses penghapusan data karyawan selesai.

8. *Activity* Diagram Mengelola Buku Kas Umum

Activity Diagram mengelola buku kas umum menggambarkan alur proses pengelolaan data pemasukan dan pengeluaran kas yang dilakukan oleh admin, mulai dari pencatatan transaksi hingga penyimpanan data ke dalam sistem. *Activity* Diagram mengelola buku kas umum dapat dilihat pada Gambar 4.11.



Gambar 4.11 Activity Diagram Mengelola Buku Kas Umum

Activity Diagram mengelola buku kas umum pada Gambar 4.11 menunjukkan urutan proses pengelolaan data kas umum yang dilakukan oleh admin pada sistem. Adapun alur proses mengelola buku kas umum dijelaskan sebagai berikut:

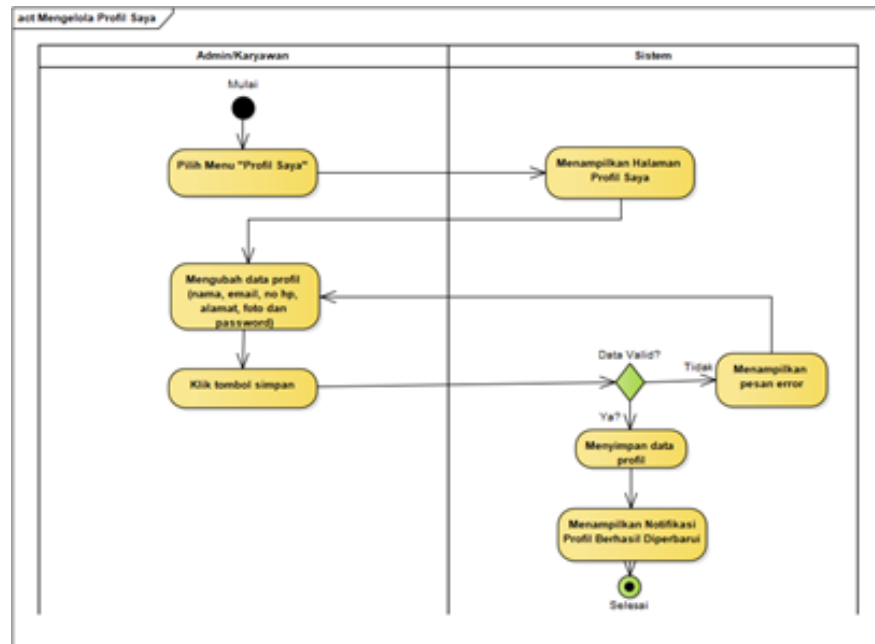
1. Admin membuka halaman buku kas umum pada sistem.
2. Sistem menampilkan daftar transaksi beserta saldo kas kepada admin.
3. Admin memilih filter periode transaksi, seperti harian, mingguan, bulanan, atau tahunan.

4. Admin memilih filter jenis transaksi berupa pemasukan atau pengeluaran.
 5. Sistem menampilkan data transaksi sesuai filter yang dipilih oleh admin.
 6. Admin memilih aksi pengelolaan data kas berupa menambah, mengubah, menghapus, atau melihat laporan transaksi.
 7. Jika admin menambahkan data kas secara manual, maka admin mengisi *form* entri transaksi.
 8. Admin menekan tombol simpan untuk menyimpan data transaksi kas.
 9. Sistem memproses aksi pengelolaan data kas yang dipilih oleh admin.
 10. Sistem merekam data transaksi kas ke dalam basis data..
 11. Sistem menampilkan pesan bahwa proses pengelolaan buku kas umum berhasil dilakukan.
 12. Proses pengelolaan buku kas umum selesai.
9. *Activity* Diagram Mengelola Profil Saya

Activity Diagram mengelola profil saya menjelaskan tahapan yang dilakukan karyawan dalam mengelola informasi akun pada sistem. Melalui fitur ini, pengguna dapat memperbarui data profil, mengganti maupun menghapus foto profil, serta mengubah kata sandi. Setiap perubahan yang dilakukan akan diperiksa terlebih

dahulu oleh sistem sebelum disimpan ke dalam basis data. *Activity*

Diagram mengelola profil saya dapat dilihat pada Gambar 4.12.



Gambar 4.12 *Activity* Diagram Mengelola Profil Saya

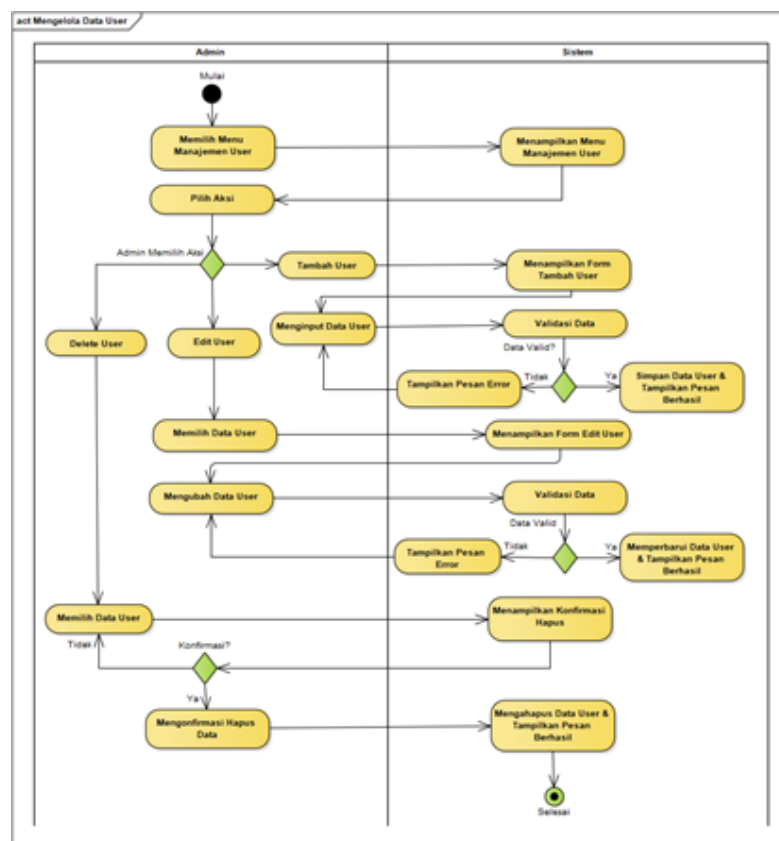
Activity diagram mengelola profil saya menunjukkan urutan proses mengelola profil saya yang dilakukan oleh admin melalui sistem. Adapun alur proses mengelola profil saya dijelaskan sebagai berikut:

1. Karyawan membuka menu Profil.
2. Sistem menampilkan data profil.
3. Karyawan melakukan pembaruan informasi profil, mengganti atau menghapus foto profil, maupun mengubah kata sandi sesuai kebutuhan.
4. Sistem melakukan validasi data.

5. Jika data valid, sistem menyimpan perubahan dan menampilkan notifikasi berhasil.
6. Jika data tidak valid, sistem menampilkan pesan kesalahan.
7. Proses mengelola profil selesai.

10. Activity Diagram Mengelola Data User

Activity diagram mengelola data *user* menjelaskan alur yang dilakukan oleh admin dalam mengelola informasi pengguna pada sistem. Kegiatan yang dapat dilakukan meliputi penambahan, perubahan, dan penghapusan data pengguna melalui menu manajemen *user*. *Activity* diagram yang menjelaskan proses tersebut dapat dilihat pada Gambar 4.13.



Gambar 4.13 Activity Diagram Mengelola Data User

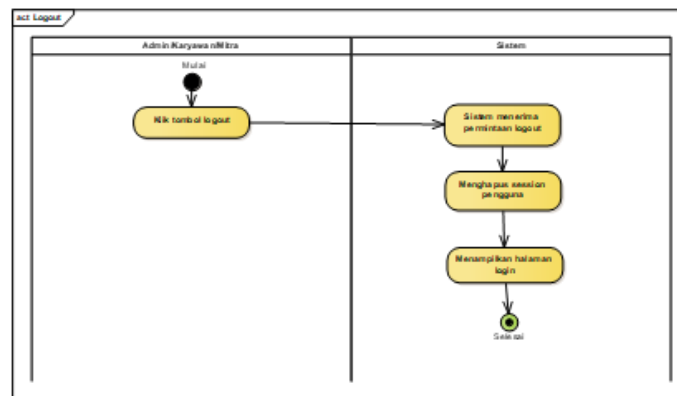
Activity Diagram mengelola data user menjelaskan tahapan yang dilakukan admin dalam mengelola akun pengguna melalui menu manajemen *user*. Proses yang tersedia meliputi penambahan, pembaruan, dan penghapusan data pengguna sesuai dengan kebutuhan pengelolaan sistem. Adapun tahapan proses tersebut dijelaskan sebagai berikut:

1. Admin memilih menu manajemen *user* pada sistem.
2. Sistem menampilkan halaman manajemen *user*.
3. Admin memilih aksi yang akan dilakukan, yaitu tambah *user*, edit *user*, atau delete *user*.
4. Jika memilih tambah *user*, sistem menampilkan formulir tambah *user* dan admin menginput data user.
5. Sistem melakukan validasi terhadap data yang diinput.
6. Jika data belum benar atau belum lengkap, sistem menampilkan pesan kesalahan sehingga admin dapat memperbaiki data yang diinput.
7. Apabila seluruh data telah memenuhi proses validasi, sistem merekam informasi pengguna ke dalam basis data dan memberikan notifikasi bahwa penambahan data berhasil dilakukan.
8. Jika memilih edit *user*, sistem menampilkan formulir edit *user* dan admin mengubah data *user* sesuai kebutuhan.
9. Sistem melakukan validasi terhadap data yang diubah.

10. Setelah data hasil perubahan dinyatakan valid, sistem menyimpan pembaruan informasi pengguna pada basis data serta menampilkan notifikasi bahwa perubahan data telah berhasil disimpan..
11. Jika memilih delete *user*, sistem menampilkan konfirmasi penghapusan data.
12. Setelah admin mengonfirmasi penghapusan, sistem menghapus data *user* dari basis data dan menampilkan notifikasi bahwa data *user* berhasil dihapus.
13. Proses mengelola data *user* selesai.

11. Activity Diagram Logout

Activity Diagram Logout menggambarkan proses pengakhiran sesi pengguna setelah menggunakan sistem. Pada proses ini pengguna yang telah *Login* dapat mengakhiri aksesnya ke sistem sehingga tidak dapat mengakses halaman-halaman yang membutuhkan autentikasi. *Activity Diagram Logout* ditunjukkan pada Gambar 4.14.



Gambar 4.14 *Activity Diagram Logout*

Berdasarkan *Activity Diagram Logout*, proses *Logout* dimulai ketika pengguna yang sedang dalam keadaan *Login* memilih opsi keluar dari sistem. Sistem kemudian menghapus data session pengguna dan mengarahkan kembali ke halaman *Login*.

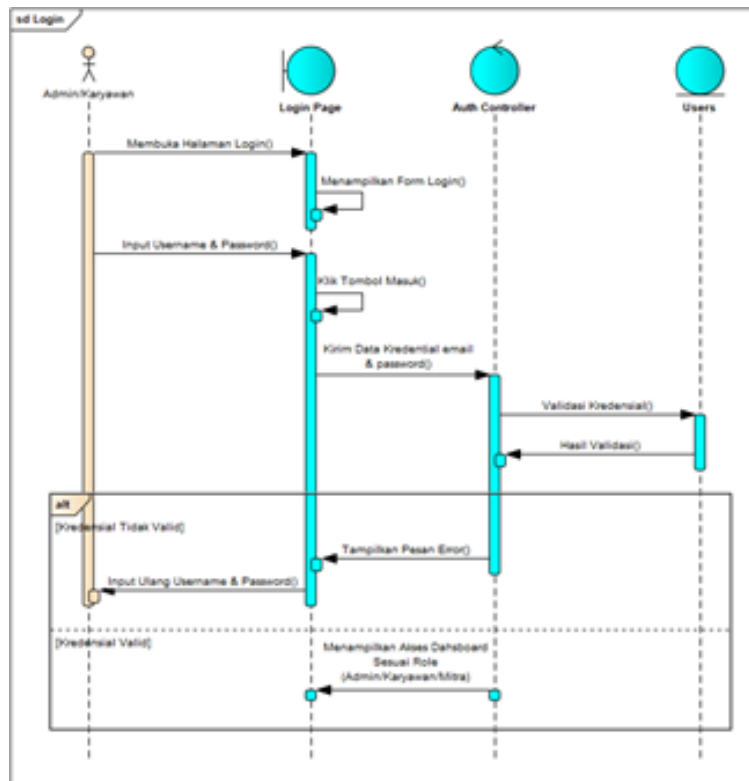
4.3.3 *Sequence diagram*

Sequence diagram digunakan untuk menggambarkan interaksi antar objek dalam sistem berdasarkan urutan waktu proses berjalan. Diagram ini menunjukkan aliran pesan atau komunikasi antara aktor, *boundary*, *control*, dan *entity* dalam menjalankan suatu fungsi pada sistem. *Sequence* diagram pada penelitian ini berfungsi untuk memberikan gambaran mengenai interaksi antarobjek pada setiap fitur utama yang tersedia di dalam sistem.

1. *Sequence Diagram Login*

Sequence Diagram Login menggambarkan interaksi antara pengguna dengan sistem dalam proses *autentikasi Login*. Diagram ini menunjukkan aliran pesan antara aktor, *boundary*,

control, dan *entity* hingga pengguna memperoleh hak akses sesuai *role* yang dimiliki. *Sequence Diagram Login* pada sistem disajikan pada Gambar 4.15.



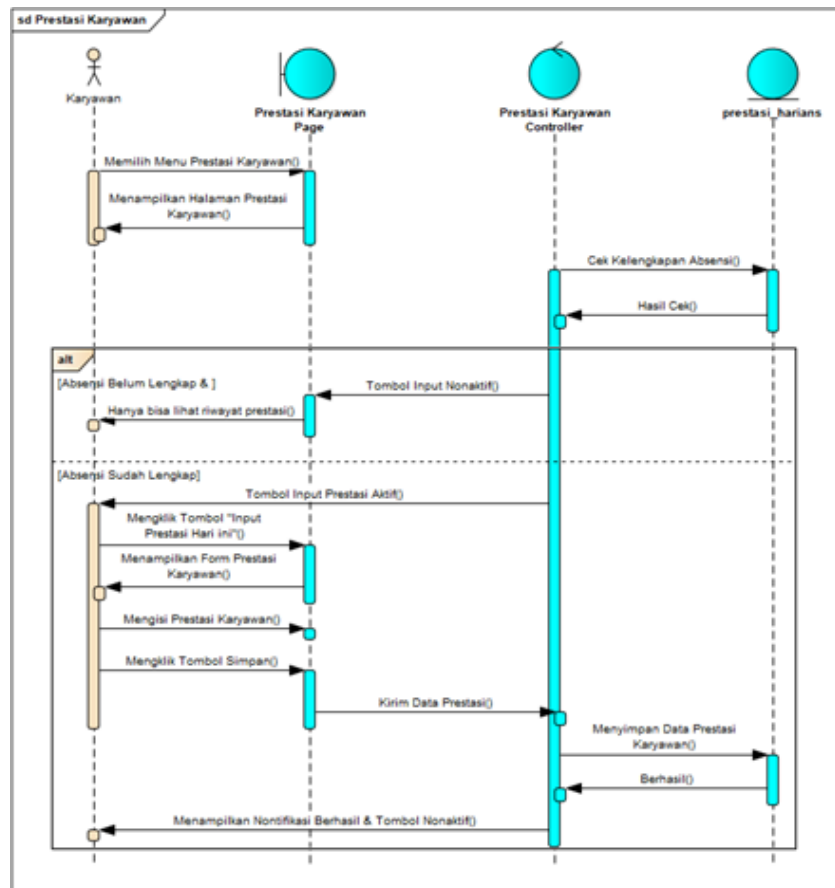
Gambar 4.15 *Sequence Diagram Login*

Sequence Diagram Login menggambarkan proses interaksi antara pengguna dengan sistem pada saat melakukan autentikasi *Login*. Proses dimulai ketika pengguna memasukkan data *Login* melalui *boundary*, kemudian data tersebut diteruskan ke *control* untuk dilakukan validasi ke *entity* atau basis data. Hasil validasi yang diperoleh selanjutnya digunakan sistem untuk menentukan hak akses pengguna. Jika data *Login* sesuai, maka sistem akan menampilkan *dashboard* sesuai *role* pengguna. Adapun jika data

yang berbasis *geolocation*. Proses diawali saat karyawan mengakses fitur presensi, di mana sistem secara otomatis meminta verifikasi izin lokasi dari perangkat pengguna. Setelah izin lokasi diperoleh, sistem melaksanakan pengambilan lokasi serta menghitung jarak lokasi pengguna terhadap lokasi UMKM. Hasil perhitungan itu digunakan oleh sistem untuk menetapkan apakah proses presensi dapat dilaksanakan atau tidak. Sistem kemudian melakukan validasi data kehadiran dan pemeriksaan hari kerja melalui *control* dan *entity* sebelum data kehadiran disimpan dalam basis data. Dalam proses ini, sistem juga mampu menyimpan data presensi masuk maupun pulang berdasarkan status presensi karyawan pada hari itu.

3. **Sequence diagram Prestasi Harian**

Sequence diagram pengelolaan prestasi harian mengilustrasikan alur interaksi antara karyawan dan sistem dalam memproses data prestasi harian. Diagram ini menunjukkan aliran pesan antara aktor, *boundary*, *control*, dan *entity* mulai dari proses membuka halaman prestasi hingga penyimpanan data prestasi harian. *Sequence* diagram mengelola prestasi harian pada sistem ditunjukkan pada Gambar 4.17.



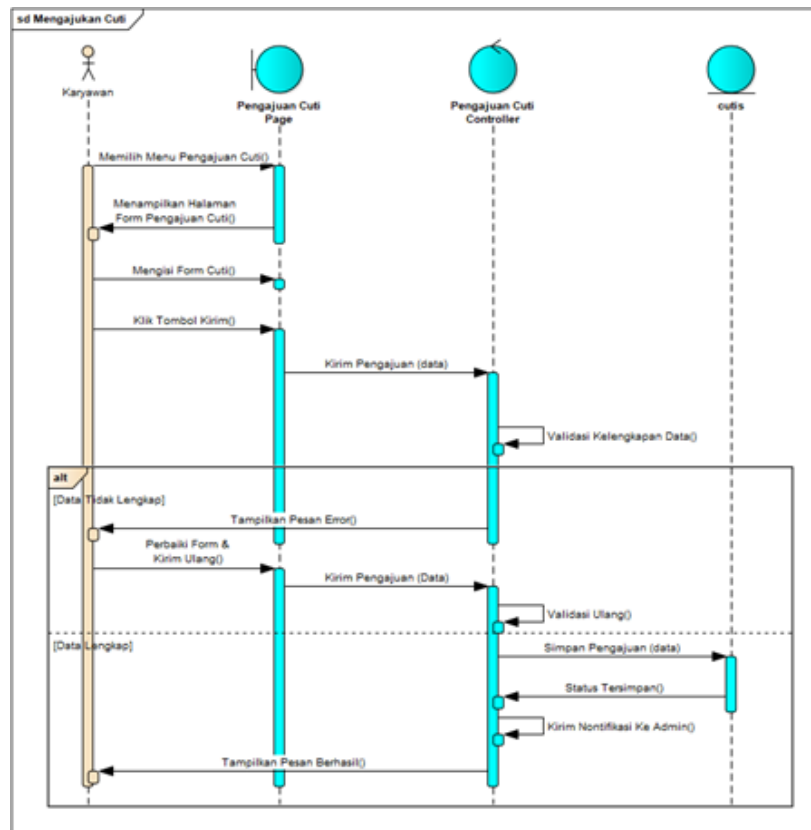
Gambar 4.17 *Sequence Diagram* Mengelola Prestasi Karyawan

Berdasarkan Gambar 4.17, *sequence diagram* mengelola prestasi harian menggambarkan proses interaksi antara karyawan dengan sistem pada saat melakukan input prestasi harian. Proses diawali ketika karyawan mengakses menu Prestasi Karyawan, kemudian sistem menampilkan halaman prestasi harian. Selanjutnya, sistem memeriksa kelengkapan data presensi melalui basis data. Apabila presensi belum memenuhi persyaratan, tombol Input Prestasi akan dinonaktifkan sehingga karyawan hanya dapat melihat riwayat prestasi yang telah tersimpan. Sebaliknya, jika data presensi telah lengkap, sistem akan mengaktifkan tombol Input

Prestasi Hari Ini, sehingga karyawan dapat melakukan pengisian data prestasi harian. Setelah form prestasi ditampilkan, karyawan mengisi data prestasi kemudian menekan tombol Simpan. Data tersebut selanjutnya dikirim oleh *boundary* ke *control* untuk diproses, kemudian *control* menyimpan data prestasi ke dalam basis data melalui *entity*. Setelah proses penyimpanan berhasil, sistem menampilkan notifikasi berhasil kepada karyawan dan tombol input prestasi kembali dinonaktifkan.

4. *Sequence Diagram Pengajuan Cuti*

Sequence diagram pengajuan cuti menggambarkan interaksi antara karyawan dengan sistem dalam proses pengajuan cuti. Diagram ini menunjukkan aliran pesan antara aktor, *boundary*, *control*, dan *entity* mulai dari proses pengisian form cuti hingga penyimpanan data pengajuan. *Sequence* diagram pengajuan cuti pada sistem ditunjukkan pada Gambar 4.18.



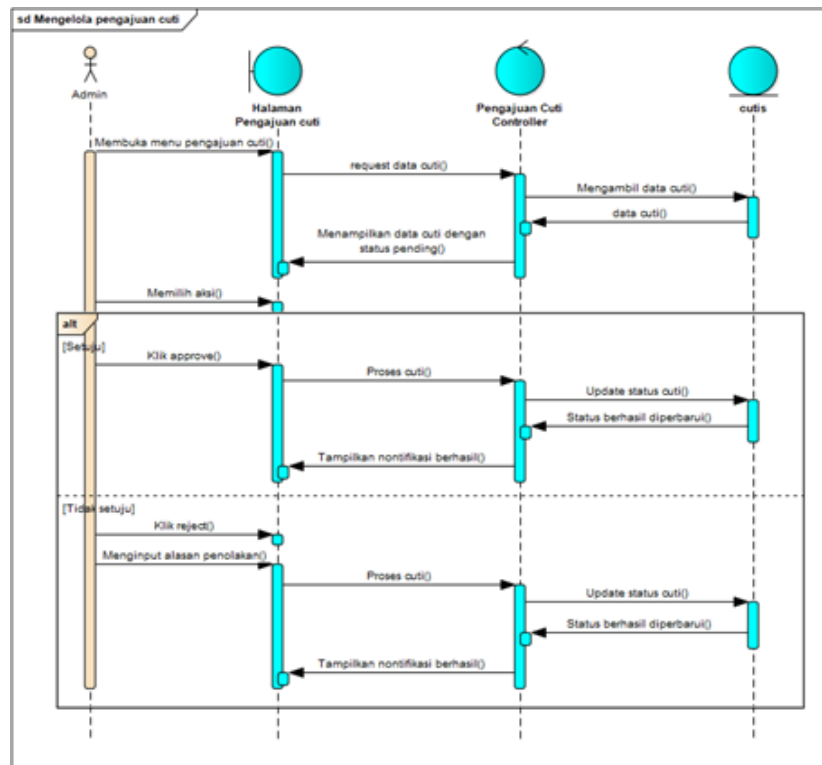
Gambar 4.18 Sequence Diagram Pengajuan Cuti

Sequence diagram pengajuan cuti menggambarkan proses interaksi antara karyawan dengan sistem pada saat pengajuan cuti. Proses dimulai ketika karyawan mengakses halaman *form* pengajuan cuti melalui menu yang tersedia pada sistem. Setelah form ditampilkan, karyawan mengisi informasi cuti yang dibutuhkan. Data tersebut kemudian dikirim oleh *boundary* ke *control* untuk dilakukan validasi kelengkapan, jika terdapat data yang kurang lengkap, sistem akan menunjukkan pesan kesalahan dan karyawan bisa memperbaiki formulir serta mengirimkan ulang permohonan, sebaliknya jika data dinyatakan lengkap, *control* menyimpan data pengajuan cuti ke dalam basis data

melalui entity dengan status "tersimpan". Selain itu, sistem secara otomatis meneruskan notifikasi kepada admin untuk memberitahukan adanya pengajuan cuti baru yang membutuhkan pemrosesan lebih lanjut. Tahapan diakhiri dengan tampilan pesan konfirmasi sukses kepada karyawan bahwa pengajuan cuti telah berhasil diterima oleh sistem.

5. *Sequence Diagram* Mengelola Pengajuan Cuti

Sequence diagram mengelola pengajuan cuti menunjukkan interaksi antara admin dengan sistem dalam proses persetujuan atau penolakan cuti karyawan. Diagram ini menggambarkan aliran informasi antara aktor, *boundary*, *control*, dan *entity* yang dimulai dari proses pengambilan data cuti sampai pembaruan status pengajuan. *Sequence* diagram untuk pengelolaan pengajuan cuti dalam sistem ditampilkan pada Gambar 4.19.



Gambar 4.19 Sequence Diagram Mengelola Pengajuan Cuti

Berdasarkan Gambar 4.19 Sequence diagram untuk pengelolaan pengajuan cuti menunjukkan proses interaksi antara admin dan sistem saat menyetujui atau menolak permohonan cuti karyawan. Proses dimulai saat admin membuka menu pengajuan cuti, setelah itu sistem mengambil data cuti dari database melalui *control* dan *entity*. Data yang ditampilkan hanya permohonan cuti dengan status "pending" (menunggu), sehingga admin dapat memfokuskan perhatian pada pengajuan yang belum diproses. Setelah admin mengambil tindakan terhadap sebuah pengajuan, ada dua kemungkinan alur tergantung pada keputusan admin. Jika admin memberikan persetujuan, sistem akan mengubah status cuti menjadi "Disetujui". Jika admin menolak, sistem

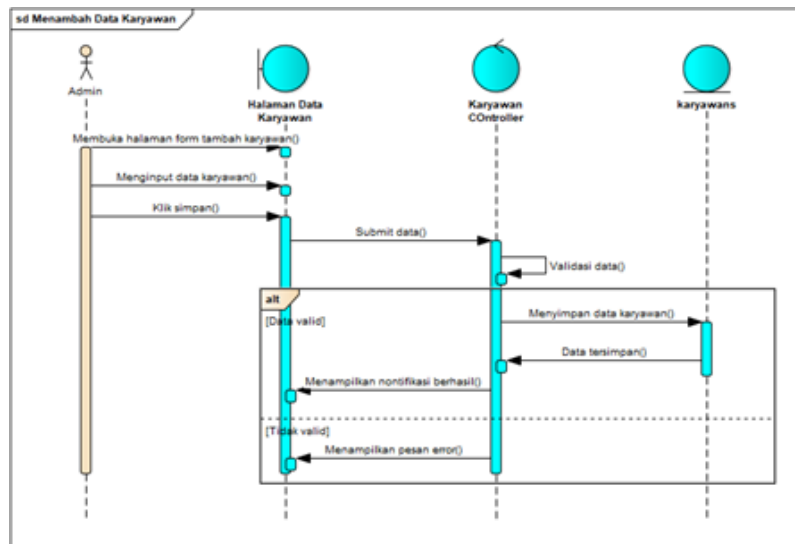
meminta admin untuk mengisi alasan penolakan terlebih dahulu sebelum mengubah status menjadi "Ditolak". Kedua alur diakhiri dengan notifikasi berhasil yang ditampilkan kepada admin sebagai konfirmasi bahwa status telah berhasil diperbarui.

6. *Sequence Diagram* Mengelola Data Karyawan

Interaksi antara aktor dan komponen sistem dalam pengelolaan data karyawan dijelaskan melalui *Sequence Diagram*. Diagram ini mengilustrasikan alur komunikasi mulai dari actor, *boundary*, *control*, hingga *entity* sampai proses berakhir. Adapun cakupan proses pada *Sequence Diagram* ini meliputi penambahan, pengubahan, serta penghapusan data karyawan.

a) *Sequence Diagram* Menambah Data Karyawan

Sequence diagram menambah data karyawan menggambarkan langkah-langkah interaksi antara admin dan sistem saat proses menambahkan data karyawan baru berlangsung. Diagram ini menampilkan proses aliran pesan yang terjadi antara aktor, *boundary*, *control*, dan *entity*, mulai dari admin membuka halaman untuk menambahkan karyawan, memasukkan data karyawan, hingga sistem menyimpan data tersebut ke dalam database. *Sequence* diagram menampilkan penambahan data karyawan, seperti yang ditunjukkan pada Gambar 4.20.



Gambar 4.20 Sequence Diagram Menambah Data Karyawan

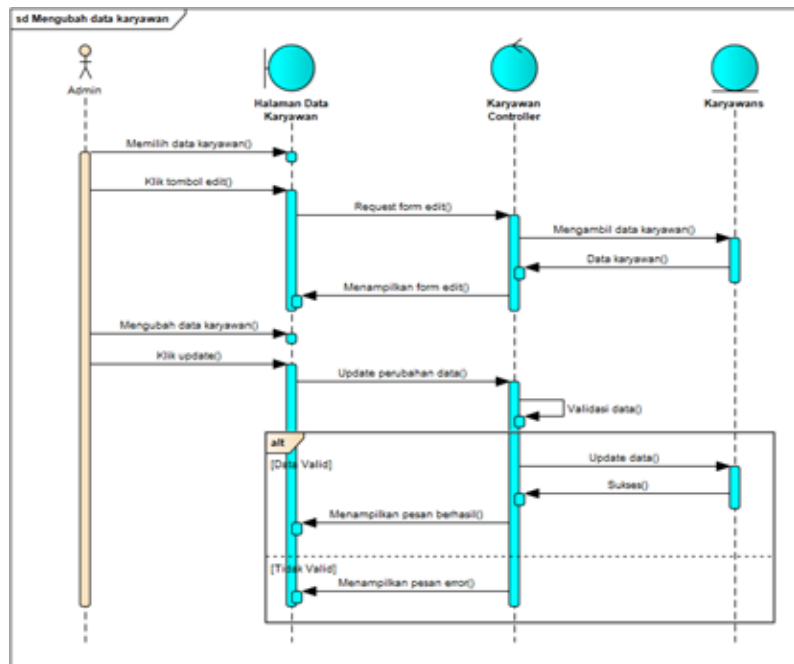
Berdasarkan Gambar 4.20, *sequence* diagram menambah data karyawan menunjukkan langkah-langkah interaksi antara admin dan sistem ketika melakukan penambahan data karyawan baru. Proses ini dimulai saat admin membuka halaman untuk menambah karyawan, lalu sistem akan menampilkan form yang harus diisi oleh admin untuk memasukkan data karyawan.

Setelah admin melengkapi data karyawan dan memilih tombol simpan, sistem secara otomatis meneruskan data tersebut ke bagian controller untuk diproses serta diverifikasi. Jika data yang dimasukkan sudah benar, maka controller akan menyimpan data karyawan ke dalam basis data melalui entity yang sudah ditentukan. Setelah data berhasil disimpan, sistem akan menampilkan pemberitahuan bahwa data karyawan telah berhasil ditambahkan. Namun,

jika data yang dimasukkan tidak benar atau terjadi kesalahan saat menyimpan data, sistem akan menampilkan pesan kesalahan kepada admin. Dengan demikian, admin dapat memperbaiki data sebelum mencoba menyimpannya kembali ke dalam sistem.

b) *Sequence Diagram* Mengubah Data Karyawan

Sequence diagram mengubah data karyawan menunjukkan langkah-langkah interaksi antara admin dan sistem saat proses perubahan data karyawan yang sudah disimpan di dalam basis data. Diagram ini menggambarkan alur pesan yang terjadi antara aktor, *boundry*, *controll*, dan *entity*, mulai dari proses pemilihan data karyawan, pengambilan data yang akan diubah, hingga tahap penyimpanan perubahan data ke dalam database. *Sequence* diagram mengubah data karyawan ditunjukkan pada Gambar 4.21.



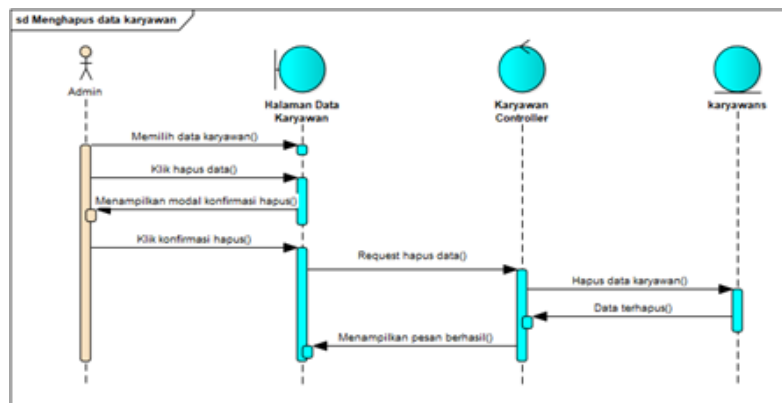
Gambar 4. 21 Sequence Diagram Mengubah Data Karyawan
Sequence diagram mengubah data karyawan menunjukkan proses interaksi antara admin dan sistem saat melakukan perubahan data karyawan. Proses dimulai ketika admin memilih data karyawan yang ingin diubah dan menekan tombol edit. Kemudian, sistem meminta *form* edit dari *controller*. *Controller* mengambil data karyawan dari database melalui *entity*. Setelah data karyawan didapatkan, sistem menampilkan *form* edit dengan data karyawan yang akan diperbarui. Admin mengubah data sesuai kebutuhan dan menekan tombol *update* untuk menyimpan perubahan.

Selanjutnya, sistem mengirim data perubahan ke *controller* untuk divalidasi. Jika data valid, *controller* akan memperbarui data karyawan di database melalui *entity*. Setelah data berhasil

diperbarui, sistem menampilkan pesan bahwa perubahan telah tersimpan. Tapi, jika data yang dimasukkan tidak valid atau terjadi kesalahan saat pembaruan, sistem akan menampilkan pesan error kepada admin. Dengan begitu, admin bisa memperbaiki dan memperbarui data sesuai kebutuhan.

c) **Sequence Diagram Menghapus Data Karyawan**

Sequence diagram menghapus data karyawan menunjukkan langkah-langkah interaksi antara admin dan sistem saat data karyawan yang tersimpan di dalam database dihapus. Diagram ini menampilkan alur pesan yang terjadi antara aktor, *boundry*, *control*, dan *entity*, mulai dari proses pemilihan data karyawan, pemberitahuan konfirmasi penghapusan, hingga langkah-langkah dalam menghapus data dari database. *Sequence* diagram menghapus data karyawan ditunjukkan pada Gambar 4.22.



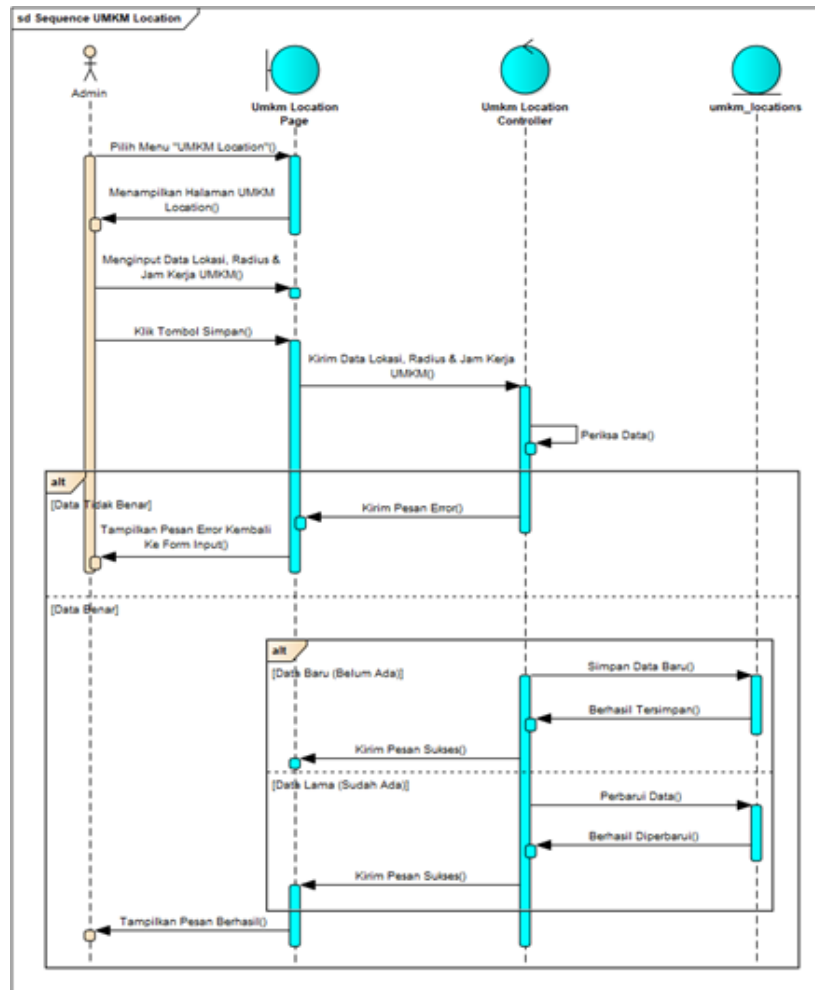
Gambar 4.22 *Sequence* Diagram Menghapus Data Karyawan

Berdasarkan Gambar 4.22, *Sequence* diagram menghapus data karyawan menjelaskan langkah-langkah yang dilakukan oleh admin untuk menghapus data karyawan yang tidak diperlukan lagi dalam sistem. Sebagai mekanisme keamanan untuk mencegah penghapusan yang tidak disengaja, sistem akan menampilkan konfirmasi terlebih dahulu sebelum data dihapus. Setelah admin mengonfirmasi tindakan tersebut, sistem meneruskan instruksi penghapusan ke basis data. Selanjutnya, data karyawan yang dipilih akan dihapus secara permanen dari penyimpanan sehingga tidak lagi muncul pada daftar karyawan. Jika operasi berhasil, sistem akan memberikan notifikasi konfirmasi penghapusan data kepada admin. Dengan adanya konfirmasi sebelum penghapusan, sistem dapat membantu menjaga konsistensi data dan mengurangi risiko kesalahan dalam pengelolaan data karyawan.

7. *Sequence* Diagram Mengelola Lokasi UMKM

Sequence diagram mengelola lokasi UMKM menunjukkan interaksi antara admin dan sistem dalam proses pengelolaan data lokasi UMKM yang digunakan sebagai dasar untuk memvalidasi kehadiran berbasis *geolocation*. Diagram ini menunjukkan aliran pesan yang terjadi mulai dari pengisian data lokasi, radius area presensi, dan jam kerja UMKM hingga proses penyimpanan atau

pembaruan data pada basis data. *Sequence* diagram mengelola lokasi UMKM ditunjukkan pada Gambar 4.23.



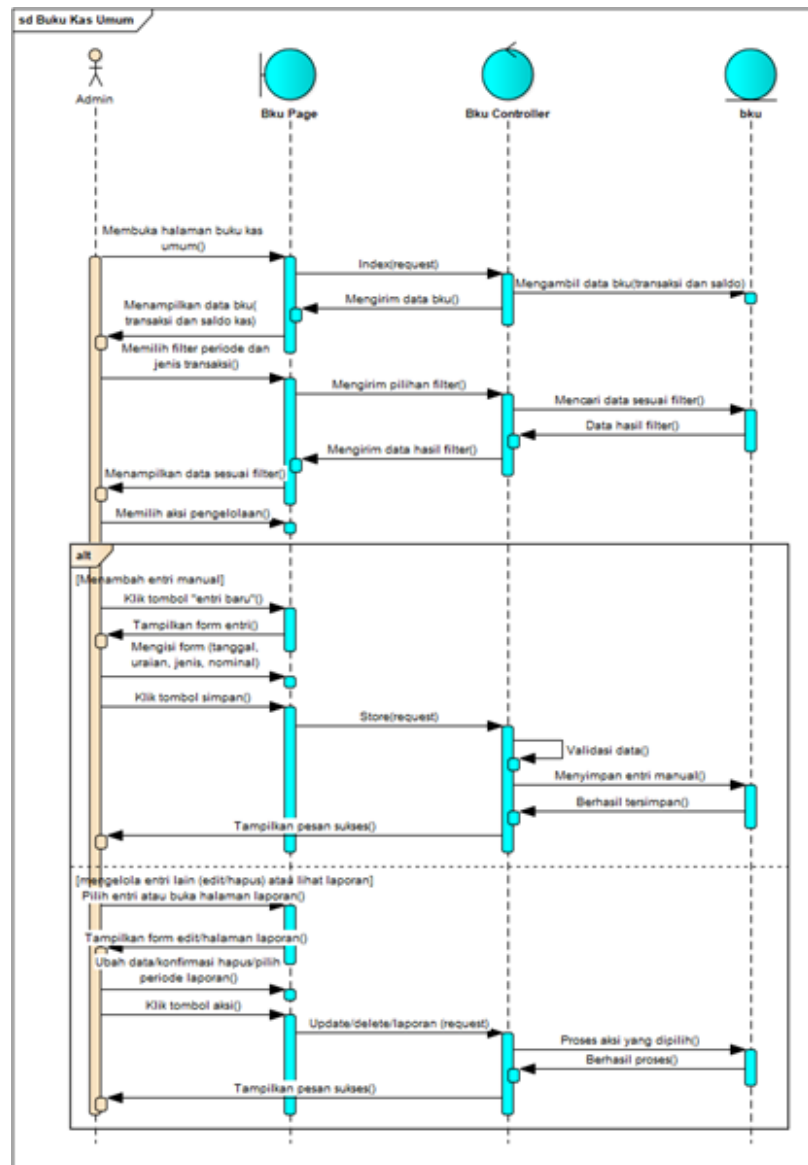
Gambar 4.23 *Sequence* Diagram Mengelola Lokasi UMKM

Berdasarkan Gambar 4.23, *Sequence* diagram mengelola lokasi UMKM menunjukkan proses pengaturan data lokasi yang dilakukan oleh admin. Data yang dikelola meliputi koordinat lokasi UMKM, radius area presensi, serta jam kerja yang akan digunakan sebagai acuan dalam proses presensi karyawan. Sebelum data disimpan, sistem terlebih dahulu melakukan pemeriksaan terhadap data yang dimasukkan untuk memastikan

kelengkapan dan kesesuaiannya. Apabila data yang diberikan belum memenuhi ketentuan, sistem akan menampilkan pesan kesalahan dan meminta admin untuk melakukan perbaikan pada data yang diinputkan. Sebaliknya, jika data telah sesuai, sistem akan memproses penyimpanan atau pembaruan data pada basis data sesuai kondisi yang terjadi. Setelah proses berhasil dilakukan, sistem memberikan notifikasi kepada admin sebagai informasi bahwa data lokasi UMKM telah berhasil disimpan atau diperbarui. Melalui pengelolaan lokasi UMKM ini, sistem dapat memastikan bahwa proses validasi lokasi presensi menggunakan *geolocation* berjalan sesuai dengan area dan aturan kerja yang telah ditetapkan oleh UMKM.

8. **Sequence Diagram Buku Kas Umum**

Sequence diagram untuk mengelola buku kas umum ini menjelaskan bagaimana admin berinteraksi dengan sistem saat mengelola buku kas umum. Proses ini dimulai dari melihat data transaksi dan saldo kas, memfilter data, hingga melakukan pencatatan *entry* manual seperti menambah, mengedit, atau menghapus data, serta membuat laporan. Ini semua menunjukkan aliran pesan antara aktor, *boundary* (Buku Kas Page), control (Buku Kas *Controller*), dan *entity* (Database). *Sequence* diagram mengelola buku kas umum dapat dilihat pada Gambar 4.24.



Gambar 4.24 Sequence Diagram Mengelola Buku Kas Umum

Berdasarkan Gambar 4.24 Sequence diagram mengelola buku kas umum menggambarkan interaksi antara admin dengan sistem. Proses dimulai saat admin membuka halaman buku kas umum; sistem lalu mengambil data transaksi dan saldo kas dari basis data untuk ditampilkan. Admin bisa memfilter data berdasarkan periode dan jenis transaksi, kemudian sistem mencari data sesuai

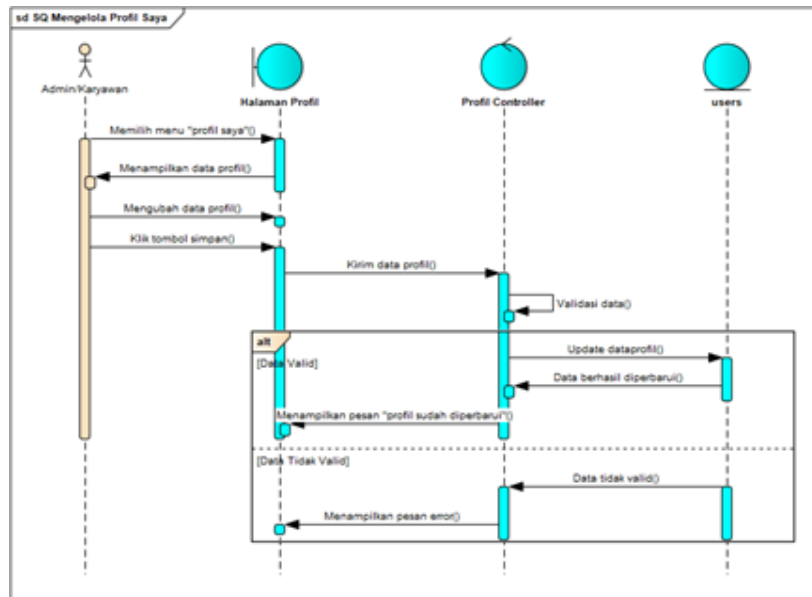
filter dan menampilkan hasilnya. Komponen control juga melakukan menghitung total entri buku kas, menghitung total transaksi (pemasukan dan pengeluaran), serta menghitung saldo kas akhir (total pemasukan dikurangi total pengeluaran).

Admin selanjutnya dapat melakukan pengelolaan data dengan memilih aksi pengelolaan. Untuk pencatatan transaksi secara manual, admin menekan tombol *Entry Baru*, kemudian sistem menampilkan formulir yang berisi tanggal, uraian, jenis transaksi, dan nominal. Setelah admin mengisi data dan menekan tombol simpan, data dikirim ke *BKU Controller* untuk divalidasi dan disimpan ke basis data. Jika proses penyimpanan berhasil, sistem menampilkan pesan sukses kepada admin. Selain itu, admin dapat mengelola entri lain dengan melakukan edit, hapus, atau membuka halaman laporan. Aksi tersebut diproses oleh *BKU Controller* dan berinteraksi dengan database, kemudian sistem menampilkan pesan keberhasilan setelah proses selesai.

9. *Sequence Diagram* Mengelola Profil Saya

Sequence diagram mengelola profil saya menggambarkan interaksi antara admin atau karyawan dengan sistem dalam proses mengelola data profil. Diagram ini menunjukkan aliran pesan antara aktor, *boundary*, *control*, dan *entity* mulai dari proses menampilkan data profil hingga penyimpanan perubahan data

profil. *Sequence* diagram mengelola profil saya pada sistem ditunjukkan pada gambar 4.25.



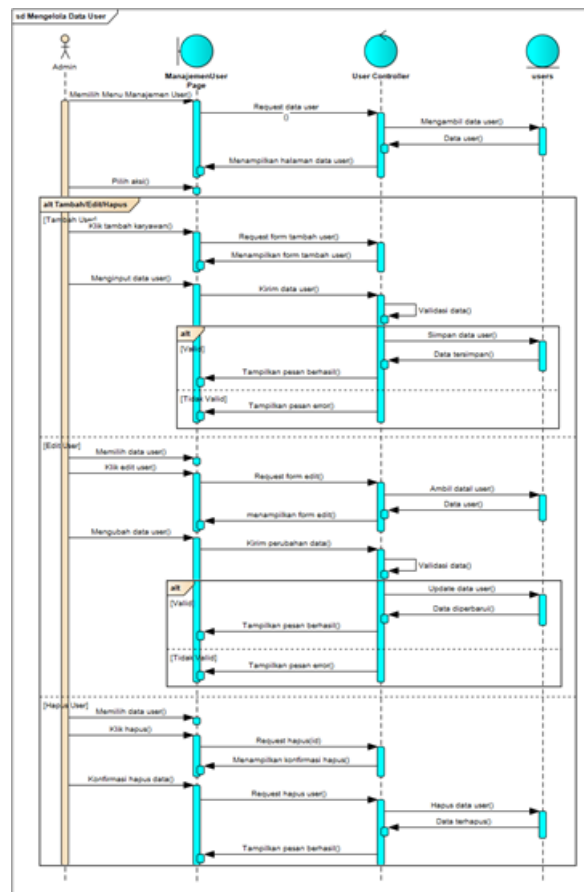
Gambar 4.25 *Sequence* Diagram Mengelola Profil Saya

Berdasarkan Gambar 4.25, *sequence* diagram mengelola profil saya menggambarkan proses interaksi antara admin atau karyawan dengan sistem pada saat mengubah data profil. Proses dimulai ketika pengguna memilih menu profil saya, kemudian sistem menampilkan data profil yang tersimpan. Selanjutnya pengguna melakukan perubahan data profil dan menekan tombol Simpan. Data tersebut kemudian dikirim oleh *boundary* ke *control* untuk dilakukan validasi. Apabila data yang dimasukkan valid, control memperbarui data profil ke dalam basis data melalui *entity*. Apabila proses pembaruan selesai dan sukses, sistem akan memberikan notifikasi bahwa profil berhasil diperbarui. Sebaliknya, jika data yang diinput dinyatakan tidak

valid, sistem secara otomatis menampilkan pesan kesalahan agar pengguna dapat mengoreksi data profil tersebut.

10. Sequence Diagram Mengelola Data User

Sequence diagram ini memperlihatkan bagaimana admin berinteraksi dengan sistem saat mengelola data pengguna (*user*). Diagram ini menunjukkan aliran pesan antara aktor, *boundary*, *control*, dan *entity* mulai dari proses menampilkan halaman manajemen *user* hingga proses penambahan, perubahan, dan penghapusan data *user*. Sequence diagram mengelola data *user* dapat dilihat pada Gambar 4.26.



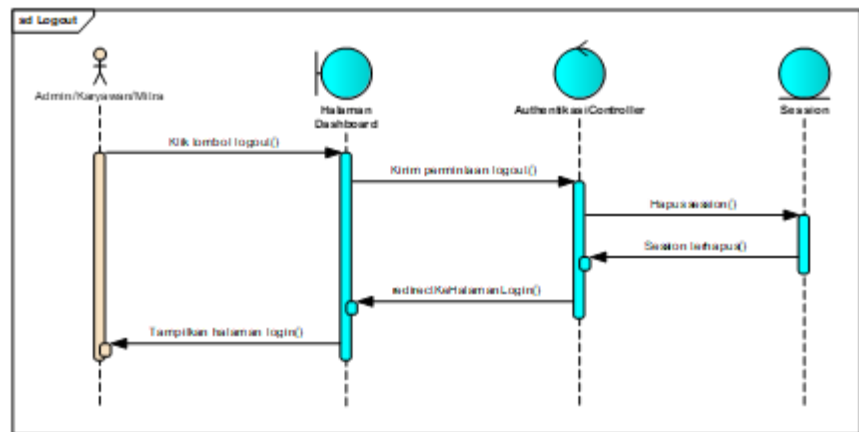
Gambar 4.26 Sequence Diagram Mengelola Data User

sequence diagram pengelolaan data user mengilustrasikan alur interaksi antara admin dan sistem dalam memproses data pengguna. Proses dimulai ketika admin memilih menu manajemen user, kemudian boundary mengirimkan permintaan data kepada control untuk mengambil data user dari entity atau basis data. Setelah data berhasil diperoleh, sistem menampilkan halaman manajemen user kepada admin. Selanjutnya admin dapat memilih aksi berupa Tambah user, edit user, atau hapus user. Pada proses penambahan dan perubahan data, admin menginput data user yang kemudian dikirim oleh boundary ke control untuk dilakukan validasi. Jika data dinyatakan valid, *control* akan menyimpan atau memperbarui data user ke basis data melalui entity, dilanjutkan dengan penampilkkan notifikasi keberhasilan oleh sistem. Sebaliknya, apabila data tidak valid, sistem secara otomatis memunculkan pesan kesalahan kepada admin. Pada proses penghapusan data, admin memilih data user yang akan dihapus kemudian melakukan konfirmasi penghapusan. Selanjutnya control menghapus data user pada basis data melalui entity dan sistem menampilkan notifikasi bahwa data user berhasil dihapus.

11. *Sequence Diagram Logout*

Sequence diagram Logout menggambarkan interaksi antara pengguna (karyawan, admin) dengan sistem dalam proses

mengakhiri sesi atau keluar dari sistem. Diagram ini menunjukkan aliran pesan antara aktor, boundary, control, dan entity mulai dari klik tombol *Logout* hingga redirect ke halaman *Login*. *Sequence diagram Logout* pada sistem ditunjukkan pada Gambar 4.27.



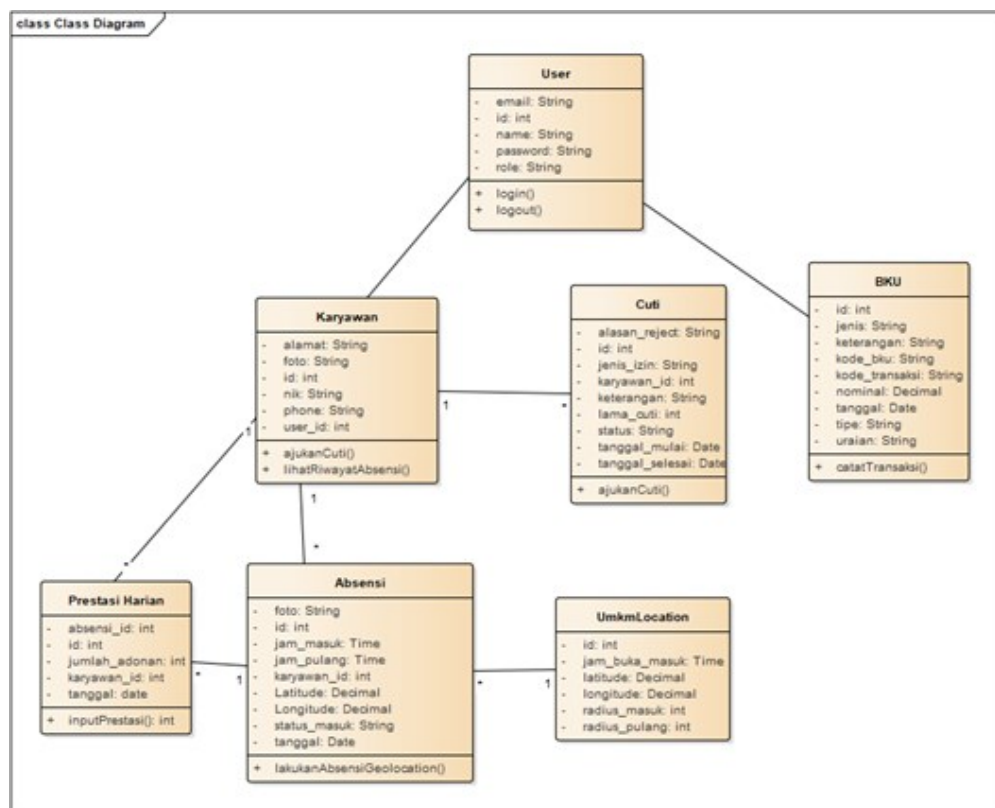
Gambar 4.27 *Sequence Diagram Logout*

sequence diagram Logout menggambarkan interaksi antara pengguna dengan sistem pada saat mengakhiri sesi. Proses dimulai ketika pengguna menekan tombol *Logout* pada sistem. Boundary mengirim permintaan *Logout* ke control, kemudian control menghapus session pengguna dari entity. Setelah session berhasil dihapus, sistem melakukan redirect ke halaman *Login* dan menampilkannya kepada pengguna. Proses *Logout* selesai.

4.3.4 *Class diagram*

Class diagram digunakan untuk menggambarkan struktur class pada sistem, termasuk hubungan antar class, atribut, dan operasi yang

dimiliki masing-masing class. Jadi, class diagram pada sistem ini dibuat berdasarkan kebutuhan sistem yang telah dirancang sebelumnya. Secara keseluruhan, Class Diagram mengilustrasikan relasi antar-data dan proses dalam sistem, sebagaimana ditunjukkan pada Gambar 4.28.



Gambar 4. 28 Class Diagram

Berdasarkan Gambar 4.28 class diagram pada sistem ini terdiri atas beberapa class utama yang saling terhubung dalam mendukung proses pengelolaan sistem. Class-class tersebut meliputi user, karyawan, prestasi harian, cuti, prestasi harian, bku, dan UMKM location. Class user berfungsi sebagai penyimpanan data akun pengguna sistem yang memiliki hak akses sesuai dengan role masing-masing. Class

karyawan digunakan untuk menyimpan data pengguna berdasarkan kebutuhan operasional sistem. Selanjutnya, class presensi dan cuti mendukung proses kehadiran serta pengajuan cuti karyawan. Class prestasi harian digunakan untuk menyimpan data hasil kerja harian karyawan berupa jumlah adonan yang berhasil diselesaikan setiap hari sebagai bentuk pencatatan prestasi kerja. Class UMKM location digunakan untuk menyimpan data lokasi UMKM, radius presensi, dan jam kerja yang menjadi acuan dalam proses validasi presensi berbasis geolocation. Selain itu, class bku digunakan untuk menyimpan data pemasukan dan pengeluaran kas UMKM yang dikelola oleh admin sebagai dasar penyusunan laporan keuangan. Relasi antar class pada sistem dijelaskan sebagai berikut:

1. Relasi antara class User dan class Karyawan dibentuk untuk mengelola data akun pengguna sistem.
2. Class Karyawan memiliki relasi dengan class Presensi karena setiap karyawan dapat memiliki beberapa data presensi.
3. Class Karyawan memiliki relasi dengan class Cuti karena setiap karyawan dapat melakukan pengajuan cuti.
4. Class Karyawan memiliki relasi dengan class Prestasi Harian karena setiap karyawan dapat mencatat data prestasi kerja harian.
5. Class Presensi memiliki relasi dengan class Prestasi Harian karena setiap data prestasi harian berkaitan dengan data presensi pada hari yang sama.

6. Class Presensi memiliki relasi dengan class UMKMLocation sebagai data acuan validasi lokasi presensi berbasis geolocation.
7. Class BKU digunakan untuk menyimpan data pemasukan dan pengeluaran kas UMKM. Data pada class ini dikelola secara langsung oleh pengguna yang memiliki hak akses sebagai admin.

4.4 Desain *Input/Output*

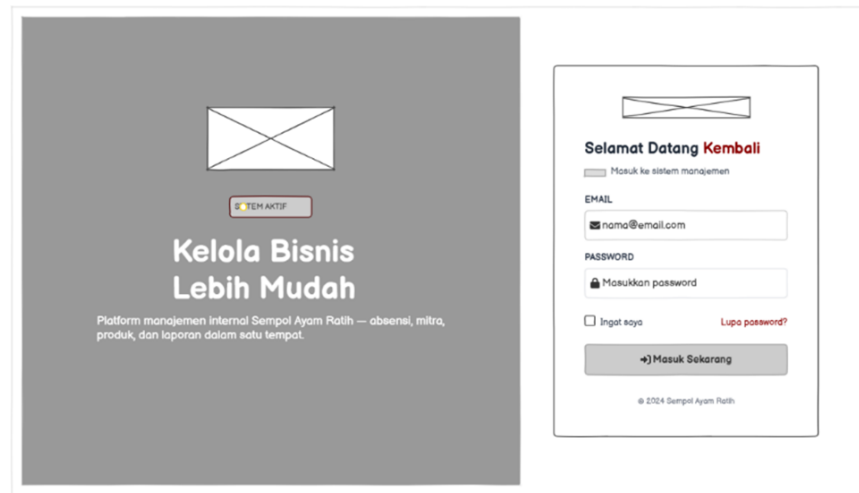
Desain *input/output* digunakan untuk menggambarkan rancangan tampilan antarmuka pada sistem yang meliputi proses memasukkan data dan penyajian informasi kepada pengguna. Perancangan ini bertujuan untuk memberikan gambaran mengenai tampilan sistem yang akan digunakan oleh admin dan karyawan dalam menjalankan fungsi-fungsi pada sistem.

4.4.1 Desain *Input*

Desain input digunakan untuk menggambarkan rancangan halaman yang digunakan pengguna dalam memasukkan data ke dalam sistem.

1. *Form Login*

Form Login dirancang sebagai antarmuka awal bagi pengguna untuk melakukan proses masuk ke dalam sistem. Halaman ini berfungsi memfasilitasi autentikasi pengguna sebelum mereka diberikan hak akses ke berbagai fitur yang tersedia. Desain *Wireframe form Login* dapat dilihat pada Gambar 4.29.



Gambar 4.29 *Wireframe Form Login*

Wireframe form Login menampilkan beberapa komponen utama yang digunakan dalam proses autentikasi pengguna ke dalam sistem. Komponen tersebut terdiri dari input email dan password yang digunakan pengguna untuk memasukkan data akun. Selain itu, terdapat fitur ingat saya yang digunakan untuk menyimpan sesi *Login* pengguna, menu lupa password untuk membantu proses pemulihan akun, serta tombol masuk sekarang yang digunakan untuk memproses data *Login* ke dalam sistem. Setelah data *Login* dimasukkan dengan benar, sistem akan menampilkan halaman dashboard sesuai hak akses pengguna.

Setelah proses perancangan antarmuka form *Login* selesai dilakukan, selanjutnya disusun perancangan pengujian sebagai acuan untuk memverifikasi fungsi-fungsi yang terdapat pada halaman *Login*. Tujuan dari perancangan ini adalah memvalidasi seluruh fitur yang dirancang agar sesuai dengan spesifikasi

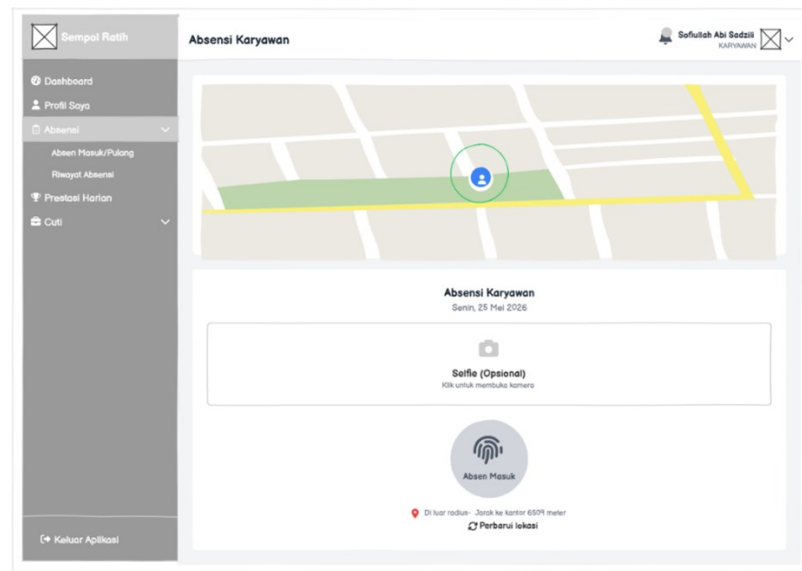
kebutuhan sistem saat memasuki tahap implementasi. Adapun skenario perancangan pengujian form *Login* dapat dilihat pada Tabel 4.2.

Tabel 4.2 Perancangan Pengujian *Login*

No	Skenario Pengujian	Hasil yang Diharapkan
1.	<i>Login</i> dengan menggunakan email dan <i>password</i> yang valid	Sistem menampilkan dashboard sesuai dengan hak akses pengguna
2.	<i>Login</i> menggunakan email yang tidak terdaftar	Sistem menampilkan pesan gagal <i>Login</i>
3.	<i>Login</i> menggunakan <i>password</i> yang salah	Sistem menampilkan pesan gagal <i>Login</i>
4.	<i>Login</i> tanpa mengisi email dan <i>password</i>	Sistem menampilkan validasi input

2. Form Presensi

Form presensi karyawan digunakan untuk melakukan proses presensi masuk dan presensi pulang berbasis geolocation. Pada halaman ini karyawan dapat melakukan presensi dengan memanfaatkan lokasi gps sebagai validasi kehadiran agar proses presensi hanya dapat dilakukan pada area yang telah ditentukan oleh sistem. Rancangan *Wireframe* untuk antarmuka form presensi karyawan disajikan pada Gambar 4.30.



Gambar 4.30 *Wireframe Form Presensi Karyawan*

Wireframe form presensi karyawan menampilkan peta lokasi pengguna yang digunakan untuk mengetahui posisi karyawan terhadap lokasi UMKM. Selain itu, tersedia fitur selfie opsional yang dapat digunakan sebagai dokumentasi presensi karyawan. Pada bagian bawah tampilan terdapat tombol presensi yang digunakan untuk melakukan presensi masuk maupun presensi pulang. Sistem juga menampilkan informasi jarak pengguna terhadap lokasi UMKM serta fitur perbarui lokasi untuk memperbarui koordinat GPS apabila lokasi pengguna belum berada pada radius presensi yang telah ditentukan. Setelah proses perancangan antarmuka form presensi selesai dilakukan, selanjutnya disusun perancangan pengujian sebagai acuan untuk memverifikasi fungsi-fungsi yang terdapat pada halaman presensi. Perancangan ini bertujuan agar setiap fitur yang telah

dirancang dapat diuji kesesuaiannya dengan kebutuhan sistem pada tahap implementasi. Adapun skenario perancangan pengujian form presensi dapat dilihat pada Tabel 4.3.

Tabel 4.3 Perancangan Pengujian Presensi Berbasis *Geolocation*

No	Skenario Pengujian	Hasil yang Diharapkan
1.	Mengakses halaman presensi dengan GPS aktif	Halaman presensi berhasil ditampilkan beserta lokasi pengguna pada peta.
2.	Melakukan presensi dalam radius lokasi UMKM	Data presensi berhasil disimpan
3.	Melakukan presensi di luar radius lokasi UMKM	Sistem menolak proses presensi
4.	Mengakses halaman presensi dengan GPS tidak aktif	Sistem meminta pengguna mengaktifkan GPS
5.	Melakukan presensi pulang setelah presensi masuk	Data presensi pulang berhasil disimpan
6.	Mengakses halaman presensi dengan GPS aktif	Halaman presensi berhasil ditampilkan beserta lokasi pengguna pada peta.

3. Form Presensi Harian

Form prestasi harian digunakan untuk mencatat hasil pekerjaan karyawan setiap hari sebagai dasar penilaian produktivitas kerja. Pada halaman ini, karyawan dapat menginput jumlah adonan yang berhasil diproduksi serta informasi jam lembur apabila melakukan pekerjaan di luar jam kerja normal. Sistem secara otomatis menyimpan data yang diinput sebagai laporan kinerja harian karyawan. Desain *Wireframe* form prestasi harian dapat dilihat pada gambar 4.31.

Gambar 4.31 *Desain Input Prestasi Karyawan*

Wireframe form prestasi harian menampilkan informasi hari dan tanggal yang terisi secara otomatis sesuai tanggal saat pengisian data. Selain itu, ditampilkan status kehadiran yang diambil dari data presensi karyawan pada hari tersebut. Form juga menyediakan kolom jumlah adonan yang digunakan untuk memasukkan hasil produksi harian serta kolom lama jam kerja untuk menampilkan atau mencatat jumlah lama jam kerja yang dilakukan karyawan. Halaman ini dilengkapi dengan tombol Simpan Prestasi di bagian bawah untuk menyimpan catatan prestasi harian, serta tombol Batal yang digunakan untuk membatalkan aksi dan navigasi kembali ke halaman sebelumnya. Setelah proses perancangan antarmuka form prestasi harian selesai dilakukan, selanjutnya disusun perancangan pengujian

sebagai acuan untuk memverifikasi fungsi-fungsi yang terdapat pada halaman prestasi harian. Perancangan ini bertujuan agar setiap fitur yang telah dirancang dapat diuji kesesuaiannya dengan kebutuhan sistem pada tahap implementasi. Adapun skenario perancangan pengujian form prestasi harian dapat dilihat pada Tabel 4.4.

Tabel 4.4 Perancangan Pengujian Prestasi Harian

No	Skenario Pengujian	Hasil yang Diharapkan
1.	Mengisi data prestasi harian dengan lengkap lalu dikirim	Data prestasi harian berhasil disimpan
2.	Mengisi data prestasi harian dengan data yang belum lengkap	Sistem menampilkan pesan error validasi

4. Form Pengajuan Cuti

Rancangan antarmuka form pengajuan cuti digunakan untuk mempermudah proses pengajuan izin atau cuti yang dilakukan oleh karyawan melalui sistem. Melalui form ini, proses pengajuan cuti dapat dilakukan sehingga data pengajuan dapat langsung diterima dan diproses oleh admin. Desain *Wireframe* form pengajuan cuti dapat dilihat pada Gambar 4.32.

The wireframe shows a web application interface for submitting a leave request. On the left is a sidebar menu with options: Dashboard, Profil Saya, Absensi, Prestasi Harian, Cuti (selected), Pengajuan Cuti, and Riwayat Cuti. The main content area is titled 'Pengajuan Cuti' and contains a form titled 'Formulir Izin & Cuti' with the subtitle 'Pengajuan izin atau cuti karyawan'. The form fields include: 'Nama Karyawan' (filled with 'Sifulah Ahi Sadzli'), 'Keperluan / Jenis Izin' (with buttons for 'Sakit', 'Acara Keluarga', and 'Lain-lain'), 'Tanggal Mulai' (mm/dd/yyyy), 'Tanggal Selesai' (mm/dd/yyyy), and a 'Keterangan Tambahan' text area with a placeholder 'Berikan alasan detail atau informasi tambahan yang relevan...'. A note at the bottom states 'Pengajuan akan diproses oleh admin. Pastikan semua data sudah benar sebelum mengirim.' and a 'Kirim Pengajuan' button is at the bottom right.

Gambar 4.32 *Wireframe Form* Pengajuan Cuti

Tampilan form pengajuan cuti dirancang agar karyawan dapat melakukan proses pengajuan izin atau cuti secara lebih mudah dan terstruktur. Pada tampilan ini karyawan dapat memilih jenis izin yang diajukan seperti sakit, acara keluarga, maupun keperluan lainnya. Selain itu, karyawan juga dapat menentukan tanggal mulai dan tanggal selesai cuti serta menambahkan keterangan tambahan sebagai informasi pendukung pengajuan. Setelah seluruh data diisi dengan lengkap, pengguna dapat mengirimkan pengajuan cuti melalui tombol kirim pengajuan sehingga data pengajuan akan diteruskan ke admin untuk dilakukan proses verifikasi dan persetujuan. Setelah proses perancangan antarmuka form pengajuan cuti selesai dilakukan, selanjutnya disusun perancangan pengujian sebagai acuan untuk memverifikasi fungsi-fungsi yang terdapat pada halaman pengajuan cuti. Perancangan ini bertujuan agar setiap fitur yang

telah dirancang dapat diuji kesesuaiannya dengan kebutuhan sistem pada tahap implementasi. Adapun skenario perancangan pengujian *form* pengajuan cuti dapat dilihat pada Tabel 4.5.

Tabel 4.5 Perancangan Pengujian Pengajuan Cuti

No	Skenario Pengujian	Hasil yang Diharapkan
1.	Mengisi seluruh data pengajuan cuti dengan lengkap lalu dikirim	Data prestasi harian berhasil disimpan
2.	Mengisi form pengajuan cuti dengan data yang belum lengkap	Sistem menampilkan pesan validasi
3.	Mengajukan cuti dengan tanggal selesai lebih awal dari tanggal mulai	Sistem menampilkan pesan validasi bahwa rentang tanggal cuti tidak valid.

5. Form Data Karyawan

Rancangan antarmuka ini digunakan oleh administrator untuk menambahkan serta mengelola entri data karyawan baru pada sistem. Form ini dirancang agar proses pendaftaran karyawan menjadi lebih terstruktur dan terintegrasi dengan akun sistem. Rancangan *Wireframe* untuk antarmuka form data karyawan disajikan pada Gambar 4.33.

Tambah Data Karyawan ×

Silakan isi formulir untuk mendaftarkan karyawan baru.

NAMA LENGKAP

Masukkan nama sesuai KTP

NIK **NO. WHATSAPP**

16 digit nomor induk 08xxxxxxxx

TEMPAT LAHIR **TANGGAL LAHIR**

Kota mm/dd/yyyy 📅

ALAMAT LENGKAP

Jl. Contoh No. 123, RT 01/RW 02...

Batal **Simpan Data**

Gambar 4.33 *Wireframe Form* Data Karyawan

Pada rancangan antarmuka data karyawan, menampilkan beberapa komponen penting yang dibutuhkan untuk memasukkan data karyawan secara lengkap. Bagian atas form terdapat pilihan integrasi akun sistem karyawan melalui dropdown. Selanjutnya terdapat input nama lengkap sesuai KTP, NIK, nomor WhatsApp, tempat lahir, tanggal lahir, serta alamat lengkap karyawan. Rancangan antarmuka form ini dibuat secara sederhana dan mudah digunakan (user-friendly), sehingga memfasilitasi admin dalam memasukkan serta mengelola data karyawan baru.

Setelah proses perancangan antarmuka form data karyawan selesai dilakukan, selanjutnya disusun perancangan pengujian sebagai acuan untuk memverifikasi fungsi-fungsi yang terdapat pada halaman data karyawan. Perancangan ini bertujuan agar setiap fitur yang telah dirancang dapat diuji kesesuaiannya

dengan kebutuhan sistem pada tahap implementasi. Adapun skenario perancangan pengujian form data karyawan dapat dilihat pada Tabel 4.6.

Tabel 4.6 Perancangan Pengujian Pengelolaan Data Karyawan

No	Skenario Pengujian	Hasil yang Diharapkan
1.	Menambahkan data karyawan baru dengan menginput seluruh <i>field</i> secara lengkap dan valid.	Sistem berhasil menyimpan data karyawan baru dan mengarahkan kembali ke halaman utama data karyawan.
2.	Menambahkan data karyawan dengan data yang tidak lengkap	Sistem menampilkan pesan validasi bahwa data wajib diisi.
3.	Mengubah data karyawan	Sistem berhasil memperbarui data karyawan.
4.	Menghapus data karyawan	Sistem berhasil menghapus data karyawan dari daftar.
5.	Menambahkan data karyawan baru dengan menginput seluruh <i>field</i> secara lengkap dan valid.	Sistem berhasil menyimpan data karyawan baru dan mengarahkan kembali ke halaman utama data karyawan.

6. Form Lokasi UMKM

Rancangan antarmuka form lokasi UMKM digunakan oleh admin untuk mengatur koordinat pusat lokasi serta parameter geofencing yang nantinya digunakan dalam fitur presensi berbasis geolocation. Admin menggunakan form ini untuk mengatur lokasi yang tepat dan menentukan area geofencing. Form lokasi UMKM sangat penting karena menentukan validitas lokasi presensi karyawan. Desain *Wireframe* dari form lokasi UMKM dapat dilihat pada Gambar 4.34.

KOORDINAT PUSAT

LAT -6.8748790

LNG 109.0562240

PARAMETER RADIUS (METER)

Radius Masuk 50m

Radius Pulang 75m

PENGATURAN JAM ABSENSI

Jam Buka Absen Masuk

05:00

Batas Jam Normal (lewat = telat)

07:00

Jam Buka Absen Pulang

12:00

Simpan Perubahan →

Gambar 4.34 *Wireframe Form* Lokasi UMKM

Pada rancangan antarmuka *Wireframe form* lokasi UMKM menampilkan pengaturan koordinat pusat berupa input *Latitude* dan *Longitude* lokasi UMKM Sempol Ayam Ratih. Selain itu, *form* ini juga menyediakan pengaturan parameter radius dalam meter, yaitu radius masuk dan radius pulang. Bagian bawah *form* terdapat pengaturan jam presensi yang meliputi jam buka absen masuk, batas jam normal, serta jam buka absen pulang. Semua pengaturan tersebut dapat disimpan melalui tombol Simpan Perubahan. Dengan adanya *form* ini, admin dapat dengan mudah mengatur dan menyesuaikan radius *geofencing* serta jadwal presensi sesuai dengan kebutuhan operasional UMKM. Setelah proses perancangan antarmuka *form* lokasi UMKM selesai

dilakukan, selanjutnya disusun perancangan pengujian sebagai acuan untuk memverifikasi fungsi-fungsi yang terdapat pada halaman lokasi UMKM. Perancangan ini bertujuan agar setiap fitur yang telah dirancang dapat diuji kesesuaiannya dengan kebutuhan sistem pada tahap implementasi. Adapun skenario perancangan pengujian *form* lokasi UMKM dapat dilihat pada Tabel 4.7.

Tabel 4.7 Perancangan Pengujian Lokasi UMKM

No	Skenario Pengujian	Hasil Yang Diharapkan
1.	Mengubah data lokasi UMKM	Sistem berhasil memperbarui data lokasi UMKM
2.	Mengubah radius lokasi presensi	Sistem berhasil memperbarui data radius lokasi UMKM
3.	Mengubah jam kerja UMKM	Sistem berhasil memperbarui data jam kerja UMKM

7. Form Buku Kas Umum

Rancangan antarmuka *form* buku kas umum digunakan sebagai tampilan bagi admin dalam melakukan pencatatan transaksi pemasukan maupun pengeluaran kas pada sistem. *Form* ini dirancang untuk memudahkan proses pengelolaan data kas. Desain *Wireframe form* buku kas umum bisa dilihat pada gambar 4.35.

Entri BKU Baru

Tanggal * 05/25/2026

Jenis * Beban Tenaga Kerja

Uraian * Keterangan singkat transaksi...

Nominal (Rp) * Rp

Keterangan Tambahan Opsional...

↑ Jenis: PENGELUARAN

Batal Simpan

Gambar 4.35 *Wireframe Form* Buku Kas Umum

Pada rancangan antarmuka *form* buku kas umum sistem menyediakan beberapa input yang digunakan untuk mencatat transaksi, seperti tanggal transaksi, jenis transaksi, uraian transaksi, nominal transaksi, dan keterangan tambahan. Sistem juga menampilkan jenis transaksi yang Anda pilih, seperti pemasukan atau pengeluaran kas. Di bagian bawah *form*, ada dua tombol yang bisa digunakan. Tombol pertama adalah tombol simpan yang digunakan untuk menyimpan data transaksi ke dalam sistem. Tombol kedua adalah tombol batal yang digunakan untuk membatalkan proses pengisian data transaksi. Setelah proses perancangan antarmuka *form* Buku Kas Umum selesai dilakukan, selanjutnya disusun perancangan pengujian sebagai acuan untuk memverifikasi fungsi-fungsi yang terdapat pada

halaman Buku Kas Umum. Perancangan ini bertujuan agar setiap fitur yang telah dirancang dapat diuji kesesuaiannya dengan kebutuhan sistem pada tahap implementasi. Adapun skenario perancangan pengujian *form* Buku Kas Umum dapat dilihat pada Tabel 4.8.

Tabel 4.8 Perancangan Pengujian Buku Kas Umum

No	Skenario Pengujian	Hasil Yang Diharapkan
1.	Menambahkan data transaksi pemasukan	Data pemasukan berhasil disimpan dan saldo kas bertambah
2.	Menambahkan data transaksi pengeluaran	Data pengeluaran berhasil disimpan dan saldo kas berkurang
3.	Mengubah data transaksi	Data transaksi berhasil diperbarui
4.	Menghapus data transaksi	Data transaksi berhasil dihapus dan saldo kas menyesuaikan
5.	Menampilkan saldo kas dan laporan keuangan	Saldo kas serta laporan keuangan ditampilkan dengan benar

8. Form Profil Saya

Antarmuka *form* profil saya berfungsi memfasilitasi pengguna dalam melihat dan mengubah informasi data diri karyawan pada sistem. Melalui halaman ini, karyawan dapat memperbarui informasi pribadi agar data yang dimiliki sistem selalu sesuai dengan kondisi terbaru. Desain *Wireframe form* profil saya dapat dilihat pada Gambar 4.36.

Profil Saya

Data Profil

Nama Lengkap

Email

No HP

Alamat

Gambar 4.36 *Desain Wireframe Input Profil Saya*

Wireframe form profil saya menampilkan beberapa data identitas karyawan, yaitu Nama Lengkap, Email, Nomor HP, dan Alamat. Seluruh data tersebut dapat diubah sesuai kebutuhan pengguna, kecuali apabila terdapat pembatasan hak akses yang telah ditentukan oleh sistem. Antarmuka ini dilengkapi dengan tombol Simpan Profil di bagian bawah guna merekam perubahan data pengguna ke *database*, serta tombol Batal untuk membatalkan entri dan navigasi kembali ke tampilan sebelumnya. Setelah proses perancangan antarmuka *form* Profil Saya selesai dilakukan, selanjutnya disusun perancangan pengujian sebagai acuan untuk memverifikasi fungsi-fungsi yang terdapat pada halaman Profil Saya. Perancangan ini bertujuan agar setiap fitur yang telah dirancang dapat diuji kesesuaiannya dengan kebutuhan sistem pada tahap implementasi. Adapun skenario perancangan pengujian *form* Profil Saya dapat dilihat pada Tabel 4.9.

Tabel 4.9 Perancangan Pengujian Profil Saya

No	Skenario Pengujian	Hasil Yang Diharapkan
1.	Mengubah informasi profil pengguna	Sistem menampilkan pesan Data profil berhasil diperbarui
2.	Mengubah kata sandi dengan data yang valid	Sistem menampilkan pesan ata sandi berhasil diperbarui
3.	Mengubah kata sandi dengan konfirmasi <i>password</i> yang tidak sesuai	Sistem menampilkan pesan validasi

9. Form Mengelola Data User

Antarmuka *form* data *user* berfungsi memfasilitasi penambahan akun pengguna baru yang akan mengakses sistem. Di halaman ini admin dapat menginput informasi pengguna yang dibutuhkan seperti nama lengkap, alamat email, peran pengguna, dan kata sandi sementara. Informasi yang dimasukkan akan berfungsi sebagai akun untuk mengakses sistem berdasarkan hak akses yang telah ditetapkan. *Wireframe* desain untuk *form* data *user* dapat dilihat pada gambar 4.37

Tambah User Baru ✕

Isi data untuk membuat akun pengguna.

Nama Lengkap

👤 Masukkan nama lengkap

Alamat Email

✉️ Masukkan alamat email

Role Pengguna

— Pilih role — ▾

Password Sementara

🔒 👁️

User akan diminta mengganti password saat login pertama kali.

Batal Simpan User

Gambar 4.37 *Wireframe Form* Mengelola Data User

Beberapa komponen input ditampilkan dalam *Wireframe form* data pengguna, termasuk Nama Lengkap, Alamat Email, *Role* Pengguna, dan *Password* Sementara. Kolom *Role* Pengguna menunjukkan hak akses pengguna seperti admin atau karyawan. Selain itu, ada ikon yang dapat digunakan untuk menampilkan atau menyembunyikan *password* untuk sementara waktu, yang memungkinkan admin untuk melakukan pengecekan saat proses pengisian data. Di bagian bawah *form*, disediakan tombol Simpan Pengguna untuk memproses penyimpanan data pengguna, serta tombol Batal yang memfasilitasi pembatalan entri data dan penutupan tampilan. Setelah data pengguna disimpan, pengguna baru akan diminta untuk mengganti *password*nya saat pertama kali masuk ke sistem. Setelah proses perancangan antarmuka

form Data User selesai dilakukan, selanjutnya disusun perancangan pengujian sebagai acuan untuk memverifikasi fungsi-fungsi yang terdapat pada halaman Data User. Perancangan ini bertujuan agar setiap fitur yang telah dirancang dapat diuji kesesuaiannya dengan kebutuhan sistem pada tahap implementasi. Adapun skenario perancangan pengujian *form* Data User dapat dilihat pada Tabel 4.10.

Tabel 4.10 Perancangan Pengujian Data User

No	Skenario Pengujian	Hasil Yang Diharapkan
1.	Menambahkan data <i>user</i> baru	Data <i>user</i> baru berhasil ditambahkan
2.	Menambahkan data <i>user</i> dengan data yang tidak lengkap	Sistem menampilkan pesan validasi bahwa data wajib diisi.
3.	Mengubah data <i>user</i>	Data <i>user</i> berhasil diperbarui
4.	Menghapus data <i>user</i>	Data <i>user</i> berhasil dihapus

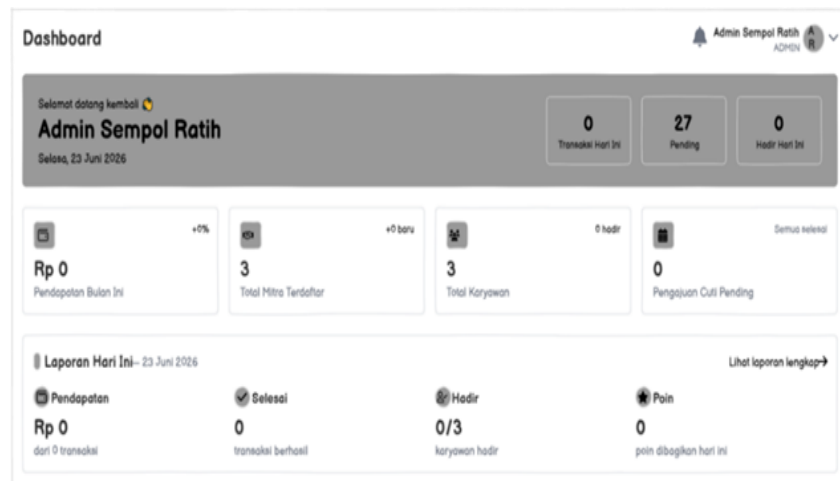
4.4.2 Desain *Output*

Desain *output* digunakan untuk menggambarkan rancangan tampilan informasi atau hasil proses yang ditampilkan oleh sistem kepada pengguna.

1. Halaman *Dashboard*

Halaman *dashboard* dirancang sebagai tampilan utama yang memberikan ringkasan informasi secara cepat sesuai dengan peran pengguna. Dalam rancangan ini, dashboard admin ditunjukkan sebagai representasi utama karena memiliki tampilan

yang paling lengkap untuk memantau operasional UMKM. Desain *Wireframe* halaman dashboard dapat dilihat pada gambar 4.38.



Gambar 4.38 *Wireframe Dashboard*

Bagian tengah *dashboard* terdiri dari beberapa *card* informasi yang menampilkan pendapatan bulan ini, total mitra terdaftar, total karyawan, dan jumlah pengajuan cuti yang masih pending. Sedangkan di bagian bawah terdapat laporan harian yang mencakup pendapatan, transaksi selesai, jumlah karyawan hadir, serta poin yang dibagikan pada hari tersebut. Rancangan *dashboard* ini dirancang dengan tampilan yang clean, menggunakan *card* dan warna yang kontras sehingga admin dapat dengan mudah memantau kondisi keseluruhan UMKM Sempol Ayam Ratih.

2. Halaman Riwayat Presensi

Halaman riwayat presensi berfungsi untuk menampilkan data riwayat kehadiran karyawan sebagai luaran (*output*) dari sistem. Halaman ini dirancang agar karyawan dapat melihat informasi kehadiran secara mudah. *Wireframe* halaman riwayat presensi ditunjukkan pada Gambar 4.39.

NO	TANGGAL	MASUK	PULANG	STATUS	KETERLAMBATAN
1	Sabtu, 07 Maret 2026	15:24:15	-	Telat	8 jam 24 menit (874 menit)

Gambar 4.39 *Wireframe* Riwayat Presensi

Rancangan antarmuka halaman riwayat presensi sistem menampilkan data presensi karyawan dalam bentuk tabel yang berisi tanggal presensi, jam masuk, jam pulang, status kehadiran, dan informasi keterlambatan. Selain itu, halaman ini juga menyediakan fitur filter berdasarkan bulan dan tahun untuk mempermudah pencarian data presensi.

Halaman riwayat presensi dilengkapi dengan tombol filter untuk menampilkan data sesuai periode yang dipilih, tombol reset untuk mengembalikan filter ke kondisi awal, serta tombol *export* PDF

yang digunakan untuk mencetak atau menyimpan data riwayat presensi dalam bentuk dokumen PDF.

3. Halaman Data Prestasi Harian

Rancangan antarmuka halaman prestasi harian berfungsi sebagai tampilan luaran (*output*) untuk menyajikan data capaian kinerja harian karyawan yang telah tersimpan di dalam sistem. Halaman ini dirancang agar karyawan dapat melihat hasil prestasi harian serta rekap jumlah adonan yang telah diproduksi pada periode tertentu. *Wireframe* halaman prestasi harian ditunjukkan pada Gambar 4.40.

No	Hari, Tanggal	Jumlah Adonan	Status Masuk	Lama Jam Kerja
 Belum ada data prestasi				

Gambar 4.40 *Wireframe* Prestasi Harian

Rancangan antarmuka halaman prestasi harian sistem menampilkan data prestasi karyawan dalam bentuk tabel yang berisi hari dan tanggal, jumlah adonan, status masuk, dan lama jam kerja. Selain itu, halaman ini juga menampilkan informasi total jumlah adonan bulan Ini sebagai rekapitulasi hasil produksi

karyawan selama satu bulan. Halaman prestasi harian dilengkapi dengan fitur filter berdasarkan bulan dan tahun untuk mempermudah pencarian data prestasi sesuai periode yang diinginkan. Selain itu, tersedia tombol filter untuk menampilkan data berdasarkan periode yang dipilih serta tombol input prestasi hari ini yang digunakan untuk menambahkan data prestasi harian baru ke dalam sistem.

4. Halaman Data Pengajuan Cuti

Rancangan antarmuka halaman data pengajuan cuti digunakan sebagai tampilan *output* untuk menampilkan informasi pengajuan cuti karyawan kepada admin. Halaman ini dirancang agar proses pengelolaan data pengajuan cuti dapat dilakukan secara mudah dan terstruktur. *Desain output Wireframe* halaman data pengajuan cuti ditunjukkan pada Gambar 4.41.

The wireframe shows a web interface for 'Pengajuan Cuti Karyawan'. At the top right, there is a user profile for 'Admin Sempol Ratih' with a dropdown arrow. Below this is a search section with the title 'Pengajuan Cuti Karyawan' and a label 'Cari Nama Karyawan'. It contains a text input field labeled 'Nama karyawan' and a 'Search' button. Below the search section is a table with the following data:

No	Nama	Jenis	Tanggal	Lama	Status	Aksi
1	Sofullah Abi Sodzili	Sakit	2026-05-27 - 2026-05-29	3 hari	Pending	[Icon]

Gambar 4.41 *Wireframe* Pengajuan Cuti

Rancangan antarmuka halaman data pengajuan cuti diatas menampilkan daftar pengajuan cuti karyawan dalam bentuk tabel, sehingga informasi menjadi lebih rapi dan mudah dipahami oleh admin. Halaman ini menampilkan rincian data yang meliputi

nomor urut, nama karyawan, kategori cuti, tanggal pengajuan, lama waktu cuti, dan status persetujuan cuti. Untuk mempermudah pencarian, halaman ini disediakan fitur pencarian nama karyawan untuk mempermudah admin dapat mencari data pengajuan cuti tertentu tanpa harus melihat seluruh data. Setiap data pengajuan cuti dilengkapi dengan tombol aksi. Melalui tombol ini, *admin* dapat melihat rincian informasi pengajuan cuti sekaligus memproses persetujuan (*approval*) atau penolakan terhadap permohonan yang diajukan oleh karyawan.

5. Halaman Data Karyawan

Halaman data karyawan dirancang sebagai antarmuka untuk menampilkan seluruh rekaman data karyawan pada sistem. Tampilan ini memuat fungsi pemantauan, pencarian, dan pengelolaan data karyawan guna memudahkan administrator dalam operasional sistem. *Wireframe* halaman data karyawan dapat dilihat pada gambar 4.42.

NO	KARYAWAN	NIK	TELEPON	ALAMAT	AKSI
1	afdzil	3329090132434354	0834567891865	Jl. Garuda	[Detail] [Edit] [Delete]
2	Sofuloh Abi Sodzil	3329090102050001	087716235351	Jln. Mangun Sarkoro, Limbangan Kulon	[Detail] [Edit] [Delete]

Gambar 4.42 *Wireframe* Data Karyawan

Rancangan halaman data karyawan diatas menyajikan daftar karyawan dalam bentuk tabel yang rapi dan informatif. Tabel tersebut memuat kolom-kolom penting seperti nomor urut, nama karyawan, NIK, nomor telepon, alamat, serta kolom aksi untuk melihat detail, mengedit, dan menghapus data. Pada bagian atas halaman terdapat fitur pencarian berdasarkan nama atau NIK, tombol untuk menambahkan karyawan baru, serta fasilitas *import* data karyawan melalui file *Excel* menggunakan template yang telah disediakan. Desain antarmuka yang ringkas ini memuat tata letak yang jelas, memungkinkan admin mengelola data karyawan secara cepat dan terstruktur.

6. Halaman Lokasi UMKM

Halaman lokasi UMKM dirancang untuk memudahkan admin dalam konfigurasi lokasi dan pengaturan *geofencing* yang

diterapkan pada sistem presensi. Halaman ini memungkinkan admin untuk memantau serta mengelola pengaturan lokasi secara menyeluruh. Rancangan *Wireframe* untuk halaman lokasi UMKM disajikan secara detail pada Gambar 4.43.

The wireframe shows a web interface for 'Pengaturan Geofencing' (Geofencing Settings). At the top right, it indicates the user is 'Admin Sempol Ratih' with an 'ADMIN' role. The main section is titled 'Konfigurasi Lokasi & Radius' (Location & Radius Configuration) with a subtitle 'Tentukan koordinat pusat dengan tampilan satelit agar posisi bangunan lebih jelas.' (Determine the center coordinates with a satellite view so the building position is clearer). A blue button 'Mode Geofencing Aktif' (Geofencing Mode Active) is present. The left sidebar contains input fields for 'KOORDINAT PUSAT' (Center Coordinates) with 'LAT' (-6.8748790) and 'LNG' (101.0562240), sliders for 'PARAMETER RADIUS (METER)' (Radius Parameter in Meters) with 'Radius Masuk' (50m) and 'Radius Pulang' (75m), and a section for 'PENGATURAN JAM ABSEN' (Absence Time Settings) with 'Jam Buka Absen Masuk' (05:00). The right panel shows a 'High-Res Satellite Preview' map with a blue circle representing the geofencing radius and a 'Deteksi Lokasi Baga' (Detect Location Baga) button.

Gambar 4.43 *Wireframe* Lokasi UMKM

Rancangan antarmuka halaman lokasi UMKM diatas menampilkan informasi konfigurasi lokasi dan radius secara lengkap. Pada bagian kiri halaman terdapat pengaturan koordinat pusat yang berisi nilai *latitude* dan *longitude* lokasi UMKM Sempol Ayam Ratih, parameter radius masuk dan radius pulang, serta pengaturan jam presensi. Sementara itu, pada bagian kanan halaman ditampilkan *preview* peta satelit beserta visualisasi lingkaran radius *geofencing* yang mengelilingi lokasi UMKM. Halaman ini juga menunjukkan status mode *geofencing* yang sedang aktif. Dengan tampilan yang informatif dan terstruktur, admin dapat memahami serta menyesuaikan pengaturan lokasi sesuai dengan kebutuhan operasional UMKM.

7. Halaman Buku Kas Umum

Halaman Laporan Buku Kas Umum (BKU) dirancang untuk membantu admin dalam memantau arus kas masuk dan keluar secara keseluruhan. Halaman ini menjadi pusat informasi keuangan UMKM yang menyajikan ringkasan keuangan serta detail transaksi. Rancangan antarmuka untuk halaman buku kas umum disajikan secara detail pada Gambar 4.44.

Buku Kas Umum (BKU)							
Buku Kas Umum Catatan seluruh arus kas masuk & keluar UMKM							
↓ TOTAL PEMASUKAN Rp 2.282.500		↑ TOTAL PENGELUARAN Rp 2.490.000		SALDO RUGI Rp 207.500			
▼ FILTER DATA							
Hari Ini Minggu Ini Bulan Ini Tahun Ini Per Bulan Custom Beri Nama Tampilkan							
NO	TANGGAL	KODE / URAIAN	JENIS	KELUAR	MASUK	SALDO	AKSI
1	7/03/26	Pemasukan transaksi mitra #ORD Hadys Shikhah-44574639 SKU-03080397	Pemasukan Transaksi Mitra	-	100.000	Rp207.500	
2	5/03/26	Pemasukan transaksi mitra #ORD Hadys Shikhah-57221348 SKU-03080388	Pemasukan Transaksi Mitra	-	1.000	Rp207.500	
3	5/03/26	Pemasukan transaksi mitra #ORD Hadys Shikhah-46247968 SKU-03080383	Pemasukan Transaksi Mitra	-	1.000	Rp207.500	
4	5/03/26	Pemasukan transaksi mitra #ORD Hadys Shikhah-46247968 SKU-03080384	Pemasukan Transaksi Mitra	-	1.000	Rp207.500	
5	5/03/26	Pemasukan transaksi mitra #ORD Hadys Shikhah-46247968 SKU-03080383	Pemasukan Transaksi Mitra	-	1.000	Rp207.500	

Gambar 4.44 *Wireframe* Buku Kas Umum

Rancangan halaman Buku Kas Umum pada gambar 4.34 diatas menampilkan informasi keuangan secara ringkas dan jelas. Bagian atas halaman menyajikan tiga ringkasan utama yaitu total pemasukan, total pengeluaran, dan saldo rugi/laba. Di bagian tengah terdapat filter data yang memungkinkan admin memfilter transaksi berdasarkan periode (hari Ini, minggu ini, bulan ini, dll). Bagian bawah halaman menampilkan tabel daftar transaksi yang mencakup nomor urut, tanggal, kode/uraian transaksi, jenis

transaksi, jumlah keluar, jumlah masuk, saldo, serta kolom aksi. Rancangan halaman ini juga dilengkapi dengan tombol export *excel*, *import*, dan entri baru untuk memudahkan pengelolaan data keuangan. *Desain* halaman buku kas umum ini dirancang agar admin dapat dengan mudah memantau kondisi keuangan UMKM Sempol Ayam Ratih dan terstruktur.

8. Halaman Profil Saya

Rancangan antarmuka halaman profil saya digunakan sebagai tampilan output untuk menampilkan informasi identitas karyawan yang tersimpan pada sistem. Halaman ini dirancang agar karyawan dapat melihat data profil pribadi serta melakukan pengelolaan informasi akun apabila diperlukan. *Wireframe* halaman profil saya ditunjukkan pada Gambar 4.45.

Profil Saya

Data Profil

Nama Lengkap

Email

No HP

Alamat

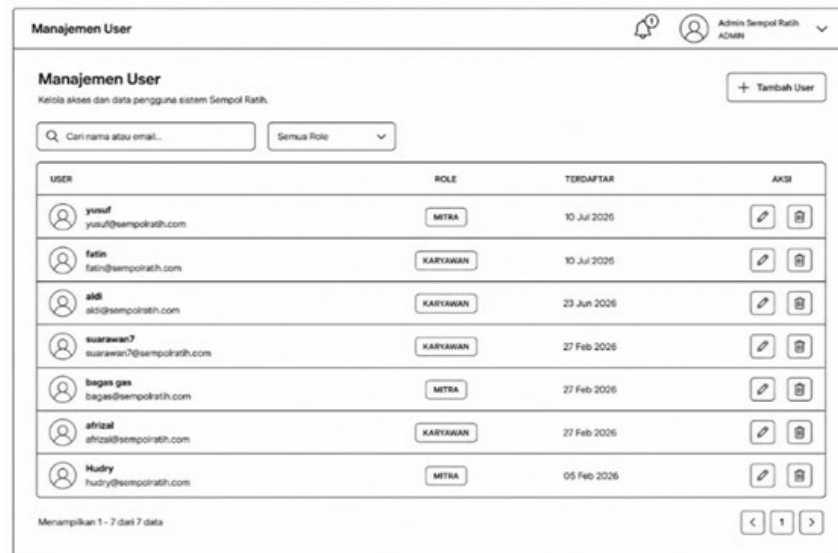
Batal Simpan Profil

Gambar 4.45 *Wireframe* Profil Saya

Rancangan antarmuka halaman profil saya menampilkan informasi identitas karyawan yang terdiri dari foto profil, nama lengkap, email, nomor HP, dan alamat. Seluruh informasi tersebut ditampilkan sebagai data profil yang tersimpan pada sistem sehingga memudahkan karyawan dalam melihat identitas pribadinya. Halaman profil saya juga dilengkapi dengan fitur Ganti Foto untuk memperbarui foto profil, Hapus Foto untuk menghapus foto profil yang telah digunakan, serta tombol Edit Profil yang digunakan untuk mengubah data identitas karyawan apabila terdapat perubahan informasi. Dengan adanya halaman ini, karyawan dapat mengelola data profil secara mandiri sehingga informasi yang tersimpan pada sistem tetap akurat dan selalu diperbarui.

9. Halaman Mengelola Data *User*

Halaman data *user* dirancang sebagai antarmuka *output* yang menampilkan rincian informasi akun pengguna terdaftar di dalam sistem. Halaman ini dirancang agar admin dapat melihat, mengelola, serta melakukan pencarian data pengguna dengan mudah. *Wireframe* halaman data *user* ditunjukkan pada gambar 4.46.



Gambar 4.46 *Wireframe* Mengelola Data User

Dengan Dengan desain antarmuka halaman data user, halaman ini menampilkan daftar akun pengguna dalam bentuk tabel yang berisi nama lengkap, alamat email, peran pengguna, dan informasi akun lainnya. Halaman ini juga dilengkapi fitur pencarian untuk memudahkan admin menemukan data pengguna berdasarkan nama atau email. Selain itu, tersedia tombol tambah *user*, edit, dan hapus untuk mengelola data pengguna. Melalui halaman ini, admin dapat mengatur akun dan hak akses pengguna sesuai dengan perannya.